Statistics: Methods and Applications

By Pawel Lewicki and Thomas Hill

Chapter 1	Elementary Concepts
Chapter 2	Basic Statistics
Chapter 3	ANOVA/MANOVA
Chapter 4	Association Rules
Chapter 5	Boosting Trees
Chapter 6	Canonical Analysis
Chapter 7	CHAID Analysis
Chapter 8	Classification and Regression Trees (C&RT)
Chapter 9	Classification Trees
Chapter 10	Cluster Analysis
Chapter 11	Correspondence Analysis
Chapter 12	Data Mining Techniques
Chapter 13	Discriminant Function Analysis
Chapter 14	Distribution Fitting
Chapter 15	Experimental Design (Industrial DOE)
Chapter 16	Principal Components and Factor Analysis
Chapter 17	General Discriminant Analysis (GDA)
Chapter 18	General Linear Models (GLM)
Chapter 19	Generalized Additive Models (GAM)
Chapter 20	Generalized Linear Models (GLZ)
Chapter 21	General Regression Models (GRM)
Chapter 22	Selected Topics in Graphical Analytic Techniques
Chapter 23	Independent Components Analysis
Chapter 24	Multiple Regression
Chapter 25	Log-Linear Analysis of Frequency Tables
Chapter 26	Multivariate Adaptive Regression Splines (MARSplines)
Chapter 27	Machine Learning

Chapter 28 Multidimensional Scaling
Chapter 29 Neural Networks
Chapter 30 Nonlinear Estimation
Chapter 31 Nonparametric Statistics
Chapter 32 Partial Least Squares (PLS)
Chapter 33 Power Analysis
Chapter 34 Process Analysis
Chapter 35 Quality Control Charts
Chapter 36 Reliability and Item Analysis
Chapter 37 Structural Equation Modeling
Chapter 38 Survival/Failure Time Analysis
Chapter 39 Text Mining
Chapter 40 Time Series Analysis
Chapter 41 Variance Components and Mixed Model ANOVA/ANCOVA
Chapter 42 Distribution Tables
References

Chapter 1
Elementary Concepts in Statistics

Overview of Elementary Concepts in Statistics. In this introduction, we will briefly discuss those elementary statistical concepts that provide the necessary foundations for more specialized expertise in any area of statistical data analysis. The selected topics illustrate the basic assumptions of most statistical methods and/or have been demonstrated in research to be necessary components of one's general understanding of the "quantitative nature" of reality (Nisbett, et al., 1987). Because of space limitations, we will focus mostly on the functional aspects of the concepts discussed and the presentation will be very short. Further information on each of those concepts can be found in statistical textbooks. Recommended introductory textbooks are: Kachigan (1986), and Runyon and Haber (1976); for a more advanced discussion of elementary theory and assumptions of statistics, see the classic books by Hays (1988), and Kendall and Stuart (1979).

- What are variables?
- Correlational vs. experimental research
- Dependent vs. independent variables
- Measurement scales
- Relations between variables
- Why relations between variables are important
- Two basic features of every relation between variables
- What is "statistical significance" (p-value)
- How to determine that a result is "really" significant
- Statistical significance and the number of analyses performed
- Strength vs. reliability of a relation between variables
- Why stronger relations between variables are more significant
- Why significance of a relation between variables depends on the size of the sample
- Example: "Baby boys to baby girls ratio"
- Why small relations can be proven significant only in large samples
- Can "no relation" be a significant result?
- How to measure the magnitude (strength) of relations between variables
- Common "general format" of most statistical tests
- How the "level of statistical significance" is calculated
- Why the "Normal distribution" is important
- Illustration of how the normal distribution is used in statistical reasoning (induction)
- Are all test statistics normally distributed?
- How do we know the consequences of violating the normality assumption?

What are variables. Variables are things that we measure, control, or manipulate in research. They differ in many respects, most notably in the role they are given in our research and in the type of measures that can be applied to them.

Correlational vs. experimental research. Most empirical research belongs clearly to one of those two general categories. In correlational research we do not (or at least try not to) influence any variables but only measure them and look for relations (correlations) between some set of variables, such as blood pressure and cholesterol level. In experimental research, we manipulate some variables and then measure the effects of this manipulation on other variables; for example, a researcher might artificially increase blood pressure and then record cholesterol level. Data analysis in experimental research also comes down to calculating "correlations" between variables, specifically, those manipulated and those affected by the manipulation. However, experimental data may potentially provide qualitatively better information: Only experimental data can conclusively demonstrate causal relations between variables. For example, if we found that whenever we change variable A then variable B changes, then we can conclude that "A influences B." Data from correlational research can only be "interpreted" in causal terms based on some theories that we have, but correlational data cannot conclusively prove causality.

Dependent vs. independent variables. Independent variables are those that are manipulated whereas dependent variables are only measured or registered. This distinction appears terminologically confusing to many because, as some students say, "all variables depend on something." However, once you get used to this distinction, it becomes indispensable. The terms dependent and independent variable apply mostly to experimental research where some variables are manipulated, and in this sense they are "independent" from the initial reaction patterns, features, intentions, etc. of the subjects. Some other variables are expected to be "dependent" on the manipulation or experimental conditions. That is to say, they depend on "what the subject will do" in response. Somewhat contrary to the nature of this distinction, these terms are also used in studies where we do not literally manipulate independent variables, but only assign subjects to "experimental groups" based on some pre-existing properties of the subjects. For example, if in an experiment, males are compared with females regarding their white cell count (WCC), Gender could be called the independent variable and WCC the dependent variable.

Measurement scales. Variables differ in "how well" they can be measured, i.e., in how much measurable information their measurement scale can provide. There is obviously some measurement error involved in every measurement, which determines the "amount of information" that we can obtain. Another factor that determines the amount of information that can be provided by a variable is its "type of measurement scale." Specifically variables are classified as (a) nominal, (b) ordinal, (c) interval or (d) ratio.

a. Nominal variables allow for only qualitative classification. That is, they can be measured only in terms of whether the individual items belong to some distinctively different categories, but we cannot quantify or even rank order those categories. For example, all we can say is that 2 individuals are different in terms of variable A (e.g., they are of different race), but we cannot say which one "has more" of the quality represented by the variable. Typical examples of nominal variables are gender, race, color, city, etc.

b. Ordinal variables allow us to rank order the items we measure in terms of which has less and which has more of the quality represented by the variable, but still they do not allow us to say "how much more." A typical example of an ordinal variable is the socioeconomic status of families. For example, we know that upper-middle is higher than middle but we cannot say that it is, for example, 18% higher. Also this very distinction between nominal, ordinal, and interval scales itself represents a good example of an ordinal variable. For example, we can say that nominal measurement provides less information than ordinal measurement, but we cannot say "how much less" or how this difference compares to the difference between ordinal and interval scales.

c. Interval variables allow us not only to rank order the items that are measured, but also to quantify and compare the sizes of differences between them. For example, temperature, as measured in degrees Fahrenheit or Celsius, constitutes an interval scale. We can say that a temperature of 40 degrees is higher than a temperature of 30 degrees, and that an increase from 20 to 40 degrees is twice as much as an increase from 30 to 40 degrees.

d. Ratio variables are very similar to interval variables; in addition to all the properties of interval variables, they feature an identifiable absolute zero point, thus they allow for statements such as x is two times more than y. Typical examples of ratio scales are measures of time or space. For example, as the Kelvin temperature scale is a ratio scale, not only can we say that a temperature of 200 degrees is higher than one of 100 degrees, we can correctly state that it is twice as high. Interval scales do not have the ratio property. Most statistical data analysis procedures do not distinguish between the interval and ratio properties of the measurement scales.

Relations between variables. Regardless of their type, two or more variables are related if in a sample of observations, the values of those variables are distributed in a consistent manner. In other words, variables are related if their values systematically correspond to each other for these observations. For example, Gender and WCC would be considered to be related if most males had high WCC and most females low WCC, or vice versa; Height is related to Weight because typically tall individuals are heavier than short ones; IQ is related to the Number of Errors in a test, if people with higher IQ's make fewer errors.

Why relations between variables are important. Generally speaking, the ultimate goal of every research or scientific analysis is finding relations between variables. The philosophy of science teaches us that there is no other way of representing "meaning" except in terms of relations between some quantities or qualities; either way involves relations between variables. Thus, the advancement of science must always involve finding new relations between variables. Correlational research involves measuring such relations in the most straightforward manner. However, experimental research is not any different in this respect. For example, the above mentioned experiment comparing WCC in males and females can be described as looking for a correlation between two variables: Gender and WCC. Statistics does nothing else but help us evaluate relations between variables. Actually, all of the hundreds of procedures that are described in this manual can be interpreted in terms of evaluating various kinds of inter-variable relations.

Two basic features of every relation between variables. The two most elementary formal properties of every relation between variables are the relation's (a) magnitude (or "size") and (b) its reliability (or "truthfulness").

a. Magnitude (or "size"). The magnitude is much easier to understand and measure than reliability. For example, if every male in our sample was found to have a higher WCC than any female in the sample, we could say that the magnitude of the relation between the two variables (Gender and WCC) is very high in our sample. In other words, we could predict one based on the other (at least among the members of our sample).
b. Reliability (or "truthfulness"). The reliability of a relation is a much less intuitive concept, but still extremely important. It pertains to the "representativeness" of the result found in our specific sample for the entire population. In other words, it says how probable it is that a similar relation would be found if the experiment was replicated with other samples drawn from the same population. Remember that we are almost never "ultimately" interested only in what is going on in our sample; we are interested in the sample only to the extent it can provide information about the population. If our study meets some specific criteria (to be mentioned later), then the reliability of a relation between variables observed in our sample can be quantitatively estimated and represented using a standard measure (technically called p-value or statistical significance level, see the next paragraph).

What is "statistical significance" (p-value). The statistical significance of a result is the probability that the observed relationship (e.g., between variables) or a difference (e.g., between means) in a sample occurred by pure chance ("luck of the draw"), and that in the population from which the sample was drawn, no such relationship or differences exist. Using less technical terms, one could say that the statistical significance of a result tells us something about the degree to which the result is "true" (in the sense of being "representative of the population"). More technically, the value of the p-value represents a decreasing index of the reliability of a result (see Brownlee, 1960). The higher the p-value, the less we can believe that the observed relation between variables in the sample is a reliable indicator of the relation between the respective variables in the population. Specifically, the p-value represents the probability of error that is involved in accepting our observed result as valid, that is, as "representative of the population." For example, a p-value of .05 (i.e.,1/20) indicates that there is a 5% probability that the relation between the variables found in our sample is a "fluke." In other words, assuming that in the population there was no relation between those variables whatsoever, and we were repeating experiments like ours one after another, we could expect that approximately in every 20 replications of the experiment there would be one in which the relation between the variables in question would be equal or stronger than in ours. (Note that this is not the same as saying that, given that there IS a relationship between the variables, we can expect to replicate the results 5% of the time or 95% of the time; when there is a relationship between the variables in the population, the probability of replicating the study and finding that relationship is related to the statistical power of the design. See also, Power Analysis). In many areas of research, the p-value of .05 is customarily treated as a "border-line acceptable" error level.

How to determine that a result is "really" significant. There is no way to avoid arbitrariness in the final decision as to what level of significance will be treated as really "significant." That is, the selection of some level of significance, up to which the results will be rejected as invalid, is arbitrary. In practice, the final decision usually depends on whether the outcome was predicted a priori or only found post hoc in the course of many analyses and comparisons performed on the data set, on the total amount of consistent supportive evidence in the entire data set, and on "traditions" existing in the particular area of research. Typically, in many sciences, results that yield $p \leq .05$ are considered borderline statistically significant but remember that this level of significance still involves a pretty high probability of error (5%). Results that are significant at the $p \leq .01$ level are commonly considered statistically significant, and $p \leq .005$ or $p \leq .001$ levels are often called "highly" significant. But remember that those classifications represent nothing else but arbitrary conventions that are only informally based on general research experience.

Statistical significance and the number of analyses performed. Needless to say, the more analyses you perform on a data set, the more results will meet "by chance" the conventional significance level. For example, if you calculate correlations between ten variables (i.e., 45 different correlation coefficients), then you should expect to find by chance that about two (i.e., one in every 20) correlation coefficients are significant at the $p \leq .05$ level, even if the values of the variables were totally random and those variables do not correlate in the population. Some statistical methods that involve many comparisons, and thus a good chance for such errors, include some "correction" or adjustment for the total number of comparisons. However, many statistical methods (especially simple exploratory data analyses) do not offer any straightforward remedies to this problem. Therefore, it is up to the researcher to carefully evaluate the reliability of unexpected findings. Many examples in this manual offer specific advice on how to do this; relevant information can also be found in most research methods textbooks.

Strength vs. reliability of a relation between variables. We said before that strength and reliability are two different features of relationships between variables. However, they are not totally independent. In general, in a sample of a particular size, the larger the magnitude of the relation between variables, the more reliable the relation (see the next paragraph).

Why stronger relations between variables are more significant. Assuming that there is no relation between the respective variables in the population, the most likely outcome would be also finding no relation between those variables in the research sample. Thus, the stronger the relation found in the sample, the less likely it is that there is no corresponding relation in the population. As you see, the magnitude and significance of a relation appear to be closely related, and we could calculate the significance from the magnitude and vice-versa; however, this is true only if the sample size is kept constant, because the relation of a given strength could be either highly significant or not significant at all, depending on the sample size (see the next paragraph).

<>

Why significance of a relation between variables depends on the size of the sample. If there are very few observations, then there are also respectively few possible combinations of the values of the variables, and thus the probability of obtaining by chance a combination of those values indicative of a strong relation is relatively high. Consider the following illustration. If we are interested in two variables (Gender: male/female and WCC: high/low) and there are only four subjects in our sample (two males and two females), then the probability that we will find, purely by chance, a 100% relation between the two variables can be as high as one-eighth. Specifically, there is a one-in-eight chance that both males will have a high WCC and both females a low WCC, or vice versa. Now consider the probability of obtaining such a perfect match by chance if our sample consisted of 100 subjects; the probability of obtaining such an outcome by chance would be practically zero. Let's look at a more general example. Imagine a theoretical population in which the average value of WCC in males and females is exactly the same. Needless to say, if we start replicating a simple experiment by drawing pairs of samples (of males and females) of a particular size from this population and calculating the difference between the average WCC in each pair of samples, most of the experiments will yield results close to 0. However, from time to time, a pair of samples will be drawn where the difference between males and females will be quite different from 0. How often will it happen? The smaller the sample size in each experiment, the more likely it is that we will obtain such erroneous results, which in this case would be results indicative of the existence of a relation between gender and WCC obtained from a population in which such a relation does not exist.

Example. "Baby boys to baby girls ratio." Consider the following example from research on statistical reasoning (Nisbett, et al., 1987). There are two hospitals: in the first one, 120 babies are born every day, in the other, only 12. On average, the ratio of baby boys to baby girls born every day in each hospital is 50/50. However, one day, in one of those hospitals twice as many baby girls were born as baby boys. In which hospital was it more likely to happen? The answer is obvious for a statistician, but as research shows, not so obvious for a lay person: It is much more likely to happen in the small hospital. The reason for this is that technically speaking, the probability of a random deviation of a particular size (from the population mean), decreases with the increase in the sample size.

Why small relations can be proven significant only in large samples. The examples in the previous paragraphs indicate that if a relationship between variables in question is "objectively" (i.e., in the population) small, then there is no way to identify such a relation in a study unless the research sample is correspondingly large. Even if our sample is in fact "perfectly representative" the effect will not be statistically significant if the sample is small. Analogously, if a relation in question is "objectively" very large (i.e., in the population), then it can be found to be highly significant even in a study based on a very small sample. Consider the following additional illustration. If a coin is slightly asymmetrical, and when tossed is somewhat more likely to produce heads than tails (e.g., 60% vs. 40%), then ten tosses would not be sufficient to convince anyone that the coin is asymmetrical, even if the outcome obtained (six heads and four tails) was perfectly representative of the bias of the coin. However, is it so that 10 tosses is not enough to prove anything? No, if the effect in question were large enough, then ten tosses could be quite enough. For instance, imagine now that the coin is so asymmetrical that no matter how you toss it, the outcome will be heads. If you tossed such a coin ten times and each toss produced heads, most people would consider it sufficient evidence that something is "wrong" with the coin. In other words, it would be considered convincing evidence that in the theoretical population of an infinite number of tosses of this coin there would be more heads than tails. Thus, if a relation is large, then it can be found to be significant even in a small sample.

Can "no relation" be a significant result? The smaller the relation between variables, the larger the sample size that is necessary to prove it significant. For example, imagine how many tosses would be necessary to prove that a coin is asymmetrical if its bias were only .000001%! Thus, the necessary minimum sample size increases as the magnitude of the effect to be demonstrated decreases. When the magnitude of the effect approaches 0, the necessary sample size to conclusively prove it approaches infinity. That is to say, if there is almost no relation between two variables, then the sample size must be almost equal to the population size, which is assumed to be infinitely large. Statistical significance represents the probability that a similar outcome would be obtained if we tested the entire population. Thus, everything that would be found after testing the entire population would be, by definition, significant at the highest possible level, and this also includes all "no relation" results.

How to measure the magnitude (strength) of relations between variables. There are very many measures of the magnitude of relationships between variables which have been developed by statisticians; the choice of a specific measure in given circumstances depends on the number of variables involved, measurement scales used, nature of the relations, etc. Almost all of them, however, follow one general principle: they attempt to somehow evaluate the observed relation by comparing it to the "maximum imaginable relation" between those specific variables. Technically speaking, a common way to perform such evaluations is to look at how differentiated are the values of the variables, and then calculate what part of this "overall available differentiation" is accounted for by instances when that differentiation is "common" in the two (or more) variables in question. Speaking less technically, we compare "what is common in those variables" to "what potentially could have been common if the variables were perfectly related." Let us consider a simple illustration. Let us say that in our sample, the average index of WCC is 100 in males and 102 in females. Thus, we could say that on average, the deviation of each individual score from the grand mean (101) contains a component due to the gender of the subject; the size of this component is 1. That value, in a sense, represents some measure of relation between Gender and WCC. However, this value is a very poor measure, because it does not tell us how relatively large this component is, given the "overall differentiation" of WCC scores. Consider two extreme possibilities:

a. If all WCC scores of males were equal exactly to 100, and those of females equal to 102, then all deviations from the grand mean in our sample would be entirely accounted for by gender. We would say that in our sample, gender is perfectly correlated with WCC, that is, 100% of the observed differences between subjects regarding their WCC is accounted for by their gender.
b. If WCC scores were in the range of 0-1000, the same difference (of 2) between the average WCC of males and females found in the study would account for such a small part of the overall differentiation of scores that most likely it would be considered negligible. For example, one more subject taken into account could change, or even reverse the direction of the difference. Therefore, every good measure of relations between variables must take into account the overall differentiation of individual scores in the sample and evaluate the relation in terms of (relatively) how much of this differentiation is accounted for by the relation in question.

Common "general format" of most statistical tests. Because the ultimate goal of most statistical tests is to evaluate relations between variables, most statistical tests follow the general format that was explained in the previous paragraph. Technically speaking, they represent a ratio of some measure of the differentiation common in the variables in question to the overall differentiation of those variables. For example, they represent a ratio of the part of the overall differentiation of the WCC scores that can be accounted for by gender to the overall differentiation of the WCC scores. This ratio is usually called a ratio of explained variation to total variation. In statistics, the term explained variation does not necessarily imply that we "conceptually understand" it. It is used only to denote the common variation in the variables in question, that is, the part of variation in one variable that is "explained" by the specific values of the other variable, and vice versa.

How the "level of statistical significance" is calculated. Let us assume that we have already calculated a measure of a relation between two variables (as explained above). The next question is "how significant is this relation?" For example, is 40% of the explained variance between the two variables enough to consider the relation significant? The answer is "it depends." Specifically, the significance depends mostly on the sample size. As explained before, in very large samples, even very small relations between variables will be significant, whereas in very small samples even very large relations cannot be considered reliable (significant). Thus, in order to determine the level of statistical significance, we need a function that represents the relationship between "magnitude" and "significance" of relations between two variables, depending on the sample size. The function we need would tell us exactly "how likely it is to obtain a relation of a given magnitude (or larger) from a sample of a given size, assuming that there is no such relation between those variables in the population." In other words, that function would give us the significance (p) level, and it would tell us the probability of error involved in rejecting the idea that the relation in question does not exist in the population. This "alternative" hypothesis (that there is no relation in the population) is usually called the null hypothesis. It would be ideal if the probability function was linear, and for example, only had different slopes for different sample sizes. Unfortunately, the function is more complex, and is not always exactly the same; however, in most cases we know its shape and can use it to determine the significance levels for our findings in samples of a particular size. Most of those functions are related to a general type of function which is called normal.

Why the "Normal distribution" is important. The "Normal distribution" is important because in most cases, it well approximates the function that was introduced in the previous paragraph (for a detailed illustration, see Are all test statistics normally distributed?). The distribution of many test statistics is normal or follows some form that can be derived from the normal distribution. In this sense, philosophically speaking, the Normal distribution represents one of the empirically verified elementary "truths about the general nature of reality," and its status can be compared to the one of fundamental laws of natural sciences. The exact shape of the normal distribution (the characteristic "bell curve") is defined by a function which has only two parameters: mean and standard deviation.

A characteristic property of the Normal distribution is that 68% of all of its observations fall within a range of ±1 standard deviation from the mean, and a range of ±2 standard deviations includes 95% of the scores. In other words, in a Normal distribution, observations that have a standardized value of less than -2 or more than +2 have a relative frequency of 5% or less. (Standardized value means that a value is expressed in terms of its difference from the mean, divided by the standard deviation.) If you have access to *STATISTICA*, you can explore the exact values of probability associated with different values in the normal distribution using the interactive Probability Calculator tool; for example, if you enter the Z value (i.e., standardized value) of 4, the associated probability computed by *STATISTICA* will be less than .0001, because in the normal distribution almost all observations (i.e., more than 99.99%) fall within the range of ±4 standard deviations. The animation below shows the tail area associated with other Z values.

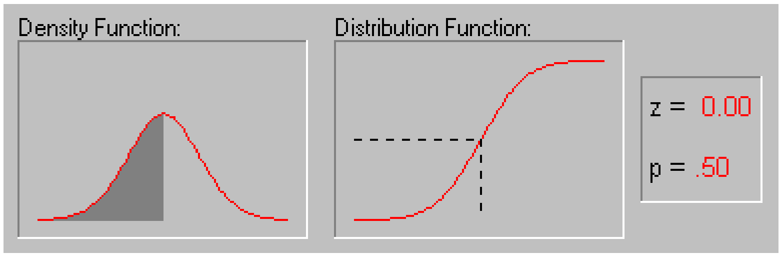
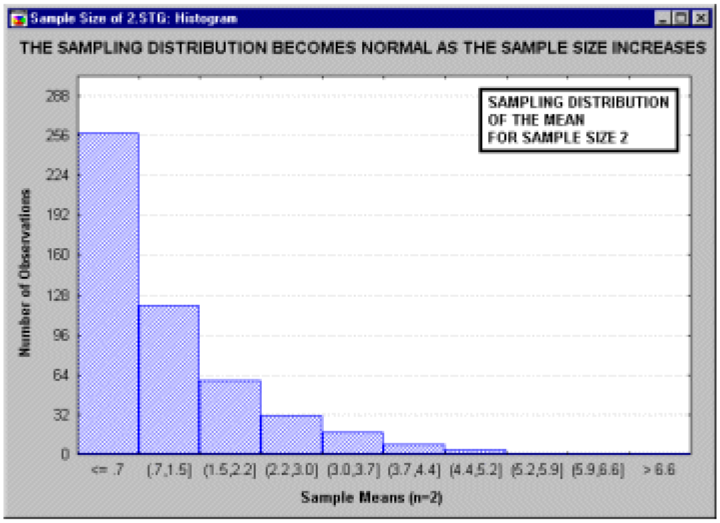


Illustration of how the normal distribution is used in statistical reasoning (induction).

Recall the example discussed above, where pairs of samples of males and females were drawn from a population in which the average value of WCC in males and females was exactly the same. Although the most likely outcome of such experiments (one pair of samples per experiment) was that the difference between the average WCC in males and females in each pair is close to zero, from time to time, a pair of samples will be drawn where the difference between males and females is quite different from 0. How often does it happen? If the sample size is large enough, the results of such replications are "normally distributed" (this important principle is explained and illustrated in the next paragraph), and thus knowing the shape of the normal curve, we can precisely calculate the probability of obtaining "by chance" outcomes representing various levels of deviation from the hypothetical population mean of 0. If such a calculated probability is so low that it meets the previously accepted criterion of statistical significance, then we have only one choice: conclude that our result gives a better approximation of what is going on in the population than the "null hypothesis" (remember that the null hypothesis was considered only for "technical reasons" as a benchmark against which our empirical result was evaluated). Note that this entire reasoning is based on the assumption that the shape of the distribution of those "replications" (technically, the "sampling distribution") is normal. This assumption is discussed in the next paragraph.

Are all test statistics normally distributed?

Not all, but most of them are either based on the normal distribution directly or on distributions that are related to, and can be derived from normal, such as t, F, or Chi-square. Typically, those tests require that the variables analyzed are themselves normally distributed in the population, that is, they meet the so-called "normality assumption." Many observed variables actually are normally distributed, which is another reason why the normal distribution represents a "general feature" of empirical reality. The problem may occur when one tries to use a normal distribution-based test to analyze data from variables that are themselves not normally distributed (see tests of normality in Nonparametrics or ANOVA/MANOVA). In such cases we have two general choices. First, we can use some alternative "nonparametric" test (or so-called "distribution-free test" see, Nonparametrics); but this is often inconvenient because such tests are typically less powerful and less flexible in terms of types of conclusions that they can provide. Alternatively, in many cases we can still use the normal distribution-based test if we only make sure that the size of our samples is large enough. The latter option is based on an extremely important principle which is largely responsible for the popularity of tests that are based on the normal function. Namely, as the sample size increases, the shape of the sampling distribution (i.e., distribution of a statistic from the sample; this term was first used by Fisher, 1928a) approaches normal shape, even if the distribution of the variable in question is not normal. This principle is illustrated in the following animation showing a series of sampling distributions (created with gradually increasing sample sizes of: 2, 5, 10, 15, and 30) using a variable that is clearly non-normal in the population, that is, the distribution of its values is clearly skewed.



However, as the sample size (of samples used to create the sampling distribution of the mean) increases, the shape of the sampling distribution becomes normal. Note that for n=30, the shape of that distribution is "almost" perfectly normal (see the close match of the fit). This principle is called the central limit theorem (this term was first used by Pólya, 1920; German, "Zentraler Grenzwertsatz").

How do we know the consequences of violating the normality assumption?

Although many of the statements made in the preceding paragraphs can be proven mathematically, some of them do not have theoretical proofs and can be demonstrated only empirically, via so-called Monte-Carlo experiments. In these experiments, large numbers of samples are generated by a computer following predesigned specifications and the results from such samples are analyzed using a variety of tests. This way we can empirically evaluate the type and magnitude of errors or biases to which we are exposed when certain theoretical assumptions of the tests we are using are not met by our data. Specifically, Monte-Carlo studies were used extensively with normal distribution-based tests to determine how sensitive they are to violations of the assumption of normal distribution of the analyzed variables in the population. The general conclusion from these studies is that the consequences of such violations are less severe than previously thought. Although these conclusions should not entirely discourage anyone from being concerned about the normality assumption, they have increased the overall popularity of the distribution-dependent statistical tests in all areas of research.

Chapter 2
Basic Statistics

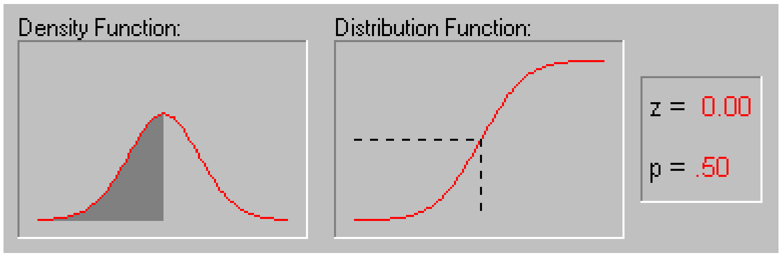
- Descriptive statistics
  - "True" Mean and Confidence Interval
  - Shape of the Distribution, Normality
- Correlations
  - Purpose (What is Correlation?)
  - Simple Linear Correlation (Pearson r)
  - How to Interpret the Values of Correlations
  - Significance of Correlations
  - Outliers
  - Quantitative Approach to Outliers
  - Correlations in Non-homogeneous Groups
  - Nonlinear Relations between Variables
  - Measuring Nonlinear Relations
  - Exploratory Examination of Correlation Matrices
  - Casewise vs. Pairwise Deletion of Missing Data
  - How to Identify Biases Caused by the Bias due to Pairwise Deletion of Missing Data
  - Pairwise Deletion of Missing Data vs. Mean Substitution
  - Spurious Correlations
  - Are correlation coefficients "additive?"
  - How to Determine Whether Two Correlation Coefficients are Significant
- t-test for independent samples
  - Purpose, Assumptions
  - Arrangement of Data
  - t-test graphs
  - More Complex Group Comparisons
- t-test for dependent samples
  - Within-group Variation
  - Purpose
  - Assumptions
  - Arrangement of Data
  - Matrices of t-tests
  - More Complex Group Comparisons
- Breakdown: Descriptive statistics by groups
  - Purpose
  - Arrangement of Data
  - Statistical Tests in Breakdowns
  - Other Related Data Analysis Techniques
  - Post-Hoc Comparisons of Means
  - Breakdowns vs. Discriminant Function Analysis
  - Breakdowns vs. Frequency Tables

  - Graphical breakdowns
- Frequency tables
  - Purpose
  - Applications
- Crosstabulation and stub-and-banner tables
  - Purpose and Arrangement of Table
  - 2x2 Table
  - Marginal Frequencies
  - Column, Row, and Total Percentages
  - Graphical Representations of Crosstabulations
  - Stub-and-Banner Tables
  - Interpreting the Banner Table
  - Multi-way Tables with Control Variables
  - Graphical Representations of Multi-way Tables
  - Statistics in crosstabulation tables
  - Multiple responses/dichotomies

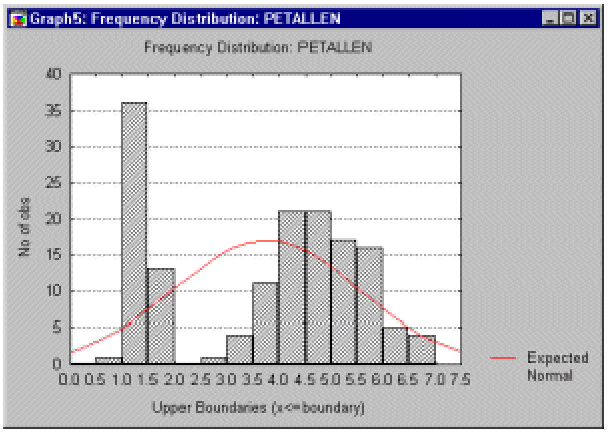
Descriptive Statistics

"True" Mean and Confidence Interval. Probably the most often used descriptive statistic is the mean. The mean is a particularly informative measure of the "central tendency" of the variable if it is reported along with its confidence intervals. As mentioned earlier, usually we are interested in statistics (such as the mean) from our sample only to the extent to which they can infer information about the population. The *confidence intervals* for the mean give us a range of values around the mean where we expect the "true" (population) mean is located (with a given level of certainty, see also *Elementary Concepts*). For example, if the mean in your sample is 23, and the lower and upper limits of the *p*=.05 confidence interval are 19 and 27 respectively, then you can conclude that there is a 95% probability that the population mean is greater than 19 and lower than 27. If you set the *p*-level to a smaller value, then the interval would become wider thereby increasing the "certainty" of the estimate, and vice versa; as we all know from the weather forecast, the more "vague" the prediction (i.e., wider the confidence interval), the more likely it will materialize. Note that the width of the confidence interval depends on the sample size and on the variation of data values. The larger the sample size, the more reliable its mean. The larger the variation, the less reliable the mean (see also *Elementary Concepts*). The calculation of confidence intervals is based on the assumption that the variable is normally distributed in the population. The estimate may not be valid if this assumption is not met, unless the sample size is large, say *n*=100 or more.

Shape of the Distribution, Normality. An important aspect of the "description" of a variable is the shape of its distribution, which tells you the frequency of values from different ranges of the variable. Typically, a researcher is interested in how well the distribution can be approximated by the normal distribution (see the animation below for an example of this distribution) (see also *Elementary Concepts*). Simple descriptive statistics can provide some information relevant to this issue. For example, if the *skewness* (which measures the deviation of the distribution from symmetry) is clearly different from 0, then that distribution is asymmetrical, while normal distributions are perfectly symmetrical. If the *kurtosis* (which measures "peakedness" of the distribution) is clearly different from 0, then the distribution is either flatter or more peaked than normal; the kurtosis of the normal distribution is 0.



More precise information can be obtained by performing one of the *tests of normality* to determine the probability that the sample came from a normally distributed population of observations (e.g., the so-called Kolmogorov-Smirnov test, or the Shapiro-Wilks' W test. However, none of these tests can entirely substitute for a visual examination of the data using a histogram (i.e., a graph that shows the frequency distribution of a variable).

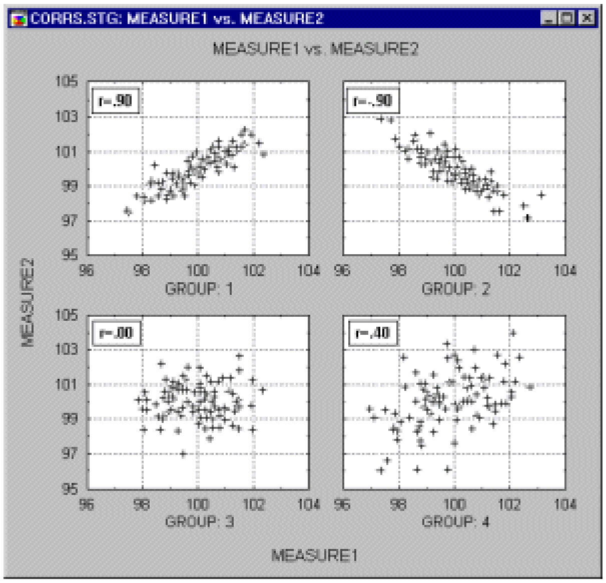


The graph allows you to evaluate the normality of the empirical distribution because it also shows the normal curve superimposed over the histogram. It also allows you to examine various aspects of the distribution *qualitatively*. For example, the distribution could be bimodal (have 2 peaks). This might suggest that the sample is not homogeneous but possibly its elements came from two different populations, each more or less normally distributed. In such cases, in order to understand the nature of the variable in question, you should look for a way to quantitatively identify the two sub-samples.

Correlations

Purpose (What is Correlation?)

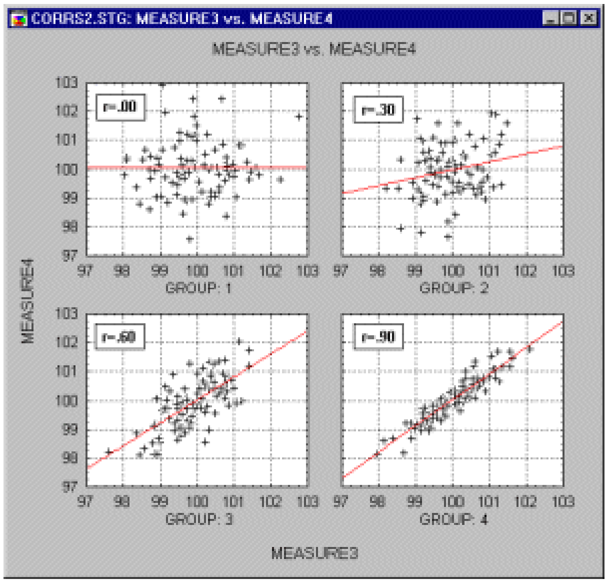
Correlation is a measure of the relation between two or more variables. The measurement scales used should be at least interval scales, but other correlation coefficients are available to handle other types of data. Correlation coefficients can range from -1.00 to +1.00. The value of -1.00 represents a perfect *negative* correlation while a value of +1.00 represents a perfect *positive* correlation. A value of 0.00 represents a lack of correlation.



The most widely-used type of correlation coefficient is *Pearson r*, also called *linear* or *product- moment* correlation.

Simple Linear Correlation (Pearson r).

Pearson correlation (hereafter called *correlation*), assumes that the two variables are measured on at least interval scales (see *Elementary Concepts*), and it determines the extent to which values of the two variables are "proportional" to each other. The value of correlation (i.e., correlation coefficient) does not depend on the specific measurement units used; for example, the correlation between height and weight will be identical regardless of whether *inches* and *pounds*, or *centimeters* and *kilograms* are used as measurement units. *Proportional* means *linearly related*; that is, the correlation is high if it can be "summarized" by a straight line (sloped upwards or downwards).

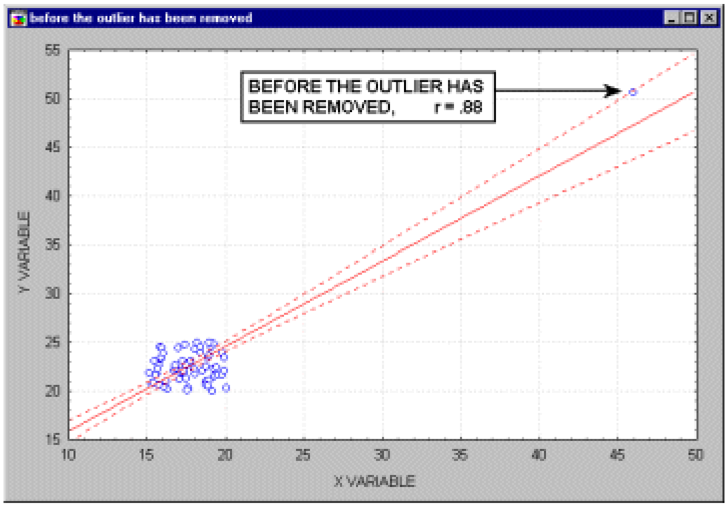


This line is called the *regression line* or *least squares line*, because it is determined such that the sum of the *squared* distances of all the data points from the line is the lowest possible. Note that the concept of *squared* distances will have important functional consequences on how the value of the correlation coefficient reacts to various specific arrangements of data (as we will later see).

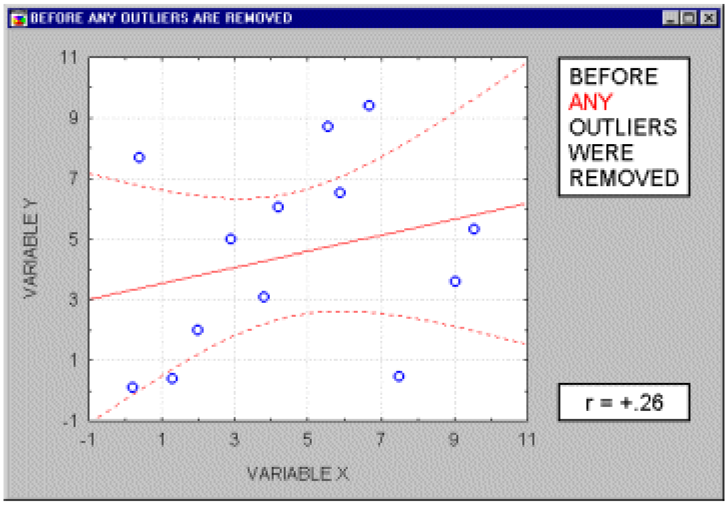
How to Interpret the Values of Correlations. As mentioned before, the correlation coefficient (r) represents the linear relationship between two variables. If the correlation coefficient is squared, then the resulting value (r^2, the coefficient of determination) will represent the proportion of common variation in the two variables (i.e., the "strength" or "magnitude" of the relationship). In order to evaluate the correlation between variables, it is important to know this "magnitude" or "strength" as well as the *significance* of the correlation.

Significance of Correlations. The significance level calculated for each correlation is a primary source of information about the reliability of the correlation. As explained before (see *Elementary Concepts*), the significance of a correlation coefficient of a particular magnitude will change depending on the size of the sample from which it was computed. The test of significance is based on the assumption that the distribution of the residual values (i.e., the deviations from the regression line) for the dependent variable *y* follows the normal distribution, and that the variability of the residual values is the same for all values of the independent variable *x*. However, Monte Carlo studies suggest that meeting those assumptions closely is not absolutely crucial if your sample size is not very small and when the departure from normality is not very large. It is impossible to formulate precise recommendations based on those Monte- Carlo results, but many researchers follow a rule of thumb that if your sample size is 50 or more then serious biases are unlikely, and if your sample size is over 100 then you should not be concerned at all with the normality assumptions. There are, however, much more common and serious threats to the validity of information that a correlation coefficient can provide; they are briefly discussed in the following paragraphs.

Outliers. Outliers are atypical (by definition), infrequent observations. Because of the way in which the regression line is determined (especially the fact that it is based on minimizing not the sum of simple distances but the sum of *squares of distances* of data points from the line), outliers have a profound influence on the slope of the regression line and consequently on the value of the correlation coefficient. A single outlier is capable of considerably changing the slope of the regression line and, consequently, the value of the correlation, as demonstrated in the following example. Note, that as shown on that illustration, just one outlier can be entirely responsible for a high value of the correlation that otherwise (without the outlier) would be close to zero. Needless to say, one should never base important conclusions on the value of the correlation coefficient alone (i.e., examining the respective scatterplot is always recommended).



Note that if the sample size is relatively small, then including or excluding specific data points that are not as clearly "outliers" as the one shown in the previous example may have a profound influence on the regression line (and the correlation coefficient). This is illustrated in the following example where we call the points being excluded "outliers;" one may argue, however, that they are not outliers but rather extreme values.

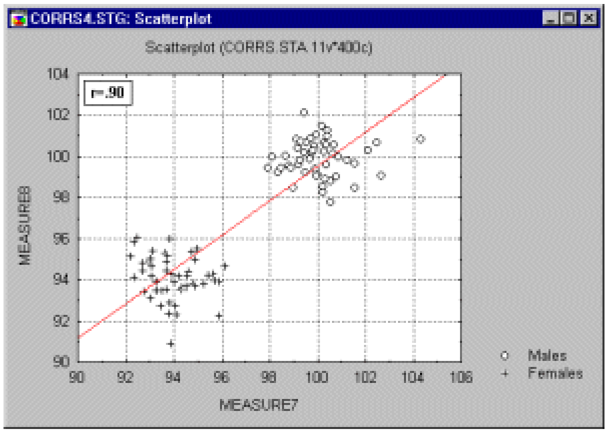


Typically, we believe that outliers represent a random error that we would like to be able to control. Unfortunately, there is no widely accepted method to remove outliers automatically (however, see the next paragraph), thus what we are left with is to identify any outliers by examining a *scatterplot* of each important correlation. Needless to say, outliers may not only artificially increase the value of a correlation coefficient, but they can also decrease the value of a "legitimate" correlation.

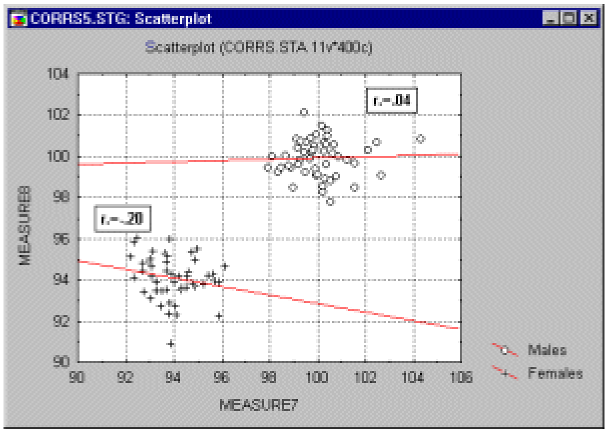
See also Confidence Ellipse.

Quantitative Approach to Outliers. Some researchers use quantitative methods to exclude outliers. For example, they exclude observations that are outside the range of ±2 standard deviations (or even ±1.5 sd's) around the group or design cell mean. In some areas of research, such "cleaning" of the data is absolutely necessary. For example, in cognitive psychology research on reaction times, even if almost all scores in an experiment are in the range of 300-700 *milliseconds*, just a few "distracted reactions" of 10-15 *seconds* will completely change the overall picture. Unfortunately, defining an outlier is subjective (as it should be), and the decisions concerning how to identify them must be made on an individual basis (taking into account specific experimental paradigms and/or "accepted practice" and general research experience in the respective area). It should also be noted that in some rare cases, the relative frequency of outliers across a number of groups or cells of a design can be subjected to analysis and provide interpretable results. For example, outliers could be indicative of the occurrence of a phenomenon that is qualitatively different than the typical pattern observed or expected in the sample, thus the relative frequency of outliers could provide evidence of a relative frequency of departure from the process or phenomenon that is typical for the majority of cases in a group. See also Confidence Ellipse.

Correlations in Non-homogeneous Groups. A lack of homogeneity in the sample from which a correlation was calculated can be another factor that biases the value of the correlation. Imagine a case where a correlation coefficient is calculated from data points which came from two different experimental groups but this fact is ignored when the correlation is calculated. Let us assume that the experimental manipulation in one of the groups increased the values of both correlated variables and thus the data from each group form a distinctive "cloud" in the scatterplot (as shown in the graph below).

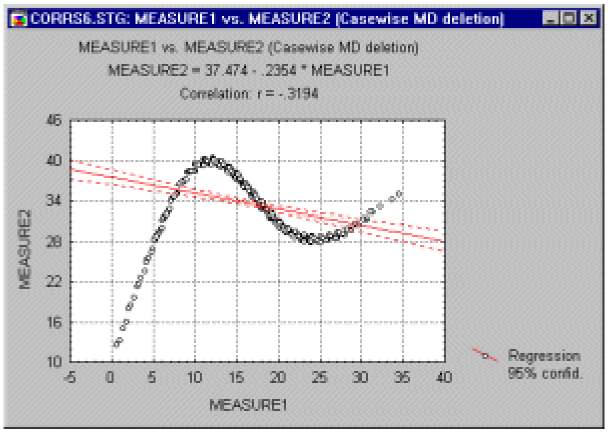


In such cases, a high correlation may result that is entirely due to the arrangement of the two groups, but which does not represent the "true" relation between the two variables, which may practically be equal to 0 (as could be seen if we looked at each group separately, see the following graph).



If you suspect the influence of such a phenomenon on your correlations and know how to identify such "subsets" of data, try to run the correlations separately in each subset of observations. If you do not know how to identify the hypothetical subsets, try to examine the data with some exploratory multivariate techniques (e.g., Cluster Analysis).

Nonlinear Relations between Variables. Another potential source of problems with the linear (*Pearson r*) correlation is the shape of the relation. As mentioned before, *Pearson r* measures a relation between two variables only to the extent to which it is linear; deviations from linearity will increase the total sum of squared distances from the regression line even if they represent a "true" and very close relationship between two variables. The possibility of such non-linear relationships is another reason why examining scatterplots is a necessary step in evaluating every correlation. For example, the following graph demonstrates an extremely strong correlation between the two variables which is not well described by the linear function.



Measuring Nonlinear Relations. What do you do if a correlation is strong but clearly nonlinear (as concluded from examining scatterplots)? Unfortunately, there is no simple answer to this question, because there is no easy-to-use equivalent of *Pearson r* that is capable of handling nonlinear relations. If the curve is monotonous (continuously decreasing or increasing) you could try to transform one or both of the variables to remove the curvilinearity and then recalculate the correlation. For example, a typical transformation used in such cases is the logarithmic function which will "squeeze" together the values at one end of the range. Another option available if the relation is monotonous is to try a nonparametric correlation (e.g., Spearman *R*, see *Nonparametrics and Distribution Fitting*) which is sensitive only to the ordinal arrangement of values, thus, by definition, it ignores monotonous curvilinearity. However, nonparametric correlations are generally less sensitive and sometimes this method will not produce any gains. Unfortunately, the two most precise methods are not easy to use and require a good deal of "experimentation" with the data. Therefore you could:

A. Try to identify the specific function that best describes the curve. After a function has been found, you can test its "goodness-of-fit" to your data.
B. Alternatively, you could experiment with dividing one of the variables into a number of segments (e.g., 4 or 5) of an equal width, treat this new variable as a grouping variable and run an analysis of variance on the data.

Exploratory Examination of Correlation Matrices. A common first step of many data analyses that involve more than a very few variables is to run a correlation matrix of all variables and then examine it for expected (and unexpected) significant relations. When this is done, you need to be aware of the general nature of statistical significance (see *Elementary Concepts*); specifically, if you run many tests (in this case, many correlations), then significant results will be found "surprisingly often" due to pure chance. For example, by definition, a coefficient significant at the .05 level will occur by chance once in every 20 coefficients. There is no "automatic" way to weed out the "true" correlations. Thus, you should treat all results that were not predicted or planned with particular caution and look for their consistency with other results; ultimately, though, the most conclusive (although costly) control for such a randomness factor is to replicate the study. This issue is general and it pertains to all analyses that involve "multiple comparisons and statistical significance." This problem is also briefly discussed in the context of *post-hoc comparisons of means* and the Breakdowns option.

Casewise vs. Pairwise Deletion of Missing Data. The default way of deleting missing data while calculating a correlation matrix is to exclude all cases that have missing data in at least one of the selected variables; that is, by *casewise deletion* of missing data. Only this way will you get a "true" correlation matrix, where all correlations are obtained from the *same* set of observations. However, if missing data are randomly distributed across cases, you could easily end up with no "valid" cases in the data set, because each of them will have at least one missing data in some variable. The most common solution used in such instances is to use so-called *pairwise deletion* of missing data in correlation matrices, where a correlation between each pair of variables is calculated from all cases that have valid data on those two variables. In many instances there is nothing wrong with that method, especially when the total percentage of missing data is low, say 10%, and they are relatively randomly distributed between cases and variables. However, it may sometimes lead to serious problems.

For example, a systematic bias may result from a "hidden" systematic distribution of missing data, causing different correlation coefficients in the same correlation matrix to be based on different subsets of subjects. In addition to the possibly biased conclusions that you could derive from such "pairwise calculated" correlation matrices, real problems may occur when you subject such matrices to another analysis (e.g., *multiple regression*, *factor analysis*, or *cluster analysis*) that expects a "true correlation matrix," with a certain level of consistency and "transitivity" between different coefficients. Thus, if you are using the pairwise method of deleting the missing data, be sure to examine the distribution of missing data across the cells of the matrix for possible systematic "patterns."

How to Identify Biases Caused by the Bias due to Pairwise Deletion of Missing Data. If the pairwise deletion of missing data does not introduce any systematic bias to the correlation matrix, then all those pairwise descriptive statistics for one variable should be very similar. However, if they differ, then there are good reasons to suspect a bias. For example, if the mean (or standard deviation) of the values of variable A that were taken into account in calculating its correlation with variable B is much lower than the mean (or standard deviation) of those values of variable A that were used in calculating its correlation with variable C, then we would have good reason to suspect that those two correlations (A-B and A-C) are based on different subsets of data, and thus, that there is a bias in the correlation matrix caused by a non-random distribution of missing data.

Pairwise Deletion of Missing Data vs. Mean Substitution. Another common method to avoid loosing data due to casewise deletion is the so-called *mean substitution* of missing data (replacing all missing data in a variable by the mean of that variable). Mean substitution offers some advantages and some disadvantages as compared to pairwise deletion. Its main advantage is that it produces "internally consistent" sets of results ("true" correlation matrices). The main disadvantages are:

A. *Mean substitution* artificially decreases the variation of scores, and this decrease in individual variables is proportional to the number of missing data (i.e., the more missing data, the more "perfectly average scores" will be artificially added to the data set).
B. Because it substitutes missing data with artificially created "average" data points, *mean substitution* may considerably change the values of correlations.

Spurious Correlations. Although you cannot prove causal relations based on correlation coefficients (see *Elementary Concepts*), you can still identify so-called *spurious* correlations; that is, correlations that are due mostly to the influences of "other" variables. For example, there is a correlation between the total amount of losses in a fire and the number of firemen that were putting out the fire; however, what this correlation does not indicate is that if you call fewer firemen then you would lower the losses. There is a third variable (the initial *size* of the fire) that influences both the amount of losses and the number of firemen. If you "control" for this variable (e.g., consider only fires of a fixed size), then the correlation will either disappear or perhaps even change its sign. The main problem with spurious correlations is that we typically do not know what the "hidden" agent is. However, in cases when we know where to look, we can use *partial correlations* that control for (*partial out*) the influence of specified variables.

Are correlation coefficients "additive?" No, they are not. For example, an average of correlation coefficients in a number of samples does not represent an "average correlation" in all those samples. Because the value of the correlation coefficient is not a linear function of the magnitude of the relation between the variables, correlation coefficients cannot simply be averaged. In cases when you need to average correlations, they first have to be converted into additive measures. For example, before averaging, you can square them to obtain *coefficients of determination* which are additive (as explained before in this section), or convert them into so-called *Fisher z* values, which are also additive.

How to Determine Whether Two Correlation Coefficients are Significant. A test is available that will evaluate the significance of differences between two correlation coefficients in two samples. The outcome of this test depends not only on the size of the raw difference between the two coefficients but also on the size of the samples and on the size of the coefficients themselves. Consistent with the previously discussed principle, the larger the sample size, the smaller the effect that can be proven significant in that sample. In general, due to the fact that the reliability of the correlation coefficient increases with its absolute value, relatively small differences between large correlation coefficients can be significant. For example, a difference of .10 between two correlations may not be significant if the two coefficients are .15 and .25, although in the same sample, the same difference of .10 can be highly significant if the two coefficients are .80 and .90.

t-test for independent samples

Purpose, Assumptions. The *t*-test is the most commonly used method to evaluate the differences in means between two groups. For example, the *t*-test can be used to test for a difference in test scores between a group of patients who were given a drug and a control group who received a placebo. Theoretically, the t-test can be used even if the sample sizes are very small (e.g., as small as 10; some researchers claim that even smaller *n*'s are possible), as long as the variables are normally distributed within each group and the variation of scores in the two groups is not reliably different (see also *Elementary Concepts*). As mentioned before, the normality assumption can be evaluated by looking at the distribution of the data (via histograms) or by performing a normality test. The equality of variances assumption can be verified with the *F* test, or you can use the more robust *Levene's test*. If these conditions are not met, then you can evaluate the differences in means between two groups using one of the nonparametric alternatives to the *t*- test (see *Nonparametrics and Distribution Fitting*).

The *p*-level reported with a *t*-test represents the probability of error involved in accepting our research hypothesis about the existence of a difference. Technically speaking, this is the probability of error associated with rejecting the hypothesis of no difference between the two categories of observations (corresponding to the groups) in the population when, in fact, the hypothesis is true. Some researchers suggest that if the difference is in the predicted direction, you can consider only one half (one "tail") of the probability distribution and thus divide the standard *p*-level reported with a *t*-test (a "two-tailed" probability) by two. Others, however, suggest that you should always report the standard, two-tailed t-test probability.

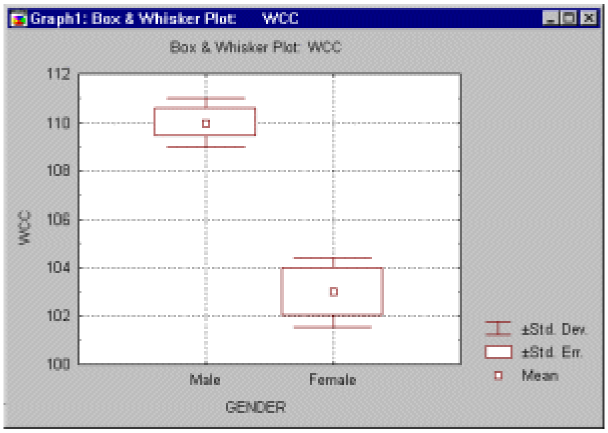
See also, Student's t Distribution.

Arrangement of Data. In order to perform the *t*-test for independent samples, one independent (*grouping*) variable (e.g., Gender: *male/female*) and at least one dependent variable (e.g., a test score) are required. The means of the dependent variable will be compared between selected groups based on the specified values (e.g., *male* and *female*) of the independent variable. The following data set can be analyzed with a *t*-test comparing the average *WCC* score in *males* and *females*.

	GENDER	WCC
case 1	male	111
case 2	male	110
case 3	male	109
case 4	female	102
case 5	female	104

	mean WCC in males = 110 **mean WCC in females = 103**

t-test graphs. In the *t*-test analysis, comparisons of means and measures of variation in the two groups can be visualized in *box and whisker* plots (for an example, see the graph below).



These graphs help you to quickly evaluate and "intuitively visualize" the strength of the relation between the grouping and the dependent variable.

More Complex Group Comparisons. It often happens in research practice that you need to compare more than two groups (e.g., *drug 1*, *drug 2*, and *placebo*), or compare groups created by more than one independent variable while controlling for the separate influence of each of them (e.g., *Gender*, *type of Drug*, and *size of Dose*). In these cases, you need to analyze the data using *Analysis of Variance*, which can be considered to be a generalization of the *t*-test. In fact, for two group comparisons, ANOVA will give results identical to a *t*-test (*t**2 [df] = F[1,df]*). However, when the design is more complex, ANOVA offers numerous advantages that *t*-tests cannot provide (even if you run a series of *t*- tests comparing various cells of the design).

t-test for dependent samples

Within-group Variation. As explained in *Elementary Concepts*, the size of a relation between two variables, such as the one measured by a difference in means between two groups, depends to a large extent on the differentiation of values *within* the group. Depending on how differentiated the values are in each group, a given "raw difference" in group means will indicate either a stronger or weaker relationship between the independent (*grouping*) and dependent variable. For example, if the mean WCC (White Cell Count) was 102 in males and 104 in females, then this difference of "only" 2 points would be extremely important if all values for males fell within a range of 101 to 103, and all scores for females fell within a range of 103 to 105; for example, we would be able to predict WCC pretty well based on gender. However, if the same difference of 2 was obtained from very differentiated scores (e.g., if their range was 0-200), then we would consider the difference entirely negligible. That is to say, reduction of the *within-group variation* increases the sensitivity of our test.

Purpose. The *t*-test for dependent samples helps us to take advantage of one specific type of design in which an important source of *within-group variation* (or so-called, *error*) can be easily identified and excluded from the analysis. Specifically, if two groups of observations (that are to be compared) are based on the same sample of subjects who were tested *twice* (e.g., *before* and *after* a treatment), then a considerable part of the within-group variation in both groups of scores can be attributed to the initial individual differences between subjects. Note that, in a sense, this fact is not much different than in cases when the two groups are entirely independent (see *t*-test for independent samples), where individual differences also contribute to the *error variance*; but in the case of independent samples, we cannot do anything about it because we cannot identify (or "subtract") the variation due to individual differences in subjects. However, if the same sample was tested twice, then we can easily identify (or "subtract") this variation. Specifically, instead of treating each group separately, and analyzing raw scores, we can look only at the differences between the two measures (e.g., "pre-test" and "post test") in each subject. By subtracting the first score from the second for each subject and then analyzing only those "pure (paired) differences," we will exclude the entire part of the variation in our data set that results from unequal base levels of individual subjects. This is precisely what is being done in the *t*-test for dependent samples, and, as compared to the *t*-test for independent samples, it always produces "better" results (i.e., it is always more sensitive).

Assumptions. The theoretical assumptions of the *t*-test for independent samples also apply to the dependent samples test; that is, the paired differences should be normally distributed. If these assumptions are clearly not met, then one of the nonparametric alternative tests should be used.

See also, Student's t Distribution.

Arrangement of Data. Technically, we can apply the *t*-test for dependent samples to any two variables in our data set. However, applying this test will make very little sense if the values of the two variables in the data set are not logically and methodologically comparable. For example, if you compare the average WCC in a sample of patients before and after a treatment, but using a different counting method or different units in the second measurement, then a highly significant *t*-test value could be obtained due to an artifact; that is, to the change of units of measurement. Following, is an example of a data set that can be analyzed using the *t*-test for dependent samples.

	WCC before	**WCC after**

case 1
case 2
case 3
case 4
case 5
...
	111.9
	109
	143
	101
	80
	...
		113
		110
		144
		102
		80.9
		...

	average change between WCC "before" and "after" = 1	

The average difference between the two conditions is relatively small (*d=1*) as compared to the differentiation (range) of the raw scores (from 80 to 143, in the first sample). However, the *t*-test for dependent samples analysis is performed only on the paired differences , "ignoring" the raw scores and their potential differentiation. Thus, the size of this particular difference of *1* will be compared not to the differentiation of raw scores but to the differentiation of the *individual difference scores*, which is relatively small: *0.2* (from *0.9* to *1.1*). Compared to that variability, the difference of *1* is extremely large and can yield a highly significant *t* value.

Matrices of t-tests. *t*-tests for dependent samples can be calculated for long lists of variables, and reviewed in the form of matrices produced with *casewise* or *pairwise* deletion of missing data, much like the correlation matrices. Thus, the precautions discussed in the context of correlations also apply to *t*-test matrices; see:

a. the issue of artifacts caused by the pairwise deletion of missing data in *t*-tests and
b. the issue of "randomly" significant test values.

More Complex Group Comparisons. If there are more than two "correlated samples" (e.g., *before treatment*, *after treatment 1*, and *after treatment 2*), then analysis of variance with *repeated measures* should be used. The repeated measures ANOVA can be considered a generalization of the t-test for dependent samples and it offers various features that increase the overall sensitivity of the analysis. For example, it can simultaneously control not only for the base level of the dependent variable, but it can control for other factors and/or include in the design more than one interrelated dependent variable (MANOVA; for additional details refer to *ANOVA/MANOVA*).

Breakdown: Descriptive Statistics by Groups

Purpose. The breakdowns analysis calculates descriptive statistics and correlations for *dependent* variables in each of a number of groups defined by one or more grouping (*independent*) variables.

Arrangement of Data. In the following example data set (spreadsheet), the dependent variable *WCC* (White Cell Count) can be broken down by 2 *independent* variables: *Gender* (values: *males* and *females*), and *Height* (values: *tall* and *short*).

	GENDER	**HEIGHT**	**WCC**
case 1	male	short	101
case 2	male	tall	110
case 3	male	tall	92
case 4	female	tall	112
case 5	female	short	95
...	...	...	...

The resulting breakdowns might look as follows (we are assuming that *Gender* was specified as the first independent variable, and *Height* as the second).

<table>
<tr><td colspan="4">Entire sample
Mean=100
SD=13
N=120</td></tr>
<tr><td colspan="2">Males
Mean=99
SD=13
N=60</td><td colspan="2">Females
Mean=101
SD=13
N=60</td></tr>
<tr><td>Tall/males
Mean=98
SD=13
N=30</td><td>Short/males
Mean=100
SD=13
N=30</td><td>Tall/females
Mean=101
SD=13
N=30</td><td>Short/females
Mean=101
SD=13
N=30</td></tr>
</table>

The composition of the "intermediate" level cells of the "breakdown tree" depends on the order in which independent variables are arranged. For example, in the above example, you see the means for "all males" and "all females" but you do not see the means for "all tall subjects" and "all short subjects" which would have been produced had you specified independent variable *Height* as the first grouping variable rather than the second.

Statistical Tests in Breakdowns. Breakdowns are typically used as an exploratory data analysis technique; the typical question that this technique can help answer is very simple: Are the groups created by the independent variables different regarding the dependent variable? If you are interested in differences concerning the means, then the appropriate test is the breakdowns one-way ANOVA (*F* test). If you are interested in variation differences, then you should test for homogeneity of variances.

Other Related Data Analysis Techniques. Although for exploratory data analysis, breakdowns can use more than one independent variable, the statistical procedures in breakdowns assume the existence of a single grouping factor (even if, in fact, the breakdown results from a combination of a number of grouping variables). Thus, those statistics do not reveal or even take into account any possible *interactions* between grouping variables in the design. For example, there could be differences between the influence of one independent variable on the dependent variable at different levels of another independent variable (e.g., tall people could have lower WCC than short ones, but only if they are males; see the "tree" data above). You can explore such effects by examining breakdowns "visually," using different orders of independent variables, but the magnitude or significance of such effects cannot be estimated by the breakdown statistics.

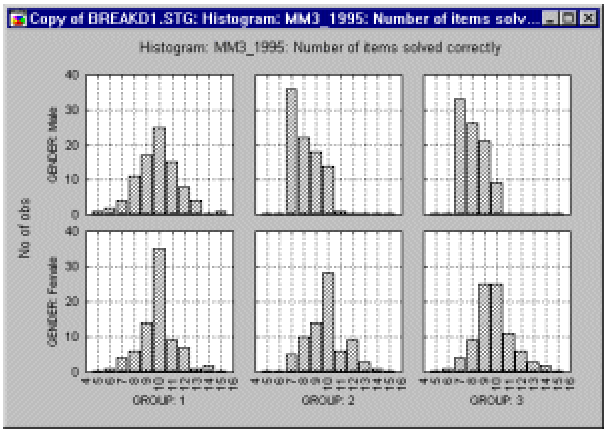
Post-Hoc Comparisons of Means. Usually, after obtaining a statistically significant *F* test from the ANOVA, one wants to know which of the means contributed to the effect (i.e., which groups are particularly different from each other). One could of course perform a series of simple *t*-tests to compare all possible pairs of means. However, such a procedure would *capitalize on chance*. This means that the reported probability levels would actually overestimate the statistical significance of mean differences. Without going into too much detail, suppose you took 20 samples of 10 random numbers each, and computed 20 means. Then, take the group (sample) with the highest mean and compare it with that of the lowest mean. The *t*-test for independent samples will test whether or not those two means are significantly different from each other, provided they were *the only two samples* taken. *Post-hoc* comparison techniques on the other hand specifically take into account the fact that more than two samples were taken.

Breakdowns vs. Discriminant Function Analysis. Breakdowns can be considered as a first step toward another type of analysis that explores differences between groups: *Discriminant function analysis*. Similar to breakdowns, discriminant function analysis explores the differences between groups created by values (group codes) of an independent (*grouping*) variable. However, unlike breakdowns, discriminant function analysis simultaneously analyzes more than one dependent variable and it identifies "patterns" of values of those dependent variables. Technically, it determines a linear combination of the dependent variables that best predicts the group membership. For example, discriminant function analysis can be used to analyze differences between three groups of persons who have chosen different professions (e.g., lawyers, physicians, and engineers) in terms of various aspects of their scholastic performance in high school. One could claim that such analysis could "explain" the choice of a profession in terms of specific talents shown in high school; thus discriminant function analysis can be considered to be an "exploratory extension" of simple breakdowns.

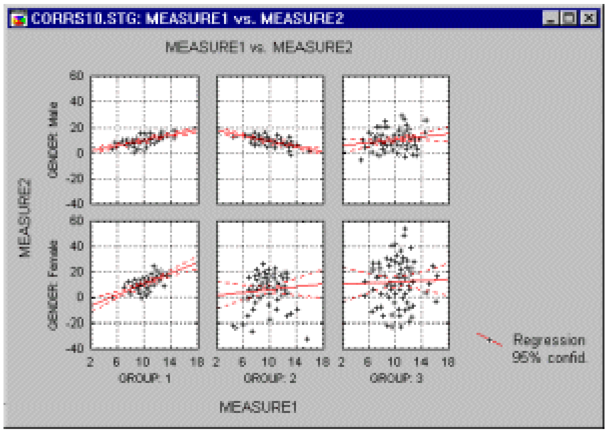
Breakdowns vs. Frequency Tables. Another related type of analysis that cannot be directly performed with breakdowns is comparisons of frequencies of cases (*n*'s) between groups. Specifically, often the *n*'s in individual cells are not equal because the assignment of subjects to those groups typically results not from an experimenter's manipulation, but from subjects' pre-existing dispositions. If, in spite of the random selection of the entire sample, the *n*'s are unequal, then it may suggest that the independent variables are related. For example, crosstabulating levels of independent variables *Age* and *Education* most likely would not create groups of equal *n*, because education is distributed differently in different age groups. If you are interested in such comparisons, you can explore specific frequencies in the breakdowns tables, trying different orders of independent variables. However, in order to subject such differences to statistical tests, you should use crosstabulations and frequency tables, Log-Linear Analysis, or Correspondence Analysis (for more advanced analyses on multi-way frequency tables).

Graphical breakdowns.

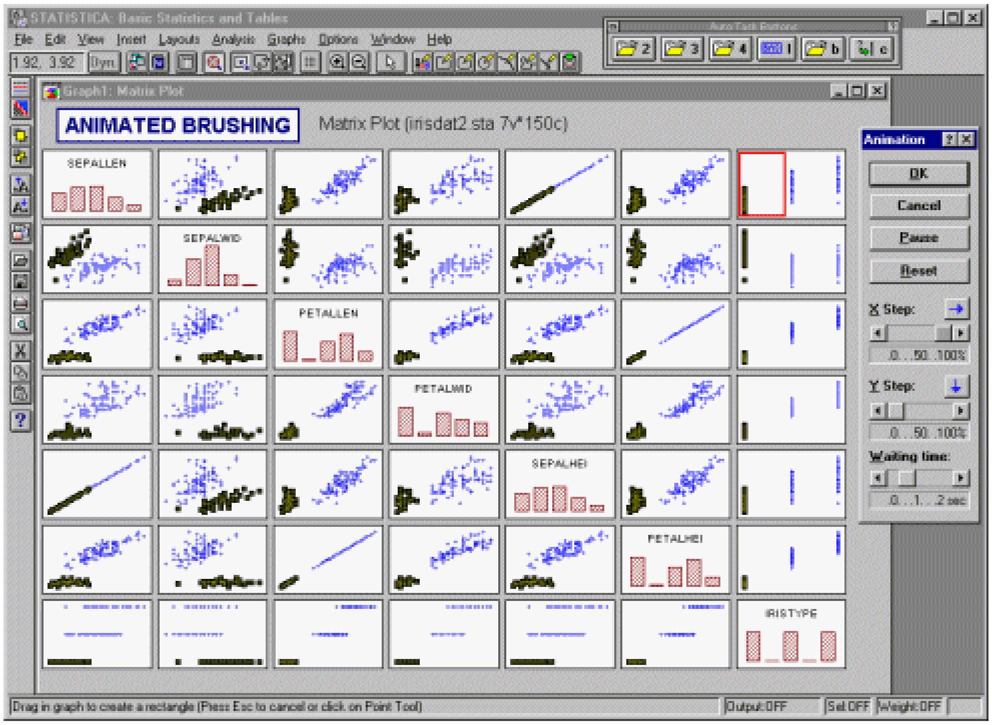
Graphs can often identify effects (both expected and unexpected) in the data more quickly and sometimes "better" than any other data analysis method. Categorized graphs allow you to plot the means, distributions, correlations, etc. across the groups of a given table (e.g., categorized histograms, categorized probability plots, categorized box and whisker plots). The graph below shows a categorized histogram which enables you to quickly evaluate and visualize the shape of the data for each group (group1-female, group2-female, etc.).



The categorized scatterplot (in the graph below) shows the differences between patterns of correlations between dependent variables across the groups.



Additionally, if the software has a brushing facility which supports animated brushing, you can select (i.e., highlight) in a matrix scatterplot all data points that belong to a certain category in order to examine how those specific observations contribute to relations between other variables in the same data set.



Frequency tables

Purpose. Frequency or one-way tables represent the simplest method for analyzing categorical (*nominal*) data (refer to *Elementary Concepts*). They are often used as one of the exploratory procedures to review how different categories of values are distributed in the sample. For example, in a survey of spectator interest in different sports, we could summarize the respondents' interest in watching football in a frequency table as follows:

STATISTICA **BASIC** **STATS**	**FOOTBALL: "Watching football"**			
Category	**Count**	**Cumulatv Count**	**Percent**	**Cumulatv Percent**
ALWAYS : Always interested	39	39	39.00000	39.0000
USUALLY : Usually interested	16	55	16.00000	55.0000
SOMETIMS: Sometimes interested	26	81	26.00000	81.0000
NEVER : Never interested	19	100	19.00000	100.0000
Missing	0	100	0.00000	100.0000

The table above shows the number, proportion, and cumulative proportion of respondents who characterized their interest in watching football as either (1) *Always interested*, (2) *Usually interested*, (3) *Sometimes interested*, or (4) *Never interested*.

Applications. In practically every research project, a first "look" at the data usually includes frequency tables. For example, in survey research, frequency tables can show the number of males and females who participated in the survey, the number of respondents from particular ethnic and racial backgrounds, and so on. Responses on some labeled attitude measurement scales (e.g., interest in watching football) can also be nicely summarized via the frequency table. In medical research, one may tabulate the number of patients displaying specific symptoms; in industrial research one may tabulate the frequency of different causes leading to catastrophic failure of products during stress tests (e.g., which parts are actually responsible for the complete malfunction of television sets under extreme temperatures?). Customarily, if a data set includes any categorical data, then one of the first steps in the data analysis is to compute a frequency table for those categorical variables.

Crosstabulation and stub-and-banner tables

Purpose and Arrangement of Table. Crosstabulation is a combination of two (or more) frequency tables arranged such that each cell in the resulting table represents a unique combination of specific values of crosstabulated variables. Thus, crosstabulation allows us to examine frequencies of observations that belong to specific categories on more than one variable. By examining these frequencies, we can identify relations between crosstabulated variables. Only categorical (*nominal*) variables or variables with a relatively small number of different meaningful values should be crosstabulated. Note that in the cases where we do want to include a continuous variable in a crosstabulation (e.g., income), we can first *recode* it into a particular number of distinct ranges (e.g., low, medium, high).

2x2 Table. The simplest form of crosstabulation is the 2 by 2 table where two variables are "crossed," and each variable has only two distinct values. For example, suppose we conduct a simple study in which males and females are asked to choose one of two different brands of soda pop (brand *A* and brand *B*); the data file can be arranged like this:

	GENDER	**SODA**
case 1	MALE	A
case 2	FEMALE	B
case 3	FEMALE	B
case 4	FEMALE	A
case 5	MALE	B
...	...	...

The resulting crosstabulation could look as follows.

	SODA: A	**SODA: B**	
GENDER: MALE	20 (40%)	30 (60%)	50 (50%)
GENDER: FEMALE	30 (60%)	20 (40%)	50 (50%)
	50 (50%)	50 (50%)	100 (100%)

Each cell represents a unique combination of values of the two crosstabulated variables (row variable *Gender* and column variable *Soda*), and the numbers in each cell tell us how many observations fall into each combination of values. In general, this table shows us that more females than males chose the soda pop brand *A*, and that more males than females chose soda *B*. Thus, gender and preference for a particular brand of soda may be related (later we will see how this relationship can be measured).

Marginal Frequencies. The values in the margins of the table are simply one-way (frequency) tables for all values in the table. They are important in that they help us to evaluate the arrangement of frequencies in individual columns or rows. For example, the frequencies of 40% and 60% of males and females (respectively) who chose soda *A* (see the first column of the above table), would not indicate any relationship between *Gender* and *Soda* if the marginal frequencies for *Gender* were also 40% and 60%; in that case they would simply reflect the different proportions of males and females in the study. Thus, the differences between the distributions of frequencies in individual rows (or columns) and in the respective margins informs us about the relationship between the crosstabulated variables.

Column, Row, and Total Percentages. The example in the previous paragraph demonstrates that in order to evaluate relationships between crosstabulated variables, we need to compare the proportions of marginal and individual column or row frequencies. Such comparisons are easiest to perform when the frequencies are presented as percentages.

Graphical Representations of Crosstabulations. For analytic purposes, the individual rows or columns of a table can be represented as column graphs. However, often it is useful to visualize the entire table in a single graph. A two-way table can be visualized in a 3-dimensional histogram; alternatively, a categorized histogram can be produced, where one variable is represented by individual histograms which are drawn at each level (category) of the other variable in the crosstabulation. The advantage of the 3D histogram is that it produces an integrated picture of the entire table; the advantage of the categorized graph is that it allows us to precisely evaluate specific frequencies in each cell of the table.

Stub-and-Banner Tables. Stub-and-Banner tables, or *Banners* for short, are a way to display several two-way tables in a compressed form. This type of table is most easily explained with an example. Let us return to the survey of sports spectators example. (Note that, in order simplify matters, only the response categories *Always* and *Usually* were tabulated in the table below.)

STATISTICA BASIC STATS	**Stub-and-Banner Table: Row Percent**		
Factor	**FOOTBALL ALWAYS**	**FOOTBALL USUALLY**	**Row Total**
BASEBALL: ALWAYS **BASEBALL: USUALLY**	92.31 61.54	7.69 38.46	66.67 33.33
BASEBALL: Total	82.05	17.95	100.00
TENNIS: ALWAYS **TENNIS: USUALLY**	87.50 87.50	12.50 12.50	66.67 33.33
TENNIS: Total	87.50	12.50	100.00
BOXING: ALWAYS **BOXING: USUALLY**	77.78 100.00	22.22 0.00	52.94 47.06
BOXING : Total	88.24	11.76	100.00

Interpreting the Banner Table. In the table above, we see the two-way tables of expressed interest in *Football* by expressed interest in *Baseball*, *Tennis*, and *Boxing*. The table entries represent percentages of rows, so that the percentages across columns will add up to 100 percent. For example, the number in the upper left hand corner of the Scrollsheet (*92.31*) shows that *92.31* percent of all respondents who said they are always interested in watching football also said that they were always interested in watching baseball. Further down we can see that the percent of those always interested in watching football who were also always interested in watching tennis was *87.50* percent; for boxing this number is *77.78* percent. The percentages in the last column (Row Total) are always relative to the total number of cases.

Multi-way Tables with Control Variables. When only two variables are crosstabulated, we call the resulting table a *two-way* table. However, the general idea of crosstabulating values of variables can be generalized to more than just two variables. For example, to return to the "soda" example presented earlier (see above), a third variable could be added to the data set. This variable might contain information about the state in which the study was conducted (either *Nebraska* or *New York*).

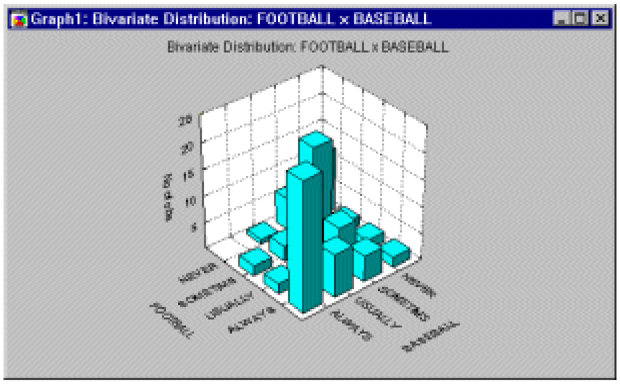
	GENDER	SODA	STATE
case 1	MALE	A	NEBRASKA
case 2	FEMALE	B	NEW YORK
case 3	FEMALE	B	NEBRASKA
case 4	FEMALE	A	NEBRASKA
case 5	MALE	B	NEW YORK
...	...	...	...

The crosstabulation of these variables would result in a 3-way table:

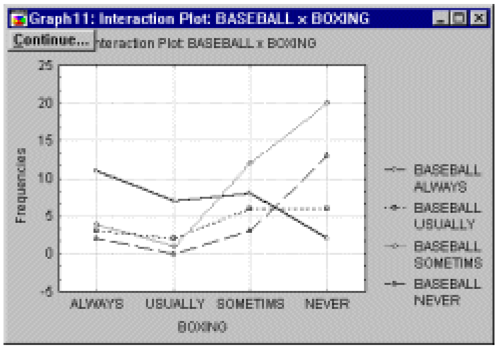
	STATE: NEW YORK			STATE: NEBRASKA		
	SODA: A	SODA: B		SODA: A	SODA: B	
G:MALE	20	30	50	5	45	50
G:FEMALE	30	20	50	45	5	50
	50	50	100	50	50	100

Theoretically, an unlimited number of variables can be crosstabulated in a single multi-way table. However, research practice shows that it is usually difficult to examine and "understand" tables that involve more than 4 variables. It is recommended to analyze relationships between the factors in such tables using modeling techniques such as *Log-Linear Analysis* or *Correspondence Analysis*.

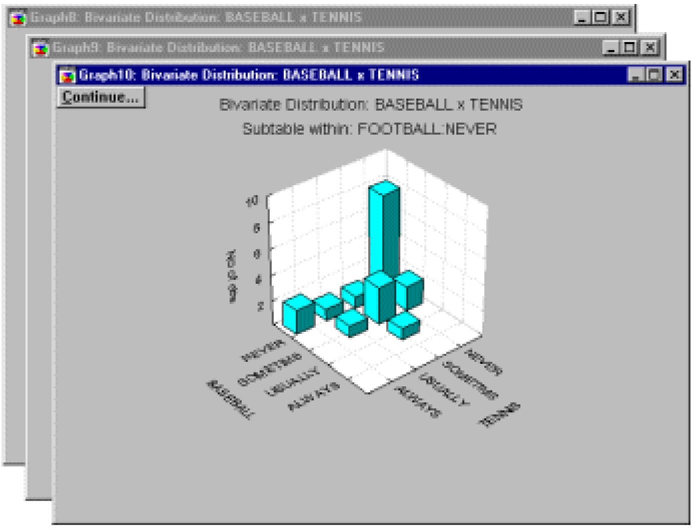
Graphical Representations of Multi-way Tables. You can produce "double categorized" histograms, 3D histograms,



or line-plots that will summarize the frequencies for up to 3 factors in a single graph.



Batches (cascades) of graphs can be used to summarize higher-way tables (as shown in the graph below).



Statistics in Crosstabulation Tables

- General Introduction
- Pearson Chi-square
- Maximum-Likelihood (M-L) Chi-square

- Yates' correction
- Fisher exact test
- McNemar Chi-square
- Coefficient Phi
- Tetrachoric correlation
- Coefficient of contingency (C)
- Interpretation of contingency measures
- Statistics Based on Ranks
- Spearman R
- Kendall tau
- Sommer's d: d(X|Y), d(Y|X)
- Gamma
- Uncertainty Coefficients: S(X,Y), S(X|Y), S(Y|X)

General Introduction. Crosstabulations generally allow us to identify relationships between the crosstabulated variables. The following table illustrates an example of a very strong relationship between two variables: variable *Age* (*Adult* vs. *Child*) and variable *Cookie* preference (*A* vs. *B*).

	COOKIE: A	COOKIE: B	
AGE: ADULT	50	0	50
AGE: CHILD	0	50	50
	50	50	100

All adults chose cookie *A*, while all children chose cookie *B*. In this case there is little doubt about the reliability of the finding, because it is hardly conceivable that one would obtain such a pattern of frequencies by chance alone; that is, without the existence of a "true" difference between the cookie preferences of adults and children. However, in real-life, relations between variables are typically much weaker, and thus the question arises as to how to measure those relationships, and how to evaluate their reliability (statistical significance). The following review includes the most common measures of relationships between *two* categorical variables; that is, measures for two-way tables. The techniques used to analyze simultaneous relations between *more than two* variables in higher order crosstabulations are discussed in the context of the *Log-Linear Analysis* module and the *Correspondence Analysis*.

Pearson Chi-square. The Pearson *Chi-square* is the most common test for significance of the relationship between categorical variables. This measure is based on the fact that we can compute the *expected* frequencies in a two-way table (i.e., frequencies that we would *expect* if there was no relationship between the variables). For example, suppose we ask 20 males and 20 females to choose between two brands of soda pop (brands *A* and *B*). If there is no relationship between preference and gender, then we would *expect* about an equal number of choices of brand *A* and brand *B* for each sex. The *Chi-square* test becomes increasingly significant as the numbers deviate further from this expected pattern; that is, the more this pattern of choices for males and females differs.

The value of the *Chi-square* and its significance level depends on the overall number of observations and the number of cells in the table. Consistent with the principles discussed in *Elementary Concepts*, relatively small deviations of the relative frequencies across cells from the expected pattern will prove significant if the number of observations is large.

The only assumption underlying the use of the *Chi-square* (other than random selection of the sample) is that the expected frequencies are not very small. The reason for this is that, actually, the *Chi-square* inherently tests the underlying *probabilities* in each cell; and when the expected cell frequencies fall, for example, below 5, those probabilities cannot be estimated with sufficient precision. For further discussion of this issue refer to Everitt (1977), Hays (1988), or Kendall and Stuart (1979).

Maximum-Likelihood Chi-square. The *Maximum-Likelihood* *Chi-square* tests the same hypothesis as the Pearson *Chi- square* statistic; however, its computation is based on Maximum-Likelihood theory. In practice, the M-L *Chi-square* is usually very close in magnitude to the Pearson *Chi- square* statistic. For more details about this statistic refer to Bishop, Fienberg, and Holland (1975), or Fienberg, S. E. (1977); the *Log-Linear Analysis* chapter of the manual also discusses this statistic in greater detail.

Yates Correction. The approximation of the *Chi-square* statistic in small 2 x 2 tables can be improved by reducing the absolute value of differences between expected and observed frequencies by 0.5 before squaring (*Yates' correction*). This correction, which makes the estimation more conservative, is usually applied when the table contains only small observed frequencies, so that some expected frequencies become less than 10 (for further discussion of this correction, see Conover, 1974; Everitt, 1977; Hays, 1988; Kendall & Stuart, 1979; and Mantel, 1974).

Fisher Exact Test. This test is only available for 2x2 tables; it is based on the following rationale: Given the marginal frequencies in the table, and assuming that in the population the two factors in the table are not related, how likely is it to obtain cell frequencies as uneven or worse than the ones that were observed? For small *n*, this probability can be computed *exactly* by counting all possible tables that can be constructed based on the marginal frequencies. Thus, the Fisher exact test computes the exact probability under the null hypothesis of obtaining the current distribution of frequencies across cells, or one that is more uneven.

McNemar Chi-square. This test is applicable in situations where the frequencies in the 2 x 2 table represent *dependent* samples. For example, in a before-after design study, we may count the number of students who fail a test of minimal math skills at the beginning of the semester and at the end of the semester. Two *Chi-square* values are reported: *A/D* and *B/C*. The *Chi-square A/D* tests the hypothesis that the frequencies in cells *A* and *D* (upper left, lower right) are identical. The *Chi-square B/C* tests the hypothesis that the frequencies in cells *B* and *C* (upper right, lower left) are identical.

Coefficient Phi. The *Phi-square* is a measure of correlation between two categorical variables in a 2 x 2 table. Its value can range from *0* (no relation between factors; *Chi-square*=0.0) to *1* (perfect relation between the two factors in the table). For more details concerning this statistic see Castellan and Siegel (1988, p. 232).

Tetrachoric Correlation. This statistic is also only computed for (applicable to) 2 x 2 tables. If the 2 x 2 table can be thought of as the result of two continuous variables that were (artificially) forced into two categories each, then the tetrachoric correlation coefficient will estimate the correlation between the two.

Coefficient of Contingency. The coefficient of contingency is a *Chi-square* based measure of the relation between two categorical variables (proposed by Pearson, the originator of the *Chi-square* test). Its advantage over the ordinary *Chi-square* is that it is more easily interpreted, since its range is always limited to *0* through *1* (where 0 means complete independence). The disadvantage of this statistic is that its specific upper limit is "limited" by the size of the table; *C* can reach the limit of *1* only if the number of categories is unlimited (see Siegel, 1956, p. 201).

Interpretation of Contingency Measures. An important disadvantage of measures of contingency (reviewed above) is that they do not lend themselves to clear interpretations in terms of probability or "proportion of variance," as is the case, for example, of the Pearson *r* (see Correlations). There is no commonly accepted measure of relation between categories that has such a clear interpretation.

Statistics Based on Ranks. In many cases the categories used in the crosstabulation contain meaningful rank-ordering information; that is, they measure some characteristic on an <>*ordinal* scale (see *Elementary Concepts*). Suppose we asked a sample of respondents to indicate their interest in watching different sports on a 4-point scale with the explicit labels (1) *always*, (2) *usually*, (3) *sometimes*, and (4) *never interested*. Obviously, we can assume that the response *sometimes interested* is indicative of less interest than *always interested*, and so on. Thus, we could rank the respondents with regard to their expressed interest in, for example, watching football. When categorical variables can be interpreted in this manner, there are several additional indices that can be computed to express the relationship between variables.

Spearman R. Spearman *R* can be thought of as the regular Pearson product-moment correlation coefficient (Pearson *r*); that is, in terms of the proportion of variability accounted for, except that Spearman *R* is computed from ranks. As mentioned above, Spearman *R* assumes that the variables under consideration were measured on at least an *ordinal* (rank order) scale; that is, the individual observations (cases) can be ranked into two ordered series. Detailed discussions of the Spearman *R* statistic, its power and efficiency can be found in Gibbons (1985), Hays (1981), McNemar (1969), Siegel

(1956), Siegel and Castellan (1988), Kendall (1948), Olds (1949), or Hotelling and Pabst (1936).

Kendall tau. Kendall *tau* is equivalent to the Spearman *R* statistic with regard to the underlying assumptions. It is also comparable in terms of its statistical power. However, Spearman *R* and Kendall *tau* are usually not identical in magnitude because their underlying logic, as well as their computational formulas are very different. Siegel and Castellan (1988) express the relationship of the two measures in terms of the inequality:

-1 < = 3 * Kendall tau - 2 * Spearman R < = 1

More importantly, Kendall *tau* and Spearman *R* imply different interpretations: While Spearman *R* can be thought of as the regular Pearson product-moment correlation coefficient as computed from ranks, Kendall *tau* rather represents a *probability*. Specifically, it is the difference between the probability that the observed data are in the same order for the two variables versus the probability that the observed data are in different orders for the two variables. Kendall (1948, 1975), Everitt (1977), and Siegel and Castellan (1988) discuss Kendall *tau* in greater detail. Two different variants of *tau* are computed, usually called tau_b and tau_c. These measures differ only with regard as to how tied ranks are handled. In most cases these values will be fairly similar, and when discrepancies occur, it is probably always safest to interpret the lowest value.

Sommer's d: d(X|Y), d(Y|X). Sommer's *d* is an asymmetric measure of association related to t_b (see Siegel & Castellan, 1988, p. 303-310).

Gamma. The *Gamma* statistic is preferable to Spearman *R* or Kendall *tau* when the data contain many tied observations. In terms of the underlying assumptions, *Gamma* is equivalent to Spearman *R* or Kendall *tau*; in terms of its interpretation and computation, it is more similar to Kendall *tau* than Spearman *R*. In short, *Gamma* is also a *probability*; specifically, it is computed as the difference between the probability that the rank ordering of the two variables agree minus the probability that they disagree, divided by 1 minus the probability of ties. Thus, *Gamma* is basically equivalent to Kendall *tau*, except that ties are explicitly taken into account. Detailed discussions of the *Gamma* statistic can be found in Goodman and Kruskal (1954, 1959, 1963, 1972), Siegel (1956), and Siegel and Castellan (1988).

Uncertainty Coefficients. These are indices of *stochastic dependence*; the concept of *stochastic dependence* is derived from the information theory approach to the analysis of frequency tables and the user should refer to the appropriate references (see Kullback, 1959; Ku & Kullback, 1968; Ku, Varner, & Kullback, 1971; see also Bishop, Fienberg, & Holland, 1975, p. 344-348). $S(Y,X)$ refers to symmetrical dependence, $S(X|Y)$ and $S(Y|X)$ refer to asymmetrical dependence.

Multiple Responses/Dichotomies. Multiple response variables or multiple dichotomies often arise when summarizing survey data. The nature of such variables or factors in a table is best illustrated with examples.

- Multiple Response Variables
- Multiple Dichotomies
- Crosstabulation of Multiple Responses/Dichotomies
- Paired Crosstabulation of Multiple Response Variables
- A Final Comment

Multiple Response Variables. As part of a larger market survey, suppose you asked a sample of consumers to name their three favorite soft drinks. The specific item on the questionnaire may look like this:

Write down your three favorite soft drinks:
1:__________ 2:__________ 3:__________

Thus, the questionnaires returned to you will contain somewhere between 0 and 3 answers to this item. Also, a wide variety of soft drinks will most likely be named. Your goal is to summarize the responses to this item; that is, to produce a table that summarizes the percent of respondents who mentioned a respective soft drink.

The next question is how to enter the responses into a data file. Suppose 50 different soft drinks were mentioned among all of the questionnaires. You could of course set up 50 variables - one for each soft drink - and then enter a *1* for the respective respondent and variable (soft drink), if he or she mentioned the respective soft drink (and a *0* if not); for example:

	COKE	**PEPSI**	**SPRITE**	**. . . .**
case 1	0	1	0	
case 2	1	1	0	
case 3	0	0	1	
...	...	...	...	

This method of coding the responses would be very tedious and "wasteful." Note that each respondent can only give a maximum of three responses; yet we use 50 variables to code those responses. (However, if we are only interested in these three soft drinks, then this method of coding just those three variables would be satisfactory; to tabulate soft drink preferences, we could then treat the three variables as a *multiple dichotomy*; see below.)

Coding multiple response variables. Alternatively, we could set up three variables, and a coding scheme for the 50 soft drinks. Then we could enter the respective codes (or alpha labels) into the three variables, in the same way that respondents wrote them down in the questionnaire.

	Resp. 1	**Resp. 2**	**Resp. 3**
case 1	COKE	PEPSI	JOLT
case 2	SPRITE	SNAPPLE	DR. PEPPER

case 3	PERRIER	GATORADE	MOUNTAIN DEW
. . .	. . .	. . .	. . .

To produce a table of the number of respondents by soft drink we would now treat *Resp.1* to *Resp3* as a *multiple response variable*. That table could look like this:

N=500 **Category**	**Count**	**Prcnt. of Responses**	**Prcnt. of Cases**
COKE: Coca Cola	44	5.23	8.80
PEPSI: Pepsi Cola	43	5.11	8.60
MOUNTAIN: Mountain Dew	81	9.62	16.20
PEPPER: Doctor Pepper	74	8.79	14.80
. . . :	..	...	...
	842	100.00	168.40

Interpreting the multiple response frequency table. The total number of respondents was *n*=500. Note that the counts in the first column of the table do not add up to 500, but rather to 842. That is the total number of *responses*; since each respondent could make up to 3 responses (write down three names of soft drinks), the total number of responses is naturally greater than the number of respondents. For example, referring back to the sample listing of the data file shown above, the first case (*Coke*, *Pepsi*, *Jolt*) "contributes" three times to the frequency table, once to the category *Coke*, once to the category *Pepsi*, and once to the category *Jolt*. The second and third columns in the table above report the percentages relative to the number of responses (second column) as well as respondents (third column). Thus, the entry 8.80 in the first row and last column in the table above means that 8.8% of all respondents mentioned *Coke* either as their first, second, or third soft drink preference.

Multiple Dichotomies. Suppose in the above example we were only interested in *Coke*, *Pepsi*, and *Sprite*. As pointed out earlier, one way to code the data in that case would be as follows:

	COKE	**PEPSI**	**SPRITE**	**. . . .**
case 1		1		
case 2	1	1		
case 3			1	
. . .	. . .	. . .	. . .	

In other words, one variable was created for each soft drink, then a value of *1* was entered into the respective variable whenever the respective drink was mentioned by the respective respondent. Note that each variable represents a *dichotomy*; that is, only "*1*"s and "*not 1*"s are allowed (we could have entered *1*'s and *0*'s, but to save typing we can also simply leave the *0*'s blank or missing). When tabulating these variables, we would like to obtain a summary table very similar to the one shown earlier for multiple response variables; that is, we would like to compute the number and percent of respondents (and responses) for each soft drink. In a sense, we "compact" the three variables *Coke*, *Pepsi*, and *Sprite* into a single variable (*Soft Drink*) consisting of *multiple dichotomies*.

Crosstabulation of Multiple Responses/Dichotomies. All of these types of variables can then be used in crosstabulation tables. For example, we could crosstabulate a multiple dichotomy for *Soft Drink* (coded as described in the previous paragraph) with a multiple response variable *Favorite Fast Foods* (with many categories such as *Hamburgers*, *Pizza*, etc.), by the simple categorical variable *Gender*. As in the frequency table, the percentages and marginal totals in that table can be computed from the total number of respondents as well as the total number of responses. For example, consider the following hypothetical respondent:

Gender	Coke	Pepsi	Sprite	Food1	Food2
FEMALE	1	1		FISH	PIZZA

This female respondent mentioned *Coke* and *Pepsi* as her favorite drinks, and *Fish* and *Pizza* as her favorite fast foods. In the complete crosstabulation table she will be counted in the following cells of the table:

		Food			. . .	TOTAL No. of RESP.
Gender	**Drink**	**HAMBURG.**	**FISH**	**PIZZA**	**. . .**	
FEMALE	COKE		X	X		2
	PEPSI		X	X		2
	SPRITE					
MALE	COKE					
	PEPSI					
	SPRITE					

This female respondent will "contribute" to (i.e., be counted in) the crosstabulation table a total of 4 times. In addition, she will be counted twice in the *Female--Coke* marginal frequency column if that column is requested to represent the total number of responses; if the marginal totals are computed as the total number of respondents, then this respondent will only be counted once.

Paired Crosstabulation of Multiple Response Variables. A unique option for tabulating multiple response variables is to treat the variables in two or more multiple response variables as matched pairs. Again, this method is best illustrated with a simple example. Suppose we conducted a survey of past and present home ownership. We asked the respondents to describe their last three (including the present) homes that they purchased. Naturally, for some respondents the present home is the first and only home; others have owned more than one home in the past. For each home we asked our respondents to write down the number of rooms in the respective house, and the number of occupants. Here is how the data for one respondent (say case number *112*) may be entered into a data file:

Case no.	**Rooms**	**1**	**2**	**3**	**No. Occ.**	**1**	**2**	**3**
112		3	3	4		2	3	5

This respondent owned three homes; the first had 3 rooms, the second also had 3 rooms, and the third had 4 rooms. The family apparently also grew; there were 2 occupants in the first home, 3 in the second, and 5 in the third.

Now suppose we wanted to crosstabulate the number of rooms by the number of occupants for all respondents. One way to do so is to prepare three different two-way tables; one for each home. We can also treat the two factors in this study (*Number of Rooms, Number of Occupants*) as multiple response variables. However, it would obviously not make any sense to count the example respondent *112* shown above in cell *3 Rooms - 5 Occupants* of the crosstabulation table (which we would, if we simply treated the two factors as ordinary multiple response variables). In other words, we want to ignore the combination of occupants in the third home with the number of rooms in the first home. Rather, we would like to count these variables in *pairs*; we would like to consider the number of rooms in the first home together with the number of occupants in the first home, the number of rooms in the second home with the number of occupants in the second home, and so on. This is exactly what will be accomplished if we asked for a paired crosstabulation of these multiple response variables.

A Final Comment. When preparing complex crosstabulation tables with multiple responses/dichotomies, it is sometimes difficult (in our experience) to "keep track" of exactly how the cases in the file are counted. The best way to verify that one understands the way in which the respective tables are constructed is to crosstabulate some simple example data, and then to trace how each case is counted. The example section of the *Crosstabulation* chapter in the manual employs this method to illustrate how data are counted for tables involving multiple response variables and multiple dichotomies.

Chapter 3
ANOVA/MANOVA

- Basic Ideas
    - The Partitioning of Sums of Squares
    - Multi-Factor ANOVA
    - Interaction Effects
- Complex Designs
    - Between-Groups and Repeated Measures
    - Incomplete (Nested) Designs
- Analysis of Covariance (ANCOVA)
    - Fixed Covariates
    - Changing Covariates
- Multivariate Designs: MANOVA/MANCOVA
    - Between-Groups Designs
    - Repeated Measures Designs
    - Sum Scores versus MANOVA
- Contrast Analysis and Post hoc Tests
    - Why Compare Individual Sets of Means?
    - Contrast Analysis
    - Post hoc Comparisons
- Assumptions and Effects of Violating Assumptions
    - Deviation from Normal Distribution
    - Homogeneity of Variances
    - Homogeneity of Variances and Covariances
    - Sphericity and Compound Symmetry
- Methods for Analysis of Variance

This chapter includes a general introduction to ANOVA and a discussion of the general topics in the analysis of variance techniques, including repeated measures designs, ANCOVA, MANOVA, unbalanced and incomplete designs, contrast effects, post-hoc comparisons, assumptions, etc. For related topics, see also Variance Components (topics related to estimation of variance components in mixed model designs), Experimental Design/DOE (topics related to specialized applications of ANOVA in industrial settings), and Repeatability and Reproducibility Analysis (topics related to specialized designs for evaluating the reliability and precision of measurement systems).

See also General Linear Models, General Regression Models; to analyze nonlinear models, see Generalized Linear Models.

Basic Ideas

The Purpose of Analysis of Variance

In general, the purpose of analysis of variance (ANOVA) is to test for significant differences between means. *Elementary Concepts* provides a brief introduction into the basics of statistical significance testing. If we are only comparing two means, then ANOVA will give the same results as the *t* test for independent samples (if we are comparing two different groups of cases or observations), or the *t* test for dependent samples (if we are comparing two variables in one set of cases or observations). If you are not familiar with those tests you may at this point want to "brush up" on your knowledge about those tests by reading *Basic Statistics and Tables*.

Why the name analysis of variance? It may seem odd to you that a procedure that compares means is called analysis of variance. However, this name is derived from the fact that in order to test for statistical significance between means, we are actually comparing (i.e., analyzing) variances.

- The Partitioning of Sums of Squares
- Multi-Factor ANOVA
- Interaction Effects

For more introductory topics, see the topic name.

- Complex Designs
- Analysis of Covariance (ANCOVA)
- Multivariate Designs: MANOVA/MANCOVA
- Contrast Analysis and Post hoc Tests
- Assumptions and Effects of Violating Assumptions

See also Methods for Analysis of Variance, Variance Components and Mixed Model ANOVA/ANCOVA, and Experimental Design (DOE).

The Partioning of Sums of Squares

At the heart of ANOVA is the fact that variances can be divided up, that is, partitioned. Remember that the variance is computed as the sum of squared deviations from the overall mean, divided by *n-1* (sample size minus one). Thus, given a certain n, the variance is a function of the sums of (deviation) squares, or *SS* for short. Partitioning of variance works as follows. Consider the following data set:

	Group 1	**Group 2**
Observation 1	2	6
Observation 2	3	7
Observation 3	1	5
Mean	2	6
Sums of Squares (SS)	2	2
Overall Mean	4	

Total Sums of Squares	28

The means for the two groups are quite different (*2* and *6*, respectively). The sums of squares *within* each group are equal to *2*. Adding them together, we get *4*. If we now repeat these computations, ignoring group membership, that is, if we compute the total *SS* based on the overall mean, we get the number *28*. In other words, computing the variance (sums of squares) based on the within-group variability yields a much smaller estimate of variance than computing it based on the total variability (the overall mean). The reason for this in the above example is of course that there is a large difference between means, and it is this difference that accounts for the difference in the *SS*. In fact, if we were to perform an ANOVA on the above data, we would get the following result:

	MAIN EFFECT				
	SS	**df**	**MS**	**F**	**p**
Effect	24.0	1	24.0	24.0	.008
Error	4.0	4	1.0		

As you can see, in the above table the total *SS (28)* was partitioned into the *SS* due to *within*-group variability (*2+2=4*) and variability due to differences between means (*28-(2+2)=24*).

SS Error and SS Effect. The within-group variability (*SS*) is usually referred to as *Error* variance. This term denotes the fact that we cannot readily explain or account for it in the current design. However, the *SS Effect* we *can* explain. Namely, it is due to the differences in means between the groups. Put another way, group membership *explains* this variability because we know that it is due to the differences in means.

Significance testing. The basic idea of statistical significance testing is discussed in *Elementary Concepts*. *Elementary Concepts* also explains why very many statistical test represent ratios of explained to unexplained variability. ANOVA is a good example of this. Here, we base this test on a comparison of the variance due to the between- groups variability (called *Mean Square Effect*, or MS_{effect}) with the within- group variability (called *Mean Square Error*, or Ms_{error}; this term was first used by Edgeworth, 1885). Under the null hypothesis (that there are no mean differences between groups in the population), we would still expect some minor random fluctuation in the means for the two groups when taking small samples (as in our example). Therefore, under the null hypothesis, the variance estimated based on within-group variability should be about the same as the variance due to between-groups variability. We can compare those two estimates of variance via the *F* test (see also F Distribution), which tests whether the ratio of the two variance estimates is significantly greater than 1. In our example above, that test is highly significant, and we would in fact conclude that the means for the two groups are significantly different from each other.

Summary of the basic logic of ANOVA. To summarize the discussion up to this point, the purpose of analysis of variance is to test differences in means (for groups or variables) for statistical significance. This is accomplished by analyzing the variance, that is, by partitioning the total variance into the component that is due to true random error (i.e., within- group *SS*) and the components that are due to differences between means. These latter variance components are then tested for statistical significance, and, if significant, we reject the null hypothesis of no differences between means, and accept the alternative hypothesis that the means (in the population) are different from each other.

Dependent and independent variables. The variables that are measured (e.g., a test score) are called *dependent* variables. The variables that are manipulated or controlled (e.g., a teaching method or some other criterion used to divide observations into groups that are compared) are called *factors* or *independent* variables. For more information on this important distinction, refer to *Elementary Concepts*.

Multi-Factor ANOVA

In the simple example above, it may have occurred to you that we could have simply computed a *t* test for independent samples to arrive at the same conclusion. And, indeed, we would get the identical result if we were to compare the two groups using this test. However, ANOVA is a much more flexible and powerful technique that can be applied to much more complex research issues.

Multiple factors. The world is complex and multivariate in nature, and instances when a single variable completely explains a phenomenon are rare. For example, when trying to explore how to grow a bigger tomato, we would need to consider factors that have to do with the plants' genetic makeup, soil conditions, lighting, temperature, etc. Thus, in a typical experiment, many factors are taken into account. One important reason for using ANOVA methods rather than multiple two-group studies analyzed via *t* tests is that the former method is more *efficient*, and with fewer observations we can gain more information. Let us expand on this statement.

Controlling for factors. Suppose that in the above two-group example we introduce another grouping factor, for example, *Gender*. Imagine that in each group we have 3 males and 3 females. We could summarize this design in a 2 by 2 table:

	Experimental Group 1	**Experimental Group 2**
Males	2 3 1	6 7 5
Mean	2	6
Females	4 5 3	8 9 7
Mean	4	8

Before performing any computations, it appears that we can partition the total variance into at least 3 sources: (1) error (within-group) variability, (2) variability due to experimental group membership, and (3) variability due to gender. (Note that there is an additional source -- *interaction* -- that we will discuss shortly.) What would have happened had we not included *gender* as a factor in the study but rather computed a simple *t* test? If you compute the *SS* ignoring the *gender* factor (use the within-group means *ignoring* or *collapsing across gender*; the result is *SS*=10+10=20), you will see that the resulting within-group SS is larger than it is when we include gender (use the within- group, within-gender means to compute those *SS*; they will be equal to 2 in each group, thus the combined SS-within is equal to 2+2+2+2=8). This difference is due to the fact that the means for *males* are systematically lower than those for *females*, and this difference in means adds variability if we ignore this factor. Controlling for error variance increases the sensitivity (power) of a test. This example demonstrates another principal of ANOVA that makes it preferable over simple two-group t test studies: In ANOVA we can test each factor while controlling for all others; this is actually the reason why ANOVA is more statistically powerful (i.e., we need fewer observations to find a significant effect) than the simple *t* test.

Interaction Effects

There is another advantage of ANOVA over simple *t*-tests: ANOVA allows us to detect *interaction* effects between variables, and, therefore, to test more complex hypotheses about reality. Let us consider another example to illustrate this point. (The term *interaction* was first used by Fisher, 1926.)

Main effects, two-way interaction. Imagine that we have a sample of highly achievement-oriented students and another of achievement "avoiders." We now create two random halves in each sample, and give one half of each sample a challenging test, the other an easy test. We measure how hard the students work on the test. The means of this (fictitious) study are as follows:

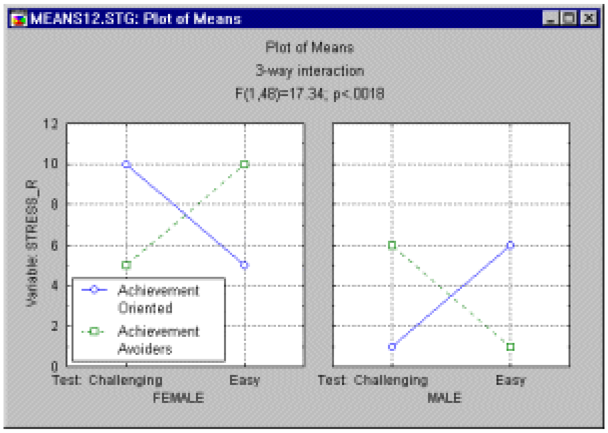
	Achievement-oriented	**Achievement-avoiders**
Challenging Test	10	5
Easy Test	5	10

How can we summarize these results? Is it appropriate to conclude that (1) challenging tests make students work harder, (2) achievement-oriented students work harder than achievement- avoiders? None of these statements captures the essence of this clearly systematic pattern of means. The appropriate way to summarize the result would be to say that challenging tests make only achievement-oriented students work harder, while easy tests make only achievement- avoiders work harder. In other words, the type of achievement orientation and test difficulty *interact* in their effect on effort; specifically, this is an example of a *two-way interaction* between achievement orientation and test difficulty. Note that statements 1 and 2 above describe so-called *main effects*.

Higher order interactions. While the previous two-way interaction can be put into words relatively easily, higher order interactions are increasingly difficult to verbalize. Imagine that we had included factor *Gender* in the achievement study above, and we had obtained the following pattern of means:

Females	Achievement-oriented	Achievement-avoiders
Challenging Test	10	5
Easy Test	5	10
Males	**Achievement-oriented**	**Achievement-avoiders**
Challenging Test	1	6
Easy Test	6	1

How could we now summarize the results of our study? Graphs of means for all effects greatly facilitate the interpretation of complex effects. The pattern shown in the table above (and in the graph below) represents a *three-way* interaction between factors.



Thus we may summarize this pattern by saying that for females there is a two-way interaction between achievement-orientation type and test difficulty: Achievement-oriented females work harder on challenging tests than on easy tests, achievement-avoiding females work harder on easy tests than on difficult tests. For males, this interaction is reversed. As you can see, the description of the interaction has become much more involved.

A general way to express interactions. A general way to express all interactions is to say that an effect is modified (qualified) by another effect. Let us try this with the two-way interaction above. The main effect for test difficulty is modified by achievement orientation. For the three-way interaction in the previous paragraph, we may summarize that the two-way interaction between test difficulty and achievement orientation is modified (qualified) by *gender*. If we have a four-way interaction, we may say that the three-way interaction is modified by the fourth variable, that is, that there are different types of interactions in the different levels of the fourth variable. As it turns out, in many areas of research five- or higher- way interactions are not that uncommon.

Complex Designs

Let us review the basic "building blocks" of complex designs.

- Between-Groups and Repeated Measures
- Incomplete (Nested) Designs

For more introductory topics, click on the topic name.

- Basic Ideas
- Analysis of Covariance (ANCOVA)
- Multivariate Designs: MANOVA/MANCOVA
- Contrast Analysis and Post hoc Tests
- Assumptions and Effects of Violating Assumptions

See also Methods for Analysis of Variance, Variance Components and Mixed Model ANOVA/ANCOVA, and Experimental Design (DOE).

Between-Groups and Repeated Measures

When we want to compare two groups, we would use the *t* test for independent samples; when we want to compare two variables given the same subjects (observations), we would use the *t* test for dependent samples. This distinction -- dependent and independent samples -- is important for ANOVA as well. Basically, if we have repeated measurements of the same variable (under different conditions or at different points in time) *on the same subjects*, then the factor is a *repeated measures factor* (also called a *within-subjects factor*, because to estimate its significance we compute the within-subjects *SS*). If we compare different groups of subjects (e.g., males and females; three strains of bacteria, etc.) then we refer to the factor as a *between-groups factor*. The computations of significance tests are different for these different types of factors; however, the logic of computations and interpretations is the same.

Between-within designs. In many instances, experiments call for the inclusion of between-groups *and* repeated measures factors. For example, we may measure math skills in male and female students (*gender*, a between-groups factor) at the beginning and the end of the semester. The two measurements *on each student* would constitute a within-subjects (repeated measures) factor. The interpretation of main effects and interactions is not affected by whether a factor is between-groups or repeated measures, and both factors may obviously interact with each other (e.g., females improve over the semester while males deteriorate).

Incomplete (Nested) Designs

There are instances where we may decide to ignore interaction effects. This happens when (1) we know that in the population the interaction effect is negligible, or (2) when a complete *factorial* design (this term was first introduced by Fisher, 1935a) cannot be used for economic reasons. Imagine a study where we want to evaluate the effect of four fuel additives on gas mileage. For our test, our company has provided us with four cars and four drivers. A complete *factorial* experiment, that is, one in which each combination of driver, additive, and car appears at least once, would require 4 x 4 x 4 = 64 individual test conditions (groups). However, we may not have the resources (time) to run all of these conditions; moreover, it seems unlikely that the type of driver would interact with the fuel additive to an extent that would be of practical relevance. Given these considerations, one could actually run a so-called *Latin square* design and "get away" with only 16 individual groups (the four additives are denoted by letters A, B, C, and D):

	Car			
	1	**2**	**3**	**4**
Driver 1	A	B	C	D
Driver 2	B	C	D	A
Driver 3	C	D	A	B
Driver 4	D	A	B	C

Latin square designs (this term was first used by Euler, 1782) are described in most textbooks on experimental methods (e.g., Hays, 1988; Lindman, 1974; Milliken & Johnson, 1984; Winer, 1962), and we do not want to discuss here the details of how they are constructed. Suffice it to say that this design is *incomplete* insofar as not all combinations of factor levels occur in the design. For example, Driver 1 will only drive Car 1 with additive A, while Driver 3 will drive that car with additive C. In a sense, the levels of the *additives* factor (A, B, C, and D) are placed into the cells of the *car* by *driver* matrix like "eggs into a nest." This mnemonic device is sometimes useful for remembering the nature of *nested* designs.

Note that there are several other statistical procedures which may be used to analyze these types of designs; see the section on Methods for Analysis of Variance for details. In particular the methods discussed in the Variance Components and Mixed Model ANOVA/ANCOVA chapter are very efficient for analyzing designs with unbalanced nesting (when the nested factors have different numbers of levels within the levels of the factors in which they are nested), very large nested designs (e.g., with more than 200 levels overall), or hierarchically nested designs (with or without random factors).

Analysis of Covariance (ANCOVA)

General Idea

The Basic Ideas section discussed briefly the idea of "controlling" for factors and how the inclusion of additional factors can reduce the error *SS* and increase the statistical power (sensitivity) of our design. This idea can be extended to continuous variables, and when such continuous variables are included as factors in the design they are called *covariates*.

- Fixed Covariates
- Changing Covariates

For more introductory topics, see the topic name.

- Basic Ideas
- Complex Designs
- Multivariate Designs: MANOVA/MANCOVA
- Contrast Analysis and Post hoc Tests
- Assumptions and Effects of Violating Assumptions

See also Methods for Analysis of Variance, Variance Components and Mixed Model ANOVA/ANCOVA, and Experimental Design (DOE).

Fixed Covariates

Suppose that we want to compare the math skills of students who were randomly assigned to one of two alternative textbooks. Imagine that we also have data about the general intelligence (IQ) for each student in the study. We would suspect that general intelligence is related to math skills, and we can use this information to make our test more sensitive. Specifically, imagine that in each one of the two groups we can compute the correlation coefficient (see *Basic Statistics and Tables*) between IQ and math skills. Remember that once we have computed the correlation coefficient we can estimate the amount of variance in math skills that is accounted for by IQ, and the amount of (residual) variance that we cannot explain with IQ (refer also to *Elementary Concepts* and *Basic Statistics and Tables*). We may use this residual variance in the ANOVA as an estimate of the true error *SS after* controlling for IQ. If the correlation between IQ and math skills is substantial, then a large reduction in the error SS may be achieved.

Effect of a covariate on the *F* test. In the *F* test (see also F Distribution), to evaluate the statistical significance of between-groups differences, we compute the ratio of the between- groups variance (*MS_{effect}*) over the error variance (*MS_{error}*). If *MS_{error}* becomes smaller, due to the explanatory power of IQ, then the overall *F* value will become larger.

Multiple covariates. The logic described above for the case of a single covariate (IQ) can easily be extended to the case of multiple covariates. For example, in addition to IQ, we might include measures of motivation, spatial reasoning, etc., and instead of a simple correlation, compute the multiple correlation coefficient (see *Multiple Regression*).

When the *F* value gets smaller. In some studies with covariates it happens that the *F* value actually becomes smaller (less significant) after including covariates in the design. This is usually an indication that the covariates are not only correlated with the dependent variable (e.g., math skills), but also with the between-groups factors (e.g., the two different textbooks). For example, imagine that we measured IQ at the end of the semester, after the students in the different experimental groups had used the respective textbook for almost one year. It is possible that, even though students were initially randomly assigned to one of the two textbooks, the different books were so different that *both* math skills *and* IQ improved differentially in the two groups. In that case, the covariate will not only partition variance away from the error variance, but also from the variance due to the between- groups factor. Put another way, after controlling for the differences in IQ that were produced by the two textbooks, the math skills are not that different. Put in yet a third way, by "eliminating" the effects of IQ, we have inadvertently eliminated the true effect of the textbooks on students' math skills.

Adjusted means. When the latter case happens, that is, when the covariate is affected by the between-groups factor, then it is appropriate to compute so-called adjusted means. These are the means that one would get after removing all differences that can be accounted for by the covariate.

Interactions between covariates and factors. Just as we can test for interactions between factors, we can also test for the interactions between covariates and between-groups factors. Specifically, imagine that one of the textbooks is particularly suited for intelligent students, while the other actually bores those students but challenges the less intelligent ones. As a result, we may find a positive correlation in the first group (the more intelligent, the better the performance), but a zero or slightly negative correlation in the second group (the more intelligent the student, the less likely he or she is to acquire math skills from the particular textbook). In some older statistics textbooks this condition is discussed as a case where the assumptions for analysis of covariance are violated (see Assumptions and Effects of Violating Assumptions). However, because ANOVA/MANOVA uses a very general approach to analysis of covariance, you can specifically estimate the statistical significance of interactions between factors and covariates.

Changing Covariates

While fixed covariates are commonly discussed in textbooks on ANOVA, changing covariates are discussed less frequently. In general, when we have repeated measures, we are interested in testing the differences in repeated measurements on the same subjects. Thus we are actually interested in evaluating the significance of *changes*. If we have a covariate that is also measured at each point when the dependent variable is measured, then we can compute the correlation between the changes in the covariate and the changes in the dependent variable. For example, we could study math anxiety and math skills at the beginning and at the end of the semester. It would be interesting to see whether any changes in math anxiety over the semester correlate with changes in math skills.

Multivariate Designs: MANOVA/MANCOVA

- Between-Groups Designs
- Repeated Measures Designs
- Sum Scores versus MANOVA

For more introductory topics, see the topic name.

- Basic Ideas
- Complex Designs
- Analysis of Covariance (ANCOVA)
- Contrast Analysis and Post hoc Tests
- Assumptions and Effects of Violating Assumptions

See also Methods for Analysis of Variance, Variance Components and Mixed Model ANOVA/ANCOVA, and Experimental Design (DOE).

Between-Groups Designs

All examples discussed so far have involved only one dependent variable. Even though the computations become increasingly complex, the *logic* and *nature* of the computations do not change when there is more than one dependent variable at a time. For example, we may conduct a study where we try two different textbooks, and we are interested in the students' improvements in math *and* physics. In that case, we have two dependent variables, and our hypothesis is that both together are affected by the difference in textbooks. We could now perform a multivariate analysis of variance (MANOVA) to test this hypothesis. Instead of a univariate *F* value, we would obtain a multivariate *F* value (Wilks' *lambda*) based on a comparison of the error variance/covariance matrix and the effect variance/covariance matrix. The "covariance" here is included because the two measures are probably correlated and we must take this correlation into account when performing the significance test. Obviously, if we were to take the *same* measure twice, then we would really not learn anything new. If we take a correlated measure, we gain *some* new information, but the new variable will also contain redundant information that is expressed in the covariance between the variables.

Interpreting results. If the overall multivariate test is significant, we conclude that the respective effect (e.g., textbook) is significant. However, our next question would of course be whether only math skills improved, only physics skills improved, or both. In fact, after obtaining a significant multivariate test for a particular main effect or interaction, customarily one would examine the univariate *F* tests (see also F Distribution) for each variable to interpret the respective effect. In other words, one would identify the specific dependent variables that contributed to the significant overall effect.

Repeated Measures Designs

If we were to measure math and physics skills at the beginning of the semester and the end of the semester, we would have a multivariate repeated measure. Again, the logic of significance testing in such designs is simply an extension of the univariate case. Note that MANOVA methods are also commonly used to test the significance of *univariate* repeated measures factors with more than two levels; this application will be discussed later in this section.

Sum Scores versus MANOVA

Even experienced users of ANOVA and MANOVA techniques are often puzzled by the differences in results that sometimes occur when performing a MANOVA on, for example, three variables as compared to a univariate ANOVA on the *sum* of the three variables. The logic underlying the *summing* of variables is that each variable contains some "true" value of the variable in question, as well as some random measurement error. Therefore, by summing up variables, the measurement error will sum to approximately 0 across all measurements, and the sum score will become more and more reliable (increasingly equal to the sum of true scores). In fact, under these circumstances, ANOVA on sums is appropriate and represents a very sensitive (powerful) method. However, if the dependent variable is truly multi- dimensional in nature, then summing is inappropriate. For example, suppose that my dependent measure consists of four indicators of success *in society*, and each indicator represents a completely independent way in which a person could "make it" in life (e.g., successful professional, successful entrepreneur, successful homemaker, etc.). Now, summing up the scores on those variables would be like adding apples to oranges, and the resulting sum score will not be a reliable indicator of a single underlying dimension. Thus, one should treat such data as multivariate indicators of success in a MANOVA.

Contrast Analysis and Post hoc Tests

- Why Compare Individual Sets of Means?
- Contrast Analysis
- Post hoc Comparisons

For more introductory topics, see the topic name.

- Basic Ideas
- Complex Designs
- Analysis of Covariance (ANCOVA)
- Multivariate Designs: MANOVA/MANCOVA
- Assumptions and Effects of Violating Assumptions

See also Methods for Analysis of Variance, Variance Components and Mixed Model ANOVA/ANCOVA, and Experimental Design (DOE).

Why Compare Individual Sets of Means?

Usually, experimental hypotheses are stated in terms that are more specific than simply main effects or interactions. We may have the *specific* hypothesis that a particular textbook will improve math skills in males, but not in females, while another book would be about equally effective for both genders, but less effective overall for males. Now generally, we are predicting an interaction here: the effectiveness of the book is modified (qualified) by the student's gender. However, we have a particular prediction concerning the *nature* of the interaction: we expect a significant difference between genders for one book, but not the other. This type of specific prediction is usually tested via contrast analysis.

Contrast Analysis

Briefly, contrast analysis allows us to test the statistical significance of predicted specific differences in particular parts of our complex design. It is a major and indispensable component of the analysis of every complex ANOVA design.

Post hoc Comparisons

Sometimes we find effects in our experiment that were not expected. Even though in most cases a creative experimenter will be able to explain almost any pattern of means, it would not be appropriate to analyze and evaluate that pattern as if one had predicted it all along. The problem here is one of capitalizing on chance when performing multiple tests *post hoc*, that is, without *a priori* hypotheses. To illustrate this point, let us consider the following "experiment." Imagine we were to write down a number between 1 and 10 on 100 pieces of paper. We then put all of those pieces into a hat and draw 20 samples (of pieces of paper) of 5 observations each, and compute the means (from the numbers written on the pieces of paper) for each group. How likely do you think it is that we will find two sample means that are significantly different from each other? It is very likely! Selecting the extreme means obtained from 20 samples is very different from taking only 2 samples from the hat in the first place, which is what the test via the contrast analysis implies. Without going into further detail, there are several so-called *post hoc* tests that are explicitly based on the first scenario (taking the extremes from 20 samples), that is, they are based on the assumption that we have chosen for our comparison the most extreme (different) means out of *k* total means in the design. Those tests apply "corrections" that are designed to offset the advantage of *post hoc* selection of the most extreme comparisons.

Assumptions and Effects of Violating Assumptions

- Deviation from Normal Distribution
- Homogeneity of Variances
- Homogeneity of Variances and Covariances
- Sphericity and Compound Symmetry

For more introductory topics, see the topic name.

- Basic Ideas
- Complex Designs
- Analysis of Covariance (ANCOVA)
- Multivariate Designs: MANOVA/MANCOVA
- Contrast Analysis and Post hoc Tests

See also Methods for Analysis of Variance, Variance Components and Mixed Model ANOVA/ANCOVA, and Experimental Design (DOE).

Deviation from Normal Distribution

Assumptions. It is assumed that the dependent variable is measured on at least an interval scale level (see *Elementary Concepts*). Moreover, the dependent variable should be normally distributed within groups.

Effects of violations. Overall, the *F* test (see also F Distribution) is remarkably robust to deviations from normality (see Lindman, 1974, for a summary). If the kurtosis (see *Basic Statistics and Tables*) is greater than 0, then the *F* tends to be too small and we cannot reject the null hypothesis even though it is incorrect. The opposite is the case when the kurtosis is less than 0. The skewness of the distribution usually does not have a sizable effect on the *F* statistic. If the *n* per cell is fairly large, then deviations from normality do not matter much at all because of the *central limit theorem*, according to which the sampling distribution of the mean approximates the normal distribution, regardless of the distribution of the variable in the population. A detailed discussion of the robustness of the *F* statistic can be found in Box and Anderson (1955), or Lindman (1974).

Homogeneity of Variances

Assumptions. It is assumed that the variances in the different groups of the design are identical; this assumption is called the *homogeneity of variances* assumption. Remember that at the beginning of this section we computed the error variance (*SS* error) by adding up the sums of squares within each group. If the variances in the two groups are different from each other, then adding the two together is not appropriate, and will not yield an estimate of the common within-group variance (since no common variance exists).

Effects of violations. Lindman (1974, p. 33) shows that the *F* statistic is quite robust against violations of this assumption (*heterogeneity* of variances; see also Box, 1954a, 1954b; Hsu, 1938).

Special case: correlated means and variances. However, one instance when the *F* statistic is *very misleading* is when the means are correlated with variances across cells of the design. A scatterplot of variances or standard deviations against the means will detect such correlations. The reason why this is a "dangerous" violation is the following: Imagine that you have 8 cells in the design, 7 with about equal means but one with a much higher mean. The *F* statistic may suggest to you a statistically significant effect. However, suppose that there also is a much larger variance in the cell with the highest mean, that is, the means and the variances are correlated across cells (the higher the mean the larger the variance). In that case, the high mean in the one cell is actually quite unreliable, as is indicated by the large variance. However, because the overall *F* statistic is based on a *pooled* within-cell variance estimate, the high mean is identified as significantly different from the others, when in fact it is not at all significantly different if one based the test on the within-cell variance in that cell alone.

This pattern -- a high mean and a large variance in one cell -- frequently occurs when there are *outliers* present in the data. One or two extreme cases in a cell with only 10 cases can greatly bias the mean, and will dramatically increase the variance.

Homogeneity of Variances and Covariances

Assumptions. In multivariate designs, with multiple dependent measures, the homogeneity of variances assumption described earlier also applies. However, since there are multiple dependent variables, it is also required that their intercorrelations (covariances) are homogeneous across the cells of the design. There are various specific tests of this assumption.

Effects of violations. The multivariate equivalent of the *F* test is Wilks' *lambda*. Not much is known about the robustness of Wilks' *lambda* to violations of this assumption.

However, because the interpretation of MANOVA results usually rests on the interpretation of significant *univariate* effects (after the overall test is significant), the above discussion concerning univariate ANOVA basically applies, and important significant univariate effects should be carefully scrutinized.

Special case: ANCOVA. A special serious violation of the homogeneity of variances/covariances assumption may occur when covariates are involved in the design. Specifically, if the correlations of the covariates with the dependent measure(s) are very different in different cells of the design, gross misinterpretations of results may occur. Remember that in ANCOVA, we in essence perform a regression analysis within each cell to partition out the variance component due to the covariates. The homogeneity of variances/covariances assumption implies that we perform this regression analysis subject to the constraint that all regression equations (slopes) across the cells of the design are the same. If this is not the case, serious biases may occur. There are specific tests of this assumption, and it is advisable to look at those tests to ensure that the regression equations in different cells are approximately the same.

Sphericity and Compound Symmetry

Reasons for Using the Multivariate Approach to Repeated Measures ANOVA. In repeated measures ANOVA containing repeated measures factors with more than two levels, additional special assumptions enter the picture: The *compound symmetry* assumption and the assumption of *sphericity*. Because these assumptions rarely hold (see below), the MANOVA approach to repeated measures ANOVA has gained popularity in recent years (both tests are automatically computed in *ANOVA/MANOVA*). The *compound symmetry* assumption requires that the variances (pooled within-group) and covariances (across subjects) of the different repeated measures are homogeneous (identical). This is a *sufficient* condition for the univariate *F* test for repeated measures to be valid (i.e., for the reported *F* values to actually follow the *F* distribution). However, it is not a *necessary* condition. The *sphericity* assumption is a necessary and sufficient condition for the *F* test to be valid; it states that the *within-subject* "model" consists of independent (orthogonal) components. The nature of these assumptions, and the effects of violations are usually not well-described in ANOVA textbooks; in the following paragraphs we will try to clarify this matter and explain what it means when the results of the univariate approach differ from the multivariate approach to repeated measures ANOVA.

The necessity of independent hypotheses. One general way of looking at ANOVA is to consider it a *model fitting* procedure. In a sense we bring to our data a set of *a priori* hypotheses; we then partition the variance (test main effects, interactions) to test those hypotheses. Computationally, this approach translates into generating a set of contrasts (comparisons between means in the design) that specify the main effect and interaction hypotheses. However, if these contrasts are not independent of each other, then the partitioning of variances runs afoul. For example, if two contrasts *A* and *B* are identical to each other and we partition out their components from the total variance, then we take the same thing out twice. Intuitively, specifying the two (*not* independent) hypotheses "the mean in Cell 1 is higher than the mean in Cell 2" *and* "the mean in Cell 1 is higher than the mean in Cell 2" is silly and simply makes no sense. Thus, hypotheses must be independent of each other, or *orthogonal* (the term *orthogonality* was first used by Yates, 1933).

Independent hypotheses in repeated measures. The general algorithm implemented will attempt to generate, for each effect, a set of independent (orthogonal) contrasts. In repeated measures ANOVA, these contrasts specify a set of hypotheses about *differences* between the levels of the repeated measures factor. However, if these differences are correlated across subjects, then the resulting contrasts are no longer independent. For example, in a study where we measured learning at three times during the experimental session, it may happen that the changes from time 1 to time 2 are negatively correlated with the changes from time 2 to time 3: subjects who learn most of the material between time 1 and time 2 improve less from time 2 to time 3. In fact, in most instances where a repeated measures ANOVA is used, one would probably suspect that the changes across levels are correlated across subjects. However, when this happens, the compound symmetry and sphericity assumptions have been violated, and independent contrasts cannot be computed.

Effects of violations and remedies. When the compound symmetry or sphericity assumptions have been violated, the univariate ANOVA table will give erroneous results. Before multivariate procedures were well understood, various approximations were introduced to compensate for the violations (e.g., Greenhouse & Geisser, 1959; Huynh & Feldt, 1970), and these techniques are still widely used.

MANOVA approach to repeated measures. To summarize, the problem of compound symmetry and sphericity pertains to the fact that multiple contrasts involved in testing repeated measures effects (with more than two levels) are not independent of each other. However, they do not need to be independent of each other if we use *multivariate* criteria to simultaneously test the statistical significance of the two or more repeated measures contrasts. This "insight" is the reason why MANOVA methods are increasingly applied to test the significance of univariate repeated measures factors with more than two levels. We wholeheartedly endorse this approach because it simply bypasses the assumption of compound symmetry and sphericity altogether.

Cases when the MANOVA approach cannot be used. There are instances (designs) when the MANOVA approach cannot be applied; specifically, when there are few subjects in the design and many levels on the repeated measures factor, there may not be enough degrees of freedom to perform the multivariate analysis. For example, if we have 12 subjects and $p = 4$ repeated measures factors, each at $k = 3$ levels, then the four-way interaction would "consume" $(k-1)^p = 2^4 = 16$ degrees of freedom. However, we have only 12 subjects, so in this instance the multivariate test cannot be performed.

Differences in univariate and multivariate results. Anyone whose research involves extensive repeated measures designs has seen cases when the univariate approach to repeated measures ANOVA gives clearly different results from the multivariate approach. To repeat the point, this means that the differences between the levels of the respective repeated measures factors are in some way correlated across subjects. Sometimes, this insight by itself is of considerable interest.

Methods for Analysis of Variance

Several chapters in this textbook discuss methods for performing analysis of variance. Although many of the available statistics overlap in the different chapters, each is best suited for particular applications.

General ANCOVA/MANCOVA: This chapter includes discussions of full factorial designs, repeated measures designs, mutivariate design (MANOVA), designs with balanced nesting (designs can be unbalanced, i.e., have unequal *n*), for evaluating planned and post-hoc comparisons, etc.

General Linear Models: This extremely comprehensive chapter discusses a complete implementation of the general linear model, and describes the sigma-restricted as well as the overparameterized approach. This chapter includes information on incomplete designs, complex analysis of covariance designs, nested designs (balanced or unbalanced), mixed model ANOVA designs (with random effects), and huge balanced ANOVA designs (efficiently). It also contains descriptions of six types of Sums of Squares.

General Regression Models: This chapter discusses the between subject designs and multivariate designs which are appropriate for stepwise regression as well as discussing how to perform stepwise and best-subset model building (for continuous as well as categorical predictors).

Mixed ANCOVA and Variance Components: This chapter includes discussions of experiments with random effects (mixed model ANOVA), estimating variance components for random effects, or large main effect designs (e.g., with factors with over 100 levels) with or without random effects, or large designs with many factors, when you do not need to estimate all interactions.

Experimental Design (DOE): This chapter includes discussions of standard experimental designs for industrial/manufacturing applications, including 2**(k-p) and 3**(k-p) designs, central composite and non-factorial designs, designs for mixtures, D and A optimal designs, and designs for arbitrarily constrained experimental regions.

Repeatability and Reproducibility Analysis (in the *Process Analysis* chapter): This section in the *Process Analysis* chapter includes a discussion of specialized designs for evaluating the reliability and precision of measurement systems; these designs usually include two or three random factors, and specialized statistics can be computed for evaluating the quality of a measurement system (typically in industrial/manufacturing applications).

Breakdown Tables (in the *Basic Statistics* chapter): This chapter includes discussions of experiments with only one factor (and many levels), or with multiple factors, when a complete ANOVA table is not required.

Chapter 4
Association Rules

- Association Rules Introductory Overview
- Computational Procedures and Terminology
- Tabular Representation of Associations
- Graphical Representation of Associations
- Interpreting and Comparing Results

Association Rules Introductory Overview

The goal of the techniques described in this section is to detect relationships or associations between specific values of categorical variables in large data sets. This is a common task in many data mining projects as well as in the data mining subcategory text mining. These powerful exploratory techniques have a wide range of applications in many areas of business practice and also research - from the analysis of consumer preferences or human resource management, to the history of language. These techniques enable analysts and researchers to uncover hidden patterns in large data sets, such as "customers who order product *A* often also order product *B* or *C*" or "employees who said positive things about initiative *X* also frequently complain about issue *Y* but are happy with issue *Z*." The implementation of the so-called a-priori algorithm (see Agrawal and Swami, 1993; Agrawal and Srikant, 1994; Han and Lakshmanan, 2001; see also Witten and Frank, 2000) allows you to process rapidly huge data sets for such associations, based on predefined "threshold" values for detection.

How association rules work. The usefulness of this technique to address unique data mining problems is best illustrated in a simple example. Suppose you are collecting data at the check-out cash registers at a large book store. Each customer transaction is logged in a database, and consists of the titles of the books purchased by the respective customer, perhaps additional magazine titles and other gift items that were purchased, and so on. Hence, each record in the database will represent one customer (transaction), and may consist of a single book purchased by that customer, or it may consist of many (perhaps hundreds of) different items that were purchased, arranged in an arbitrary order depending on the order in which the different items (books, magazines, and so on) came down the conveyor belt at the cash register. The purpose of the analysis is to find associations between the items that were purchased, i.e., to derive association rules that identify the items and co-occurrences of different items that appear with the greatest (co-)frequencies. For example, you want to learn which books are likely to be purchased by a customer who you know already purchased (or is about to purchase) a particular book. This type of information could then quickly be used to suggest to the customer those additional titles. You may already be "familiar" with the results of these types of analyses, if you are a customer of various on-line (Web-based) retail businesses; many times when making a purchase on-line, the vendor will suggest similar items (to the ones purchased by you) at the time of "check-out", based on some rules such as "customers who buy book title *A* are also likely to purchase book title *B*," and so on.

Unique data analysis requirements. Crosstabulation tables, and in particular Multiple Response tables can be used to analyze data of this kind. However, in cases when the number of different items (categories) in the data is very large (and not known ahead of time), and when the "factorial degree" of important association rules is not known ahead of time, then these tabulation facilities may be too cumbersome to use, or simply not applicable: Consider once more the simple "bookstore-example" discussed earlier. First, the number of book titles is practically unlimited. In other words, if we would make a table where each book title would represent one dimension, and the purchase of that book (yes/no) would be the classes or categories for each dimension, then the complete crosstabulation table would be huge and sparse (consisting mostly of empty cells). Alternatively, we could construct all possible two-way tables from all items available in the store; this would allow us to detect two-way associations (association rules) between items. However, the number of tables that would have to be constructed would again be huge, most of the two-way tables would be sparse, and worse, if there were any three-way association rules "hiding" in the data, we would miss them completely. The a-priori algorithm implemented in *Association Rules* will not only automatically detect the relationships ("cross-tabulation tables") that are important (i.e., cross-tabulation tables that are not sparse, not containing mostly zero's), but also determine the factorial degree of the tables that contain the important association rules.

To summarize, *Association Rules* will allow you to find rules of the kind *If X then (likely) Y* where *X* and *Y* can be single values, items, words, etc., or conjunctions of values, items, words, etc. (e.g., *if (Car=Porsche and Gender=Male and Age<20) then (Risk=High and Insurance=High)*). The program can be used to analyze simple categorical variables, dichotomous variables, and/or multiple response variables. The algorithm will determine association rules without requiring the user to specify the number of distinct categories present in the data, or any prior knowledge regarding the maximum factorial degree or complexity of the important associations. In a sense, the algorithm will construct cross-tabulation tables without the need to specify the number of dimensions for the tables, or the number of categories for each dimension. Hence, this technique is particularly well suited for data and text mining of huge databases.

Computational Procedures and Terminology

Categorical or class variables. Categorical variables are single variables that contains codes or text values to denote distinct classes; for example, a variable *Gender* would have the categories *Male* and *Female*.

Multiple response variables. Multiple response variables usually consist of multiple variables (i.e., a list of variables) that can contain, for each observations, codes or text values describing a single "dimension" or transaction. A good example of a multiple response variable would be if a vendor recorded the purchases made by a customer in a single record, where each record could contain one or more items purchased, in arbitrary order. This is a typical format in which customer transaction data would be kept.

Multiple dichotomies. In this data format, each variable would represent one item or category, and the dichotomous data in each variable would indicate whether or not the respective item or category applies to the respective case. For example, suppose a vendor created a data spreadsheet where each column represented one of the products available for purchase. Each transaction (row of the data spreadsheet) would record whether or not the respective customer did or did not purchase that product, i.e., whether or not the respective transaction involved each item.

Association Rules: If Body then Head. The A-priori algorithm attempts to derive from the data association rules of the form: *If "Body" then "Head"*, where *Body* and *Head* stand for simple codes or text values (items), or the conjunction of codes and text values (items; e.g., *if (Car=Porsche and Age<20) then (Risk=High and Insurance=High)*; here the logical conjunction before the then would be the *Body*, and the logical conjunction following the then would be the *Head* of the association rule).

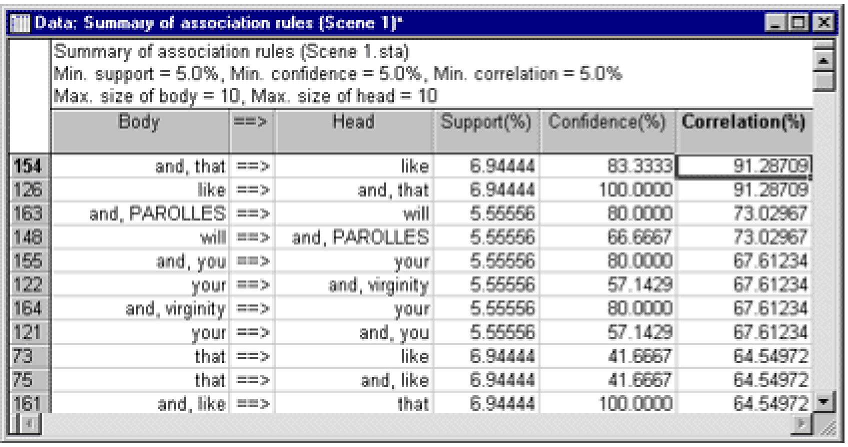
Initial Pass Through the Data: The Support Value. First the program will scan all variables to determine the unique codes or text values (items) found in the variables selected for the analysis. In this initial pass, the relative frequencies with which the individual codes or text values occur in each transaction will also be computed. The probability that a transaction contains a particular code or text value is called *Support*; the *Support* value is also computed in consecutive passes through the data, as the joint probability (relative frequency of co-occurrence) of pairs, triplets, etc. of codes or text values (items), i.e., separately for the *Body* and *Head* of each association rule.

Second Pass Through the Data: The Confidence Value; Correlation Value. After the initial pass through the data, all items with a support value less than some predefined minimum support value will be "remembered" for subsequent passes through the data: Specifically, the conditional probabilities will be computed for all pairs of codes or text values that have support values greater than the minimum support value. This conditional probability - that an observation (transaction) that contains a code or text value *X* also contains a code or text value *Y* -- is called the *Confidence Value*. In general (in later passes through the data) the confidence value denotes the conditional probability of the *Head* of the association rule, given the *Body* of the association rule.

In addition, the support value will be computed for each pair of codes or text values, and a *Correlation* value based on the support values. The correlation value for a pair of codes or text values {*X*, *Y*} is computed as the support value for that pair, divided by the square root of the product of the support values for *X* and *Y*. After the second pass through the data those pairs of codes or text values that (1) have a confidence value that is greater than some user-defined minimum confidence value, (2) have a support value that is greater than some user-defined minimum support value, and (3) have a correlation value that is greater than some minimum correlation value will be retained.

Subsequent Passes Through The Data: Maximum Item Size in Body, Head. The data in subsequent steps, the data will be further scanned computing support, confidence, and correlation values for pairs of codes or text values (associations between single codes or text values), triplets of codes or text values, and so on. To reiterate, in general, at each association rules will be derived of the general form if "*Body*" then "*Head*", where *Body* and *Head* stand for simple codes or text values (items), or the conjunction of codes and text values (items).

Unless the process stops because no further associations can be found that satisfy the minimum support, confidence, and correlation conditions, the process could continue to build very complex association rules (e.g., *if X1 and X2 .. and X20 then Y1 and Y2 ... and Y20*). To avoid excessive complexity, additionally, the user can specify the maximum number of codes or text values (items) in the *Body* and *Head* of the association rules; this value is referred to as the maximum item set size in the *Body* and *Head* of an association rule.



Data: Summary of association rules (Scene 1)*

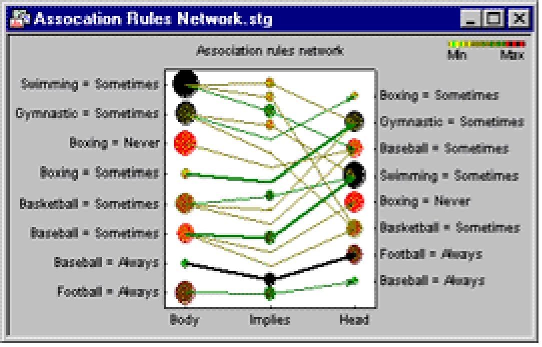
Summary of association rules (Scene 1.sta)
Min. support = 5.0%, Min. confidence = 5.0%, Min. correlation = 5.0%
Max. size of body = 10, Max. size of head = 10

	Body	==>	Head	Support(%)	Confidence(%)	Correlation(%)
154	and, that	==>	like	6.94444	83.3333	91.28709
126	like	==>	and, that	6.94444	100.0000	91.28709
163	and, PAROLLES	==>	will	5.55556	80.0000	73.02967
148	will	==>	and, PAROLLES	5.55556	66.6667	73.02967
155	and, you	==>	your	5.55556	80.0000	67.61234
122	your	==>	and, virginity	5.55556	57.1429	67.61234
164	and, virginity	==>	your	5.55556	80.0000	67.61234
121	your	==>	and, you	5.55556	57.1429	67.61234
73	that	==>	like	6.94444	41.6667	64.54972
75	that	==>	and, like	6.94444	41.6667	64.54972
161	and, like	==>	that	6.94444	100.0000	64.54972

Tabular Representation of Associations

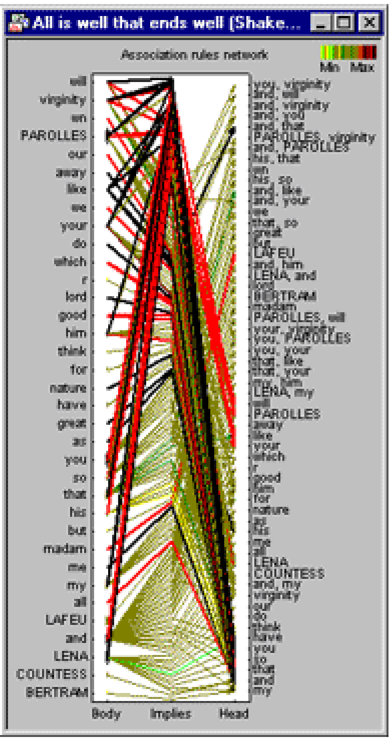
Association rules are generated of the general form *if Body then Head*, where *Body* and *Head* stand for single codes or text values (items) or conjunctions of codes or text values (items; e.g., *if (Car=Porsche and Age<20) then (Risk=High and Insurance=High)*. The major statistics computed for the association rules are *Support* (relative frequency of the *Body* or *Head* of the rule), *Confidence* (conditional probability of the *Head* given the *Body* of the rule), and *Correlation* (support for *Body* and *Head*, divided by the square root of the product of the support for the *Body* and the support for the *Head*). These statistics can be summarized in a spreadsheet, as shown below.

This results spreadsheet shows an example of how association rules can be applied to text mining tasks. This analysis was performed on the paragraphs (dialog spoken by the characters in the play) in the first scene of Shakespeare's "All's Well That Ends Well," after removing a few very frequent words like *is*, *of*, etc. The values for support, confidence, and correlation are expressed in percent.



Graphical Representation of Associations

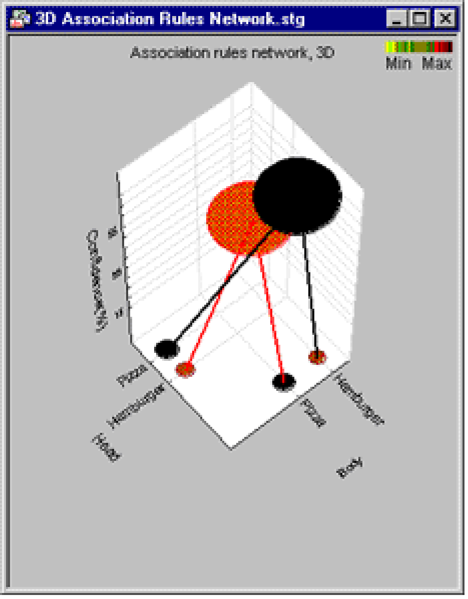
As a result of applying Association Rules data mining techniques to large datasets rules of the form *if "Body" then "Head"* will be derived, where *Body* and *Head* stand for simple codes or text values (items), or the conjunction of codes and text values (items; e.g., *if (Car=Porsche and Age<20) then (Risk=High and Insurance=High)*). These rules can be reviewed in textual format or tables, or in graphical format (see below).



Association Rules Networks, 2D. For example, consider the data that describe a (fictitious) survey of 100 patrons of sports bars and their preferences for watching various sports on television. This would be an example of simple categorical variables, where each variable represents one sport. For each sport, each respondent indicated how frequently s/he watched the respective type of sport on television. The association rules derived from these data could be summarized as follows:

In this graph, the support values for the *Body* and *Head* portions of each association rule are indicated by the sizes and colors of each. The thickness of each line indicates the confidence value (conditional probability of Head given Body) for the respective association rule; the sizes and colors of the circles in the center, above the *Implies* label, indicate the joint support (for the co-occurences) of the respective *Body* and *Head* components of the respective association rules. Hence, in this graphical summary, the strongest support value was found for *Swimming=Sometimes*, which was associated *Gymnastic=Sometimes*, *Baseball = Sometimes*, and *Basketball=Sometimes*. Incidentally. Unlike simple frequency and crosstabulation tables, the absolute frequencies with which individual codes or text values (items) occur in the data are often not reflected in the association rules; instead, only those codes or text values (items) are retained that show sufficient values for support, confidence, and correlation, i.e., that co-occur with other codes or text values (items) with sufficient relative (co-)frequency.

The results that can be summarized in 2D Association Rules networks can be relatively simple, or complex, as illustrated in the network shown to the left.


3D Association Rules Network.stg
Association rules network, 3D
Min
Max
Confidence(%)
Pizza
Hamburger
Head
Pizza
Hamburger
Body

This is an example of how association rules can be applied to text mining tasks. This analysis was performed on the paragraphs (dialog spoken by the characters in the play) in the first scene of Shakespeare's "All's Well That Ends Well," after removing a few very frequent words like *is*, *of*, etc. Of course, the specific words and phrases removed during the data preparation phase of text (or data) mining projects will depend on the purpose of the research.

Association Rules Networks, 3D. Association rules can be graphically summarized in 2D Association Networks, as well as 3D Association Networks. Shown below are some (very clear) results from an analysis. Respondents in a survey were asked to list their (up to) 3 favorite fast-foods. The association rules derived from those data are summarized in a 3D Association Network display.

As in the 2D Association Network, the support values for the *Body* and *Head* portions of each association rule are indicated by the sizes and colors of each circle in the 2D. The thickness of each line indicates the confidence value (joint probability) for the respective association rule; the sizes and colors of the "floating" circles plotted against the (vertical) z-axis indicate the joint support (for the co-occurences) of the respective *Body* and *Head* components of the association rules. The plot position of each circle along the vertical z - axis indicates the respective confidence value. Hence, this particular graphical summary clearly shows two simple rules: Respondents who name *Pizza* as a preferred fast food also mention *Hamburger*, and vice versa.

Interpreting and Comparing Results

When comparing the results of applying association rules to those from simple frequency or cross-tabulation tables, you may notice that in some cases very high-frequency codes or text values (items) are not part of any association rule. This can sometimes be perplexing.

To illustrate how this pattern of findings can occur, consider this example: Suppose you analyzed data from a survey of insurance rates for different makes of automobiles in America. Simple tabulation would very likely show that many people drive automobiles manufactured by Ford, GM, and Chrysler; however, none of these makes may be associated with particular patterns in insurance rates, i.e., none of these brands may be involved in high-confidence, high-correlation association rules linking them to particular categories of insurance rates. However, when applying association rules methods, automobile makes which occur in the sample with relatively low frequency (e.g., Porsche) may be found to be associated with high insurance rates (allowing you to infer, for example, a rule that *if Car=Porsche then Insurance=High*). If you only reviewed a simple cross-tabulation table (make of car by insurance rate) this high-confidence association rule may well have gone unnoticed.

Chapter 5
Boosting Trees for Regression and Classification

- Boosting Trees for Regression and Classification Introductory Overview
- Gradient Boosting Trees
- The Problem of Overfitting; Stochastic Gradient Boosting
- Stochastic Gradient Boosting Trees and Classification
- Large Numbers of Categories

Boosting Trees for Regression and Classification Introductory Overview

The general computational approach of stochastic gradient boosting is also known by the names TreeNet (TM Salford Systems, Inc.) and MART (TM Jerill, Inc.). Over the past few years, this technique has emerged as one of the most powerful methods for predictive data mining. Some implementations of these powerful algorithms allow them to be used for regression as well as classification problems, with continuous and/or categorical predictors. Detailed technical descriptions of these methods can be found in Friedman (1999a, b) as well as Hastie, Tibshirani, & Friedman (2001).

Gradient Boosting Trees

The algorithm for Boosting Trees evolved from the application of boosting methods to regression trees. The general idea is to compute a sequence of (very) simple trees, where each successive tree is built for the prediction residuals of the preceding tree. As described in the General Classification and Regression Trees Introductory Overview, this method will build binary trees, i.e., partition the data into two samples at each split node. Now suppose that you were to limit the complexities of the trees to 3 nodes only: a root node and two child nodes, i.e., a single split. Thus, at each step of the boosting (boosting trees algorithm), a simple (best) partitioning of the data is determined, and the deviations of the observed values from the respective means (residuals for each partition) are computed. The next 3-node tree will then be fitted to those residuals, to find another partition that will further reduce the residual (error) variance for the data, given the preceding sequence of trees.

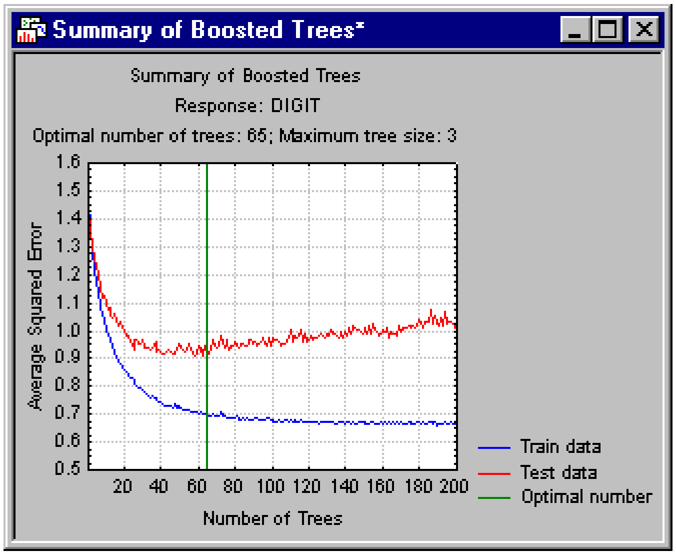
It can be shown that such "additive weighted expansions" of trees can eventually produce an excellent fit of the predicted values to the observed values, even if the specific nature of the relationships between the predictor variables and the dependent variable of interest is very complex (nonlinear in nature). Hence, the method of gradient boosting - fitting a weighted additive expansion of simple trees - represents a very general and powerful machine learning algorithm.

The Problem of Overfitting; Stochastic Gradient Boosting

One of the major problems of all machine learning algorithms is to "know when to stop," i.e., how to prevent the learning algorithm to fit esoteric aspects of the training data that are not likely to improve the predictive validity of the respective model. This issue is also known as the problem of overfitting. To reiterate, this is a general problem applicable to most machine learning algorithms used in predictive data mining. A general solution to this problem is to evaluate the quality of the fitted model by predicting observations in a test-sample of data that have not been used before to estimate the respective model(s). In this manner, one hopes to gage the predictive accuracy of the solution, and to detect when overfitting has occurred (or is starting to occur).

A similar approach is for each consecutive simple tree to be built for only a randomly selected subsample of the full data set. In other words, each consecutive tree is built for the prediction residuals (from all preceding trees) of an independently drawn random sample. The introduction of a certain degree of randomness into the analysis in this manner can serve as a powerful safeguard against overfitting (since each consecutive tree is built for a different sample of observations), and yield models (additive weighted expansions of simple trees) that generalize well to new observations, i.e., exhibit good predictive validity. This technique, i.e., performing consecutive boosting computations on independently drawn samples of observations, is knows as stochastic gradient boosting.

Below is a plot of the prediction error function for the training data over successive trees and also an independently sampled testing data set at each stage.


Summary of Boosted Trees

Response: DIGIT

Optimal number of trees: 65; Maximum tree size: 3

With this graph, you can identify very quickly the point where the model (consisting of a certain number of successive trees) begins to overfit the data. Notice how the prediction error for the training data steadily decreases as more and more additive terms (trees) are added to the model. However, somewhere past 35 trees, the performance for independently sampled testing data actually begins to deteriorate, clearly indicating the point where the model begins to overfit the data.

Stochastic Gradient Boosting Trees and Classification

So far, the discussion of boosting trees has exclusively focused on regression problems, i.e., on the prediction of a continuous dependent variable. The technique can easily be expanded to handle classification problems as well (this is described in detail in Friedman, 1999a, section 4.6; in particular, see Algorithm 6):

First, different boosting trees are built for (fitted to) each category or class of the categorical dependent variable, after creating a coded variable (vector) of values for each class with the values 1 or 0 to indicate whether or not an observation does or does not belong to the respective class. In successive boosting steps, the algorithm will apply the logistic transformation (see also *Nonlinear Estimation*) to compute the residuals for subsequent boosting steps. To compute the final classification probabilities, the logistic transformation is again applied to the predictions for each 0/1 coded vector (class). This algorithm is described in detail in Friedman (1999a; see also Hastie, Tibshirani, and Freedman, 2001, for a description of this general procedure).

Large Numbers of Categories

Note that the procedure for applying this method to classification problems requires that separate sequences of (boosted) trees be built for each category or class. Hence, the computational effort generally becomes larger by a multiple of what it takes to solve a simple regression prediction problem (for a single continuous dependent variable). Therefore, it is not prudent to analyze categorical dependent variables (class variables) with more than, approximately, 100 or so classes; past that point, the computations performed may require an unreasonable amount of effort and time. (For example, a problem with 200 boosting steps and 100 categories or classes for the dependent variable would yield 200 * 100 = 20,000 individual trees!)

Chapter 6
Canonical Analysis

- General Purpose
- Computational Methods and Results
- Assumptions
- General Ideas
- Sum Scores
- Canonical Roots/Variates
- Number of Roots
- Extraction of Roots

General Purpose

There are several measures of correlation to express the relationship between two or more variables. For example, the standard Pearson product moment correlation coefficient (*r*) measures the extent to which two variables are related; there are various nonparametric measures of relationships that are based on the similarity of ranks in two variables; *Multiple Regression* allows one to assess the relationship between a dependent variable and a set of independent variables; Multiple Correspondence Analysis is useful for exploring the relationships between a set of categorical variables.

Canonical Correlation is an additional procedure for assessing the relationship between variables. Specifically, this analysis allows us to investigate the relationship between *two sets* of variables. For example, an educational researcher may want to compute the (simultaneous) relationship between three measures of scholastic ability with five measures of success in school. A sociologist may want to investigate the relationship between two predictors of social mobility based on interviews, with actual subsequent social mobility as measured by four different indicators. A medical researcher may want to study the relationship of various risk factors to the development of a group of symptoms. In all of these cases, the researcher is interested in the relationship between two sets of variables, and *Canonical Correlation* would be the appropriate method of analysis.

In the following topics we will briefly introduce the major concepts and statistics in canonical correlation analysis. We will assume that you are familiar with the correlation coefficient as described in *Basic Statistics*, and the basic ideas of multiple regression as described in the overview section of *Multiple Regression*.

Computational Methods and Results

Some of the computational issues involved in canonical correlation and the major results that are commonly reported will now be reviewed.

Eigenvalues. When extracting the canonical roots, you will compute the *eigenvalues*. These can be interpreted as the proportion of variance accounted for by the correlation between the respective canonical variates. Note that the proportion here is computed relative to the variance of the canonical variates, that is, of the weighted sum scores of the two sets of variables; the eigenvalues do *not* tell how much variability is explained in either set of variables. You will compute as many eigenvalues as there are canonical roots, that is, as many as the minimum number of variables in either of the two sets.

Successive eigenvalues will be of smaller and smaller size. First, compute the weights that maximize the correlation of the two sum scores. After this first root has been extracted, you will find the weights that produce the second largest correlation between sum scores, subject to the constraint that the next set of sum scores does not correlate with the previous one, and so on.

Canonical correlations. If the square root of the eigenvalues is taken, then the resulting numbers can be interpreted as correlation coefficients. Because the correlations pertain to the canonical variates, they are called *canonical correlations*. Like the eigenvalues, the correlations between successively extracted canonical variates are smaller and smaller. Therefore, as an overall index of the canonical correlation between two sets of variables, it is customary to report the largest correlation, that is, the one for the first root. However, the other canonical variates can also be correlated in a meaningful and interpretable manner (see below).

Significance of Roots. The significance test of the canonical correlations is straightforward in principle. Simply stated, the different canonical correlations are tested, one by one, beginning with the largest one. Only those roots that are statistically significant are then retained for subsequent interpretation. Actually, the nature of the significance test is somewhat different. First, evaluate the significance of all roots combined, then of the roots remaining after removing the first root, the second root, etc.

Some authors have criticized this sequential testing procedure for the significance of canonical roots (e.g., Harris, 1976). However, this procedure was "rehabilitated" in a subsequent Monte Carlo study by Mendoza, Markos, and Gonter (1978).

In short, the results of that study showed that this testing procedure will detect strong canonical correlations most of the time, even with samples of relatively small size (e.g., $n = 50$). Weaker canonical correlations (e.g., $R = .3$) require larger sample sizes ($n > 200$) to be detected at least 50% of the time. Note that canonical correlations of small magnitude are often of little practical value, as they account for very little actual variability in the data. This issue, as well as the sample size issue, will be discussed shortly.

Canonical weights. After determining the number of significant canonical roots, the question arises as to how to interpret each (significant) root. Remember that each root actually represents two weighted sums, one for each set of variables. One way to interpret the "meaning" of each canonical root would be to look at the weights for each set. These weights are called the *canonical weights* .

In general, the larger the weight (i.e., the absolute value of the weight), the greater is the respective variable's unique positive or negative contribution to the sum. To facilitate comparisons between weights, the canonical weights are usually reported for the standardized variables, that is, for the *z* transformed variables with a mean of *0* and a standard deviation of *1*.

If you are familiar with multiple regression, you may interpret the canonical weights in the same manner as you would interpret the beta weights in a multiple regression equation. In a sense, they represent the *partial correlations* of the variables with the respective canonical root. If you are familiar with factor analysis, you may interpret the canonical weights in the same manner as you would interpret the *factor score coefficients*. To summarize, the canonical weights allow the user to understand the "make-up" of each canonical root, that is, it lets the user see how each variable in each set uniquely contributes to the respective weighted sum (canonical variate).

Canonical Scores. Canonical weights can also be used to compute actual values of the canonical variates; that is, you can simply use the weights to compute the respective sums. Again, remember that the canonical weights are customarily reported for the standardized (*z* transformed) variables.

Factor structure. Another way of interpreting the canonical roots is to look at the simple correlations between the canonical variates (or *factors*) and the variables in each set. These correlations are also called canonical factor *loadings*. The logic here is that variables that are highly correlated with a canonical variate have more in common with it. Therefore, you should weigh them more heavily when deriving a meaningful interpretation of the respective canonical variate. This method of interpreting canonical variates is identical to the manner in which factors are interpreted in factor analysis.

Factor structure versus canonical weights. Sometimes, the canonical weights for a variable are nearly zero, but the respective loading for the variable is very high. The opposite pattern of results may also occur. At first, such a finding may seem contradictory; however, remember that the canonical weights pertain to the unique contribution of each variable, while the canonical factor loadings represent simple overall correlations. For example, suppose you included in your satisfaction survey two items which measured basically the same thing, namely: (1) "Are you satisfied with your supervisors?" and (2) "Are you satisfied with your bosses?" Obviously, these items are very redundant. When the program computes the weights for the weighted sums (canonical variates) in each set so that they correlate maximally, it only "needs" to include one of the items to capture the essence of what they measure. Once a large weight is assigned to the first item, the contribution of the second item is redundant;

consequently, it will receive a zero or negligibly small canonical weight. Nevertheless, if you then look at the simple correlations between the respective sum score with the two items (i.e., the factor *loadings*), those may be substantial for *both*. To reiterate, the canonical weights pertain to the *unique contributions* of the respective variables with a particular weighted sum or canonical variate; the canonical factor loadings pertain to the *overall correlation* of the respective variables with the canonical variate.

Variance extracted. As discussed earlier, the canonical correlation coefficient refers to the correlation between the weighted sums of the two sets of variables. It tells nothing about how much variability (variance) each canonical root explains in the *variables*. However, you can infer the proportion of variance extracted from each set of variables by a particular root by looking at the canonical factor loadings. Remember that those loadings represent correlations between the canonical variates and the variables in the respective set. If you square those correlations, the resulting numbers reflect the *proportion* of variance accounted for in each variable. For each root, you can take the average of those proportions across variables to get an indication of how much variability is explained, on the average, by the respective canonical variate in that set of variables. Put another way, you can compute in this manner the average proportion of *variance extracted* by each root.

Redundancy. The canonical correlations can be squared to compute the proportion of variance shared by the sum scores (canonical variates) in each set. If you multiply this proportion by the proportion of variance extracted, you arrive at a measure of *redundancy*, that is, of how redundant one set of variables is, given the other set of variables. In equation form, you may express the redundancy as:

$$\text{Redundancy}_{\text{left}} = [\Sigma(\text{loadings}_{\text{left}}^2)/p]*R_c^2$$
$$\text{Redundancy}_{\text{right}} = [\Sigma(\text{loadings}_{\text{right}}^2)/q]*R_c^2$$

In these equations, p denotes the number of variables in the first (left) set of variables, and q denotes the number of variables in the second (*right*) set of variables; R_c^2 is the respective squared canonical correlation.

Note that you can compute the redundancy of the first (*left*) set of variables given the second (*right*) set, and the redundancy of the second (*right*) set of variables, given the first (*left*) set. Because successively extracted canonical roots are uncorrelated, you could sum up the redundancies across all (or only the first significant) roots to arrive at a single index of redundancy (as proposed by Stewart and Love, 1968).

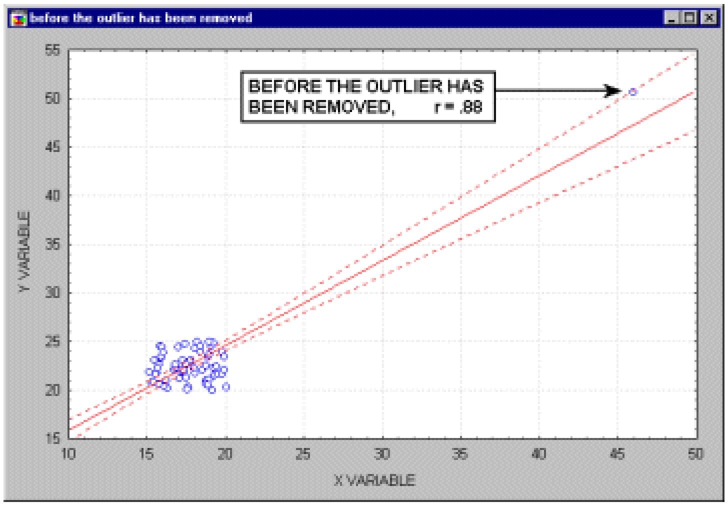
Practical significance. The measure of redundancy is also useful for assessing the *practical* significance of canonical roots. With large sample sizes (see below), canonical correlations of magnitude $R = .30$ may become statistically significant (see above). If you square this coefficient ($R\text{-}square = .09$) and use it in the redundancy formula shown above, it becomes clear that such canonical roots account for only very little variability in the variables. Of course, the final assessment of what does and does not constitute a finding of practical significance is subjective by nature. However, to maintain a realistic appraisal of how much actual variance (in the variables) is accounted for by a canonical root, it is important to always keep in mind the redundancy measure, that is, how much of the actual variability in one set of variables is explained by the other.

Assumptions

The following discussion provides only a list of the most important assumptions of canonical correlation analysis, and the major threats to the reliability and validity of results. Distributions. The tests of significance of the canonical correlations is based on the assumption that the distributions of the variables in the population (from which the sample was drawn) are multivariate normal. Little is known about the effects of violations of the multivariate normality assumption. However, with a sufficiently large sample size (see below) the results from canonical correlation analysis are usually quite robust.

Sample sizes. Stevens (1986) provides a very thorough discussion of the sample sizes that should be used in order to obtain reliable results. As mentioned earlier, if there are strong canonical correlations in the data (e.g., $R > .7$), then even relatively small samples (e.g., $n = 50$) will detect them most of the time. However, in order to arrive at reliable estimates of the canonical factor loadings (for interpretation), Stevens recommends that there should be at least 20 times as many cases as variables in the analysis, if one wants to interpret the most significant canonical root only. To arrive at reliable estimates for two canonical roots, Barcikowski and Stevens (1975) recommend, based on a Monte Carlo study, to include 40 to 60 times as many cases as variables.

Outliers. Outliers can greatly affect the magnitudes of correlation coefficients. Since canonical correlation analysis is based on (computed from) correlation coefficients, they can also seriously affect the canonical correlations. Of course, the larger the sample size, the smaller is the impact of one or two outliers. However, it is a good idea to examine various scatterplots to detect possible outliers (as shown in the example animation below).



See also Confidence Ellipse.

Matrix Ill-Conditioning. One assumption is that the variables in the two sets should not be completely redundant. For example, if you included the *same* variable twice in one of the sets, then it is not clear how to assign different weights to each of them. Computationally, such complete redundancies will "upset" the canonical correlation analysis. When there are perfect correlations in the correlation matrix, or if any of the multiple correlations between one variable and the others is perfect (*R = 1.0*), then the correlation matrix cannot be inverted, and the computations for the canonical analysis cannot be performed. Such correlation matrices are said to be *ill-conditioned*.

Once again, this assumption appears trivial on the surface; however, it often is "almost" violated when the analysis includes very many highly redundant measures, as is often the case when analyzing questionnaire responses.

General Ideas

Suppose you conduct a study in which you measure satisfaction at work with three questionnaire items, and satisfaction in various other domains with an additional seven items. The general question that you may want to answer is how satisfaction at work relates to the satisfaction in those other domains.

Sum Scores

A first approach that you might take is simply to add up the responses to the work satisfaction items, and to correlate that sum with the responses to all other satisfaction items. If the correlation between the two sums is statistically significant, we could conclude that work satisfaction is related to satisfaction in other domains.

In a way this is a rather "crude" conclusion. We still know nothing about the particular domains of satisfaction that are related to work satisfaction. In fact, we could potentially have *lost* important information by simply adding up items. For example, suppose there were two items, one measuring satisfaction with one's relationship with the spouse, the other measuring satisfaction with one's financial situation. Adding the two together is, obviously, like adding "apples to oranges." Doing so implies that a person who is dissatisfied with her finances but happy with her spouse is comparable overall to a person who is satisfied financially but not happy in the relationship with her spouse. Most likely, people's psychological make-up is not that simple...

The problem then with simply correlating two sums is that one might lose important information in the process, and, in the worst case, actually "destroy" important relationships between variables by adding "apples to oranges."

Using a weighted sum. It seems reasonable to correlate some kind of a weighted sum instead, so that the "structure" of the variables in the two sets is reflected in the weights. For example, if satisfaction with one's spouse is only marginally related to work satisfaction, but financial satisfaction is strongly related to work satisfaction, then we could assign a smaller weight to the first item and a greater weight to the second item. We can express this general idea in the following equation:

$$a_1*y_1 + a_2*y_2 + ... + a_p*y_p = b_1*x_1 + b_2*x_2 + ... + b_q*x_q$$

If we have two sets of variables, the first one containing *p* variables and the second one containing *q* variables, then we would like to correlate the weighted sums on each side of the equation with each other.

Determining the weights. We have now formulated the general "model equation" for canonical correlation. The only problem that remains is how to determine the weights for the two sets of variables. It seems to make little sense to assign weights so that the two weighted sums do not correlate with each other. A reasonable approach to take is to impose the condition that the two weighted sums shall correlate maximally with each other.

Canonical Roots/Variates

In the terminology of canonical correlation analysis, the weighted sums define a *canonical root* or *variate*. You can think of those canonical variates (weighted sums) as describing some underlying "latent" variables. For example, if for a set of diverse satisfaction items we were to obtain a weighted sum marked by large weights for all items having to do with work, we could conclude that the respective canonical variate measures satisfaction with work.

Number of Roots

So far we have pretended as if there is only one set of weights (weighted sum) that can be extracted from the two sets of variables. However, suppose that we had among our work satisfaction items particular questions regarding satisfaction with pay, and questions pertaining to satisfaction with one's social relationships with other employees. It is possible that the pay satisfaction items correlate with satisfaction with one's finances, and that the social relationship satisfaction items correlate with the reported satisfaction with one's spouse. If so, we should really derive two weighted sums to reflect this "complexity" in the structure of satisfaction.

In fact, the computations involved in canonical correlation analysis will lead to more than one set of weighted sums. To be precise, the number of roots extracted will be equal to the minimum number of variables in either set. For example, if we have three work satisfaction items and seven general satisfaction items, then three canonical roots will be extracted.

Extraction of Roots

As mentioned before, you can extract roots so that the resulting correlation between the canonical variates is maximal. When extracting more than one root, each successive root will explain a *unique* additional proportion of variability in the two sets of variables. Therefore, successively extracted canonical roots will be uncorrelated with each other, and account for less and less variability.

Chapter 7
CHAID Analysis

- General CHAID Introductory Overview
- Basic Tree-Building Algorithm: CHAID and Exhaustive CHAID
- General Computation Issues of CHAID
- CHAID, C&RT, and QUEST

General CHAID Introductory Overview

The acronym CHAID stands for *Chi*-squared Automatic Interaction Detector. It is one of the oldest tree classification methods originally proposed by Kass (1980; according to Ripley, 1996, the CHAID algorithm is a descendent of THAID developed by Morgan and Messenger, 1973). CHAID will "build" non-binary trees (i.e., trees where more than two branches can attach to a single root or node), based on a relatively simple algorithm that is particularly well suited for the analysis of larger datasets. Also, because the CHAID algorithm will often effectively yield many multi-way frequency tables (e.g., when classifying a categorical response variable with many categories, based on categorical predictors with many classes), it has been particularly popular in marketing research, in the context of market segmentation studies.

Both CHAID and C&RT techniques will construct trees, where each (non-terminal) node identifies a split condition, to yield optimum prediction (of continuous dependent or response variables) or classification (for categorical dependent or response variables). Hence, both types of algorithms can be applied to analyze regression-type problems or classification-type.

Basic Tree-Building Algorithm: CHAID and Exhaustive CHAID

The acronym CHAID stands for *Chi*-squared Automatic Interaction Detector. This name derives from the basic algorithm that is used to construct (non-binary) trees, which for classification problems (when the dependent variable is categorical in nature) relies on the *Chi*-square test to determine the best next split at each step; for regression-type problems (continuous dependent variable) the program will actually compute F-tests. Specifically, the algorithm proceeds as follows:

Preparing predictors. The first step is to create categorical predictors out of any continuous predictors by dividing the respective continuous distributions into a number of categories with an approximately equal number of observations. For categorical predictors, the categories (classes) are "naturally" defined.

Merging categories. The next step is to cycle through the predictors to determine for each predictor the pair of (predictor) categories that is least significantly different with respect to the dependent variable; for classification problems (where the dependent variable is categorical as well), it will compute a *Chi*-square test (Pearson *Chi*-square); for regression problems (where the dependent variable is continuous), F tests. If the respective test for a given pair of predictor categories is not statistically significant as defined by an alpha-to-merge value, then it will merge the respective predictor categories and repeat this step (i.e., find the next pair of categories, which now may include previously merged categories). If the statistical significance for the respective pair of predictor categories is significant (less than the respective alpha-to-merge value), then (optionally) it will compute a Bonferroni adjusted *p*-value for the set of categories for the respective predictor.

Selecting the split variable. The next step is to choose the split the predictor variable with the smallest adjusted *p*-value, i.e., the predictor variable that will yield the most significant split; if the smallest (Bonferroni) adjusted *p*-value for any predictor is greater than some alpha-to-split value, then no further splits will be performed, and the respective node is a terminal node.

Continue this process until no further splits can be performed (given the alpha-to-merge and alpha-to-split values).

CHAID and Exhaustive CHAID Algorithms. A modification to the basic CHAID algorithm, called Exhaustive CHAID, performs a more thorough merging and testing of predictor variables, and hence requires more computing time. Specifically, the merging of categories continues (without reference to any alpha-to-merge value) until only two categories remain for each predictor. The algorithm then proceeds as described above in the *Selecting the split variable* step, and selects among the predictors the one that yields the most significant split. For large datasets, and with many continuous predictor variables, this modification of the simpler CHAID algorithm may require significant computing time.

General Computation Issues of CHAID

Reviewing large trees: Unique analysis management tools. A general issue that arises when applying tree classification or regression methods is that the final trees can become very large. In practice, when the input data are complex and, for example, contain many different categories for classification problems, and many possible predictors for performing the classification, then the resulting trees can become very large. This is not so much a computational problem as it is a problem of presenting the trees in a manner that is easily accessible to the data analyst, or for presentation to the "consumers" of the research.

Analyzing ANCOVA-like designs. The classic CHAID algorithms can accommodate both continuous and categorical predictor. However, in practice, it is not uncommon to combine such variables into analysis of variance/covariance (ANCOVA) like predictor designs with main effects or interaction effects for categorical and continuous predictors. This method of analyzing coded ANCOVA-like designs is relatively new. However, it is easy to see how the use of coded predictor designs expands these powerful classification and regression techniques to the analysis of data from experimental.

CHAID, C&RT, and QUEST

For classification-type problems (categorical dependent variable), all three algorithms can be used to build a tree for prediction. QUEST is generally faster than the other two algorithms, however, for very large datasets, the memory requirements are usually larger, so using the QUEST algorithms for classification with very large input data sets may be impractical.

For regression-type problems (continuous dependent variable), the QUEST algorithm is not applicable, so only CHAID and C&RT can be used. CHAID will build non-binary trees that tend to be "wider". This has made the CHAID method particularly popular in market research applications: CHAID often yields many terminal nodes connected to a single branch, which can be conveniently summarized in a simple two-way table with multiple categories for each variable or dimension of the table. This type of display matches well the requirements for research on market segmentation, for example, it may yield a split on a variable *Income*, dividing that variable into 4 categories and groups of individuals belonging to those categories that are different with respect to some important consumer-behavior related variable (e.g., types of cars most likely to be purchased). C&RT will always yield binary trees, which can sometimes not be summarized as efficiently for interpretation and/or presentation.

As far as predictive accuracy is concerned, it is difficult to derive general recommendations, and this issue is still the subject of active research. As a practical matter, it is best to apply different algorithms, perhaps compare them with user-defined interactively derived trees, and decide on the most reasonably and best performing model based on the prediction errors. For a discussion of various schemes for combining predictions from different models, see, for example, Witten and Frank, 2000.

Chapter 8
Classification and Regression Trees (C&RT)

- C&RT Introductory Overview - Basic Ideas
- Computational Details
- Computational Formulas

Introductory Overview - Basic Ideas

Overview

C&RT builds classification and regression trees for predicting continuous dependent variables (regression) and categorical predictor variables (classification). The classic *C&RT* algorithm was popularized by Breiman et al. (Breiman, Friedman, Olshen, & Stone, 1984; see also Ripley, 1996). A general introduction to tree-classifiers, specifically to the QUEST (Quick, Unbiased, Efficient Statistical Trees) algorithm, is also presented in the context of the *Classification Trees Analysis* facilities, and much of the following discussion presents the same information, in only a slightly different context. Another, similar type of tree building algorithm is CHAID (Chi-square Automatic Interaction Detector; see Kass, 1980).

Classification and Regression Problems

There are numerous algorithms for predicting continuous variables or categorical variables from a set of continuous predictors and/or categorical factor effects. For example, in *GLM (General Linear Models)* and *GRM (General Regression Models)*, you can specify a linear combination (design) of continuous predictors and categorical factor effects (e.g., with two-way and three-way interaction effects) to predict a continuous dependent variable. In *GDA (General Discriminant Function Analysis)*, you can specify such designs for predicting categorical variables, i.e., to solve classification problems.

Regression-type problems. Regression-type problems are generally those where one attempts to predict the values of a continuous variable from one or more continuous and/or categorical predictor variables. For example, you may want to predict the selling prices of single family homes (a continuous dependent variable) from various other continuous predictors (e.g., square footage) as well as categorical predictors (e.g., style of home, such as ranch, two-story, etc.; zip code or telephone area code where the property is located, etc.; note that this latter variable would be categorical in nature, even though it would contain numeric values or codes). If you used simple multiple regression, or some general linear model (*GLM*) to predict the selling prices of single family homes, you would determine a linear equation for these variables that can be used to compute predicted selling prices. There are many different analytic procedures for fitting linear models (*GLM*, *GRM*, *Regression*), various types of nonlinear models (e.g., *Generalized Linear/Nonlinear Models (GLZ)*, *Generalized Additive Models (GAM)*, etc.), or completely custom-defined nonlinear models (see *Nonlinear Estimation*), where you can type in an arbitrary equation containing parameters to be estimated. CHAID also analyzes regression-type problems, and produces results that are similar (in nature) to those computed by *C&RT*. Note that various neural network architectures are also applicable to solve regression-type problems.

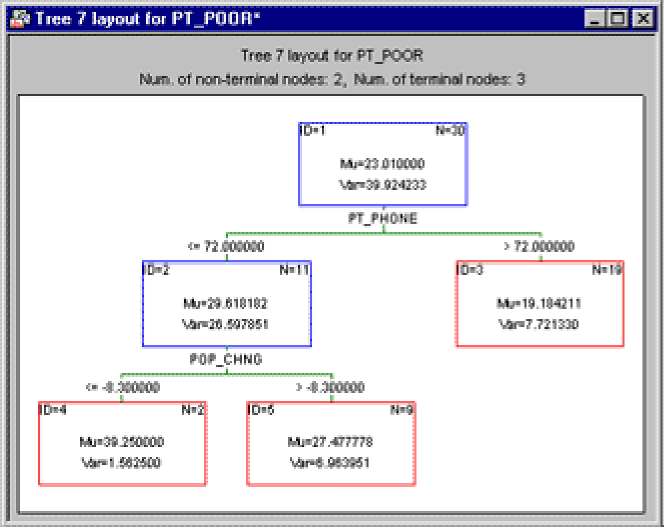
Classification-type problems. Classification-type problems are generally those where one attempts to predict values of a categorical dependent variable (class, group membership, etc.) from one or more continuous and/or categorical predictor variables. For example, you may be interested in predicting who will or will not graduate from college, or who will or will not renew a subscription. These would be examples of simple binary classification problems, where the categorical dependent variable can only assume two distinct and mutually exclusive values. In other cases one might be interested in predicting which one of multiple different alternative consumer products (e.g., makes of cars) a person decides to purchase, or which type of failure occurs with different types of engines. In those cases there are multiple categories or classes for the categorical dependent variable. There are a number of methods for analyzing classification-type problems and to compute predicted classifications, either from simple continuous predictors (e.g., binomial or multinomial logit regression in *GLZ*), from categorical predictors (e.g., *Log-Linear analysis* of multi-way frequency tables), or both (e.g., via ANCOVA-like designs in *GLZ* or *GDA*). The *CHAID* also analyzes classification-type problems, and produces results that are similar (in nature) to those computed by *C&RT*. Note that various neural network architectures are also applicable to solve classification-type problems.

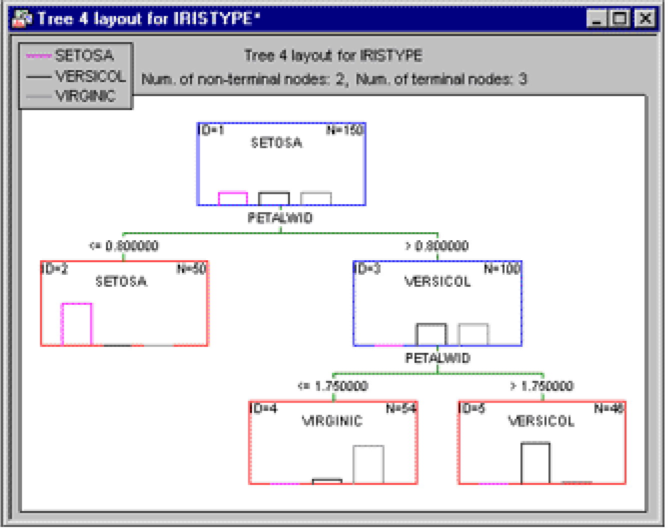
Classification and Regression Trees (*C&RT*)

In most general terms, the purpose of the analyses via tree-building algorithms is to determine a set of *if-then* logical (split) conditions that permit accurate prediction or classification of cases.

Classification Trees

For example, consider the widely referenced Iris data classification problem introduced by Fisher [1936; see also *Discriminant Function Analysis* and *General Discriminant Analysis (GDA)*]. The data file *Irisdat* reports the lengths and widths of sepals and petals of three types of irises (Setosa, Versicol, and Virginic). The purpose of the analysis is to learn how one can discriminate between the three types of flowers, based on the four measures of width and length of petals and sepals. Discriminant function analysis will estimate several linear combinations of predictor variables for computing classification scores (or probabilities) that allow the user to determine the predicted classification for each observation. A classification tree will determine a set of logical if-then conditions (instead of linear equations) for predicting or classifying cases instead:



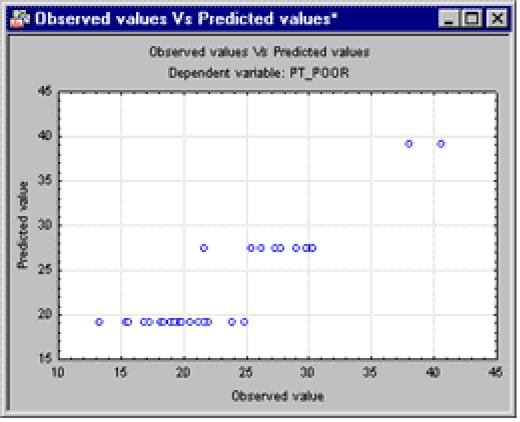


The interpretation of this tree is straightforward: If the petal width is less than or equal to 0.8, the respective flower would be classified as Setosa; if the petal width is greater than 0.8 and less than or equal to 1.75, then the respective flower would be classified as Virginic; else, it belongs to class Versicol.

Regression Trees

The general approach to derive predictions from few simple if-then conditions can be applied to regression problems as well. This example is based on the data file *Poverty*, which contains 1960 and 1970 Census figures for a random selection of 30 counties. The research question (for that example) was to determine the correlates of poverty, that is, the variables that best predict the percent of families below the poverty line in a county. A reanalysis of those data, using the regression tree analysis [and v-fold cross-validation, yields the following results:

Again, the interpretation of these results is rather straightforward: Counties where the percent of households with a phone is greater than 72% have generally a lower poverty rate. The greatest poverty rate is evident in those counties that show less than (or equal to) 72% of households with a phone, and where the population change (from the 1960 census to the 170 census) is less than -8.3 (minus 8.3). These results are straightforward, easily presented, and intuitively clear as well: There are some affluent counties (where most households have a telephone), and those generally have little poverty. Then there are counties that are generally less affluent, and among those the ones that shrunk most showed the greatest poverty rate. A quick review of the scatterplot of observed vs. predicted values shows how the discrimination between the latter two groups is particularly well "explained" by the tree model.



Advantages of *Classification and Regression Trees* (*C&RT*) Methods

As mentioned earlier, there are a large number of methods that an analyst can choose from when analyzing classification or regression problems. Tree classification techniques, when they "work" and produce accurate predictions or predicted classifications based on few logical if-then conditions, have a number of advantages over many of those alternative techniques.

Simplicity of results. In most cases, the interpretation of results summarized in a tree is very simple. This simplicity is useful not only for purposes of rapid classification of new observations (it is much easier to evaluate just one or two logical conditions, than to compute classification scores for each possible group, or predicted values, based on all predictors and using possibly some complex nonlinear model equations), but can also often yield a much simpler "model" for explaining why observations are classified or predicted in a particular manner (e.g., when analyzing business problems, it is much easier to present a few simple if-then statements to management, than some elaborate equations).

Tree methods are nonparametric and nonlinear. The final results of using tree methods for classification or regression can be summarized in a series of (usually few) logical if-then conditions (tree nodes). Therefore, there is no implicit assumption that the underlying relationships between the predictor variables and the dependent variable are linear, follow some specific non-linear link function [e.g., see *Generalized Linear/Nonlinear Models (GLZ)*], or that they are even monotonic in nature. For example, some continuous outcome variable of interest could be positively related to a variable Income if the income is less than some certain amount, but negatively related if it is more than that amount (i.e., the tree could reveal multiple splits based on the same variable Income, revealing such a non-monotonic relationship between the variables). Thus, tree methods are particularly well suited for data mining tasks, where there is often little *a priori* knowledge nor any coherent set of theories or predictions regarding which variables are related and how. In those types of data analyses, tree methods can often reveal simple relationships between just a few variables that could have easily gone unnoticed using other analytic techniques.

General Computation Issues and Unique Solutions of C&RT

The computational details involved in determining the best split conditions to construct a simple yet useful and informative tree are quite complex. Refer to Breiman et al. (1984) for a discussion of their CART® algorithm to learn more about the general theory of and specific computational solutions for constructing classification and regression trees. An excellent general discussion of tree classification and regression methods, and comparisons with other approaches to pattern recognition and neural networks, is provided in Ripley (1996).

Avoiding Over-Fitting: Pruning, Crossvalidation, and V-fold Crossvalidation

A major issue that arises when applying regression or classification trees to "real" data with much random error noise concerns the decision when to stop splitting. For example, if you had a data set with 10 cases, and performed 9 splits (determined 9 if-then conditions), you could perfectly predict every single case. In general, if you only split a sufficient number of times, eventually you will be able to "predict" ("reproduce" would be the more appropriate term here) your original data (from which you determined the splits). Of course, it is far from clear whether such complex results (with many splits) will replicate in a sample of new observations; most likely they will not.

This general issue is also discussed in the literature on tree classification and regression methods, as well as neural networks, under the topic of "overlearning" or "overfitting." If not stopped, the tree algorithm will ultimately "extract" all information from the data, including information that is not and cannot be predicted in the population with the current set of predictors, i.e., random or noise variation. The general approach to addressing this issue is first to stop generating new split nodes when subsequent splits only result in very little overall improvement of the prediction. For example, if you can predict 90% of all cases correctly from 10 splits, and 90.1% of all cases from 11 splits, then it obviously makes little sense to add that 11th split to the tree. There are many such criteria for automatically stopping the splitting (tree-building) process.

Once the tree building algorithm has stopped, it is always useful to further evaluate the quality of the prediction of the current tree in samples of observations that did not participate in the original computations. These methods are used to "prune back" the tree, i.e., to eventually (and ideally) select a simpler tree than the one obtained when the tree building algorithm stopped, but one that is equally as accurate for predicting or classifying "new" observations.

Crossvalidation. One approach is to apply the tree computed from one set of observations (learning sample) to another completely independent set of observations (testing sample). If most or all of the splits determined by the analysis of the learning sample are essentially based on "random noise," then the prediction for the testing sample will be very poor. Hence one can infer that the selected tree is not very good (useful), and not of the "right size."

V-fold crossvalidation. Continuing further along this line of reasoning (described in the context of crossvalidation above), why not repeat the analysis many times over with different randomly drawn samples from the data, for every tree size starting at the root of the tree, and applying it to the prediction of observations from randomly selected testing samples. Then use (interpret, or accept as your final result) the tree that shows the best average accuracy for cross-validated predicted classifications or predicted values. In most cases, this tree will not be the one with the most terminal nodes, i.e., the most complex tree. This method for pruning a tree, and for selecting a smaller tree from a sequence of trees, can be very powerful, and is particularly useful for smaller data sets. It is an essential step for generating useful (for prediction) tree models, and because it can be computationally difficult to do, this method is often not found in tree classification or regression software.

Reviewing Large Trees: Unique Analysis Management Tools

Another general issue that arises when applying tree classification or regression methods is that the final trees can become very large. In practice, when the input data are complex and, for example, contain many different categories for classification problems and many possible predictors for performing the classification, then the resulting trees can become very large. This is not so much a computational problem as it is a problem of presenting the trees in a manner that is easily accessible to the data analyst, or for presentation to the "consumers" of the research.

Analyzing ANCOVA-like Designs

The classic (Breiman et. al., 1984) classification and regression trees algorithms can accommodate both continuous and categorical predictor. However, in practice, it is not uncommon to combine such variables into analysis of variance/covariance (ANCOVA) like predictor designs with main effects or interaction effects for categorical and continuous predictors. This method of analyzing coded ANCOVA-like designs is relatively new and. However, it is easy to see how the use of coded predictor designs expands these powerful classification and regression techniques to the analysis of data from experimental designs (e.g., see for example the detailed discussion of experimental design methods for quality improvement in the context of the Experimental Design module of Industrial Statistics).

Computational Details

The process of computing classification and regression trees can be characterized as involving four basic steps:

- Specifying the criteria for predictive accuracy
- Selecting splits
- Determining when to stop splitting
- Selecting the "right-sized" tree.

These steps are very similar to those discussed in the context of Classification Trees Analysis (see also Breiman et al., 1984, for more details). See also, Computational Formulas.

Specifying the Criteria for Predictive Accuracy

The classification and regression trees (*C&RT*) algorithms are generally aimed at achieving the best possible predictive accuracy. Operationally, the most accurate prediction is defined as the prediction with the minimum costs. The notion of costs was developed as a way to generalize, to a broader range of prediction situations, the idea that the best prediction has the lowest misclassification rate. In most applications, the cost is measured in terms of proportion of misclassified cases, or variance. In this context, it follows, therefore, that a prediction would be considered best if it has the lowest misclassification rate or the smallest variance. The need for minimizing costs, rather than just the proportion of misclassified cases, arises when some predictions that fail are more catastrophic than others, or when some predictions that fail occur more frequently than others.

Priors. In the case of a categorical response (classification problem), minimizing costs amounts to minimizing the proportion of misclassified cases when priors are taken to be proportional to the class sizes and when misclassification costs are taken to be equal for every class.

The *a priori* probabilities used in minimizing costs can greatly affect the classification of cases or objects. Therefore, care has to be taken while using the priors. If differential base rates are not of interest for the study, or if one knows that there are about an equal number of cases in each class, then one would use equal priors. If the differential base rates are reflected in the class sizes (as they would be, if the sample is a probability sample), then one would use priors estimated by the class proportions of the sample. Finally, if you have specific knowledge about the base rates (for example, based on previous research), then one would specify priors in accordance with that knowledge The general point is that the relative size of the priors assigned to each class can be used to "adjust" the importance of misclassifications for each class. However, no priors are required when one is building a regression tree.

Misclassification costs. Sometimes more accurate classification of the response is desired for some classes than others for reasons not related to the relative class sizes. If the criterion for predictive accuracy is Misclassification costs, then minimizing costs would amount to minimizing the proportion of misclassified cases when priors are considered proportional to the class sizes and misclassification costs are taken to be equal for every class.

Case weights. Case weights are treated strictly as case multipliers. For example, the misclassification rates from an analysis of an aggregated data set using case weights will be identical to the misclassification rates from the same analysis where the cases are replicated the specified number of times in the data file.

However, note that the use of case weights for aggregated data sets in classification problems is related to the issue of minimizing costs. Interestingly, as an alternative to using case weights for aggregated data sets, one could specify appropriate priors and/or misclassification costs and produce the same results while avoiding the additional processing required to analyze multiple cases with the same values for all variables. Suppose that in an aggregated data set with two classes having an equal number of cases, there are case weights of 2 for all cases in the first class, and case weights of 3 for all cases in the second class. If you specified priors of .4 and .6, respectively, specified equal misclassification costs, and analyzed the data without case weights, you will get the same misclassification rates as you would get if you specified priors estimated by the class sizes, specified equal misclassification costs, and analyzed the aggregated data set using the case weights. You would also get the same misclassification rates if you specified priors to be equal, specified the costs of misclassifying class 1 cases as class 2 cases to be 2/3 of the costs of misclassifying class 2 cases as class 1 cases, and analyzed the data without case weights.

Selecting Splits

The second basic step in classification and regression trees is to select the splits on the predictor variables that are used to predict membership in classes of the categorical dependent variables, or to predict values of the continuous dependent (response) variable. In general terms, the split at each node will be found that will generate the greatest improvement in predictive accuracy. This is usually measured with some type of node impurity measure, which provides an indication of the relative homogeneity (the inverse of impurity) of cases in the terminal nodes. If all cases in each terminal node show identical values, then node impurity is minimal, homogeneity is maximal, and prediction is perfect (at least for the cases used in the computations; predictive validity for new cases is of course a different matter...).

For classification problems, C&RT gives the user the choice of several impurity measures: The Gini index, Chi-square, or G-square. The Gini index of node impurity is the measure most commonly chosen for classification-type problems. As an impurity measure, it reaches a value of zero when only one class is present at a node. With priors estimated from class sizes and equal misclassification costs, the Gini measure is computed as the sum of products of all pairs of class proportions for classes present at the node; it reaches its maximum value when class sizes at the node are equal; the Gini index is equal to zero if all cases in a node belong to the same class. The Chi-square measure is similar to the standard Chi-square value computed for the expected and observed classifications (with priors adjusted for misclassification cost), and the G-square measure is similar to the maximum-likelihood Chi-square (as for example computed in the Log-Linear module). For regression-type problems, a least-squares deviation criterion (similar to what is computed in least squares regression) is automatically used. Computational Formulas provides further computational details.

Determining When to Stop Splitting

As discussed in Basic Ideas, in principal, splitting could continue until all cases are perfectly classified or predicted. However, this wouldn't make much sense since one would likely end up with a tree structure that is as complex and "tedious" as the original data file (with many nodes possibly containing single observations), and that would most likely not be very useful or accurate for predicting new observations. What is required is some reasonable stopping rule. In *C&RT*, two options are available that can be used to keep a check on the splitting process; namely Minimum n and Fraction of objects.

Minimum n. One way to control splitting is to allow splitting to continue until all terminal nodes are pure or contain no more than a specified minimum number of cases or objects. In *C&RT* this is done by using the option Minimum n that allows you to specify the desired minimum number of cases as a check on the splitting process. This option can be used when *Prune on misclassification error*, *Prune on deviance*, or *Prune on variance* is active as the *Stopping rule* for the analysis.

Fraction of objects. Another way to control splitting is to allow splitting to continue until all terminal nodes are pure or contain no more cases than a specified minimum fraction of the sizes of one or more classes (in the case of classification problems, or all cases in regression problems). This option can be used when FACT-style direct stopping has been selected as the *Stopping rule* for the analysis. In *C&RT*, the desired minimum fraction can be specified as the Fraction of objects. For classification problems, if the priors used in the analysis are equal and class sizes are equal as well, then splitting will stop when all terminal nodes containing more than one class have no more cases than the specified fraction of the class sizes for one or more classes. Alternatively, if the priors used in the analysis are not equal, splitting will stop when all terminal nodes containing more than one class have no more cases than the specified fraction for one or more classes. See Loh and Vanichestakul, 1988 for details.

Pruning and Selecting the "Right-Sized" Tree

The size of a tree in the classification and regression trees analysis is an important issue, since an unreasonably big tree can only make the interpretation of results more difficult. Some generalizations can be offered about what constitutes the "right-sized" tree. It should be sufficiently complex to account for the known facts, but at the same time it should be as simple as possible. It should exploit information that increases predictive accuracy and ignore information that does not. It should, if possible, lead to greater understanding of the phenomena it describes. The options available in *C&RT* allow the use of either, or both, of two different strategies for selecting the "right-sized" tree from among all the possible trees. One strategy is to grow the tree to just the right size, where the right size is determined by the user, based on the knowledge from previous research, diagnostic information from previous analyses, or even intuition. The other strategy is to use a set of well-documented, structured procedures developed by Breiman et al. (1984) for selecting the "right-sized" tree. These procedures are not foolproof, as Breiman et al. (1984) readily acknowledge, but at least they take subjective judgment out of the process of selecting the "right-sized" tree.

FACT-style direct stopping. We will begin by describing the first strategy, in which the user specifies the size to grow the tree. This strategy is followed by selecting FACT-style direct stopping as the stopping rule for the analysis, and by specifying the Fraction of objects which allows the tree to grow to the desired size. *C&RT* provides several options for obtaining diagnostic information to determine the reasonableness of the choice of size for the tree. Specifically, three options are available for performing cross-validation of the selected tree; namely Test sample, V-fold, and Minimal cost-complexity.

Test sample cross-validation. The first, and most preferred type of cross-validation is the test sample cross-validation. In this type of cross-validation, the tree is computed from the learning sample, and its predictive accuracy is tested by applying it to predict the class membership in the test sample. If the costs for the test sample exceed the costs for the learning sample, then this is an indication of poor cross-validation. In that case, a different sized tree might cross-validate better. The test and learning samples can be formed by collecting two independent data sets, or if a large learning sample is available, by reserving a randomly selected proportion of the cases, say a third or a half, for use as the test sample.

In the *C&RT* module, test sample cross-validation is performed by specifying a sample identifier variable which contains codes for identifying the sample (learning or test) to which each case or object belongs.

V-fold cross-validation. The second type of cross-validation available in *C&RT* is V-fold cross-validation. This type of cross-validation is useful when no test sample is available and the learning sample is too small to have the test sample taken from it. The user-specified 'v' value for v-fold cross-validation (its default value is 3) determines the number of random subsamples, as equal in size as possible, that are formed from the learning sample. A tree of the specified size is computed 'v' times, each time leaving out one of the subsamples from the computations, and using that subsample as a test sample for cross-validation, so that each subsample is used (v - 1) times in the learning sample and just once as the test sample. The CV costs (cross-validation cost) computed for each of the 'v' test samples are then averaged to give the v-fold estimate of the CV costs.

Minimal cost-complexity cross-validation pruning. In *C&RT*, minimal cost-complexity cross-validation pruning is performed, if *Prune on misclassification error* has been selected as the *Stopping rule*. On the other hand, if *Prune on deviance* has been selected as the *Stopping rule*, then minimal deviance-complexity cross-validation pruning is performed. The only difference in the two options is the measure of prediction error that is used. *Prune on misclassification error* uses the costs that equals the misclassification rate when priors are estimated and misclassification costs are equal, while *Prune on deviance* uses a measure, based on maximum-likelihood principles, called the deviance (see Ripley, 1996). For details about the algorithms used in *C&RT* to implement Minimal cost-complexity cross-validation pruning, see also the Introductory Overview and Computational Methods sections of Classification Trees Analysis.

The sequence of trees obtained by this algorithm have a number of interesting properties. They are nested, because the successively pruned trees contain all the nodes of the next smaller tree in the sequence. Initially, many nodes are often pruned going from one tree to the next smaller tree in the sequence, but fewer nodes tend to be pruned as the root node is approached. The sequence of largest trees is also optimally pruned, because for every size of tree in the sequence, there is no other tree of the same size with lower costs. Proofs and/or explanations of these properties can be found in Breiman et al. (1984).

Tree selection after pruning. The pruning, as discussed above, often results in a sequence of optimally pruned trees. So the next task is to use an appropriate criterion to select the "right-sized" tree from this set of optimal trees. A natural criterion would be the CV costs (cross-validation costs). While there is nothing wrong with choosing the tree with the minimum CV costs as the "right-sized" tree, oftentimes there will be several trees with CV costs close to the minimum. Following Breiman et al. (1984) one could use the "automatic" tree selection procedure and choose as the "right-sized" tree the smallest-sized (least complex) tree whose CV costs do not differ appreciably from the minimum CV costs. In particular, they proposed a "1 SE rule" for making this selection, i.e., choose as the "right-sized" tree the smallest-sized tree whose CV costs do not exceed the minimum CV costs plus 1 times the standard error of the CV costs for the minimum CV costs tree. In *C&RT*, a multiple other than the 1 (the default) can also be specified for the SE rule. Thus, specifying a value of 0.0 would result in the minimal CV cost tree being selected as the "right-sized" tree. Values greater than 1.0 could lead to trees much smaller than the minimal CV cost tree being selected as the "right-sized" tree. One distinct advantage of the "automatic" tree selection procedure is that it helps to avoid "over fitting" and "under fitting" of the data.

As can be been seen, minimal cost-complexity cross-validation pruning and subsequent "right-sized" tree selection is a truly "automatic" process. The algorithms make all the decisions leading to the selection of the "right-sized" tree, except for, perhaps, specification of a value for the SE rule. V-fold cross-validation allows you to evaluate how well each tree "performs" when repeatedly cross-validated in different samples randomly drawn from the data.

Computational Formulas

In Classification and Regression Trees, estimates of accuracy are computed by different formulas for categorical and continuous dependent variables (classification and regression-type problems). For classification-type problems (categorical dependent variable) accuracy is measured in terms of the true classification rate of the classifier, while in the case of regression (continuous dependent variable) accuracy is measured in terms of mean squared error of the predictor.

In addition to measuring accuracy, the following measures of node impurity are used for classification problems: The Gini measure, generalized Chi-square measure, and generalized G-square measure. The Chi-square measure is similar to the standard Chi-square value computed for the expected and observed classifications (with priors adjusted for misclassification cost), and the G-square measure is similar to the maximum-likelihood Chi-square (as for example computed in the Log-Linear module). The Gini measure is the one most often used for measuring purity in the context of classification problems, and it is described below.

For continuous dependent variables (regression-type problems), the least squared deviation (LSD) measure of impurity is automatically applied.

Estimation of Accuracy in Classification

In classification problems (categorical dependent variable), three estimates of the accuracy are used: resubstitution estimate, test sample estimate, and v-fold cross-validation. These estimates are defined here.

Resubstitution estimate. Resubstitution estimate is the proportion of cases that are misclassified by the classifier constructed from the entire sample. This estimate is computed in the following manner:

$$R(d) = \frac{1}{N}\sum_{i=1}^{N} X(d(x_n) \neq j_n)$$

where *X* is the indicator function;

X = 1, if the statement $X(d(x_n) \neq j_n)$is true

X = 0, if the statement $X(d(x_n) \neq j_n)$is false

and *d* (*x*) is the classifier.

The resubstitution estimate is computed using the same data as used in constructing the classifier *d* .

Test sample estimate. The total number of cases are divided into two subsamples □$_1$ and □$_□$. The test sample estimate is the proportion of cases in the subsample □$_□$ which are misclassified by the classifier constructed from the subsample □$_1$. This estimate is computed in the following way.

Let the learning sample □ of size N be partitioned into subsamples □$_1$ and □$_□$ of sizes N and N$_2$, respectively.

$$R^{ts}(d) = \frac{1}{N_2}\sum_{(x_n, j_n)\in Z_2} X(d(x_n) \neq j_n)$$

where □$_□$ is the sub sample that is not used for constructing the classifier.

v-fold crossvalidation. The total number of cases are divided into *v* sub samples □$_1$, □$_□$, ..., □$_v$ of almost equal sizes. v-fold cross validation estimate is the proportion of cases in the subsample □ that are misclassified by the classifier constructed from the subsample □̣̃ □$_v$. This estimate is computed in the following way.

Let the learning sample □ of size N be partitioned into *v* sub samples □$_1$, □$_□$, ..., □$_v$ of almost sizes N$_1$, N$_□$, ..., N$_v$, respectively.

$$R^{ts}(d^{(v)}) = \frac{1}{N_v} \sum_{(x_n, j_n) \in Z_v} X(d^{(v)}(x_n) \neq j_n)$$

where $d^{(v)}(x)$is computed from the sub sample □̣̃ □$_v$.

Estimation of Accuracy in Regression

In the regression problem (continuous dependent variable) three estimates of the accuracy are used: resubstitution estimate, test sample estimate, and v-fold cross-validation. These estimates are defined here.

Resubstitution estimate. The resubstitution estimate is the estimate of the expected squared error using the predictor of the continuous dependent variable. This estimate is computed in the following way.

$$R(d) = \frac{1}{N} \sum_{i=1}^{N} (y_i - d(x_i))^2$$

where the learning sample □ consists of $(x_i,y_i), i = 1,2,...,N$. The resubstitution estimate is computed using the same data as used in constructing the predictor *d* .

Test sample estimate. The total number of cases are divided into two subsamples □$_1$ and □$_□$. The test sample estimate of the mean squared error is computed in the following way:

Let the learning sample □ of size N be partitioned into subsamples □$_1$ and □$_□$ of sizes N and N$_2$, respectively.

$$R^{ts}(d) = \frac{1}{N_2} \sum_{(x_i, y_i) \in Z_2}^{N} (y_i - d(x_i))^2$$

where □$_□$ is the sub-sample that is not used for constructing the predictor.

v-fold cross-validation. The total number of cases are divided into *v* sub samples □$_1$, □$_□$, ..., □$_v$ of almost equal sizes. The subsample □̣̃ □$_v$ is used to construct the predictor *d*. Then v-fold cross validation estimate is computed from the subsample □$_v$ in the following way:

Let the learning sample □ of size N be partitioned into *v* sub samples □$_1$, □$_□$, ..., □$_v$ of almost sizes N$_1$, N$_□$, ..., N$_v$, respectively.

$$R^{CV}(d) = \frac{1}{N_v}\sum_{v}\sum_{(x_n,y_n)\in Z_v}(y_i - d^{(v)}(x_n))^2$$

where $d^{(v)}(x)$is computed from the sub sample $\tilde{□}$ □$_v$.

Estimation of Node Impurity: Gini Measure

The Gini measure is the measure of impurity of a node and is commonly used when the dependent variable is a categorical variable, defined as:

$$g(t) = \sum_{j\neq i} p(j/t)\,p(i/t)$$

if costs of misclassification are not specified,

$$= \sum_{j\neq i} C(i/j)\,p(j/t)\;p(i/t)$$

if costs of misclassification are specified,

where the sum extends over all *k* categories. *p*(*j* / *t*) is the probability of category *j* at the node *t* and *C*(*i* / *j*) is the probability of misclassifying a category *j* case as category *i*.

Estimation of Node Impurity: Least-Squared Deviation

Least-squared deviation (LSD) is used as the measure of impurity of a node when the response variable is continuous, and is computed as:

$$R(t) = \frac{1}{N_w(t)}\sum_{i\in t} w_i\, f_i\,(y_i - \bar{y}(t))^2$$

where $N_w(t)$ is the weighted number of cases in node t, w_i is the value of the weighting variable for case *i*, f_i is the value of the frequency variable, y_i is the value of the response variable, and $y(t)$ is the weighted mean for node *t*.

Chapter 9
Classification Trees

- Basic Ideas
- Characteristics of Classification Trees
    - Hierarchical Nature of Classification Trees
    - Flexibility of Classification Trees
    - The Power and Pitfalls of Classification Trees
- Computational Methods
    - Specifying the Criteria for Predictive Accuracy
    - Selecting Splits
    - Determining When to Stop Splitting
    - Selecting the "Right-Sized" Tree
- A Brief Comparison of Classification Tree Programs

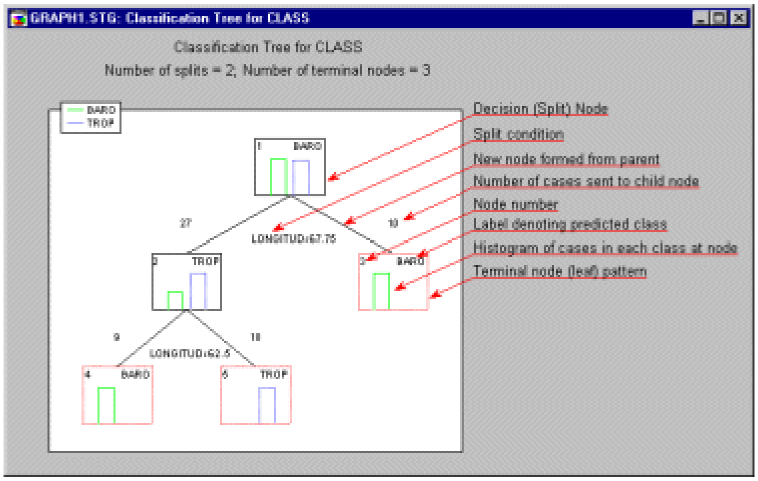
Basic Ideas

Classification trees are used to predict membership of cases or objects in the classes of a categorical dependent variable from their measurements on one or more predictor variables. *Classification tree* analysis is one of the main techniques used in so-called *Data Mining*.

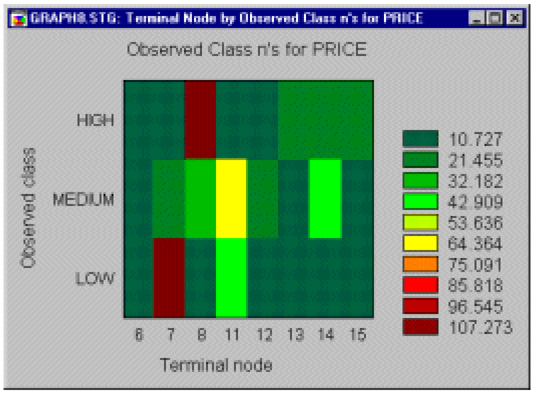
The goal of *classification trees* is to predict or explain responses on a categorical dependent variable, and as such, the available techniques have much in common with the techniques used in the more traditional methods of *Discriminant Analysis*, *Cluster Analysis*, *Nonparametric Statistics*, and *Nonlinear Estimation*. The flexibility of *classification trees* make them a very attractive analysis option, but this is not to say that their use is recommended to the exclusion of more traditional methods. Indeed, when the typically more stringent theoretical and distributional assumptions of more traditional methods are met, the traditional methods may be preferable. But as an exploratory technique, or as a technique of last resort when traditional methods fail, *classification trees* are, in the opinion of many researchers, unsurpassed.

What are *classification trees*? Imagine that you want to devise a system for sorting a collection of coins into different classes (perhaps pennies, nickels, dimes, quarters). Suppose that there is a measurement on which the coins differ, say diameter, which can be used to devise a *hierarchical* system for sorting coins. You might roll the coins on edge down a narrow track in which a slot the diameter of a dime is cut. If the coin falls through the slot it is classified as a dime, otherwise it continues down the track to where a slot the diameter of a penny is cut. If the coin falls through the slot it is classified as a penny, otherwise it continues down the track to where a slot the diameter of a nickel is cut, and so on. You have just constructed a *classification tree*. The decision process used by your *classification tree* provides an efficient method for sorting a pile of coins, and more generally, can be applied to a wide variety of classification problems.

The study and use of *classification trees* are not widespread in the fields of probability and statistical pattern recognition (Ripley, 1996), but *classification trees* are widely used in applied fields as diverse as medicine (diagnosis), computer science (data structures), botany (classification), and psychology (decision theory). *Classification trees* readily lend themselves to being displayed graphically, helping to make them easier to interpret than they would be if only a strict numerical interpretation were possible.


GRAPH1.STG: Classification Tree for CLASS
Classification Tree for CLASS
Number of splits = 2; Number of terminal nodes = 3
BARD
TROP
Decision (Split) Node
Split condition
New node formed from parent
Number of cases sent to child node
Node number
Label denoting predicted class
Histogram of cases in each class at node
Terminal node (leaf) pattern
1
BARD
27
10
LONGITUD:67.75
2
TROP
3
BARD
9
18
LONGITUD:62.5
4
BARD
5
TROP

Classification trees can be and sometimes are quite complex. However, graphical procedures can be developed to help simplify interpretation even for complex trees. If one's interest is mainly in the conditions that produce a particular class of response, perhaps a *High* response, a *3D Contour Plot* can be produced to identify which *terminal node* of the *classification tree* classifies most of the cases with *High* responses.



In the example illustrated by this *3D Contour Plot*, one could "follow the branches" leading to *terminal node 8* to obtain an understanding of the conditions leading to *High* responses.

Amenability to graphical display and ease of interpretation are perhaps partly responsible for the popularity of *classification trees* in applied fields, but two features that characterize *classification trees* more generally are their hierarchical nature and their flexibility.

For information on techniques and issues in computing *classification trees*, see *Computational Methods*. See also Exploratory Data Analysis and Data Mining Techniques.

Characteristics of Classification Trees

Hierarchical Nature of Classification Trees

Breiman et al. (1984) give a number of examples of the use of *classification trees*. As one example, when heart attack patients are admitted to a hospital, dozens of tests are often performed to obtain physiological measures such as heart rate, blood pressure, and so on. A wide variety of other information is also obtained, such as the patient's age and medical history. Patients subsequently can be tracked to see if they survive the heart attack, say, at least 30 days. It would be useful in developing treatments for heart attack patients, and in advancing medical theory on heart failure, if measurements taken soon after hospital admission could be used to identify high-risk patients (those who are not likely to survive at least 30 days). One *classification tree* that Breiman et al. (1984) developed to address this problem was a simple, three question decision tree. Verbally, the binary *classification tree* can be described by the statement, "If the patient's minimum systolic blood pressure over the initial 24 hour period is greater than 91, then if the patient's age is over 62.5 years, then if the patient displays sinus tachycardia, then and only then the patient is predicted not to survive for at least 30 days." It is easy to conjure up the image of a decision "tree" from such a statement. A hierarchy of questions are asked and the final decision that is made depends on the answers to all the previous questions. Similarly, the relationship of a leaf to the tree on which it grows can be described by the hierarchy of splits of branches (starting from the trunk) leading to the last branch from which the leaf hangs. The hierarchical nature of *classification trees* is one of their most basic features (but the analogy with trees in nature should not be taken too far; most decision trees are drawn downward on paper, so the more exact analogy in nature would be a decision root system leading to the root tips, hardly a poetic image).

The hierarchical nature of *classification trees* is illustrated by a comparison to the decision-making procedure employed in *Discriminant Analysis*. A traditional *linear discriminant analysis* of the heart attack data would produce a set of coefficients defining the single linear combination of blood pressure, patient age, and sinus tachycardia measurements that best differentiates low risk from high risk patients. A score for each patient on the linear discriminant function would be computed as a composite of each patient's measurements on the three predictor variables, weighted by the respective discriminant function coefficients. The predicted classification of each patient as a low risk or a high risk patient would be made by *simultaneously* considering the patient's scores on the three predictor variables. That is, suppose *P* (minimum systolic blood Pressure over the 24 hour period), *A* (Age in years), and *T* (presence of sinus Tachycardia: 0 = not present; 1 = present) are the predictor variables, *p, a,* and *t*, are the corresponding linear discriminant function coefficients, and *c* is the "cut point" on the discriminant function for separating the two classes of heart attack patients. The decision equation for each patient would be of the form, "if *pP + aA + tT - c is less than or equal to zero, the patient is low risk, else the patient is in high risk.*"

In comparison, the decision tree developed by Breiman et al. (1984) would have the following *hierarchical* form, where *p, a,* and *t* would be -91, -62.5, and 0, respectively, "If *p + P is less than or equal to zero, the patient is low risk, else if a + A is less than or equal to zero, the patient is low risk, else if t + T is less than or equal to zero, the patient is low risk, else the patient is high risk.*" Superficially, the *Discriminant Analysis* and *classification tree* decision processes might appear similar, because both involve coefficients and decision equations. But the difference of the *simultaneous* decisions of *Discriminant Analysis* from the *hierarchical* decisions of *classification trees* cannot be emphasized enough.

The distinction between the two approaches can perhaps be made most clear by considering how each analysis would be performed in *Regression*. Because risk in the example of Breiman et al. (1984) is a dichotomous dependent variable, the *Discriminant Analysis* predictions could be reproduced by a *simultaneous* multiple regression of risk on the three predictor variables for all patients. The *classification tree* predictions could only be reproduced by three *separate* simple regression analyses, where risk is first regressed on *P* for all patients, then risk is regressed on *A* for patients not classified as low risk in the first regression, and finally, risk is regressed on *T* for patients not classified as low risk in the second regression. This clearly illustrates the *simultaneous* nature of *Discriminant Analysis* decisions as compared to the *recursive, hierarchical* nature of *classification trees* decisions, a characteristic of *classification trees* that has far-reaching implications.

Flexibility of Classification Trees

Another distinctive characteristic of *classification trees* is their flexibility. The ability of *classification trees* to examine the effects of the predictor variables one at a time, rather than just all at once, has already been described, but there are a number of other ways in which *classification trees* are more flexible than traditional analyses. The ability of *classification trees* to perform *univariate splits*, examining the effects of predictors one at a time, has implications for the variety of *types* of predictors that can be analyzed. In the Breiman et al. (1984) heart attack example, blood pressure and age were continuous predictors, but presence of sinus tachycardia was a categorical (two-level) predictor. Even if sinus tachycardia was measured as a three-level categorical predictor (perhaps coded as 0 = not present; 1 = present; 3 = unknown or unsure), without any underlying continuous dimension represented by the values assigned to its levels, univariate splits on the predictor variables could still be easily performed. Additional decisions would be added to the decision tree to exploit any additional information on risk provided by the additional category. To summarize, *classification trees* can be computed for categorical predictors, continuous predictors, or any mix of the two types of predictors when univariate splits are used.

Traditional *linear discriminant analysis* requires that the predictor variables be measured on at least an *interval scale*. For *classification trees* based on univariate splits for *ordinal scale* predictor variables, it is interesting that any *monotonic transformation* of the predictor variables (i.e., any transformation that preserves the *order* of values on the variable) will produce splits yielding the same predicted classes for the cases or objects (if the *C&RT-style univariate split selection method* is used, see Breimen et al., 1984). Therefore, *classification trees* based on univariate splits can be computed without concern for whether a unit change on a continuous predictor represents a unit change on the dimension underlying the values on the predictor variable; it need only be assumed that predictors are measured on at least an *ordinal scale*. In short, assumptions regarding the level of measurement of predictor variables are less stringent.

Classification trees are not limited to univariate splits on the predictor variables. When continuous predictors are indeed measured on at least an *interval scale*, *linear combination splits,* similar to the splits for *linear discriminant analysis*, can be computed for *classification trees*. However, the linear combination splits computed for *Classification Trees* do differ in important ways from the linear combination splits computed for *Discriminant Analysis*. In linear discriminant analysis the number of *linear discriminant functions* that can be extracted is the lesser of the number of predictor variables or the number of classes on the dependent variable minus one. The *recursive* approach implemented for *Classification Trees*module does not face this limitation. For example, dozens of *recursive, linear combination splits* potentially could be performed when there are dozens of predictor variables but only two classes on the dependent variable. This compares with the single linear combination split that could be performed using traditional, *non-recursive* Iinear discriminant analysis, which could leave a substantial amount of the information in the predictor variables unused.

Now consider the situation in which there are many categories but few predictors. Suppose you were trying to sort coins into classes (perhaps pennies, nickels, dimes, and quarters) based only on thickness and diameter measurements. Using traditional linear discriminant analysis, at most two linear discriminant functions could be extracted, and the coins could be successfully sorted only if there were no more than two dimensions represented by linear combinations of thickness and diameter on which the coins differ. Again, the approach implemented for *Classification Trees* does not face a limitation on the number of linear combination splits that can be formed.

The approach implemented for *Classification Trees* for linear combination splits can also be used as the analysis method for constructing *classification trees* using univariate splits. Actually, a univariate split is just a special case of a linear combination split. Imagine a linear combination split in which the coefficients for creating the weighted composite were zero for all predictor variables except one. Since scores on the weighted composite would depend only on the scores on the one predictor variable with the nonzero coefficient, the resulting split would be a univariate split.

The approach implemented for *Classification Trees* for the *Discriminant-based univariate split selection method for categorical and ordered predictors* and for the *Discriminant-based linear combination split selection method for ordered predictors* is an adaption of the algorithms used in QUEST (Quick, Unbiased, Efficient Statistical Trees). QUEST is a *classification tree* program developed by Loh and Shih (1997) that employs a modification of recursive quadratic discriminant analysis and includes a number of innovative features for improving the reliability and efficiency of the *classification trees* that it computes.

The algorithms used in QUEST are fairly technical, but the *Classification Trees* module also offers a *Split selection method* option based on a conceptually simpler approach. The *C&RT-style univariate split selection method* is an adaption of the algorithms used in C&RT, as described by Breiman et al. (1984). C&RT (Classification And Regression Trees) is a *classification tree* program that uses an exhaustive grid search of all possible univariate splits to find the splits for a *classification tree*.

The QUEST and C&RT analysis options compliment each other nicely. C&RT searches can be lengthy when there are a large number of predictor variables with many levels, and it is biased toward choosing predictor variables with more levels for splits, but because it employs an exhaustive search, it is guaranteed to find the splits producing the best classification (in the *learning sample*, but not necessarily in *cross-validation samples*).

QUEST is fast and unbiased. The speed advantage of QUEST over C&RT is particularly dramatic when the predictor variables have dozens of levels (Loh & Shih, 1997, report an analysis completed by QUEST in 1 CPU second that took C&RT 30.5 CPU hours to complete). QUEST's lack of bias in variable selection for splits is also a distinct advantage when some predictor variable have few levels and other predictor variables have many levels (predictors with many levels are more likely to produce "fluke theories," which fit the data well but have low predictive accuracy, see Doyle, 1973, and Quinlan & Cameron-Jones, 1995). Finally, QUEST does not sacrifice predictive accuracy for speed (Lim, Loh, & Shih, 1997). Together, the QUEST and C&RT options allow one to fully exploit the flexibility of *classification trees*.

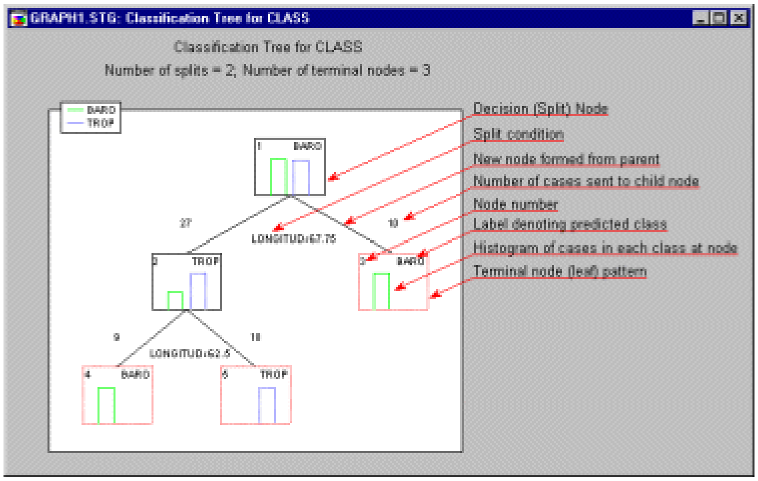
The Power and Pitfalls of Classification Trees

The advantages of *classification trees* over traditional methods such as *linear discriminant analysis*, at least in some applications, can be illustrated using a simple, fictitious data set. To keep the presentation even-handed, other situations in which *linear discriminant analysis* would outperform *classification trees* are illustrated using a second data set.

Suppose you have records of the *Longitude* and *Latitude* coordinates at which 37 storms reached hurricane strength for two classifications of hurricanes--*Baro* hurricanes and *Trop* hurricanes. The fictitious data shown below were presented for illustrative purposes by Elsner, Lehmiller, and Kimberlain (1996), who investigated the differences between baroclinic and tropical North Atlantic hurricanes.

DATA: Barotrop.sta 3v		
LONGITUD	**LATITUDE**	**CLASS**
59.00	17.00	BARO
59.50	21.00	BARO
60.00	12.00	BARO
60.50	16.00	BARO
61.00	13.00	BARO
61.00	15.00	BARO
61.50	17.00	BARO
61.50	19.00	BARO
62.00	14.00	BARO
63.00	15.00	TROP
63.50	19.00	TROP
64.00	12.00	TROP
64.50	16.00	TROP
65.00	12.00	TROP
65.00	15.00	TROP
65.00	17.00	TROP
65.50	16.00	TROP
65.50	19.00	TROP
65.50	21.00	TROP
66.00	13.00	TROP
66.00	14.00	TROP
66.00	17.00	TROP
66.50	17.00	TROP
66.50	18.00	TROP
66.50	21.00	TROP
67.00	14.00	TROP
67.50	18.00	TROP
68.00	14.00	BARO
68.50	18.00	BARO
69.00	13.00	BARO
69.00	15.00	BARO
69.50	17.00	BARO
69.50	19.00	BARO
70.00	12.00	BARO
70.50	16.00	BARO
71.00	17.00	BARO
71.50	21.00	BARO

A linear discriminant analysis of hurricane *Class (Baro* or *Trop)* using *Longitude* and *Latitude* as predictors correctly classifies only 20 of the 37 hurricanes (54%). A *classification tree* for *Class* using the *C&RT-style exhaustive search for univariate splits* option correctly classifies all 37 hurricanes. The *Tree graph* for the *classification tree* is shown below.


GRAPH1.STG: Classification Tree for CLASS
Classification Tree for CLASS
Number of splits = 2; Number of terminal nodes = 3
BARO
TROP
1
BARO
27
LONGITUD<67.75
10
2
TROP
3
BARO
9
LONGITUD<62.5
18
4
BARO
5
TROP
Decision (Split) Node
Split condition
New node formed from parent
Number of cases sent to child node
Node number
Label denoting predicted class
Histogram of cases in each class at node
Terminal node (leaf) pattern

The headings of the graph give the summary information that the *classification tree* has 2 splits and 3 *terminal nodes*. Terminal nodes, or terminal leaves as they are sometimes called, are points on the tree beyond which no further decisions are made. In the graph itself, terminal nodes are outlined with dotted red lines, while the remaining *decision nodes* or *split nodes* are outlined with solid black lines. The tree starts with the top decision node, sometimes called the *root node*. In the graph it is labeled as node 1 in its top-left corner. Initially, all 37 hurricanes are assigned to the root node and tentatively classified as *Baro* hurricanes, as indicated by the *Baro* label in the top-right corner of the root node. *Baro* is chosen as the initial classification because there are slightly more *Baro* than *Trop* hurricanes, as indicated by the *histogram* plotted within the *root node*. The *legend* identifying which bars in the *node histograms* correspond to *Baro* and *Trop* hurricanes is located in the top-left corner of the graph.

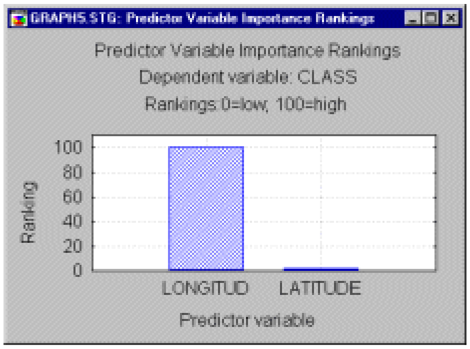
The root node is split, forming two new nodes. The text below the root node describes the split. It indicates that hurricanes with *Longitude* coordinate values of less than or equal to 67.75 are sent to node number 2 and tentatively classified as *Trop* hurricanes, and that hurricanes with *Longitude* coordinate values of greater than 67.75 are assigned to node number 3 and classified as *Baro* hurricanes. The values of 27 and 10 printed above nodes 2 and 3, respectively, indicate the number of cases sent to each of these two *child nodes* from their *parent*, the root node. Similarly, node 2 is subsequently split. The split is such that the 9 hurricanes with *Longitude* coordinate values of less than or equal to 62.5 are sent to node number 4 and classified as *Baro* hurricanes, and the remaining 18 hurricanes with *Longitude* coordinate values of greater than 62.5 are sent to node number 5 and classified as *Trop* hurricanes.

The *Tree graph* presents all this information in a simple, straightforward way, and probably allows one to digest the information in much less time than it takes to read the two preceding paragraphs. Getting to the bottom line, the histograms plotted within the tree's terminal nodes show that the *classification tree* classifies the hurricanes perfectly. Each of the terminal nodes is "pure," containing no misclassified hurricanes. All the information in the *Tree graph* is also available in the *Tree structure* Scrollsheet shown below.

Tree Structure (barotrop.sta)							
CLASSIF. TREES	**Child nodes, observed class n's, predicted class, and split condition for each node**						
Node	**Left branch**	**Right branch**	**n in cls BARO**	**n in cls TROP**	**Predict. class**	**Split constant**	**Split variable**
1	2	3	19	18	BARO	-67.75	LONGITUD
2	4	5	9	18	TROP	-62.50	LONGITUD
3			10	0	BARO		
4			9	0	BARO		
5			0	18	TROP		

Note that in the Scrollsheet nodes 3 through 5 are identified as terminal nodes because no split is performed at those nodes. Also note the signs of the *Split constants* displayed in the Scrollsheet, for example, *-67.75* for the split at node 1. In the *Tree graph*, the *split condition* at node 1 is described as *LONGITUD 67.75* rather than as (the equivalent) *-67.75 + LONGITUD 0.* This is done simply to save space on the graph.

When univariate splits are performed, the predictor variables can be ranked on a 0 - 100 scale in terms of their potential importance in accounting for responses on the dependent variable. For this example, *Longitude* is clearly very important and *Latitude* is relatively unimportant.

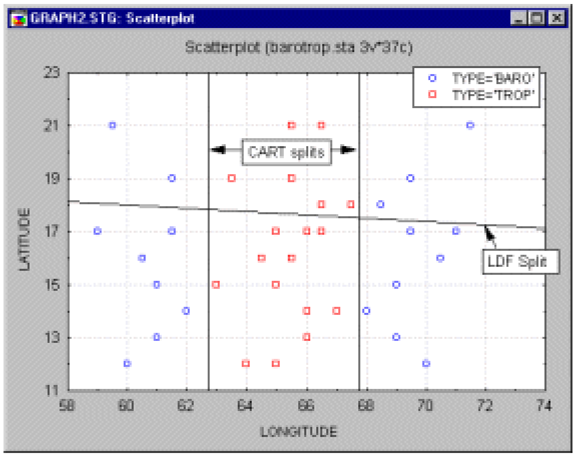


A *classification tree* *Class* using the *Discriminant-based univariate split selection method* option produces similar results. The *Tree structure* Scrollsheet shown for this analysis shows that the splits of *-63.4716* and *-67.7516* are quite similar to the splits found using the *C&RT-style exhaustive search for univariate splits* option, although 1 *Trop* hurricane in terminal node 2 is misclassified as *Baro*.

Tree Structure (barotrop.sta)	
CLASSIF.	**Child nodes, observed class n's,**

TREES	predicted class, and split condition for each node						
Node	Left branch	Right branch	n in cls BARO	n in cls TROP	Predict. class	Split constant	Split variable
1	2	3	19	18	BARO	-63.4716	LONGITUD
2			9	1	BARO		
3	4	5	10	17	TROP	-67.7516	LONGITUD
4			0	17	TROP		
5			10	0	BARO		

A categorized scatterplot for *Longitude* and *Latitude* clearly shows why linear discriminant analysis fails so miserably at predicting *Class*, and why the *classification tree* succeeds so well.



The plot clearly shows that there is no strong linear relationship of longitude or latitude coordinates with *Class*, or of any possible linear combination of longitude and latitude with *Class*. *Class* is not functionally related to longitude or latitude, at least in the linear sense. The LDF (Linear Discriminant Function) Split shown on the graph is almost a "shot in the dark" at trying to separate predicted *Trop* hurricanes (above the split line) from predicted *Baro* hurricanes (below the split line). The C&RT univariate splits, because they are not restricted to a single linear combination of longitude and latitude scores, find the "cut points" on the *Longitude* dimension that allow the best possible (in this case, perfect) classification of hurricane *Class*.

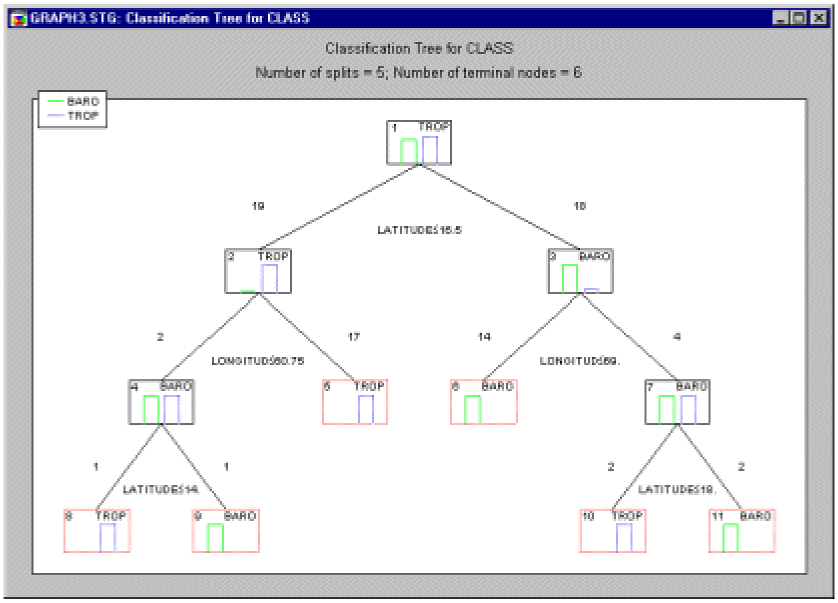
Now we can examine a situation illustrating the pitfalls of *classification tree*. Suppose that the following hurricane data were available.

DATA: Barotro2.sta 3v		
LONGITUD	LATITUDE	CLASS
59.00	17.00	BARO
59.50	21.00	BARO

60.00	12.00	TROP
60.50	16.00	BARO
61.00	13.00	TROP
61.00	15.00	TROP
61.50	17.00	BARO
61.50	19.00	BARO
62.00	14.00	TROP
63.00	15.00	TROP
63.50	19.00	BARO
64.00	12.00	TROP
64.50	16.00	TROP
65.00	12.00	TROP
65.00	15.00	TROP
65.00	17.00	BARO
65.50	16.00	TROP
65.50	19.00	BARO
65.50	21.00	BARO
66.00	13.00	TROP
66.00	14.00	TROP
66.00	17.00	BARO
66.50	17.00	BARO
66.50	18.00	BARO
66.50	21.00	BARO
67.00	14.00	TROP
67.50	18.00	BARO
68.00	14.00	TROP
68.50	18.00	BARO
69.00	13.00	TROP
69.00	15.00	TROP
69.50	17.00	TROP
69.50	19.00	BARO
70.00	12.00	TROP
70.50	16.00	TROP
71.00	17.00	TROP
71.50	21.00	BARO

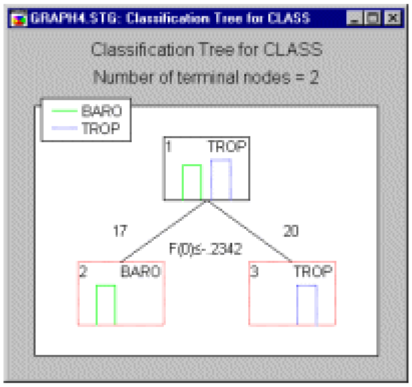
A linear discriminant analysis of hurricane *Class* (*Baro* or *Trop*) using *Longitude* and *Latitude* as predictors correctly classifies all 37 of the hurricanes. A classification tree analysis for *Class* using the *C&RT-style exhaustive search for univariate splits* option also correctly classifies all 37 hurricanes, but the tree requires 5 splits producing 6 terminal nodes. Which results are easier to interpret? In the linear discriminant analysis, the raw canonical discriminant function coefficients for *Longitude* and *Latitude* on the (single) discriminant function are *.122073* and -*.633124*, respectively, and hurricanes with higher longitude and lower latitude coordinates are classified as *Trop*. The interpretation would be that hurricanes in the western Atlantic at low latitudes are likely to be *Trop* hurricanes, and that hurricanes further east in the Atlantic at higher latitudes are likely to be *Baro* hurricanes.

The *Tree graph* for the classification tree analysis using the *C&RT-style exhaustive search for univariate splits* option is shown below.



One could methodically describe the splits in this *classification tree*, exactly as was done in the previous example, but because there are so many splits, the interpretation would necessarily be more complex than the simple interpretation provided by the single discriminant function from the linear discrimination analysis.

However, recall that in describing the flexibility of *Classification Trees* , it was noted that an option exists for *Discriminant-based linear combination splits for ordered predictors* using algorithms from QUEST. The *Tree graph* for the *classification tree* analysis using linear combination splits is shown below.


GRAPH4.STG: Classification Tree for CLASS
Classification Tree for CLASS
Number of terminal nodes = 2
BARO
TROP
1 TROP
17
20
F(0)≤-.2342
2 BARO
3 TROP

Note that in this tree, just one split yields perfect prediction. Each of the terminal nodes is "pure," containing no misclassified hurricanes. The linear combination split used to split the root node into its *left child node* and *right child node* is summarized by the description "*F(0) -.2342.*" This indicates that if a hurricane has a score of less than or equal to -*.2342* on the *split function*--abbreviated as *F(0)*--then it is sent to the left child node and classified as *Baro*, otherwise it is sent to the right child node and classified as *Trop*. The split function coefficients (*.011741* for *Longitude* and -*.060896* for *Latitude*) have the same signs and are similar in their relative magnitude to the corresponding linear discriminant function coefficients from the linear discriminant analysis, so the two analyses are functionally identical, at least in terms of their predictions of hurricane *Class*.

The moral of this story of the power and pitfalls of *classification trees* is that *classification trees* are only as good as the choice of analysis option used to produce them. For finding models that predict well, there is no substitute for a thorough understanding of the nature of the relationships between the predictor and dependent variables.

We have seen that *classification trees* analysis can be characterized as a hierarchical, highly flexible set of techniques for predicting membership of cases or objects in the classes of a categorical dependent variable from their measurements on one or more predictor variables. With this groundwork behind us, we now are ready to look at the methods for computing *classification trees* in greater detail.

For information on the basic purpose of *classification trees*, see *Basic Ideas*. See also, Exploratory Data Analysis and Data Mining Techniques.

Computational Methods

The process of computing *classification trees* can be characterized as involving four basic steps:

1. Specifying the criteria for predictive accuracy,
2. Selecting splits,
3. Determining when to stop splitting, and
4. Choosing the "right-sized" tree.

Specifying the Criteria for Predictive Accuracy

The goal of classification tree analysis, simply stated, is to obtain the most accurate prediction possible. Unfortunately, an operational definition of accurate prediction is hard to come by. To solve the problem of defining predictive accuracy, the problem is "stood on its head," and the most accurate prediction is operationally defined as the prediction with the minimum *costs*. The term *costs* need not seem mystifying. In many typical applications, *costs* simply correspond to the proportion of misclassified cases. The notion of *costs* was developed as a way to generalize, to a broader range of prediction situations, the idea that the best prediction has the lowest misclassification rate.

The need for minimizing costs, rather than just the proportion of misclassified cases, arises when some predictions that fail are more catastrophic than others, or when some predictions that fail occur more frequently than others. The costs to a gambler of losing a single bet (or prediction) on which the gambler's whole fortune is at stake are greater than the costs of losing many bets (or predictions) on which a tiny part of the gambler's fortune is at stake. Conversely, the costs of losing many small bets can be larger than the costs of losing just a few bigger bets. One should spend proportionately more effort in minimizing losses on bets where losing (making errors in prediction) costs you more.

Priors. Minimizing costs, however, does correspond to minimizing the proportion of misclassified cases when *Priors* are taken to be proportional to the class sizes and when *Misclassification costs* are taken to be equal for every class. We will address *Priors* first. *Priors*, or, *a priori* probabilities, specify how likely it is, without using any prior knowledge of the values for the predictor variables in the model, that a case or object will fall into one of the classes. For example, in an educational study of high school drop-outs, it may happen that, overall, there are fewer drop-outs than students who stay in school (i.e., there are different *base rates*); thus, the *a priori* probability that a student drops out is lower than that a student remains in school.

The *a priori* probabilities used in minimizing costs can greatly affect the classification of cases or objects. If differential base rates are not of interest for the study, or if one knows that there are about an equal number of cases in each class, then one would use *equal priors.* If the differential base rates are reflected in the class sizes (as they would be, if the sample is a probability sample) then one would use *priors estimated by the class proportions of the sample*. Finally, if you have specific knowledge about the base rates (for example, based on previous research), then one would specify *priors* in accordance with that knowledge. For example, *a priori* probabilities for carriers of a recessive gene could be specified as twice as high as for individuals who display a disorder caused by the recessive gene. The general point is that the relative size of the *priors* assigned to each class can be used to "adjust" the importance of misclassifications for each class. Minimizing costs corresponds to minimizing the overall proportion of misclassified cases when *Priors* are taken to be proportional to the class sizes (and *Misclassification costs* are taken to be equal for every class), because prediction should be better in larger classes to produce an overall lower misclassification rate.

Misclassification costs. Sometimes more accurate classification is desired for some classes than others for reasons unrelated to relative class sizes. Regardless of their relative frequency, carriers of a disease who are contagious to others might need to be more accurately predicted than carriers of the disease who are not contagious to others. If one assumes that little is lost in avoiding a non-contagious person but much is lost in not avoiding a contagious person, higher *misclassification costs* could be specified for misclassifying a contagious carrier as non-contagious than for misclassifying a non-contagious person as contagious. But to reiterate, minimizing costs corresponds to minimizing the proportion of misclassified cases when *Priors* are taken to be proportional to the class sizes and when *Misclassification costs* are taken to be equal for every class.

Case weights. A little less conceptually, the use of *case weights* on a *weighting variable* as *case multipliers* for *aggregated data sets* is also related to the issue of minimizing costs. Interestingly, as an alternative to using case weights for aggregated data sets, one could specify appropriate *priors* and/or *misclassification costs* and produce the same results while avoiding the additional processing required to analyze multiple cases with the same values for all variables. Suppose that in an aggregated data set with two classes having an equal number of cases, there are case weights of 2 for all the cases in the first class, and case weights of 3 for all the cases in the second class. If you specify *priors* of .4 and .6, respectively, specify equal *misclassification costs*, and analyze the data without case weights, you will get the same misclassification rates as you would get if you specify *priors* estimated by the class sizes, specify equal *misclassification costs*, and analyze the aggregated data set using the case weights. You would also get the same misclassification rates if you specify *priors* to be equal, specify the costs of misclassifying class 1 cases as class 2 cases to be 2/3 of the costs of misclassifying class 2 cases as class 1 cases, and analyze the data without case weights.

The relationships between *priors*, *misclassification costs*, and *case weights* become quite complex in all but the simplest situations (for discussions, see Breiman et al, 1984; Ripley, 1996). In analyses where minimizing *costs* corresponds to minimizing the misclassification rate, however, these issues need not cause any concern. *Priors*, *misclassification costs*, and *case weights* are brought up here, however, to illustrate the wide variety of prediction situations that can be handled using the concept of minimizing *costs*, as compared to the rather limited (but probably typical) prediction situations that can be handled using the narrower (but simpler) idea of minimizing misclassification rates. Furthermore, minimizing *costs* is an underlying goal of classification tree analysis, and is explicitly addressed in the fourth and final basic step in classification tree analysis, where in trying to select the "right-sized" tree, one chooses the tree with the minimum *estimated costs*. Depending on the type of prediction problem you are trying to solve, understanding the idea of reduction of *estimated costs* may be important for understanding the results of the analysis.

Selecting Splits

The second basic step in classification tree analysis is to select the splits on the predictor variables which are used to predict membership in the classes of the dependent variables for the cases or objects in the analysis. Not surprisingly, given the hierarchical nature of *classification trees*, these splits are selected one at time, starting with the split at the root node, and continuing with splits of resulting child nodes until splitting stops, and the child nodes which have not been split become terminal nodes. Three *Split selection methods* are discussed here.

Discriminant-based univariate splits. The first step in split selection when the *Discriminant-based univariate splits* option is chosen is to determine the best terminal node to split in the current tree, and which predictor variable to use to perform the split. For each terminal node, *p*-levels are computed for tests of the significance of the relationship of class membership with the levels of each predictor variable. For categorical predictors, the *p*-levels are computed for *Chi-square* tests of independence of the classes and the levels of the categorical predictor that are present at the node. For ordered predictors, the *p*-levels are computed for *ANOVAs* of the relationship of the classes to the values of the ordered predictor that are present at the node. If the smallest computed *p*-level is smaller than the default Bonferoni-adjusted *p*-level for multiple comparisons of .05 (a different threshold value can be used), the predictor variable producing that smallest *p*-level is chosen to split the corresponding node. If no *p*-level smaller than the threshold *p*-level is found, *p*-levels are computed for statistical tests that are robust to distributional violations, such as *Levene's F*. Details concerning node and predictor variable selection when no *p*-level is smaller than the specified threshold are described in Loh and Shih (1997).

The next step is to determine the split. For ordered predictors, the 2-means clustering algorithm of Hartigan and Wong (1979, see also *Cluster Analysis*) is applied to create two "superclasses" for the node. The two roots are found for a quadratic equation describing the difference in the means of the "superclasses" on the ordered predictor, and the values for a split corresponding to each root are computed. The split closest to a "superclass" mean is selected. For categorical predictors, dummy-coded variables representing the levels of the categorical predictor are constructed, and then singular value decomposition methods are applied to transform the dummy-coded variables into a set of non-redundant ordered predictors. The procedures for ordered predictors are then applied and the obtained split is "mapped back" onto the original levels of the categorical variable and represented as a contrast between two sets of levels of the categorical variable. Again, further details about these procedures are described in Loh and Shih (1997). Although complicated, these procedures reduce a bias in split selection that occurs when using the *C&RT-style exhaustive search method* for selecting splits. This is the bias toward selecting variables with more levels for splits, a bias which can skew the interpretation of the relative importance of the predictors in explaining responses on the dependent variable (Breiman et. al., 1984).

Discriminant-based linear combination splits. The second split selection method is the *Discriminant-based linear combination split* option for ordered predictor variables (however, the predictors are assumed to be measured on at least *interval scales*). Surprisingly, this method works by treating the continuous predictors from which linear combinations are formed in a manner which is similar to the way categorical predictors are treated in the previous method. Singular value decomposition methods are used to transform the continuous predictors into a new set of non-redundant predictors. The procedures for creating "superclasses" and finding the split closest to a "superclass" mean are then applied, and the results are "mapped back" onto the original continuous predictors and represented as a univariate split on a linear combination of predictor variables.

C&RT-style exhaustive search for univariate splits. The third split-selection method is the *C&RT-style exhaustive search for univariate splits* method for categorical or ordered predictor variables. With this method, all possible splits for each predictor variable at each node are examined to find the split producing the largest improvement in *goodness* *of fit* (or equivalently, the largest reduction in lack of fit). What determines the domain of possible splits at a node? For categorical predictor variables with *k* levels present at a node, there are $2^{(k-1)}$ - 1 possible contrasts between two sets of levels of the predictor. For ordered predictors with *k* distinct levels present at a node, there are *k* -1 midpoints between distinct levels. Thus it can be seen that the number of possible splits that must be examined can become very large when there are large numbers of predictors with many levels which must be examined at many nodes.

How is improvement in *goodness of fit* determined? Three choices of *Goodness of fit* measures are discussed here. The *Gini measure of node impurity* is a measure which reaches a value of zero when only one class is present at a node (with *priors estimated from class sizes* and *equal misclassification costs*, the *Gini measure* is computed as the sum of products of all pairs of class proportions for classes present at the node; it reaches its maximum value when class sizes at the node are equal). The *Gini measure* was the measure of *goodness of fit* preferred by the developers of C&RT (Breiman et. al., 1984). The two other indices are the *Chi-square* measure, which is similar to Bartlett's Chi-square (Bartlett, 1948), and the *G-square* measure, which is similar to the maximum-likelihood Chi-square used in *structural equation modeling*. The *C&RT-style exhaustive search for univariate splits* method works by searching for the split that maximizes the reduction in the value of the selected *goodness of fit* measure. When the fit is perfect, classification is perfect.

Determining When to Stop Splitting

The third step in classification tree analysis is to determine when to stop splitting. One characteristic of *classification trees* is that if no limit is placed on the number of splits that are performed, eventually "pure" classification will be achieved, with each terminal node containing only one class of cases or objects. However, "pure" classification is usually unrealistic. Even a simple *classification tree* such as a coin sorter can produce impure classifications for coins whose sizes are distorted or if wear changes the lengths of the slots cut in the track. This potentially could be remedied by further sorting of the coins that fall into each slot, but to be practical, at some point the sorting would have to stop and you would have to accept that the coins have been reasonably well sorted.

Likewise, if the observed classifications on the dependent variable or the levels on the predicted variable in a classification tree analysis are measured with error or contain "noise," it is unrealistic to continue to sort until every terminal node is "pure." Two options for controlling when splitting stops will be discussed here. These two options are linked to the choice of the *Stopping rule* specified for the analysis.

Minimum n. One option for controlling when splitting stops is to allow splitting to continue until all terminal nodes are pure or contain no more than a specified minimum number of cases or objects. The desired minimum number of cases can be specified as the *Minimum n*, and splitting will stop when all terminal nodes containing more than one class have no more than the specified number of cases or objects.

Fraction of objects. Another option for controlling when splitting stops is to allow splitting to continue until all terminal nodes are pure or contain no more cases than a specified minimum fraction of the sizes of one or more classes. The desired minimum fraction can be specified as the *Fraction of objects* and, if the *priors* used in the analysis are equal and class sizes are equal, splitting will stop when all terminal nodes containing more than one class have no more cases than the specified fraction of the class sizes for one or more classes. If the *priors* used in the analysis are not equal, splitting will stop when all terminal nodes containing more than one class have no more cases than the specified fraction for one or more classes.

Selecting the "Right-Sized" Tree

After a night at the horse track, a studious gambler computes a huge *classification tree* with numerous splits that perfectly account for the win, place, show, and no show results for every horse in every race. Expecting to become rich, the gambler takes a copy of the *Tree graph* to the races the next night, sorts the horses racing that night using the *classification tree*, makes his or her predictions and places his or her bets, and leaves the race track later much less rich than had been expected. The poor gambler has foolishly assumed that a *classification tree* computed from a *learning sample* in which the outcomes are *already known* will perform equally well in *predicting* outcomes in a second, independent *test sample.* The gambler's *classification tree* performed poorly during *cross-validation*. The gambler's payoff might have been larger using a smaller *classification tree* that did not *classify* perfectly in the *learning sample*, but which was expected to *predict* equally well in the *test sample*.

Some generalizations can be offered about what constitutes the "right-sized" *classification tree*. It should be sufficiently complex to account for the known facts, but at the same time it should be as simple as possible. It should exploit information that increases predictive accuracy and ignore information that does not. It should, if possible, lead to greater understanding of the phenomena which it describes. Of course, these same characteristics apply to any scientific theory, so we must try to be more specific about what constitutes the "right-sized" *classification tree*. One strategy is to grow the tree to just the right size, where the right size is determined by the user from knowledge from previous research, diagnostic information from previous analyses, or even intuition. The other strategy is to use a set of well-documented, structured procedures developed by Breiman et al. (1984) for selecting the "right-sized" tree. These procedures are not foolproof, as Breiman et al. (1984) readily acknowledge, but at least they take subjective judgment out of the process of selecting the "right-sized" tree.

FACT-style direct stopping. We will begin by describing the first strategy, in which the researcher specifies the size to grow the *classification tree*. This strategy is followed by using *FACT-style direct stopping* as the *Stopping rule* for the analysis, and by specifying the *Fraction of objects* which allows the tree to grow to the desired size. There are several options for obtaining diagnostic information to determine the reasonableness of the choice of size for the tree. Three options for performing *cross-validation* of the selected *classification tree* are discussed below.

Test sample cross-validation. The first, and most preferred type of cross-validation is *test sample cross-validation.* In this type of cross-validation, the *classification tree* is computed from the learning sample, and its predictive accuracy is tested by applying it to predict class membership in the test sample. If the *costs* for the test sample exceed the *costs* for the learning sample (remember, *costs* equal the proportion of misclassified cases when *priors* are *estimated* and *misclassification costs* are *equal*), this indicates poor cross-validation and that a different sized tree might cross-validate better. The test and learning samples can be formed by collecting two independent data sets, or if a large learning sample is available, by reserving a randomly selected proportion of the cases, say a third or a half, for use as the test sample.

V-fold cross-validation. This type of cross-validation is useful when no test sample is available and the learning sample is too small to have the test sample taken from it. A specified *V* value for *V-fold cross-validation* determines the number of random subsamples, as equal in size as possible, that are formed from the learning sample. The *classification tree* of the specified size is computed *V* times, each time leaving out one of the subsamples from the computations, and using that subsample as a test sample for cross-validation, so that each subsample is used *V* - 1 times in the learning sample and just once as the test sample. The *CV costs* computed for each of the *V* test samples are then averaged to give the *V-fold estimate of the CV costs*.

Global cross-validation. In *global cross-validation*, the entire analysis is replicated a specified number of times holding out a fraction of the learning sample equal to 1 over the specified number of times, and using each hold-out sample in turn as a test sample to cross-validate the selected *classification tree*. This type of cross-validation is probably no more useful than *V-fold cross-validation* when *FACT-style direct stopping* is used, but can be quite useful as a method validation procedure when automatic tree selection techniques are used (for discussion, see Breiman et. al., 1984). This brings us to the second of the two strategies that can used to select the "right-sized" tree, an automatic tree selection method based on a technique developed by Breiman et al. (1984) called *minimal cost-complexity cross-validation pruning.*

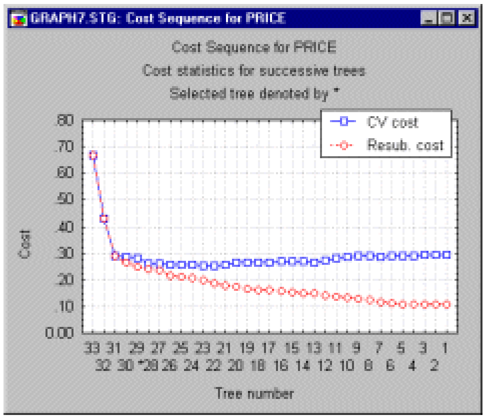
Minimal cost-complexity cross-validation pruning. Two methods of pruning can be used depending on the Stopping Rule you choose to use. *Minimal cost-complexity cross-validation pruning* is performed when you decide to *Prune on misclassification error* (as a *Stopping rule*), and *minimal deviance-complexity cross-validation pruning* is performed when you choose to *Prune on deviance* (as a *Stopping rule*). The only difference in the two options is the measure of *prediction error* that is used. *Prune on misclassification error* uses the *costs* that we have discussed repeatedly (which equal the misclassification rate when *priors* are *estimated* and *misclassification costs* are *equal). Prune on deviance* uses a measure, based on maximum-likelihood principles, called the *deviance* (see Ripley, 1996). We will focus on *cost-complexity cross-validation* pruning (as originated by Breiman et. al., 1984), since *deviance-complexity pruning* merely involves a different measure of *prediction error.*

The costs needed to perform *cost-complexity pruning* are computed as the tree is being grown, starting with the split at the root node up to its maximum size, as determined by the specified *Minimum n*. The learning sample *costs* are computed as each split is added to the tree, so that a sequence of generally decreasing *costs* (reflecting better classification) are obtained corresponding to the number of splits in the tree. The learning sample *costs* are called *resubstitution costs* to distinguish them from *CV costs,* because *V-fold cross-validation* is also performed as each split is added to the tree. Use the *estimated CV costs* from *V-fold cross-validation* as the *costs* for the root node. Note that tree size can be taken to be the number of terminal nodes, because for binary trees the tree size starts at one (the root node) and increases by one with each added split. Now, define a parameter called the complexity parameter whose initial value is zero, and for every tree (including the first, containing only the root node), compute the value for a function defined as the *costs* for the tree plus the complexity parameter times the tree size. Increase the complexity parameter continuously until the value of the function for the largest tree exceeds the value of the function for a smaller-sized tree. Take the smaller-sized tree to be the new largest tree, continue increasing the complexity parameter continuously until the value of the function for the largest tree exceeds the value of the function for a smaller-sized tree, and continue the process until the root node is the largest tree. (Those who are familiar with numerical analysis will recognize the use of a *penalty function* in this algorithm. The function is a linear combination of *costs*, which generally decrease with tree size, and tree size, which increases linearly. As the complexity parameter is increased, larger trees are penalized for their *complexity* more and more, until a discrete threshold is reached at which a smaller-sized tree's higher *costs* are outweighed by the largest tree's higher complexity)

The sequence of largest trees obtained by this algorithm have a number of interesting properties. They are nested, because successively pruned trees contain all the nodes of the next smaller tree in the sequence. Initially, many nodes are often pruned going from one tree to the next smaller tree in the sequence, but fewer nodes tend to be pruned as the root node is approached. The sequence of largest trees is also optimally pruned, because for every size of tree in the sequence, there is no other tree of the same size with lower *costs*. Proofs and/or explanations of these properties can be found in Breiman et al. (1984).

Tree selection after pruning. We now select the "right-sized" tree from the sequence of optimally pruned trees. A natural criterion is the *CV costs*. While there is nothing wrong with choosing the tree with the minimum *CV costs* as the "right-sized" tree, oftentimes there will be several trees with *CV costs* close to the minimum. Breiman et al. (1984) make the reasonable suggestion that one should choose as the "right-sized" tree the smallest-sized (least complex) tree whose *CV costs* do not differ appreciably from the minimum *CV costs*. They proposed a "1 SE rule" for making this selection, i.e., choose as the "right-sized" tree the smallest-sized tree whose *CV costs* do not exceed the minimum *CV costs* plus 1 times the *Standard error of the CV costs* for the minimum *CV costs* tree.

One distinct advantage of the "automatic" tree selection procedure is that it helps to avoid "overfitting" and "underfitting" of the data. The graph below shows a typical plot of the *Resubstitution costs* and *CV costs* for the sequence of successively pruned trees.



As shown in this graph, the *Resubstitution costs* (e.g., the misclassification rate in the learning sample) rather consistently decrease as tree size increases. The *CV costs*, on the other hand, approach the minimum quickly as tree size initially increases, but actually start to rise as tree size becomes very large. Note that the selected "right-sized" tree is close to the inflection point in the curve, that is, close to the point where the initial sharp drop in *CV costs* with increased tree size starts to level out. The "automatic" tree selection procedure is designed to select the simplest (smallest) tree with close to minimum *CV costs*, and thereby avoid the loss in predictive accuracy produced by "underfitting" or "overfitting" the data (note the similarity to the logic underlying the use of a "scree plot" to determine the number of factors to retain in *Factor Analysis*; see also Reviewing the Results of a Principal Components Analysis).

As has been seen, *minimal cost-complexity cross-validation pruning* and subsequent "right-sized" tree selection is a truly "automatic" process. The algorithms make all the decisions leading to selection of the "right-sized" tree, except for, perhaps, specification of a value for the *SE rule*. One issue that arises with the use of such "automatic" procedures is how well the results replicate, where replication might involve the selection of trees of quite different sizes across replications, given the "automatic" selection process that is used. This is where *global cross-validation* can be very useful. As explained previously, in *global cross-validation*, the entire analysis is replicated a specified number of times (3 is the default) holding out a fraction of the cases to use as a test sample to cross-validate the selected *classification tree*. If the average of the *costs* for the test samples, called the *global CV costs*, exceeds the *CV costs* for the selected tree, or if the *standard error of the global CV costs* exceeds the *standard error of the CV costs* for the selected tree, this indicates that the "automatic" tree selection procedure is allowing too much variability in tree selection rather than consistently selecting a tree with minimum estimated *costs*.

Classification trees and traditional methods. As can be seen in the methods used in computing *classification trees*, in a number of respects *classification trees* are decidedly different from traditional statistical methods for predicting class membership on a categorical dependent variable. They employ a hierarchy of predictions, with many predictions sometimes being applied to particular cases, to sort the cases into predicted classes. Traditional methods use simultaneous techniques to make one and only one class membership prediction for each and every case. In other respects, such as having as its goal accurate prediction, classification tree analysis is indistinguishable from traditional methods. Time will tell if classification tree analysis has enough to commend itself to become as accepted as the traditional methods.

For information on the basic purpose of *classification trees*, see *Basic Ideas*. For information on the *hierarchical nature* and *flexibility* of *classification trees*, see *Characteristics of Classification Trees*. See also, Exploratory Data Analysis and Data Mining Techniques.

A Brief Comparison of Classification Tree Programs

A variety of classification tree programs have been developed to predict membership of cases or objects in the classes of a categorical dependent variable from their measurements on one or more predictor variables. In the previous section, Computational Methods, we have discussed the QUEST (Loh & Shih, 1997) and C&RT (Breiman et. al., 1984) programs for computing binary classification trees based on univariate splits for categorical predictor variables, ordered predictor variables (measured on at least an ordinal scale), or a mix of both types of predictors. We have also discussed computing classification trees based on linear combination splits for interval scale predictor variables.

Some classification trees programs, such as FACT (Loh & Vanichestakul, 1988) and THAID (Morgan & Messenger, 1973, as well as the related programs AID, for Automatic Interaction Detection, Morgan & Sonquist, 1963, and CHAID, for Chi-Square Automatic Interaction Detection, Kass, 1980) perform multi-level splits rather than binary splits when computing classification trees. A multi-level split performs k - 1 splits (where k is the number of levels of the splittting variable), as compared to a binary split which performs one split (regardless of the number of levels of the splittting variable). However, it should be noted that there is no inherent advantage of multi-level splits, because any multi-level split can be represented as a series of binary splits, and there may be disadvantages of using multi-level splits. With multi-level splits, predictor variables can be used for splitting only once, so the resulting classification trees may be unrealistically short and uninteresting (Loh & Shih, 1997). A more serious problem is bias in variable selection for splits. This bias is possible in any program such as THAID (Morgan & Sonquist, 1963) that employs an exhaustive search for finding splits (for a discussion, see Loh & Shih, 1997). Bias in variable selection is the bias toward selecting variables with more levels for splits, a bias which can skew the interpretation of the relative importance of the predictors in explaining responses on the dependent variable (Breiman et. al., 1984).

Bias in variable selection can be avoided by using the Discriminant-based (univariate or linear combination) split options. These options make use of the algorithms in QUEST

(Loh & Shih, 1997) to prevent bias in variable selection. The C&RT-style exhaustive search for univariate splits option is useful if one's goal is to find splits producing the best possible classification in the learning sample (but not necessarily in independent cross-validiation samples). For reliable splits, as well as computational speed, the Discriminant-based split options are recommended. For information on techniques and issues in computing classification trees, see the Computational Methods section.

Building trees interactively. In contrast, another method for building trees that has proven popular in applied research and data exploration is based on experts' knowledge about the domain or area under investigation, and relies on interactive choices (for how to grow the tree) by such experts to arrive at "good" (valid) models for prediction or predictive classification. In other words, instead of building trees automatically, using sophisticated algorithms for choosing good predictors and splits (for growing the branches of the tree), a user may want to determine manually which variables to include in the tree, and how to split those variables to create the branches of the tree. This enables the user to experiment with different variables and scenarios, and ideally to derive a better understanding of the phenomenon under investigation by combining her or his expertise with the analytic capabilities and options for building the. In practice, it may often be most useful to combine the automatic methods for building trees with "educated guesses" and domain-specific expertise. You may want to grow some portions of the tree using automatic methods and refine and modify the tree based on your expertise. Another common situation where this type of combined automatic and interactive tree building is called for is when some variables that are chosen automatically for some splits are not easily observable because they cannot be measured reliably or economically (i.e., obtaining such measurements would be too expensive). For example, suppose the automatic analysis at some point selects a variable Income as a good predictor for the next split; however, you may not be able to obtain reliable data on income from the new sample to which you want to apply the results of the current analysis (e.g., for predicting some behavior of interest, such as whether or not the person will purchase something from your catalog). In this case, you may want to select a "surrogate" variable, i.e., a variable that you can observe easily and that is likely related or similar to variable *Income* (with respect to its predictive power; for example, a variable *Number of years of education* may be related to *Income* and have similar predictive power; while most people are reluctant to reveal their level of income, they are more likely to report their level of education, and hence, this latter variable is more easily measured).

Chapter 10
Cluster Analysis

- General Purpose
- Statistical Significance Testing
- Area of Application
- Joining (Tree Clustering)
  - Hierarchical Tree
  - Distance Measures
  - Amalgamation or Linkage Rules
- Two-way Joining
  - Introductory Overview
  - Two-way Joining
- *k*-Means Clustering
  - Example
  - Computations
  - Interpretation of results
- EM (Expectation Maximization) Clustering
  - Introductory Overview
  - The EM Algorithm
- Finding the Right Number of Clusters in *k*-Means and EM Clustering: v-Fold Cross-Validation

General Purpose

The term *cluster analysis* (first used by Tryon, 1939) encompasses a number of different algorithms and methods for grouping objects of similar kind into respective categories. A general question facing researchers in many areas of inquiry is how to *organize* observed data into meaningful structures, that is, to develop taxonomies. In other words cluster analysis is an exploratory data analysis tool which aims at sorting different objects into groups in a way that the degree of association between two objects is maximal if they belong to the same group and minimal otherwise. Given the above, cluster analysis can be used to discover structures in data without providing an explanation/interpretation. In other words, cluster analysis simply discovers structures in data without explaining why they exist.

We deal with clustering in almost every aspect of daily life. For example, a group of diners sharing the same table in a restaurant may be regarded as a cluster of people. In food stores items of similar nature, such as different types of meat or vegetables are displayed in the same or nearby locations. There is a countless number of examples in which clustering playes an important role. For instance, biologists have to organize the different species of animals before a meaningful description of the differences between animals is possible. According to the modern system employed in biology, man belongs to the primates, the mammals, the amniotes, the vertebrates, and the animals. Note how in this classification, the higher the level of aggregation the less similar are the members in the respective class. Man has more in common with all other primates (e.g., apes) than it does with the more "distant" members of the mammals (e.g., dogs), etc. For a review of the general categories of cluster analysis methods, see *Joining (Tree Clustering)*, *Two-way Joining (Block Clustering)*, and *k-Means Clustering*. In short, whatever the nature of your business is, sooner or later you will run into a clustering problem of one form or another.

Statistical Significance Testing

Note that the above discussions refer to clustering algorithms and do not mention anything about statistical significance testing. In fact, cluster analysis is not as much a typical statistical test as it is a "collection" of different algorithms that "put objects into clusters according to well defined similarity rules." The point here is that, unlike many other statistical procedures, cluster analysis methods are mostly used when we do not have any a priori hypotheses, but are still in the exploratory phase of our research. In a sense, cluster analysis finds the "most significant solution possible." Therefore, statistical significance testing is really not appropriate here, even in cases when p-levels are reported (as in *k*-means clustering).

Area of Application

Clustering techniques have been applied to a wide variety of research problems. Hartigan (1975) provides an excellent summary of the many published studies reporting the results of cluster analyses. For example, in the field of medicine, clustering diseases, cures for diseases, or symptoms of diseases can lead to very useful taxonomies. In the field of psychiatry, the correct diagnosis of clusters of symptoms such as paranoia, schizophrenia, etc. is essential for successful therapy. In archeology, researchers have attempted to establish taxonomies of stone tools, funeral objects, etc. by applying cluster analytic techniques. In general, whenever one needs to classify a "mountain" of information into manageable meaningful piles, cluster analysis is of great utility.

Joining (Tree Clustering)

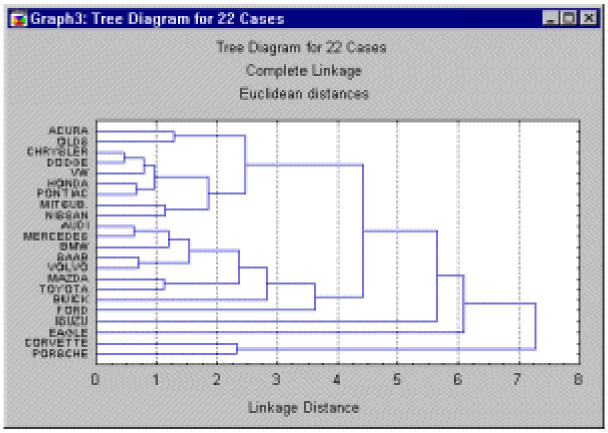
- Hierarchical Tree
- Distance Measures
- Amalgamation or Linkage Rules

General Logic

The example in the General Purpose Introduction illustrates the goal of the joining or tree clustering algorithm. The purpose of this algorithm is to join together objects (e.g., animals) into successively larger clusters, using some measure of similarity or distance. A typical result of this type of clustering is the hierarchical tree.

Hierarchical Tree

Consider a *Horizontal Hierarchical Tree Plot* (see graph below), on the left of the plot, we begin with each object in a class by itself. Now imagine that, in very small steps, we "relax" our criterion as to what is and is not unique. Put another way, we lower our threshold regarding the decision when to declare two or more objects to be members of the same cluster.



As a result we *link* more and more objects together and aggregate (*amalgamate*) larger and larger clusters of increasingly dissimilar elements. Finally, in the last step, all objects are joined together. In these plots, the horizontal axis denotes the linkage distance (in *Vertical Icicle Plots*, the vertical axis denotes the linkage distance). Thus, for each node in the graph (where a new cluster is formed) we can read off the criterion distance at which the respective elements were linked together into a new single cluster. When the data contain a clear "structure" in terms of clusters of objects that are similar to each other, then this structure will often be reflected in the hierarchical tree as distinct branches. As the result of a successful analysis with the joining method, one is able to detect clusters (branches) and interpret those branches.

Distance Measures

The joining or tree clustering method uses the dissimilarities (similarities) or distances between objects when forming the clusters. Similarities are a set of rules that serve as criteria for grouping or separating items. In the previous example the rule for grouping a number of dinners was whether they shared the same table or not. These distances (similarities) can be based on a single dimension or multiple dimensions, with each dimension representing a rule or condition for grouping objects. For example, if we were to cluster fast foods, we could take into account the number of calories they contain, their price, subjective ratings of taste, etc. The most straightforward way of computing distances between objects in a multi-dimensional space is to compute Euclidean distances. If we had a two- or three-dimensional space this measure is the actual geometric distance between objects in the space (i.e., as if measured with a ruler). However, the joining algorithm does not "care" whether the distances that are "fed" to it are actual real distances, or some other derived measure of distance that is more meaningful to the researcher; and it is up to the researcher to select the right method for his/her specific application.

Euclidean distance. This is probably the most commonly chosen type of distance. It simply is the geometric distance in the multidimensional space. It is computed as:

$$\text{distance}(x,y) = \{\Sigma_i (x_i - y_i)^2 \}^{½}$$

Note that Euclidean (and squared Euclidean) distances are usually computed from raw data, and not from standardized data. This method has certain advantages (e.g., the distance between any two objects is not affected by the addition of new objects to the analysis, which may be outliers). However, the distances can be greatly affected by differences in scale among the dimensions from which the distances are computed. For example, if one of the dimensions denotes a measured length in centimeters, and you then convert it to millimeters (by multiplying the values by 10), the resulting Euclidean or squared Euclidean distances (computed from multiple dimensions) can be greatly affected (i.e., biased by those dimensions which have a larger scale), and consequently, the results of cluster analyses may be very different. Generally, it is good practice to transform the dimensions so they have similar scales.

Squared Euclidean distance. You may want to square the standard Euclidean distance in order to place progressively greater weight on objects that are further apart. This distance is computed as (see also the note in the previous paragraph):

$$\text{distance}(x,y) = \Sigma_i (x_i - y_i)^2$$

City-block (Manhattan) distance. This distance is simply the average difference across dimensions. In most cases, this distance measure yields results similar to the simple Euclidean distance. However, note that in this measure, the effect of single large differences (outliers) is dampened (since they are not squared). The city-block distance is computed as:

$$\text{distance}(x,y) = \Sigma_i |x_i - y_i|$$

Chebychev distance. This distance measure may be appropriate in cases when one wants to define two objects as "different" if they are different on any one of the dimensions. The Chebychev distance is computed as:

$\text{distance}(x,y) = \text{Maximum}|x_i - y_i|$

Power distance. Sometimes one may want to increase or decrease the progressive weight that is placed on dimensions on which the respective objects are very different. This can be accomplished via the *power distance*. The power distance is computed as:

$\text{distance}(x,y) = (\Sigma_i |x_i - y_i|^p)^{1/r}$

where *r* and *p* are user-defined parameters. A few example calculations may demonstrate how this measure "behaves." Parameter *p* controls the progressive weight that is placed on differences on individual dimensions, parameter *r* controls the progressive weight that is placed on larger differences between objects. If *r* and *p* are equal to 2, then this distance is equal to the Euclidean distance.

Percent disagreement. This measure is particularly useful if the data for the dimensions included in the analysis are categorical in nature. This distance is computed as:

$\text{distance}(x,y) = (\text{Number of } x_i \neq y_i)/ i$

Amalgamation or Linkage Rules

At the first step, when each object represents its own cluster, the distances between those objects are defined by the chosen distance measure. However, once several objects have been linked together, how do we determine the distances between those new clusters? In other words, we need a linkage or amalgamation rule to determine when two clusters are sufficiently similar to be linked together. There are various possibilities: for example, we could link two clusters together when *any* two objects in the two clusters are closer together than the respective linkage distance. Put another way, we use the "nearest neighbors" across clusters to determine the distances between clusters; this method is called *single linkage*. This rule produces "stringy" types of clusters, that is, clusters "chained together" by only single objects that happen to be close together. Alternatively, we may use the neighbors across clusters that are furthest away from each other; this method is called *complete linkage*. There are numerous other linkage rules such as these that have been proposed.

Single linkage (nearest neighbor). As described above, in this method the distance between two clusters is determined by the distance of the two closest objects (nearest neighbors) in the different clusters. This rule will, in a sense, *string* objects together to form clusters, and the resulting clusters tend to represent long "chains."

Complete linkage (furthest neighbor). In this method, the distances between clusters are determined by the greatest distance between any two objects in the different clusters (i.e., by the "furthest neighbors"). This method usually performs quite well in cases when the objects actually form naturally distinct "clumps." If the clusters tend to be somehow elongated or of a "chain" type nature, then this method is inappropriate.

Unweighted pair-group average. In this method, the distance between two clusters is calculated as the average distance between all pairs of objects in the two different clusters. This method is also very efficient when the objects form natural distinct "clumps," however, it performs equally well with elongated, "chain" type clusters. Note that in their book, Sneath and Sokal (1973) introduced the abbreviation UPGMA to refer to this method as *unweighted pair-group method using arithmetic averages*.

Weighted pair-group average. This method is identical to the *unweighted pair-group average* method, except that in the computations, the size of the respective clusters (i.e., the number of objects contained in them) is used as a weight. Thus, this method (rather than the previous method) should be used when the cluster sizes are suspected to be greatly uneven. Note that in their book, Sneath and Sokal (1973) introduced the abbreviation *WPGMA* to refer to this method as *weighted pair-group method using arithmetic averages*.

Unweighted pair-group centroid. The *centroid* of a cluster is the average point in the multidimensional space defined by the dimensions. In a sense, it is the *center of gravity* for the respective cluster. In this method, the distance between two clusters is determined as the difference between centroids. Sneath and Sokal (1973) use the abbreviation *UPGMC* to refer to this method as *unweighted pair-group method using the centroid average*.

Weighted pair-group centroid (median). This method is identical to the previous one, except that weighting is introduced into the computations to take into consideration differences in cluster sizes (i.e., the number of objects contained in them). Thus, when there are (or one suspects there to be) considerable differences in cluster sizes, this method is preferable to the previous one. Sneath and Sokal (1973) use the abbreviation *WPGMC* to refer to this method as *weighted pair-group method using the centroid average*.

Ward's method. This method is distinct from all other methods because it uses an analysis of variance approach to evaluate the distances between clusters. In short, this method attempts to minimize the Sum of Squares (SS) of any two (hypothetical) clusters that can be formed at each step. Refer to Ward (1963) for details concerning this method. In general, this method is regarded as very efficient, however, it tends to create clusters of small size.

For an overview of the other two methods of clustering, see Two-way Joining and *k*-Means Clustering.

Two-way Joining

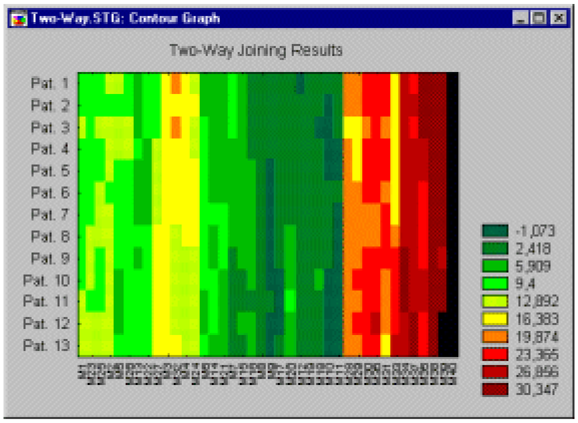
- Introductory Overview
- Two-way Joining

Introductory Overview

Previously, we have discussed this method in terms of "objects" that are to be clustered (see Joining (Tree Clustering)). In all other types of analyses the research question of interest is usually expressed in terms of cases (observations) or variables. It turns out that the clustering of both may yield useful results. For example, imagine a study where a medical researcher has gathered data on different measures of physical fitness (variables) for a sample of heart patients (cases). The researcher may want to cluster cases (patients) to detect clusters of patients with similar syndromes. At the same time, the researcher may want to cluster variables (fitness measures) to detect clusters of measures that appear to tap similar physical abilities.

Two-way Joining

Given the discussion in the paragraph above concerning whether to cluster cases or variables, one may wonder why not cluster both simultaneously? Two-way joining is useful in (the relatively rare) circumstances when one expects that both cases and variables will simultaneously contribute to the uncovering of meaningful patterns of clusters.



For example, returning to the example above, the medical researcher may want to identify clusters of patients that are similar with regard to particular clusters of similar measures of physical fitness. The difficulty with interpreting these results may arise from the fact that the similarities between different clusters may pertain to (or be caused by) somewhat different subsets of variables. Thus, the resulting structure (clusters) is by nature not homogeneous. This may seem a bit confusing at first, and, indeed, compared to the other clustering methods described (see Joining (Tree Clustering) and *k*-Means Clustering), two-way joining is probably the one least commonly used. However, some researchers believe that this method offers a powerful exploratory data analysis tool (for more information you may want to refer to the detailed description of this method in Hartigan, 1975).

---

k-Means Clustering

- Example
- Computations
- Interpretation of results

General logic

This method of clustering is very different from the Joining (Tree Clustering) and Two-way Joining. Suppose that you already have hypotheses concerning the number of clusters in your cases or variables. You may want to "tell" the computer to form exactly 3 clusters that are to be as distinct as possible. This is the type of research question that can be addressed by the k- means clustering algorithm. In general, the *k*-means method will produce exactly *k* different clusters of greatest possible distinction. It should be mentioned that the best number of clusters *k* leading to the greatest separation (distance) is not known as *a priori* and must be computed from the data (see Finding the Right Number of Clusters).

Example

In the physical fitness example (see *Two-way Joining*), the medical researcher may have a "hunch" from clinical experience that her heart patients fall basically into three different categories with regard to physical fitness. She might wonder whether this intuition can be quantified, that is, whether a *k*-means cluster analysis of the physical fitness measures would indeed produce the three clusters of patients as expected. If so, the means on the different measures of physical fitness for each cluster would represent a quantitative way of expressing the researcher's hypothesis or intuition (i.e., patients in cluster 1 are high on measure 1, low on measure 2, etc.).

Computations

Computationally, you may think of this method as analysis of variance (ANOVA) "in reverse." The program will start with *k* random clusters, and then move objects between those clusters with the goal to 1) minimize variability within clusters and 2) maximize variability between clusters. In other words, the similarity rules will apply maximally to the members of one cluster and minimally to members belonging to the rest of the clusters. This is analogous to "ANOVA in reverse" in the sense that the significance test in ANOVA evaluates the between group variability against the within-group variability when computing the significance test for the hypothesis that the means in the groups are different from each other. In *k*-means clustering, the program tries to move objects (e.g., cases) in and out of groups (clusters) to get the most significant ANOVA results.

Interpretation of results

Usually, as the result of a *k*-means clustering analysis, we would examine the means for each cluster on each dimension to assess how distinct our *k* clusters are. Ideally, we would obtain very different means for most, if not all dimensions, used in the analysis. The magnitude of the *F* values from the analysis of variance performed on each dimension is another indication of how well the respective dimension discriminates between clusters.

EM (Expectation Maximization) Clustering

- Introductory Overview
- The EM Algorithm

Introductory Overview

The methods described here are similar to the *k*-Means algorithm described above, and you may want to review that section for a general overview of these techniques and their applications. The general purpose of these techniques is to detect clusters in observations (or variables) and to assign those observations to the clusters. A typical example application for this type of analysis is a marketing research study in which a number of consumer behavior related variables are measured for a large sample of respondents. The purpose of the study is to detect "market segments," i.e., groups of respondents that are somehow more similar to each other (to all other members of the same cluster) when compared to respondents that "belong to" other clusters. In addition to identifying such clusters, it is usually equally of interest to determine how the clusters are different, i.e., determine the specific variables or dimensions that vary and how they vary in regard to members in different clusters.

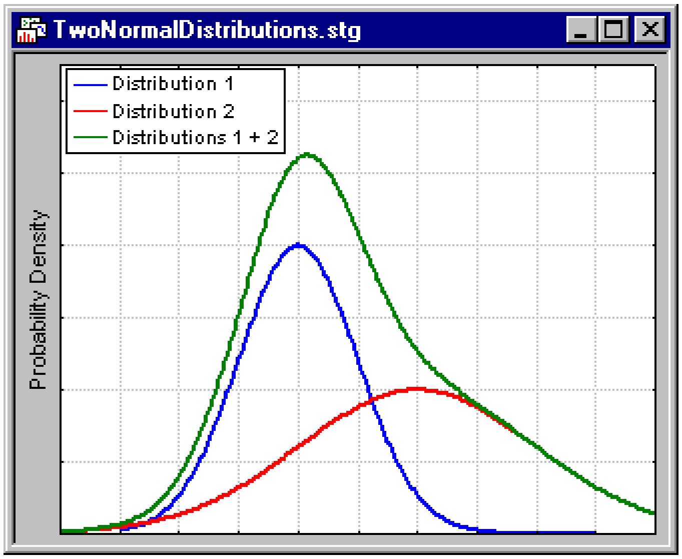
***k*-means clustering.** To reiterate, the classic *k*-Means algorithm was popularized and refined by Hartigan (1975; see also Hartigan and Wong, 1978). The basic operation of that algorithm is relatively simple: Given a fixed number of (desired or hypothesized) *k* clusters, assign observations to those clusters so that the means across clusters (for all variables) are as different from each other as possible.

Extensions and generalizations. The *EM* (expectation maximization) algorithm extends this basic approach to clustering in two important ways:

1. Instead of assigning cases or observations to clusters to maximize the differences in means for continuous variables, the *EM* clustering algorithm computes probabilities of cluster memberships based on one or more probability distributions. The goal of the clustering algorithm then is to maximize the overall probability or likelihood of the data, given the (final) clusters.
2. Unlike the classic implementation of *k*-means clustering, the general *EM* algorithm can be applied to both continuous and categorical variables (note that the classic *k*-means algorithm can also be modified to accommodate categorical variables).

The EM Algorithm

The *EM* algorithm for clustering is described in detail in Witten and Frank (2001). The basic approach and logic of this clustering method is as follows. Suppose you measure a single continuous variable in a large sample of observations. Further, suppose that the sample consists of two clusters of observations with different means (and perhaps different standard deviations); within each sample, the distribution of values for the continuous variable follows the normal distribution. The resulting distribution of values (in the population) may look like this:



Mixtures of distributions. The illustration shows two normal distributions with different means and different standard deviations, and the sum of the two distributions. Only the mixture (sum) of the two normal distributions (with different means and standard deviations) would be observed. The goal of *EM* clustering is to estimate the means and standard deviations for each cluster so as to maximize the likelihood of the observed data (distribution). Put another way, the *EM* algorithm attempts to approximate the observed distributions of values based on mixtures of different distributions in different clusters.

With the implementation of the *EM* algorithm in some computer programs, you may be able to select (for continuous variables) different distributions such as the normal, log-normal, and Poisson distributions. You can select different distributions for different variables and, thus, derive clusters for mixtures of different types of distributions.

Categorical variables. The *EM* algorithm can also accommodate categorical variables. The method will at first randomly assign different probabilities (weights, to be precise) to each class or category, for each cluster. In successive iterations, these probabilities are refined (adjusted) to maximize the likelihood of the data given the specified number of clusters.

Classification probabilities instead of classifications. The results of *EM* clustering are different from those computed by *k*-means clustering. The latter will assign observations to clusters to maximize the distances between clusters. The *EM* algorithm does not compute actual assignments of observations to clusters, but classification *probabilities*. In other words, each observation belongs to each cluster with a certain probability. Of course, as a final result you can usually review an actual assignment of observations to clusters, based on the (largest) classification probability.

Finding the Right Number of Clusters in *k*-Means and EM Clustering: v-Fold Cross-Validation

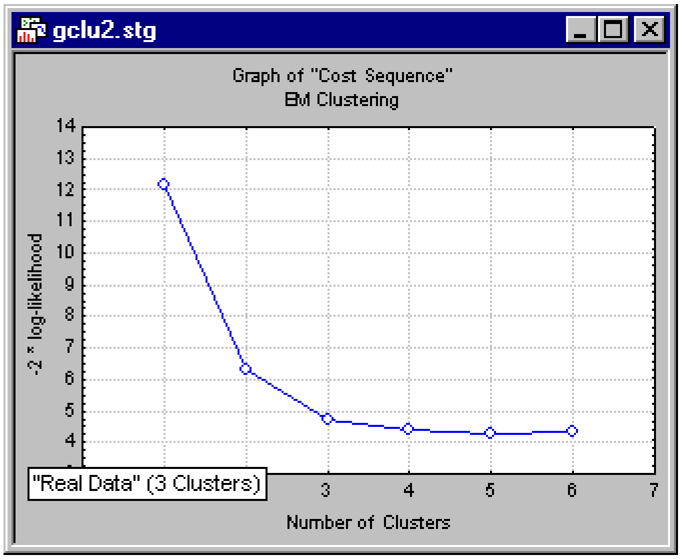
An important question that needs to be answered before applying the *k*-means or *EM* clustering algorithms is how many clusters there are in the data. This is not known *a priori* and, in fact, there might be no definite or unique answer as to what value *k* should take. In other words, *k* is a nuisance parameter of the clustering model. Luckily, an estimate of *k* can be obtained from the data using the method of cross-validation. Remember that the *k*-means and *EM* methods will determine cluster solutions for a particular user-defined number of clusters. The *k*-means and *EM* clustering techniques (described above) can be optimized and enhanced for typical applications in data mining. The general metaphor of data mining implies the situation in which an analyst searches for useful structures and "nuggets" in the data, usually without any strong *a priori* expectations of what the analysist might find (in contrast to the hypothesis-testing approach of scientific research). In practice, the analyst usually does not know ahead of time how many clusters there might be in the sample. For that reason, some programs include an implementation of a *v-fold cross-validation* algorithm for automatically determining the number of clusters in the data.

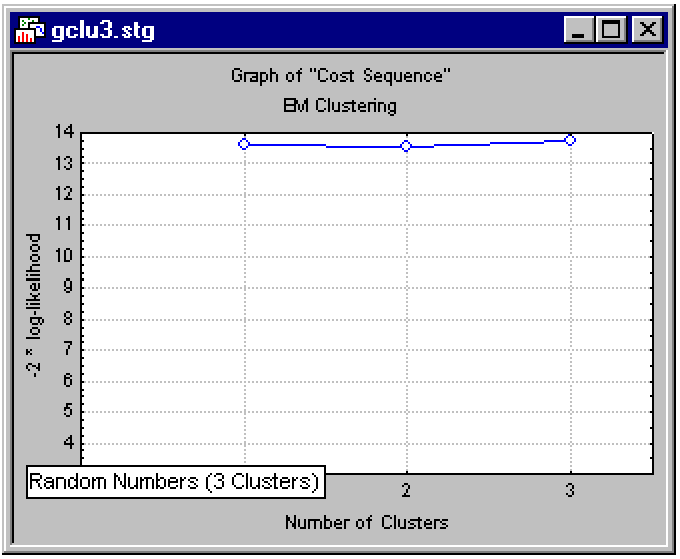
This unique algorithm is immensely useful in all general "pattern-recognition" tasks - to determine the number of market segments in a marketing research study, the number of distinct spending patterns in studies of consumer behavior, the number of clusters of different medical symptoms, the number of different types (clusters) of documents in text mining, the number of weather patterns in meteorological research, the number of defect patterns on silicon wafers, and so on.

The v-fold cross-validation algorithm applied to clustering. The v-fold cross-validation algorithm is described in some detail in *Classification Trees* and *General Classification and Regression Trees (GC&RT)*. The general idea of this method is to divide the overall sample into a number of *v* folds. The same type of analysis is then successively applied to the observations belonging to the *v-1* folds (training sample), and the results of the analyses are applied to sample *v* (the sample or fold that was not used to estimate the parameters, build the tree, determine the clusters, etc.; this is the testing sample) to compute some index of predictive validity. The results for the *v* replications are aggregated (averaged) to yield a single measure of the stability of the respective model, i.e., the validity of the model for predicting new observations.

Cluster analysis is an unsupervised learning technique, and we cannot observe the (real) number of clusters in the data. However, it is reasonable to replace the usual notion (applicable to supervised learning) of "accuracy" with that of "distance." In general, we can apply the *v-fold cross-validation* method to a range of numbers of clusters in *k*-means or *EM* clustering, and observe the resulting average distance of the observations (in the cross-validation or testing samples) from their cluster centers (for *k*-means clustering); for *EM* clustering, an appropriate equivalent measure would be the average negative (log-) likelihood computed for the observations in the testing samples.

Reviewing the results of v-fold cross-validation. The results of *v-fold cross-validation* are best reviewed in a simple line graph.





Shown here is the result of analyzing a data set widely known to contain three clusters of observations (specifically, the well-known *Iris* data file reported by Fisher, 1936, and widely referenced in the literature on discriminant function analysis). Also shown (in the graph to the right) are the results for analyzing simple normal random numbers. The "real" data (shown to the left) exhibit the characteristic scree-plot pattern (see also *Factor Analysis*), where the cost function (in this case, 2 times the log-likelihood of the cross-validation data, given the estimated parameters) quickly decreases as the number of clusters increases, but then (past 3 clusters) levels off, and even increases as the data are overfitted. Alternatively, the random numbers show no such pattern, in fact, there is basically no decrease in the cost function at all, and it quickly begins to increase as the number of clusters increases and overfitting occurs.

It is easy to see from this simple illustration how useful the *v-fold cross-validation* technique, applied to *k*-means and *EM* clustering can be for determining the "right" number of clusters in the data.

Chapter 11
Correspondence Analysis

- General Purpose
- Supplementary Points
- Multiple Correspondence Analysis
- Burt Tables

General Purpose

Correspondence analysis is a descriptive/exploratory technique designed to analyze simple two-way and multi-way tables containing some measure of correspondence between the rows and columns. The results provide information which is similar in nature to those produced by *Factor Analysis* techniques, and they allow one to explore the structure of categorical variables included in the table. The most common kind of table of this type is the two-way frequency crosstabulation table (see, for example, *Basic Statistics* or *Log-Linear*).

In a typical correspondence analysis, a crosstabulation table of frequencies is first standardized, so that the relative frequencies across all cells sum to 1.0. One way to state the goal of a typical analysis is to represent the entries in the table of relative frequencies in terms of the distances between individual rows and/or columns in a low-dimensional space. This is best illustrated by a simple example, which will be described below. There are several parallels in interpretation between correspondence analysis and *Factor Analysis*, and some similar concepts will also be pointed out below.

For a comprehensive description of this method, computational details, and its applications (in the English language), refer to the classic text by Greenacre (1984). These methods were originally developed primarily in France by Jean-Paul Benzérci in the early 1960's and 1970's (e.g., see Benzérci, 1973; see also Lebart, Morineau, and Tabard, 1977), but have only more recently gained increasing popularity in English-speaking countries (see, for example, Carrol, Green, and Schaffer, 1986; Hoffman and Franke, 1986). (Note that similar techniques were developed independently in several countries, where they were known as optimal scaling, reciprocal averaging, optimal scoring, quantification method, or homogeneity analysis). In the following paragraphs, a general introduction to correspondence analysis will be presented.

Overview. Suppose you collected data on the smoking habits of different employees in a company. The following data set is presented in Greenacre (1984, p. 55).

	Smoking Category				
Staff	**(1)**	**(2)**	**(3)**	**(4)**	**Row**

Group	None	Light	Medium	Heavy	Totals
(1) Senior Managers	4	2	3	2	11
(2) Junior Managers	4	3	7	4	18
(3) Senior Employees	25	10	12	4	51
(4) Junior Employees	18	24	33	13	88
(5) Secretaries	10	6	7	2	25
Column Totals	61	45	62	25	193

One may think of the 4 column values in each row of the table as coordinates in a 4-dimensional space, and one could compute the (Euclidean) distances between the 5 row points in the 4- dimensional space. The distances between the points in the 4-dimensional space summarize all information about the similarities between the rows in the table above. Now suppose one could find a lower-dimensional space, in which to position the row points in a manner that retains all, or almost all, of the information about the differences between the rows. You could then present all information about the similarities between the rows (types of employees in this case) in a simple 1, 2, or 3-dimensional graph. While this may not appear to be particularly useful for small tables like the one shown above, one can easily imagine how the presentation and interpretation of very large tables (e.g., differential preference for 10 consumer items among 100 groups of respondents in a consumer survey) could greatly benefit from the simplification that can be achieved via correspondence analysis (e.g., represent the 10 consumer items in a two- dimensional space).

Mass. To continue with the simpler example of the two-way table presented above, computationally, the program will first compute the relative frequencies for the frequency table, so that the sum of all table entries is equal to 1.0 (each element will be divided by the total, i.e., *193*). One could say that this table now shows how one unit of *mass* is distributed across the cells. In the terminology of correspondence analysis, the row and column totals of the matrix of relative frequencies are called the row mass and column mass, respectively.

Inertia. The term *inertia* in correspondence analysis is used by analogy with the definition in applied mathematics of "moment of inertia," which stands for the integral of mass times the squared distance to the centroid (e.g., Greenacre, 1984, p. 35). Inertia is defined as the total Pearson *Chi-square* for the two-way divided by the total sum (*193* in the present example).

Inertia and row and column profiles. If the rows and columns in a table are completely independent of each other, the entries in the table (distribution of mass) can be reproduced from the row and column totals alone, or row and column *profiles* in the terminology of correspondence analysis. According to the well-known formula for computing the *Chi-square* statistic for two-way tables, the expected frequencies in a table, where the column and rows are independent of each other, are equal to the respective column total times the row total, divided by the grand total. Any deviations from the expected values (expected under the hypothesis of complete independence of the row and column variables) will contribute to the overall *Chi-square*. Thus, another way of looking at correspondence analysis is to consider it a method for decomposing the overall *Chi-square* statistic (or *Inertia=Chi- square/Total N*) by identifying a small number of dimensions in which the deviations from the expected values can be represented. This is similar to the goal of *Factor Analysis*, where the total variance is decomposed, so as to arrive at a lower-dimensional representation of the variables that allows one to reconstruct most of the variance/covariance matrix of variables.

Analyzing rows and columns. This simple example began with a discussion of the row-points in the table shown above. However, one may rather be interested in the column totals, in which case one could plot the column points in a small-dimensional space, which satisfactorily reproduces the similarity (and distances) between the relative frequencies for the columns, across the rows, in the table shown above. In fact it is customary to simultaneously plot the column points and the row points in a single graph, to summarize the information contained in a two-way table.

Reviewing results. Let us now look at some of the results for the table shown above. First, shown below are the so-called *singular values* , *eigenvalues*, *percentages of inertia explained*, *cumulative percentages*, and the contribution to the overall *Chi- square*.

Eigenvalues and Inertia for all Dimensions
Input Table (Rows x Columns): 5 x 4
Total Inertia = .08519 Chi^2 = 16.442

No. of Dims	Singular Values	Eigen-Values	Perc. of Inertia	Cumulatv Percent	Chi Squares
1	.273421	.074759	87.75587	87.7559	14.42851
2	.100086	.010017	11.75865	99.5145	1.93332
3	.020337	.000414	.48547	100.0000	.07982

Note that the dimensions are "extracted" so as to maximize the distances between the row or column points, and successive dimensions (which are independent of or orthogonal to each other) will "explain" less and less of the overall *Chi-square* value (and, thus, inertia). Thus, the extraction of the dimensions is similar to the extraction of *principal components* in *Factor Analysis*.

First, it appears that, with a single dimension, 87.76% of the inertia can be "explained," that is, the relative frequency values that can be reconstructed from a single dimension can reproduce 87.76% of the total *Chi-square* value (and, thus, of the inertia) for this two-way table; two dimensions allow you to explain 99.51%.

Maximum number of dimensions. Since the sums of the frequencies across the columns must be equal to the row totals, and the sums across the rows equal to the column totals, there are in a sense only (*no. of columns-1*) independent entries in each row, and (*no. of rows-1*) independent entries in each column of the table (once you know what these entries are, you can fill in the rest based on your knowledge of the column and row marginal totals). Thus, the maximum number of eigenvalues that can be extracted from a two- way table is equal to the minimum of the number of columns minus 1, and the number of rows minus 1. If you choose to extract (i.e., interpret) the maximum number of dimensions that can be extracted, then you can reproduce exactly all information contained in the table.

Row and column coordinates. Next look at the coordinates for the two-dimensional solution.

Row Name	Dim. 1	Dim. 2
(1) Senior Managers	-.065768	.193737
(2) Junior Managers	.258958	.243305
(3) Senior Employees	-.380595	.010660
(4) Junior Employees	.232952	-.057744
(5) Secretaries	-.201089	-.078911

Of course, you can plot these coordinates in a two-dimensional scatterplot. Remember that the purpose of correspondence analysis is to reproduce the distances between the row and/or column points in a two-way table in a lower-dimensional display; note that, as in *Factor Analysis*, the actual rotational orientation of the axes is arbitrarily chosen so that successive dimensions "explain" less and less of the overall *Chi-square* value (or inertia). You could, for example, reverse the signs in each column in the table shown above, thereby effectively rotating the respective axis in the plot by 180 degrees.

What is important are the distances of the points in the two-dimensional display, which are informative in that row points that are close to each other are similar with regard to the pattern of relative frequencies across the columns. If you have produced this plot you will see that, along the most important first axis in the plot, the *Senior employees* and *Secretaries* are relatively close together on the left side of the origin (scale position 0). If you looked at the table of relative row frequencies (i.e., frequencies standardized, so that their sum in each row is equal to 100%), you will see that these two groups of employees indeed show very similar patterns of relative frequencies across the categories of smoking intensity.

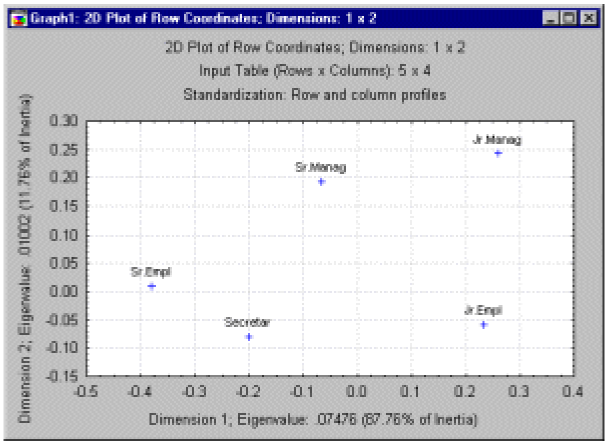
Percentages of Row Totals					
	Smoking Category				
Staff Group	(1) None	(2) Light	(3) Medium	(4) Heavy	Row Totals
(1) Senior Managers	36.36	18.18	27.27	18.18	100.00
(2) Junior Managers	22.22	16.67	38.89	22.22	100.00
(3) Senior Employees	49.02	19.61	23.53	7.84	100.00
(4) Junior Employees	20.45	27.27	37.50	14.77	100.00
(5) Secretaries	40.00	24.00	28.00	8.00	100.00

Obviously the final goal of correspondence analysis is to find theoretical interpretations (i.e., meaning) for the extracted dimensions. One method that may aid in interpreting extracted dimensions is to plot the column points. Shown below are the column coordinates for the first and second dimension.

Smoking category	Dim. 1	Dim. 2
None	-.393308	.030492
Light	.099456	-.141064
Medium	.196321	-.007359
Heavy	.293776	.197766

It appears that the first dimension distinguishes mostly between the different degrees of smoking, and in particular between category None and the others. Thus one can interpret the greater similarity of *Senior Managers* with *Secretaries*, with regard to their position on the first axis, as mostly deriving from the relatively large numbers of *None* smokers in these two groups of employees.

Compatibility of row and column coordinates. It is customary to summarize the row and column coordinates in a single plot. However, it is important to remember that in such plots, one can only interpret the distances between row points, and the distances between column points, but not the distances between row points and column points.



To continue with this example, it would not be appropriate to say that the category *None* is similar to *Senior Employees* (the two points are very close in the simultaneous plot of row and column coordinates). However, as was indicated earlier, it is appropriate to make general statements about the nature of the dimensions, based on which side of the origin particular points fall. For example, because category *None* is the only column point on the left side of the origin for the first axis, and since employee group *Senior Employees* also falls onto that side of the first axis, one may conclude that the first axis separates *None* smokers from the other categories of smokers, and that *Senior Employees* are different from, for example, *Junior Employees*, in that there are relatively more non-smoking *Senior Employees*.

Scaling of the coordinates (standardization options). Another important decision that the analyst must make concerns the scaling of the coordinates. The nature of the choice pertains to whether or not you want to analyze the relative row percentages, column percentages, or both. In the context of the example described above, the row percentages were shown to illustrate how the patterns of those percentages across the columns are similar for points which appear more closely together in the graphical display of the row coordinates. Put another way, the coordinates are based on the analysis of the *row profile matrix*, where the sum of the table entries in a row, across all columns, is equal to 1.0 (each entry r_{ij} in the row profile matrix can be interpreted as the conditional probability that a case belongs to column *j*, given its membership in row i). Thus, the coordinates are computed so as to maximize the differences between the points with respect to the row profiles (row percentages). The row coordinates are computed from the row profile matrix, the column coordinates are computed from the column profile matrix.

A fourth option, *Canonical* standardization (see Gifi, 1981), is also provided, and it amounts to a standardization of the columns and rows of the matrix of relative frequencies. This standardization amounts to a rescaling of the coordinates based on the row profile standardization and the column profile standardization, and this type of standardization is not widely used. Note also that a variety of other custom standardizations can be easily performed if you have the raw *eigenvalues* and *eigenvector* matrices.

Metric of coordinate system. In several places in this introduction, the term *distance* was (loosely) used to refer to the differences between the pattern of relative frequencies for the rows across the columns, and columns across the rows, which are to be reproduced in a lower-dimensional solution as a result of the correspondence analysis. Actually, these distances represented by the coordinates in the respective space are not simple Euclidean distances computed from the relative row or column frequencies, but rather, they are weighted distances. Specifically, the weighting that is applied is such that the metric in the lower- dimensional space is a *Chi-square* metric, provided that (1) you are comparing row points, and chose either row-profile standardization or both row- and column-profile standardization, or (2) you are comparing column points, and chose either column-profile standardization or both row- and column-profile standardization.

In that case (but *not* if you chose the canonical standardization), the squared Euclidean distance between, for example, two row points *i* and *i'* in the respective coordinate system of a given number of dimensions actually approximates a weighted (i.e., *Chi-square*) distance between the relative frequencies (see Hoffman and Franke, 1986, formula 21):

$$d_{ii'}^{2} = \Sigma_j \left(1/c_j \left(p_{ij} /r_i - p^2_{i'j} /r_{i'}\right)\right)$$

In this formula, $d_{ii'}$ stands for the squared distance between the two points, c_j stands for the column total for the *j*'th column of the standardized frequency table (where the sum of all entries or mass is equal to *1.0*), p_{ij} stands for the individual cell entries in the standardized frequency table (row *i*, column *j*), r_i stands for the row total for the *i*'th column of the relative frequency table, and the summation $\Sigma$is over the columns of the table. To reiterate, only the distances between row points, and correspondingly, between column points are interpretable in this manner; the distances between row points and column points cannot be interpreted.

Judging the quality of a solution. A number of auxiliary statistics are reported, to aid in the evaluation of the quality of the respective chosen numbers of dimensions. The general concern here is that all (or at least most) points are properly represented by the respective solution, that is, that their distances to other points can be approximated to a satisfactory degree. Shown below are all statistics reported for the row coordinates for the example table discussed so far, based on a one-dimensional solution only (i.e., only one dimension is used to reconstruct the patterns of relative frequencies across the columns).

Row Coordinates and Contributions to Inertia

Staff Group	Coordin. Dim.1	Mass	Quality	Relative Inertia	Inertia Dim.1	Cosine^2 Dim.1
(1) Senior Managers	-.065768	.056995	.092232	.031376	.003298	.092232
(2) Junior Managers	.258958	.093264	.526400	.139467	.083659	.526400
(3) Senior Employees	-.380595	.264249	.999033	.449750	.512006	.999033
(4) Junior Employees	.232952	.455959	.941934	.308354	.330974	.941934
(5) Secretaries	-.201089	.129534	.865346	.071053	.070064	.865346

Coordinates. The first numeric column shown in the table above contains the coordinates, as discussed in the previous paragraphs. To reiterate, the specific interpretation of these coordinates depends on the standardization chosen for the solution (see above). The number of dimensions is chosen by the user (in this case we chose only one dimension), and coordinate values will be shown for each dimension (i.e., there will be one column with coordinate values for each dimension).

Mass. The *Mass* column contains the row totals (since these are the row coordinates) for the table of relative frequencies (i.e., for the table where each entry is the respective *mass*, as discussed earlier in this section). Remember that the coordinates are computed based on the matrix of conditional probabilities shown in the *Mass* column.

Quality. The *Quality* column contains information concerning the quality of representation of the respective row point in the coordinate system defined by the respective numbers of dimensions, as chosen by the user. In the table shown above, only one dimension was chosen, and the numbers in the *Quality* column pertain to the quality of representation in the one-dimensional space. To reiterate, computationally, the goal of the correspondence analysis is to reproduce the distances between points in a low-dimensional space. If you extracted (i.e., interpreted) the maximum number of dimensions (which is equal to the minimum of the number of rows and the number of columns, minus 1), you could reconstruct all distances exactly. The *Quality* of a point is defined as the ratio of the squared distance of the point from the origin in the chosen number of dimensions, over the squared distance from the origin in the space defined by the maximum number of dimensions (remember that the metric here is *Chi-square*, as described earlier). By analogy to *Factor Analysis*, the quality of a point is similar in its interpretation to the communality for a variable in factor analysis.

Note that the *Quality* measure reported is independent of the chosen method of standardization, and always pertains to the default standardization (i.e., the distance metric is *Chi-square*, and the quality measure can be interpreted as the "proportion of *Chi- square* accounted for" for the respective row, given the respective number of dimensions). A low quality means that the current number of dimensions does not well represent the respective row (or column). In the table shown above, the quality for the first row (*Senior Managers*) is less than *.1*, indicating that this row point is not well represented by the one- dimensional representation of the points.

Relative inertia. The *Quality* of a point (see above) represents the proportion of the contribution of that point to the overall inertia (*Chi-square*) that can be accounted for by the chosen number of dimensions. However, it does not indicate whether or not, and to what extent, the respective point does in fact contribute to the overall inertia (*Chi- square* value). The relative inertia represents the proportion of the total inertia accounted for by the respective point, and it is independent of the number of dimensions chosen by the user. Note that a particular solution may represent a point very well (high *Quality*), but the same point may not contribute much to the overall inertia (e.g., a row point with a pattern of relative frequencies across the columns that is similar to the average pattern across all rows).

Relative inertia for each dimension. This column contains the relative contribution of the respective (row) point to the inertia "accounted for" by the respective dimension. Thus, this value will be reported for each (row or column) point, for each dimension.

Cosine^2 (quality or squared correlations with each dimension). This column contains the quality for each point, by dimension. The sum of the values in these columns across the dimensions is equal to the total *Quality* value discussed above (since in the example table above, only one dimension was chose, the values in this column are identical to the values in the overall *Quality* column). This value may also be interpreted as the "correlation" of the respective point with the respective dimension. The term *Cosine^2* refers to the fact that this value is also the squared cosine value of the angle the point makes with the respective dimension (refer to Greenacre, 1984, for details concerning the geometric aspects of correspondence analysis).

A note about "statistical significance." It should be noted at this point that correspondence analysis is an exploratory technique. Actually, the method was developed based on a philosophical orientation that emphasizes the development of models that fit the data, rather than the rejection of hypotheses based on the lack of fit (Benzecri's "second principle" states that "The model must fit the data, not vice versa;" see Greenacre, 1984, p. 10). Therefore, there are no statistical significance tests that are customarily applied to the results of a correspondence analysis; the primary purpose of the technique is to produce a simplified (low- dimensional) representation of the information in a large frequency table (or tables with similar measures of correspondence).

Supplementary Points

The introductory section provides an overview of how to interpret the coordinates and related statistics computed in a correspondence analysis. An important aid in the interpretation of the results from a correspondence analysis is to include supplementary row or column points, that were not used to perform the original analyses. For example, consider the following results which are based on the example given in the introductory (based on Greenacre, 1984).

Row Name	**Dim. 1**	**Dim. 2**
(1) Senior Managers	-.065768	.193737
(2) Junior Managers	.258958	.243305
(3) Senior Employees	-.380595	.010660
(4) Junior Employees	.232952	-.057744
(5) Secretaries	-.201089	-.078911
National Average	-.258368	-.117648

The table above shows the coordinate values (for two dimensions) computed for a frequency table of different types of employees by type of smoking habit. The row labeled *National Average* contains the coordinate values for the supplementary point, which is the national average (percentages) for the different smoking categories (which make up the columns of the table; those fictitious percentages reported in Greenacre (1984) are: Nonsmokers: 42%, light smokers: 29%, medium smokers, 20%; heavy smokers: 9%). If you plotted these coordinates in a two-dimensional scatterplot, along with the column coordinates, it would be apparent that the *National Average* supplementary row point is plotted close to the point representing the Secretaries group, and on the same side of the horizontal axis (first dimension) as the *Nonsmokers* column point. If you refer back to the original two-way table shown in the introductory section, this finding is consistent with the entries in the table of row frequencies, that is, there are relatively more nonsmokers among the *Secretaries*, and in the *National Average*. Put another way, the sample represented in the original frequency table contains more smokers than the national average.

While this type of information could have been easily gleaned from the original frequency table (that was used as the input to the analysis), in the case of very large tables, such conclusions may not be as obvious.

Quality of representation of supplementary points. Another interesting result for supplementary points concerns the quality of their representation in the chosen number of dimensions (see the introductory section for a more detailed discussion of the concept of *quality of representation*). To reiterate, the goal of the correspondence analysis is to reproduce the distances between the row or column coordinates (patterns of relative frequencies across the columns or rows, respectively) in a low-dimensional solution. Given such a solution, one may ask whether particular supplementary points of interest can be represented equally well in the final space, that is, whether or not their distances from the other points in the table can also be represented in the chosen numbers of dimensions. Shown below are the summary statistics for the original points, and the supplementary row point *National Average*, for the two-dimensional solution.

Staff Group	Quality	Cosine² Dim.1	Cosine² Dim.2
(1) Senior Managers	.892568	.092232	.800336
(2) Junior Managers	.991082	.526400	.464682
(3) Senior Employees	.999817	.999033	.000784
(4) Junior Employees	.999810	.941934	.057876
(5) Secretaries	.998603	.865346	.133257
National Average	.761324	.630578	.130746

The statistics reported in the table above are discussed in the introductory section. In short, the *Quality* of a row or column point is defined as the ratio of the squared distance of the point from the origin in the chosen number of dimensions, over the squared distance from the origin in the space defined by the maximum number of dimensions (remember that the metric here is *Chi-square*, as described in the introductory section). In a sense, the overall quality is the "proportion of squared distance-from-the-overall-centroid accounted for." The supplementary row point *National Average* has a quality of *.76*, indicating that it is reasonably well represented in the two-dimensional solution. The *Cosine*^{2} statistic is the quality "accounted for" by the respective row point, by the respective dimension (the sum of the *Cosine*^{2} values over the respective number of dimensions is equal to the total *Quality*, see also the introductory section).

Multiple Correspondence Analysis (MCA)

Multiple correspondence analysis (MCA) may be considered to be an extension of simple correspondence analysis to more than two variables. For an introductory overview of simple correspondence analysis, refer to the introductory section . Multiple correspondence analysis is a simple correspondence analysis carried out on an indicator (or design) matrix with cases as rows and categories of variables as columns. Actually, one usually analyzes the inner product of such a matrix, called the *Burt Table* in an MCA; this will be discussed later. However, to clarify the interpretation of the results from a multiple correspondence analysis, it is easier to discuss the simple correspondence analysis of an indicator or design matrix.

Indicator or design matrix. Consider again the simple two-way table presented in the introductory section:

	Smoking Category				
Staff Group	(1) None	(2) Light	(3) Medium	(4) Heavy	Row Totals
(1) Senior Managers	4	2	3	2	11
(2) Junior Managers	4	3	7	4	18
(3) Senior Employees	25	10	12	4	51
(4) Junior Employees	18	24	33	13	88
(5) Secretaries	10	6	7	2	25
Column Totals	61	45	62	25	193

Suppose you had entered the data for this table in the following manner, as an *indicator* or *design* matrix:

	Staff Group					Smoking			
Case Number	Senior Manager	Junior Manager	Senior Employee	Junior Employee	Secretary	None	Light	Medium	Heavy
1	1	0	0	0	0	1	0	0	0
2	1	0	0	0	0	1	0	0	0
3	1	0	0	0	0	1	0	0	0
4	1	0	0	0	0	1	0	0	0
5	1	0	0	0	0	0	1	0	0
...	.	.	.	.	.	.	.	.	.
...	.	.	.	.	.	.	.	.	.
...	.	.	.	.	.	.	.	.	.
191	0	0	0	0	1	0	0	1	0
192	0	0	0	0	1	0	0	0	1
193	0	0	0	0	1	0	0	0	1

Each one of the 193 total cases in the table is represented by one case in this data file. For each case a *1* is entered into the category where the respective case "belongs," and a *0* otherwise. For example, case *1* represents a *Senior Manager* who is a *None* smoker. As can be seen in the table above, there are a total of 4 such cases in the two-way table, and thus there will be four cases like this in the indicator matrix. In all, there will be *193* cases in the indicator or design matrix.

Analyzing the design matrix. If you now analyzed this data file (design or indicator matrix) shown above as if it were a two-way frequency table, the results of the correspondence analysis would provide column coordinates that would allow you to relate the different categories to each other, based on the distances between the row points, i.e., between the individual cases. In fact, the two-dimensional display you would obtain for the column coordinates would look very similar to the combined display for row and column coordinates, if you had performed the simple correspondence analysis on the two-way frequency table (note that the metric will be different, but the relative positions of the points will be very similar).

More than two variables. The approach to analyzing categorical data outlined above can easily be extended to more than two categorical variables. For example, the indicator or design matrix could contain two additional variables *Male* and *Female*, again coded *0* and *1*, to indicate the subjects' gender; and three variables could be added to indicate to which one of three age groups a case belongs. Thus, in the final display, one could represent the relationships (similarities) between *Gender, Age, Smoking habits,* and *Occupation (Staff Groups).*

Fuzzy coding. It is not necessary that each case is assigned exclusively to only one category of each categorical variable. Rather than the *0*-or-*1* coding scheme, one could enter probabilities for membership in a category, or some other measure that represents a fuzzy rule for group membership. Greenacre (1984) discusses different types of coding schemes of this kind. For example, suppose in the example design matrix shown earlier, you had missing data for a few cases regarding their smoking habits. Instead of discarding those cases entirely from the analysis (or creating a new category *Missing data*), you could assign to the different smoking categories proportions (which should add to 1.0) to represent the probabilities that the respective case belongs to the respective category (e.g., you could enter proportions based on your knowledge about estimates for the national averages for the different categories).

Interpretation of coordinates and other results. To reiterate, the results of a multiple correspondence analysis are identical to the results you would obtain for the column coordinates from a simple correspondence analysis of the design or indicator matrix. Therefore, the interpretation of coordinate values, quality values, cosine^2's and other statistics reported as the results from a multiple correspondence analysis can be interpreted in the same manner as described in the context of the simple correspondence analysis (see introductory section), however, these statistics pertain to the total inertia associated with the entire design matrix.

Supplementary column points and "multiple regression" for categorical variables. Another application of the analysis of design matrices via correspondence analysis techniques is that it allows you to perform the equivalent of a *Multiple Regression* for categorical variables, by adding supplementary columns to the design matrix. For example, suppose you added to the design matrix shown earlier two columns to indicate whether or not the respective subject had or had not been ill over the past year (i.e., you could add one column *Ill* and another column *Not ill*, and again enter *0*'s and *1*'s to indicate each subject's health status). If, in a simple correspondence analysis of the design matrix, you added those columns as supplementary columns to the analysis, then (1) the summary statistics for the quality of representation (see the introductory section) for those columns would give you an indication of how well you can "explain" illness as a function of the other variables in the design matrix, and (2) the display of the column points in the final coordinate system would provide an indication of the nature (e.g., direction) of the relationships between the columns in the design matrix and the column points indicating illness; this technique (adding supplementary points to an MCA analysis) is also sometimes called *predictive mapping*.

The Burt table. The actual computations in multiple correspondence analysis are not performed on a design or indicator matrix (which, potentially, may be very large if there are many cases), but on the inner product of this matrix; this matrix is also called the *Burt* matrix. With frequency tables, this amounts to tabulating the stacked categories against each other; for example the *Burt* for the two-way frequency table presented earlier would look like this.

	Employee					**Smoking**			
	(1)	**(2)**	**(3)**	**(4)**	**(5)**	**(1)**	**(2)**	**(3)**	**(4)**
(1) Senior Managers	11	0	0	0	0	4	2	3	2
(2) Junior Managers	0	18	0	0	0	4	3	7	4
(3) Senior Employees	0	0	51	0	0	25	10	12	4
(4) Junior Employees	0	0	0	88	0	18	24	33	13
(5) Secretaries	0	0	0	0	25	10	6	7	2
(1) Smoking:None	4	4	25	18	10	61	0	0	0
(2) Smoking:Light	2	3	10	24	6	0	45	0	0
(3) Smoking:Medium	3	7	12	33	7	0	0	62	0
(4) Smoking:Heavy	2	4	4	13	2	0	0	0	25

The *Burt* has a clearly defined structure. In the case of two categorical variables (shown above), it consists of 4 partitions: (1) the crosstabulation of variable *Employee* against itself, (2) the crosstabulation of variable *Employee* against variable *Smoking*, (3), the crosstabulation of variable *Smoking* against variable *Employee*, and (4) the crosstabulation of variable *Smoking* against itself. Note that the matrix is symmetrical, and that the sum of the diagonal elements in each partition representing the crosstabulation of a variable against itself must be the same (e.g., there were a total of 193 observations in the present example, and hence, the diagonal elements in the crosstabulation tables of variable *Employee* against itself, and *Smoking* against itself must also be equal to 193).

Note that the off-diagonal elements in the partitions representing the crosstabulations of a variable against itself are equal to *0* in the table shown above. However, this is not necessarily always the case, for example, when the *Burt* was derived from a design or indicator matrix that included fuzzy coding of category membership (see above).

Burt Tables

The Burt table is the result of the inner product of a design or indicator matrix, and the multiple correspondence analysis results are identical to the results one would obtain for the column points from a simple correspondence analysis of the indicator or design matrix (see also *MCA*).

For example, suppose you had entered data concerning the *Survival* for different *Age* groups in different *Locations* like this:

	SURVIVAL		AGE			LOCATION		
Case No.	**NO**	**YES**	**LESST50**	**A50TO69**	**OVER69**	**TOKYO**	**BOSTON**	**GLAMORGN**
1	0	1	0	1	0	0	0	1
2	1	0	1	0	0	1	0	0
3	0	1	0	1	0	0	1	0
4	0	1	0	0	1	0	0	1
...	.	.	.	.	.	.	.	.
...	.	.	.	.	.	.	.	.
...	.	.	.	.	.	.	.	.
762	1	0	0	1	0	1	0	0
763	0	1	1	0	0	0	1	0
764	0	1	0	1	0	0	0	1

In this data arrangement, for each case a *1* was entered to indicate to which category, of a particular set of categories, a case belongs (e.g., *Survival*, with the categories *No* and *Yes*). For example, case *1* survived (a *0* was entered for variable *No*, and a 1 was entered for variable *Yes*), case *1* is between age 50 and 69 (a *1* was entered for variable *A50to69*), and was observed in *Glamorgn*). Overall there are *764* observations in the data set.

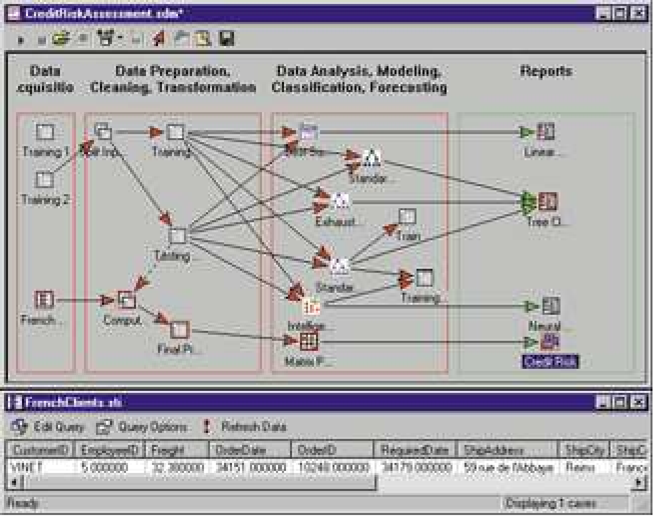
If you denote the data (design or indicator matrix) shown above as matrix X, then matrix product X'X is a *Burt* table); shown below is an example of a *Burt* table that one might obtain in this manner.

	SURVIVAL		AGE			LOCATION		
	NO	**YES**	**<50**	**50-69**	**69+**	**TOKYO**	**BOSTON**	**GLAMORGN**
SURVIVAL:NO	210	0	68	93	49	60	82	68
SURVIVAL:YES	0	554	212	258	84	230	171	153
AGE:UNDER_50	68	212	280	0	0	151	58	71
AGE:A_50TO69	93	258	0	351	0	120	122	109
AGE:OVER_69	49	84	0	0	133	19	73	41
LOCATION:TOKYO	60	230	151	120	19	290	0	0
LOCATION:BOSTON	82	171	58	122	73	0	253	0
LOCATION:GLAMORGN	68	153	71	109	41	0	0	221

The *Burt* table has a clearly defined structure. Overall, the data matrix is symmetrical. In the case of 3 categorical variables (as shown above), the data matrix consists 3 x 3 = 9 partitions, created by each variable being tabulated against itself, and against the categories of all other variables. Note that the sum of the diagonal elements in each diagonal partition (i.e., where the respective variables are tabulated against themselves) is constant (equal to 764 in this case).

The off-diagonal elements in each diagonal partition in this example are all *0*. If the cases in the design or indicator matrix are assigned to categories via fuzzy coding (i.e., if probabilities are used to indicate likelihood of membership in a category, rather than 0/1 coding to indicate actual membership), then the off-diagonal elements of the diagonal partitions are not necessarily equal to 0.

Chapter 12
Data Mining Techniques



CustomerID	EmployeeID	Freight	OrderDate	OrderID	RequiredDate	ShipAddress	ShipCity	ShipC
VINET	5.000000	32.380000	34151.000000	10248.000000	34179.000000	59 rue de l'Abbaye	Reims	Franc

- Data Mining
- Crucial Concepts in Data Mining
- Data Warehousing
- On-Line Analytic Processing (OLAP)
- Exploratory Data Analysis (EDA) and Data Mining Techniques
  - EDA vs. Hypothesis Testing
  - Computational EDA Techniques
  - Graphical (data visualization) EDA techniques
  - Verification of results of EDA
- Neural Networks

Data Mining

Data Mining is an analytic process designed to explore data (usually large amounts of data - typically business or market related) in search of consistent patterns and/or systematic relationships between variables, and then to validate the findings by applying the detected patterns to new subsets of data. The ultimate goal of data mining is prediction - and predictive data mining is the most common type of data mining and one that has the most direct business applications. The process of data mining consists of three stages: (1) the initial exploration, (2) model building or pattern identification with validation/verification, and (3) deployment (i.e., the application of the model to new data in order to generate predictions).

Stage 1: Exploration. This stage usually starts with data preparation which may involve cleaning data, data transformations, selecting subsets of records and - in case of data sets with large numbers of variables ("fields") - performing some preliminary feature selection operations to bring the number of variables to a manageable range (depending on the statistical methods which are being considered). Then, depending on the nature of the analytic problem, this first stage of the process of data mining may involve anywhere between a simple choice of straightforward predictors for a regression model, to elaborate exploratory analyses using a wide variety of graphical and statistical methods (see *Exploratory Data Analysis (EDA)*) in order to identify the most relevant variables and determine the complexity and/or the general nature of models that can be taken into account in the next stage.

Stage 2: Model building and validation. This stage involves considering various models and choosing the best one based on their predictive performance (i.e., explaining the variability in question and producing stable results across samples). This may sound like a simple operation, but in fact, it sometimes involves a very elaborate process. There are a variety of techniques developed to achieve that goal - many of which are based on so-called "competitive evaluation of models," that is, applying different models to the same data set and then comparing their performance to choose the best. These techniques

- which are often considered the core of predictive data mining - include: Bagging (Voting, Averaging), Boosting, Stacking (Stacked Generalizations), and Meta-Learning.

Stage 3: Deployment. That final stage involves using the model selected as best in the previous stage and applying it to new data in order to generate predictions or estimates of the expected outcome.

The concept of *Data Mining* is becoming increasingly popular as a business information management tool where it is expected to reveal knowledge structures that can guide decisions in conditions of limited certainty. Recently, there has been increased interest in developing new analytic techniques specifically designed to address the issues relevant to business *Data Mining* (e.g., *Classification Trees*), but Data Mining is still based on the conceptual principles of statistics including the traditional *Exploratory Data Analysis (EDA)* and modeling and it shares with them both some components of its general approaches and specific techniques.

However, an important general difference in the focus and purpose between *Data Mining* and the traditional *Exploratory Data Analysis (EDA)* is that *Data Mining* is more oriented towards applications than the basic nature of the underlying phenomena. In other words, *Data Mining* is relatively less concerned with identifying the specific relations between the involved variables. For example, uncovering the nature of the underlying functions or the specific types of interactive, multivariate dependencies between variables are not the main goal of *Data Mining*. Instead, the focus is on producing a solution that can generate useful predictions. Therefore, *Data Mining* accepts among others a "black box" approach to data exploration or knowledge discovery and uses not only the traditional *Exploratory Data Analysis (EDA)* techniques, but also such techniques as *Neural Networks* which can generate valid predictions but are not capable of identifying the specific nature of the interrelations between the variables on which the predictions are based.

Data Mining is often considered to be "*a blend of statistics, AI [artificial intelligence], and data base research*" (Pregibon, 1997, p. 8), which until very recently was not commonly recognized as a field of interest for statisticians, and was even considered by some "*a dirty word in Statistics*" (Pregibon, 1997, p. 8). Due to its applied importance, however, the field emerges as a rapidly growing and major area (also in statistics) where important theoretical advances are being made (see, for example, the recent annual *International Conferences on Knowledge Discovery and Data Mining*, co-hosted by the *American Statistical Association*).

For information on *Data Mining* techniques, please review the summary topics included below in this chapter of the *Electronic Statistics Textbook*. There are numerous books that review the theory and practice of data mining; the following books offer a representative sample of recent general books on data mining, representing a variety of approaches and perspectives:

Berry, M., J., A., & Linoff, G., S., (2000). *Mastering data mining*. New York: Wiley.

Edelstein, H., A. (1999). *Introduction to data mining and knowledge discovery (3rd ed)*. Potomac, MD: Two Crows Corp.

Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., & Uthurusamy, R. (1996). *Advances in knowledge discovery & data mining*. Cambridge, MA: MIT Press.

Han, J., Kamber, M. (2000). *Data mining: Concepts and Techniques*. New York: Morgan-Kaufman.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2001). *The elements of statistical learning : Data mining, inference, and prediction*. New York: Springer.

Pregibon, D. (1997). *Data Mining*. Statistical Computing and Graphics, 7, 8.

Weiss, S. M., & Indurkhya, N. (1997). *Predictive data mining: A practical guide*. New York: Morgan-Kaufman.

Westphal, C., Blaxton, T. (1998). *Data mining solutions*. New York: Wiley.

Witten, I. H., & Frank, E. (2000). *Data mining*. New York: Morgan-Kaufmann.

Crucial Concepts in Data Mining

Bagging (Voting, Averaging)
The concept of bagging (voting for classification, averaging for regression-type problems with continuous dependent variables of interest) applies to the area of predictive data mining, to combine the predicted classifications (prediction) from multiple models, or from the same type of model for different learning data. It is also used to address the inherent instability of results when applying complex models to relatively small data sets. Suppose your data mining task is to build a model for predictive classification, and the dataset from which to train the model (learning data set, which contains observed classifications) is relatively small. You could repeatedly sub-sample (with replacement) from the dataset, and apply, for example, a tree classifier (e.g., C&RT and CHAID) to the successive samples. In practice, very different trees will often be grown for the different samples, illustrating the instability of models often evident with small datasets. One method of deriving a single prediction (for new observations) is to use all trees found in the different samples, and to apply some simple voting: The final classification is the one most often predicted by the different trees. Note that some weighted combination of predictions (weighted vote, weighted average) is also possible, and commonly used. A sophisticated (machine learning) algorithm for generating weights for weighted prediction or voting is the Boosting procedure.

Boosting
The concept of boosting applies to the area of predictive data mining, to generate multiple models or classifiers (for prediction or classification), and to derive weights to

combine the predictions from those models into a single prediction or predicted classification (see also Bagging).

A simple algorithm for boosting works like this: Start by applying some method (e.g., a tree classifier such as C&RT or CHAID) to the learning data, where each observation is assigned an equal weight. Compute the predicted classifications, and apply weights to the observations in the learning sample that are inversely proportional to the accuracy of the classification. In other words, assign greater weight to those observations that were difficult to classify (where the misclassification rate was high), and lower weights to those that were easy to classify (where the misclassification rate was low). In the context of C&RT for example, different misclassification costs (for the different classes) can be applied, inversely proportional to the accuracy of prediction in each class. Then apply the classifier again to the weighted data (or with different misclassification costs), and continue with the next iteration (application of the analysis method for classification to the re-weighted data).

Boosting will generate a sequence of classifiers, where each consecutive classifier in the sequence is an "expert" in classifying observations that were not well classified by those preceding it. During deployment (for prediction or classification of new cases), the predictions from the different classifiers can then be combined (e.g., via voting, or some weighted voting procedure) to derive a single best prediction or classification.

Note that boosting can also be applied to learning methods that do not explicitly support weights or misclassification costs. In that case, random sub-sampling can be applied to the learning data in the successive steps of the iterative boosting procedure, where the probability for selection of an observation into the subsample is inversely proportional to the accuracy of the prediction for that observation in the previous iteration (in the sequence of iterations of the boosting procedure).

CRISP

See Models for Data Mining.

Data Preparation (in Data Mining)

Data preparation and cleaning is an often neglected but extremely important step in the data mining process. The old saying "garbage-in-garbage-out" is particularly applicable to the typical data mining projects where large data sets collected via some automatic methods (e.g., via the Web) serve as the input into the analyses. Often, the method by which the data where gathered was not tightly controlled, and so the data may contain out-of-range values (e.g., Income: -100), impossible data combinations (e.g., Gender: Male, Pregnant: Yes), and the like. Analyzing data that has not been carefully screened for such problems can produce highly misleading results, in particular in predictive data mining.

Data Reduction (for Data Mining)

The term Data Reduction in the context of data mining is usually applied to projects where the goal is to aggregate or amalgamate the information contained in large datasets into manageable (smaller) information nuggets. Data reduction methods can include simple tabulation, aggregation (computing descriptive statistics) or more sophisticated techniques like clustering, principal components analysis, etc.

See also predictive data mining, drill-down analysis.

Deployment
The concept of deployment in predictive data mining refers to the application of a model for prediction or classification to new data. After a satisfactory model or set of models has been identified (trained) for a particular application, one usually wants to deploy those models so that predictions or predicted classifications can quickly be obtained for new data. For example, a credit card company may want to deploy a trained model or set of models (e.g., neural networks, meta-learner) to quickly identify transactions which have a high probability of being fraudulent.

Drill-Down Analysis
The concept of drill-down analysis applies to the area of data mining, to denote the interactive exploration of data, in particular of large databases. The process of drill-down analyses begins by considering some simple break-downs of the data by a few variables of interest (e.g., Gender, geographic region, etc.). Various statistics, tables, histograms, and other graphical summaries can be computed for each group. Next one may want to "drill-down" to expose and further analyze the data "underneath" one of the categorizations, for example, one might want to further review the data for males from the mid-west. Again, various statistical and graphical summaries can be computed for those cases only, which might suggest further break-downs by other variables (e.g., income, age, etc.). At the lowest ("bottom") level are the raw data: For example, you may want to review the addresses of male customers from one region, for a certain income group, etc., and to offer to those customers some particular services of particular utility to that group.

Feature Selection
One of the preliminary stage in predictive data mining, when the data set includes more variables than could be included (or would be efficient to include) in the actual model building phase (or even in initial exploratory operations), is to select predictors from a large list of candidates. For example, when data are collected via automated (computerized) methods, it is not uncommon that measurements are recorded for thousands or hundreds of thousands (or more) of predictors. The standard analytic methods for predictive data mining, such as neural network analyses, classification and regression trees, generalized linear models, or general linear models become impractical when the number of predictors exceed more than a few hundred variables.

Feature selection selects a subset of predictors from a large list of candidate predictors without assuming that the relationships between the predictors and the dependent or outcome variables of interest are linear, or even monotone. Therefore, this is used as a pre-processor for predictive data mining, to select manageable sets of predictors that are likely related to the dependent (outcome) variables of interest, for further analyses with any of the other methods for regression and classification.

Machine Learning
Machine learning, computational learning theory, and similar terms are often used in the context of *Data Mining*, to denote the application of generic model-fitting or classification algorithms for predictive data mining. Unlike traditional statistical data analysis, which is usually concerned with the estimation of population parameters by statistical inference, the emphasis in data mining (and machine learning) is usually on the accuracy of prediction (predicted classification), regardless of whether or not the "models" or techniques that are used to generate the prediction is interpretable or open to simple explanation. Good examples of this type of technique often applied to predictive data mining are neural networks or meta-learning techniques such as boosting, etc. These methods usually involve the fitting of very complex "generic" models, that are not related to any reasoning or theoretical understanding of underlying causal processes; instead, these techniques can be shown to generate accurate predictions or classification in crossvalidation samples.

Meta-Learning
The concept of meta-learning applies to the area of predictive data mining, to combine the predictions from multiple models. It is particularly useful when the types of models included in the project are very different. In this context, this procedure is also referred to as Stacking (Stacked Generalization).

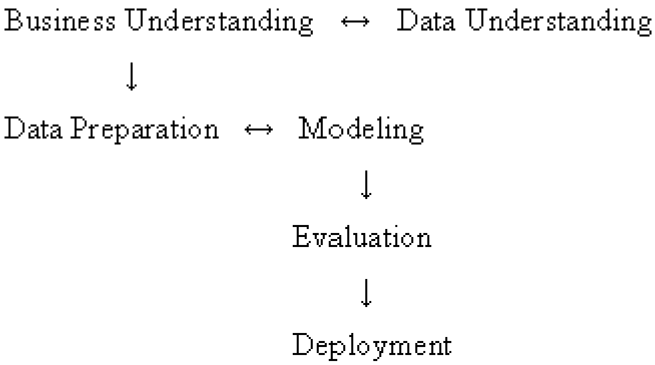
Suppose your data mining project includes tree classifiers, such as C&RT and CHAID, linear discriminant analysis (e.g., see GDA), and *Neural Networks*. Each computes predicted classifications for a crossvalidation sample, from which overall goodness-of-fit statistics (e.g., misclassification rates) can be computed. Experience has shown that combining the predictions from multiple methods often yields more accurate predictions than can be derived from any one method (e.g., see Witten and Frank, 2000). The predictions from different classifiers can be used as input into a meta-learner, which will attempt to combine the predictions to create a final best predicted classification. So, for example, the predicted classifications from the tree classifiers, linear model, and the neural network classifier(s) can be used as input variables into a neural network meta-classifier, which will attempt to "learn" from the data how to combine the predictions from the different models to yield maximum classification accuracy.

One can apply meta-learners to the results from different meta-learners to create "meta-meta"-learners, and so on; however, in practice such exponential increase in the amount of data processing, in order to derive an accurate prediction, will yield less and less marginal utility.

Models for Data Mining
In the business environment, complex data mining projects may require the coordinate efforts of various experts, stakeholders, or departments throughout an entire organization. In the data mining literature, various "general frameworks" have been proposed to serve

as blueprints for how to organize the process of gathering data, analyzing data, disseminating results, implementing results, and monitoring improvements.

One such model, CRISP (Cross-Industry Standard Process for data mining) was proposed in the mid-1990s by a European consortium of companies to serve as a non-proprietary standard process model for data mining. This general approach postulates the following (perhaps not particularly controversial) general sequence of steps for data mining projects:



Another approach - the Six Sigma methodology - is a well-structured, data-driven methodology for eliminating defects, waste, or quality control problems of all kinds in manufacturing, service delivery, management, and other business activities. This model has recently become very popular (due to its successful implementations) in various American industries, and it appears to gain favor worldwide. It postulated a sequence of, so-called, DMAIC steps -

Define → Measure → Analyze → Improve → Control

- that grew up from the manufacturing, quality improvement, and process control traditions and is particularly well suited to production environments (including "production of services," i.e., service industries).

Another framework of this kind (actually somewhat similar to Six Sigma) is the approach proposed by SAS Institute called SEMMA -

Sample → Explore → Modify → Model → Assess

- which is focusing more on the technical activities typically involved in a data mining project.

All of these models are concerned with the process of how to integrate data mining methodology into an organization, how to "convert data into information," how to involve important stake-holders, and how to disseminate the information in a form that can easily be converted by stake-holders into resources for strategic decision making.

Some software tools for data mining are specifically designed and documented to fit into one of these specific frameworks.

The general underlying philosophy of StatSoft's *STATISTICA* Data Miner is to provide a flexible data mining workbench that can be integrated into any organization, industry, or organizational culture, regardless of the general data mining process-model that the organization chooses to adopt. For example, *STATISTICA* Data Miner can include the complete set of (specific) necessary tools for ongoing company wide Six Sigma quality control efforts, and users can take advantage of its (still optional) DMAIC-centric user interface for industrial data mining tools. It can equally well be integrated into ongoing marketing research, CRM (Customer Relationship Management) projects, etc. that follow either the CRISP or SEMMA approach - it fits both of them perfectly well without favoring either one. Also, *STATISTICA* Data Miner offers all the advantages of a general data mining oriented "development kit" that includes easy to use tools for incorporating into your projects not only such components as custom database gateway solutions, prompted interactive queries, or proprietary algorithms, but also systems of access privileges, workgroup management, and other collaborative work tools that allow you to design large scale, enterprise-wide systems (e.g., following the CRISP, SEMMA, or a combination of both models) that involve your entire organization.

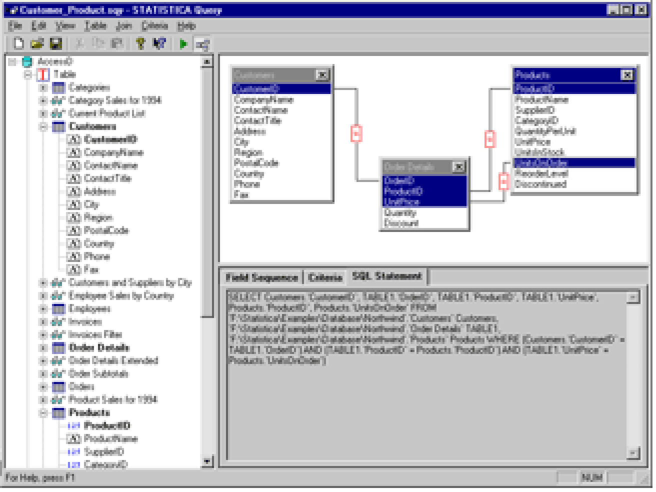
Predictive Data Mining
The term Predictive Data Mining is usually applied to identify data mining projects with the goal to identify a statistical or neural network model or set of models that can be used to predict some response of interest. For example, a credit card company may want to engage in predictive data mining, to derive a (trained) model or set of models (e.g., neural networks, meta-learner) that can quickly identify transactions which have a high probability of being fraudulent. Other types of data mining projects may be more exploratory in nature (e.g., to identify cluster or segments of customers), in which case drill-down descriptive and exploratory methods would be applied. Data reduction is another possible objective for data mining (e.g., to aggregate or amalgamate the information in very large data sets into useful and manageable chunks).

SEMMA
See Models for Data Mining.

Stacked Generalization
See Stacking.

Stacking (Stacked Generalization)
The concept of stacking (short for Stacked Generalization) applies to the area of predictive data mining, to combine the predictions from multiple models. It is particularly useful when the types of models included in the project are very different.

Suppose your data mining project includes tree classifiers, such as C&RT or CHAID, linear discriminant analysis (e.g., see GDA), and *Neural Networks*. Each computes predicted classifications for a crossvalidation sample, from which overall goodness-of-fit



```
SELECT Customers.`CustomerID`, TABLE1.`OrderID`, TABLE1.`ProductID`, TABLE1.`UnitPrice`,
Products.`ProductID`, Products.`UnitsOnOrder` FROM
`F:\Statistica\Examples\Database\Northwind`.`Customers` Customers,
`F:\Statistica\Examples\Database\Northwind`.`Order Details` TABLE1,
`F:\Statistica\Examples\Database\Northwind`.`Products` Products WHERE (Customers.`CustomerID` =
TABLE1.`OrderID`) AND (TABLE1.`ProductID` = Products.`ProductID`) AND (TABLE1.`UnitPrice` =
Products.`UnitsOnOrder`)
```

statistics (e.g., misclassification rates) can be computed. Experience has shown that combining the predictions from multiple methods often yields more accurate predictions than can be derived from any one method (e.g., see Witten and Frank, 2000). In stacking, the predictions from different classifiers are used as input into a meta-learner, which attempts to combine the predictions to create a final best predicted classification. So, for example, the predicted classifications from the tree classifiers, linear model, and the neural network classifier(s) can be used as input variables into a neural network meta-classifier, which will attempt to "learn" from the data how to combine the predictions from the different models to yield maximum classification accuracy.

Other methods for combining the prediction from multiple models or methods (e.g., from multiple datasets used for learning) are Boosting and Bagging (Voting).

Text Mining

While *Data Mining* is typically concerned with the detection of patterns in numeric data, very often important (e.g., critical to business) information is stored in the form of text. Unlike numeric data, text is often amorphous, and difficult to deal with. Text mining generally consists of the analysis of (multiple) text documents by extracting key phrases, concepts, etc. and the preparation of the text processed in that manner for further analyses with numeric data mining techniques (e.g., to determine co-occurrences of concepts, key phrases, names, addresses, product names, etc.).

Voting

See Bagging.

Data Warehousing

StatSoft defines *data warehousing* as a process of organizing the storage of large, multivariate data sets in a way that facilitates the retrieval of information for analytic purposes.

The most efficient data warehousing architecture will be capable of incorporating or at least referencing all data available in the relevant enterprise-wide information management systems, using designated technology suitable for corporate data base management (e.g., *Oracle*, *Sybase*, *MS SQL Server*. Also, a flexible, high-performance (see the IDP technology), open architecture approach to data warehousing - that flexibly integrates with the existing corporate systems and allows the users to organize and efficiently reference for analytic purposes enterprise repositories of data of practically any complexity - is offered in StatSoft enterprise systems such as *SEDAS* (*STATISTICA Enterprise-wide Data Analysis System*) and *SEWSS* (*STATISTICA Enterprise-wide SPC System*), which can also work in conjunction with *STATISTICA Data Miner* and *WebSTATISTICA Server Applications*.

On-Line Analytic Processing (OLAP)

The term *On-Line Analytic Processing - OLAP* (or *Fast Analysis of Shared Multidimensional Information - FASMI*) refers to technology that allows users of multidimensional databases to generate on-line descriptive or comparative summaries ("views") of data and other analytic queries. Note that despite its name, analyses referred to as *OLAP* do not need to be performed truly "on-line" (or in real-time); the term applies to analyses of multidimensional databases (that may, obviously, contain dynamically updated information) through efficient "multidimensional" queries that reference various types of data. *OLAP* facilities can be integrated into corporate (enterprise-wide) database systems and they allow analysts and managers to monitor the performance of the business (e.g., such as various aspects of the manufacturing process or numbers and types of completed transactions at different locations) or the market. The final result of *OLAP* techniques can be very simple (e.g., frequency tables, descriptive statistics, simple cross-tabulations) or more complex (e.g., they may involve seasonal adjustments, removal of outliers, and other forms of cleaning the data). Although Data Mining techniques can operate on any kind of unprocessed or even unstructured information, they can also be applied to the data views and summaries generated by *OLAP* to provide more in-depth and often more multidimensional knowledge. In this sense, Data Mining techniques could be considered to represent either a different analytic approach (serving different purposes than *OLAP*) or as an analytic extension of *OLAP*.

Exploratory Data Analysis (EDA)

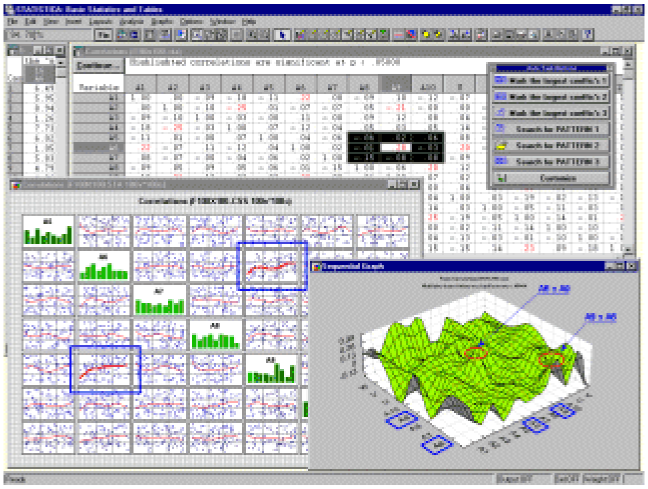
EDA vs. Hypothesis Testing

As opposed to traditional *hypothesis testing* designed to verify *a priori* hypotheses about relations between variables (e.g., *"There is a positive correlation between the AGE of a person and his/her RISK TAKING disposition"*), *exploratory data analysis (EDA)* is used to identify systematic relations between variables when there are no (or not complete) *a priori* expectations as to the nature of those relations. In a typical exploratory data analysis process, many variables are taken into account and compared, using a variety of techniques in the search for systematic patterns.

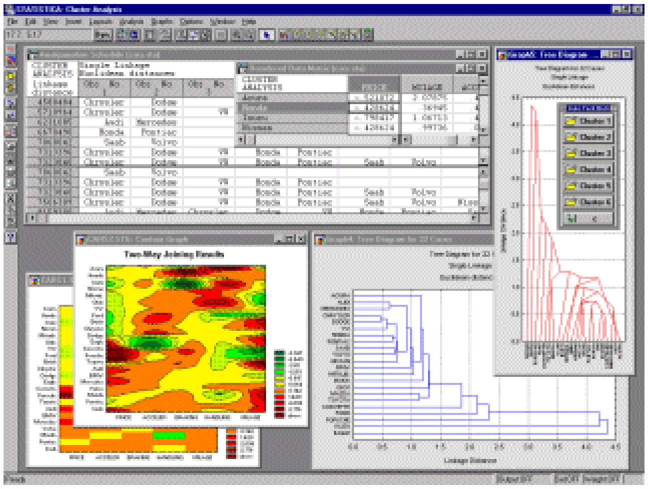
Computational EDA techniques

Computational exploratory data analysis methods include both simple basic statistics and more advanced, designated multivariate exploratory techniques designed to identify patterns in multivariate data sets.

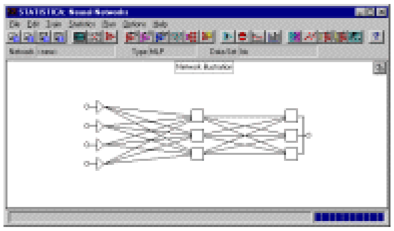
Basic statistical exploratory methods. The basic statistical exploratory methods include such techniques as examining distributions of variables (e.g., to identify highly skewed or non-normal, such as bi-modal patterns), reviewing large correlation matrices for coefficients that meet certain thresholds (see example above), or examining multi-way frequency tables (e.g., "slice by slice" systematically reviewing combinations of levels of control variables).


STATISTICA: Basic Statistics and Tables
File Edit View Insert Layouts Analysis Graphs Options Window Help
Continue...
Variable
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
Mark the largest coeffic's 1
Mark the largest coeffic's 2
Mark the largest coeffic's 3
Search for PATTERN 1
Search for PATTERN 2
Search for PATTERN 3
Continue
Sequential Graph
Ready

Multivariate exploratory techniques. Multivariate exploratory techniques designed specifically to identify patterns in multivariate (or univariate, such as sequences of measurements) data sets include: Cluster Analysis, Factor Analysis, Discriminant Function Analysis, Multidimensional Scaling, Log-linear Analysis, Canonical Correlation, Stepwise Linear and Nonlinear (e.g., Logit) Regression, Correspondence Analysis, Time Series Analysis, and Classification Trees.



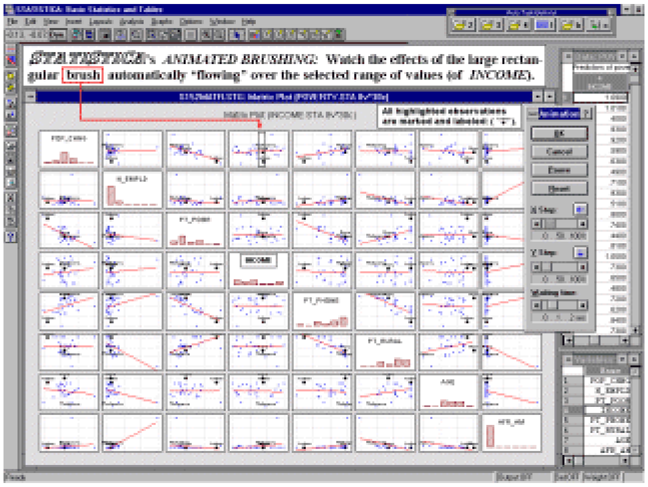
Neural Networks. *Neural Networks* are analytic techniques modeled after the (hypothesized) processes of learning in the cognitive system and the neurological functions of the brain and capable of predicting new observations (on specific variables) from other observations (on the same or other variables) after executing a process of so-called learning from existing data.



For more information, see Neural Networks; see also *STATISTICA Neural Networks*.

Graphical (data visualization) EDA techniques

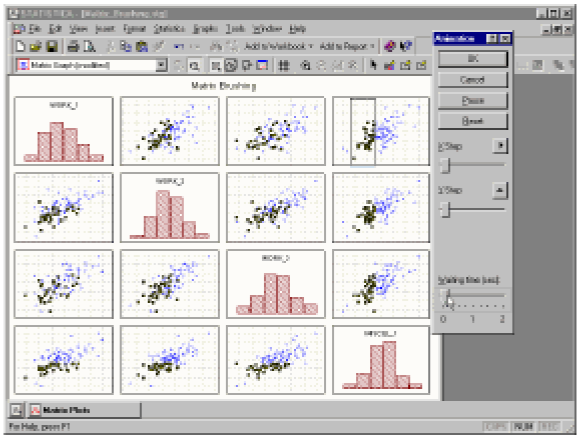
A large selection of powerful exploratory data analytic techniques is also offered by graphical data visualization methods that can identify relations, trends, and biases "hidden" in unstructured data sets.

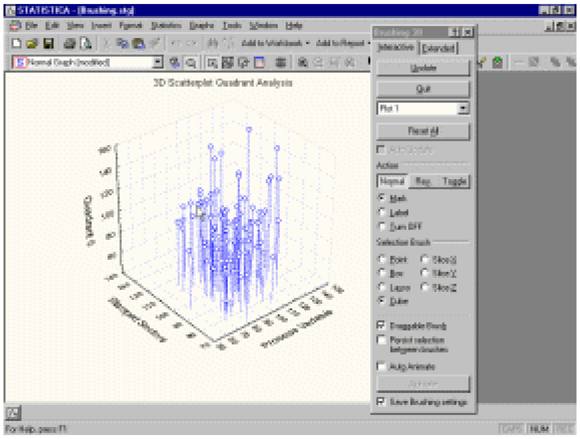


STATISTICA's ANIMATED BRUSHING: Watch the effects of the large rectangular brush automatically "flowing" over the selected range of values (of INCOME).

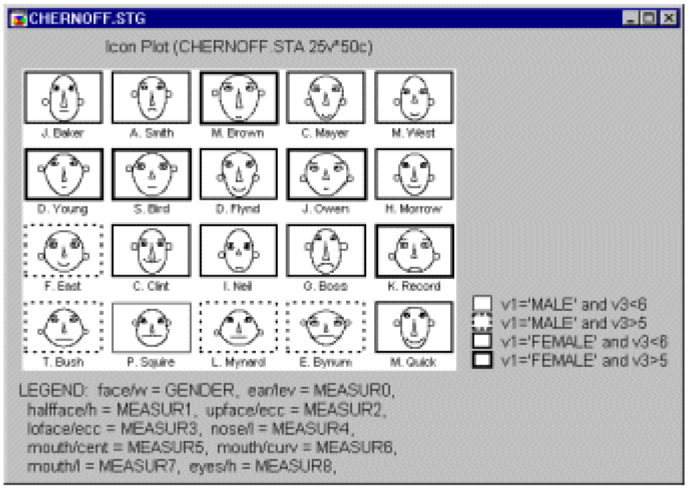
All highlighted observations are marked and labeled ("+").

Brushing. Perhaps the most common and historically first widely used technique explicitly identified as *graphical exploratory data analysis* is brushing, an interactive method allowing one to select on-screen specific data points or subsets of data and identify their (e.g., common) characteristics, or to examine their effects on relations between relevant variables. Those relations between variables can be visualized by fitted functions (e.g., 2D lines or 3D surfaces) and their confidence intervals, thus, for example, one can examine changes in those functions by interactively (temporarily) removing or adding specific subsets of data. For example, one of many applications of the brushing technique is to select (i.e., highlight) in a matrix scatterplot all data points that belong to a certain category (e.g., a "medium" income level, see the highlighted subset in the fourth component graph of the first row in the illustration left) in order to examine how those specific observations contribute to relations between other variables in the same data set (e.g, the correlation between the "debt" and "assets" in the current example). If the brushing facility supports features like "animated brushing" or "automatic function re-fitting", one can define a dynamic brush that would move over the consecutive ranges of a criterion variable (e.g., "income" measured on a continuous scale or a discrete [3-level] scale as on the illustration above) and examine the dynamics of the contribution of the criterion variable to the relations between other relevant variables in the same data set.



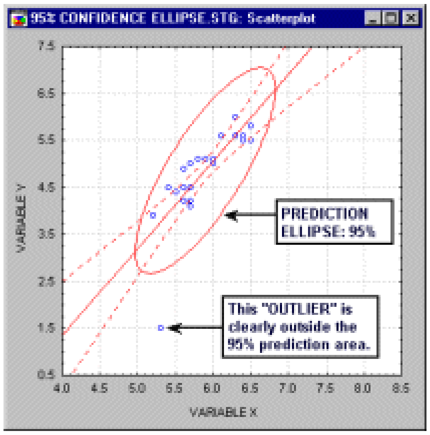


Other graphical EDA techniques. Other graphical exploratory analytic techniques include function fitting and plotting, data smoothing, overlaying and merging of multiple displays, categorizing data, splitting/merging subsets of data in graphs, aggregating data in graphs, identifying and marking subsets of data that meet specific conditions, icon plots,



LEGEND: face/w = GENDER, ear/lev = MEASUR0,
halfface/h = MEASUR1, upface/ecc = MEASUR2,
loface/ecc = MEASUR3, nose/l = MEASUR4,
mouth/cent = MEASUR5, mouth/curv = MEASUR6,
mouth/l = MEASUR7, eyes/h = MEASUR8,

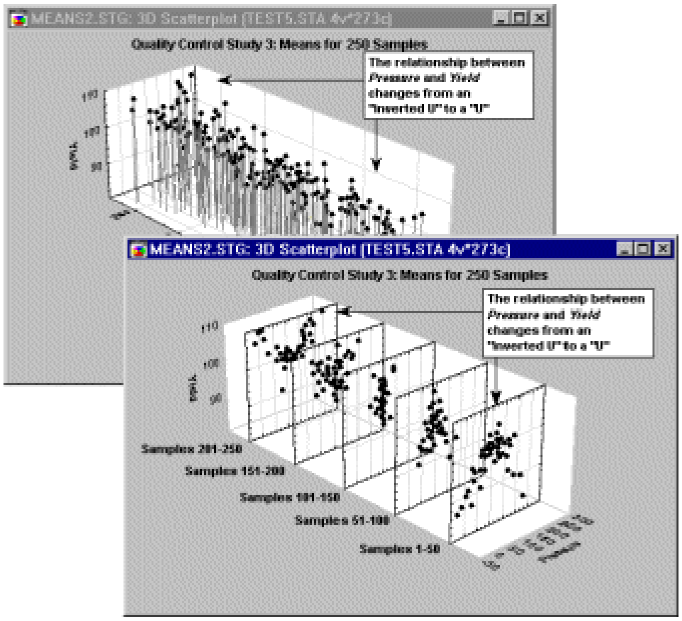
shading, plotting confidence intervals and confidence areas (e.g., ellipses),



PREDICTION ELLIPSE: 95%

This "OUTLIER" is clearly outside the 95% prediction area.

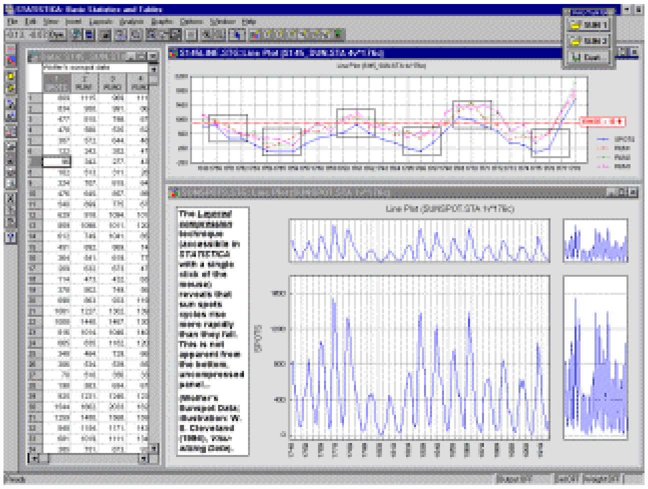
generating tessellations, spectral planes,


MEANS2.STG: 3D Scatterplot (TEST5.STA 4v*273c)
Quality Control Study 3: Means for 250 Samples
Yield
110
100
90
Samples 201-250
Samples 151-200
Samples 101-150
Samples 51-100
Samples 1-50
Pressure

The relationship between *Pressure* and *Yield* changes from an "inverted U" to a "U"

The relationship between *Pressure* and *Yield* changes from an "inverted U" to a "U"

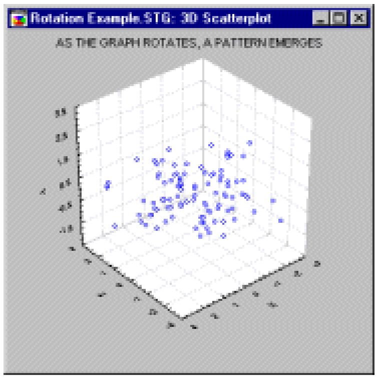
integrated layered compressions,


STATISTICA: Basic Statistics and Tables
File Edit View Insert Layouts Analysis Graphs Options Window Help
Wolfer's sunspot data
SPOTS
RUN1
RUN2
RUN3
Line Plot (SUNSPOT.STA 1v*176c)
SPOTS
Ready

The Layered compression technique (accessible in STATISTICA with a single click of the mouse) reveals that sun spots cycles rise more rapidly than they fall. This is not apparent from the bottom, uncompressed panel...

(Wolfer's Sunspot Data; Illustration: W. S. Cleveland (1993), Visualizing Data).

and projected contours, data image reduction techniques, interactive (and continuous) rotation


Rotation Example.STG: 3D Scatterplot
AS THE GRAPH ROTATES, A PATTERN EMERGES

with animated stratification (cross-sections) of 3D displays, and selective highlighting of specific series and blocks of data.

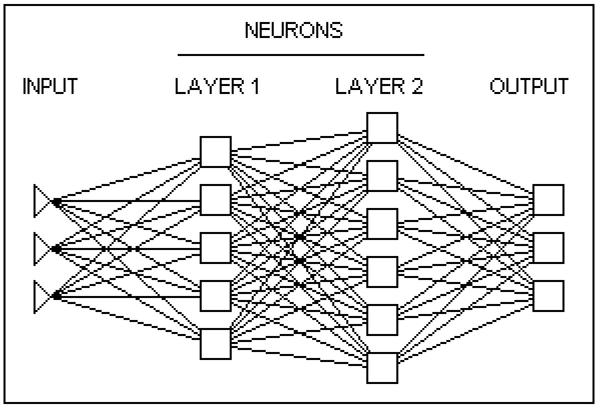
Verification of results of EDA

The exploration of data can only serve as the first stage of data analysis and its results can be treated as tentative at best as long as they are not confirmed, e.g., crossvalidated, using a different data set (or and independent subset). If the result of the exploratory stage suggests a particular model, then its validity can be verified by applying it to a new data set and testing its fit (e.g., testing its *predictive validity*). Case selection conditions can be used to quickly define subsets of data (e.g., for estimation and verification), and for testing the robustness of results.

Neural Networks

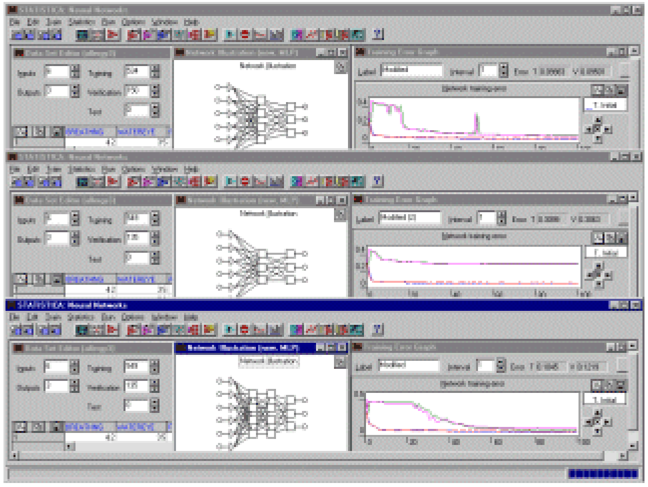
(see also Neural Networks chapter)

Neural Networks are analytic techniques modeled after the (hypothesized) processes of learning in the cognitive system and the neurological functions of the brain and capable of predicting new observations (on specific variables) from other observations (on the same or other variables) after executing a process of so-called *learning* from existing data. Neural Networks is one of the Data Mining techniques.



The first step is to design a specific network architecture (that includes a specific number of "layers" each consisting of a certain number of "neurons"). The size and structure of the network needs to match the nature (e.g., the formal complexity) of the investigated phenomenon. Because the latter is obviously not known very well at this early stage, this task is not easy and often involves multiple "trials and errors." (Now, there is, however, neural network software that applies artificial intelligence techniques to aid in that tedious task and finds "the best" network architecture.)

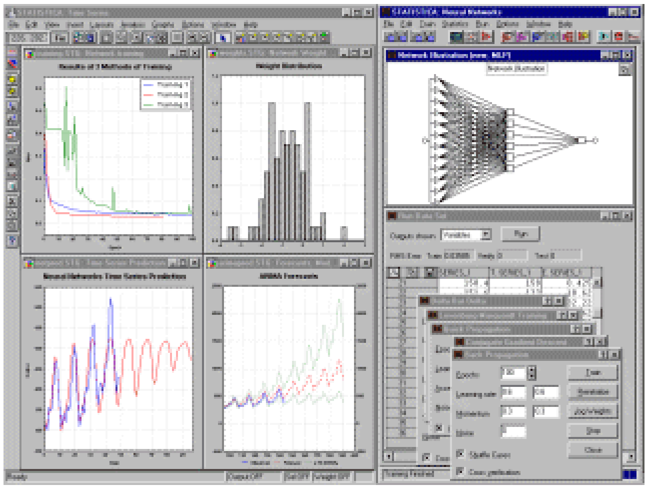
The new network is then subjected to the process of "training." In that phase, neurons apply an iterative process to the number of inputs (variables) to adjust the weights of the network in order to optimally predict (in traditional terms one could say, find a "fit" to) the sample data on which the "training" is performed. After the phase of learning from an existing data set, the new network is ready and it can then be used to generate predictions.



The resulting *"network"* developed in the process of *"learning"* represents a pattern detected in the data. Thus, in this approach, the *"network"* is the functional equivalent of a model of relations between variables in the traditional *model building* approach. However, unlike in the traditional *models*, in the *"network,"* those relations cannot be articulated in the usual terms used in statistics or methodology to describe relations between variables (such as, for example, *"A is positively correlated with B but only for* *observations where the value of C is low and D is high"*). Some *neural networks* can produce highly accurate predictions; they represent, however, a typical a-theoretical (one can say, "a black box") research approach. That approach is concerned only with practical considerations, that is, with the predictive validity of the solution and its applied relevance and not with the nature of the underlying mechanism or its relevance for any "theory" of the underlying phenomena.

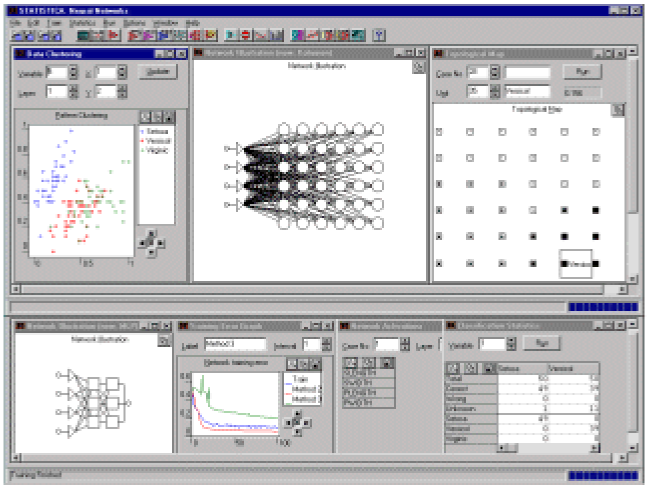
However, it should be mentioned that *Neural Network* techniques can also be used as a component of analyses designed to build explanatory models because *Neural Networks* can help explore data sets in search for relevant variables or groups of variables; the results of such explorations can then facilitate the process of model building. Moreover, now there is neural network software that uses sophisticated algorithms to search for the most relevant input variables, thus potentially contributing directly to the model building process.

One of the major advantages of *neural networks* is that, theoretically, they are capable of approximating any continuous function, and thus the researcher does not need to have any hypotheses about the underlying model, or even to some extent, which variables matter. An important disadvantage, however, is that the final solution depends on the initial conditions of the network, and, as stated before, it is virtually impossible to "interpret" the solution in traditional, analytic terms, such as those used to build theories that explain phenomena.


STATISTICA: Time Series
STATISTICA: Neural Networks
Results of 3 Methods of Training
Weight Distribution
Network Illustration (one: MLP)
Neural Networks Time Series Prediction
ARIMA Forecasts
Epochs
Learning rate
Momentum
Noise
Train
Stop
Close
Training Finished
Ready

Some authors stress the fact that *neural networks* use, or one should say, are expected to use, massively parallel computation models. For example Haykin (1994) defines *neural network* as:

"a massively parallel distributed processor that has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects: (1) Knowledge is acquired by the network through a learning process, and (2) Interneuron connection strengths known as synaptic weights are used to store the knowledge." (p. 2).



However, as Ripley (1996) points out, the vast majority of contemporary neural network applications run on single-processor computers and he argues that a large speed-up can be achieved not only by developing software that will take advantage of multiprocessor hardware by also by designing better (more efficient) learning algorithms.

Neural networks is one of the methods used in Data Mining; see also Exploratory Data Analysis. For more information on *neural networks*, see Haykin (1994), Masters (1995), Ripley (1996), and Welstead (1994). For a discussion of *neural networks* as statistical tools, see Warner and Misra (1996). See also, *STATISTICA Neural Networks*.

Chapter 13
Discriminant Function Analysis

- General Purpose
- Computational Approach
- Stepwise Discriminant Analysis
- Interpreting a Two-Group Discriminant Function
- Discriminant Functions for Multiple Groups
- Assumptions
- Classification

General Purpose

Discriminant function analysis is used to determine which variables discriminate between two or more naturally occurring groups. For example, an educational researcher may want to investigate which variables discriminate between high school graduates who decide (1) to go to college, (2) to attend a trade or professional school, or (3) to seek no further training or education. For that purpose the researcher could collect data on numerous variables prior to students' graduation. After graduation, most students will naturally fall into one of the three categories. *Discriminant Analysis* could then be used to determine which variable(s) are the best predictors of students' subsequent educational choice.

A medical researcher may record different variables relating to patients' backgrounds in order to learn which variables best predict whether a patient is likely to recover completely (group 1), partially (group 2), or not at all (group 3). A biologist could record different characteristics of similar types (groups) of flowers, and then perform a discriminant function analysis to determine the set of characteristics that allows for the best discrimination between the types.

Computational Approach

Computationally, discriminant function analysis is very similar to analysis of variance (*ANOVA*). Let us consider a simple example. Suppose we measure height in a random sample of 50 males and 50 females. Females are, on the average, not as tall as males, and this difference will be reflected in the difference in means (for the variable *Height*). Therefore, variable height allows us to discriminate between males and females with a better than chance probability: if a person is tall, then he is likely to be a male, if a person is short, then she is likely to be a female.

We can generalize this reasoning to groups and variables that are less "trivial." For example, suppose we have two groups of high school graduates: Those who choose to attend college after graduation and those who do not. We could have measured students' stated intention to continue on to college one year prior to graduation. If the means for the two groups (those who actually went to college and those who did not) are different, then we can say that intention to attend college as stated one year prior to graduation allows us to discriminate between those who are and are not college bound (and this information may be used by career counselors to provide the appropriate guidance to the respective students).

To summarize the discussion so far, the basic idea underlying discriminant function analysis is to determine whether groups differ with regard to the mean of a variable, and then to use that variable to predict group membership (e.g., of new cases).

Analysis of Variance. Stated in this manner, the discriminant function problem can be rephrased as a one-way analysis of variance (ANOVA) problem. Specifically, one can ask whether or not two or more groups are *significantly different* from each other with respect to the mean of a particular variable. To learn more about how one can test for the statistical significance of differences between means in different groups you may want to read the Overview section to *ANOVA/MANOVA*. However, it should be clear that, if the means for a variable are significantly different in different groups, then we can say that this variable discriminates between the groups.

In the case of a single variable, the final significance test of whether or not a variable discriminates between groups is the *F* test. As described in *Elementary Concepts* and *ANOVA /MANOVA*, *F* is essentially computed as the ratio of the between-groups variance in the data over the pooled (average) within-group variance. If the between-group variance is significantly larger then there must be significant differences between means.

Multiple Variables. Usually, one includes several variables in a study in order to see which one(s) contribute to the discrimination between groups. In that case, we have a matrix of total variances and covariances; likewise, we have a matrix of pooled within-group variances and covariances. We can compare those two matrices via multivariate *F* tests in order to determined whether or not there are any significant differences (with regard to all variables) between groups. This procedure is identical to multivariate analysis of variance or *MANOVA*. As in *MANOVA*, one could first perform the multivariate test, and, if statistically significant, proceed to see which of the variables have significantly different means across the groups. Thus, even though the computations with multiple variables are more complex, the principal reasoning still applies, namely, that we are looking for variables that discriminate between groups, as evident in observed mean differences.

Stepwise Discriminant Analysis

Probably the most common application of discriminant function analysis is to include many measures in the study, in order to determine the ones that discriminate between groups. For example, an educational researcher interested in predicting high school graduates' choices for further education would probably include as many measures of personality, achievement motivation, academic performance, etc. as possible in order to learn which one(s) offer the best prediction.

Model. Put another way, we want to build a "model" of how we can best predict to which group a case belongs. In the following discussion we will use the term "in the model" in order to refer to variables that are included in the prediction of group membership, and we will refer to variables as being "not in the model" if they are not included.

Forward stepwise analysis. In stepwise discriminant function analysis, a model of discrimination is built step-by-step. Specifically, at each step all variables are reviewed and evaluated to determine which one will contribute most to the discrimination between groups. That variable will then be included in the model, and the process starts again.

Backward stepwise analysis. One can also step backwards; in that case all variables are included in the model and then, at each step, the variable that contributes least to the prediction of group membership is eliminated. Thus, as the result of a successful discriminant function analysis, one would only keep the "important" variables in the model, that is, those variables that contribute the most to the discrimination between groups.

***F* to enter, *F* to remove.** The stepwise procedure is "guided" by the respective *F* to enter and *F* to remove values. The *F* value for a variable indicates its statistical significance in the discrimination between groups, that is, it is a measure of the extent to which a variable makes a unique contribution to the prediction of group membership. If you are familiar with stepwise multiple regression procedures, then you may interpret the *F* to enter/remove values in the same way as in stepwise regression.

Capitalizing on chance. A common misinterpretation of the results of stepwise discriminant analysis is to take statistical significance levels at face value. By nature, the stepwise procedures will capitalize on chance because they "pick and choose" the variables to be included in the model so as to yield maximum discrimination. Thus, when using the stepwise approach the researcher should be aware that the significance levels do not reflect the true *alpha* error rate, that is, the probability of erroneously rejecting H_0 (the null hypothesis that there is no discrimination between groups).

Interpreting a Two-Group Discriminant Function

In the two-group case, discriminant function analysis can also be thought of as (and is analogous to) multiple regression (see Multiple Regression; the two-group discriminant analysis is also called *Fisher linear discriminant analysis* after Fisher, 1936; computationally all of these approaches are analogous). If we code the two groups in the analysis as *1* and *2*, and use that variable as the dependent variable in a multiple regression analysis, then we would get results that are analogous to those we would obtain via *Discriminant Analysis*. In general, in the two-group case we fit a linear equation of the type:

$$Group = a + b_1*x_1 + b_2*x_2 + ... + b_m*x_m$$

where *a* is a constant and b_1 through b_m are regression coefficients. The interpretation of the results of a two-group problem is straightforward and closely follows the logic of multiple regression: Those variables with the largest (standardized) regression coefficients are the ones that contribute most to the prediction of group membership.

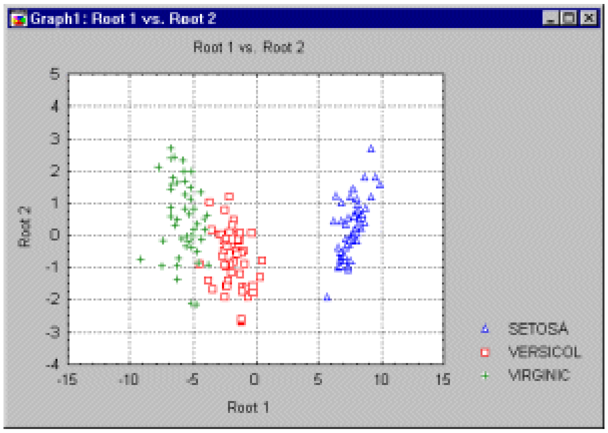
Discriminant Functions for Multiple Groups

When there are more than two groups, then we can estimate more than one discriminant function like the one presented above. For example, when there are three groups, we could estimate (1) a function for discriminating between group 1 and groups 2 and 3 combined, and (2) another function for discriminating between group 2 and group 3. For example, we could have one function that discriminates between those high school graduates that go to college and those who do not (but rather get a job or go to a professional or trade school), and a second function to discriminate between those graduates that go to a professional or trade school versus those who get a job. The *b* coefficients in those discriminant functions could then be interpreted as before.

Canonical analysis. When actually performing a multiple group discriminant analysis, we do not have to specify how to combine groups so as to form different discriminant functions. Rather, you can automatically determine some optimal combination of variables so that the first function provides the most overall discrimination between groups, the second provides second most, and so on. Moreover, the functions will be independent or *orthogonal*, that is, their contributions to the discrimination between groups will not overlap. Computationally, you will perform a *canonical correlation* analysis (see also *Canonical Correlation*) that will determine the successive functions and canonical *roots* (the term root refers to the eigenvalues that are associated with the respective canonical function). The maximum number of functions will be equal to the number of groups minus one, or the number of variables in the analysis, whichever is smaller.

Interpreting the discriminant functions. As before, we will get *b* (and standardized *beta*) coefficients for each variable in each discriminant (now also called *canonical*) function, and they can be interpreted as usual: the larger the standardized coefficient, the greater is the contribution of the respective variable to the discrimination between groups.

(Note that we could also interpret the *structure coefficients*; see below.) However, these coefficients do not tell us between which of the groups the respective functions discriminate. We can identify the nature of the discrimination for each discriminant (canonical) function by looking at the means for the functions across groups. We can also visualize how the two functions discriminate between groups by plotting the individual scores for the two discriminant functions (see the example graph below).



In this example, *Root* (function) *1* seems to discriminate mostly between groups *Setosa*, and *Virginic* and *Versicol* combined. In the vertical direction (*Root 2*), a slight trend of *Versicol* points to fall below the center line (*0*) is apparent.

Factor structure matrix. Another way to determine which variables "mark" or define a particular discriminant function is to look at the factor structure. The factor structure coefficients are the correlations between the variables in the model and the discriminant functions; if you are familiar with factor analysis (see *Factor Analysis*) you may think of these correlations as factor *loadings* of the variables on each discriminant function.

Some authors have argued that these structure coefficients should be used when interpreting the substantive "meaning" of discriminant functions. The reasons given by those authors are that (1) supposedly the structure coefficients are more stable, and (2) they allow for the interpretation of factors (discriminant functions) in the manner that is analogous to factor analysis. However, subsequent Monte Carlo research (Barcikowski & Stevens, 1975; Huberty, 1975) has shown that the discriminant function coefficients and the structure coefficients are about equally unstable, unless the n is fairly large (e.g., if there are 20 times more cases than there are variables). The most important thing to remember is that the discriminant function coefficients denote the unique (partial) contribution of each variable to the discriminant function(s), while the structure coefficients denote the simple correlations between the variables and the function(s). If one wants to assign substantive "meaningful" labels to the discriminant functions (akin to the interpretation of factors in factor analysis), then the structure coefficients should be used (interpreted); if one wants to learn what is each variable's unique contribution to the discriminant function, use the discriminant function coefficients (weights).

Significance of discriminant functions. One can test the number of roots that add *significantly* to the discrimination between group. Only those found to be statistically significant should be used for interpretation; non-significant functions (roots) should be ignored.

Summary. To summarize, when interpreting multiple discriminant functions, which arise from analyses with more than two groups and more than one variable, one would first test the different functions for statistical significance, and only consider the significant functions for further examination. Next, we would look at the standardized *b* coefficients for each variable for each significant function. The larger the standardized *b* coefficient, the larger is the respective variable's unique contribution to the discrimination specified by the respective discriminant function. In order to derive substantive "meaningful" labels for the discriminant functions, one can also examine the factor structure matrix with the correlations between the variables and the discriminant functions. Finally, we would look at the means for the significant discriminant functions in order to determine between which groups the respective functions seem to discriminate.

Assumptions

As mentioned earlier, discriminant function analysis is computationally very similar to MANOVA, and all assumptions for *MANOVA* mentioned in ANOVA/MANOVA apply. In fact, you may use the wide range of diagnostics and statistical tests of assumption that are available to examine your data for the discriminant analysis.

Normal distribution. It is assumed that the data (for the variables) represent a sample from a multivariate normal distribution. You can examine whether or not variables are normally distributed with histograms of frequency distributions. However, note that violations of the normality assumption are usually not "fatal," meaning, that the resultant significance tests etc. are still "trustworthy." You may use specific tests for normality in addition to graphs.

Homogeneity of variances/covariances. It is assumed that the variance/covariance matrices of variables are homogeneous across groups. Again, minor deviations are not that important; however, before accepting final conclusions for an important study it is probably a good idea to review the within-groups variances and correlation matrices. In particular a scatterplot matrix can be produced and can be very useful for this purpose. When in doubt, try re-running the analyses excluding one or two groups that are of less interest. If the overall results (interpretations) hold up, you probably do not have a problem. You may also use the numerous tests available to examine whether or not this assumption is violated in your data. However, as mentioned in *ANOVA/MANOVA*, the multivariate Box *M* test for homogeneity of variances/covariances is particularly sensitive to deviations from multivariate normality, and should not be taken too "seriously."

Correlations between means and variances. The major "real" threat to the validity of significance tests occurs when the means for variables across groups are correlated with the variances (or standard deviations). Intuitively, if there is large variability in a group with particularly high means on some variables, then those high means are not reliable. However, the overall significance tests are based on pooled variances, that is, the average variance across all groups. Thus, the significance tests of the relatively larger means (with the large variances) would be based on the relatively smaller pooled variances, resulting erroneously in statistical significance. In practice, this pattern may occur if one group in the study contains a few extreme outliers, who have a large impact on the means, and also increase the variability. To guard against this problem, inspect the descriptive statistics, that is, the means and standard deviations or variances for such a correlation.

The matrix ill-conditioning problem. Another assumption of discriminant function analysis is that the variables that are used to discriminate between groups are not completely redundant. As part of the computations involved in discriminant analysis, you will invert the variance/covariance matrix of the variables in the model. If any one of the variables is completely redundant with the other variables then the matrix is said to be *ill-conditioned*, and it cannot be inverted. For example, if a variable is the sum of three other variables that are also in the model, then the matrix is ill-conditioned.

Tolerance values. In order to guard against matrix ill-conditioning, constantly check the so-called tolerance value for each variable. This tolerance value is computed as *1 minus R-square* of the respective variable with all other variables included in the current model. Thus, it is the proportion of variance that is unique to the respective variable. You may also refer to Multiple Regression to learn more about multiple regression and the interpretation of the tolerance value. In general, when a variable is almost completely redundant (and, therefore, the matrix ill-conditioning problem is likely to occur), the tolerance value for that variable will approach 0.

Classification

Another major purpose to which discriminant analysis is applied is the issue of predictive classification of cases. Once a model has been finalized and the discriminant functions have been derived, how well can we *predict* to which group a particular case belongs?

***A priori* and *post hoc* predictions.** Before going into the details of different estimation procedures, we would like to make sure that this difference is clear. Obviously, if we estimate, based on some data set, the discriminant functions that best discriminate between groups, and then use the *same* data to evaluate how accurate our prediction is, then we are very much capitalizing on chance. In general, one will *always* get a worse classification when predicting cases that were not used for the estimation of the discriminant function. Put another way, *post hoc* predictions are always better than *a* *priori* predictions. (The trouble with predicting the future *a priori* is that one does not know what will happen; it is much easier to find ways to predict what we already know has happened.) Therefore, one should never base one's confidence regarding the correct classification of future observations on the same data set from which the discriminant functions were derived; rather, if one wants to classify cases predictively, it is necessary to collect new data to "try out" (cross-validate) the utility of the discriminant functions.

Classification functions. These are not to be confused with the discriminant functions. The classification functions can be used to determine to which group each case most likely belongs. There are as many classification functions as there are groups. Each function allows us to compute *classification scores* for each case for each group, by applying the formula:

$$S_i = c_i + w_{i1}*x_1 + w_{i2}*x_2 + ... + w_{im}*x_m$$

In this formula, the subscript i denotes the respective group; the subscripts $1, 2, ..., m$ denote the m variables; c_i is a constant for the i'th group, w_{ij} is the weight for the j'th variable in the computation of the classification score for the i'th group; x_j is the observed value for the respective case for the j'th variable. S_i is the resultant classification score.

We can use the classification functions to directly compute classification scores for some new observations.

Classification of cases. Once we have computed the classification scores for a case, it is easy to decide how to classify the case: in general we classify the case as belonging to the group for which it has the highest classification score (unless the *a priori* classification probabilities are widely disparate; see below). Thus, if we were to study high school students' post-graduation career/educational choices (e.g., attending college, attending a professional or trade school, or getting a job) based on several variables assessed one year prior to graduation, we could use the classification functions to predict what each student is most likely to do after graduation. However, we would also like to know the *probability* that the student will make the predicted choice. Those probabilities are called *posterior* probabilities, and can also be computed. However, to understand how those probabilities are derived, let us first consider the so-called *Mahalanobis* distances.

Mahalanobis distances. You may have read about these distances in other parts of the manual. In general, the Mahalanobis distance is a measure of distance between two points in the space defined by two or more correlated variables. For example, if there are two variables that are uncorrelated, then we could plot points (cases) in a standard two-dimensional scatterplot; the Mahalanobis distances between the points would then be identical to the Euclidean distance; that is, the distance as, for example, measured by a ruler. If there are three uncorrelated variables, we could also simply use a ruler (in a 3-D plot) to determine the distances between points. If there are more than 3 variables, we cannot represent the distances in a plot any more. Also, when the variables are correlated, then the axes in the plots can be thought of as being *non-orthogonal*; that is, they would not be positioned in right angles to each other. In those cases, the simple Euclidean distance is not an appropriate measure, while the Mahalanobis distance will adequately account for the correlations.

Mahalanobis distances and classification. For each group in our sample, we can determine the location of the point that represents the means for all variables in the multivariate space defined by the variables in the model. These points are called group *centroids*. For each case we can then compute the Mahalanobis distances (of the respective case) from each of the group centroids. Again, we would classify the case as belonging to the group to which it is closest, that is, where the Mahalanobis distance is smallest.

Posterior classification probabilities. Using the Mahalanobis distances to do the classification, we can now derive probabilities. The probability that a case belongs to a particular group is basically proportional to the Mahalanobis distance from that group centroid (it is not exactly proportional because we assume a multivariate normal distribution around each centroid). Because we compute the location of each case from our prior knowledge of the values for that case on the variables in the model, these probabilities are called *posterior* probabilities. In summary, the posterior probability is the probability, based on our knowledge of the values of other variables, that the respective case belongs to a particular group. Some software packages will automatically compute those probabilities for all cases (or for selected cases only for cross-validation studies).

A priori classification probabilities. There is one additional factor that needs to be considered when classifying cases. Sometimes, we know ahead of time that there are more observations in one group than in any other; thus, the *a priori* probability that a case belongs to that group is higher. For example, if we know ahead of time that 60% of the graduates from our high school usually go to college (20% go to a professional school, and another 20% get a job), then we should adjust our prediction accordingly: *a priori*, and all other things being equal, it is more likely that a student will attend college that choose either of the other two options. You can specify different *a priori* probabilities, which will then be used to adjust the classification of cases (and the computation of posterior probabilities) accordingly.

In practice, the researcher needs to ask him or herself whether the unequal number of cases in different groups in the sample is a reflection of the true distribution in the population, or whether it is only the (random) result of the sampling procedure. In the former case, we would set the *a priori* probabilities to be proportional to the sizes of the groups in our sample, in the latter case we would specify the *a priori* probabilities as being equal in each group. The specification of different *a priori* probabilities can greatly affect the accuracy of the prediction.

Summary of the prediction. A common result that one looks at in order to determine how well the current classification functions predict group membership of cases is the *classification matrix*. The classification matrix shows the number of cases that were correctly classified (on the diagonal of the matrix) and those that were misclassified.

Another word of caution. To reiterate, *post hoc* predicting of what has happened in the past is not that difficult. It is not uncommon to obtain very good classification if one uses the same cases from which the classification functions were computed. In order to get an idea of how well the current classification functions "perform," one must classify (*a priori*) *different* cases, that is, cases that were not used to estimate the classification functions. You can include or exclude cases from the computations; thus, the classification matrix can be computed for "old" cases as well as "new" cases. Only the classification of new cases allows us to assess the predictive validity of the classification functions (see also cross-validation); the classification of old cases only provides a useful diagnostic tool to identify outliers or areas where the classification function seems to be less adequate.

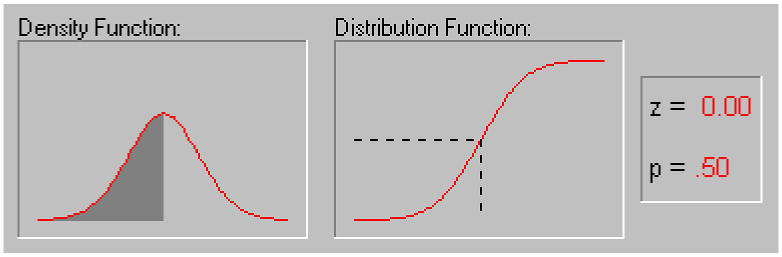
Summary. In general *Discriminant Analysis* is a very useful tool (1) for detecting the variables that allow the researcher to discriminate between different (naturally occurring) groups, and (2) for classifying cases into different groups with a better than chance accuracy.

Chapter 14
Distribution Fitting

- General Purpose
- Fit of the Observed Distribution
- Types of Distributions
    - Bernoulli Distribution
    - Beta Distribution
    - Binomial Distribution
    - Cauchy Distribution
    - Chi-square Distribution
    - Exponential Distribution
    - Extreme Value Distribution
    - F Distribution
    - Gamma Distribution
    - Geometric Distribution
    - Gompertz Distribution
    - Laplace Distribution
    - Logistic Distribution
    - Log-normal Distribution
    - Normal Distribution
    - Pareto Distribution
    - Poisson Distribution
    - Rayleigh Distribution
    - Rectangular Distribution
    - Student's t Distribution
    - Weibull Distribution

General Purpose

In some research applications one can formulate hypotheses about the specific distribution of the variable of interest. For example, variables whose values are determined by an infinite number of independent random events will be distributed following the normal distribution: one can think of a person's height as being the result of very many independent factors such as numerous specific genetic predispositions, early childhood diseases, nutrition, etc. (see the animation below for an example of the normal distribution). As a result, height tends to be normally distributed in the U.S. population. On the other hand, if the values of a variable are the result of very rare events, then the variable will be distributed according to the *Poisson* distribution (sometimes called the distribution of rare events). For example, industrial accidents can be thought of as the result of the intersection of a series of unfortunate (and unlikely) events, and their frequency tends to be distributed according to the Poisson distribution. These and other distributions are described in greater detail in the respective glossary topics.



Another common application where distribution fitting procedures are useful is when one wants to verify the assumption of normality before using some parametric test (see General Purpose of Nonparametric Tests). For example, you may want to use the Kolmogorov-Smirnov test for normality or the Shapiro-Wilks' W test to test for normality.

Fit of the Observed Distribution

For predictive purposes it is often desirable to understand the shape of the underlying distribution of the population. To determine this underlying distribution, it is common to fit the observed distribution to a theoretical distribution by comparing the frequencies observed in the data to the expected frequencies of the theoretical distribution (i.e., a Chi-square goodness of fit test). In addition to this type a test, some software packages also allow you to compute Maximum Likelihood tests and Method of Matching Moments (see Fitting Distributions by Moments in the Process Analysis chapter) tests.

Which Distribution to use. As described above, certain types of variables follow specific distributions. Variables whose values are determined by an infinite number of independent random events will be distributed following the normal distribution, whereas variables whose values are the result of an extremely rare event would follow the Poisson distribution. The major distributions that have been proposed for modeling survival or failure times are the exponential (and linear exponential) distribution, the Weibull distribution of extreme events, and the Gompertz distribution. The section on types of distributions contains a number of distributions generally giving a brief example of what type of data would most commonly follow a specific distribution as well as the probability density functin (pdf) for each distribution.

Types of Distributions

Bernoulli Distribution. This distribution best describes all situations where a "trial" is made resulting in either "success" or "failure," such as when tossing a coin, or when modeling the success or failure of a surgical procedure. The Bernoulli distribution is defined as:

$f(x) = p^x *(1-p)^{1-x}$, for x □ {0,1}

where

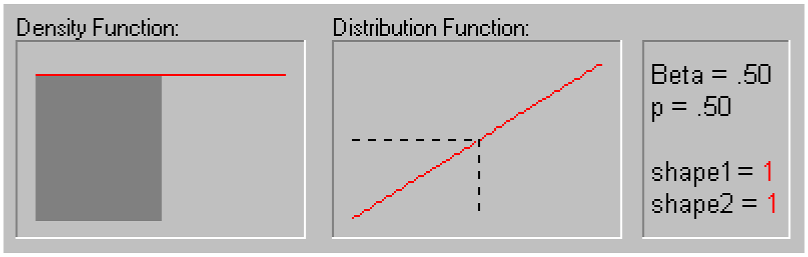
p is the probability that a particular event (e.g., success) will occur.

Beta Distribution. The beta distribution arises from a transformation of the F distribution and is typically used to model the distribution of order statistics. Because the beta distribution is bounded on both sides, it is often used for representing processes with natural lower and upper limits. For examples, refer to Hahn and Shapiro (1967). The beta distribution is defined as:

$f(x) = \square(\square+\square)/[\square(\square)\square(\square)] * x^{\square-1}*(1-x)^{\square-1}$, for $0 < x < 1$, $\square > 0$, $\square > 0$

where

□ is the Gamma function
□, □ are the shape parameters (Shape1 and Shape2, respectively)



The animation above shows the beta distribution as the two shape parameters change.

Binomial Distribution. The binomial distribution is useful for describing distributions of binomial events, such as the number of males and females in a random sample of companies, or the number of defective components in samples of 20 units taken from a production process. The binomial distribution is defined as:

$f(x) = [n!/(x!*(n-x)!)]*p^x * q^{n-x}$, for x = 0,1,2,...,n

where

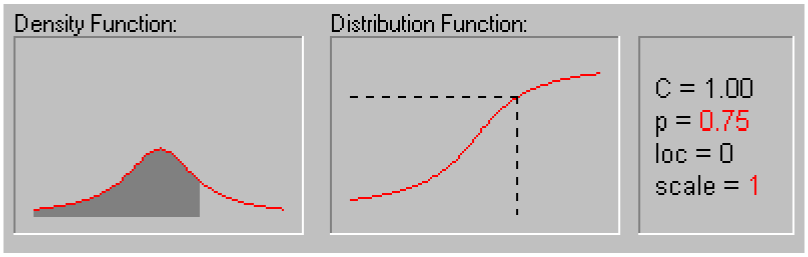
p is the probability that the respective event will occur
q is equal to 1-p
n is the maximum number of independent trials.

Cauchy Distribution. The Cauchy distribution is interesting for theoretical reasons. Although its mean can be taken as zero, since it is symmetrical about zero, the expectation, variance, higher moments, and moment generating function do not exist. The Cauchy distribution is defined as:

$f(x) = 1/(\square*\square*\{1+[(x- \square)/ \square]^2\})$, **for** $0 < \square$

where

□ is the location parameter (median)
□ is the scale parameter
□ is the constant Pi (3.1415...)



Density Function:

Distribution Function:

C = 1.00
p = 0.75
loc = 0
scale = 1

The animation above shows the changing shape of the Cauchy distribution when the location parameter equals 0 and the scale parameter equals 1, 2, 3, and 4.

Chi-square Distribution. The sum of □ independent squared random variables, each distributed following the standard normal distribution, is distributed as Chi-square with □ degrees of freedom. This distribution is most frequently used in the modeling of random variables (e.g., representing frequencies) in statistical applications. The Chi-square distribution is defined by:

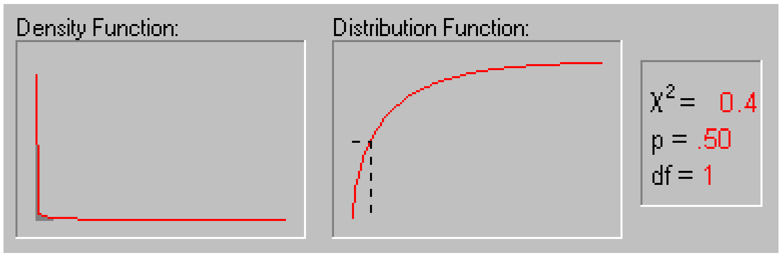
$f(x) = \{1/[2^{\square/2}* \square(\square/2)]\} * [x^{(\square/2)-1} * e^{-x/2}]$, **for** $\square = 1, 2, ..., 0 < x$

where

□ is the degrees of freedom

e is the base of the natural logarithm, sometimes called Euler's e (2.71...)

□ (gamma) is the Gamma function.



The above animation shows the shape of the Chi-square distribution as the degrees of freedom increase (1, 2, 5, 10, 25 and 50).

Exponential Distribution. If *T* is the time between occurrences of rare events that happen on the average with a rate l per unit of time, then *T* is distributed exponentially with parameter □ (lambda). Thus, the exponential distribution is frequently used to model the time interval between successive random events. Examples of variables distributed in this manner would be the gap length between cars crossing an intersection, life-times of electronic devices, or arrivals of customers at the check-out counter in a grocery store. The exponential distribution function is defined as:

$$f(x) = \square * e^{-\square x} \quad \text{for } 0 \ \square \ x < \square,\ \square > 0$$

where

□ is an exponential function parameter (an alternative parameterization is scale parameter b=1/□)

e is the base of the natural logarithm, sometimes called Euler's e (2.71...)

Extreme Value. The extreme value distribution is often used to model extreme events, such as the size of floods, gust velocities encountered by airplanes, maxima of stock marked indices over a given year, etc.; it is also often used in reliability testing, for example in order to represent the distribution of failure times for electric circuits (see Hahn and Shapiro, 1967). The extreme value (Type I) distribution has the probability density function:

$$f(x) = 1/b * e\wedge[-(x-a)/b] * e\wedge\{-e\wedge[-(x-a)/b]\}, \quad \text{for } -\square < x < \square,\ b > 0$$

where

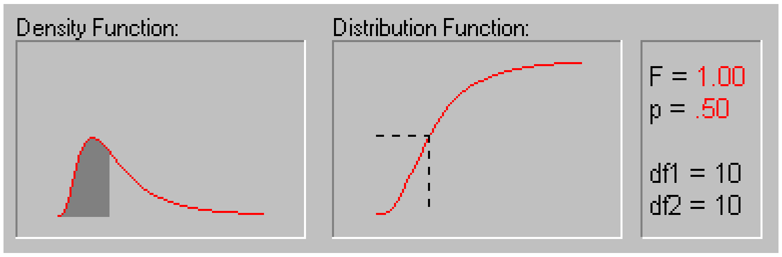
a is the location parameter
b is the scale parameter
e is the base of the natural logarithm, sometimes called Euler's e (2.71...)

F Distribution. Snedecor's F distribution is most commonly used in tests of variance (e.g., ANOVA). The ratio of two chi-squares divided by their respective degrees of freedom is said to follow an F distribution. The F distribution (for x > 0) has the probability density function (for □ = 1, 2, ...; □ = 1, 2, ...):

$$f(x) = [\square\{(\square+\square)/2\}]/[\square(\square/2)\square(\square/2)] * (\square/\square)^{(\square/2)} * x^{[(\square/2)-1]} * \{1+[(\square/\square)*x]\}^{[-(\square+\square)/2]},$$
$$\text{for } 0 \ \square \ x < \square \quad \square=1,2,\ldots, \ \square=1,2,\ldots$$

where

□, □ are the shape parameters, degrees of freedom
□ is the Gamma function



F = 1.00
p = .50

df1 = 10
df2 = 10

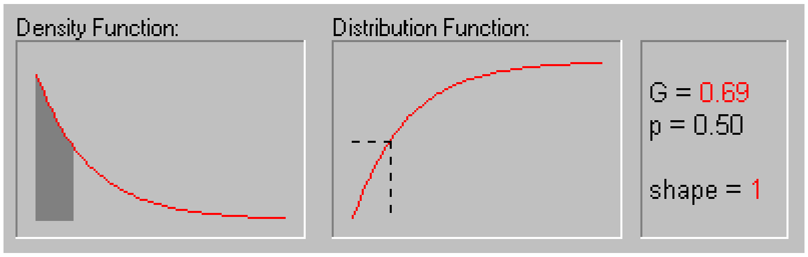
The animation above shows various tail areas (p-values) for an F distribution with both degrees of freedom equal to 10.

Gamma Distribution. The probability density function of the exponential distribution has a mode of zero. In many instances, it is known *a priori* that the mode of the distribution of a particular random variable of interest is not equal to zero (e.g., when modeling the distribution of the life-times of a product such as an electric light bulb, or the serving time taken at a ticket booth at a baseball game). In those cases, the gamma distribution is more appropriate for describing the underlying distribution. The gamma distribution is defined as:

$$f(x) = \{1/[b\square(c)]\}*[x/b]^{c-1}*e^{-x/b} \quad \text{for } 0 \ \square \ x, c > 0$$

where

□ is the Gamma function
c is the Shape parameter
b is the Scale parameter.
e is the base of the natural logarithm, sometimes called Euler's e (2.71...)



The animation above shows the gamma distribution as the shape parameter changes from 1 to 6.

Geometric Distribution. If independent Bernoulli trials are made until a "success" occurs, then the total number of trials required is a geometric random variable. The geometric distribution is defined as:

$f(x) = p*(1-p)^x$, for $x = 1,2,...$

where

p is the probability that a particular event (e.g., success) will occur.

Gompertz Distribution. The Gompertz distribution is a theoretical distribution of survival times. Gompertz (1825) proposed a probability model for human mortality, based on the assumption that the "average exhaustion of a man's power to avoid death to be such that at the end of equal infinetely small intervals of time he lost equal portions of his remaining power to oppose destruction which he had at the commencement of these intervals" (Johnson, Kotz, Blakrishnan, 1995, p. 25). The resultant hazard function:

$r(x)=Bc^x$, for x □ $0, B > 0, c$ □ 1

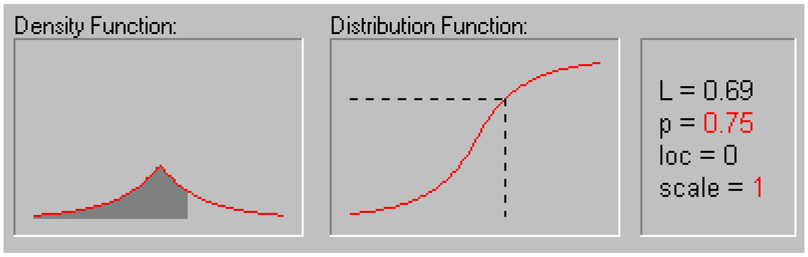
is often used in survival analysis. See Johnson, Kotz, Blakrishnan (1995) for additional details.

Laplace Distribution. For interesting mathematical applications of the Laplace distribution see Johnson and Kotz (1995). The Laplace (or Double Exponential) distribution is defined as:

$f(x) = 1/(2b) * e^{[-(|x-a|/b)]}$, for $-\square < x < \square$

where

a is the location parameter (mean)
b is the scale parameter
e is the base of the natural logarithm, sometimes called Euler's e (2.71...)



L = 0.69
p = 0.75
loc = 0
scale = 1

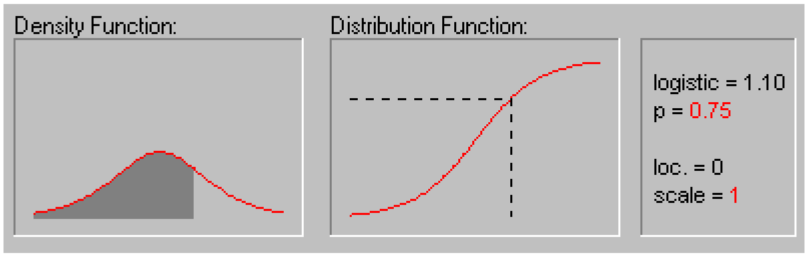
The graphic above shows the changing shape of the Laplace distribution when the location parameter equals 0 and the scale parameter equals 1, 2, 3, and 4.

Logistic Distribution. The logistic distribution is used to model binary responses (e.g., Gender) and is commonly used in logistic regression. The logistic distribution is defined as:

$f(x) = (1/b) * e^{[-(x-a)/b]} * \{1+e^{[-(x-a)/b]}\}^{\wedge-2}$, for $-\square < x < \square$, $0 < b$

where

a is the location parameter (mean)
b is the scale parameter
e is the base of the natural logarithm, sometimes called Euler's e (2.71...)



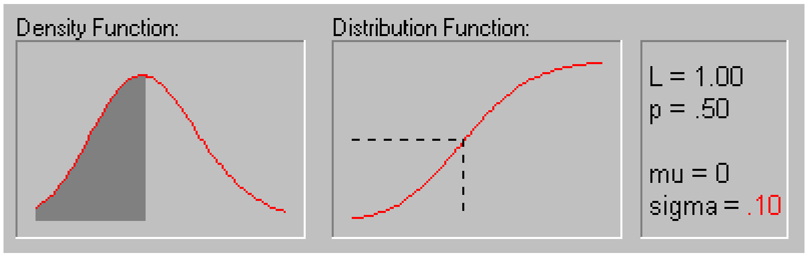
The graphic above shows the changing shape of the logistic distribution when the location parameter equals 0 and the scale parameter equals 1, 2, and 3.

Log-normal Distribution. The log-normal distribution is often used in simulations of variables such as personal incomes, age at first marriage, or tolerance to poison in animals. In general, if x is a sample from a normal distribution, then $y = e^x$ is a sample from a log-normal distribution. Thus, the log-normal distribution is defined as:

$$f(x) = 1/[x□(2)^{1/2}] * e^{-[\log(x)-□]**2/2□**2}, \quad \text{for } 0 < x < □,\ □ > 0,\ □ > 0$$

where

□ is the scale parameter
□ is the shape parameter
e is the base of the natural logarithm, sometimes called Euler's e (2.71...)



The animation above shows the log-normal distribution with mu equal to 0 and sigma equals .10, .30, .50, .70, and .90.

Normal Distribution. The normal distribution (the "bell-shaped curve" which is symmetrical about the mean) is a theoretical function commonly used in inferential statistics as an approximation to sampling distributions (see also Elementary Concepts). In general, the normal distribution provides a good model for a random variable, when:

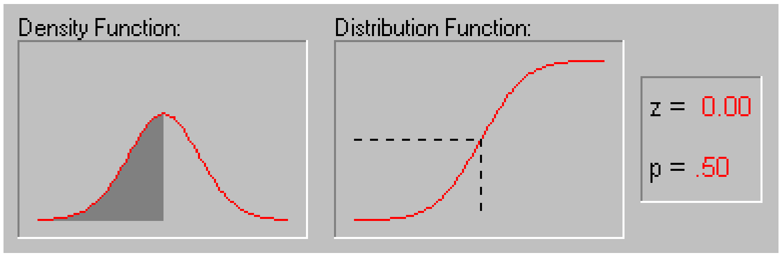
1. There is a strong tendency for the variable to take a central value;
2. Positive and negative deviations from this central value are equally likely;
3. The frequency of deviations falls off rapidly as the deviations become larger.

As an underlying mechanism that produces the normal distribution, one may think of an infinite number of independent random (binomial) events that bring about the values of a particular variable. For example, there are probably a nearly infinite number of factors that determine a person's height (thousands of genes, nutrition, diseases, etc.). Thus, height can be expected to be normally distributed in the population. The normal distribution function is determined by the following formula:

$f(x) = 1/[(2*\square)^{1/2}*\square] * e**\{-1/2*[(x-\square)/\square]^{2}\}$, for $-\square < x < \square$

where

□	is the mean
□	is the standard deviation
e	is the base of the natural logarithm, sometimes called Euler's e (2.71...)
□	is the constant Pi (3.14...)



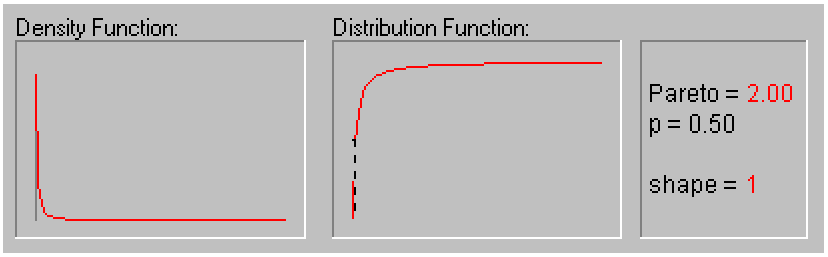
The animation above shows several tail areas of the standard normal distribution (i.e., the normal distribution with a mean of 0 and a standard deviation of 1). The standard normal distribution is often used in hypothesis testing.

Pareto Distribution. The Pareto distribution is commonly used in monitoring production processes (see Quality Control and Process Analysis). For example, a machine which produces copper wire will occasionally generate a flaw at some point along the wire. The Pareto distribution can be used to model the length of wire between successive flaws. The standard Pareto distribution is defined as:

$f(x) = c/x^{c+1}$, for $1 \square x, c < 0$

where

c is the shape parameter



The animation above shows the Pareto distribution for the shape parameter equal to 1, 2, 3, 4, and 5.

Poisson Distribution. The Poisson distribution is also sometimes referred to as the distribution of rare events. Examples of Poisson distributed variables are number of accidents per person, number of sweepstakes won per person, or the number of catastrophic defects found in a production process. It is defined as:

$$f(x) = (\square^{x}*e^{-\square})/x!, \quad \text{for } x = 0,1,2,..., 0 < \square$$

where

□ (lambda) is the expected value of x (the mean)

e is the base of the natural logarithm, sometimes called Euler's e (2.71...)

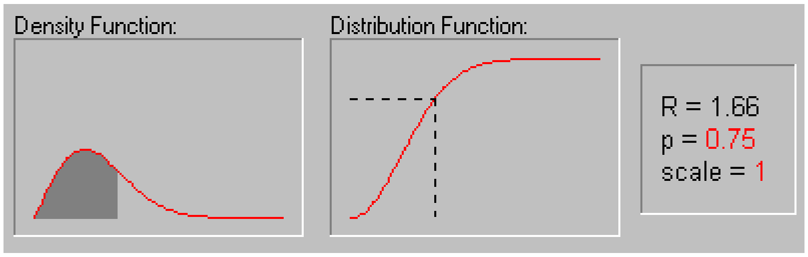
Rayleigh Distribution. If two independent variables y_1 and y_2 are independent from each other and normally distributed with equal variance, then the variable $x = \square(y_1^2+ y_2^2)$ will follow the Rayleigh distribution. Thus, an example (and appropriate metaphor) for such a variable would be the distance of darts from the target in a dart-throwing game, where the errors in the two dimensions of the target plane are independent and normally distributed. The Rayleigh distribution is defined as:

$$f(x) = x/b^2 * e\text{^}[-(x^2/2b^2)], \quad \text{for } 0 \square x < \square, b > 0$$

where

b is the scale parameter

e is the base of the natural logarithm, sometimes called Euler's e (2.71...)



R = 1.66
p = 0.75
scale = 1

The graphic above shows the changing shape of the Rayleigh distribution when the scale parameter equals 1, 2, and 3.

Rectangular Distribution. The rectangular distribution is useful for describing random variables with a constant probability density over the defined range a<b.

f(x) = 1/(b-a), for a<x<b
= 0 , elsewhere

where

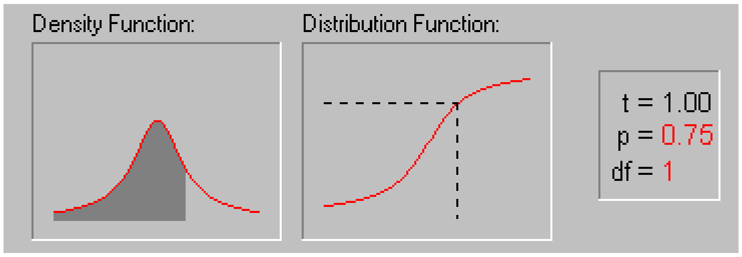
a<b are constants.

Student's t Distribution. The student's t distribution is symmetric about zero, and its general shape is similar to that of the standard normal distribution. It is most commonly used in testing hypothesis about the mean of a particular population. The student's t distribution is defined as (for n = 1, 2, . . .):

$$f(x) = \square[(\square+1)/2] \,/\, \square(\square/2) * (\square*\square)^{-1/2} * [1 + (x^2/\square)^{-(\square+1)/2}$$

where

□ is the shape parameter, degrees of freedom
□ is the Gamma function
□ is the constant Pi (3.14 . . .)



t = 1.00
p = 0.75
df = 1

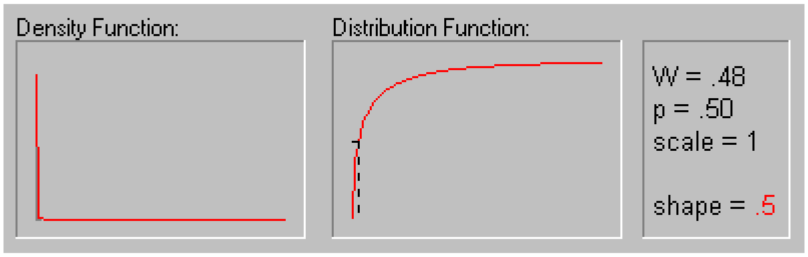
The shape of the student's t distribution is determined by the degrees of freedom. As shown in the animation above, its shape changes as the degrees of freedom increase.

Weibull Distribution. As described earlier, the exponential distribution is often used as a model of time-to-failure measurements, when the failure (hazard) rate is constant over time. When the failure probability varies over time, then the Weibull distribution is appropriate. Thus, the Weibull distribution is often used in reliability testing (e.g., of electronic relays, ball bearings, etc.; see Hahn and Shapiro, 1967). The Weibull distribution is defined as:

$f(x) = c/b*(x/b)^{(c-1)} * e^{[-(x/b)\hat{}c]}$, for $0 \square x < \square$, $b > 0$, $c > 0$

where

b is the scale parameter
c is the shape parameter
e is the base of the natural logarithm, sometimes called Euler's e (2.71...)



W = .48
p = .50
scale = 1
shape = .5

The animation above shows the Weibull distribution as the shape parameter increases (.5, 1, 2, 3, 4, 5, and 10).

Chapter 15
Experimental Design (Industrial DOE)

- DOE Overview
    - Experiments in Science and Industry
    - Differences in techniques
    - Overview
    - General Ideas
    - Computational Problems
    - Components of Variance, Denominator Synthesis
    - Summary
- 2**(k-p) Fractional Factorial Designs
    - Basic Idea
    - Generating the Design
    - The Concept of Design Resolution
    - Plackett-Burman (Hadamard Matrix) Designs for Screening
    - Enhancing Design Resolution via Foldover
    - Aliases of Interactions: Design Generators
    - Blocking
    - Replicating the Design
    - Adding Center Points
    - Analyzing the Results of a 2**(k-p) Experiment
    - Graph Options
    - Summary
- 2**(k-p) Maximally Unconfounded and Minimum Aberration Designs
    - Basic Idea
    - Design Criteria
    - Summary
- 3**(k-p) , Box-Behnken, and Mixed 2 and 3 Level Factorial Designs
    - Overview
    - Designing 3**(k-p) Experiments
    - An Example 3**(4-1) Design in 9 Blocks
    - Box-Behnken Designs
    - Analyzing the 3**(k-p) Design
    - ANOVA Parameter Estimates
    - Graphical Presentation of Results
    - Designs for Factors at 2 and 3 Levels
- Central Composite and Non-Factorial Response Surface Designs
    - Overview
    - Design Considerations
    - Alpha for Rotatability and Orthogonality
    - Available Standard Designs
    - Analyzing Central Composite Designs
    - The Fitted Response Surface

  - Categorized Response Surfaces
- Latin Square Designs
  - Overview
  - Latin Square Designs
  - Analyzing the Design
  - Very Large Designs, Random Effects, Unbalanced Nesting
- Taguchi Methods: Robust Design Experiments
  - Overview
  - Quality and Loss Functions
  - Signal-to-Noise (S/N) Ratios
  - Orthogonal Arrays
  - Analyzing Designs
  - Accumulation Analysis
  - Summary
- Mixture designs and triangular surfaces
  - Overview
  - Triangular Coordinates
  - Triangular Surfaces and Contours
  - The Canonical Form of Mixture Polynomials
  - Common Models for Mixture Data
  - Standard Designs for Mixture Experiments
  - Lower Constraints
  - Upper and Lower Constraints
  - Analyzing Mixture Experiments
  - Analysis of Variance
  - Parameter Estimates
  - Pseudo-Components
  - Graph Options
- Designs for constrained surfaces and mixtures
  - Overview
  - Designs for Constrained Experimental Regions
  - Linear Constraints
  - The Piepel & Snee Algorithm
  - Choosing Points for the Experiment
  - Analyzing Designs for Constrained Surfaces and Mixtures
- Constructing D- and A-optimal designs
  - Overview
  - Basic Ideas
  - Measuring Design Efficiency
  - Constructing Optimal Designs
  - General Recommendations
  - Avoiding Matrix Singularity
  - "Repairing" Designs
  - Constrained Experimental Regions and Optimal Design
- Special Topics
  - Profiling Predicted Responses and Response Desirability

- Residuals Analysis
- Box-Cox Transformations of Dependent Variables

DOE Overview

Experiments in Science and Industry

Experimental methods are widely used in research as well as in industrial settings, however, sometimes for very different purposes. The primary goal in scientific research is usually to show the statistical significance of an effect that a particular factor exerts on the dependent variable of interest (for details concerning the concept of statistical significance see *Elementary Concepts*).

In industrial settings, the primary goal is usually to extract the maximum amount of unbiased information regarding the factors affecting a production process from as few (costly) observations as possible. While in the former application (in science) analysis of variance (*ANOVA*) techniques are used to uncover the interactive nature of reality, as manifested in higher-order interactions of factors, in industrial settings interaction effects are often regarded as a "nuisance" (they are often of no interest; they only complicate the process of identifying important factors).

Differences in techniques

These differences in purpose have a profound effect on the techniques that are used in the two settings. If you review a standard ANOVA text for the sciences, for example the classic texts by Winer (1962) or Keppel (1982), you will find that they will primarily discuss designs with up to, perhaps, five factors (designs with more than six factors are usually impractical; see the ANOVA/MANOVA chapter). The focus of these discussions is how to derive valid and robust statistical significance tests. However, if you review standard texts on experimentation in industry (Box, Hunter, and Hunter, 1978; Box and Draper, 1987; Mason, Gunst, and Hess, 1989; Taguchi, 1987) you will find that they will primarily discuss designs with many factors (e.g., 16 or 32) in which interaction effects cannot be evaluated, and the primary focus of the discussion is how to derive unbiased main effect (and, perhaps, two-way interaction) estimates with a minimum number of observations.

This comparison can be expanded further, however, a more detailed description of experimental design in industry will now be discussed and other differences will become clear. Note that the *General Linear Models* and *ANOVA/MANOVA* chapters contain detailed discussions of typical design issues in scientific research; the General Linear Model procedure is a very comprehensive implementation of the general linear model approach to ANOVA/MANOVA (univariate and multivariate ANOVA). There are of course applications in industry where general ANOVA designs, as used in scientific research, can be immensely useful. You may want to read the *General Linear Models* and

ANOVA/MANOVA chapters to gain a more general appreciation of the range of methods encompassed by the term Experimental Design.

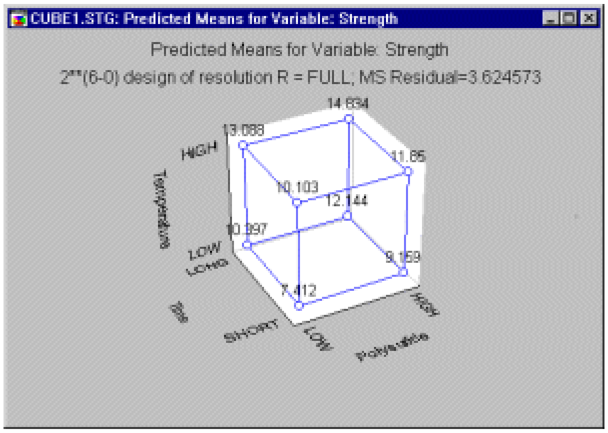
Overview

The general ideas and principles on which experimentation in industry is based, and the types of designs used will be discussed in the following paragraphs. The following paragraphs are meant to be introductory in nature. However, it is assumed that you are familiar with the basic ideas of analysis of variance and the interpretation of main effects and interactions in ANOVA. Otherwise, it is strongly recommend that you read the *Introductory Overview* section for *ANOVA/MANOVA* and the *General Linear Models* chapter.

General Ideas

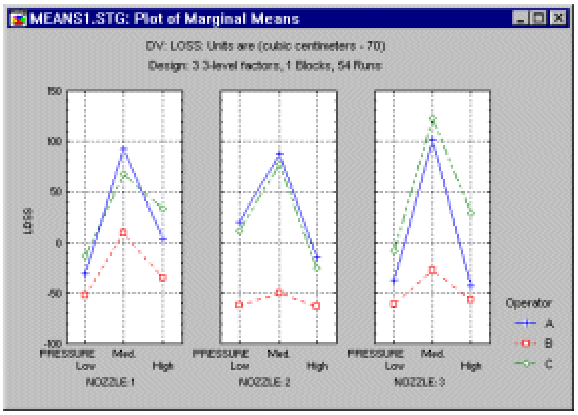
In general, every machine used in a production process allows its operators to adjust various settings, affecting the resultant quality of the product manufactured by the machine. Experimentation allows the production engineer to adjust the settings of the machine in a *systematic* manner and to learn which factors have the greatest impact on the resultant quality. Using this information, the settings can be constantly improved until optimum quality is obtained. To illustrate this reasoning, here are a few examples:

Example 1: Dyestuff manufacture. Box and Draper (1987, page 115) report an experiment concerned with the manufacture of certain dyestuff. *Quality* in this context can be described in terms of a desired (specified) hue and brightness and maximum fabric strength. Moreover, it is important to know what to change in order to produce a different hue and brightness should the consumers' taste change. Put another way, the experimenter would like to identify the factors that affect the brightness, hue, and strength of the final product. In the example described by Box and Draper, there are 6 different factors that are evaluated in a 2**(6-0) design (the *2**(k-p)* notation is explained below). The results of the experiment show that the three most important factors determining fabric strength are the *Polysulfide index*, *Time*, and *Temperature* (see Box and Draper, 1987, page 116). One can summarize the expected effect (predicted means) for the variable of interest (i.e., fabric strength in this case) in a so- called cube-plot. This plot shows the expected (predicted) mean fabric strength for the respective low and high settings for each of the three variables (factors).



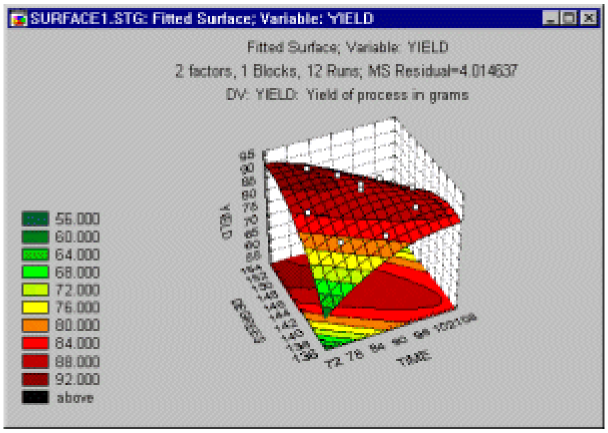
Example 1.1: Screening designs. In the previous example, 6 different factors were simultaneously evaluated. It is not uncommon, that there are very many (e.g., 100) different factors that may potentially be important. Special designs (e.g., Plackett-Burman designs, see Plackett and Burman, 1946) have been developed to screen such large numbers of factors in an efficient manner, that is, with the least number of observations necessary. For example, you can design and analyze an experiment with 127 factors and only 128 runs (observations); still, you will be able to estimate the main effects for each factor, and thus, you can quickly identify which ones are important and most likely to yield improvements in the process under study.

Example 2: 33 design.** Montgomery (1976, page 204) describes an experiment conducted in order identify the factors that contribute to the loss of soft drink syrup due to frothing during the filling of five- gallon metal containers. Three factors where considered: (a) the nozzle configuration, (b) the operator of the machine, and (c) the operating pressure. Each factor was set at three different levels, resulting in a complete 3**(3-0) experimental design (the 3**(k-p) notation is explained below).



Moreover, two measurements were taken for each combination of factor settings, that is, the 3**(3-0) design was completely replicated once.

Example 3: Maximizing yield of a chemical reaction. The yield of many chemical reactions is a function of time and temperature. Unfortunately, these two variables often do not affect the resultant yield in a linear fashion. In other words, it is not so that "the longer the time, the greater the yield" and "the higher the temperature, the greater the yield." Rather, both of these variables are usually related in a *curvilinear* fashion to the resultant yield.

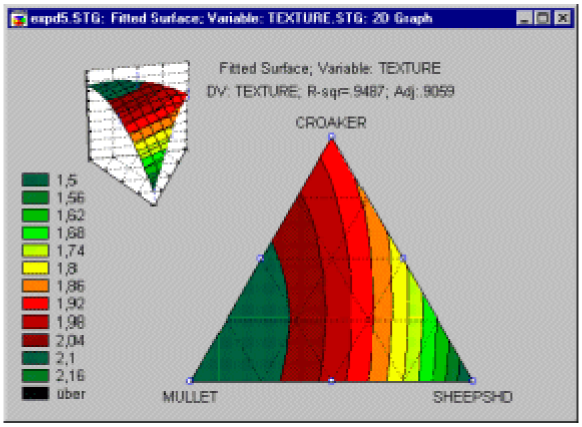


Thus, in this example your goal as experimenter would be to *optimize* the yield *surface* that is created by the two variables: *time* and *temperature*.

Example 4: Testing the effectiveness of four fuel additives. Latin square designs are useful when the factors of interest are measured at more than two levels, and the nature of the problem suggests some *blocking*. For example, imagine a study of 4 fuel additives on the reduction in oxides of nitrogen (see Box, Hunter, and Hunter, 1978, page 263). You may have 4 drivers and 4 cars at your disposal. You are not particularly interested in any effects of particular cars or drivers on the resultant oxide reduction; however, you do not want the results for the fuel additives to be biased by the particular driver or car. Latin square designs allow you to estimate the main effects of all factors in the design in an unbiased manner. With regard to the example, the arrangement of treatment levels in a Latin square design assures that the variability among drivers or cars does not affect the estimation of the effect due to different fuel additives.

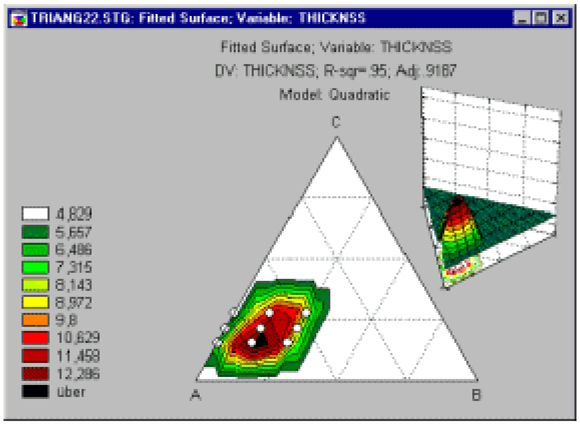
Example 5: Improving surface uniformity in the manufacture of polysilicon wafers. The manufacture of reliable microprocessors requires very high consistency in the manufacturing process. Note that in this instance, it is equally, if not more important to control the *variability* of certain product characteristics than it is to control the average for a characteristic. For example, with regard to the average surface thickness of the polysilicon layer, the manufacturing process may be perfectly under control; yet, if the variability of the surface thickness on a wafer fluctuates widely, the resultant microchips will not be reliable. Phadke (1989) describes how different characteristics of the manufacturing process (such as deposition temperature, deposition pressure, nitrogen flow, etc.) affect the variability of the polysilicon surface thickness on wafers. However, no theoretical model exists that would allow the engineer to *predict* how these factors affect the uniformness of wafers. Therefore, systematic experimentation with the factors is required to optimize the process. This is a typical example where *Taguchi robust design* methods would be applied.

Example 6: Mixture designs. Cornell (1990, page 9) reports an example of a typical (simple) mixture problem. Specifically, a study was conducted to determine the optimum texture of fish patties as a result of the relative proportions of different types of fish (Mullet, Sheepshead, and Croaker) that made up the patties. Unlike in non-mixture experiments, the total sum of the proportions must be equal to a constant, for example, to 100%. The results of such experiments are usually graphically represented in so-called triangular (or ternary) graphs.



In general, the overall constraint -- that the three components must sum to a constant -- is reflected in the triangular shape of the graph (see above).

Example 6.1: Constrained mixture designs. It is particularly common in mixture designs that the relative amounts of components are further constrained (in addition to the constraint that they must sum to, for example, 100%). For example, suppose we wanted to design the best-tasting fruit *punch* consisting of a mixture of juices from five fruits. Since the resulting mixture is supposed to be a fruit punch, pure blends consisting of the pure juice of only one fruit are necessarily excluded. Additional constraints may be placed on the "universe" of mixtures due to cost constraints or other considerations, so that one particular fruit cannot, for example, account for more than 30% of the mixtures (otherwise the fruit punch would be too expensive, the shelf-life would be compromised, the punch could not be produced in large enough quantities, etc.). Such so-called constrained experimental regions present numerous problems, which, however, can be addressed.


TRIANG22.STG: Fitted Surface; Variable: THICKNSS
Fitted Surface; Variable: THICKNSS
DV: THICKNSS; R-sqr=.95; Adj:.9187
Model: Quadratic
C
A
B
4,829
5,657
6,486
7,315
8,143
8,972
9,8
10,629
11,458
12,286
über

In general, under those conditions, one seeks to design an experiment that can potentially extract the maximum amount of information about the respective response function (e.g., taste of the fruit punch) in the experimental region of interest.

Computational Problems

There are basically two general issues to which *Experimental Design* is addressed:

1. How to design an optimal experiment, and
2. How to analyze the results of an experiment.

With regard to the first question, there are different considerations that enter into the different types of designs, and they will be discussed shortly. In the most general terms, the goal is always to allow the experimenter to evaluate in an unbiased (or least biased) way, the consequences of changing the settings of a particular factor, that is, regardless of how other factors were set. In more technical terms, you attempt to generate designs where main effects are unconfounded among themselves, and in some cases, even unconfounded with the interaction of factors.

Components of Variance, Denominator Synthesis

There are several statistical methods for analyzing designs with random effects (see Methods for Analysis of Variance). The Variance Components and Mixed Model ANOVA/ANCOVA chapter discusses numerous options for estimating variance components for random effects, and for performing approximate F tests based on synthesized error terms.

Summary

Experimental methods are finding increasing use in manufacturing to optimize the production process. Specifically, the goal of these methods is to identify the optimum settings for the different factors that affect the production process. In the discussion so far, the major classes of designs that are typically used in industrial experimentation have been introduced: 2**(k-p) (two-level, multi-factor) designs, screening designs for large numbers of factors, 3**(k-p) (three-level, multi-factor) designs (mixed designs with 2 and 3 level factors are also supported), *central composite (or response surface)* designs, *Latin square* designs, *Taguchi robust design* analysis, *mixture* designs, and special procedures for constructing experiments in constrained experimental regions. Interestingly, many of these experimental techniques have "made their way" from the production plant into management, and successful implementations have been reported in profit planning in business, cash-flow optimization in banking, etc. (e.g., see Yokyama and Taguchi, 1975).

These techniques will now be described in greater detail in the following sections:

1. 2**(k-p) Fractional Factorial Designs
2. 2**(k-p) Maximally Unconfounded and Minimum Aberration Designs
3. 3**(k-p) , Box-Behnken, and Mixed 2 and 3 Level Factorial Designs
4. Central Composite and Non-Factorial Response Surface Designs
5. Latin Square Designs
6. Taguchi Methods: Robust Design Experiments
7. Mixture designs and triangular surfaces
8. Designs for constrained surfaces and mixtures
9. Constructing D- and A-optimal designs for surfaces and mixtures

2**(k-p) Fractional Factorial Designs at 2 Levels

Basic Idea

In many cases, it is sufficient to consider the factors affecting the production process at two levels. For example, the temperature for a chemical process may either be set a little higher or a little lower, the amount of solvent in a dyestuff manufacturing process can either be slightly increased or decreased, etc. The experimenter would like to determine whether any of these changes affect the results of the production process. The most intuitive approach to study those factors would be to vary the factors of interest in a full factorial design, that is, to try all possible combinations of settings. This would work fine, except that the number of necessary runs in the experiment (observations) will increase geometrically. For example, if you want to study 7 factors, the necessary number of runs in the experiment would be 2**7 = 128. To study 10 factors you would need 2**10 = 1,024 runs in the experiment. Because each run may require time-consuming and costly setting and resetting of machinery, it is often not feasible to require that many different production runs for the experiment. In these conditions, *fractional factorials* are used that "sacrifice" interaction effects so that main effects may still be computed correctly.

Generating the Design

A technical description of how fractional factorial designs are constructed is beyond the scope of this introduction. Detailed accounts of how to design 2**(k-p) experiments can be found, for example, in Bayne and Rubin (1986), Box and Draper (1987), Box, Hunter, and Hunter (1978), Montgomery (1991), Daniel (1976), Deming and Morgan (1993), Mason, Gunst, and Hess (1989), or Ryan (1989), to name only a few of the many text books on this subject. In general, it will successively "use" the highest-order interactions to generate new factors. For example, consider the following design that includes 11 factors but requires only 16 runs (observations).

Design: 2(11-7), Resolution III**

Run	A	B	C	D	E	F	G	H	I	J	K
1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	-1	1	-1	-1	-1	-1	1	1
3	1	1	-1	1	-1	-1	-1	1	-1	1	-1
4	1	1	-1	-1	-1	1	1	-1	1	1	-1
5	1	-1	1	1	-1	-1	1	-1	-1	-1	1
6	1	-1	1	-1	-1	1	-1	1	1	-1	1
7	1	-1	-1	1	1	1	-1	-1	1	-1	-1
8	1	-1	-1	-1	1	-1	1	1	-1	-1	-1
9	-1	1	1	1	-1	1	-1	-1	-1	-1	-1
10	-1	1	1	-1	-1	-1	1	1	1	-1	-1
11	-1	1	-1	1	1	-1	1	-1	1	-1	1
12	-1	1	-1	-1	1	1	-1	1	-1	-1	1
13	-1	-1	1	1	1	-1	-1	1	1	1	-1
14	-1	-1	1	-1	1	1	1	-1	-1	1	-1
15	-1	-1	-1	1	-1	1	1	1	-1	1	1
16	-1	-1	-1	-1	-1	-1	-1	-1	1	1	1

Reading the design. The design displayed above should be interpreted as follows. Each column contains +1's or -1's to indicate the setting of the respective factor (high or low, respectively). So for example, in the first run of the experiment, set all factors *A* through *K* to the plus setting (e.g., a little higher than before); in the second run, set factors *A*, *B*, and *C* to the positive setting, factor *D* to the negative setting, and so on. Note that there are numerous options provided to display (and save) the design using notation other than *±1* to denote factor settings. For example, you may use actual values of factors (e.g., *90 degrees* Celsius and *100 degrees* Celsius) or text labels (*Low* temperature, *High* temperature).

Randomizing the runs. Because many other things may change from production run to production run, it is always a good practice to randomize the order in which the systematic runs of the designs are performed.

The Concept of Design Resolution

The design above is described as a 2**(11-7) design of *resolution* III (three). This means that you study overall *k = 11* factors (the first number in parentheses); however, *p = 7* of those factors (the second number in parentheses) were generated from the interactions of a full 2**[(11-7) = 4] factorial design. As a result, the design does not give full *resolution*; that is, there are certain interaction effects that are confounded with (identical to) other effects. In general, a design of resolution *R* is one where no *l*-way interactions are confounded with any other interaction of order less than *R-l*. In the current example, *R* is equal to 3. Here, no *l* = 1 level interactions (i.e., main effects) are confounded with any other interaction of order less than *R-l* = 3-1 = 2. Thus, main effects in this design are confounded with two- way interactions; and consequently, all higher-order interactions are equally confounded. If you had included 64 runs, and generated a 2**(11-5) design, the resultant resolution would have been *R* = *IV* (four). You would have concluded that no *l*=1-way interaction (main effect) is confounded with any other interaction of order less than *R-l* = 4-1 = 3. In this design then, main effects are not confounded with two-way interactions, but only with three-way interactions. What about the two-way interactions? No *l*=2-way interaction is confounded with any other interaction of order less than *R-l* = 4-2 = 2. Thus, the two-way interactions in that design are confounded with each other.

Plackett-Burman (Hadamard Matrix) Designs for Screening

When one needs to screen a large number of factors to identify those that may be important (i.e., those that are related to the dependent variable of interest), one would like to employ a design that allows one to test the largest number of factor main effects with the least number of observations, that is to construct a resolution III design with as few runs as possible. One way to design such experiments is to confound all interactions with "new" main effects. Such designs are also sometimes called *saturated* designs, because all information in those designs is used to estimate the parameters, leaving no degrees of freedom to estimate the error term for the ANOVA. Because the added factors are created by equating (aliasing, see below), the "new" factors with the interactions of a full factorial design, these designs always will have 2**k runs (e.g., 4, 8, 16, 32, and so on). Plackett and Burman (1946) showed how full factorial design can be fractionalized in a different manner, to yield saturated designs where the number of runs is a multiple of 4, rather than a power of 2. These designs are also sometimes called Hadamard matrix designs. Of course, you do not have to use all available factors in those designs, and, in fact, sometimes you want to generate a saturated design for one more factor than you are expecting to test. This will allow you to estimate the random error variability, and test for the statistical significance of the parameter estimates.

Enhancing Design Resolution via Foldover

One way in which a resolution III design can be enhanced and turned into a resolution IV design is via *foldover* (e.g., see Box and Draper, 1987, Deming and Morgan, 1993): Suppose you have a 7-factor design in 8 runs:

Design: 2(7-4) design**							
Run	**A**	**B**	**C**	**D**	**E**	**F**	**G**
1	1	1	1	1	1	1	1

2	1	1	-1	1	-1	-1	-1
3	1	-1	1	-1	1	-1	-1
4	1	-1	-1	-1	-1	1	1
5	-1	1	1	-1	-1	1	-1
6	-1	1	-1	-1	1	-1	1
7	-1	-1	1	1	-1	-1	1
8	-1	-1	-1	1	1	1	-1

This is a resolution III design, that is, the two-way interactions will be confounded with the main effects. You can turn this design into a resolution IV design via the Foldover (enhance resolution) option. The foldover method copies the entire design and appends it to the end, reversing all signs:

Design: 2(7-4) design (+Foldover)**								
Run	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**New: H**
1	1	1	1	1	1	1	1	1
2	1	1	-1	1	-1	-1	-1	1
3	1	-1	1	-1	1	-1	-1	1
4	1	-1	-1	-1	-1	1	1	1
5	-1	1	1	-1	-1	1	-1	1
6	-1	1	-1	-1	1	-1	1	1
7	-1	-1	1	1	-1	-1	1	1
8	-1	-1	-1	1	1	1	-1	1
9	-1	-1	-1	-1	-1	-1	-1	-1
10	-1	-1	1	-1	1	1	1	-1
11	-1	1	-1	1	-1	1	1	-1
12	-1	1	1	1	1	-1	-1	-1
13	1	-1	-1	1	1	-1	1	-1
14	1	-1	1	1	-1	1	-1	-1
15	1	1	-1	-1	1	1	-1	-1
16	1	1	1	-1	-1	-1	1	-1

Thus, the standard run number 1 was *-1, -1, -1, 1, 1, 1, -1*; the new run number 9 (the first run of the "folded-over" portion) has all signs reversed: *1, 1, 1, -1, -1, -1, 1*. In addition to enhancing the resolution of the design, we also have gained an 8'th factor (factor *H*), which contains all +1's for the first eight runs, and -1's for the folded-over portion of the new design. Note that the resultant design is actually a 2**(8-4) design of resolution IV (see also Box and Draper, 1987, page 160).

Aliases of Interactions: Design Generators

To return to the example of the resolution *R* = *III* design, now that you know that main effects are confounded with two-way interactions, you may ask the question, "Which interaction is confounded with which main effect?"

	Fractional Design Generators
	2(11-7) design**
	(Factors are denoted by numbers)
Factor	**Alias**
5	123
6	234
7	134
8	124
9	1234
10	12
11	13

Design generators. The design generators shown above are the "key" to how factors *5* through *11* were generated by assigning them to particular interactions of the first 4 factors of the full factorial 2**4 design. Specifically, factor *5* is identical to the *123* (factor 1 by factor 2 by factor 3) interaction. Factor *6* is identical to the *234* interaction, and so on. Remember that the design is of resolution III (three), and you expect some main effects to be confounded with some two-way interactions; indeed, factor *10* (ten) is identical to the *12* (factor 1 by factor 2) interaction, and factor *11* (eleven) is identical to the *13* (factor 1 by factor 3) interaction. Another way in which these equivalencies are often expressed is by saying that the main effect for factor *10* (ten) is an *alias* for the interaction of *1* by 2. (The term *alias* was first used by Finney, 1945).

To summarize, whenever you want to include fewer observations (runs) in your experiment than would be required by the full factorial 2**k design, you "sacrifice" interaction effects and assign them to the levels of factors. The resulting design is no longer a full factorial but a *fractional* factorial.

The fundamental identity. Another way to summarize the design generators is in a simple equation. Namely, if, for example, factor *5* in a fractional factorial design is identical to the *123* (factor *1* by factor *2* by factor *3*) interaction, then it follows that multiplying the coded values for the *123* interaction by the coded values for factor *5* will always result in +*1* (if all factor levels are coded ±*1*); or:

I = 1235

where *I* stands for +*1* (using the standard notation as, for example, found in Box and Draper, 1987). Thus, we also know that factor *1* is confounded with the *235* interaction, factor *2* with the *135*, interaction, and factor *3* with the *125* interaction, because, in each instance their product must be equal to *1*. The confounding of two-way interactions is also defined by this equation, because the *12* interaction multiplied by the *35* interaction must yield *1*, and hence, they are identical or confounded. Therefore, one can summarize all confounding in a design with such a *fundamental identity* equation.

Blocking

In some production processes, units are produced in natural "chunks" or blocks. You want to make sure that these blocks do not bias your estimates of main effects. For example, you may have a kiln to produce special ceramics, but the size of the kiln is limited so that you cannot produce all runs of your experiment at once. In that case you need to break up the experiment into blocks. However, you do not want to run positive settings of all factors in one block, and all negative settings in the other. Otherwise, any incidental differences between blocks would systematically affect all estimates of the main effects of the factors of interest. Rather, you want to distribute the runs over the blocks so that any differences between blocks (i.e., the blocking *factor*) do not bias your results for the factor effects of interest. This is accomplished by treating the blocking factor as another factor in the design. Consequently, you "lose" another interaction effect to the blocking factor, and the resultant design will be of lower resolution. However, these designs often have the advantage of being statistically more powerful, because they allow you to estimate and control the variability in the production process that is due to differences between blocks.

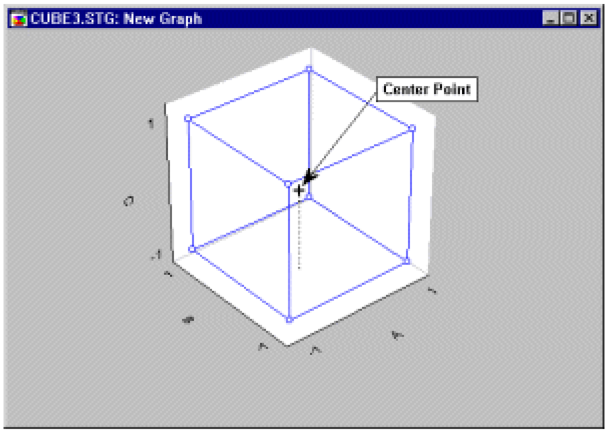
Replicating the Design

It is sometimes desirable to replicate the design, that is, to run each combination of factor levels in the design more than once. This will allow you to later estimate the so-called *pure error* in the experiment. The analysis of experiments is further discussed below; however, it should be clear that, when replicating the design, one can compute the variability of measurements within each unique combination of factor levels. This variability will give an indication of the random error in the measurements (e.g., due to uncontrolled factors, unreliability of the measurement instrument, etc.), because the replicated observations are taken under identical conditions (settings of factor levels). Such an estimate of the pure error can be used to evaluate the size and statistical significance of the variability that can be attributed to the manipulated factors.

Partial replications. When it is not possible or feasible to replicate each unique combination of factor levels (i.e., the full design), one can still gain an estimate of pure error by replicating only some of the runs in the experiment. However, one must be careful to consider the possible bias that may be introduced by selectively replicating only some runs. If one only replicates those runs that are most easily repeated (e.g., gathers information at the points where it is "cheapest"), one may inadvertently only choose those combinations of factor levels that happen to produce very little (or very much) random variability -- causing one to underestimate (or overestimate) the true amount of pure error. Thus, one should carefully consider, typically based on your knowledge about the process that is being studied, which runs should be replicated, that is, which runs will yield a good (unbiased) estimate of pure error.

Adding Center Points

Designs with factors that are set at two levels implicitly assume that the effect of the factors on the dependent variable of interest (e.g., fabric *Strength*) is linear. It is impossible to test whether or not there is a non-linear (e.g., quadratic) component in the relationship between a factor *A* and a dependent variable, if *A* is only evaluated at two points (.i.e., at the *low* and *high* settings). If one suspects that the relationship between the factors in the design and the dependent variable is rather curve-linear, then one should include one or more runs where all (continuous) factors are set at their midpoint. Such runs are called center-point runs (or center points), since they are, in a sense, in the center of the design (see graph).



Later in the analysis (see below), one can compare the measurements for the dependent variable at the center point with the average for the rest of the design. This provides a check for *curvature* (see Box and Draper, 1987): If the mean for the dependent variable at the center of the design is significantly different from the overall mean at all other points of the design, then one has good reason to believe that the simple assumption that the factors are linearly related to the dependent variable, does not hold.

Analyzing the Results of a 2**(k-p) Experiment

Analysis of variance. Next, one needs to determine exactly which of the factors significantly affected the dependent variable of interest. For example, in the study reported by Box and Draper (1987, page 115), it is desired to learn which of the factors involved in the manufacture of dyestuffs affected the strength of the fabric. In this example, factors 1 (*Polysulfide*), 4 (*Time*), and 6 (*Temperature*) significantly affected the strength of the fabric. Note that to simplify matters, only main effects are shown below.

ANOVA; Var.:STRENGTH; R-sqr = .60614; Adj:.56469 (fabrico.sta)					
	2**(6-0) design; MS Residual = 3.62509 DV: STRENGTH				
	SS	df	MS	F	p

(1)POLYSUFD	48.8252	1	48.8252	13.46867	.000536
(2)REFLUX	7.9102	1	7.9102	2.18206	.145132
(3)MOLES	.1702	1	.1702	.04694	.829252
(4)TIME	142.5039	1	142.5039	39.31044	.000000
(5)SOLVENT	2.7639	1	2.7639	.76244	.386230
(6)TEMPERTR	115.8314	1	115.8314	31.95269	.000001
Error	206.6302	57	3.6251		
Total SS	524.6348	63			

Pure error and lack of fit. If the experimental design is at least partially replicated, then one can estimate the error variability for the experiment from the variability of the replicated runs. Since those measurements were taken under identical conditions, that is, at identical settings of the factor levels, the estimate of the error variability from those runs is independent of whether or not the "true" model is linear or non-linear in nature, or includes higher-order interactions. The error variability so estimated represents *pure error*, that is, it is entirely due to unreliabilities in the measurement of the dependent variable. If available, one can use the estimate of pure error to test the significance of the residual variance, that is, all remaining variability that cannot be accounted for by the factors and their interactions that are currently in the model. If, in fact, the residual variability is significantly larger than the pure error variability, then one can conclude that there is still some statistically significant variability left that is attributable to differences between the groups, and hence, that there is an overall *lack* of fit of the current model.

ANOVA; Var.:STRENGTH; R-sqr = .58547; Adj:.56475 (fabrico.sta)					
	2(3-0) design; MS Pure Error = 3.594844 DV: STRENGTH**				
	SS	**df**	**MS**	**F**	**p**
(1)POLYSUFD	48.8252	1	48.8252	13.58200	.000517
(2)TIME	142.5039	1	142.5039	39.64120	.000000
(3)TEMPERTR	115.8314	1	115.8314	32.22154	.000001
Lack of Fit	16.1631	4	4.0408	1.12405	.354464
Pure Error	201.3113	56	3.5948		
Total SS	524.6348	63			

For example, the table above shows the results for the three factors that were previously identified as most important in their effect on fabric strength; all other factors where ignored in the analysis. As you can see in the row with the label *Lack of Fit*, when the residual variability for this model (i.e., after removing the three main effects) is compared against the pure error estimated from the within-group variability, the resulting *F* test is not statistically significant. Therefore, this result additionally supports the conclusion that, indeed, factors *Polysulfide, Time*, and *Temperature* significantly affected resultant fabric strength in an additive manner (i.e., there are no interactions). Or, put another way, all differences between the means obtained in the different experimental conditions can be sufficiently explained by the simple additive model for those three variables.

Parameter or effect estimates. Now, look at *how* these factors affected the strength of the fabrics.

	Effect	Std.Err.	t (57)	p
Mean/Interc.	11.12344	.237996	46.73794	.000000
(1)POLYSUFD	1.74688	.475992	3.66997	.000536
(2)REFLUX	.70313	.475992	1.47718	.145132
(3)MOLES	.10313	.475992	.21665	.829252
(4)TIME	2.98438	.475992	6.26980	.000000
(5)SOLVENT	-.41562	.475992	-.87318	.386230
(6)TEMPERTR	2.69062	.475992	5.65267	.000001

The numbers above are the effect or parameter estimates. With the exception of the overall *Mean/Intercept*, these estimates are the *deviations* of the mean of the negative settings from the mean of the positive settings for the respective factor. For example, if you change the setting of factor *Time* from *low* to *high*, then you can expect an improvement in *Strength* by *2.98*; if you set the value for factor *Polysulfd* to its high setting, you can expect a further improvement by *1.75*, and so on.

As you can see, the same three factors that were statistically significant show the largest parameter estimates; thus the settings of these three factors were most important for the resultant strength of the fabric.

For analyses including interactions, the interpretation of the effect parameters is a bit more complicated. Specifically, the two-way interaction parameters are defined as half the difference between the main effects of one factor at the two levels of a second factor (see Mason, Gunst, and Hess, 1989, page 127); likewise, the three-way interaction parameters are defined as half the difference between the two-factor interaction effects at the two levels of a third factor, and so on.

Regression coefficients. One can also look at the parameters in the multiple regression model (see *Multiple Regression*). To continue this example, consider the following prediction equation:

$$\text{Strength} = \text{const} + b_1 *x_1 + \ldots + b_6 *x_6$$

Here x_1 through x_6 stand for the 6 factors in the analysis. The Effect Estimates shown earlier also contains these parameter estimates:

	Coeff.	Std.Err. Coeff.	-95.% Cnf.Limt	+95.% Cnf.Limt
Mean/Interc.	11.12344	.237996	10.64686	11.60002
(1)POLYSUFD	.87344	.237996	.39686	1.35002
(2)REFLUX	.35156	.237996	-.12502	.82814

(3)MOLES	.05156	.237996	-.42502	.52814
(4)TIME	1.49219	.237996	1.01561	1.96877
(5)SOLVENT	-.20781	.237996	-.68439	.26877
(6)TEMPERTR	1.34531	.237996	.86873	1.82189

Actually, these parameters contain little "new" information, as they simply are one-half of the parameter values (except for the *Mean/Intercept*) shown earlier. This makes sense since now, the coefficient can be interpreted as the deviation of the high-setting for the respective factors from the center. However, note that this is only the case if the factor values (i.e., their levels) are coded as *-1* and *+1*, respectively. Otherwise, the scaling of the factor values will affect the magnitude of the parameter estimates. In the example data reported by Box and Draper (1987, page 115), the settings or values for the different factors were recorded on very different scales:

data file: FABRICO.STA [64 cases with 9 variables]
2(6-0) Design, Box & Draper, p. 117**

	POLYSUFD	**REFLUX**	**MOLES**	**TIME**	**SOLVENT**	**TEMPERTR**	**STRENGTH**	**HUE**	**BRIGTHNS**
1	6	150	1.8	24	30	120	3.4	15.0	36.0
2	7	150	1.8	24	30	120	9.7	5.0	35.0
3	6	170	1.8	24	30	120	7.4	23.0	37.0
4	7	170	1.8	24	30	120	10.6	8.0	34.0
5	6	150	2.4	24	30	120	6.5	20.0	30.0
6	7	150	2.4	24	30	120	7.9	9.0	32.0
7	6	170	2.4	24	30	120	10.3	13.0	28.0
8	7	170	2.4	24	30	120	9.5	5.0	38.0
9	6	150	1.8	36	30	120	14.3	23.0	40.0
10	7	150	1.8	36	30	120	10.5	1.0	32.0
11	6	170	1.8	36	30	120	7.8	11.0	32.0
12	7	170	1.8	36	30	120	17.2	5.0	28.0
13	6	150	2.4	36	30	120	9.4	15.0	34.0
14	7	150	2.4	36	30	120	12.1	8.0	26.0
15	6	170	2.4	36	30	120	9.5	15.0	30.0
...	...	...	...	...	...	...	...	...	...

Shown below are the regression coefficient estimates based on the uncoded original factor values:

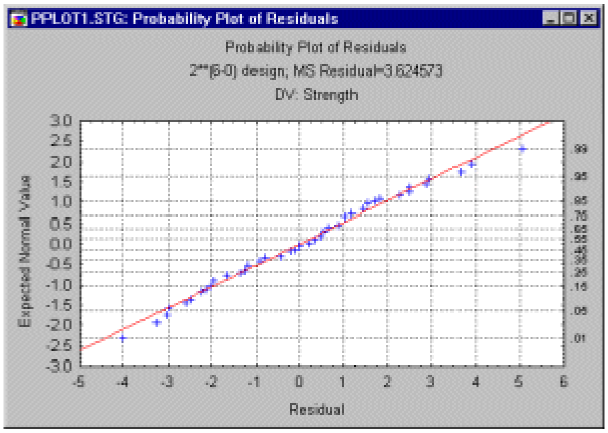
	Regressn Coeff.	**Std.Err.**	**t (57)**	**p**

Mean/Interc.	-46.0641	8.109341	-5.68037	.000000
(1)POLYSUFD	1.7469	.475992	3.66997	.000536
(2)REFLUX	.0352	.023800	1.47718	.145132
(3)MOLES	.1719	.793320	.21665	.829252
(4)TIME	.2487	.039666	6.26980	.000000
(5)SOLVENT	-.0346	.039666	-.87318	.386230
(6)TEMPERTR	.2691	.047599	5.65267	.000001

Because the metric for the different factors is no longer compatible, the magnitudes of the regression coefficients are not compatible either. This is why it is usually more informative to look at the ANOVA parameter estimates (for the coded values of the factor levels), as shown before. However, the regression coefficients can be useful when one wants to make predictions for the dependent variable, based on the original metric of the factors.

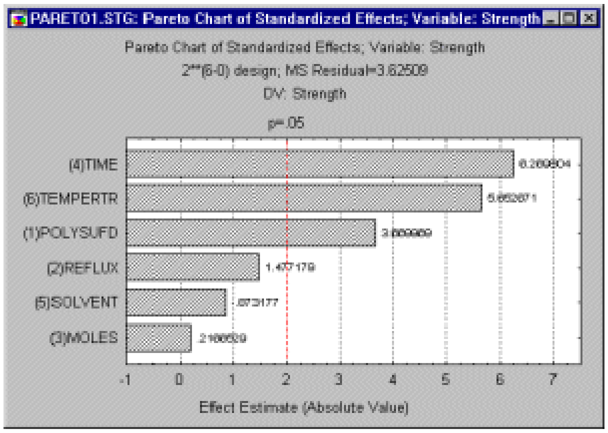
Graph Options

Diagnostic plots of residuals. To start with, before accepting a particular "model" that includes a particular number of effects (e.g., main effects for *Polysulfide, Time,* and *Temperature* in the current example), one should always examine the distribution of the residual values. These are computed as the difference between the predicted values (as predicted by the current model) and the observed values. You can compute the histogram for these residual values, as well as probability plots (as shown below).



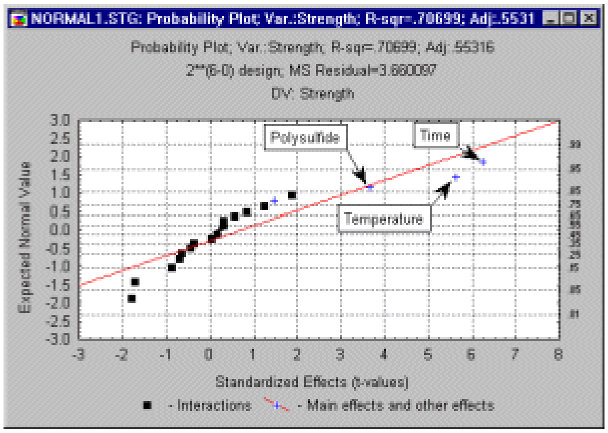
The parameter estimates and ANOVA table are based on the assumption that the residuals are normally distributed (see also [Elementary Concepts]). The histogram provides one way to check (visually) whether this assumption holds. The so-called *normal probability* plot is another common tool to assess how closely a set of observed values (residuals in this case) follows a theoretical distribution. In this plot the actual residual values are plotted along the horizontal *X*-axis; the vertical *Y*-axis shows the expected normal values for the respective values, after they were rank-ordered. If all values fall onto a straight line, then one can be satisfied that the residuals follow the normal distribution.

Pareto chart of effects. The Pareto chart of effects is often an effective tool for communicating the results of an experiment, in particular to laymen.

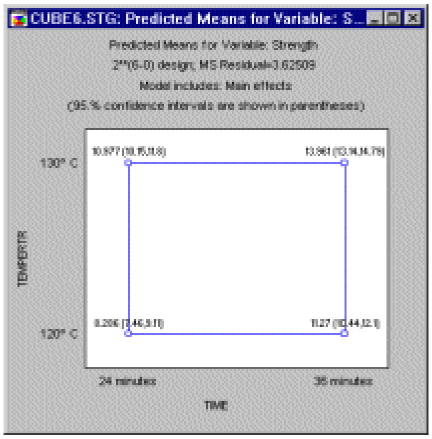


In this graph, the ANOVA effect estimates are sorted from the largest absolute value to the smallest absolute value. The magnitude of each effect is represented by a column, and often, a line going across the columns indicates how large an effect has to be (i.e., how long a column must be) to be statistically significant.

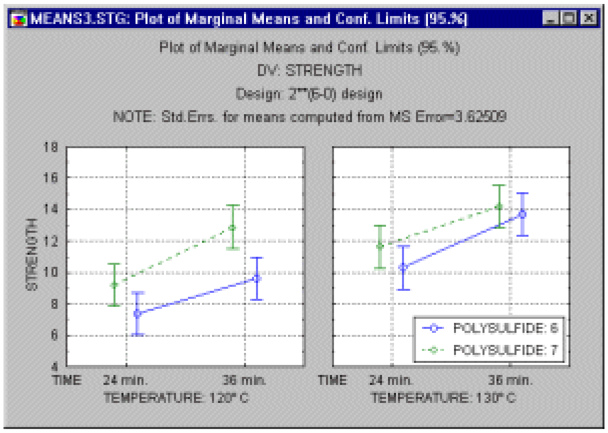
Normal probability plot of effects. Another useful, albeit more technical summary graph, is the *normal probability* plot of the estimates. As in the normal probability plot of the residuals, first the effect estimates are rank ordered, and then a normal *z* score is computed based on the assumption that the estimates are normally distributed. This *z* score is plotted on the *Y*-axis; the observed estimates are plotted on the *X*-axis (as shown below).



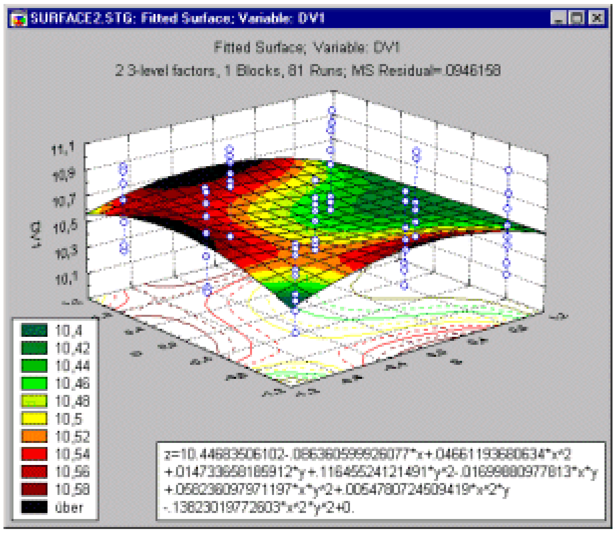
Square and cube plots. These plots are often used to summarize predicted values for the dependent variable, given the respective high and low setting of the factors. The square plot (see below) will show the predicted values (and, optionally, their confidence intervals) for two factors at a time. The cube plot will show the predicted values (and, optionally, confidence intervals) for three factors at a time.


CUBE6.STG: Predicted Means for Variable: S...
Predicted Means for Variable: Strength
2**(6-0) design; MS Residual=3.62509
Model includes: Main effects
(95.% confidence intervals are shown in parentheses)
10.977 (10.15,11.8)
13.961 (13.14,14.78)
130° C
TEMPERTR
120° C
8.206 (7.46,9.11)
11.27 (10.44,12.1)
24 minutes
36 minutes
TIME

Interaction plots. A general graph for showing the means is the standard interaction plot, where the means are indicated by points connected by lines. This plot (see below) is particularly useful when there are significant interaction effects in the model.



Surface and contour plots. When the factors in the design are continuous in nature, it is often also useful to look at surface and contour plots of the dependent variable as a function of the factors.


SURFACE2.STG: Fitted Surface; Variable: DV1

Fitted Surface; Variable: DV1

2 3-level factors, 1 Blocks, 81 Runs; MS Residual=.0946158

```
z=10.44683506102-.086360599926077*x+.04661193680634*x^2
+.014733658185912*y+.11645524121491*y^2-.016998809778813*x*y
+.056236097971197*x*y^2+.0054780724509419*x^2*y
-.13823019772603*x^2*y^2+0.
```

These types of plots will further be discussed later in this section, in the context of 3**(k-p), and central composite and response surface designs.

Summary

2**(k-p) designs are the "workhorse" of industrial experiments. The impact of a large number of factors on the production process can simultaneously be assessed with relative efficiency (i.e., with few experimental runs). The logic of these types of experiments is straightforward (each factor has only two settings).

Disadvantages. The simplicity of these designs is also their major flaw. As mentioned before, underlying the use of two-level factors is the belief that the resultant changes in the dependent variable (e.g., fabric strength) are basically *linear* in nature. This is often not the case, and many variables are related to quality characteristics in a non-linear fashion. In the example above, if you were to continuously increase the temperature factor (which was significantly related to fabric strength), you would of course eventually hit a "peak," and from there on the fabric strength would decrease as the temperature increases. While this types of *curvature* in the relationship between the factors in the design and the dependent variable can be detected if the design included center point runs, one cannot fit explicit nonlinear (e.g., quadratic) models with 2**(k-p) designs (however, central composite designs will do exactly that).

Another problem of fractional designs is the implicit assumption that higher-order interactions do not matter; but sometimes they do, for example, when some other factors are set to a particular level, temperature may be *negatively* related to fabric strength. Again, in fractional factorial designs, higher-order interactions (greater than two-way) particularly will escape detection.

2**(k-p) Maximally Unconfounded and Minimum Aberration Designs

Basic Idea

*2***(*k*-*p*) fractional factorial designs are often used in industrial experimentation because of the economy of data collection that they provide. For example, suppose an engineer needed to investigate the effects of varying 11 factors, each with 2 levels, on a manufacturing process. Let us call the number of factors *k*, which would be 11 for this example. An experiment using a full factorial design, where the effects of every combination of levels of each factor are studied, would require *2***(*k*) experimental runs, or 2048 runs for this example. To minimize the data collection effort, the engineer might decide to forego investigation of higher-order interaction effects of the 11 factors, and focus instead on identifying the main effects of the 11 factors and any low-order interaction effects that could be estimated from an experiment using a smaller, more reasonable number of experimental runs. There is another, more theoretical reason for not conducting huge, full factorial 2 level experiments. In general, it is not logical to be concerned with identifying higher-order interaction effects of the experimental factors, while ignoring lower-order nonlinear effects, such as quadratic or cubic effects, which cannot be estimated if only 2 levels of each factor are employed. So althrough practical considerations often lead to the need to design experiments with a reasonably small number of experimental runs, there is a logical justification for such experiments.

The alternative to the *2***(*k*) full factorial design is the *2***(*k*-*p*) fractional factorial design, which requires only a "fraction" of the data collection effort required for full factorial designs. For our example with *k*=11 factors, if only 64 experimental runs can be conducted, a 2**(11-5) fractional factorial experiment would be designed with 2**6 = 64 experimental runs. In essence, a *k*-*p* = 6 way full factorial experiment is designed, with the levels of the *p* factors being "generated" by the levels of selected higher order interactions of the other 6 factors. Fractional factorials "sacrifice" higher order interaction effects so that lower order effects may still be computed correctly. However, different criteria can be used in choosing the higher order interactions to be used as generators, with different criteria sometimes leading to different "best" designs.

*2***(*k*-*p*) fractional factorial designs can also include blocking factors. In some production processes, units are produced in natural "chunks" or blocks. To make sure that these blocks do not bias your estimates of the effects for the *k* factors, blocking factors can be added as additional factors in the design. Consequently, you may "sacrifice" additional interaction effects to generate the blocking factors, but these designs often have the advantage of being statistically more powerful, because they allow you to estimate and control the variability in the production process that is due to differences between blocks.

Design Criteria

Many of the concepts discussed in this overview are also addressed in the *Overview of 2**(k-p) Fractional factorial designs*. However, a technical description of how fractional factorial designs are constructed is beyond the scope of either introductory overview. Detailed accounts of how to design *2***(*k*-*p*) experiments can be found, for example, in Bayne and Rubin (1986), Box and Draper (1987), Box, Hunter, and Hunter (1978), Montgomery (1991), Daniel (1976), Deming and Morgan (1993), Mason, Gunst, and Hess (1989), or Ryan (1989), to name only a few of the many text books on this subject.

In general, the *2**(k-p) maximally unconfounded and minimum aberration designs* techniques will successively select which higher-order interactions to use as generators for the *p* factors. For example, consider the following design that includes 11 factors but requires only 16 runs (observations).

Design: 2**(11-7), Resolution III											
Run	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**
1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	-1	1	-1	-1	-1	-1	1	1
3	1	1	-1	1	-1	-1	-1	1	-1	1	-1
4	1	1	-1	-1	-1	1	1	-1	1	1	-1
5	1	-1	1	1	-1	-1	1	-1	-1	-1	1
6	1	-1	1	-1	-1	1	-1	1	1	-1	1
7	1	-1	-1	1	1	1	-1	-1	1	-1	-1
8	1	-1	-1	-1	1	-1	1	1	-1	-1	-1
9	-1	1	1	1	-1	1	-1	-1	-1	-1	-1
10	-1	1	1	-1	-1	-1	1	1	1	-1	-1
11	-1	1	-1	1	1	-1	1	-1	1	-1	1
12	-1	1	-1	-1	1	1	-1	1	-1	-1	1
13	-1	-1	1	1	1	-1	-1	1	1	1	-1
14	-1	-1	1	-1	1	1	1	-1	-1	1	-1
15	-1	-1	-1	1	-1	1	1	1	-1	1	1
16	-1	-1	-1	-1	-1	-1	-1	-1	1	1	1

Interpreting the design. The design displayed in the Scrollsheet above should be interpreted as follows. Each column contains +1's or -1's to indicate the setting of the respective factor (high or low, respectively). So for example, in the first run of the experiment, all factors *A* through *K* are set to the higher level, and in the second run, factors *A*, *B*, and *C* are set to the higher level, but factor *D* is set to the lower level, and so on. Notice that the settings for each experimental run for factor *E* can be produced by multiplying the respective settings for factors *A*, *B*, and *C*. The *A x B x C* interaction effect therefore cannot be estimated independently of the factor *E* effect in this design because these two effects are confounded. Likewise, the settings for factor *F* can be produced by multiplying the respective settings for factors *B*, *C*, and *D*. We say that *ABC* and *BCD* are the generators for factors *E* and *F*, respectively.

The maximum resolution design criterion. In the Scrollsheet shown above, the design is described as a 2**(11-7) design of *resolution* III (three). This means that you study overall *k = 11* factors, but *p = 7* of those factors were generated from the interactions of a full 2**[(11-7) = 4] factorial design. As a result, the design does not give full *resolution*; that is, there are certain interaction effects that are confounded with (identical to) other effects. In general, a design of resolution *R* is one where no *l*-way interactions are confounded with any other interaction of order less than *R - l*. In the current example, *R* is equal to 3. Here, no *l* = 1-way interactions (i.e., main effects) are confounded with any other interaction of order less than *R - l* = 3 -1 = 2. Thus, main effects in this design are unconfounded with each other, but are confounded with two-factor interactions; and consequently, with other higher-order interactions. One obvious, but nevertheless very important overall design criterion is that the higher-order interactions to be used as generators should be chosen such that the resolution of the design is as high as possible.

The maximum unconfounding design criterion. Maximizing the resolution of a design, however, does not by itself ensure that the selected generators produce the "best" design. Consider, for example, two different resolution IV designs. In both designs, main effects would be unconfounded with each other and 2-factor interactions would be unconfounded with main effects, i.e, no *l* = 2-way interactions are confounded with any other interaction of order less than *R - l* = 4 - 2 = 2. The two designs might be different, however, with regard to the degree of confounding for the 2-factor interactions. For resolution IV designs, the "crucial order," in which confounding of effects first appears, is for 2-factor interactions. In one design, none of the "crucial order," 2-factor interactions might be unconfounded with all other 2-factor interactions, while in the other design, virtually all of the 2-factor interactions might be unconfounded with all of the other 2-factor interactions. The second "almost resolution V" design would be preferable to the first "just barely resolution IV" design. This suggests that even though the maximum resolution design criterion should be the primary criterion, a subsidiary criterion might be that generators should be chosen such that the maximum number of interactions of less than or equal to the crucial order, given the resolution, are unconfounded with all other interactions of the crucial order. This is called the maximum unconfounding design criterion, and is one of the optional, subsidiary design criterion to use in a search for a *2**(k-p)* design.

The minimum aberration design criterion. The miniminum aberration design criterion is another optional, subsidiary criterion to use in a search for a *2**(k-p)* design. In some respects, this criterion is similar to the maximum unconfounding design criterion. Technically, the minimum aberration design is defined as the design of maximum resolution "which minimizes the number of words in the defining relation that are of minimum length" (Fries & Hunter, 1980). Less technically, the criterion apparently operates by choosing generators that produce the smallest number of pairs of confounded interactions of the crucial order. For example, the minimum aberration resolution IV design would have the minimum number of pairs of confounded 2-factor interactions.

To illustrate the difference between the maximum unconfounding and minimum aberration criteria, consider the maximally unconfounded 2**(9-4) design and the minimum aberration 2**(9-4) design, as for example, listed in Box, Hunter, and Hunter (1978). If you compare these two designs, you will find that in the maximally unconfounded design, 15 of the 36 2-factor interactions are unconfounded with any other 2-factor interactions, while in the minimum aberration design, only 8 of the 36 2-factor interactions are unconfounded with any other 2-factor interactions. The minimum aberration design, however, produces 18 pairs of confounded interactions, while the maximally unconfounded design produces 21 pairs of confounded interactions. So, the two criteria lead to the selection of generators producing different "best" designs.

Fortunately, the choice of whether to use the maximum unconfounding criterion or the minimum aberration criterion makes no difference in the design which is selected (except for, perhaps, relabeling of the factors) when there are 11 or fewer factors, with the single exception of the 2**(9-4) design described above (see Chen, Sun, & Wu, 1993). For designs with more than 11 factors, the two criteria can lead to the selection of very different designs, and for lack of better advice, we suggest using both criteria, comparing the designs that are produced, and choosing the design that best suits your needs. We will add, editorially, that maximizing the number of totally unconfounded effects often makes more sense than minimizing the number of pairs of confounded effects.

Summary

*2**(k-p)* fractional factorial designs are probably the most frequently used type of design in industrial experimentation. Things to consider in designing any *2***(*k-p*) fractional factorial experiment include the number of factors to be investigated, the number of experimental runs, and whether there will be blocks of experimental runs. Beyond these basic considerations, one should also take into account whether the number of runs will allow a design of the required resolution and degree of confounding for the crucial order of interactions, given the resolution.

3**(k-p), Box-Behnken, and Mixed 2 and 3 Level Factorial Designs

Overview

In some cases, factors that have more than 2 levels have to be examined. For example, if one suspects that the effect of the factors on the dependent variable of interest is not simply linear, then, as discussed earlier (see 2**(k-p) designs), one needs at least 3 levels in order to test for the linear and quadratic effects (and interactions) for those factors. Also, sometimes some factors may be categorical in nature, with more than 2 categories. For example, you may have three different machines that produce a particular part.

Designing 3**(k-p) Experiments

The general mechanism of generating fractional factorial designs at 3 levels (3**(k-p) designs) is very similar to that described in the context of 2**(k-p) designs. Specifically, one starts with a full factorial design, and then uses the interactions of the full design to construct "new" factors (or blocks) by making their factor levels identical to those for the respective interaction terms (i.e., by making the new factors aliases of the respective interactions).

For example, consider the following simple 3**(3-1) factorial design:

3(3-1) fractional factorial design, 1 block , 9 runs**

Standard Run	A	B	C
1	0	0	0
2	0	1	2
3	0	2	1
4	1	0	2
5	1	1	1
6	1	2	0
7	2	0	1
8	2	1	0
9	2	2	2

As in the case of 2**(k-p) designs, the design is constructed by starting with the full *3-1=2* factorial design; those factors are listed in the first two columns (factors *A* and *B*). Factor *C* is constructed from the interaction *AB* of the first two factors. Specifically, the values for factor *C* are computed as

$$C = 3 - \mathrm{mod}_3\ (A+B)$$

Here, $mod_3(x)$ stands for the so-called *modulo-3* operator, which will first find a number *y* that is less than or equal to *x*, and that is evenly divisible by 3, and then compute the difference (remainder) between number *y* and *x*. For example, $mod_3(0)$ is equal to *0*, $mod_3(1)$ is equal to *1*, $mod_3(3)$ is equal to *0*, $mod_3(5)$ is equal to *2* (*3* is the largest number that is less than or equal to *5*, and that is evenly divisible by *3*; finally, *5-3=2*), and so on.

Fundamental identity. If you apply this function to the sum of columns *A* and *B* shown above, you will obtain the third column C. Similar to the case of 2**(k-p) designs (see 2**(k-p) designs for a discussion of the *fundamental* identity in the context of 2**(k-p) designs), this confounding of interactions with "new" main effects can be summarized in an expression:

$$0 = \mathrm{mod}_3\ (A+B+C)$$

If you look back at the 3**(3-1) design shown earlier, you will see that, indeed, if you add the numbers in the three columns they will all sum to either *0*, *3*, or *6*, that is, values that are evenly divisible by 3 (and hence: $mod_3(A+B+C)=0$). Thus, one could write as a shortcut notation ABC=0, in order to summarize the confounding of factors in the fractional 3**(k-p) design.

Some of the designs will have fundamental identities that contain the number 2 as a multiplier; e.g.,

$0 = mod_3 (B+C*2+D+E*2+F)$

This notation can be interpreted exactly as before, that is, the $modulo_3$ of the sum $B+2*C+D+2*E+F$ must be equal to 0. The next example shows such an identity.

An Example 3**(4-1) Design in 9 Blocks

Here is the summary for a 4-factor 3-level fractional factorial design in 9 blocks, that requires only 27 runs.

SUMMARY: 3**(4-1) fractional factorial
Design generators: ABCD
Block generators: AB,AC2
Number of factors (independent variables): 4
Number of runs (cases, experiments): 27
Number of blocks: 9

This design will allow you to test for linear and quadratic main effects for 4 factors in 27 observations, which can be gathered in 9 blocks of 3 observations each. The fundamental identity or design generator for the design is *ABCD*, thus the modulo$_3$ of the sum of the factor levels across the four factors is equal to *0*. The fundamental identity also allows you to determine the confounding of factors and interactions in the design (see McLean and Anderson, 1984, for details).

Unconfounded Effects (experi3.sta)		
EXPERIM. DESIGN	**List of uncorrelated factors and interactions 3**(4-1) fractional factorial design, 9 blocks, 27 runs**	
	Unconf. Effects (excl. blocks)	**Unconfounded if blocks included?**
1	(1)A (L)	Yes
2	A (Q)	Yes
3	(2)B (L)	Yes
4	B (Q)	Yes
5	(3)C (L)	Yes
6	C (Q)	Yes
7	(4)D (L)	Yes
8	D (Q)	Yes

As you can see, in this 3**(4-1) design the main effects are not confounded with each other, even when the experiment is run in 9 blocks.

Box-Behnken Designs

In the case of 2**(k-p) designs, Plackett and Burman (1946) developed highly fractionalized designs to screen the maximum number of (main) effects in the least number of experimental runs. The equivalent in the case of 3**(k-p) designs are the so-called Box-Behnken designs (Box and Behnken, 1960; see also Box and Draper, 1984). These designs do not have simple design generators (they are constructed by combining two-level factorial designs with incomplete block designs), and have complex confounding of interaction. However, the designs are economical and therefore particularly useful when it is expensive to perform the necessary experimental runs.

Analyzing the 3**(k-p) Design

The analysis of these types of designs proceeds basically in the same way as was described in the context of 2**(k-p) designs. However, for each effect, one can now test for the linear effect and the quadratic (non-linear effect). For example, when studying the yield of chemical process, then temperature may be related in a non-linear fashion, that is, the maximum yield may be attained when the temperature is set at the medium level. Thus, non-linearity often occurs when a process performs near its optimum.

ANOVA Parameter Estimates

To estimate the ANOVA parameters, the factors levels for the factors in the analysis are internally recoded so that one can test the linear and quadratic components in the relationship between the factors and the dependent variable. Thus, regardless of the original metric of factor settings (e.g., *100 degrees C*, *110 degrees C*, *120 degrees* C), you can always recode those values to *-1*, *0*, and *+1* to perform the computations. The resultant ANOVA parameter estimates can be interpreted analogously to the parameter estimates for 2**(k-p) designs.

For example, consider the following ANOVA results:

Factor	**Effect**	**Std.Err.**	**t (69)**	**p**
Mean/Interc.	103.6942	.390591	265.4805	0.000000
BLOCKS(1)	.8028	1.360542	.5901	.557055
BLOCKS(2)	-1.2307	1.291511	-.9529	.343952
(1)TEMPERAT (L)	-.3245	.977778	-.3319	.740991
TEMPERAT (Q)	-.5111	.809946	-.6311	.530091
(2)TIME (L)	.0017	.977778	.0018	.998589
TIME (Q)	.0045	.809946	.0056	.995541
(3)SPEED (L)	-10.3073	.977778	-10.5415	.000000
SPEED (Q)	-3.7915	.809946	-4.6812	.000014
1L by 2L	3.9256	1.540235	2.5487	.013041
1L by 2Q	.4384	1.371941	.3195	.750297

1Q by 2L	.4747	1.371941	.3460	.730403
1Q by 2Q	-2.7499	.995575	-2.7621	.007353

Main-effect estimates. By default, the *Effect* estimate for the linear effects (marked by the *L* next to the factor name) can be interpreted as the difference between the average response at the low and high settings for the respective factors. The estimate for the quadratic (non-linear) effect (marked by the *Q* next to the factor name) can be interpreted as the difference between the average response at the center (medium) settings and the combined high and low settings for the respective factors.

Interaction effect estimates. As in the case of 2**(k-p) designs, the linear-by-linear interaction effect can be interpreted as half the difference between the linear main effect of one factor at the high and low settings of another. Analogously, the interactions by the quadratic components can be interpreted as half the difference between the quadratic main effect of one factor at the respective settings of another; that is, either the high or low setting (quadratic by linear interaction), or the medium or high and low settings combined (quadratic by quadratic interaction).

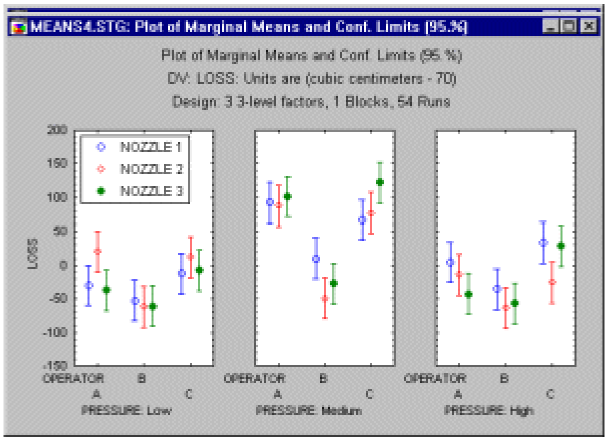
In practice, and from the standpoint of "interpretability of results," one would usually try to avoid quadratic interactions. For example, a quadratic-by-quadratic *A*-by-*B* interaction indicates that the non- linear effect of factor *A* is modified in a nonlinear fashion by the setting of *B*. This means that there is a fairly complex interaction between factors present in the data that will make it difficult to understand and optimize the respective process. Sometimes, performing nonlinear transformations (e.g., performing a *log* transformation) of the dependent variable values can remedy the problem.

Centered and non-centered polynomials. As mentioned above, the interpretation of the effect estimates applies only when you use the default parameterization of the model. In that case, you would code the quadratic factor interactions so that they become maximally "untangled" from the linear main effects.

Graphical Presentation of Results

The same diagnostic plots (e.g., of residuals) are available for 3**(k-p) designs as were described in the context of 2**(k-p) designs. Thus, before interpreting the final results, one should always first look at the distribution of the residuals for the final fitted model. The ANOVA assumes that the residuals (errors) are normally distributed.

Plot of means. When an interaction involves categorical factors (e.g., type of machine, specific operator of machine, and some distinct setting of the machine), then the best way to understand interactions is to look at the respective interaction plot of means.



Surface plot. When the factors in an interaction are continuous in nature, you may want to look at the surface plot that shows the response surface applied by the fitted model. Note that this graph also contains the prediction equation (in terms of the original metric of factors), that produces the respective response surface.

Designs for Factors at 2 and 3 Levels

You can also generate standard designs with 2 and 3 level factors. Specifically, you can generate the standard designs as enumerated by Connor and Young for the US National Bureau of Standards (see McLean and Anderson, 1984). The technical details of the method used to generate these designs are beyond the scope of this introduction. However, in general the technique is, in a sense, a combination of the procedures described in the context of 2**(k-p) and 3**(k-p) designs. It should be noted however, that, while all of these designs are very efficient, they are not necessarily orthogonal with respect to all main effects. This is, however, not a problem, if one uses a general algorithm for estimating the ANOVA parameters and sums of squares, that does not require orthogonality of the design.

The design and analysis of these experiments proceeds along the same lines as discussed in the context of 2**(k-p) and 3**(k-p) experiments.

Central Composite and Non-Factorial Response Surface Designs

Overview

The 2**(k-p) and 3**(k-p) designs all require that the levels of the factors are set at, for example, 2 or 3 levels. In many instances, such designs are not feasible, because, for example, some factor combinations are constrained in some way (e.g., factors *A* and *B* cannot be set at their *high* levels simultaneously). Also, for reasons related to efficiency, which will be discussed shortly, it is often desirable to explore the experimental region of interest at particular points that cannot be represented by a factorial design.

The designs (and how to analyze them) discussed in this section all pertain to the estimation (fitting) of response surfaces, following the general model equation:

$$y = b_0 + b_1 * x_1 + ... + b_k * x_k + b_{12} * x_1 * x_2 + b_{13} * x_1 * x_3 + ... + b_{k-1,k} * x_{k-1} * x_k + b_{11} * x_1^2 + ... + b_{kk} * x_k^2$$

Put into words, one is fitting a model to the observed values of the dependent variable y, that include (1) main effects for factors x_1 , ..., x_k, (2) their interactions ($x_1 * x_2$, $x_1 * x_3$, ... ,$x_{k-1} * x_k$), and (3) their quadratic components (x_1**2, ..., x_k**2). No assumptions are made concerning the "levels" of the factors, and you can analyze any set of continuous values for the factors.

There are some considerations concerning design efficiency and biases, which have led to standard designs that are ordinarily used when attempting to fit these response surfaces, and those standard designs will be discussed shortly (e.g., see Box, Hunter, and Hunter, 1978; Box and Draper, 1987; Khuri and Cornell, 1987; Mason, Gunst, and Hess, 1989; Montgomery, 1991). But, as will be discussed later, in the context of constrained surface designs and D- and A-optimal designs, these standard designs can sometimes not be used for practical reasons. However, the central composite design analysis options do not make any assumptions about the structure of your data file, that is, the number of distinct factor values, or their combinations across the runs of the experiment, and, hence, these options can be used to analyze any type of design, to fit to the data the general model described above.

Design Considerations

Orthogonal designs. One desirable characteristic of any design is that the main effect and interaction estimates of interest are independent of each other. For example, suppose you had a two- factor experiments, with both factors at two levels. Your design consists of four runs:

	A	B
Run 1	1	1
Run 2	1	1
Run 3	-1	-1
Run 4	-1	-1

For the first two runs, both factors *A* and *B* are set at their high levels (+*1*). In the last two runs, both are set at their low levels (-*1*). Suppose you wanted to estimate the independent contributions of factors *A* and *B* to the prediction of the dependent variable of interest. Clearly this is a silly design, because there is no way to estimate the *A* main effect and the *B* main effect. One can only estimate one effect -- the difference between *Runs 1+2* vs. *Runs 3+4* -- which represents the combined effect of *A* and *B*.

The point here is that, in order to assess the independent contributions of the two factors, the factor levels in the four runs must be set so that the "columns" in the design (under *A* and *B* in the illustration above) are independent of each other. Another way to express this requirement is to say that the columns of the design matrix (with as many columns as there are main effect and interaction parameters that one wants to estimate) should be *orthogonal* (this term was first used by Yates, 1933). For example, if the four runs in the design are arranged as follows:

	A	**B**
Run 1	1	1
Run 2	1	-1
Run 3	-1	1
Run 4	-1	-1

then the *A* and *B* columns are orthogonal. Now you can estimate the *A* main effect by comparing the high level for *A* within each level of *B*, with the low level for *A* within each level of *B*; the *B* main effect can be estimated in the same way.

Technically, two columns in a design matrix are orthogonal if the sum of the products of their elements within each row is equal to zero. In practice, one often encounters situations, for example due to loss of some data in some runs or other constraints, where the columns of the design matrix are not completely orthogonal. In general, the rule here is that the more orthogonal the columns are, the better the design, that is, the more independent information can be extracted from the design regarding the respective effects of interest. Therefore, one consideration for choosing standard central composite designs is to find designs that are orthogonal or near-orthogonal.

Rotatable designs. The second consideration is related to the first requirement, in that it also has to do with how best to extract the maximum amount of (unbiased) information from the design, or specifically, from the experimental region of interest. Without going into details (see Box, Hunter, and Hunter, 1978; Box and Draper, 1987, Chapters 14; see also Deming and Morgan, 1993, Chapter 13), it can be shown that the standard error for the prediction of dependent variable values is proportional to:

$$(1 + f(x)' * (X'X)^{¨1} * f(x))**½$$

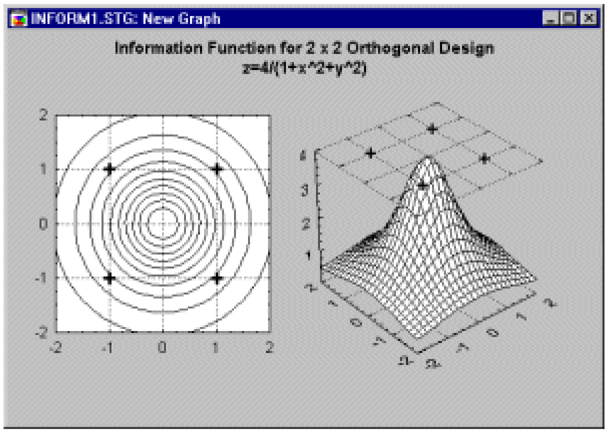
where *f(x)* stands for the (coded) factor effects for the respective model (*f(x)* is a vector, *f(x)'* is the transpose of that vector), and *X* is the design matrix for the experiment, that is, the matrix of coded factor effects for all runs; *X'X**-1* is the inverse of the crossproduct matrix. Deming and Morgan (1993) refer to this expression as the *normalized uncertainty*; this function is also related to the *variance function* as defined by Box and Draper (1987). The amount of uncertainty in the prediction of dependent variable values depends on the variability of the design points, and their covariance over the runs. (Note that it is inversely proportional to the determinant of *X'X*; this issue is further discussed in the section on D- and A-optimal designs).

The point here is that, again, one would like to choose a design that extracts the most *information* regarding the dependent variable, and leaves the least amount of uncertainty for the prediction of future values. It follows, that the amount of *information* (or *normalized information* according to Deming and Morgan, 1993) is the inverse of the normalized uncertainty.

For the simple 4-run orthogonal experiment shown earlier, the information function is equal to

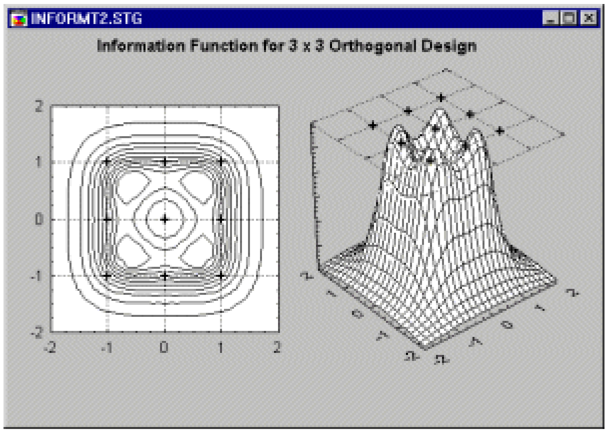
$I_x = 4/(1 + x_1^2 + x_2^2)$

where x_1 and x_2 stand for the factor settings for factors *A* and *B*, respectively (see Box and Draper, 1987).



Inspection of this function in a plot (see above) shows that it is constant on circles centered at the origin. Thus any kind of rotation of the original design points will generate the same amount of information, that is, generate the same information function. Therefore, the 2-by-2 orthogonal design in 4 runs shown earlier is said to be *rotatable*.

As pointed out before, in order to estimate the second order, quadratic, or non-linear component of the relationship between a factor and the dependent variable, one needs at least 3 levels for the respective factors. What does the information function look like for a simple 3-by-3 factorial design, for the second-order quadratic model as shown at the beginning of this section?

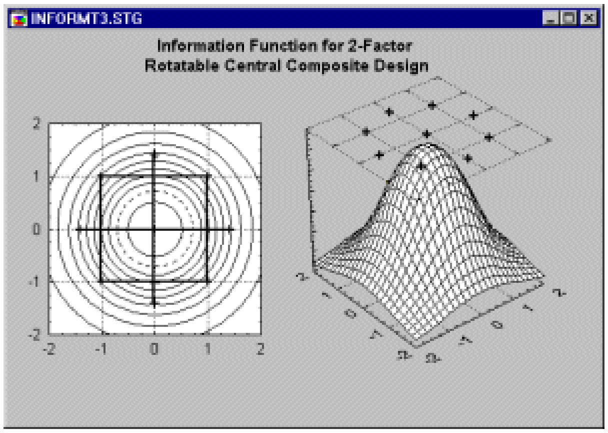


As it turns out (see Box and Draper, 1987 and Montgomery, 1991; refer also to the manual), this function looks more complex, contains "pockets" of high-density information at the edges (which are probably of little particular interest to the experimenter), and clearly it is not constant on circles around the origin. Therefore, it is not rotatable, meaning different rotations of the design points will extract different amounts of information from the experimental region.

Star-points and rotatable second-order designs. It can be shown that by adding so-called star- points to the simple (square or cube) 2-level factorial design points, one can achieve rotatable, and often orthogonal or nearly orthogonal designs. For example, adding to the simple 2-by-2 orthogonal design shown earlier the following points, will produce a rotatable design.

	A	B
Run 1	1	1
Run 2	1	-1
Run 3	-1	1
Run 4	-1	-1
Run 5	-1.414	0
Run 6	1.414	0
Run 7	0	-1.414
Run 8	0	1.414
Run 9	0	0
Run 10	0	0

The first four runs in this design are the previous 2-by-2 factorial design points (or *square points* or *cube points*); runs 5 through 8 are the so-called *star points* or *axial points*, and runs 9 and 10 are center points.


INFORMT3.STG
Information Function for 2-Factor
Rotatable Central Composite Design
2
1
0
-1
-2
-2
-1
0
1
2

The information function for this design for the second-order (quadratic) model is rotatable, that is, it is constant on the circles around the origin.

Alpha for Rotatability and Orthogonality

The two design characteristics discussed so far -- orthogonality and rotatability -- depend on the number of center points in the design and on the so-called *axial distance* α (*alpha*), which is the distance of the star points from the center of the design (i.e., *1.414* in the design shown above). It can be shown (e.g., see Box, Hunter, and Hunter, 1978; Box and Draper, 1987, Khuri and Cornell, 1987; Montgomery, 1991) that a design is rotatable if:

$$\alpha = (n_c)^{¼}$$

where n_c stands for the number of cube points in the design (i.e., points in the factorial portion of the design).

A central composite design is orthogonal, if one chooses the axial distance so that:

$$\alpha = \{[(n_c + n_s + n_0)^{½} - n_c^{½}]^2 * n_c/4\}^{¼}$$

where
n_c is the number of cube points in the design
n_s is the number of star points in the design
n_0 is the number of center points in the design

To make a design both (approximately) orthogonal and rotatable, one would first choose the axial distance for rotatability, and then add center points (see Kkuri and Cornell, 1987), so that:

$$n_0 \approx 4*n_c^{½} + 4 - 2k$$

where k stands for the number of factors in the design.

Finally, if blocking is involved, Box and Draper (1987) give the following formula for computing the axial distance to achieve orthogonal blocking, and in most cases also reasonable information function contours, that is, contours that are close to spherical:

$$\alpha = [k*(l+n_{s0}/n_s)/(1+n_{c0}/n_c)]^{1/2}$$

where
n_{s0} is the number of center points in the star portion of the design
n_s is the number of non-center star points in the design
n_{c0} is the number of center points in the cube portion of the design
n_c is the number of non-center cube points in the design

Available Standard Designs

The standard central composite designs are usually constructed from a 2**(k-p) design for the cube portion of the design, which is augmented with center points and star points. Box and Draper (1987) list a number of such designs.

Small composite designs. In the standard designs, the cube portion of the design is typically of resolution V (or higher). This is, however, not necessary, and in cases when the experimental runs are expensive, or when it is not necessary to perform a statistically powerful test of model adequacy, then one could choose for the cube portion designs of resolution III. For example, it could be constructed from highly fractionalized Plackett-Burman designs. Hartley (1959) described such designs.

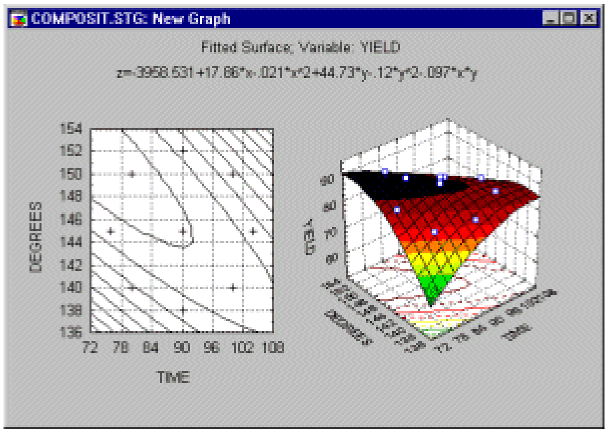
Analyzing Central Composite Designs

The analysis of central composite designs proceeds in much the same way as for the analysis of 3**(k-p) designs. You fit to the data the general model described above; for example, for two variables you would fit the model:

$$y = b_0 + b_1*x_1 + b_2*x_2 + b_{12}*x_1*x_2 + b_{11}*x_1^2 + b_{22}*x_2^2$$

The Fitted Response Surface

The shape of the fitted overall response can best be summarized in graphs and you can generate both contour plots and response surface plots (see examples below) for the fitted model.



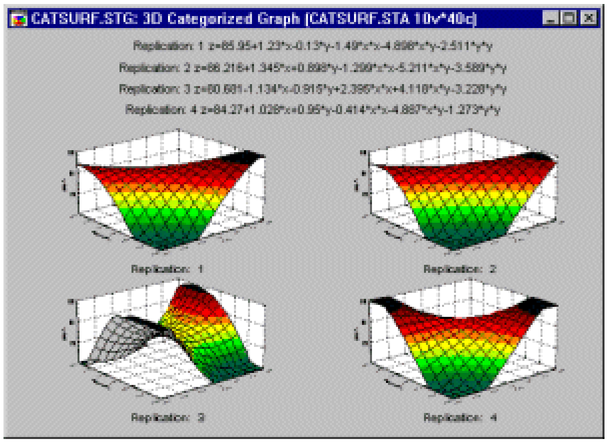
COMPOSIT.STG: New Graph

Fitted Surface; Variable: YIELD

z=-3958.531+17.86*x-.021*x^2+44.73*y-.12*y^2-.097*x*y

Categorized Response Surfaces

You can fit 3D surfaces to your data, categorized by some other variable. For example, if you replicated a standard central composite design 4 times, it may be very informative to see how similar the surfaces are when fitted to each replication.



CATSURF.STG: 3D Categorized Graph [CATSURF.STA 10v*40c]

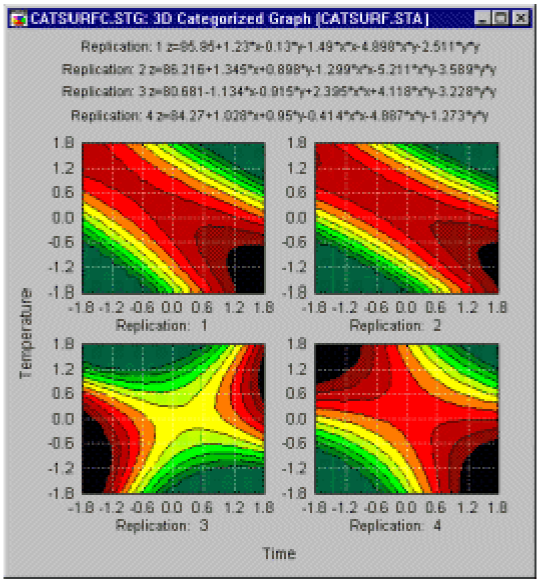
Replication: 1 z=85.95+1.23*x-0.13*y-1.49*x*x-4.898*x*y-2.511*y*y

Replication: 2 z=88.216+1.345*x+0.898*y-1.299*x*x-5.211*x*y-3.589*y*y

Replication: 3 z=80.681-1.134*x-0.915*y+2.396*x*x+4.118*x*y-3.228*y*y

Replication: 4 z=84.27+1.028*x+0.95*y-0.414*x*x-4.887*x*y-1.273*y*y

This would give you a graphical indication of the reliability of the results and where (e.g., in which region of the surface) deviations occur.



Clearly, the third replication produced a different surface. In replications *1*, *2*, and *4*, the fitted surfaces are very similar to each other. Thus, one should investigate what could have caused this noticeable difference in the third replication of the design.

Latin Square Designs

Overview

Latin square designs (the term *Latin square* was first used by Euler, 1782) are used when the factors of interest have more than two levels and you know ahead of time that there are no (or only negligible) interactions between factors. For example, if you wanted to examine the effect of 4 fuel additives on reduction in oxides of nitrogen and had 4 cars and 4 drivers at your disposal, then you could of course run a full 4 x 4 x 4 factorial design, resulting in 64 experimental runs. However, you are not really interested in any (minor) interactions between the fuel additives and drivers, fuel additives and cars, or cars and drivers. You are mostly interested in estimating main effects, in particular the one for the fuel additives factor. At the same time, you want to make sure that the main effects for drivers and cars do not affect (bias) your estimate of the main effect for the fuel additive.

If you labeled the additives with the letters A, B, C, and D, the Latin square design that would allow you to derive unconfounded main effects estimates could be summarized as follows (see also Box, Hunter, and Hunter, 1978, page 263):

	Car			
Driver	**1**	**2**	**3**	**4**
1	A	B	D	C
2	D	C	A	B
3	B	D	C	A
4	C	A	B	D

Latin Square Designs

The example shown above is actually only one of the three possible arrangements in effect estimates. These "arrangements" are also called *Latin square*. The example above constitutes a 4 x 4 Latin square; and rather than requiring the 64 runs of the complete factorial, you can complete the study in only 16 runs.

Greco-Latin square. A nice feature of Latin Squares is that they can be superimposed to form what are called *Greco-Latin squares* (this term was first used by Fisher and Yates, 1934). For example, the following two 3 x 3 Latin squares can be superimposed to form a Greco-Latin square:

```
a b c       α β τ                aα bβ cτ
b c a  and  τ α β  results in    bτ cα aβ
c a b       β τ α                cβ aτ bα
```

In the resultant Greco-Latin square design, you can evaluate the main effects of four 3-level factors (row factor, column factor, Roman letters, Greek letters) in only 9 runs.

Hyper-Greco Latin square. For some numbers of levels, there are more than two possible Latin square arrangements. For example, there are three possible arrangements for 4-level Latin squares. If all three of them are superimposed, you get a *Hyper-Greco Latin square* design. In that design you can estimate the main effects of all five 4-level factors with only 16 runs in the experiment.

Analyzing the Design

Analyzing Latin square designs is straightforward. Also, plots of means can be produced to aid in the interpretation of results.

Very Large Designs, Random Effects, Unbalanced Nesting

Note that there are several other statistical methods that can also analyze these types of designs; see the section on Methods for Analysis of Variance for details. In particular the Variance Components and Mixed Model ANOVA/ANCOVA chapter discusses very efficient methods for analyzing designs with unbalanced nesting (when the nested factors have different numbers of levels within the levels of the factors in which they are nested), very large nested designs (e.g., with more than 200 levels overall), or hierarchically nested designs (with or without random factors).

Taguchi Methods: Robust Design Experiments

Overview

Applications. Taguchi methods have become increasingly popular in recent years. The documented examples of sizable quality improvements that resulted from implementations of these methods (see, for example, Phadke, 1989; Noori, 1989) have added to the curiosity among American manufacturers. In fact, some of the leading manufacturers in this country have begun to use these methods with usually great success. For example, AT&T is using these methods in the manufacture of very large scale integrated (*VLSI*) circuits; also, Ford Motor Company has gained significant quality improvements due to these methods (American Supplier Institute, 1984 to 1988). However, as the details of these methods are becoming more widely known, critical appraisals are also beginning to appear (for example, Bhote, 1988; Tribus and Szonyi, 1989).

Overview. Taguchi robust design methods are set apart from traditional quality control procedures (see Quality Control and Process Analysis) and industrial experimentation in various respects. Of particular importance are:

1. The concept of *quality loss functions*,
2. The use of *signal-to-noise* (S/N) ratios, and
3. The use of *orthogonal arrays*.

These basic aspects of robust design methods will be discussed in the following sections. Several books have recently been published on these methods, for example, Peace (1993), Phadke (1989), Ross (1988), and Roy (1990), to name a few, and it is recommended that you refer to those books for further specialized discussions. Introductory overviews of Taguchi's ideas about quality and quality improvement can also be found in Barker (1986), Garvin (1987), Kackar (1986), and Noori (1989).

Quality and Loss Functions

What is quality. Taguchi's analysis begins with the question of how to define quality. It is not easy to formulate a simple definition of what constitutes *quality*; however, when your new car stalls in the middle of a busy intersection -- putting yourself and other motorists at risk -- you know that your car is *not* of high quality. Put another way, the definition of the *inverse* of quality is rather straightforward: it is the total *loss* to you and society due to functional variations and harmful side effects associated with the respective product. Thus, as an operational definition, you can measure quality in terms of this loss, and the greater the quality loss the lower the quality.

Discontinuous (step-shaped) loss function. You can formulate hypotheses about the general nature and shape of the loss function. Assume a specific *ideal* point of highest quality; for example, a perfect car with no quality problems. It is customary in statistical process control (*SPC; see also Process Analysis*) to define tolerances around the nominal ideal point of the production process. According to the traditional view implied by common SPC methods, as long as you are within the manufacturing tolerances you do not have a problem. Put another way, within the tolerance limits the quality loss is zero; once you move outside the tolerances, the quality loss is declared to be unacceptable. Thus, according to traditional views, the quality loss function is a *discontinuous step function*: as long as you are within the tolerance limits, quality loss is negligible; when you step outside those tolerances, quality loss becomes unacceptable.

Quadratic loss function. Is the step function implied by common SPC methods a good model of quality loss? Return to the "perfect automobile" example. Is there a difference between a car that, within one year after purchase, has *nothing* wrong with it, and a car where minor rattles develop, a few fixtures fall off, and the clock in the dashboard breaks (all in-warranty repairs, mind you...)? If you ever bought a new car of the latter kind, you know very well how annoying those admittedly minor quality problems can be. The point here is that it is not realistic to assume that, as you move away from the nominal specification in your production process, the quality loss is zero as long as you stay within the set tolerance limits. Rather, if you are not exactly "on target," then loss will result, for example in terms of customer satisfaction. Moreover, this loss is probably not a linear function of the deviation from nominal specifications, but rather a *quadratic function* (inverted *U*). A rattle in one place in your new car is annoying, but you would probably not get too upset about it; add two more rattles, and you might declare the car "junk." Gradual deviations from the nominal specifications do not produce proportional increments in loss, but rather squared increments.

Conclusion: Controlling variability. If, in fact, quality loss is a quadratic function of the deviation from a nominal value, then the goal of your quality improvement efforts should be to *minimize the squared deviations or variance* of the product around nominal (ideal) specifications, rather than the *number of units within specification limits* (as is done in traditional SPC procedures).

Signal-to-Noise (S/N) Ratios

Measuring quality loss. Even though you have concluded that the quality loss function is probably quadratic in nature, you still do not know precisely how to measure quality loss. However, you know that whatever measure you decide upon should reflect the quadratic nature of the function.

Signal, noise, and control factors. The product of ideal quality should always respond in exactly the same manner to the *signals* provided by the user. When you turn the key in the ignition of your car you expect that the starter motor turns and the engine starts. In the ideal-quality car, the starting process would always proceed in exactly the same manner -- for example, after three turns of the starter motor the engine comes to life. If, in response to the same signal (turning the ignition key) there is random variability in this process, then you have less than ideal quality. For example, due to such uncontrollable factors as extreme cold, humidity, engine wear, etc. the engine may sometimes start only after turning over 20 times and finally not start at all. This example illustrates the key principle in measuring quality according to Taguchi: You want to minimize the variability in the product's performance in response to *noise* factors while maximizing the variability in response to *signal* factors.

Noise factors are those that are not under the control of the operator of a product. In the car example, those factors include temperature changes, different qualities of gasoline, engine wear, etc. *Signal* factors are those factors that are set or controlled by the operator of the product to make use of its intended functions (turning the ignition key to start the car).

Finally, the goal of your quality improvement effort is to find the best settings of factors under your *control* that are involved in the production process, in order to maximize the S/N ratio; thus, the factors in the experiment represent *control* factors.

S/N ratios. The conclusion of the previous paragraph is that quality can be quantified in terms of the respective product's response to noise factors and signal factors. The ideal product will only respond to the operator's signals and will be unaffected by random noise factors (weather, temperature, humidity, etc.). Therefore, the goal of your quality improvement effort can be stated as attempting to *maximize the signal-to-noise (S/N)* ratio for the respective product. The S/N ratios described in the following paragraphs have been proposed by Taguchi (1987).

Smaller-the-better. In cases where you want to minimize the occurrences of some undesirable product characteristics, you would compute the following S/N ratio:

$\text{Eta} = -10 * \log_{10} [(1/n) * \Sigma(y_i^2)]$ for i = 1 to no. vars   see outer arrays

Here, *Eta* is the resultant S/N ratio; *n* is the number of observations on the particular product, and *y* is the respective characteristic. For example, the number of flaws in the paint on an automobile could be measured as the *y* variable and analyzed via this S/N ratio. The effect of the signal factors is zero, since zero flaws is the only intended or desired state of the paint on the car. Note how this S/N ratio is an expression of the assumed *quadratic* nature of the loss function. The factor *10* ensures that this ratio measures the inverse of "bad quality;" the more flaws in the paint, the greater is the sum of the squared number of flaws, and the smaller (i.e., more negative) the S/N ratio. Thus, maximizing this ratio will increase quality.

Nominal-the-best. Here, you have a fixed signal value (nominal value), and the variance around this value can be considered the result of noise factors:

$\text{Eta} = 10 * \log_{10} (\text{Mean}^2/\text{Variance})$

This signal-to-noise ratio could be used whenever ideal quality is equated with a particular nominal value. For example, the size of piston rings for an automobile engine must be as close to specification as possible to ensure high quality.

Larger-the-better. Examples of this type of engineering problem are fuel economy (miles per gallon) of an automobile, strength of concrete, resistance of shielding materials, etc. The following S/N ratio should be used:

$\text{Eta} = -10 * \log_{10} [(1/n) * \Sigma(1/y_i^2)]$ for i = 1 to no. vars see outer arrays

Signed target. This type of S/N ratio is appropriate when the quality characteristic of interest has an ideal value of 0 (zero), and both positive and negative values of the quality characteristic may occur. For example, the dc offset voltage of a differential operational amplifier may be positive or negative (see Phadke, 1989). The following S/N ratio should be used for these types of problems:

$\text{Eta} = -10 * \log_{10}(s^2)$ for i = 1 to no. vars see outer arrays

where s^2 stands for the variance of the quality characteristic across the measurements (variables).

Fraction defective. This S/N ratio is useful for minimizing scrap, minimizing the percent of patients who develop side-effects to a drug, etc. Taguchi also refers to the resultant Eta values as Omegas; note that this S/N ratio is identical to the familiar logit transformation (see also Nonlinear Estimation):

$\text{Eta} = -10 * \log_{10}[p/(1-p)]$

where
p is the proportion defective

Ordered categories (the accumulation analysis). In some cases, measurements on a quality characteristic can only be obtained in terms of categorical judgments. For example, consumers may rate a product as *excellent*, *good*, *average*, or *below average*. In that case, you would attempt to maximize the number of *excellent* or *good* ratings. Typically, the results of an accumulation analysis are summarized graphically in a stacked bar plot.

Orthogonal Arrays

The third aspect of Taguchi robust design methods is the one most similar to traditional techniques. Taguchi has developed a system of tabulated designs (arrays) that allow for the maximum number of main effects to be estimated in an unbiased (orthogonal) manner, with a minimum number of runs in the experiment. Latin square designs, 2**(k-p) designs (Plackett-Burman designs, in particular), and Box-Behnken designs main are also aimed at accomplishing this goal. In fact, many of the standard orthogonal arrays tabulated by Taguchi are identical to fractional two-level factorials, Plackett-Burman designs, Box-Behnken designs, Latin square, Greco-Latin squares, etc.

Analyzing Designs

Most analyses of robust design experiments amount to a standard ANOVA of the respective S/N ratios, ignoring two-way or higher-order interactions. However, when estimating error variances, one customarily pools together main effects of negligible size.

Analyzing S/N ratios in standard designs. It should be noted at this point that, of course, all of the designs discussed up to this point (e.g., 2**(k-p), 3**(k-p), mixed 2 and 3 level factorials, Latin squares, central composite designs) can be used to analyze S/N ratios that you computed. In fact, the many additional diagnostic plots and other options available for those designs (e.g., estimation of quadratic components, etc.) may prove very useful when analyzing the variability (S/N ratios) in the production process.

Plot of means. A visual summary of the experiment is the plot of the average *Eta* (S/N ratio) by factor levels. In this plot, the optimum setting (i.e., largest S/N ratio) for each factor can easily be identified.

Verification experiments. For prediction purposes, you can compute the expected S/N ratio given a user-defined combination of settings of factors (ignoring factors that were pooled into the error term). These predicted S/N ratios can then be used in a verification experiment, where the engineer actually sets the machine accordingly and compares the resultant observed S/N ratio with the predicted S/N ratio from the experiment. If major deviations occur, one must conclude that the simple main effect model is not appropriate.

In those cases, Taguchi (1987) recommends transforming the dependent variable to accomplish additivity of factors, that is, to "make" the main effects model fit. Phadke (1989, Chapter 6) also discusses in detail methods for achieving additivity of factors.

Accumulation Analysis

When analyzing ordered categorical data, ANOVA is not appropriate. Rather, you produce a cumulative plot of the number of observations in a particular category. For each level of each factor, you plot the cumulative proportion of the number of defectives. Thus, this graph provides valuable information concerning the distribution of the categorical counts across the different factor settings.

Summary

To briefly summarize, when using Taguchi methods you first need to determine the *design* or *control* factors that can be set by the designer or engineer. Those are the factors in the experiment for which you will try different levels. Next, you decide to select an appropriate orthogonal array for the experiment. Next, you need to decide on how to measure the quality characteristic of interest. Remember that most S/N ratios require that multiple measurements are taken in each run of the experiment; for example, the variability around the nominal value cannot otherwise be assessed. Finally, you conduct the experiment and identify the factors that most strongly affect the chosen S/N ratio, and you reset your machine or production process accordingly.

Mixture Designs and Triangular Surfaces

Overview

Special issues arise when analyzing mixtures of components that must sum to a constant. For example, if you wanted to optimize the taste of a fruit-punch, consisting of the juices of 5 fruits, then the sum of the proportions of all juices in each mixture must be 100%. Thus, the task of optimizing mixtures commonly occurs in food-processing, refining, or the manufacturing of chemicals. A number of designs have been developed to address specifically the analysis and modeling of mixtures (see, for example, Cornell, 1990a, 1990b; Cornell and Khuri, 1987; Deming and Morgan, 1993; Montgomery, 1991).

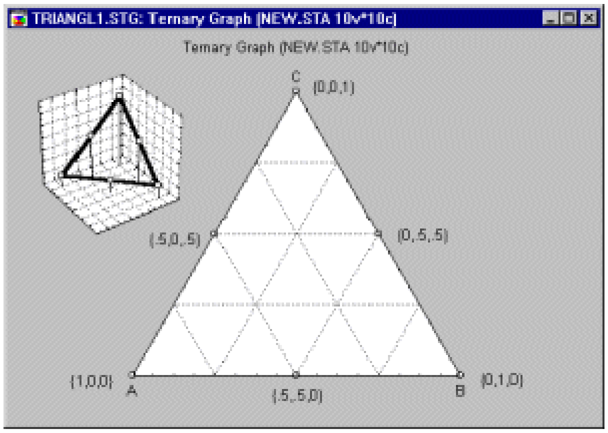
Triangular Coordinates

The common manner in which mixture proportions can be summarized is via triangular (ternary) graphs. For example, suppose you have a mixture that consists of 3 components *A*, *B*, and *C*. Any mixture of the three components can be summarized by a point in the triangular coordinate system defined by the three variables.

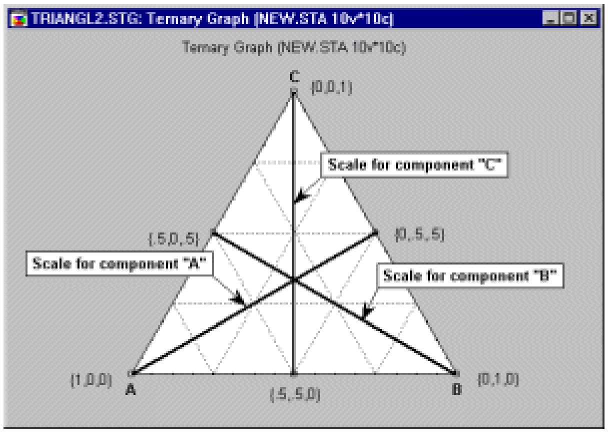
For example, take the following 6 different mixtures of the 3 components.

A	B	C
1	0	0
0	1	0
0	0	1
0.5	0.5	0
0.5	0	0.5
0	0.5	0.5

The sum for each mixture is 1.0, so the values for the components in each mixture can be interpreted as proportions. If you graph these data in a regular 3D scatterplot, it becomes apparent that the points form a triangle in the 3D space. Only the points inside the triangle where the sum of the component values is equal to 1 are valid mixtures. Therefore, one can simply plot only the triangle to summarize the component values (proportions) for each mixture.


TRIANGL1.STG: Ternary Graph (NEW.STA 10v*10c)
Ternary Graph (NEW.STA 10v*10c)
C
(0,0,1)
(.5,0,.5)
(0,.5,.5)
(1,0,0)
A
(.5,.5,0)
B
(0,1,0)

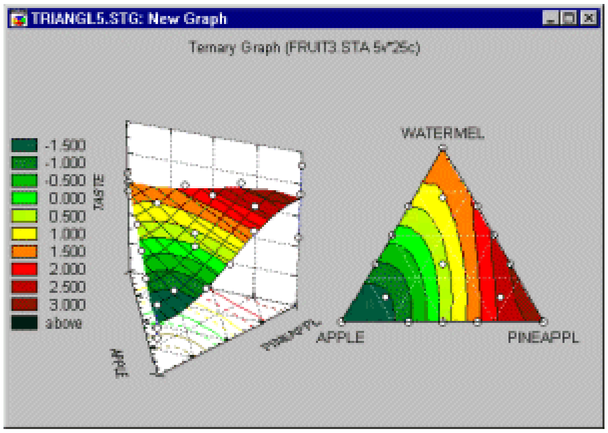
To read-off the coordinates of a point in the triangular graph, you would simply "drop" a line from each respective vertex to the side of the triangle below.



At the vertex for the particular factor, there is a pure blend, that is, one that only contains the respective component. Thus, the coordinates for the vertex point is *1* (or *100%*, or however else the mixtures are scaled) for the respective component, and *0* (zero) for all other components. At the side opposite to the respective vertex, the value for the respective component is *0* (zero), and *.5* (or *50%*, etc.) for the other components.

Triangular Surfaces and Contours

One can now add to the triangle a fourth dimension, that is perpendicular to the first three. Using that dimension, one could plot the values for a dependent variable, or function (surface) that was fit to the dependent variable. Note that the response surface can either be shown in 3D, where the predicted response (*Taste* rating) is indicated by the distance of the surface from the triangular plane, or it can be indicated in a contour plot where the contours of constant height are plotted on the 2D triangle.



It should be mentioned at this point that you can produce categorized ternary graphs. These are very useful, because they allow you to fit to a dependent variable (e.g., Taste) a response surface, for different levels of a fourth component.

The Canonical Form of Mixture Polynomials

Fitting a response surface to mixture data is, in principle, done in the same manner as fitting surfaces to, for example, data from central composite designs. However, there is the issue that mixture data are constrained, that is, the sum of all component values must be constant.

Consider the simple case of two factors *A* and *B*. One may want to fit the simple linear model:

$$y = b_0 + b_A*x_A + b_B*x_B$$

Here y stands for the dependent variable values, b_A and b_B stand for the regression coefficients, x_A and x_B stand for the values of the factors. Suppose that x_A and x_B must sum to 1; you can multiple b_0 by $1=(x_A + x_B)$:

$$y = (b_0*x_A + b_0*x_B) + b_A*x_A + b_B*x_B$$

or:

$$y = b'_A*x_A + b'_B*x_B$$

where $b'_A = b_0 + b_A$ *and* $b'_B = b_0 + b_B$. Thus, the estimation of this model comes down to fitting a no- intercept multiple regression model. (See also Multiple Regression, for details concerning multiple regression.)

Common Models for Mixture Data

The quadratic and cubic model can be similarly simplified (as illustrated for the simple linear model above), yielding four standard models that are customarily fit to the mixture data. Here are the formulas for the 3-variable case for those models (see Cornell, 1990, for additional details).

Linear model:

$$y = b_1*x_1 + b_2*x_2 + b_3*x_3$$

Quadratic model:

$$y = b_1*x_1 + b_2*x_2 + b_3*x_3 + b_{12}*x_1*x_2 + b_{13}*x_1*x_3 + b_{23}*x_2*x_3$$

Special cubic model:

$$y = b_1*x_1 + b_2*x_2 + b_3*x_3 + b_{12}*x_1*x_2 + b_{13}*x_1*x_3 + b_{23}*x_2*x_3 + b_{123}*x_1*x_2*x_3$$

Full cubic model:

$$y = b_1*x_1 + b_2*x_2 + b_3*x_3 + b_{12}*x_1*x_2 + b_{13}*x_1*x_3 + b_{23}*x_2*x_3 + d_{12}*x_1*x_2*(x_1 - x_2) + d_{13}*x_1*x_3*(x_1 - x_3) + d_{23}*x_2*x_3*(x_2 - x_3) + b_{123}*x_1*x_2*x_3$$

(Note that the d_{ij}'s are also parameters of the model.)

Standard Designs for Mixture Experiments

Two different types of standard designs are commonly used for experiments with mixtures. Both of them will evaluate the triangular response surface at the vertices (i.e., the corners of the triangle) and the centroids (sides of the triangle). Sometimes, those designs are enhanced with additional interior points.

Simplex-lattice designs. In this arrangement of design points, *m+1* equally spaced proportions are tested for each factor or component in the model:

$$x_i = 0, 1/m, 2/m, ..., 1 \quad i = 1,2,...,q$$

and all combinations of factor levels are tested. The resulting design is called a *{q,m} simplex lattice* design. For example, a *{q=3, m=2}* simplex lattice design will include the following mixtures:

A	B	C
1	0	0
0	1	0
0	0	1

.5	.5	0
.5	0	.5
0	.5	.5

A *{q=3,m=3}* simplex lattice design will include the points:

A	B	C
1	0	0
0	1	0
0	0	1
1/3	2/3	0
1/3	0	2/3
0	1/3	2/3
2/3	1/3	0
2/3	0	1/3
0	2/3	1/3
1/3	1/3	1/3

Simplex-centroid designs. An alternative arrangement of settings introduced by Scheffé (1963) is the so-called *simplex-centroid* design. Here the design points correspond to all permutations of the pure blends (e.g., *1 0 0; 0 1 0; 0 0 1*), the permutations of the binary blends (*½ ½ 0; ½ 0 ½; 0 ½ ½*), the permutations of the blends involving three components, and so on. For example, for 3 factors the simplex centroid design consists of the points:

A	B	C
1	0	0
0	1	0
0	0	1
1/2	1/2	0
1/2	0	1/2
0	1/2	1/2
1/3	1/3	1/3

Adding interior points. These designs are sometimes augmented with interior points (see Khuri and Cornell, 1987, page 343; Mason, Gunst, Hess; 1989; page 230). For example, for 3 factors one could add the interior points:

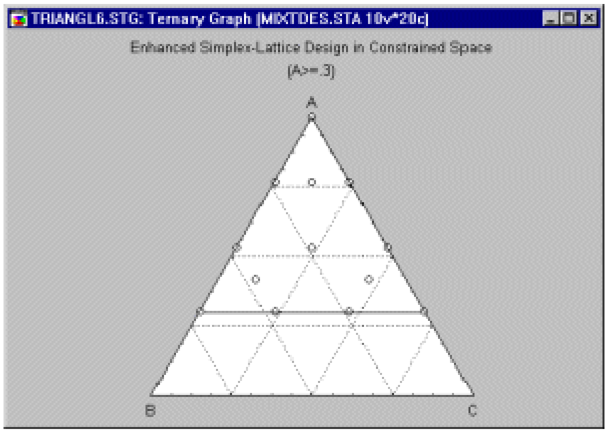
A	B	C
2/3	1/6	1/6
1/6	2/3	1/6
1/6	1/6	2/3

If you plot these points in a scatterplot with triangular coordinates; one can see how these designs evenly cover the experimental region defined by the triangle.

Lower Constraints

The designs described above all require vertex points, that is, pure blends consisting of only one ingredient. In practice, those points may often not be valid, that is, pure blends cannot be produced because of cost or other constraints. For example, suppose you wanted to study the effect of a food- additive on the taste of the fruit-punch. The additional ingredient may only be varied within small limits, for example, it may not exceed a certain percentage of the total. Clearly, a fruit punch that is a pure blend, consisting only of the additive, would not be a fruit punch at all, or worse, may be toxic. These types of constraints are very common in many applications of mixture experiments.

Let us consider a 3-component example, where component *A* is constrained so that $x_A \geq .3$. The total of the 3-component mixture must be equal to 1. This constraint can be visualized in a triangular graph by a line at the triangular coordinate for $x_A=.3$, that is, a line that is parallel to the triangle's edge opposite to the *A* vertex point.

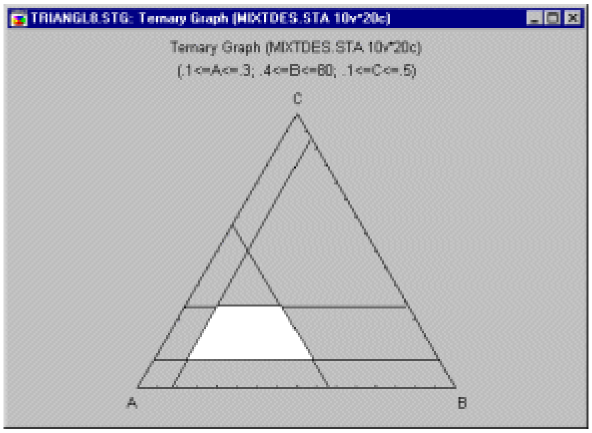

TRIANGL6.STG: Ternary Graph (MIXTDES.STA 10v*20c)
Enhanced Simplex-Lattice Design in Constrained Space
(A>=.3)
A
B
C

One can now construct the design as before, except that one side of the triangle is defined by the constraint. Later, in the analysis, one can review the parameter estimates for the so-called *pseudo-components*, treating the constrained triangle as if it were a full triangle.

Multiple constraints. Multiple lower constraints can be treated analogously, that is, you can construct the sub-triangle within the full triangle, and then place the design points in that sub-triangle according to the chosen design.

Upper and Lower Constraints

When there are both upper and lower constraints (as is often the case in experiments involving mixtures), then the standard simplex-lattice and simplex-centroid designs can no longer be constructed, because the subregion defined by the constraints is no longer a triangle. There is a general algorithm for finding the vertex and centroid points for such constrained designs.


TRIANGL8.STG: Ternary Graph (MIXTDES.STA 10v*20c)
Ternary Graph (MIXTDES.STA 10v*20c)
(.1<=A<=.3; .4<=B<=80; .1<=C<=.5)
C
A
B

Note that you can still analyze such designs by fitting the standard models to the data.

Analyzing Mixture Experiments

The analysis of mixture experiments amounts to a multiple regression with the intercept set to zero. As explained earlier, the mixture constraint -- that the sum of all components must be constant -- can be accommodated by fitting multiple regression models that do not include an intercept term. If you are not familiar with multiple regression, you may want to review at this point Multiple Regression.

The specific models that are usually considered were described earlier. To summarize, one fits to the dependent variable response surfaces of increasing complexity, that is, starting with the linear model, then the quadratic model, special cubic model, and full cubic model. Shown below is a table with the number of terms or parameters in each model, for a selected number of components (see also Table 4, Cornell, 1990):

	Model (Degree of Polynomial)			
No. of Comp.	**Linear**	**Quadr.**	**Special Cubic**	**Full Cubic**
2	2	3	--	--
3	3	6	7	10

4	4	10	14	20
5	5	15	25	35
6	6	21	41	56
7	7	28	63	84
8	8	36	92	120

Analysis of Variance

To decide which of the models of increasing complexity provides a sufficiently good fit to the observed data, one usually compares the models in a hierarchical, stepwise fashion. For example, consider a 3- component mixture to which the full cubic model was fitted.

ANOVA; Var.:DV (mixt4.sta)										
	3 Factor mixture design; Mixture total=1., 14 Runs **Sequential fit of models of increasing complexity**									
Model	**SS Effect**	**df Effect**	**MS Effect**	**SS Error**	**df Error**	**MS Error**	**F**	**p**	**R-sqr**	**R-sqr Adj.**
Linear	44.755	2	22.378	46.872	11	4.2611	5.2516	.0251	.4884	.3954
Quadratic	30.558	3	10.186	16.314	8	2.0393	4.9949	.0307	.8220	.7107
Special Cubic	.719	1	.719	15.596	7	2.2279	.3225	.5878	.8298	.6839
Cubic	8.229	3	2.743	7.367	4	1.8417	1.4893	.3452	.9196	.7387
Total Adjusted	91.627	13	7.048							

First, the linear model was fit to the data. Even though this model has 3 parameters, one for each component, this model has only 2 degrees of freedom. This is because of the overall mixture constraint, that the sum of all component values is constant. The simultaneous test for all parameters of this model is statistically significant ($F(2,11)=5.25$; $p<.05$). The addition of the 3 quadratic model parameters ($b_{12}*x_1*x_2$, $b_{13}*x_1*x_3$, $b_{23}*x_2*x_3$) further significantly improves the fit of the model ($F(3,8)=4.99$; $p<.05$). However, adding the parameters for the special cubic and cubic models does not significantly improve the fit of the surface. Thus one could conclude that the quadratic model provides an adequate fit to the data (of course, pending further examination of the residuals for outliers, etc.).

R-square. The *R-square* value can be interpreted as the proportion of variability around the mean for the dependent variable, that can be accounted for by the respective model. (Note that for non- intercept models, some multiple regression programs will only compute the *R-square* value pertaining to the proportion of variance around 0 (zero) accounted for by the independent variables; for more information, see Kvalseth, 1985; Okunade, Chang, and Evans, 1993.)

Pure error and lack of fit. The usefulness of the estimate of *pure error* for assessing the overall *lack of fit* was discussed in the context of central composite designs. If some runs in the design were replicated, then one can compute an estimate of error variability based only on the variability between replicated runs. This variability provides a good indication of the unreliability in the measurements, independent of the model that was fit to the data, since it is based on identical factor settings (or blends in this case). One can test the residual variability after fitting the current model against this estimate of *pure error*. If this test is statistically significant, that is, if the residual variability is significantly larger than the pure error variability, then one can conclude that, most likely, there are additional significant differences between blends that cannot be accounted for by the current model. Thus, there may be an overall *lack of fit* of the current model. In that case, try a more complex model, perhaps by only adding individual terms of the next higher-order model (e.g., only the $b_{13}*x_1*x_3$ to the linear model).

Parameter Estimates

Usually, after fitting a particular model, one would next review the parameter estimates. Remember that the linear terms in mixture models are constrained, that is, the sum of the components must be constant. Hence, independent statistical significance tests for the linear components cannot be performed.

Pseudo-Components

To allow for scale-independent comparisons of the parameter estimates, during the analysis, the component settings are customarily recoded to so-called pseudo-components so that (see also Cornell, 1993, Chapter 3):

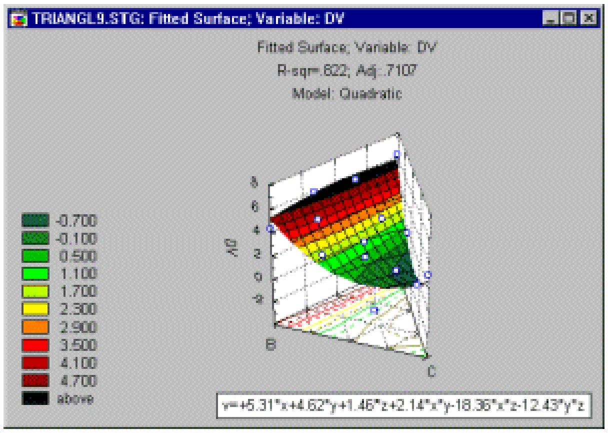
$$x'_i = (x_i-L_i)/(\text{Total}-L)$$

Here, x'_i stands for the i'th pseudo-component, x_i stands for the original component value, L_i stands for the lower constraint (limit) for the i'th component, L stands for the sum of all lower constraints (limits) for all components in the design, and Total is the mixture total.

The issue of lower constraints was also discussed earlier in this section. If the design is a standard simplex-lattice or simplex-centroid design (see above), then this transformation amounts to a rescaling of factors so as to form a sub-triangle (sub-simplex) as defined by the lower constraints. However, you can compute the parameter estimates based on the original (untransformed) metric of the components in the experiment. If you want to use the fitted parameter values for prediction purposes (i.e., to predict dependent variable values), then the parameters for the untransformed components are often more convenient to use. Note that the results dialog for mixture experiments contains options to make predictions for the dependent variable for user-defined values of the components, in their original metric.

Graph Options

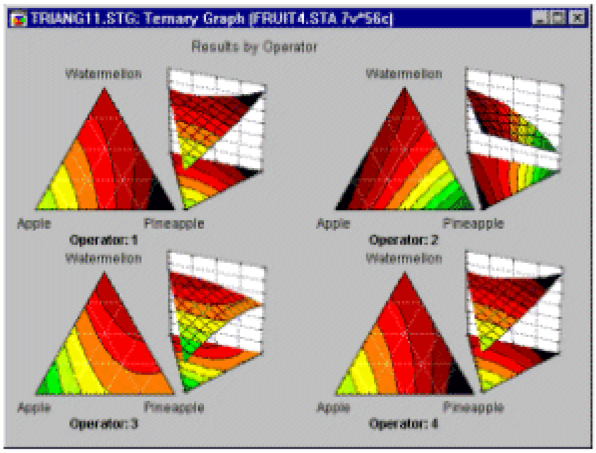
Surface and contour plots. The respective fitted model can be visualized in triangular surface plots or contour plots, which, optionally, can also include the respective fitted function.



y=+5.31*x+4.62*y+1.46*z+2.14*x*y-18.36*x*z-12.43*y*z

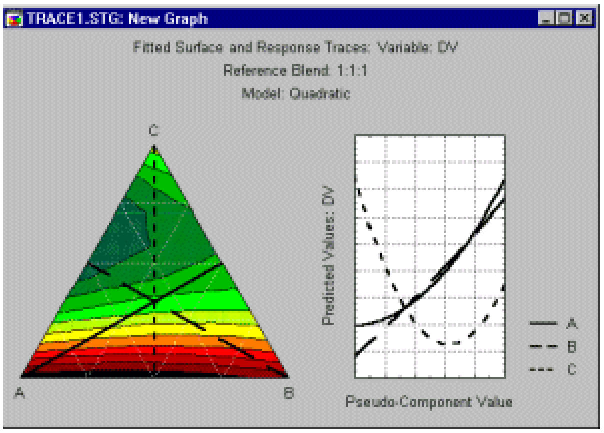
Note that the fitted function displayed in the surface and contour plots always pertains to the parameter estimates for the pseudo-components.

Categorized surface plots. If your design involves replications (and the replications are coded in your data file), then you can use *3D Ternary Plots* to look at the respective fit, replication by replication.



Of course, if you have other categorical variables in your study (e.g., operator or experimenter; machine, etc.) you can also categorize the 3D surface plot by those variables.

Trace plots. One aid for interpreting the triangular response surface is the so-called *trace plot*. Suppose you looked at the contour plot of the response surface for three components. Then, determine a reference blend for two of the components, for example, hold the values for *A* and *B* at *1/3* each. Keeping the relative proportions of *A* and *B* constant (i.e., equal proportions in this case), you can then plot the estimated response (values for the dependent variable) for different values of *C*.



If the reference blend for *A* and *B* is *1:1*, then the resulting line or *response trace* is the axis for factor *C*; that is, the line from the C vertex point connecting with the opposite side of the triangle at a right angle. However, trace plots for other reference blends can also be produced. Typically, the trace plot contains the traces for all components, given the current reference blend.

Residual plots. Finally, it is important, after deciding on a model, to review the prediction residuals, in order to identify outliers or regions of misfit-fit. In addition, one should review the standard normal probability plot of residuals and the scatterplot of observed versus predicted values. Remember that the multiple regression analysis (i.e., the process of fitting the surface) assumes that the residuals are normally distributed, and one should carefully review the residuals for any apparent outliers.

Designs for Constrained Surfaces and Mixtures

Overview

As mentioned in the context of mixture designs, it often happens in real-world studies that the experimental region of interest is constrained, that is, that not all factors settings can be combined with all settings for the other factors in the study. There is an algorithm suggested by Piepel (1988) and Snee (1985) for finding the vertices and centroids for such constrained regions.

Designs for Constrained Experimental Regions

When in an experiment with many factors, there are constraints concerning the possible values of those factors and their combinations, it is not clear how to proceed. A reasonable approach is to include in the experiments runs at the extreme vertex points and centroid points of the constrained region, which should usually provide good coverage of the constrained experimental region (e.g., see Piepel, 1988; Snee, 1975). In fact, the mixture designs reviewed in the previous section provide examples for such designs, since they are typically constructed to include the vertex and centroid points of the constrained region that consists of a triangle (simplex).

Linear Constraints

One general way in which one can summarize most constraints that occur in real world experimentation is in terms of a linear equation (see Piepel, 1988):

$$A_1x_1 + A_2x_2 + ... + A_qx_q + A_0 \geq 0$$

Here, A_0, .., A_q are the parameters for the linear constraint on the q factors, and x_1,.., x_q stands for the factor values (levels) for the q factors. This general formula can accommodate even very complex constraints. For example, suppose that in a two-factor experiment the first factor must always be set at least twice as high as the second, that is, $x_1 \geq 2*x_2$. This simple constraint can be rewritten as $x_1-2*x_2 \geq 0$. The ratio constraint $2*x_1 /x_2 \geq 1$ can be rewritten as $2*x_1 - x_2 \geq 0$, and so on.

The problem of multiple upper and lower constraints on the component values in mixtures was discussed earlier, in the context of mixture experiments. For example, suppose in a three-component mixture of fruit juices, the upper and lower constraints on the components are (see example 3.2, in Cornell 1993):

40% $\leq$ Watermelon (x_1) $\leq$ 80%
10% $\leq$ Pineapple (x_2) $\leq$ 50%
10% $\leq$ Orange (x_3) $\leq$ 30%

These constraints can be rewritten as linear constraints into the form:

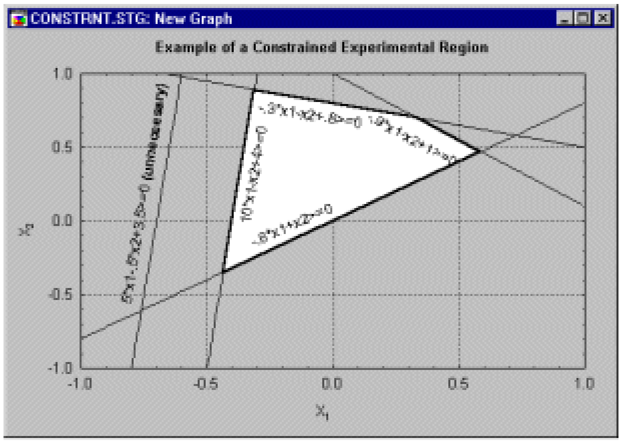
Watermelon:	$x_1-40 \geq 0$
	$-x_1+80 \geq 0$
Pineapple:	$x_2-10 \geq 0$
	$-x_2+50 \geq 0$
Orange:	$x_3-10 \geq 0$
	$-x_3+30 \geq 0$

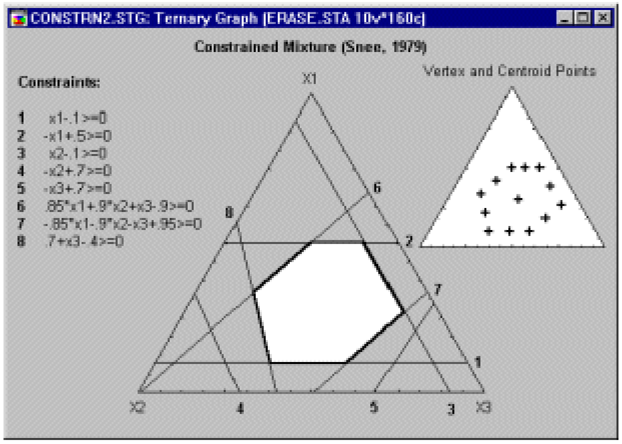
Thus, the problem of finding design points for mixture experiments with components with multiple upper and lower constraints is only a special case of general linear constraints.

The Piepel & Snee Algorithm

For the special case of constrained mixtures, algorithms such as the *XVERT* algorithm (see, for example, Cornell, 1990) are often used to find the vertex and centroid points of the constrained region (inside the triangle of three components, tetrahedron of four components, etc.). The general algorithm proposed by Piepel (1988) and Snee (1979) for finding vertices and centroids can be applied to mixtures as well as non-mixtures. The general approach of this algorithm is described in detail by Snee (1979).

Specifically, it will consider one-by-one each constraint, written as a linear equation as described above. Each constraint represents a line (or plane) through the experimental region. For each successive constraint you will evaluate whether or not the current (new) constraint crosses into the current valid region of the design. If so, new vertices will be computed which define the new valid experimental region, updated for the most recent constraint. It will then check whether or not any of the previously processed constraints have become redundant, that is, define lines or planes in the experimental region that are now entirely outside the valid region. After all constraints have been processed, it will then compute the centroids for the sides of the constrained region (of the order requested by the user). For the two-dimensional (two-factor) case, one can easily recreate this process by simply drawing lines through the experimental region, one for each constraint; what is left is the valid experimental region.




CONSTRN2.STG: Ternary Graph [ERASE.STA 10v*160c]
Constrained Mixture (Snee, 1979)
Vertex and Centroid Points
X1
X2
X3
1
2
3
4
5
6
7
8

Constraints:

1 x1-.1>=0
2 -x1+.5>=0
3 x2-.1>=0
4 -x2+.7>=0
5 -x3+.7>=0
6 .85*x1+.9*x2+x3-.9>=0
7 -.85*x1-.9*x2-x3+.95>=0
8 .7+x3-.4>=0

For more information, see Piepel (1988) or Snee (1979).

Choosing Points for the Experiment

Once the vertices and centroids have been computed, you may face the problem of having to select a subset of points for the experiment. If each experimental run is costly, then it may not be feasible to simply run all vertex and centroid points. In particular, when there are many factors and constraints, then the number of centroids can quickly get very large.

If you are screening a large number of factors, and are not interested in non-linear effects, then choosing the vertex points only will usually yield good coverage of the experimental region. To increase statistical power (to increase the degrees of freedom for the ANOVA error term), you may also want to include a few runs with the factors set at the overall centroid of the constrained region.

If you are considering a number of different models that you might fit once the data have been collected, then you may want to use the D- and A-optimal design options. Those options will help you select the design points that will extract the maximum amount of information from the constrained experimental region, given your models.

Analyzing Designs for Constrained Surfaces and Mixtures

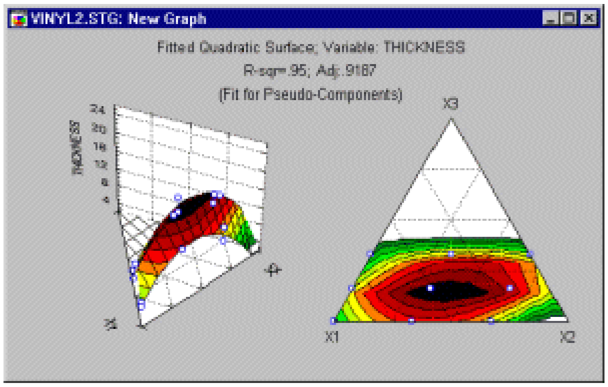
As mentioned in the section on central composite designs and mixture designs, once the constrained design points have been chosen for the final experiment, and the data for the dependent variables of interest have been collected, the analysis of these designs can proceed in the standard manner.

For example, Cornell (1990, page 68) describes an experiment of three plasticizers, and their effect on resultant vinyl thickness (for automobile seat covers). The constraints for the three plasticizers components x_1, x_2, and x_3 are:

$.409 \leq x_1 \leq .849$
$.000 \leq x_2 \leq .252$
$.151 \leq x_3 \leq .274$

(Note that these values are already rescaled, so that the total for each mixture must be equal to 1.) The vertex and centroid points generated are:

x_1	x_2	x_3
.8490	.0000	.1510
.7260	.0000	.2740
.4740	.2520	.2740
.5970	.2520	.1510
.6615	.1260	.2125
.7875	.0000	.2125
.6000	.1260	.2740
.5355	.2520	.2125
.7230	.1260	.1510


VINYL2.STG: New Graph
Fitted Quadratic Surface; Variable: THICKNESS
R-sqr=.95; Adj:.9187
(Fit for Pseudo-Components)
THICKNESS
24
20
16
12
8
4
X1
X2
X3
X1
X2

Constructing D- and A-Optimal Designs

Overview

In the sections on standard factorial designs (see *2**(k-p) Fractional Factorial Designs* and *3**(k-p), Box Behnken, and Mixed 2 and 3 Level Factorial Designs*) and *Central Composite Designs*, the property of *orthogonality* of factor effects was discussed. In short, when the factor level settings for two factors in an experiment are uncorrelated, that is, when they are varied independently of each other, then they are said to be *orthogonal* to each other. (If you are familiar with matrix and vector algebra, two column vectors X_1 and X_2 in the design matrix are orthogonal if $X_1'*X_2= 0$). Intuitively, it should be clear that one can extract the maximum amount of information regarding a dependent variable from the experimental region (the region defined by the settings of the factor levels), if all factor effects are orthogonal to each other. Conversely, suppose one ran a four-run experiment for two factors as follows:

	x_1	x_2
Run 1	1	1
Run 2	1	1
Run 3	-1	-1
Run 4	-1	-1

Now the columns of factor settings for X_1 and X_2 are identical to each other (their correlation is *1*), and there is no way in the results to distinguish between the main effect for X_1 and X_2.

The *D*- and *A*-optimal design procedures provide various options to select from a list of valid (candidate) points (i.e., combinations of factor settings) those points that will extract the maximum amount of information from the experimental region, given the respective model that you expect to fit to the data. You need to supply the list of candidate points, for example the vertex and centroid points computed by the *Designs for constrained surface and mixtures* option, specify the type of model you expect to fit to the data, and the number of runs for the experiment. It will then construct a design with the desired number of cases, that will provide as much orthogonality between the columns of the design matrix as possible.

The reasoning behind *D*- and *A*-optimality is discussed, for example, in Box and Draper (1987, Chapter 14). The different algorithms used for searching for optimal designs are described in Dykstra (1971), Galil and Kiefer (1980), and Mitchell (1974a, 1974b). A detailed comparison study of the different algorithms is discussed in Cook and Nachtsheim (1980).

Basic Ideas

A technical discussion of the reasoning (and limitations) of *D*- and *A*-optimal designs is beyond the scope of this introduction. However, the general ideas are fairly straightforward. Consider again the simple two-factor experiment in four runs.

	x_1	x_2
Run 1	1	1
Run 2	1	1
Run 3	-1	-1
Run 4	-1	-1

As mentioned above, this design, of course, does not allow one to test, independently, the statistical significance of the two variables' contribution to the prediction of the dependent variable. If you computed the correlation matrix for the two variables, they would correlate at *1*:

	x_1	x_2
x_1	1.0	1.0
x_2	1.0	1.0

Normally, one would run this experiment so that the two factors are varied independently of each other:

	x_1	x_2
Run 1	1	1
Run 2	1	-1
Run 3	-1	1
Run 4	-1	-1

Now the two variables are uncorrelated, that is, the correlation matrix for the two factors is:

	x_1	x_2

x_1	1.0	0.0
x_2	0.0	1.0

Another term that is customarily used in this context is that the two factors are orthogonal. Technically, if the sum of the products of the elements of two columns (*vectors*) in the design (design *matrix*) is equal to 0 (zero), then the two columns are orthogonal.

The determinant of the design matrix. The determinant *D* of a square matrix (like the 2-by-2 correlation matrices shown above) is a specific numerical value, that reflects the amount of independence or redundancy between the columns and rows of the matrix. For the 2-by-2 case, it is simply computed as the product of the diagonal elements minus the off-diagonal elements of the matrix (for larger matrices the computations are more complex). For example, for the two matrices shown above, the determinant *D* is:

D_1 = |1.0 1.0| = 1*1 - 1*1 = 0
|1.0 1.0|

D_2 = |1.0 0.0| = 1*1 - 0*0 = 1
|0.0 1.0|

Thus, the determinant for the first matrix computed from completely redundant factor settings is equal to *0*. The determinant for the second matrix, when the factors are orthogonal, is equal to *1*.

D-optimal designs. This basic relationship extends to larger design matrices, that is, the more redundant the vectors (columns) of the design matrix, the closer to *0* (zero) is the determinant of the correlation matrix for those vectors; the more independent the columns, the larger is the determinant of that matrix. Thus, finding a design matrix that maximizes the determinant *D* of this matrix means finding a design where the factor effects are maximally independent of each other. This criterion for selecting a design is called the *D-optimality* criterion.

Matrix notation. Actually, the computations are commonly not performed on the correlation matrix of vectors, but on the simple cross-product matrix. In matrix notation, if the design matrix is denoted by *X*, then the quantity of interest here is the determinant of *X'X (X*- transposed times *X*). Thus, the search for *D-optimal* designs aims to maximize |X'X|, where the vertical lines (|..|) indicate the determinant.

A-optimal designs. Looking back at the computations for the determinant, another way to look at the issue of independence is to maximize the diagonal elements of the *X'X* matrix, while minimizing the off-diagonal elements. The so-called *trace criterion* or *A-optimality* criterion expresses this idea. Technically, the *A*-criterion is defined as:

$$A = \mathrm{trace}(X'X)^{-1}$$

where *trace* stands for the sum of the diagonal elements (of the $(X'X)^{-1}$ matrix).

The information function. It should be mentioned at this point that *D*-optimal designs minimize the expected prediction error for the dependent variable, that is, those designs will maximize the precision of prediction, and thus the *information* (which is defined as the inverse of the error) that is extracted from the experimental region of interest.

Measuring Design Efficiency

A number of standard measures have been proposed to summarize the efficiency of a design.

D-efficiency. This measure is related to the *D*-optimality criterion:

$$\text{D-efficiency} = 100 * (|X'X|^{1/p}/N)$$

Here, *p* is the number of factor effects in the design (columns in *X*), and *N* is the number of requested runs. This measure can be interpreted as the relative number of runs (in percent) that would be required by an orthogonal design to achieve the same value of the determinant $|X'X|$. However, remember that an orthogonal design may not be possible in many cases, that is, it is only a theoretical "yard-stick." Therefore, you should use this measure rather as a relative indicator of efficiency, to compare other designs of the same size, and constructed from the same design points candidate list. Also note that this measure is only meaningful (and will only be reported) if you chose to recode the factor settings in the design (i.e., the factor settings for the design points in the candidate list), so that they have a minimum of *-1* and a maximum of *+1*.

A-efficiency. This measure is related to the A-optimality criterion:

$$\text{A-efficiency} = 100 * p/\mathrm{trace}(N*(X'X)^{-1})$$

Here, *p* stands for the number of factor effects in the design, *N* is the number of requested runs, and *trace* stands for the sum of the diagonal elements (of $(N*(X'X)^{-1})$). This measure can be interpreted as the relative number of runs (in percent) that would be required by an orthogonal design to achieve the same value of the trace of $(X'X)^{-1}$. However, again you should use this measure as a relative indicator of efficiency, to compare other designs of the same size and constructed from the same design points candidate list; also this measure is only meaningful if you chose to recode the factor settings in the design to the *-1* to *+1* range.

G-efficiency. This measure is computed as:

G-efficiency = 100 * square root(p/N)/σ_M

Again, *p* stands for the number of factor effects in the design and *N* is the number of requested runs; σ_M *(sigma$_M$)* stands for the maximum standard error for prediction across the list of candidate points. This measure is related to the so-called *G- optimality* criterion; *G-optimal* designs are defined as those that will minimize the maximum value of the standard error of the predicted response.

Constructing Optimal Designs

The optimal design facilities will "search for" optimal designs, given a list of "candidate points." Put another way, given a list of points that specifies which regions of the design are valid or feasible, and given a user-specified number of runs for the final experiment, it will select points to optimize the respective criterion. This "searching for" the best design is not an exact method, but rather an algorithmic procedure that employs certain search strategies to find the best design (according to the respective optimality criterion).

The search procedures or algorithms that have been proposed are described below (for a review and detailed comparison, see Cook and Nachtsheim, 1980). They are reviewed here in the order of speed, that is, the *Sequential* or *Dykstra* method is the fastest method, but often most likely to fail, that is, to yield a design that is not optimal (e.g., only locally optimal; this issue will be discussed shortly).

Sequential or Dykstra method. This algorithm is due to Dykstra (1971). Starting with an empty design, it will search through the candidate list of points, and choose in each step the one that maximizes the chosen criterion. There are no iterations involved, they will simply pick the requested number of points sequentially. Thus, this method is the fastest of the ones discussed. Also, by default, this method is used to construct the initial designs for the remaining methods.

Simple exchange (Wynn-Mitchell) method. This algorithm is usually attributed to Mitchell and Miller (1970) and Wynn (1972). The method starts with an initial design of the requested size (by default constructed via the *sequential* search algorithm described above). In each iteration, one point (run) in the design will be dropped from the design and another added from the list of candidate points. The choice of points to be dropped or added is sequential, that is, at each step the point that contributes least with respect to the chosen optimality criterion (*D* or *A*) is dropped from the design; then the algorithm chooses a point from the candidate list so as to optimize the respective criterion. The algorithm stops when no further improvement is achieved with additional exchanges.

DETMAX algorithm (exchange with excursions). This algorithm, due to Mitchell (1974b), is probably the best known and most widely used optimal design search algorithm. Like the simple exchange method, first an initial design is constructed (by default, via the *sequential* search algorithm described above). The search begins with a simple exchange as described above. However, if the respective criterion (*D* or *A*) does not improve, the algorithm will undertake *excursions*. Specifically, the algorithm will add or subtract more than one point at a time, so that, during the search, the number of points in the design may vary between N_D+ $N_{excursion}$ and N_D- $N_{excursion}$, where N_D is the requested design size, and $N_{excursion}$ refers to the maximum allowable excursion, as specified by the user. The iterations will stop when the chosen criterion (*D* or *A*) no longer improves within the maximum excursion.

Modified Fedorov (simultaneous switching). This algorithm represents a modification (Cook and Nachtsheim, 1980) of the basic Fedorov algorithm described below. It also begins with an initial design of the requested size (by default constructed via the *sequential* search algorithm). In each iteration, the algorithm will exchange each point in the design with one chosen from the candidate list, so as to optimize the design according to the chosen criterion (*D* or *A*). Unlike the simple exchange algorithm described above, the exchange is not sequential, but simultaneous. Thus, in each iteration each point in the design is compared with each point in the candidate list, and the exchange is made for the pair that optimizes the design. The algorithm terminates when there are no further improvements in the respective optimality criterion.

Fedorov (simultaneous switching). This is the original simultaneous switching method proposed by Fedorov (see Cook and Nachtsheim, 1980). The difference between this procedure and the one described above (*modified Fedorov*) is that in each iteration only a single exchange is performed, that is, in each iteration all possible pairs of points in the design and those in the candidate list are evaluated. The algorithm will then exchange the pair that optimizes the design (with regard to the chosen criterion). Thus, it is easy to see that this algorithm potentially can be somewhat slow, since in each iteration N_D*N_C comparisons are performed, in order to exchange a single point.

General Recommendations

If you think about the basic strategies represented by the different algorithms described above, it should be clear that there are usually no exact solutions to the optimal design problem. Specifically, the determinant of the *X'X* matrix (and trace of its inverse) are complex functions of the list of candidate points. In particular, there are usually several "local minima" with regard to the chosen optimality criterion; for example, at any point during the search a design may appear optimal unless you simultaneously discard half of the points in the design and choose certain other points from the candidate list; but, if you only exchange individual points or only a few points (via DETMAX), then no improvement occurs.

Therefore, it is important to try a number of different initial designs and algorithms. If after repeating the optimization several times with random starts the same, or very similar, final optimal design results, then you can be reasonably sure that you are not "caught" in a local minimum or maximum.

Also, the methods described above vary greatly with regard to their ability to get "trapped" in local minima or maxima. As a general rule, the slower the algorithm (i.e., the further down on the list of algorithms described above), the more likely is the algorithm to yield a truly optimal design. However, note that the modified *Fedorov* algorithm will practically perform just as well as the unmodified algorithm (see Cook and Nachtsheim, 1980); therefore, if time is not a consideration, we recommend the modified Fedorov algorithm as the best method to use.

D-optimality and A-optimality. For computational reasons (see Galil and Kiefer, 1980), updating the trace of a matrix (for the *A*-optimality criterion) is much slower than updating the determinant (for *D*-optimality). Thus, when you choose the *A*-optimality criterion, the computations may require significantly more time as compared to the *D*-optimality criterion. Since in practice, there are many other factors that will affect the quality of an experiment (e.g., the measurement reliability for the dependent variable), we generally recommend that you use the *D* optimality criterion. However, in difficult design situations, for example, when there appear to be many local maxima for the *D* criterion, and repeated trials yield very different results, you may want to run several optimization trials using the *A* criterion to learn more about the different types of designs that are possible.

Avoiding Matrix Singularity

It may happen during the search process that it cannot compute the inverse of the *X'X* matrix (for *A*-optimality), or that the determinant of the matrix becomes almost 0 (zero). At that point, the search can usually not continue. To avoid this situation, perform the optimization based on an augmented *X'X* matrix:

$$X'X_{augmented} = X'X + \alpha *(X_0'X_0/N_0)$$

where X_0 stands for the design matrix constructed from the list of all N_0 candidate points, and α (*alpha*) is a user-defined small constant. Thus, you can turn off this feature by setting α to 0 (zero).

"Repairing" Designs

The optimal design features can be used to "repair" designs. For example, suppose you ran an orthogonal design, but some data were lost (e.g., due to equipment malfunction), and now some effects of interest can no longer be estimated. You could of course make up the lost runs, but suppose you do not have the resources to redo them all. In that case, you can set up the list of candidate points from among all valid points for the experimental region, add to that list all the points that you have already run, and instruct it to always force those points into the final design (and never to drop them out; you can mark points in the candidate list for such forced inclusion). It will then only consider to exclude those points from the design that you did not actually run. In this manner you can, for example, find the best single run to add to an existing experiment, that would optimize the respective criterion.

Constrained Experimental Regions and Optimal Design

A typical application of the optimal design features is to situations when the experimental region of interest is constrained. As described earlier in this section, there are facilities for finding vertex and centroid points for linearly constrained regions and mixtures. Those points can then be submitted as the candidate list for constructing an optimal design of a particular size for a particular model. Thus, these two facilities combined provide a very powerful tool to cope with the difficult design situation when the design region of interest is subject to complex constraints, and one wants to fit particular models with the least number of runs.

Special Topics

The following sections introduce several analysis techniques. The sections describe Response/desirability profiling, conducting Residual analyses, and performing Box-Cox transformations of the dependent variable.

See also ANOVA/MANOVA, Methods for Analysis of Variance, and Variance Components and Mixed Model ANOVA/ANCOVA.

Profiling Predicted Responses and Response Desirability

Basic Idea. A typical problem in product development is to find a set of conditions, or levels of the input variables, that produces the most desirable product in terms of its characteristics, or responses on the output variables. The procedures used to solve this problem generally involve two steps: (1) predicting responses on the dependent, or *Y* variables, by fitting the observed responses using an equation based on the levels of the independent, or *X* variables, and (2) finding the levels of the *X* variables which simultaneously produce the most desirable predicted responses on the *Y* variables. Derringer and Suich (1980) give, as an example of these procedures, the problem of finding the most desirable tire tread compound. There are a number of *Y* variables, such as PICO Abrasion Index, 200 percent modulus, elongation at break, and hardness. The characteristics of the product in terms of the response variables depend on the ingredients, the *X* variables, such as hydrated silica level, silane coupling agent level, and sulfur. The problem is to select the levels for the *X*'s which will maximize the desirability of the responses on the *Y*'s. The solution must take into account the fact that the levels for the *X*'s that maximize one response may not maximize a different response.

When analyzing *2**(k-p) (two-level factorial) designs*, *2-level screening designs*, *2**(k-p) maximally unconfounded and minimum aberration designs*, *3**(k-p) and Box Behnken designs*, *Mixed 2 and 3 level designs*, *central composite designs*, and *mixture designs*, *Response/desirability profiling* allows you to inspect the response surface produced by fitting the observed responses using an equation based on levels of the independent variables.

Prediction Profiles. When you analyze the results of any of the designs listed above, a separate prediction equation for each dependent variable (containing different coefficients but the same terms) is fitted to the observed responses on the respective dependent variable. Once these equations are constructed, predicted values for the dependent variables can be computed at any combination of levels of the predictor variables. A *prediction profile* for a dependent variable consists of a series of graphs, one for each independent variable, of the predicted values for the dependent variable at different levels of one independent variable, holding the levels of the other independent variables constant at specified values, called *current values*. If appropriate *current values* for the independent variables have been selected, inspecting the *prediction profile* can show which levels of the predictor variables produce the most desirable predicted response on the dependent variable.

One might be interested in inspecting the predicted values for the dependent variables only at the actual levels at which the independent variables were set during the experiment. Alternatively, one also might be interested in inspecting the predicted values for the dependent variables at levels other than the actual levels of the independent variables used during the experiment, to see if there might be intermediate levels of the independent variables that could produce even more desirable responses. Also, returning to the Derringer and Suich (1980) example, for some response variables, the most desirable values may not necessarily be the most extreme values, for example, the most desirable value of elongation may fall within a narrow range of the possible values.

Response Desirability. Different dependent variables might have different kinds of relationships between scores on the variable and the desirability of the scores. Less filling beer may be more desirable, but better tasting beer can also be more desirable--lower "fillingness" scores and higher "taste" scores are both more desirable. The relationship between predicted responses on a dependent variable and the desirability of responses is called the desirability function. Derringer and Suich (1980) developed a procedure for specifying the relationship between predicted responses on a dependent variable and the desirability of the responses, a procedure that provides for up to three "inflection" points in the function. Returning to the tire tread compound example described above, their procedure involved transforming scores on each of the four tire tread compound outcome variables into desirability scores that could range from 0.0 for undesirable to 1.0 for very desirable. For example, their desirability function for hardness of the tire tread compound was defined by assigning a desirability value of 0.0 to hardness scores below 60 or above 75, a desirability value of 1.0 to mid-point hardness scores of 67.5, a desirability value that increased linearly from 0.0 up to 1.0 for hardness scores between 60 and 67.5 and a desirability value that decreased linearly from 1.0 down to 0.0 for hardness scores between 67.5 and 75.0. More generally, they suggested that procedures for defining desirability functions should accommodate curvature in the "falloff" of desirability between inflection points in the functions.

After transforming the predicted values of the dependent variables at different combinations of levels of the predictor variables into individual desirability scores, the overall desirability of the outcomes at different combinations of levels of the predictor variables can be computed. Derringer and Suich (1980) suggested that overall desirability be computed as the geometric mean of the individual desirabilities (which makes intuitive sense, because if the individual desirability of any outcome is 0.0, or unacceptable, the overall desirability will be 0.0, or unacceptable, no matter how desirable the other individual outcomes are--the geometric mean takes the product of all of the values, and raises the product to the power of the reciprocal of the number of values). Derringer and Suich's procedure provides a straightforward way for transforming predicted values for multiple dependent variables into a single overall desirability score. The problem of simultaneously optimization of several response variables then boils down to selecting the levels of the predictor variables that maximize the overall desirability of the responses on the dependent variables.

Summary. When one is developing a product whose characteristics are known to depend on the "ingredients" of which it is constituted, producing the best product possible requires determining the effects of the ingredients on each characteristic of the product, and then finding the balance of ingredients that optimizes the overall desirability of the product. In data analytic terms, the procedure that is followed to maximize product desirability is to (1) find adequate models (i.e., prediction equations) to predict characteristics of the product as a function of the levels of the independent variables, and (2) determine the optimum levels of the independent variables for overall product quality. These two steps, if followed faithfully, will likely lead to greater success in product improvement than the fabled, but statistically dubious technique of hoping for accidental breakthroughs and discoveries that radically improve product quality.

Residuals Analysis

Basic Idea. *Extended residuals analysis* is a collection of methods for inspecting different residual and predicted values, and thus to examine the adequacy of the prediction model, the need for transformations of the variables in the model, and the existence of outliers in the data.

Residuals are the deviations of the observed values on the dependent variable from the predicted values, given the current model. The ANOVA models used in analyzing responses on the dependent variable make certain assumptions about the distributions of residual (but not predicted) values on the dependent variable. These assumptions can be summarized by saying that the ANOVA model assumes *normality*, *linearity*, *homogeneity of variances and covariances*, and *independence* of residuals. All of these properties of the residuals for a dependent variable can be inspected using *Residuals analysis*.

Box-Cox Transformations of Dependent Variables

Basic Idea. It is assumed in analysis of variance that the variances in the different groups (experimental conditions) are homogeneous, and that they are uncorrelated with the means. If the distribution of values within each experimental condition is skewed, and the means are correlated with the standard deviations, then one can often apply an appropriate power transformation to the dependent variable to stabilize the variances, and to reduce or eliminate the correlation between the means and standard deviations. The *Box-Cox transformation* is useful for selecting an appropriate (power) transformation of the dependent variable.

Selecting the *Box-Cox transformation* option will produce a plot of the *Residual Sum of Squares*, given the model, as a function of the value of *lambda*, where *lambda* is used to define a transformation of the dependent variable,

y' = (y**(lambda) - 1) / (g**(lambda-1) * lambda)   if lambda ≥ 0
y' = g * natural log(y)   if lambda = 0

in which *g* is the geometric mean of the dependent variable and all values of the dependent variable are non-negative. The value of *lambda* for which the *Residual Sum of Squares* is a minimum is the maximum likelihood estimate for this parameter. It produces the variance stabilizing transformation of the dependent variable that reduces or eliminates the correlation between the group means and standard deviations.

In practice, it is not important that you use the *exact* estimated value of *lambda* for transforming the dependent variable. Rather, as a rule of thumb, one should consider the following transformations:

Approximate lambda	**Suggested transorfmation of y**
-1	Reciprocal
-0.5	Reciprocal square root
0	Natural logarithm
0.5	Square root
1	None

For additional information regarding this family of transformations, see Box and Cox (1964), Box and Draper (1987), and Maddala (1977).

Chapter 16
Principal Components and Factor Analysis

- General Purpose
- Basic Idea of Factor Analysis as a Data Reduction Method
- Factor Analysis as a Classification Method
- Miscellaneous Other Issues and Statistics

General Purpose

The main applications of factor analytic techniques are: (1) to *reduce* the number of variables and (2) to *detect structure* in the relationships between variables, that is to *classify variables*. Therefore, factor analysis is applied as a data reduction or structure detection method (the term *factor analysis* was first introduced by Thurstone, 1931). The topics listed below will describe the principles of factor analysis, and how it can be applied towards these two purposes. We will assume that you are familiar with the basic logic of statistical reasoning as described in *Elementary Concepts*. Moreover, we will also assume that you are familiar with the concepts of variance and correlation; if not, we advise that you read the *Basic Statistics* chapter at this point.

There are many excellent books on factor analysis. For example, a hands-on how-to approach can be found in Stevens (1986); more detailed technical descriptions are provided in Cooley and Lohnes (1971); Harman (1976); Kim and Mueller, (1978a, 1978b); Lawley and Maxwell (1971); Lindeman, Merenda, and Gold (1980); Morrison (1967); or Mulaik (1972). The interpretation of secondary factors in hierarchical factor analysis, as an alternative to traditional oblique rotational strategies, is explained in detail by Wherry (1984).

Confirmatory factor analysis. *Structural Equation Modeling (SEPATH)* allows you to test specific hypotheses about the factor structure for a set of variables, in one or several samples (e.g., you can compare factor structures across samples).

Correspondence analysis. Correspondence analysis is a descriptive/exploratory technique designed to analyze two-way and multi-way tables containing some measure of correspondence between the rows and columns. The results provide information which is similar in nature to those produced by factor analysis techniques, and they allow one to explore the structure of categorical variables included in the table. For more information regarding these methods, refer to *Correspondence Analysis*.

Basic Idea of Factor Analysis as a Data Reduction Method

Suppose we conducted a (rather "silly") study in which we measure 100 people's height in inches and centimeters. Thus, we would have two variables that measure height. If in future studies, we want to research, for example, the effect of different nutritional food supplements on height, would we continue to use both measures? Probably not; height is one characteristic of a person, regardless of how it is measured.

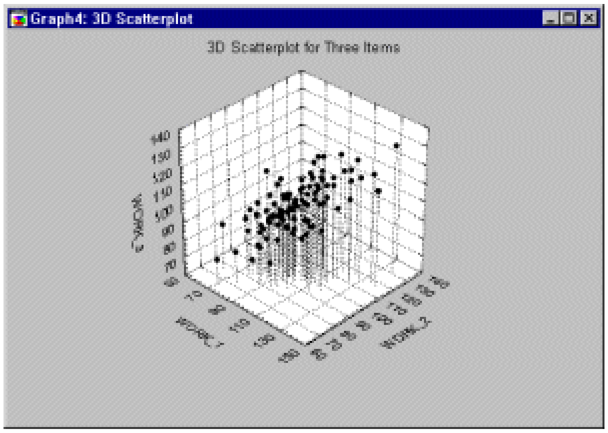
Let us now extrapolate from this "silly" study to something that one might actually do as a researcher. Suppose we want to measure people's satisfaction with their lives. We design a satisfaction questionnaire with various items; among other things we ask our subjects how satisfied they are with their hobbies (item 1) and how intensely they are pursuing a hobby (item 2). Most likely, the responses to the two items are highly correlated with each other. (If you are not familiar with the correlation coefficient, we recommend that you read the description in Basic Statistics - Correlations) Given a high correlation between the two items, we can conclude that they are quite redundant.

Combining Two Variables into a Single Factor. One can summarize the correlation between two variables in a scatterplot. A regression line can then be fitted that represents the "best" summary of the linear relationship between the variables. If we could define a variable that would approximate the regression line in such a plot, then that variable would capture most of the "essence" of the two items. Subjects' single scores on that new factor, represented by the regression line, could then be used in future data analyses to represent that essence of the two items. In a sense we have reduced the two variables to one factor. Note that the new factor is actually a linear combination of the two variables.

Principal Components Analysis. The example described above, combining two correlated variables into one factor, illustrates the basic idea of factor analysis, or of principal components analysis to be precise (we will return to this later). If we extend the two-variable example to multiple variables, then the computations become more involved, but the basic principle of expressing two or more variables by a single factor remains the same.

Extracting Principal Components. We do not want to go into the details about the computational aspects of principal components analysis here, which can be found elsewhere (references were provided at the beginning of this section). However, basically, the extraction of principal components amounts to a *variance maximizing (varimax) rotation* of the original variable space. For example, in a scatterplot we can think of the regression line as the original *X* axis, rotated so that it approximates the regression line. This type of rotation is called *variance maximizing* because the criterion for (goal of) the rotation is to maximize the variance (variability) of the "new" variable (factor), while minimizing the variance around the new variable (see *Rotational Strategies*).

Generalizing to the Case of Multiple Variables. When there are more than two variables, we can think of them as defining a "space," just as two variables defined a plane. Thus, when we have three variables, we could plot a three- dimensional scatterplot, and, again we could fit a plane through the data.

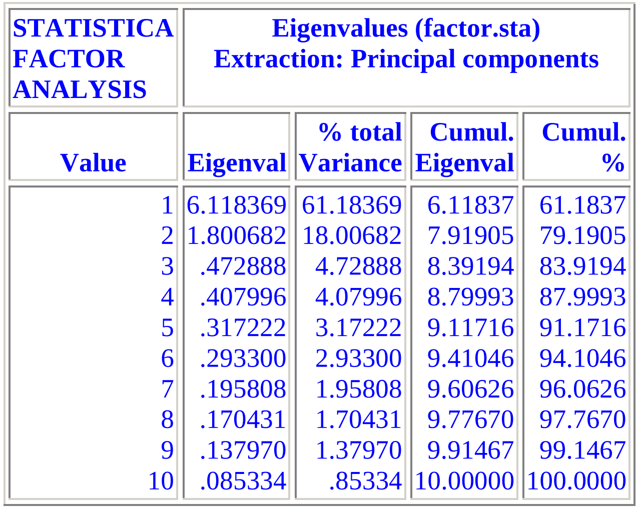


With more than three variables it becomes impossible to illustrate the points in a scatterplot, however, the logic of rotating the axes so as to maximize the variance of the new factor remains the same.

Multiple orthogonal factors. After we have found the line on which the variance is maximal, there remains some variability around this line. In principal components analysis, after the first factor has been extracted, that is, after the first line has been drawn through the data, we continue and define another line that maximizes the remaining variability, and so on. In this manner, consecutive factors are extracted. Because each consecutive factor is defined to maximize the variability that is not captured by the preceding factor, consecutive factors are independent of each other. Put another way, consecutive factors are uncorrelated or *orthogonal* to each other.

How many Factors to Extract? Remember that, so far, we are considering principal components analysis as a data reduction method, that is, as a method for reducing the number of variables. The question then is, how many factors do we want to extract? Note that as we extract consecutive factors, they account for less and less variability. The decision of when to stop extracting factors basically depends on when there is only very little "random" variability left. The nature of this decision is arbitrary; however, various guidelines have been developed, and they are reviewed in *Reviewing the Results of a Principal Components Analysis* under *Eigenvalues and the Number-of- Factors Problem*.

Reviewing the Results of a Principal Components Analysis. Without further ado, let us now look at some of the standard results from a principal components analysis. To reiterate, we are extracting factors that account for less and less variance. To simplify matters, one usually starts with the correlation matrix, where the variances of all variables are equal to 1.0. Therefore, the total variance in that matrix is equal to the number of variables. For example, if we have 10 variables each with a variance of 1 then the total variability that can potentially be extracted is equal to 10 times 1. Suppose that in the satisfaction study introduced earlier we included 10 items to measure different aspects of satisfaction at home and at work. The variance accounted for by successive factors would be summarized as follows:



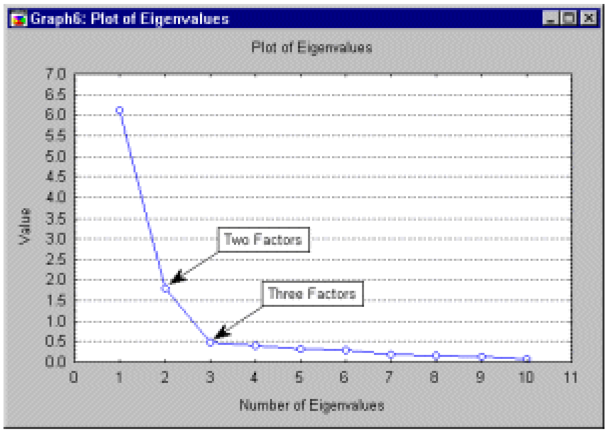
STATISTICA FACTOR ANALYSIS	Eigenvalues (factor.sta) Extraction: Principal components			
Value	Eigenval	% total Variance	Cumul. Eigenval	Cumul. %
1	6.118369	61.18369	6.11837	61.1837
2	1.800682	18.00682	7.91905	79.1905
3	.472888	4.72888	8.39194	83.9194
4	.407996	4.07996	8.79993	87.9993
5	.317222	3.17222	9.11716	91.1716
6	.293300	2.93300	9.41046	94.1046
7	.195808	1.95808	9.60626	96.0626
8	.170431	1.70431	9.77670	97.7670
9	.137970	1.37970	9.91467	99.1467
10	.085334	.85334	10.00000	100.0000

Eigenvalues
In the second column (*Eigenvalue*) above, we find the variance on the new factors that were successively extracted. In the third column, these values are expressed as a percent of the total variance (in this example, 10). As we can see, factor 1 accounts for 61 percent of the variance, factor 2 for 18 percent, and so on. As expected, the sum of the eigenvalues is equal to the number of variables. The third column contains the cumulative variance extracted. The variances extracted by the factors are called the *eigenvalues*. This name derives from the computational issues involved.

Eigenvalues and the Number-of-Factors Problem
Now that we have a measure of how much variance each successive factor extracts, we can return to the question of how many factors to retain. As mentioned earlier, by its nature this is an arbitrary decision. However, there are some guidelines that are commonly used, and that, in practice, seem to yield the best results.

The Kaiser criterion. First, we can retain only factors with eigenvalues greater than 1. In essence this is like saying that, unless a factor extracts at least as much as the equivalent of one original variable, we drop it. This criterion was proposed by Kaiser (1960), and is probably the one most widely used. In our example above, using this criterion, we would retain 2 factors (principal components).

The scree test. A graphical method is the *scree* test first proposed by Cattell (1966). We can plot the eigenvalues shown above in a simple line plot.



Cattell suggests to find the place where the smooth decrease of eigenvalues appears to level off to the right of the plot. To the right of this point, presumably, one finds only "factorial scree" -- "scree" is the geological term referring to the debris which collects on the lower part of a rocky slope. According to this criterion, we would probably retain 2 or 3 factors in our example.

Which criterion to use. Both criteria have been studied in detail (Browne, 1968; Cattell & Jaspers, 1967; Hakstian, Rogers, & Cattell, 1982; Linn, 1968; Tucker, Koopman & Linn, 1969). Theoretically, one can evaluate those criteria by generating random data based on a particular number of factors. One can then see whether the number of factors is accurately detected by those criteria. Using this general technique, the first method (*Kaiser criterion*) sometimes retains too many factors, while the second technique (scree test) sometimes retains too few; however, both do quite well under normal conditions, that is, when there are relatively few factors and many cases. In practice, an additional important aspect is the extent to which a solution is interpretable. Therefore, one usually examines several solutions with more or fewer factors, and chooses the one that makes the best "sense." We will discuss this issue in the context of factor rotations below.

Principal Factors Analysis

Before we continue to examine the different aspects of the typical output from a principal components analysis, let us now introduce principal factors analysis. Let us return to our satisfaction questionnaire example to conceive of another "mental model" for factor analysis. We can think of subjects' responses as being dependent on two components. First, there are some underlying common factors, such as the "satisfaction-with-hobbies" factor we looked at before. Each item measures some part of this common aspect of satisfaction. Second, each item also captures a unique aspect of satisfaction that is not addressed by any other item.

Communalities. If this model is correct, then we should not expect that the factors will extract all variance from our items; rather, only that proportion that is due to the common factors and shared by several items. In the language of factor analysis, the proportion of variance of a particular item that is due to common factors (shared with other items) is called *communality*. Therefore, an additional task facing us when applying this model is to estimate the communalities for each variable, that is, the proportion of variance that each item has in common with other items. The proportion of variance that is unique to each item is then the respective item's total variance minus the communality. A common starting point is to use the squared multiple correlation of an item with all other items as an estimate of the communality (refer to *Multiple Regression* for details about multiple regression). Some authors have suggested various iterative "post-solution improvements" to the initial multiple regression communality estimate; for example, the so-called MINRES method (minimum residual factor method; Harman & Jones, 1966) will try various modifications to the factor loadings with the goal to minimize the residual (unexplained) sums of squares.

Principal factors vs. principal components. The defining characteristic then that distinguishes between the two factor analytic models is that in principal components analysis we assume that *all* variability in an item should be used in the analysis, while in principal factors analysis we only use the variability in an item that it has in common with the other items. A detailed discussion of the pros and cons of each approach is beyond the scope of this introduction (refer to the general references provided in *Principal components and Factor Analysis - Introductory Overview*). In most cases, these two methods usually yield very similar results. However, principal components analysis is often preferred as a method for data reduction, while principal factors analysis is often preferred when the goal of the analysis is to detect structure (see *Factor Analysis as a Classification Method*).

Factor Analysis as a Classification Method

Let us now return to the interpretation of the standard results from a factor analysis. We will henceforth use the term *factor analysis* generically to encompass both principal components and principal factors analysis. Let us assume that we are at the point in our analysis where we basically know how many factors to extract. We may now want to know the meaning of the factors, that is, whether and how we can interpret them in a meaningful manner. To illustrate how this can be accomplished, let us work "backwards," that is, begin with a meaningful structure and then see how it is reflected in the results of a factor analysis. Let us return to our satisfaction example; shown below is the correlation matrix for items pertaining to satisfaction at work and items pertaining to satisfaction at home.

STATISTICA FACTOR ANALYSIS	**Correlations (factor.sta) Casewise deletion of MD n=100**					
Variable	**WORK_1**	**WORK_2**	**WORK_3**	**HOME_1**	**HOME_2**	**HOME_3**
WORK_1	1.00	.65	.65	.14	.15	.14
WORK_2	.65	1.00	.73	.14	.18	.24
WORK_3	.65	.73	1.00	.16	.24	.25
HOME_1	.14	.14	.16	1.00	.66	.59

HOME_2	.15	.18	.24	.66	1.00	.73
HOME_3	.14	.24	.25	.59	.73	1.00

The work satisfaction items are highly correlated amongst themselves, and the home satisfaction items are highly intercorrelated amongst themselves. The correlations across these two types of items (work satisfaction items with home satisfaction items) is comparatively small. It thus seems that there are two relatively independent factors reflected in the correlation matrix, one related to satisfaction at work, the other related to satisfaction at home.

Factor Loadings. Let us now perform a principal components analysis and look at the two-factor solution. Specifically, let us look at the correlations between the variables and the two factors (or "new" variables), as they are extracted by default; these correlations are also called factor *loadings*.

STATISTICA FACTOR ANALYSIS	**Factor Loadings (Unrotated) Principal components**	
Variable	**Factor 1**	**Factor 2**
WORK_1	.654384	.564143
WORK_2	.715256	.541444
WORK_3	.741688	.508212
HOME_1	.634120	-.563123
HOME_2	.706267	-.572658
HOME_3	.707446	-.525602
Expl.Var	2.891313	1.791000
Prp.Totl	.481885	.298500

Apparently, the first factor is generally more highly correlated with the variables than the second factor. This is to be expected because, as previously described, these factors are extracted successively and will account for less and less variance overall.

Rotating the Factor Structure. We could plot the factor loadings shown above in a scatterplot. In that plot, each variable is represented as a point. In this plot we could rotate the axes in any direction without changing the *relative* locations of the points to each other; however, the actual coordinates of the points, that is, the factor loadings would of course change. In this example, if you produce the plot it will be evident that if we were to rotate the axes by about 45 degrees we might attain a clear pattern of loadings identifying the work satisfaction items and the home satisfaction items.

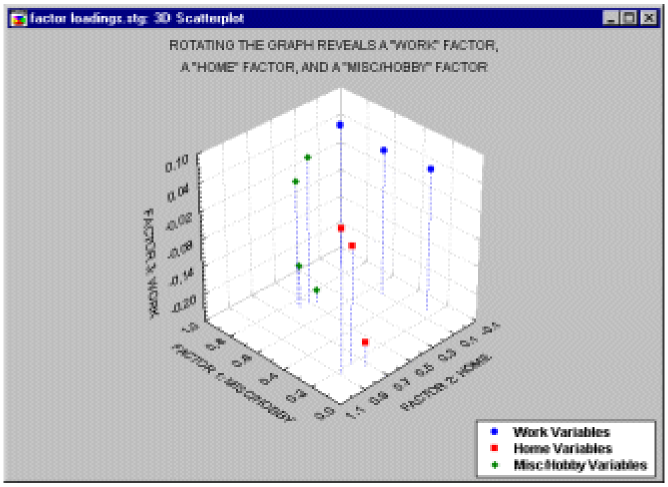
Rotational strategies. There are various rotational strategies that have been proposed. The goal of all of these strategies is to obtain a clear pattern of loadings, that is, factors that are somehow clearly marked by high loadings for some variables and low loadings for others. This general pattern is also sometimes referred to as *simple structure* (a more formalized definition can be found in most standard textbooks). Typical rotational strategies are *varimax*, *quartimax*, and *equamax*.

We have described the idea of the varimax rotation before (see Extracting Principal Components), and it can be applied to this problem as well. As before, we want to find a rotation that maximizes the variance on the new axes; put another way, we want to obtain a pattern of loadings on each factor that is as diverse as possible, lending itself to easier interpretation. Below is the table of rotated factor loadings.

STATISTICA FACTOR ANALYSIS	**Factor Loadings (Varimax normalized)** **Extraction: Principal components**	
Variable	**Factor 1**	**Factor 2**
WORK_1	.862443	.051643
WORK_2	.890267	.110351
WORK_3	.886055	.152603
HOME_1	.062145	.845786
HOME_2	.107230	.902913
HOME_3	.140876	.869995
Expl.Var	2.356684	2.325629
Prp.Totl	.392781	.387605

Interpreting the Factor Structure. Now the pattern is much clearer. As expected, the first factor is marked by high loadings on the work satisfaction items, the second factor is marked by high loadings on the home satisfaction items. We would thus conclude that satisfaction, as measured by our questionnaire, is composed of those two aspects; hence we have arrived at a *classification* of the variables.

Consider another example, this time with four additional Hobby/Misc variables added to our earlier example.



In the plot of factor loadings above, 10 variables were reduced to three specific factors, a work factor, a home factor and a hobby/misc. factor. Note that factor loadings for each factor are spread out over the values of the other two factors but are high for its own values. For example, the factor loadings for the hobby/misc variables (in green) have both high and low "work" and "home" values, but all four of these variables have high factor loadings on the "hobby/misc" factor.

Oblique Factors. Some authors (e.g., Catell & Khanna; Harman, 1976; Jennrich & Sampson, 1966; Clarkson & Jennrich, 1988) have discussed in some detail the concept of *oblique* (non-orthogonal) factors, in order to achieve more interpretable simple structure. Specifically, computational strategies have been developed to rotate factors so as to best represent "clusters" of variables, without the constraint of orthogonality of factors. However, the oblique factors produced by such rotations are often not easily interpreted. To return to the example discussed above, suppose we would have included in the satisfaction questionnaire above four items that measured other, "miscellaneous" types of satisfaction. Let us assume that people's responses to those items were affected about equally by their satisfaction at home (*Factor 1*) and at work (*Factor 2*). An oblique rotation will likely produce two correlated factors with less-than- obvious meaning, that is, with many cross-loadings.

Hierarchical Factor Analysis. Instead of computing loadings for often difficult to interpret oblique factors, you can use a strategy first proposed by Thompson (1951) and Schmid and Leiman (1957), which has been elaborated and popularized in the detailed discussions by Wherry (1959, 1975, 1984). In this strategy, you first identify clusters of items and rotate axes through those clusters; next the correlations between those (oblique) factors is computed, and that correlation matrix of oblique factors is further factor-analyzed to yield a set of orthogonal factors that divide the variability in the items into that due to shared or common variance (secondary factors), and unique variance due to the clusters of similar variables (items) in the analysis (primary factors). To return to the example above, such a hierarchical analysis might yield the following factor loadings:

STATISTICA FACTOR ANALYSIS	**Secondary & Primary Factor Loadings**		
Factor	**Second. 1**	**Primary 1**	**Primary 2**
WORK_1	.483178	.649499	.187074
WORK_2	.570953	.687056	.140627
WORK_3	.565624	.656790	.115461
HOME_1	.535812	.117278	.630076
HOME_2	.615403	.079910	.668880
HOME_3	.586405	.065512	.626730
MISCEL_1	.780488	.466823	.280141
MISCEL_2	.734854	.464779	.238512
MISCEL_3	.776013	.439010	.303672
MISCEL_4	.714183	.455157	.228351

Careful examination of these loadings would lead to the following conclusions:

1. There is a general (secondary) satisfaction factor that likely affects all types of satisfaction measured by the 10 items;
2. There appear to be two primary unique areas of satisfaction that can best be described as satisfaction with work and satisfaction with home life.

Wherry (1984) discusses in great detail examples of such hierarchical analyses, and how meaningful and interpretable secondary factors can be derived.

Confirmatory Factor Analysis. Over the past 15 years, so-called confirmatory methods have become increasingly popular (e.g., see Jöreskog and Sörbom, 1979). In general, one can specify *a priori*, a pattern of factor loadings for a particular number of orthogonal or oblique factors, and then test whether the observed correlation matrix can be reproduced given these specifications. Confirmatory factor analyses can be performed via *Structural Equation Modeling (SEPATH)*.

Miscellaneous Other Issues and Statistics

Factor Scores. We can estimate the actual values of individual cases (observations) for the factors. These factor scores are particularly useful when one wants to perform further analyses involving the factors that one has identified in the factor analysis.

Reproduced and Residual Correlations. An additional check for the appropriateness of the respective number of factors that were extracted is to compute the correlation matrix that would result if those were indeed the only factors. That matrix is called the *reproduced* correlation matrix. To see how this matrix deviates from the observed correlation matrix, one can compute the difference between the two; that matrix is called the matrix of *residual* correlations. The residual matrix may point to "misfits," that is, to particular correlation coefficients that cannot be reproduced appropriately by the current number of factors.

Matrix Ill-conditioning. If, in the correlation matrix there are variables that are 100% redundant, then the inverse of the matrix cannot be computed. For example, if a variable is the sum of two other variables selected for the analysis, then the correlation matrix of those variables cannot be inverted, and the factor analysis can basically not be performed. In practice this happens when you are attempting to factor analyze a set of highly intercorrelated variables, as it, for example, sometimes occurs in correlational research with questionnaires. Then you can artificially lower all correlations in the correlation matrix by adding a small constant to the diagonal of the matrix, and then restandardizing it. This procedure will usually yield a matrix that now can be inverted and thus factor-analyzed; moreover, the factor patterns should not be affected by this procedure. However, note that the resulting estimates are not exact.

Chapter 17

General Discriminant Analysis (GDA)

- Introductory Overview
- Advantages of GDA

Introductory Overview

General Discriminant Analysis (GDA) is called a "general" discriminant analysis because it applies the methods of the general linear model (see also *General Linear Models (GLM)*) to the discriminant function analysis problem. A general overview of discriminant function analysis, and the traditional methods for fitting linear models with categorical dependent variables and continuous predictors, is provided in the context of *Discriminant Analysis*. In *GDA*, the discriminant function analysis problem is "recast" as a general multivariate linear model, where the dependent variables of interest are (dummy-) coded vectors that reflect the group membership of each case. The remainder of the analysis is then performed as described in the context of *General Regression Models (GRM)*, with a few additional features noted below.

Advantages of GDA

Specifying models for predictor variables and predictor effects. One advantage of applying the general linear model to the discriminant analysis problem is that you can specify complex models for the set of predictor variables. For example, you can specify for a set of continuous predictor variables, a polynomial regression model, response surface model, factorial regression, or mixture surface regression (without an intercept). Thus, you could analyze a constrained mixture experiment (where the predictor variable values must sum to a constant), where the dependent variable of interest is categorical in nature. In fact, *GDA* does not impose any particular restrictions on the type of predictor variable (categorical or continuous) that can be used, or the models that can be specified. However, when using categorical predictor variables, caution should be used (see "A note of caution for models with categorical predictors, and other advanced techniques" below).

Stepwise and best-subset analyses. In addition to the traditional stepwise analyses for single continuous predictors provided in *Discriminant Analysis*, *General Discriminant Analysis* makes available the options for stepwise and best-subset analyses provided in *General Regression Models (GRM)*. Specifically, you can request stepwise and best-subset selection of predictors or sets of predictors (in multiple-degree of freedom effects, involving categorical predictors), based on the *F-to-enter* and *p-to-enter* statistics (associated with the multivariate Wilks' *Lambda* test statistic). In addition, when a cross-validation sample is specified, best-subset selection can also be based on the misclassification rates for the cross-validation sample; in other words, after estimating the discriminant functions for a given set of predictors, the misclassification rates for the cross-validation sample are computed, and the model (subset of predictors) that yields the lowest misclassification rate for the cross-validation sample is chosen. This is a powerful technique for choosing models that may yield good predictive validity, while avoiding overfitting of the data (see also *Neural Networks*).

Desirability profiling of posterior classification probabilities. Another unique option of *General Discriminant Analysis (GDA)* is the inclusion of *Response/desirability profiler* options. These options are described in some detail in the context of *Experimental Design (DOE)*. In short, the predicted response values for each dependent variable are computed, and those values can be combined into a single desirability score. A graphical summary can then be produced to show the "behavior" of the predicted responses and the desirability score over the ranges of values for the predictor variables. In *GDA*, you can profile both simple predicted values (like in *General Regression Models*) for the coded dependent variables (i.e., dummy-coded categories of the categorical dependent variable), and you can also profile posterior prediction probabilities. This unique latter option allows you to evaluate how different values for the predictor variables affect the predicted classification of cases, and is particularly useful when interpreting the results for complex models that involve categorical and continuous predictors and their interactions.

A note of caution for models with categorical predictors, and other advanced techniques. General Discriminant Analysis provides functionality that makes this technique a general tool for classification and data mining. However, most -- if not all -- textbook treatments of discriminant function analysis are limited to simple and stepwise analyses with single degree of freedom continuous predictors. No "experience" (in the literature) exists regarding issues of robustness and effectiveness of these techniques, when they are generalized in the manner provided in this very powerful analysis. The use of best-subset methods, in particular when used in conjunction with categorical predictors or when using the misclassification rates in a cross-validation sample for choosing the best subset of predictors, should be considered a heuristic search method, rather than a statistical analysis technique.

The use of categorical predictor variables. The use of categorical predictor variables or effects in a discriminant function analysis model may be (statistically) questionable. For example, you can use *GDA* to analyze a 2 by 2 frequency table, by specifying one variable in the 2 by 2 table as the dependent variable, and the other as the predictor. Clearly, the (ab)use of *GDA* in this manner would be silly (although, interestingly, in most cases you will get results that are generally compatible with those you would get by computing a simple *Chi*-square test for the 2 by 2 table). On the other hand, if you only consider the parameter estimates computed by *GDA* as the least squares solution to a set of linear (prediction) equations, then the use of categorical predictors in *GDA* is fully justified; moreover, it is not uncommon in applied research to be confronted with a mixture of continuous and categorical predictors (e.g., income or age which are continuous, along with occupational status, which is categorical) for predicting a categorical dependent variable. In those cases, it can be very instructive to consider specific models involving the categorical predictors, and possibly interactions between categorical and continuous predictors for classifying observations. However, to reiterate, the use of categorical predictor variables in discriminant function analysis is not widely documented, and you should proceed cautiously before accepting the results of statistical significance tests, and before drawing final conclusions from your analyses. Also remember that there are alternative methods available to perform similar analyses, namely, the multinomial logit models available in *Generalized Linear Models (GLZ)*, and the methods for analyzing multi-way frequency tables in *Log-Linear*.

Chapter 18
General Linear Models (GLM)

- Basic Ideas: The General Linear Model
  - Historical background
  - The purpose of multiple regression
  - Computations for solving the multiple regression equation
  - Extension of multiple regression to the general linear model
  - The sigma-restricted vs. overparameterized model
  - Summary of computations
- Types of Analyses
  - Between-subject designs
  - Within-subject (repeated measures) designs
  - Multivariate designs
- Estimation and Hypothesis Testing
  - Whole model tests
  - Six types of sums of squares
  - Error terms for tests
  - Testing specific hypotheses
  - Testing hypotheses for repeated measures and dependent variables

This chapter describes the use of the general linear model in a wide variety of statistical analyses. If you are unfamiliar with the basic methods of ANOVA and regression in linear models, it may be useful to first review the basic information on these topics in *Elementary Concepts*. A detailed discussion of univariate and multivariate ANOVA techniques can also be found in the *ANOVA/MANOVA* chapter.

Basic Ideas: The General Linear Model

The following topics summarize the historical, mathematical, and computational foundations for the general linear model. For a basic introduction to ANOVA (MANOVA, ANCOVA) techniques, refer to *ANOVA/MANOVA*; for an introduction to multiple regression, see *Multiple Regression*; for an introduction to the design an analysis of experiments in applied (industrial) settings, see *Experimental Design*.

Historical Background

The roots of the general linear model surely go back to the origins of mathematical thought, but it is the emergence of the theory of algebraic invariants in the 1800's that made the general linear model, as we know it today, possible. The theory of algebraic invariants developed from the groundbreaking work of 19th century mathematicians such as Gauss, Boole, Cayley, and Sylvester. The theory seeks to identify those quantities in systems of equations which remain unchanged under linear transformations of the variables in the system. Stated more imaginatively (but in a way in which the originators of the theory would *not* consider an overstatement), the theory of algebraic invariants searches for the eternal and unchanging amongst the chaos of the transitory and the illusory. That is no small goal for any theory, mathematical or otherwise.

The wonder of it all is the theory of algebraic invariants was successful far beyond the hopes of its originators. Eigenvalues, eigenvectors, determinants, matrix decomposition methods; all derive from the theory of algebraic invariants. The contributions of the theory of algebraic invariants to the development of statistical theory and methods are numerous, but a simple example familiar to even the most casual student of statistics is illustrative. The correlation between two variables is unchanged by linear transformations of either or both variables. We probably take this property of correlation coefficients for granted, but what would data analysis be like if we did not have statistics that are invariant to the scaling of the variables involved? Some thought on this question should convince you that without the theory of algebraic invariants, the development of useful statistical techniques would be nigh impossible.

The development of the linear regression model in the late 19th century, and the development of correlational methods shortly thereafter, are clearly direct outgrowths of the theory of algebraic invariants. Regression and correlational methods, in turn, serve as the basis for the general linear model. Indeed, the general linear model can be seen as an extension of linear multiple regression for a single dependent variable. Understanding the multiple regression model is fundamental to understanding the general linear model, so we will look at the purpose of multiple regression, the computational algorithms used to solve regression problems, and how the regression model is extended in the case of the general linear model. A basic introduction to multiple regression methods and the analytic problems to which they are applied is provided in the *Multiple Regression*.

The Purpose of Multiple Regression

The general linear model can be seen as an extension of linear multiple regression for a single dependent variable, and understanding the multiple regression model is fundamental to understanding the general linear model. The general purpose of multiple regression (the term was first used by Pearson, 1908) is to quantify the relationship between several independent or predictor variables and a dependent or criterion variable. For a detailed introduction to multiple regression, also refer to the *Multiple Regression* chapter. For example, a real estate agent might record for each listing the size of the house (in square feet), the number of bedrooms, the average income in the respective neighborhood according to census data, and a subjective rating of appeal of the house. Once this information has been compiled for various houses it would be interesting to see whether and how these measures relate to the price for which a house is sold. For example, one might learn that the number of bedrooms is a better predictor of the price for which a house sells in a particular neighborhood than how "pretty" the house is (subjective rating). One may also detect "outliers," for example, houses that should really sell for more, given their location and characteristics.

Personnel professionals customarily use multiple regression procedures to determine equitable compensation. One can determine a number of factors or dimensions such as "amount of responsibility" (*Resp*) or "number of people to supervise" (*No_Super*) that one believes to contribute to the value of a job. The personnel analyst then usually conducts a salary survey among comparable companies in the market, recording the salaries and respective characteristics (i.e., values on dimensions) for different positions. This information can be used in a multiple regression analysis to build a regression equation of the form:

Salary = .5*Resp + .8*No_Super

Once this so-called regression equation has been determined, the analyst can now easily construct a graph of the expected (predicted) salaries and the actual salaries of job incumbents in his or her company. Thus, the analyst is able to determine which position is underpaid (below the regression line) or overpaid (above the regression line), or paid equitably.

In the social and natural sciences multiple regression procedures are very widely used in research. In general, multiple regression allows the researcher to ask (and hopefully answer) the general question "what is the best predictor of ...". For example, educational researchers might want to learn what are the best predictors of success in high-school. Psychologists may want to determine which personality variable best predicts social adjustment. Sociologists may want to find out which of the multiple social indicators best predict whether or not a new immigrant group will adapt and be absorbed into society.

Computations for Solving the Multiple Regression Equation

A one dimensional surface in a two dimensional or two-variable space is a line defined by the equation $Y = b_0 + b_1X$. According to this equation, the Y variable can be expressed in terms of or as a function of a constant (b_0) and a slope (b_1) times the X variable. The constant is also referred to as the intercept, and the slope as the regression coefficient. For example, *GPA* may best be predicted as *1+.02*IQ*. Thus, knowing that a student has an *IQ* of 130 would lead us to predict that her *GPA* would be 3.6 (since, 1+.02*130=3.6). In the multiple regression case, when there are multiple predictor variables, the regression surface usually cannot be visualized in a two dimensional space, but the computations are a straightforward extension of the computations in the single predictor case. For example, if in addition to *IQ* we had additional predictors of achievement (e.g., *Motivation*, *Self-discipline*) we could construct a linear equation containing all those variables. In general then, multiple regression procedures will estimate a linear equation of the form:

$$Y = b_0 + b_1X_1 + b_2X_2 + ... + b_kX_k$$

where k is the number of predictors. Note that in this equation, the regression coefficients (or $b_1 \dots b_k$ coefficients) represent the *independent* contributions of each in dependent variable to the prediction of the dependent variable. Another way to express this fact is to say that, for example, variable X_1 is correlated with the Y variable, after controlling for all other independent variables. This type of correlation is also referred to as a *partial correlation* (this term was first used by Yule, 1907). Perhaps the following example will clarify this issue. One would probably find a significant negative correlation between hair length and height in the population (i.e., short people have longer hair). At first this may seem odd; however, if we were to add the variable Gender into the multiple regression equation, this correlation would probably disappear. This is because women, on the average, have longer hair than men; they also are shorter on the average than men. Thus, after we remove this gender difference by entering Gender into the equation, the relationship between hair length and height disappears because hair length does *not* make any unique contribution to the prediction of height, above and beyond what it shares in the prediction with variable Gender. Put another way, after controlling for the variable Gender, the *partial correlation* between hair length and height is zero.

The regression surface (a line in simple regression, a plane or higher-dimensional surface in multiple regression) expresses the best prediction of the dependent variable (Y), given the independent variables (X's). However, nature is rarely (if ever) perfectly predictable, and usually there is substantial variation of the observed points from the fitted regression surface. The deviation of a particular point from the nearest corresponding point on the predicted regression surface (its predicted value) is called the *residual* value. Since the goal of linear regression procedures is to fit a surface, which is a linear function of the X variables, as closely as possible to the observed Y variable, the *residual* values for the observed points can be used to devise a criterion for the "best fit." Specifically, in regression problems the surface is computed for which the sum of the squared deviations of the observed points from that surface are minimized. Thus, this general procedure is sometimes also referred to as *least squares* estimation. (see also the description of *weighted least squares* estimation).

The actual computations involved in solving regression problems can be expressed compactly and conveniently using matrix notation. Suppose that there are n observed values of Y and n associated observed values for each of k different X variables. Then Y_i, X_{ik}, and e_i can represent the ith observation of the Y variable, the ith observation of each of the X variables, and the ith unknown residual value, respectively. Collecting these terms into matrices we have

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ \vdots \\ \vdots \\ \vdots \\ Y_n \end{bmatrix}, \mathbf{X} = \begin{bmatrix} 1 & X_{11} & \cdots & \cdots & \cdots & X_{1k} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & X_{n1} & \cdots & \cdots & \cdots & X_{nk} \end{bmatrix}, \mathbf{e} = \begin{bmatrix} e_1 \\ \vdots \\ \vdots \\ \vdots \\ e_n \end{bmatrix}$$

The multiple regression model in matrix notation then can be expressed as

$$\mathbf{Y = Xb + e}$$

where ***b*** is a column vector of 1 (for the intercept) + *k* unknown regression coefficients. Recall that the goal of multiple regression is to minimize the sum of the squared residuals. Regression coefficients that satisfy this criterion are found by solving the set of normal equations

$$\mathbf{X'Xb = X'Y}$$

When the *X* variables are linearly independent (i.e., they are nonredundant, yielding an ***X'X*** matrix which is of full rank) there is a unique solution to the normal equations. Premultiplying both sides of the matrix formula for the normal equations by the inverse of ***X'X*** gives

$$(\boldsymbol{X'X})^{-1}\mathbf{X'Xb} = \mathbf{(X'X)}^{-1}\mathbf{X'Y}$$

or

$$\mathbf{b = (X'X)}^{-1}\mathbf{X'Y}$$

This last result is very satisfying in view of its simplicity and its generality. With regard to its simplicity, it expresses the solution for the regression equation in terms just 2 matrices (***X*** and ***Y***) and 3 basic matrix operations, (1) matrix transposition, which involves interchanging the elements in the rows and columns of a matrix, (2) matrix multiplication, which involves finding the sum of the products of the elements for each row and column combination of two conformable (i.e., multipliable) matrices, and (3) matrix inversion, which involves finding the matrix equivalent of a numeric reciprocal, that is, the matrix that satisfies

$$\mathbf{A}^{-1}\mathbf{AA=A}$$

for a matrix ***A***.

It took literally centuries for the ablest mathematicians and statisticians to find a satisfactory method for solving the linear least square regression problem. But their efforts have paid off, for it is hard to imagine a simpler solution.

With regard to the generality of the multiple regression model, its only notable limitations are that (1) it can be used to analyze only a single dependent variable, (2) it cannot provide a solution for the regression coefficients when the *X* variables are not linearly independent and the inverse of **X'X** therefore does not exist. These restrictions, however, can be overcome, and in doing so the multiple regression model is transformed into the general linear model.

Extension of Multiple Regression to the General Linear Model

One way in which the general linear model differs from the multiple regression model is in terms of the number of dependent variables that can be analyzed. The ***Y*** vector of *n* observations of a single *Y* variable can be replaced by a ***Y*** matrix of *n* observations of *m* different *Y* variables. Similarly, the ***b*** vector of regression coefficients for a single ***Y*** variable can be replaced by a ***b*** matrix of regression coefficients, with one vector of ***b*** coefficients for each of the *m* dependent variables. These substitutions yield what is sometimes called the multivariate regression model, but it should be emphasized that the matrix formulations of the multiple and multivariate regression models are identical, except for the number of columns in the ***Y*** and ***b*** matrices. The method for solving for the ***b*** coefficients is also identical, that is, *m* different sets of regression coefficients are separately found for the *m* different dependent variables in the multivariate regression model.

The general linear model goes a step beyond the multivariate regression model by allowing for linear transformations or linear combinations of multiple dependent variables. This extension gives the general linear model important advantages over the multiple and the so-called multivariate regression models, both of which are inherently univariate (single dependent variable) methods. One advantage is that multivariate tests of significance can be employed when responses on multiple dependent variables are correlated. Separate univariate tests of significance for correlated dependent variables are not independent and may not be appropriate. Multivariate tests of significance of independent linear combinations of multiple dependent variables also can give insight into which dimensions of the response variables are, and are not, related to the predictor variables. Another advantage is the ability to analyze effects of repeated measure factors. Repeated measure designs, or within-subject designs, have traditionally been analyzed using ANOVA techniques. Linear combinations of responses reflecting a repeated measure effect (for example, the difference of responses on a measure under differing conditions) can be constructed and tested for significance using either the univariate or multivariate approach to analyzing repeated measures in the general linear model.

A second important way in which the general linear model differs from the multiple regression model is in its ability to provide a solution for the normal equations when the *X* variables are not linearly independent and the inverse of ***X'X*** does not exist. Redundancy of the *X* variables may be incidental (e.g., two predictor variables might happen to be perfectly correlated in a small data set), accidental (e.g., two copies of the same variable might unintentionally be used in an analysis) or designed (e.g., indicator variables with exactly opposite values might be used in the analysis, as when both *Male* and *Female* predictor variables are used in representing *Gender*). Finding the regular inverse of a non-full-rank matrix is reminiscent of the problem of finding the reciprocal of 0 in ordinary arithmetic. No such inverse or reciprocal exists because division by 0 is not permitted. This problem is solved in the general linear model by using a generalized inverse of the ***X'X*** matrix in solving the normal equations. A generalized inverse is any matrix that satisfies

$$\mathbf{AA^{-}A = A}$$

for a matrix ***A***.

A generalized inverse is unique and is the same as the regular inverse only if the matrix ***A*** is full rank. A generalized inverse for a non-full-rank matrix can be computed by the simple expedient of zeroing the elements in redundant rows and columns of the matrix. Suppose that an ***X'X*** matrix with *r* non-redundant columns is partitioned as

$$X'X = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

where $\boldsymbol{A_{11}}$ is an *r* by *r* matrix of rank *r*. Then the regular inverse of $\boldsymbol{A_{11}}$ exists and a generalized inverse of ***X'X*** is

$$(X'X)^{-} = \begin{bmatrix} A_{11}^{-1} & 0_{12} \\ 0_{21} & 0_{22} \end{bmatrix}$$

where each ***0*** (null) matrix is a matrix of 0's (zeroes) and has the same dimensions as the corresponding ***A*** matrix.

In practice, however, a particular generalized inverse of ***X'X*** for finding a solution to the normal equations is usually computed using the sweep operator (Dempster, 1960). This generalized inverse, called a g2 inverse, has two important properties. One is that zeroing of the elements in redundant rows is unnecessary. Another is that partitioning or reordering of the columns of ***X'X*** is unnecessary, so that the matrix can be inverted "in place."

There are infinitely many generalized inverses of a non-full-rank ***X'X*** matrix, and thus, infinitely many solutions to the normal equations. This can make it difficult to understand the nature of the relationships of the predictor variables to responses on the dependent variables, because the regression coefficients can change depending on the particular generalized inverse chosen for solving the normal equations. It is not cause for dismay, however, because of the invariance properties of many results obtained using the general linear model.

A simple example may be useful for illustrating one of the most important invariance properties of the use of generalized inverses in the general linear model. If both *Male* and *Female* predictor variables with exactly opposite values are used in an analysis to represent *Gender*, it is essentially arbitrary as to which predictor variable is considered to be redundant (e.g., *Male* can be considered to be redundant with *Female*, or vice versa). No matter which predictor variable is considered to be redundant, no matter which corresponding generalized inverse is used in solving the normal equations, and no matter which resulting regression equation is used for computing predicted values on the dependent variables, the predicted values and the corresponding residuals for males and females will be unchanged. In using the general linear model, one must keep in mind that finding a particular arbitrary solution to the normal equations is primarily a means to the end of accounting for responses on the dependent variables, and not necessarily an end in itself.

Sigma-Restricted and Overparameterized Model

Unlike the multiple regression model, which is usually applied to cases where the *X* variables are continuous, the general linear model is frequently applied to analyze any ANOVA or MANOVA design with categorical predictor variables, any ANCOVA or MANCOVA design with both categorical and continuous predictor variables, as well as any multiple or multivariate regression design with continuous predictor variables. To illustrate, *Gender* is clearly a nominal level variable (anyone who attempts to rank order the sexes on any dimension does so at his or her own peril in today's world). There are two basic methods by which *Gender* can be coded into one or more (non-offensive) predictor variables, and analyzed using the general linear model.

Sigma-restricted model (coding of categorical predictors). Using the first method, males and females can be assigned any two arbitrary, but distinct values on a single predictor variable. The values on the resulting predictor variable will represent a quantitative contrast between males and females. Typically, the values corresponding to group membership are chosen not arbitrarily but rather to facilitate interpretation of the regression coefficient associated with the predictor variable. In one widely used strategy, cases in the two groups are assigned values of 1 and -1 on the predictor variable, so that if the regression coefficient for the variable is positive, the group coded as 1 on the predictor variable will have a higher predicted value (i.e., a higher group mean) on the dependent variable, and if the regression coefficient is negative, the group coded as -1 on the predictor variable will have a higher predicted value on the dependent variable. An additional advantage is that since each group is coded with a value one unit from zero, this helps in interpreting the magnitude of differences in predicted values between groups, because regression coefficients reflect the units of change in the dependent variable for each unit change in the predictor variable. This coding strategy is aptly called the sigma-restricted parameterization, because the values used to represent group membership (1 and -1) sum to zero.

Note that the sigma-restricted parameterization of categorical predictor variables usually leads to ***X'X*** matrices which do not require a generalized inverse for solving the normal equations. Potentially redundant information, such as the characteristics of maleness and femaleness, is literally reduced to full-rank by creating quantitative contrast variables representing differences in characteristics.

Overparameterized model (coding of categorical predictors). The second basic method for recoding categorical predictors is the indicator variable approach. In this method a separate predictor variable is coded for each group identified by a categorical predictor variable. To illustrate, females might be assigned a value of 1 and males a value of 0 on a first predictor variable identifying membership in the female *Gender* group, and males would then be assigned a value of 1 and females a value of 0 on a second predictor variable identifying membership in the male *Gender* group. Note that this method of recoding categorical predictor variables will almost always lead to ***X'X*** matrices with redundant columns, and thus require a generalized inverse for solving the normal equations. As such, this method is often called the overparameterized model for representing categorical predictor variables, because it results in more columns in the ***X'X*** than are necessary for determining the relationships of categorical predictor variables to responses on the dependent variables.

True to its description as general, the general linear model can be used to perform analyses with categorical predictor variables which are coded using either of the two basic methods that have been described.

Summary of Computations

To conclude this discussion of the ways in which the general linear model extends and generalizes regression methods, the general linear model can be expressed as

$$\mathbf{YM = Xb + e}$$

Here ***Y***, ***X***, ***b***, and ***e*** are as described for the multivariate regression model and ***M*** is an *m* x *s* matrix of coefficients defining *s* linear transformation of the dependent variables. The normal equations are

$$\mathbf{X'Xb = X'YM}$$

and a solution for the normal equations is given by

$$\mathbf{b = (X'X)^{-}X'YM}$$

Here the inverse of ***X'X*** is a generalized inverse if ***X'X*** contains redundant columns.

Add a provision for analyzing linear combinations of multiple dependent variables, add a method for dealing with redundant predictor variables and recoded categorical predictor variables, and the major limitations of multiple regression are overcome by the general linear model.

Types of Analyses

A wide variety of types of designs can be analyzed using the general linear model. In fact, the flexibility of the general linear model allows it to handle so many different types of designs that it is difficult to develop simple typologies of the ways in which these designs might differ. Some general ways in which designs might differ can be suggested, but keep in mind that any particular design can be a "hybrid" in the sense that it could have combinations of features of a number of different types of designs.

In the following discussion, references will be made to the *design matrix* ***X***, as well as sigma-restricted and overparameterized model coding. For an explanation of this terminology, refer to the section entitled *Basic Ideas: The General Linear Model*, or, for a brief summary, to the *Summary of computations* section.

A basic discussion to univariate and multivariate ANOVA techniques can also be found in the *ANOVA/MANOVA* chapter; a discussion of mutiple regression methods is also provided in the *Multiple Regression* chapter.

Between-Subject Designs

- Overview
- One-way ANOVA
- Main effect ANOVA
- Factorial ANOVA
- Nested designs
- Balanced ANOVA
- Simple regression
- Multiple regression
- Factorial regression
- Polynomial regression
- Response surface regression
- Mixture surface regression
- Analysis of covariance (ANCOVA)
- Separate slopes designs
- Homogeneity of slopes
- Mixed-model ANOVA and ANCOVA

Overview. The levels or values of the predictor variables in an analysis describe the differences between the *n* subjects or the *n* valid cases that are analyzed. Thus, when we speak of the between subject design (or simply the between design) for an analysis, we are referring to the nature, number, and arrangement of the predictor variables.

Concerning the nature or type of predictor variables, between designs which contain only categorical predictor variables can be called ANOVA (analysis of variance) designs, between designs which contain only continuous predictor variables can be called regression designs, and between designs which contain both categorical and continuous predictor variables can be called ANCOVA (analysis of covariance) designs. Further, continuous predictors are always considered to have fixed values, but the levels of categorical predictors can be considered to be fixed or to vary randomly. Designs which contain random categorical factors are called mixed-model designs (see the *Variance Components and Mixed Model ANOVA/ANCOVA* chapter).

Between designs may involve only a single predictor variable and therefore be described as simple (e.g., simple regression) or may employ numerous predictor variables (e.g., multiple regression).

Concerning the arrangement of predictor variables, some between designs employ only "main effect" or first-order terms for predictors, that is, the values for different predictor variables are independent and raised only to the first power. Other between designs may employ higher-order terms for predictors by raising the values for the original predictor variables to a power greater than 1 (e.g., in polynomial regression designs), or by forming products of different predictor variables (i.e., interaction terms). A common arrangement for ANOVA designs is the full-factorial design, in which every combination of levels for each of the categorical predictor variables is represented in the design. Designs with some but not all combinations of levels for each of the categorical predictor variables are aptly called fractional factorial designs. Designs with a hierarchy of combinations of levels for the different categorical predictor variables are called nested designs.

These basic distinctions about the nature, number, and arrangement of predictor variables can be used in describing a variety of different types of between designs. Some of the more common between designs can now be described.

One-Way ANOVA. A design with a single categorical predictor variable is called a one-way ANOVA design. For example, a study of 4 different fertilizers used on different individual plants could be analyzed via one-way ANOVA, with four levels for the factor *Fertilizer*.

In genera, consider a single categorical predictor variable *A* with 1 case in each of its 3 categories. Using the sigma-restricted coding of A into 2 quantitative contrast variables, the matrix ***X*** defining the between design is

$$\mathbf{X} = \begin{array}{c} \\ A_1 \\ A_2 \\ A_3 \end{array} \begin{array}{c} \begin{array}{ccc} X_0 & X_1 & X_2 \end{array} \\ \left[\begin{array}{ccc} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & -1 & -1 \end{array}\right] \end{array}$$

That is, cases in groups A_1, A_2, and A_3 are all assigned values of 1 on X_0 (the intercept), the case in group A_1 is assigned a value of 1 on X_1 and a value 0 on X_2, the case in group

A_2 is assigned a value of 0 on X_1 and a value 1 on X_2, and the case in group A_3 is assigned a value of -1 on X_1 and a value -1 on X_2. Of course, any additional cases in any of the 3 groups would be coded similarly. If there were 1 case in group A_1, 2 cases in group A_2, and 1 case in group A_3, the ***X*** matrix would be

$$\mathbf{X} = \begin{array}{c|ccc|} & X_0 & X_1 & X_2 \\ A_{11} & 1 & 1 & 0 \\ A_{12} & 1 & 0 & 1 \\ A_{22} & 1 & 0 & 1 \\ A_{13} & 1 & -1 & -1 \end{array}$$

where the first subscript for *A* gives the replicate number for the cases in each group. For brevity, replicates usually are not shown when describing ANOVA design matrices.

Note that in one-way designs with an equal number of cases in each group, sigma-restricted coding yields $X_1 \ldots X_k$ variables all of which have means of 0.

Using the overparameterized model to represent A, the ***X*** matrix defining the between design is simply

$$\mathbf{X} = \begin{array}{c|cccc|} & X_0 & X_1 & X_2 & X_3 \\ A_1 & 1 & 1 & 0 & 0 \\ A_2 & 1 & 0 & 1 & 0 \\ A_3 & 1 & 0 & 0 & 1 \end{array}$$

These simple examples show that the ***X*** matrix actually serves two purposes. It specifies (1) the coding for the levels of the original predictor variables on the *X* variables used in the analysis as well as (2) the nature, number, and arrangement of the X variables, that is, the between design.

Main Effect ANOVA. Main effect ANOVA designs contain separate one-way ANOVA designs for 2 or more categorical predictors. A good example of main effect ANOVA would be the typical analysis performed on *screening designs* as described in the context of the *Experimental Design* chapter.

Consider 2 categorical predictor variables *A* and *B* each with 2 categories. Using the sigma-restricted coding, the ***X*** matrix defining the between design is

$$\mathbf{X} = \begin{array}{c|ccc|} & X_0 & X_1 & X_2 \\ A_1B_1 & 1 & 1 & 1 \\ A_1B_2 & 1 & 1 & -1 \\ A_2B_1 & 1 & -1 & 1 \\ A_2B_2 & 1 & -1 & -1 \end{array}$$

Note that if there are equal numbers of cases in each group, the sum of the cross-products of values for the X_1 and X_2 columns is 0, for example, with 1 case in each group (1*1)+(1*-1)+(-1*1)+(-1*-1)=0. Using the overparameterized model, the matrix ***X*** defining the between design is

$$\mathbf{X} = \begin{array}{c|ccccc|} & X_0 & X_1 & X_2 & X_3 & X_4 \\ A_1B_1 & 1 & 1 & 0 & 1 & 0 \\ A_1B_2 & 1 & 1 & 0 & 0 & 1 \\ A_2B_1 & 1 & 0 & 1 & 1 & 0 \\ A_2B_2 & 1 & 0 & 1 & 0 & 1 \end{array}$$

Comparing the two types of coding, it can be seen that the overparameterized coding takes almost twice as many values as the sigma-restricted coding to convey the same information.

Factorial ANOVA. Factorial ANOVA designs contain *X* variables representing combinations of the levels of 2 or more categorical predictors (e.g., a study of boys and girls in four age groups, resulting in a *2 (Gender) x 4 (Age Group)* design). In particular, full-factorial designs represent all possible combinations of the levels of the categorical predictors. A full-factorial design with 2 categorical predictor variables *A* and *B* each with 2 levels each would be called a 2 x 2 full-factorial design. Using the sigma-restricted coding, the ***X*** matrix for this design would be

$$\mathbf{X} = \begin{array}{c|cccc|} & X_0 & X_1 & X_2 & X_3 \\ A_1B_1 & 1 & 1 & 1 & 1 \\ A_1B_2 & 1 & 1 & -1 & -1 \\ A_2B_1 & 1 & -1 & 1 & -1 \\ A_2B_2 & 1 & -1 & -1 & 1 \end{array}$$

Several features of this ***X*** matrix deserve comment. Note that the X_1 and X_2 columns represent main effect contrasts for one variable, (i.e., *A* and *B*, respectively) collapsing across the levels of the other variable. The X_3 column instead represents a contrast between different combinations of the levels of *A* and *B*. Note also that the values for X_3 are products of the corresponding values for X_1 and X_2. Product variables such as X_3 represent the multiplicative or interaction effects of their factors, so X_3 would be said to represent the 2-way interaction of *A* and *B*. The relationship of such product variables to the dependent variables indicate the interactive influences of the factors on responses above and beyond their independent (i.e., main effect) influences on responses. Thus, factorial designs provide more information about the relationships between categorical predictor variables and responses on the dependent variables than is provided by corresponding one-way or main effect designs.

When many factors are being investigated, however, full-factorial designs sometimes require more data than reasonably can be collected to represent all possible combinations of levels of the factors, and high-order interactions between many factors can become difficult to interpret. With many factors, a useful alternative to the full-factorial design is the fractional factorial design. As an example, consider a 2 x 2 x 2 fractional factorial design to degree 2 with 3 categorical predictor variables each with 2 levels. The design would include the main effects for each variable, and all 2-way interactions between the three variables, but would not include the 3-way interaction between all three variables. Using the overparameterized model, the ***X*** matrix for this design is

$$\mathbf{X} = \begin{array}{c} \\ A_1B_1C_1 \\ A_1B_1C_2 \\ A_1B_2C_1 \\ A_1B_2C_2 \\ A_2B_1C_1 \\ A_2B_1C_2 \\ A_2B_2C_1 \\ A_2B_2C_2 \end{array} \begin{array}{c} \begin{array}{cc} \text{......main effects......} & \text{..............2 - way interactio ns..........} \end{array} \\ \left[\begin{array}{ccccccccccccccccccc} 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right] \end{array}$$

The 2-way interactions are the highest degree effects included in the design. These types of designs are discussed in detail the *2**(k-p) Fractional Factorial Designs* section of the *Experimental Design* chapter.

Nested ANOVA Designs. Nested designs are similar to fractional factorial designs in that all possible combinations of the levels of the categorical predictor variables are not represented in the design. In nested designs, however, the omitted effects are lower-order effects. Nested effects are effects in which the nested variables never appear as main effects. Suppose that for 2 variables *A* and *B* with 3 and 2 levels, respectively, the design includes the main effect for *A* and the effect of *B* nested within the levels of *A*. The ***X*** matrix for this design using the overparameterized model is

$$\mathbf{X} = \begin{array}{c} \\ A_1B_1 \\ A_1B_2 \\ A_2B_1 \\ A_2B_2 \\ A_3B_1 \\ A_3B_2 \end{array} \begin{array}{c} \begin{array}{cccccccccc} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 & X_8 & X_9 \end{array} \\ \left[\begin{array}{cccccccccc} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \end{array}$$

Note that if the sigma-restricted coding were used, there would be only 2 columns in the ***X*** matrix for the *B* nested within *A* effect instead of the 6 columns in the ***X*** matrix for this effect when the overparameterized model coding is used (i.e., columns X_4 through X_9). The sigma-restricted coding method is overly-restrictive for nested designs, so only the overparameterized model is used to represent nested designs.

Balanced ANOVA. Most of the between designs discussed in this section can be analyzed much more efficiently, when they are balanced, i.e., when all cells in the

ANOVA design have equal n, when there are no missing cells in the design, and, if nesting is present, when the nesting is balanced so that equal numbers of levels of the factors that are nested appear in the levels of the factor(s) that they are nested in. In that case, the ***X'X*** matrix (where ***X*** stands for the design matrix) is a diagonal matrix, and many of the computations necessary to compute the ANOVA results (such as matrix inversion) are greatly simplified.

Simple Regression. Simple regression designs involve a single continuous predictor variable. If there were 3 cases with values on a predictor variable *P* of, say, 7, 4, and 9, and the design is for the first-order effect of *P*, the ***X*** matrix would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{cc} X_0 & X_1 \end{array} \\ \left[\begin{array}{cc} 1 & 7 \\ 1 & 4 \\ 1 & 9 \end{array}\right] \end{array}$$

and using *P* for X_1 the regression equation would be

$$Y = b_0 + b_1P$$

If the simple regression design is for a higher-order effect of *P*, say the quadratic effect, the values in the X_1 column of the design matrix would be raised to the 2nd power, that is, squared

$$\mathbf{X} = \begin{array}{c} \begin{array}{cc} X_0 & X_1 \end{array} \\ \left[\begin{array}{cc} 1 & 49 \\ 1 & 16 \\ 1 & 81 \end{array}\right] \end{array}$$

and using P^2 for X_1 the regression equation would be

$$Y = b_0 + b_1P^2$$

The sigma-restricted and overparameterized coding methods do not apply to simple regression designs and any other design containing only continuous predictors (since there are no categorical predictors to code). Regardless of which coding method is chosen, values on the continuous predictor variables are raised to the desired power and used as the values for the *X* variables. No recoding is performed. It is therefore sufficient, in describing regression designs, to simply describe the regression equation without explicitly describing the design matrix ***X***.

Multiple Regression. Multiple regression designs are to continuous predictor variables as main effect ANOVA designs are to categorical predictor variables, that is, multiple regression designs contain the separate simple regression designs for 2 or more continuous predictor variables. The regression equation for a multiple regression design for the first-order effects of 3 continuous predictor variables *P*, *Q*, and *R* would be

$$Y = b_0 + b_1P + b_2Q + b_3R$$

Factorial Regression. Factorial regression designs are similar to factorial ANOVA designs, in which combinations of the levels of the factors are represented in the design. In factorial regression designs, however, there may be many more such possible combinations of distinct levels for the continuous predictor variables than there are cases in the data set. To simplify matters, full-factorial regression designs are defined as designs in which all possible products of the continuous predictor variables are represented in the design. For example, the full-factorial regression design for two continuous predictor variables *P* and *Q* would include the main effects (i.e., the first-order effects) of *P* and *Q* and their 2-way *P* by *Q* interaction effect, which is represented by the product of *P* and *Q* scores for each case. The regression equation would be

$$Y = b_0 + b_1P + b_2Q + b_3P*Q$$

Factorial regression designs can also be fractional, that is, higher-order effects can be omitted from the design. A fractional factorial design to degree 2 for 3 continuous predictor variables *P*, *Q*, and *R* would include the main effects and all 2-way interactions between the predictor variables

$$Y = b_0 + b_1P + b_2Q + b_3R + b_4P*Q + b_5P*R + b_6Q*R$$

Polynomial Regression. Polynomial regression designs are designs which contain main effects and higher-order effects for the continuous predictor variables but do not include interaction effects between predictor variables. For example, the polynomial regression design to degree 2 for three continuous predictor variables *P*, *Q*, and *R* would include the main effects (i.e., the first-order effects) of *P*, *Q*, and *R* and their quadratic (i.e., second-order) effects, but not the 2-way interaction effects or the *P* by *Q* by *R* 3-way interaction effect.

$$Y = b_0 + b_1P + b_2P^2 + b_3Q + b_4Q^2 + b_5R + b_6R^2$$

Polynomial regression designs do not have to contain all effects up to the same degree for every predictor variable. For example, main, quadratic, and cubic effects could be included in the design for some predictor variables, and effects up the fourth degree could be included in the design for other predictor variables.

Response Surface Regression. Quadratic response surface regression designs are a hybrid type of design with characteristics of both polynomial regression designs and fractional factorial regression designs. Quadratic response surface regression designs contain all the same effects of polynomial regression designs to degree 2 and additionally the 2-way interaction effects of the predictor variables. The regression equation for a quadratic response surface regression design for 3 continuous predictor variables *P*, *Q*, and *R* would be

$$Y = b_0 + b_1P + b_2P^2 + b_3Q + b_4Q^2 + b_5R + b_6R^2 + b_7P*Q + b_8P*R + b_9Q*R$$

These types of designs are commonly employed in applied research (e.g., in industrial experimation), and a detailed discussion of these types of designs is also presented in the *Experimental Design* chapter (see *Central composite designs*).

Mixture Surface Regression. Mixture surface regression designs are identical to factorial regression designs to degree 2 except for the omission of the intercept. Mixtures, as the name implies, add up to a constant value; the sum of the proportions of ingredients in different recipes for some material all must add up 100%. Thus, the proportion of one ingredient in a material is redundant with the remaining ingredients. Mixture surface regression designs deal with this redundancy by omitting the intercept from the design. The design matrix for a mixture surface regression design for 3 continuous predictor variables *P*, *Q*, and *R* would be

$$Y = b_1P + b_2Q + b_3R + b_4P*Q + b_5P*R + b_6Q*R$$

These types of designs are commonly employed in applied research (e.g., in industrial experimentation), and a detailed discussion of these types of designs is also presented in the *Experimental Design* chapter (see *Mixture designs and triangular surfaces*).

Analysis of Covariance. In general, between designs which contain both categorical and continuous predictor variables can be called ANCOVA designs. Traditionally, however, ANCOVA designs have referred more specifically to designs in which the first-order effects of one or more continuous predictor variables are taken into account when assessing the effects of one or more categorical predictor variables. A basic introduction to analysis of covariance can also be found in the *Analysis of covariance (ANCOVA)* topic of the *ANOVA/MANOVA* chapter.

To illustrate, suppose a researcher wants to assess the influences of a categorical predictor variable *A* with 3 levels on some outcome, and that measurements on a continuous predictor variable *P*, known to covary with the outcome, are available. If the data for the analysis are

$$\begin{array}{cc} P & \text{Group} \\ \begin{bmatrix} 7 \\ 4 \\ 9 \\ 3 \\ 6 \\ 8 \end{bmatrix} & \begin{bmatrix} A_1 \\ A_1 \\ A_2 \\ A_2 \\ A_3 \\ A_3 \end{bmatrix} \end{array}$$

then the sigma-restricted ***X*** matrix for the design that includes the separate first-order effects of *P* and *A* would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{cccc} X_0 & X_1 & X_2 & X_3 \end{array} \\ \left[\begin{array}{cccc} 1 & 7 & 1 & 0 \\ 1 & 4 & 1 & 0 \\ 1 & 9 & 0 & 1 \\ 1 & 3 & 0 & 1 \\ 1 & 6 & -1 & -1 \\ 1 & 8 & -1 & -1 \end{array}\right] \end{array}$$

The b_2 and b_3 coefficients in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3$$

represent the influences of group membership on the *A* categorical predictor variable, controlling for the influence of scores on the *P* continuous predictor variable. Similarly, the b_1 coefficient represents the influence of scores on *P* controlling for the influences of group membership on *A*. This traditional ANCOVA analysis gives a more sensitive test of the influence of *A* to the extent that *P* reduces the prediction error, that is, the residuals for the outcome variable.

The ***X*** matrix for the same design using the overparameterized model would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{ccccc} X_0 & X_1 & X_2 & X_3 & X_4 \end{array} \\ \left[\begin{array}{ccccc} 1 & 7 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ 1 & 9 & 0 & 1 & 0 \\ 1 & 3 & 0 & 1 & 0 \\ 1 & 6 & 0 & 0 & 1 \\ 1 & 8 & 0 & 0 & 1 \end{array}\right] \end{array}$$

The interpretation is unchanged except that the influences of group membership on the *A* categorical predictor variables are represented by the b_2, b_3 and b_4 coefficients in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4$$

Separate Slope Designs. The traditional analysis of covariance (ANCOVA) design for categorical and continuous predictor variables is inappropriate when the categorical and continuous predictors interact in influencing responses on the outcome. The appropriate design for modeling the influences of the predictors in this situation is called the separate slope design. For the same example data used to illustrate traditional ANCOVA, the overparameterized ***X*** matrix for the design that includes the main effect of the three-level categorical predictor *A* and the 2-way interaction of *P* by *A* would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{ccccccc} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 \end{array} \\ \left[\begin{array}{ccccccc} 1 & 1 & 0 & 0 & 7 & 0 & 0 \\ 1 & 1 & 0 & 0 & 4 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 9 & 0 \\ 1 & 0 & 1 & 0 & 0 & 3 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 6 \\ 1 & 0 & 0 & 1 & 0 & 0 & 8 \end{array}\right] \end{array}$$

The b_4, b_5, and b_6 coefficients in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + b_5X_5 + b_6X_6$$

give the separate slopes for the regression of the outcome on *P* within each group on *A*, controlling for the main effect of *A*.

As with nested ANOVA designs, the sigma-restricted coding of effects for separate slope designs is overly restrictive, so only the overparameterized model is used to represent separate slope designs. In fact, separate slope designs are identical in form to nested ANOVA designs, since the main effects for continuous predictors are omitted in separate slope designs.

Homogeneity of Slopes. The appropriate design for modeling the influences of continuous and categorical predictor variables depends on whether the continuous and categorical predictors interact in influencing the outcome. The traditional analysis of covariance (ANCOVA) design for continuous and categorical predictor variables is appropriate when the continuous and categorical predictors do not interact in influencing responses on the outcome, and the separate slope design is appropriate when the continuous and categorical predictors do interact in influencing responses. The homogeneity of slopes designs can be used to test whether the continuous and categorical predictors interact in influencing responses, and thus, whether the traditional ANCOVA design or the separate slope design is appropriate for modeling the effects of the predictors. For the same example data used to illustrate the traditional ANCOVA and separate slope designs, the overparameterized ***X*** matrix for the design that includes the main effect of *P*, the main effect of the three-level categorical predictor *A*, and the 2-way interaction of *P* by *A* would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{cccccccc} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 \end{array} \\ \left[\begin{array}{cccccccc} 1 & 7 & 1 & 0 & 0 & 7 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 & 4 & 0 & 0 \\ 1 & 9 & 0 & 1 & 0 & 0 & 9 & 0 \\ 1 & 3 & 0 & 1 & 0 & 0 & 3 & 0 \\ 1 & 6 & 0 & 0 & 1 & 0 & 0 & 6 \\ 1 & 8 & 0 & 0 & 1 & 0 & 0 & 8 \end{array}\right] \end{array}$$

If the b_5, b_6, or b_7 coefficient in the regression equation

$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + b_5X_5 + b_6X_6 + b_7X_7$

is non-zero, the separate slope model should be used. If instead all 3 of these regression coefficients are zero the traditional ANCOVA design should be used.

The sigma-restricted ***X*** matrix for the homogeneity of slopes design would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{cccccc} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 \end{array} \\ \left[\begin{array}{cccccc} 1 & 7 & 1 & 0 & 7 & 0 \\ 1 & 4 & 1 & 0 & 4 & 0 \\ 1 & 9 & 0 & 1 & 0 & 9 \\ 1 & 3 & 0 & 1 & 0 & 3 \\ 1 & 6 & -1 & -1 & -6 & -6 \\ 1 & 8 & -1 & -1 & -8 & -8 \end{array}\right] \end{array}$$

Using this ***X*** matrix, if the b_4, or b_5 coefficient in the regression equation

$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + b_5X_5$

is non-zero, the separate slope model should be used. If instead both of these regression coefficients are zero the traditional ANCOVA design should be used.

Mixed Model ANOVA and ANCOVA. Designs which contain random effects for one or more categorical predictor variables are called mixed-model designs. Random effects are classification effects where the levels of the effects are assumed to be randomly selected from an infinite population of possible levels. The solution for the normal equations in mixed-model designs is identical to the solution for fixed-effect designs (i.e., designs which do not contain Random effects. Mixed-model designs differ from fixed-effect designs only in the way in which effects are tested for significance. In fixed-effect designs, between effects are always tested using the mean squared residual as the error term. In mixed-model designs, between effects are tested using relevant error terms based on the covariation of random sources of variation in the design. Specifically, this is done using Satterthwaite's method of denominator synthesis (Satterthwaite, 1946), which finds the linear combinations of sources of random variation that serve as appropriate error terms for testing the significance of the respective effect of interest. A basic discussion of these types of designs, and methods for estimating variance components for the random effects can also be found in the *Variance Components and Mixed Model ANOVA/ANCOVA* chapter.

Mixed-model designs, like nested designs and separate slope designs, are designs in which the sigma-restricted coding of categorical predictors is overly restrictive. Mixed-model designs require estimation of the covariation between the levels of categorical predictor variables, and the sigma-restricted coding of categorical predictors suppresses this covariation. Thus, only the overparameterized model is used to represent mixed-model designs (some programs will use the sigma-restricted approach and a so-called "restricted model" for random effects; however, only the overparameterized model as described in General Linear Models applies to both balanced and unbalanced designs, as well as designs with missing cells; see Searle, Casella, & McCullock, 1992, p. 127). It is important to recognize, however, that sigma-restricted coding can be used to represent any between design, with the exceptions of mixed-model, nested, and separate slope designs. Furthermore, some types of hypotheses can only be tested using the sigma-restricted coding (i.e., the effective hypothesis, Hocking, 1996), thus the greater generality of the overparameterized model for representing between designs does not justify it being used exclusively for representing categorical predictors in the general linear model.

Within-Subject (Repeated Measures) Designs

- Overview
- One-way within-subject designs
- Multi-way within-subject designs
- The multivariate approach to Repeated Measures
- Doubly multivariate within-subject designs

Overview. It is quite common for researchers to administer the same test to the same subjects repeatedly over a period of time or under varying circumstances. In essence, one is interested in examining differences within each subject, for example, subjects' improvement over time. Such designs are referred to as within-subject designs or repeated measures designs. A basic introduction to repeated measures designs is also provided in the *Between-groups and repeated measures* topic of the *ANOVA/MANOVA* chapter.

For example, imagine that one wants to monitor the improvement of students' algebra skills over two months of instruction. A standardized algebra test is administered after one month (level 1 of the repeated measures factor), and a comparable test is administered after two months (level 2 of the repeated measures factor). Thus, the repeated measures factor (*Time*) has 2 levels.

Now, suppose that scores for the 2 algebra tests (i.e., values on the Y_1 and Y_2 variables at *Time 1* and *Time 2*, respectively) are transformed into scores on a new composite variable (i.e., values on the T_1), using the linear transformation

$$\mathbf{T} = \mathbf{YM}$$

where $\boldsymbol{M}$ is an orthonormal contrast matrix. Specifically, if

$$\begin{bmatrix} T_{11} \\ \vdots \\ \vdots \\ \vdots \\ T_{n1} \end{bmatrix} = \begin{bmatrix} Y_{11} & Y_{12} \\ \vdots & \vdots \\ \vdots & \vdots \\ \vdots & \vdots \\ Y_{n1} & Y_{n2} \end{bmatrix} \begin{bmatrix} \sqrt{.5} \\ -\sqrt{.5} \end{bmatrix}$$

then the difference of the mean score on T_1 from 0 indicates the improvement (or deterioration) of scores across the 2 levels of *Time*.

One-Way Within-Subject Designs. The example algebra skills study with the *Time* repeated measures factor (see also within-subjects design *Overview*) illustrates a one-way within-subject design. In such designs, orthonormal contrast transformations of the scores on the original dependent *Y* variables are performed via the ***M*** transformation (orthonormal transformations correspond to orthogonal rotations of the original variable axes). If any b_0 coefficient in the regression of a transformed *T* variable on the intercept is non-zero, this indicates a change in responses across the levels of the repeated measures factor, that is, the presence of a main effect for the repeated measure factor on responses.

What if the between design includes effects other than the intercept? If any of the b_1 through b_k coefficients in the regression of a transformed *T* variable on ***X*** are non-zero, this indicates a different change in responses across the levels of the repeated measures factor for different levels of the corresponding between effect, i.e., the presence of a within by between interaction effect on responses.

The same between-subject effects that can be tested in designs with no repeated-measures factors can also be tested in designs that do include repeated-measures factors. This is accomplished by creating a transformed dependent variable which is the sum of the original dependent variables divided by the square root of the number of original dependent variables. The same tests of between-subject effects that are performed in designs with no repeated-measures factors (including tests of the between intercept) are performed on this transformed dependent variable.

Multi-Way Within-Subject Designs. Suppose that in the example algebra skills study with the *Time* repeated measures factor (see the within-subject designs Overview), students were given a number problem test and then a word problem test on each testing occasion. *Test* could then be considered as a second repeated measures factor, with scores on the number problem tests representing responses at level 1 of the *Test* repeated measure factor, and scores on the word problem tests representing responses at level 2 of the *Test* repeated measure factor. The within subject design for the study would be a 2 *(Time)* by 2 (*Test*) full-factorial design, with effects for *Time, Test,* and the *Time* by *Test* interaction.

To construct transformed dependent variables representing the effects of *Time*, *Test*, and the *Time* by *Test* interaction, three respective ***M*** transformations of the original dependent *Y* variables are performed. Assuming that the original *Y* variables are in the order *Time 1*

- *Test 1*, *Time 1* - *Test 2*, *Time 2* - *Test 1*, and *Time 2* - *Test 2*, the ***M*** matrices for the *Time*, *Test*, and the *Time* by *Test* interaction would be

$$\mathbf{M}_{Time} = \begin{bmatrix} .5 \\ .5 \\ -.5 \\ -.5 \end{bmatrix}, \quad \mathbf{M}_{Test} = \begin{bmatrix} .5 \\ -.5 \\ .5 \\ -.5 \end{bmatrix}, \quad \mathbf{M}_{Time \times Test} = \begin{bmatrix} .5 \\ -.5 \\ -.5 \\ .5 \end{bmatrix}$$

The differences of the mean scores on the transformed *T* variables from 0 are then used to interpret the corresponding within-subject effects. If the b_0 coefficient in the regression of a transformed *T* variable on the intercept is non-zero, this indicates a change in responses across the levels of a repeated measures effect, that is, the presence of the corresponding main or interaction effect for the repeated measure factors on responses.

Interpretation of within by between interaction effects follow the same procedures as for one-way within designs, except that now within by between interactions are examined for each within effect by between effect combination.

Multivariate Approach to Repeated Measures. When the repeated measures factor has more than 2 levels, then the ***M*** matrix will have more than a single column. For example, for a repeated measures factor with 3 levels (e.g., *Time 1*, *Time 2*, *Time 3*), the ***M*** matrix will have 2 columns (e.g., the two transformations of the dependent variables could be (1) *Time 1* vs. *Time 2* and *Time 3* combined, and (2) *Time 2* vs. *Time 3*). Consequently, the nature of the design is really multivariate, that is, there are two simultaneous dependent variables, which are transformations of the original dependent variables. Therefore, when testing repeated measures effects involving more than a single degree of freedom (e.g., a repeated measures main effect with more than 2 levels), you can compute multivariate test statistics to test the respective hypotheses. This is a different (and usually the preferred) approach than the univariate method that is still widely used. For a further discussion of the multivariate approach to testing repeated measures effects, and a comparison to the traditional univariate approach, see the *Sphericity and compound symmetry* topic of the *ANOVA/MANOVA* chapter.

Doubly Multivariate Designs. If the product of the number of levels for each within-subject factor is equal to the number of original dependent variables, the within-subject design is called a univariate repeated measures design. The within design is univariate because there is one dependent variable representing each combination of levels of the within-subject factors. Note that this use of the term *univariate design* is not to be confused with the univariate and multivariate approach to the analysis of repeated measures designs, both of which can be used to analyze such univariate (single-dependent-variable-only) designs. When there are two or more dependent variables for each combination of levels of the within-subject factors, the within-subject design is called a multivariate repeated measures design, or more commonly, a doubly multivariate within-subject design. This term is used because the analysis for each dependent measure can be done via the multivariate approach; so when there is more than one dependent measure, the design can be considered doubly-multivariate.

Doubly multivariate design are analyzed using a combination of univariate repeated measures and multivariate analysis techniques. To illustrate, suppose in an algebra skills study, tests are administered three times (repeated measures factor *Time* with 3 levels). Two test scores are recorded at each level of *Time*: a *Number Problem* score and a *Word Problem* score. Thus, scores on the two types of tests could be treated as multiple measures on which improvement (or deterioration) across *Time* could be assessed. ***M*** transformed variables could be computed for each set of test measures, and multivariate tests of significance could be performed on the multiple transformed measures, as well as on the each individual test measure.

Multivariate Designs

Overview. When there are multiple dependent variables in a design, the design is said to be multivariate. Multivariate measures of association are by nature more complex than their univariate counterparts (such as the correlation coefficient, for example). This is because multivariate measures of association must take into account not only the relationships of the predictor variables with responses on the dependent variables, but also the relationships among the multiple dependent variables. By doing so, however, these measures of association provide information about the strength of the relationships between predictor and dependent variables independent of the dependent variable interrelationships. A basic discussion of multivariate designs is also presented in the *Multivariate Designs* topic in the *ANOVA/MANOVA* chapter.

The most commonly used multivariate measures of association all can be expressed as functions of the eigenvalues of the product matrix

$$\mathbf{E}^{-1}\mathbf{H}$$

where ***E*** is the error SSCP matrix (i.e., the matrix of sums of squares and cross-products for the dependent variables that are not accounted for by the predictors in the between design), and ***H*** is a hypothesis SSCP matrix (i.e., the matrix of sums of squares and cross-products for the dependent variables that are accounted for by all the predictors in the between design, or the sums of squares and cross-products for the dependent variables that are accounted for by a particular effect). If

$\square_i$ = the ordered eigenvalues of $\mathbf{E}^{-1}\mathbf{H}$, if $\mathbf{E}^{-1}$ exists

then the 4 commonly used multivariate measures of association are

Wilks' lambda = $\square[1/(1+\square_i)]$

Pillai's trace = $\square\square_i/(1+\square_i)$

Hotelling-Lawley trace = $\square\square_i$

Roy's largest root = $\square_1$

These 4 measures have different upper and lower bounds, with Wilks' lambda perhaps being the most easily interpretable of the 4 measures. Wilks' lambda can range from 0 to 1, with 1 indicating no relationship of predictors to responses and 0 indicating a perfect relationship of predictors to responses. 1 - Wilks' lambda can be interpreted as the multivariate counterpart of a univariate R-squared, that is, it indicates the proportion of generalized variance in the dependent variables that is accounted for by the predictors.

The 4 measures of association are also used to construct multivariate tests of significance. These multivariate tests are covered in detail in a number of sources (e.g., Finn, 1974; Tatsuoka, 1971).

Estimation and Hypothesis Testing

The following sections discuss details concerning hypothesis testing in the context of *STATISTICA*'s *VGLM* module, for example, how the test for the overall model fit is computed, the options for computing tests for categorical effects in unbalanced or incomplete designs, how and when custom-error terms can be chosen, and the logic of testing custom-hypotheses in factorial or regression designs.

Whole model tests

Partitioning Sums of Squares. A fundamental principle of least squares methods is that variation on a dependent variable can be partitioned, or divided into parts, according to the sources of the variation. Suppose that a dependent variable is regressed on one or more predictor variables, and that for covenience the dependent variable is scaled so that its mean is 0. Then a basic least squares identity is that the total sum of squared values on the dependent variable equals the sum of squared predicted values plus the sum of squared residual values. Stated more generally,

$$□(y - y\text{-bar})^2 = □(y\text{-hat} - y\text{-bar})^2 + □(y - y\text{-hat})^2$$

where the term on the left is the total sum of squared deviations of the observed values on the dependent variable from the dependent variable mean, and the respective terms on the right are (1) the sum of squared deviations of the predicted values for the dependent variable from the dependent variable mean and (2) the sum of the squared deviations of the observed values on the dependent variable from the predicted values, that is, the sum of the squared residuals. Stated yet another way,

Total SS = Model SS + Error SS

Note that the Total SS is always the same for any particular data set, but that the Model SS and the Error SS depend on the regression equation. Assuming again that the dependent variable is scaled so that its mean is 0, the Model SS and the Error SS can be computed using

Model SS = $\mathbf{b'X'Y}$

Error SS = $\mathbf{Y'Y - b'X'Y}$

Testing the Whole Model. Given the Model SS and the Error SS, one can perform a test that all the regression coefficients for the *X* variables (*b1* through *bk*) are zero. This test is equivalent to a comparison of the fit of the regression surface defined by the predicted values (computed from the whole model regression equation) to the fit of the regression surface defined solely by the dependent variable mean (computed from the reduced regression equation containing only the intercept). Assuming that ***X'X*** is full-rank, the whole model hypothesis mean square

MSH = (Model SS)/k

is an estimate of the variance of the predicted values. The error mean square

s^2 = MSE = (Error SS)/(n-k-1)

is an unbiased estimate of the residual or error variance. The test statistic is

F = MSH/MSE

where *F* has (*k*, *n* - *k* - 1) degrees of freedom.

If ***X'X*** is not full rank, *r* + 1 is substituted for *k*, where *r* is the rank or the number of non-redundant columns of ***X'X***.

Note that in the case of non-intercept models, some multiple regression programs will compute the full model test based on the proportion of variance around 0 (zero) accounted for by the predictors; for more information (see Kvålseth, 1985; Okunade, Chang, and Evans, 1993), while other will actually compute both values (i.e., based on the residual variance around 0, and around the respective dependent variable means.

Limitations of Whole Model Tests. For designs such as one-way ANOVA or simple regression designs, the whole model test by itself may be sufficient for testing general hypotheses about whether or not the single predictor variable is related to the outcome. In more complex designs, however, hypotheses about specific *X* variables or subsets of *X* variables are usually of interest. For example, one might want to make inferences about whether a subset of regression coefficients are 0, or one might want to test whether subpopulation means corresponding to combinations of specific *X* variables differ. The whole model test is usually insufficient for such purposes.

A variety of methods have been developed for testing specific hypotheses. Like whole model tests, many of these methods rely on comparisons of the fit of different models (e.g., Type I, Type II, and the effective hypothesis sums of squares). Other methods construct tests of linear combinations of regression coefficients in order to test mean differences (e.g., Type III, Type IV, and Type V sums of squares). For designs that contain only first-order effects of continuous predictor variables (i.e., multiple regression designs), many of these methods are equivalent (i.e., Type II through Type V sums of squares all test the significance of partial regression coefficients). However, there are important distinctions between the different hypothesis testing techniques for certain types of ANOVA designs (i.e., designs with unequal cell *n*'s and/or missing cells).

All methods for testing hypotheses, however, involve the same hypothesis testing strategy employed in whole model tests, that is, the sums of squares attributable to an effect (using a given criterion) is computed, and then the mean square for the effect is tested using an appropriate error term.

Six types of sums of squares

- Contained effects
- Type I sums of squares
- Type II sums of squares
- Type III sums of squares
- Type IV sums of squares
- Type V sums of squares
- Type VI (effective hypothesis) sums of squares

When there are categorical predictors in the model, arranged in a factorial ANOVA design, then one is typically interested in the main effects for and interaction effects between the categorical predictors. However, when the design is not balanced (has unequal cell *n*'s, and consequently, the coded effects for the categorical factors are usually correlated), or when there are missing cells in a full factorial ANOVA design, then there is ambiguity regarding the specific comparisons between the (population, or least-squares) cell means that constitute the main effects and interactions of interest. These issues are discussed in great detail in Milliken and Johnson (1986), and if you routinely analyze incomplete factorial designs, you should consult their discussion of various problems and approaches to solving them.

In addition to the widely used methods that are commonly labeled *Type I, II, III*, and *IV* sums of squares (see Goodnight, 1980), we also offer different methods for testing effects in incomplete designs, that are widely used in other areas (and traditions) of research.

Type V sums of squares. Specifically, we propose the term *Type V sums of squares* to denote the approach that is widely used in industrial experimentation, to analyze fractional factorial designs; these types of designs are discussed in detail in the *2**(k-p) Fractional Factorial Designs* section of the *Experimental Design* chapter. In effect, for those effects for which tests are performed all population marginal means (least squares means) are estimable.

Type VI sums of squares. Second, in keeping with the *Type i* labeling convention, we propose the term *Type VI sums of squares* to denote the approach that is often used in programs that only implement the sigma-restricted model (which is not well suited for certain types of designs; we offer a choice between the sigma-restricted and overparameterized model models). This approach is identical to what is described as the *effective hypothesis* method in Hocking (1996).

Contained Effects. The following descriptions will use the term *contained effect*. An effect *E1* (e.g., *A * B* interaction) is contained in another effect *E2* if:

- Both effects involve the same continuous predictor variable (if included in the model; e.g., *A * B * X* would be contained in *A * C * X*, where *A*, *B*, and *C* are categorical predictors, and *X* is a continuous predictor); or
- *E2* has more categorical predictors than does *E1*, and, if *E1* includes any categorical predictors, they also appear in *E2* (e.g., *A * B* would be contained in the *A * B * C* interaction).

Type I Sums of Squares. Type I sums of squares involve a sequential partitioning of the whole model sums of squares. A hierarchical series of regression equations are estimated, at each step adding an additional effect into the model. In Type I sums of squares, the sums of squares for each effect are determined by subtracting the predicted sums of squares with the effect in the model from the predicted sums of squares for the preceding model not including the effect. Tests of significance for each effect are then performed on the increment in the predicted sums of squares accounted for by the effect. Type I sums of squares are therefore sometimes called sequential or hierarchical sums of squares.

Type I sums of squares are appropriate to use in balanced (equal *n*) ANOVA designs in which effects are entered into the model in their natural order (i.e., any main effects are entered before any two-way interaction effects, any two-way interaction effects are entered before any three-way interaction effects, and so on). Type I sums of squares are also useful in polynomial regression designs in which any lower-order effects are entered before any higher-order effects. A third use of Type I sums of squares is to test hypotheses for hierarchically nested designs, in which the first effect in the design is nested within the second effect, the second effect is nested within the third, and so on.

One important property of Type I sums of squares is that the sums of squares attributable to each effect add up to the whole model sums of squares. Thus, Type I sums of squares provide a complete decomposition of the predicted sums of squares for the whole model. This is not generally true for any other type of sums of squares. An important limitation of Type I sums of squares, however, is that the sums of squares attributable to a specific effect will generally depend on the order in which the effects are entered into the model.

This lack of invariance to order of entry into the model limits the usefulness of Type I sums of squares for testing hypotheses for certain designs (e.g., fractional factorial designs).

Type II Sums of Squares. Type II sums of squares are sometimes called partially sequential sums of squares. Like Type I sums of squares, Type II sums of squares for an effect controls for the influence of other effects. Which other effects to control for, however, is determined by a different criterion. In Type II sums of squares, the sums of squares for an effect is computed by controlling for the influence of all other effects of equal or lower degree. Thus, sums of squares for main effects control for all other main effects, sums of squares for two-way interactions control for all main effects and all other two-way interactions, and so on.

Unlike Type I sums of squares, Type II sums of squares are invariant to the order in which effects are entered into the model. This makes Type II sums of squares useful for testing hypotheses for multiple regression designs, for main effect ANOVA designs, for full-factorial ANOVA designs with equal cell *n*s, and for hierarchically nested designs.

There is a drawback to the use of Type II sums of squares for factorial designs with unequal cell *n*s. In these situations, Type II sums of squares test hypotheses that are complex functions of the cell *ns* that ordinarily are not meaningful. Thus, a different method for testing hypotheses is usually preferred.

Type III Sums of Squares. Type I and Type II sums of squares usually are not appropriate for testing hypotheses for factorial ANOVA designs with unequal *n*s. For ANOVA designs with unequal *n*s, however, Type III sums of squares test the same hypothesis that would be tested if the cell *n*s were equal, provided that there is at least one observation in every cell. Specifically, in no-missing-cell designs, Type III sums of squares test hypotheses about differences in subpopulation (or marginal) means. When there are no missing cells in the design, these subpopulation means are least squares means, which are the best linear-unbiased estimates of the marginal means for the design (see, Milliken and Johnson, 1986).

Tests of differences in least squares means have the important property that they are invariant to the choice of the coding of effects for categorical predictor variables (e.g., the use of the sigma-restricted or overparameterized model) and to the choice of the particular g2 inverse of ***X'X*** used to solve the normal equations. Thus, tests of linear combinations of least squares means in general, including Type III tests of differences in least squares means, are said to not depend on the parameterization of the design. This makes Type III sums of squares useful for testing hypotheses for any design for which Type I or Type II sums of squares are appropriate, as well as for any unbalanced ANOVA design with no missing cells.

The Type III sums of squares attributable to an effect is computed as the sums of squares for the effect controlling for any effects of equal or lower degree and orthogonal to any higher-order interaction effects (if any) that contain it. The orthogonality to higher-order containing interactions is what gives Type III sums of squares the desirable properties associated with linear combinations of least squares means in ANOVA designs with no missing cells. But for ANOVA designs with missing cells, Type III sums of squares generally do not test hypotheses about least squares means, but instead test hypotheses that are complex functions of the patterns of missing cells in higher-order containing interactions and that are ordinarily not meaningful. In this situation Type V sums of squares or tests of the effective hypothesis (Type VI sums of squares) are preferred.

Type IV Sums of Squares. Type IV sums of squares were designed to test "balanced" hypotheses for lower-order effects in ANOVA designs with missing cells. Type IV sums of squares are computed by equitably distributing cell contrast coefficients for lower-order effects across the levels of higher-order containing interactions.

Type IV sums of squares are not recommended for testing hypotheses for lower-order effects in ANOVA designs with missing cells, even though this is the purpose for which they were developed. This is because Type IV sum-of-squares are invariant to some but not all g2 inverses of ***X'X*** that could be used to solve the normal equations. Specifically, Type IV sums of squares are invariant to the choice of a g2 inverse of ***X'X*** given a particular ordering of the levels of the categorical predictor variables, but are not invariant to different orderings of levels. Furthermore, as with Type III sums of squares, Type IV sums of squares test hypotheses that are complex functions of the patterns of missing cells in higher-order containing interactions and that are ordinarily not meaningful.

Statisticians who have examined the usefulness of Type IV sums of squares have concluded that Type IV sums of squares are not up to the task for which they were developed:

- Milliken & Johnson (1992, p. 204) write: "It seems likely that few, if any, of the hypotheses tested by the Type IV analysis of *[some programs]* will be of particular interest to the experimenter."
- Searle (1987, p. 463-464) writes: "In general, [Type IV] hypotheses determined in this nature are not necessarily of any interest."; and (p. 465) "This characteristic of Type IV sums of squares for rows depending on the sequence of rows establishes their non-uniqueness, and this in turn emphasizes that the hypotheses they are testing are by no means necessarily of any general interest."
- Hocking (1985, p. 152), in an otherwise comprehensive introduction to general linear models, writes: "For the missing cell problem, *[some programs]* offers a fourth analysis, Type IV, which we shall not discuss."

So, we recommend that you use the Type IV sums of squares solution with caution, and that you understand fully the nature of the (often non-unique) hypotheses that are being testing, before attempting interpretations of the results. Furthermore, in ANOVA designs with no missing cells, Type IV sums of squares are always equal to Type III sums of squares, so the use of Type IV sums of squares is either (potentially) inappropriate, or unnecessary, depending on the presence of missing cells in the design.

Type V Sums of Squares. Type V sums of squares were developed as an alternative to Type IV sums of squares for testing hypotheses in ANOVA designs in missing cells. Also, this approach is widely used in industrial experimentation, to analyze fractional factorial designs; these types of designs are discussed in detail in the *2**(k-p) Fractional Factorial Designs* section of the *Experimental Design* chapter. In effect, for effects for which tests are performed all population marginal means (least squares means) are estimable.

Type V sums of squares involve a combination of the methods employed in computing Type I and Type III sums of squares. Specifically, whether or not an effect is eligible to be dropped from the model is determined using Type I procedures, and then hypotheses are tested for effects not dropped from the model using Type III procedures. Type V sums of squares can be illustrated by using a simple example. Suppose that the effects considered are *A*, *B*, and *A* by *B*, in that order, and that *A* and *B* are both categorical predictors with, say, 3 and 2 levels, respectively. The intercept is first entered into the model. Then *A* is entered into the model, and its degrees of freedom are determined (i.e., the number of non-redundant columns for *A* in ***X'X***, given the intercept). If *A*'s degrees of freedom are less than 2 (i.e., its number of levels minus 1), it is eligible to be dropped. Then *B* is entered into the model, and its degrees of freedom are determined (i.e., the number of non-redundant columns for *B* in ***X'X***, given the intercept and *A*). If *B*'s degrees of freedom are less than 1 (i.e., its number of levels minus 1), it is eligible to be dropped. Finally, *A* by *B* is entered into the model, and its degrees of freedom are determined (i.e., the number of non-redundant columns for *A* by *B* in ***X'X***, given the intercept, *A*, and *B*). If *B*'s degrees of freedom are less than 2 (i.e., the product of the degrees of freedom for its factors if there were no missing cells), it is eligible to be dropped. Type III sums of squares are then computed for the effects that were not found to be eligible to be dropped, using the reduced model in which any eligible effects are dropped. Tests of significance, however, use the error term for the whole model prior to dropping any eligible effects.

Note that Type V sums of squares involve determining a reduced model for which all effects remaining in the model have at least as many degrees of freedom as they would have if there were no missing cells. This is equivalent to finding a subdesign with no missing cells such that the Type III sums of squares for all effects in the subdesign reflect differences in least squares means.

Appropriate caution should be exercised when using Type V sums of squares. Dropping an effect from a model is the same as assuming that the effect is unrelated to the outcome (see, e.g., Hocking, 1996). The reasonableness of the assumption does not necessarily insure its validity, so when possible the relationships of dropped effects to the outcome should be inspected. It is also important to note that Type V sums of squares are not invariant to the order in which eligibility for dropping effects from the model is evaluated. Different orders of effects could produce different reduced models.

In spite of these limitations, Type V sums of squares for the reduced model have all the same properties of Type III sums of squares for ANOVA designs with no missing cells. Even in designs with many missing cells (such as fractional factorial designs, in which many high-order interaction effects are assumed to be zero), Type V sums of squares provide tests of meaningful hypotheses, and sometimes hypotheses that cannot be tested using any other method.

Type VI (Effective Hypothesis) Sums of Squares. Type I through Type V sums of squares can all be viewed as providing tests of hypotheses that subsets of partial regression coefficients (controlling for or orthogonal to appropriate additional effects) are zero. Effective hypothesis tests (developed by Hocking, 1996) are based on the philosophy that the only unambiguous estimate of an effect is the proportion of variability on the outcome that is uniquely attributable to the effect. The overparameterized coding of effects for categorical predictor variables generally cannot be used to provide such unique estimates for lower-order effects. Effective hypothesis tests, which we propose to call Type VI sums of squares, use the sigma-restricted coding of effects for categorical predictor variables to provide unique effect estimates even for lower-order effects.

The method for computing Type VI sums of squares is straightforward. The sigma-restricted coding of effects is used, and for each effect, its Type VI sums of squares is the difference of the model sums of squares for all other effects from the whole model sums of squares. As such, the Type VI sums of squares provide an unambiguous estimate of the variability of predicted values for the outcome uniquely attributable to each effect.

In ANOVA designs with missing cells, Type VI sums of squares for effects can have fewer degrees of freedom than they would have if there were no missing cells, and for some missing cell designs, can even have zero degrees of freedom. The philosophy of Type VI sums of squares is to test as much as possible of the original hypothesis given the observed cells. If the pattern of missing cells is such that no part of the original hypothesis can be tested, so be it. The inability to test hypotheses is simply the price one pays for having no observations at some combinations of the levels of the categorical predictor variables. The philosophy is that it is better to admit that a hypothesis cannot be tested than it is to test a distorted hypothesis which may not meaningfully reflect the original hypothesis.

Type VI sums of squares cannot generally be used to test hypotheses for nested ANOVA designs, separate slope designs, or mixed-model designs, because the sigma-restricted coding of effects for categorical predictor variables is overly restrictive in such designs. This limitation, however, does not diminish the fact that Type VI sums of squares can b

Error terms for tests

Lack-of-Fit Tests using Pure Error. Whole model tests and tests based on the 6 types of sums of squares use the mean square residual as the error term for tests of significance. For certain types of designs, however, the residual sum of squares can be further partitioned into meaningful parts which are relevant for testing hypotheses. One such type of design is a simple regression design in which there are subsets of cases all having the same values on the predictor variable. For example, performance on a task could be measured for subjects who work on the task under several different room temperature conditions. The test of significance for the *Temperature* effect in the linear regression of *Performance* on *Temperature* would not necessarily provide complete information on how *Temperature* relates to *Performance*; the regression coefficient for *Temperature* only reflects its linear effect on the outcome.

One way to glean additional information from this type of design is to partition the residual sums of squares into lack-of-fit and pure error components. In the example just described, this would involve determining the difference between the sum of squares that cannot be predicted by *Temperature* levels, given the linear effect of *Temperature* (residual sums of squares) and the pure error; this difference would be the sums of squares associated with the lack-of-fit (in this example, of the linear model). The test of lack-of-fit, using the mean square pure error as the error term, would indicate whether non-linear effects of *Temperature* are needed to adequately model *Tempature's* influence on the outcome. Further, the linear effect could be tested using the pure error term, thus providing a more sensitive test of the linear effect independent of any possible nonlinear effect.

Designs with Zero Degrees of Freedom for Error. When the model degrees of freedom equal the number of cases or subjects, the residual sums of squares will have zero degrees of freedom and preclude the use of standard hypothesis tests. This sometimes occurs for overfitted designs (designs with many predictors, or designs with categorical predictors having many levels). However, in some designed experiments, such as experiments using split-plot designs or highly fractionalized factorial designs as commonly used in industrial experimentation, it is no accident that the residual sum of squares has zero degrees of freedom. In such experiments, mean squares for certain effects are planned to be used as error terms for testing other effects, and the experiment is designed with this in mind. It is entirely appropriate to use alternatives to the mean square residual as error terms for testing hypotheses in such designs.

Tests in Mixed Model Designs. Designs which contain random effects for one or more categorical predictor variables are called mixed-model designs. These types of designs, and the analysis of those designs, is also described in detail in the *Variance Components and Mixed Model ANOVA/ANCOVA* chapter. Random effects are classification effects where the levels of the effects are assumed to be randomly selected from an infinite population of possible levels. The solution for the normal equations in mixed-model designs is identical to the solution for fixed-effect designs (i.e., designs which do not contain random effects). Mixed-model designs differ from fixed-effect designs only in the way in which effects are tested for significance. In fixed-effect designs, between effects are always tested using the mean square residual as the error term. In mixed-model designs, between effects are tested using relevant error terms based on the covariation of sources of variation in the design. Also, only the overparameterized model is used to code effects for categorical predictors in mixed-models, because the sigma-restricted model is overly restrictive.

The covariation of sources of variation in the design is estimated by the elements of a matrix called the Expected Mean Squares (EMS) matrix. This non-square matrix contains elements for the covariation of each combination of pairs of sources of variation and for each source of variation with *Error*. Specifically, each element is the mean square for one effect (indicated by the column) that is expected to be accounted by another effect (indicated by the row), given the observed covariation in their levels. Note that expected mean squares can be computing using any type of sums of squares from Type I through Type V. Once the EMS matrix is computed, it is used to the solve for the linear combinations of sources of random variation that are appropriate to use as error terms for testing the significance of the respective effects. This is done using Satterthwaite's method of denominator synthesis (Satterthwaite, 1946). Detailed discussions of methods for testing effects in mixed-models, and related methods for estimating variance components for random effects, can be found in the *Variance Components and Mixed Model ANOVA/ANCOVA* chapter.

Testing Specific Hypotheses

Whole model tests and tests based on sums of squares attributable to specific effects illustrate two general types of hypotheses that can be tested using the general linear model. Still, there may be other types of hypotheses the researcher wishes to test that do not fall into either of these categories. For example, hypotheses about subsets of effects may be of interest, or hypotheses involving comparisons of specific levels of categorical predictor variables may be of interest.

Estimability of Hypotheses. Before considering tests of specific hypotheses of this sort, it is important to address the issue of estimability. A test of a specific hypothesis using the general linear model must be framed in terms of the regression coefficients for the solution of the normal equations. If the ***X'X*** matrix is less than full rank, the regression coefficients depend on the particular g2 inverse used for solving the normal equations, and the regression coefficients will not be unique. When the regression coefficients are not unique, linear functions (*f*) of the regression coefficients having the form

$$f = \mathbf{Lb}$$

where ***L*** is a vector of coefficients, will also in general not be unique. However, ***Lb*** for an ***L*** which satisfies

$$\mathbf{L = L(X'X)^{-}X'X}$$

is invariant for all possible g2 inverses, and is therefore called an estimable function.

The theory of estimability of linear functions is an advanced topic in the theory of algebraic invariants (Searle, 1987, provides a comprehensive introduction), but its implications are clear enough. One instance of non-estimability of a hypothesis has been encountered in tests of the effective hypothesis which have zero degrees of freedom. On the other hand, Type III sums of squares for categorical predictor variable effects in ANOVA designs with no missing cells (and the least squares means in such designs) provide an example of estimable functions which do not depend on the model parameterization (i.e., the particular g2 inverse used to solve the normal equations). The general implication of the theory of estimability of linear functions is that hypotheses which cannot be expressed as linear combinations of the rows of ***X*** (i.e., the combinations of observed levels of the categorical predictor variables) are not estimable, and therefore cannot be tested. Stated another way, we simply cannot test specific hypotheses that are not represented in the data. The notion of estimability is valuable because the test for estimability makes explicit which specific hypotheses can be tested and which cannot.

Linear Combinations of Effects. In multiple regression designs, it is common for hypotheses of interest to involve subsets of effects. In mixture designs, for example, one might be interested in simultaneously testing whether the main effect and any of the two-way interactions involving a particular predictor variable are non-zero. It is also common in multiple regression designs for hypotheses of interest to involves comparison of slopes. For example, one might be interested in whether the regression coefficients for two predictor variables differ. In both factorial regression and factorial ANOVA designs with many factors, it is often of interest whether sets of effects, say, all three-way and higher-order interactions, are nonzero.

Tests of these types of specific hypotheses involve (1) constructing one or more ***L***s reflecting the hypothesis, (2) testing the estimability of the hypothesis by determining whether

$$\mathbf{L = L(X'X)^{-}X'X}$$

and if so, using (3)

$$\mathbf{(Lb)'<L(X'X)^{-}L')^{-1}(Lb)}$$

to estimate the sums of squares accounted for by the hypothesis. Finally, (4) the hypothesis is tested for significance using the usual mean square residual as the error term. To illustrate this 4-step procedure, suppose that a test of the difference in the regression slopes is desired for the (intercept plus) 2 predictor variables in a first-order multiple regression design. The coefficients for ***L*** would be

$$\mathbf{L} = [0\ 1\ {-1}]$$

(note that the first coefficient 0 excludes the intercept from the comparison) for which ***Lb*** is estimable if the 2 predictor variables are not redundant with each other. The hypothesis sums of squares reflect the difference in the partial regression coefficients for the 2 predictor variables, which is tested for significance using the mean square residual as the error term.

Planned Comparisons of Least Square Means. Usually, experimental hypotheses are stated in terms that are more specific than simply main effects or interactions. We may have the specific hypothesis that a particular textbook will improve math skills in males, but not in females, while another book would be about equally effective for both genders, but less effective overall for males. Now generally, we are predicting an interaction here: the effectiveness of the book is modified (qualified) by the student's gender. However, we have a particular prediction concerning the nature of the interaction: we expect a significant difference between genders for one book, but not the other. This type of specific prediction is usually tested by testing planned comparisons of least squares means (estimates of the population marginal means), or as it is sometimes called, contrast analysis.

Briefly, contrast analysis allows us to test the statistical significance of predicted specific differences in particular parts of our complex design. The 4-step procedure for testing specific hypotheses is used to specify and test specific predictions. Contrast analysis is a major and indispensable component of the analysis of many complex experimental designs (see also for details).

To learn more about the logic and interpretation of contrast analysis refer to the *ANOVA/MANOVA* chapter *Overview* section.

Post-Hoc Comparisons. Sometimes we find effects in an experiment that were not expected. Even though in most cases a creative experimenter will be able to explain almost any pattern of means, it would not be appropriate to analyze and evaluate that pattern as if one had predicted it all along. The problem here is one of capitalizing on chance when performing multiple tests post-hoc, that is, without a priori hypotheses. To illustrate this point, let us consider the following "experiment." Imagine we were to write down a number between 1 and 10 on 100 pieces of paper. We then put all of those pieces into a hat and draw 20 samples (of pieces of paper) of 5 observations each, and compute the means (from the numbers written on the pieces of paper) for each group. How likely do you think it is that we will find two sample means that are significantly different from each other? It is very likely! Selecting the extreme means obtained from 20 samples is very different from taking only 2 samples from the hat in the first place, which is what the test via the contrast analysis implies. Without going into further detail, there are several so-called post-hoc tests that are explicitly based on the first scenario (taking the extremes from 20 samples), that is, they are based on the assumption that we have chosen for our comparison the most extreme (different) means out of *k* total means in the design. Those tests apply "corrections" that are designed to offset the advantage of post-hoc selection of the most extreme comparisons. Whenever one finds unexpected results in an experiment one should use those post-hoc procedures to test their statistical significance.

Testing hypotheses for repeated measures and dependent variables

In the discussion of different hypotheses that can be tested using the general linear model, the tests have been described as tests for "the dependent variable" or "the outcome." This has been done solely to simplify the discussion. When there are multiple dependent variables reflecting the levels of repeated measure factors, the general linear model performs tests using orthonormalized ***M***-transformations of the dependent variables. When there are multiple dependent variables but no repeated measure factors, the general linear model performs tests using the hypothesis sums of squares and cross-products for the multiple dependent variables, which are tested against the residual sums of squares and cross-products for the multiple dependent variables. Thus, the same hypothesis testing procedures which apply to univariate designs with a single dependent variable also apply to repeated measure and multivariate designs.

Chapter 19
Generalized Additive Models (GAM)

- Additive models
- Generalized linear models
- Distributions and link functions
- Generalized additive models
- Estimating the non-parametric function of predictors via scatterplot smoothers
- A specific example: The generalized additive logistic model
- Fitting generalized additive models
- Interpreting the results
- Degrees of freedom
- A Word of Caution

The methods available in *Generalized Additive Models* are implementations of techniques developed and popularized by Hastie and Tibshirani (1990). A detailed description of these and related techniques, the algorithms used to fit these models, and discussions of recent research in this area of statistical modeling can also be found in Schimek (2000).

Additive models. The methods described in this section represent a generalization of multiple regression (which is a special case of general linear models). Specifically, in linear regression, a linear least-squares fit is computed for a set of predictor or X variables, to predict a dependent Y variable. The well known linear regression equation with m predictors, to predict a dependent variable Y, can be stated as:

$$\mathbf{Y = b_0 + b_1 * X_1 + ... + b_m * X_m}$$

Where Y stands for the (predicted values of the) dependent variable, X_1through X_m represent the m values for the predictor variables, and b_0, and b_1 through b_m are the regression coefficients estimated by multiple regression. A generalization of the multiple regression model would be to maintain the additive nature of the model, but to replace the simple terms of the linear equation b_i*X_i with $f_i(X_i)$ where f_i is a non-parametric function of the predictor X_i. In other words, instead of a single coefficient for each variable (additive term) in the model, in additive models an unspecified (non-parametric) function is estimated for each predictor, to achieve the best prediction of the dependent variable values.

Generalized linear models. To summarize the basic idea, the generalized linear model differs from the general linear model (of which multiple regression is a special case) in two major respects: First, the distribution of the dependent or response variable can be (explicitly) non-normal, and does not have to be continuous, e.g., it can be binomial; second, the dependent variable values are predicted from a linear combination of predictor variables, which are "connected" to the dependent variable via a link function. The general linear model for a single dependent variable can be considered a special case of the generalized linear model: In the general linear model the dependent variable values are expected to follow the normal distribution, and the link function is a simple identity function (i.e., the linear combination of values for the predictor variables is not transformed).

To illustrate, in the general linear model a response variable Y is linearly associated with values on the X variables while the relationship in the generalized linear model is assumed to be

$$\mathbf{Y = g(b_0 + b_1*X_1 + ... + b_m*X_m)}$$

where *g(…)* is a function. Formally, the inverse function of g(…), say gi(…), is called the link function; so that:

$$\mathbf{gi(muY) = b_0 + b_1*X_1 + ... + b_m*X_m}$$

where *mu-Y* stands for the expected value of Y.

Distributions and link functions. *Generalized Additive Models* allows you to choose from a wide variety of distributions for the dependent variable, and link functions for the effects of the predictor variables on the dependent variable (see McCullagh and Nelder, 1989; Hastie and Tibshirani, 1990; see also *GLZ Introductory Overview* - Computational Approach for a discussion of link functions and distributions):

Normal, *Gamma*, and Poisson distributions:

Log link: **f(z) = log(z)**

Inverse link: **f(z) = 1/z**

Identity link: **f(z) = z**

Binomial distributions:

Logit link: **f(z)=log(z/(1-z))**

Generalized additive models. We can combine the notion of additive models with generalized linear models, to derive the notion of generalized additive models, as:

$$gi(muY) = \square_i(f_i(X_i))$$

In other words, the purpose of generalized additive models is to maximize the quality of prediction of a dependent variable Y from various distributions, by estimating unspecific (non-parametric) functions of the predictor variables which are "connected" to the dependent variable via a link function.

Estimating the non-parametric function of predictors via scatterplot smoothers. A unique aspect of generalized additive models are the non-parametric functions f_i of the predictor variables X_i. Specifically, instead of some kind of simple or complex parametric functions, Hastie and Tibshirani (1990) discuss various general scatterplot smoothers that can be applied to the X variable values, with the target criterion to maximize the quality of prediction of the (transformed) Y variable values. One such scatterplot smoother is the cubic smoothing splines smoother, which generally produces a smooth generalization of the relationship between the two variables in the scatterplot. Computational details regarding this smoother can be found in Hastie and Tibshirani (1990; see also Schimek, 2000).

To summarize, instead of estimating single parameters (like the regression weights in multiple regression), in generalized additive models, we find a general unspecific (non-parametric) function that relates the predicted (transformed) Y values to the predictor values.

A specific example: The generalized additive logistic model. Let us consider a specific example of the generalized additive models: A generalization of the logistic (logit) model for binary dependent variable values. As also described in detail in the context of *Nonlinear Estimation* and *Generalized Linear/Nonlinear Models*, the logistic regression model for binary responses can be written as follows:

$$y=exp(b_0+b_1*x_1+...+b_m*x_m)/\{1+exp(b_0+b_1*x_1+...+b_m*x_m)\}$$

Note that the distribution of the dependent variable is assumed to be binomial, i.e., the response variable can only assume the values 0 or 1 (e.g., in a market research study, the purchasing decision would be binomial: The customer either did or did not make a particular purchase). We can apply the logistic link function to the probability p (ranging between 0 and 1) so that:

$$p' = \log \{p/(1-p)\}$$

By applying the logistic link function, we can now rewrite the model as:

$$p' = b_0 + b_1*X_1 + ... + b_m*X_m$$

Finally, we substitute the simple single-parameter additive terms to derive the generalized additive logistic model:

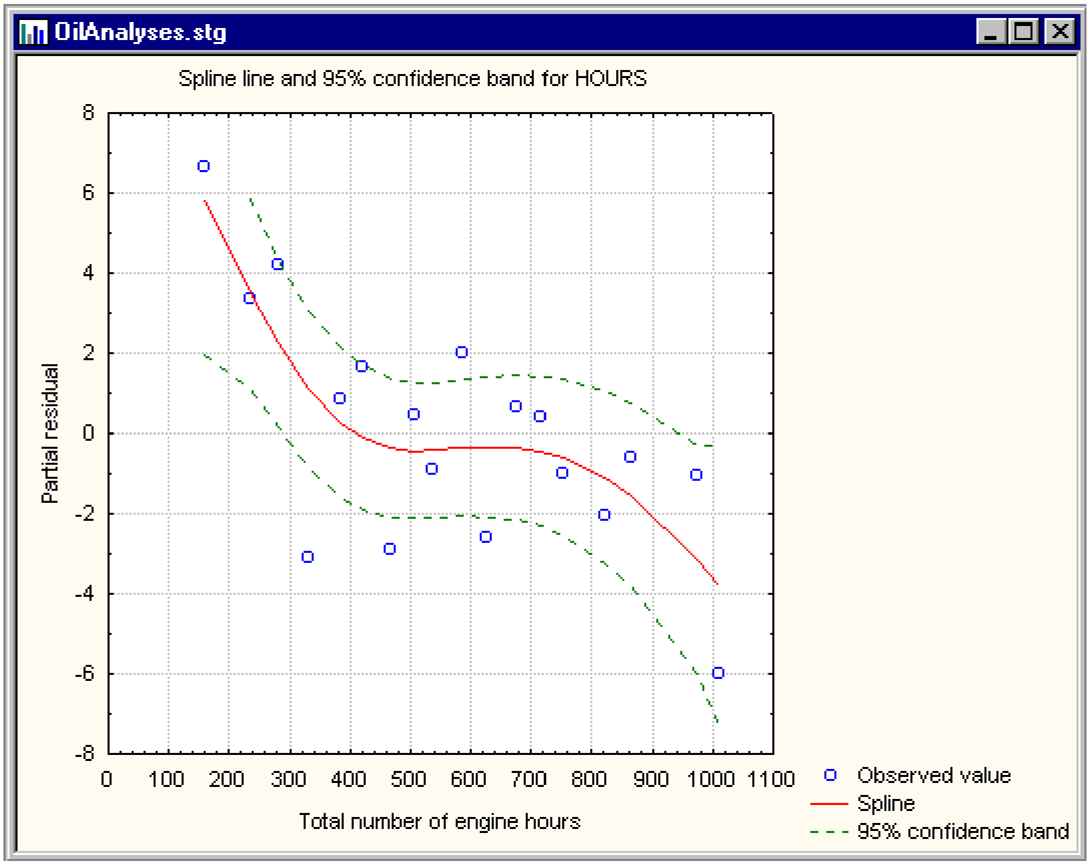
$$p' = b_0 + f_1(X_1) + ... + f_m(X_m)$$

An example application of the this model can be found in Hastie and Tibshirani (1990).

Fitting generalized additive models. Detailed descriptions of how generalized additive models are fit to data can be found in Hastie and Tibshirani (1990), as well as Schimek (2000, p. 300). In general there are two separate iterative operations involved in the algorithm, which are usually labeled the *outer* and *inner* loop. The purpose of the outer loop is to maximize the overall fit of the model, by minimizing the overall likelihood of the data given the model (similar to the maximum likelihood estimation procedures as described in, for example, the context of Nonlinear Estimation). The purpose of the inner loop is to refine the scatterplot smoother, which is the cubic splines smoother. The smoothing is performed with respect to the partial residuals; i.e., for every predictor k, the weighted cubic spline fit is found that best represents the relationship between variable k and the (partial) residuals computed by removing the effect of all other j predictors (j □ k). The iterative estimation procedure will terminate, when the likelihood of the data given the model can not be improved.

Interpreting the results. Many of the standard results statistics computed by *Generalized Additive Models* are similar to those customarily reported by linear or nonlinear model fitting procedures. For example, predicted and residual values for the final model can be computed, and various graphs of the residuals can be displayed to help the user identify possible outliers, etc. Refer also to the description of the residual statistics computed by *Generalized Linear/Nonlinear Models* for details.

The main result of interest, of course, is how the predictors are related to the dependent variable. Scatterplots can be computed showing the smoothed predictor variable values plotted against the partial residuals, i.e., the residuals after removing the effect of all other predictor variables.



This plot allows you to evaluate the nature of the relationship between the predictor with the residualized (adjusted) dependent variable values (see Hastie & Tibshirani, 1990; in particular formula 6.3), and hence the nature of the influence of the respective predictor in the overall model.

Degrees of freedom. To reiterate, the generalized additive models approach replaces the simple products of (estimated) parameter values times the predictor values with a cubic spline smoother for each predictor. When estimating a single parameter value, we lose one degree of freedom, i.e., we add one degree of freedom to the overall model. It is not clear how many degrees of freedom are lost due to estimating the cubic spline smoother for each variable. Intuitively, a smoother can either be very smooth, not following the pattern of data in the scatterplot very closely, or it can be less smooth, following the pattern of the data more closely. In the most extreme case, a simple line would be very smooth, and require us to estimate a single slope parameter, i.e., we would use one degree of freedom to fit the smoother (simple straight line); on the other hand, we could force a very "non-smooth" line to connect each actual data point, in which case we could "use-up" approximately as many degrees of freedom as there are points in the plot. Generalized Additive Models allows you to specify the degrees of freedom for the cubic spline smoother; the fewer degrees of freedom you specify, the smoother is the cubic spline fit to the partial residuals, and typically, the worse is the overall fit of the model. The issue of degrees of freedom for smoothers is discussed in detail in Hastie and Tibshirani (1990).

A word of caution. Generalized additive models are very flexible, and can provide an excellent fit in the presence of nonlinear relationships and significant noise in the predictor variables. However, note that because of this flexibility, one must be extra cautious not to over-fit the data, i.e., apply an overly complex model (with many degrees of freedom) to data so as to produce a good fit that likely will not replicate in subsequent validation studies. Also, compare the quality of the fit obtained from *Generalized Additive Models* to the fit obtained via *Generalized Linear/Nonlinear Models*. In other words, evaluate whether the added complexity (generality) of generalized additive models (regression smoothers) is necessary in order to obtain a satisfactory fit to the data. Often, this is not the case, and given a comparable fit of the models, the simpler generalized linear model is preferable to the more complex generalized additive model. These issues are discussed in greater detail in Hastie and Tibshirani (1990).

Another issue to keep in mind pertains to the interpretability of results obtained from (generalized) linear models vs. generalized additive models. Linear models are easily understood, summarized, and communicated to others (e.g., in technical reports). Moreover, parameter estimates can be used to predict or classify new cases in a simple and straightforward manner. Generalized additive models are not easily interpreted, in particular when they involve complex nonlinear effects of some or all of the predictor variables (and, of course, it is in those instances where generalized additive models may yield a better fit than generalized linear models). To reiterate, it is usually preferable to rely on a simple well understood model for predicting future cases, than on a complex model that is difficult to interpret and summarize.

Chapter 20
Generalized Linear Models (GLZ)

- Basic Ideas
- Computational Approach
- Types of Analyses
    - Between-subject Designs
- Model Building
- Interpretation of Results and Diagnostics

This chapter describes the use of the generalized linear model for analyzing linear and non-linear effects of continuous and categorical predictor variables on a discrete or continuous dependent variable. If you are unfamiliar with the basic methods of regression in linear models, it may be useful to first review the basic information on these topics in the *Elementary Concepts* chapter. Discussion of the ways in which the linear regression model is extended by the general linear model can be found in the General Linear Models chapter.

For additional information about generalized linear models, see also Dobson (1990), Green and Silverman (1994), or McCullagh and Nelder (1989).

Basic Ideas

The Generalized Linear Model (GLZ) is a generalization of the general linear model (see, e.g., the *General Linear Models*, *Multiple Regression*, and *ANOVA/MANOVA* chapters). In its simplest form, a linear model specifies the (linear) relationship between a dependent (or response) variable *Y*, and a set of predictor variables, the *X*'s, so that

$$Y = b_0 + b_1X_1 + b_2X_2 + ... + b_kX_k$$

In this equation b_0 is the regression coefficient for the intercept and the b_i values are the regression coefficients (for variables 1 through *k*) computed from the data.

So for example, one could estimate (i.e., predict) a person's weight as a function of the person's height and gender. You could use linear regression to estimate the respective regression coefficients from a sample of data, measuring height, weight, and observing the subjects' gender. For many data analysis problems, estimates of the linear relationships between variables are adequate to describe the observed data, and to make reasonable predictions for new observations (see the *Multiple Regression* chapter for additional details).

However, there are many relationships that cannot adequately be summarized by a simple linear equation, for two major reasons:

Distribution of dependent variable. First, the dependent variable of interest may have a non-continuous distribution, and thus, the predicted values should also follow the respective distribution; any other predicted values are not logically possible. For example, a researcher may be interested in predicting one of three possible discrete outcomes (e.g., a consumer's choice of one of three alternative products). In that case, the dependent variable can only take on 3 distinct values, and the distribution of the dependent variable is said to be *multinomial*. Or suppose you are trying to predict people's family planning choices, specifically, how many children families will have, as a function of income and various other socioeconomic indicators. The dependent variable -- number of children -- is discrete (i.e., a family may have 1, 2, or 3 children and so on, but cannot have 2.4 children), and most likely the distribution of that variable is highly skewed (i.e., most families have 1, 2, or 3 children, fewer will have 4 or 5, very few will have 6 or 7, and so on). In this case it would be reasonable to assume that the dependent variable follows a Poisson distribution.

Link function. A second reason why the linear (multiple regression) model might be inadequate to describe a particular relationship is that the effect of the predictors on the dependent variable may not be linear in nature. For example, the relationship between a person's age and various indicators of health is most likely not linear in nature: During early adulthood, the (average) health status of people who are 30 years old as compared to the (average) health status of people who are 40 years old is not markedly different. However, the difference in health status of 60 year old people and 70 year old people is probably greater. Thus, the relationship between age and health status is likely non-linear in nature. Probably some kind of a power function would be adequate to describe the relationship between a person's age and health, so that each increment in years of age at older ages will have greater impact on health status, as compared to each increment in years of age during early adulthood. Put in other words, the *link* between age and health status is best described as non-linear, or as a power relationship in this particular example.

The generalized linear model can be used to predict responses both for dependent variables with discrete distributions and for dependent variables which are nonlinearly related to the predictors.

Computational Approach

To summarize the *basic ideas*, the generalized linear model differs from the general linear model (of which, for example, multiple regression is a special case) in two major respects: First, the distribution of the dependent or response variable can be (explicitly) non-normal, and does not have to be continuous, i.e., it can be binomial, multinomial, or ordinal multinomial (i.e., contain information on ranks only); second, the dependent variable values are predicted from a linear combination of predictor variables, which are "connected" to the dependent variable via a link function. The general linear model for a single dependent variable can be considered a special case of the generalized linear model: In the general linear model the dependent variable values are expected to follow the normal distribution, and the link function is a simple identity function (i.e., the linear combination of values for the predictor variables is not transformed).

To illustrate, in the general linear model a response variable *Y* is linearly associated with values on the *X* variables by

$$Y = b_0 + b_1X_1 + b_2X_2 + ... + b_kX_k) + e$$

(where *e* stands for the error variability that cannot be accounted for by the predictors; note that the expected value of *e* is assumed to be 0), while the relationship in the generalized linear model is assumed to be

$$Y = g (b_0 + b_1X_1 + b_2X_2 + ... + b_kX_k + e$$

where *e* is the error, and *g(…)* is a function. Formally, the inverse function of *g(…)*, say *f(…)*, is called the link function; so that:

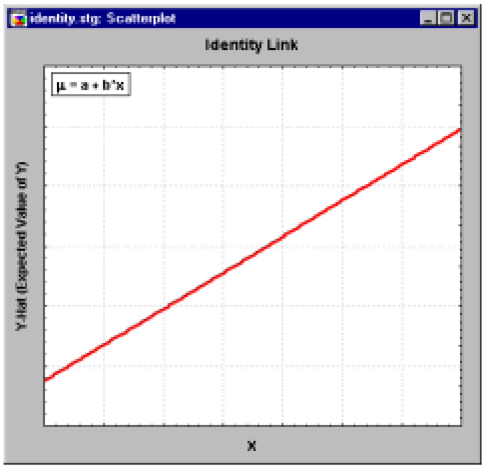
$$f(mu_y) = b_0 + b_1X_1 + b_2X_2 + ... + b_kX_k$$

where mu_y stands for the expected value of *y*.

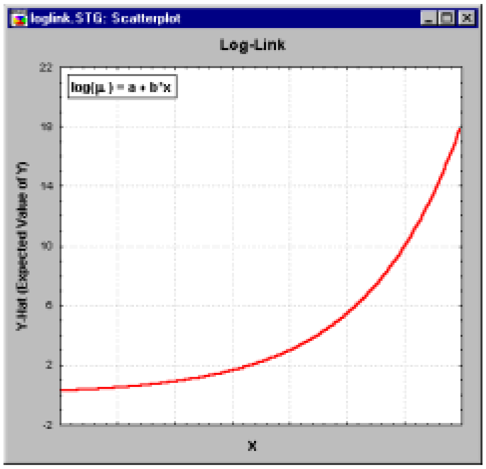
Link functions and distributions. Various link functions (see McCullagh and Nelder, 1989) can be chosen, depending on the assumed distribution of the *y* variable values:

Normal, Gamma, Inverse normal, and Poisson distributions:

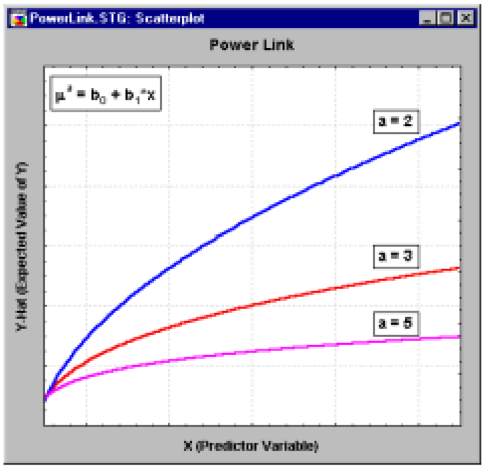
Identity link: $\mathbf{f(z) = z}$



Log link: **f(z) = log(z)**



Power link: **f(z) = z^a,** for a given **a**

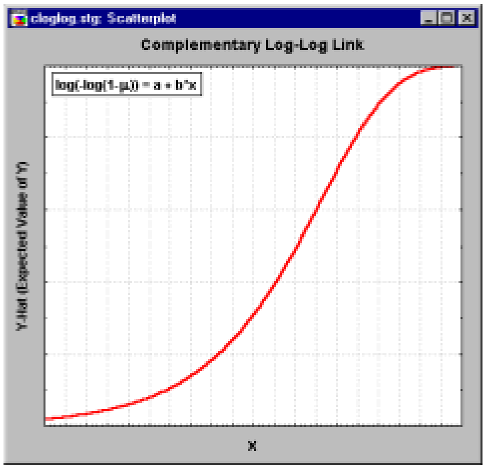


Binomial, *and Ordinal Multinomial* distributions:

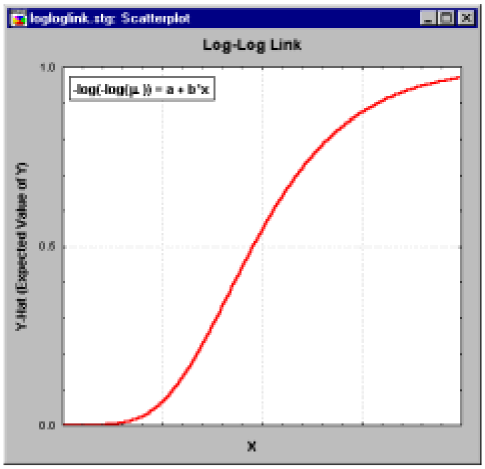
Logit link: **f(z)=log(z/(1-z))**

Probit link: **f(z)=invnorm(z)** where *invnorm* is the inverse of the standard normal cumulative distribution function.

Complementary log-log link: **f(z)=log(-log(1-z))**



Log-log link: **f(z)=-log(-log(z))**



Multinomial distribution:

Generalized logit link: **f(z1|z2,…,zc)=log(x1/(1-z1-…-zc))**

where the model has **c+1** categories.

Estimation in the generalized linear model. The values of the parameters (b_0 through b_k and the scale parameter) in the generalized linear model are obtained by maximum likelihood (ML) estimation, which requires iterative computational procedures. There are many iterative methods for ML estimation in the generalized linear model, of which the

Newton-Raphson and Fisher-Scoring methods are among the most efficient and widely used (see Dobson,1990). The Fisher-scoring (or iterative re-weighted least squares) method in particular provides a unified algorithm for all generalized linear models, as well as providing the expected variance-covariance matrix of parameter estimates as a byproduct of its computations.

Statistical significance testing Tests for the significance of the effects in the model can be performed via the Wald statistic, the likelihood ratio (LR), or score statistic. Detailed descriptions of these tests can be found in McCullagh and Nelder (1989). The Wald statistic (e.g., see Dobson,1990), which is computed as the generalized inner product of the parameter estimates with the respective variance-covariance matrix, is an easily computed, efficient statistic for testing the significance of effects. The score statistic is obtained from the generalized inner product of the score vector with the Hessian matrix (the matrix of the second-order partial derivatives of the maximum likelihood parameter estimates). The likelihood ratio (LR) test requires the greatest computational effort (another iterative estimation procedure) and is thus not as fast as the first two methods; however, the LR test provides the most asymptotically efficient test known. For details concerning these different test statistics, see Agresti(1996), McCullagh and Nelder(1989), and Dobson(1990).

Diagnostics in the generalized linear model. The two basic types of residuals are the so-called Pearson residuals and deviance residuals. Pearson residuals are based on the difference between observed responses and the predicted values; deviance residuals are based on the contribution of the observed responses to the log-likelihood statistic. In addition, leverage scores, studentized residuals, generalized Cook's D, and other observational statistics (statistics based on individual observations) can be computed. For a description and discussion of these statistics, see Hosmer and Lemeshow (1989).

Types of Analyses

The design for an analysis can include effects for continuous as well as categorical predictor variables. Designs may include polynomials for continuous predictors (e.g., squared or cubic terms) as well as interaction effects (i.e., product terms) for continuous predictors. For categorical predictor variables, one can fit ANOVA-like designs, including full factorial, nested, and fractional factorial designs, etc. Designs can be incomplete (i.e., involve missing cells), and effects for categorical predictor variables can be represented using either the sigma-restricted parameterization or the overparameterized (i.e., indicator variable) representation of effects.

The topics below give complete descriptions of the types of designs that can be analyzed using the generalized linear model, as well as types of designs that can be analyzed using the general linear model.

Signal detection theory. The list of designs shown below is by no means comprehensive, i.e., it does not describe all possible research problems to which the generalized linear model can be applied. For example, an important application of the generalized linear model is the estimation of parameters for *Signal detection theory* (SDT) models. SDT is an application of statistical decision theory used to detect a signal embedded in noise. SDT is used in psychophysical studies of detection, recognition, and discrimination, and in other areas such as medical research, weather forecasting, survey research, and marketing research. For example, DeCarlo (1998) shows how *signal detection* models based on different underlying distributions can easily be considered by using the generalized linear model with different link functions.

For discussion of the generalized linear model and the link functions which it uses, see Computational Approaches.

Between-Subject Designs

- Overview
- One-way ANOVA
- Main effect ANOVA
- Factorial ANOVA
- Nested designs
- Simple regression
- Multiple regression
- Factorial regression
- Polynomial regression
- Response surface regression
- Mixture surface regression
- Analysis of covariance (ANCOVA)
- Separate slopes designs
- Homogeneity of slopes

Overview. The levels or values of the predictor variables in an analysis describe the differences between the *n* subjects or the *n* valid cases that are analyzed. Thus, when we speak of the between subject design (or simply the between design) for an analysis, we are referring to the nature, number, and arrangement of the predictor variables.

Concerning the nature or type of predictor variables, between designs which contain only categorical predictor variables can be called ANOVA (analysis of variance) designs, between designs which contain only continuous predictor variables can be called regression designs, and between designs which contain both categorical and continuous predictor variables can be called ANCOVA (analysis of covariance) designs. Further, continuous predictors are always considered to have fixed values, but the levels of categorical predictors can be considered to be fixed or to vary randomly. Designs which contain random categorical factors are called mixed-model designs (see the *Variance Components and Mixed Model ANOVA/ANCOVA* chapter).

Between designs may involve only a single predictor variable and therefore be described as simple (e.g., simple regression) or may employ numerous predictor variables (e.g., multiple regression).

Concerning the arrangement of predictor variables, some between designs employ only "main effect" or first-order terms for predictors, that is, the values for different predictor variables are independent and raised only to the first power. Other between designs may employ higher-order terms for predictors by raising the values for the original predictor variables to a power greater than 1 (e.g., in polynomial regression designs), or by forming products of different predictor variables (i.e., interaction terms). A common arrangement for ANOVA designs is the full-factorial design, in which every combination of levels for each of the categorical predictor variables is represented in the design. Designs with some but not all combinations of levels for each of the categorical predictor variables are aptly called fractional factorial designs. Designs with a hierarchy of combinations of levels for the different categorical predictor variables are called nested designs.

These basic distinctions about the nature, number, and arrangement of predictor variables can be used in describing a variety of different types of between designs. Some of the more common between designs can now be described.

One-Way ANOVA. A design with a single categorical predictor variable is called a one-way ANOVA design. For example, a study of 4 different fertilizers used on different individual plants could be analyzed via one-way ANOVA, with four levels for the factor *Fertilizer*.

In genera, consider a single categorical predictor variable *A* with 1 case in each of its 3 categories. Using the sigma-restricted coding of A into 2 quantitative contrast variables, the matrix ***X*** defining the between design is

$$\mathbf{X} = \begin{array}{c} \\ A_1 \\ A_2 \\ A_3 \end{array} \begin{array}{c} \begin{array}{ccc} X_0 & X_1 & X_2 \end{array} \\ \left[\begin{array}{ccc} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & -1 & -1 \end{array}\right] \end{array}$$

That is, cases in groups A_1, A_2, and A_3 are all assigned values of 1 on X_0 (the intercept), the case in group A_1 is assigned a value of 1 on X_1 and a value 0 on X_2, the case in group A_2 is assigned a value of 0 on X_1 and a value 1 on X_2, and the case in group A_3 is assigned a value of -1 on X_1 and a value -1 on X_2. Of course, any additional cases in any of the 3 groups would be coded similarly. If there were 1 case in group A_1, 2 cases in group A_2, and 1 case in group A_3, the ***X*** matrix would be

$$\mathbf{X} = \begin{array}{c} \\ A_{11} \\ A_{12} \\ A_{22} \\ A_{13} \end{array} \begin{array}{ccc} X_0 & X_1 & X_2 \\ \left[\begin{array}{c} 1 \\ 1 \\ 1 \\ 1 \end{array}\right. & \begin{array}{c} 1 \\ 0 \\ 0 \\ -1 \end{array} & \left.\begin{array}{c} 0 \\ 1 \\ 1 \\ -1 \end{array}\right] \end{array}$$

where the first subscript for *A* gives the replicate number for the cases in each group. For brevity, replicates usually are not shown when describing ANOVA design matrices.

Note that in one-way designs with an equal number of cases in each group, sigma-restricted coding yields $X_1 \ldots X_k$ variables all of which have means of 0.

Using the overparameterized model to represent A, the ***X*** matrix defining the between design is simply

$$\mathbf{X} = \begin{array}{c} \\ A_1 \\ A_2 \\ A_3 \end{array} \begin{array}{cccc} X_0 & X_1 & X_2 & X_3 \\ \left[\begin{array}{c} 1 \\ 1 \\ 1 \end{array}\right. & \begin{array}{c} 1 \\ 0 \\ 0 \end{array} & \begin{array}{c} 0 \\ 1 \\ 0 \end{array} & \left.\begin{array}{c} 0 \\ 0 \\ 1 \end{array}\right] \end{array}$$

These simple examples show that the ***X*** matrix actually serves two purposes. It specifies (1) the coding for the levels of the original predictor variables on the *X* variables used in the analysis as well as (2) the nature, number, and arrangement of the X variables, that is, the between design.

Main Effect ANOVA. Main effect ANOVA designs contain separate one-way ANOVA designs for 2 or more categorical predictors. A good example of main effect ANOVA would be the typical analysis performed on *screening designs* as described in the context of the *Experimental Design* chapter.

Consider 2 categorical predictor variables *A* and *B* each with 2 categories. Using the sigma-restricted coding, the ***X*** matrix defining the between design is

$$\mathbf{X} = \begin{array}{c} \\ A_1B_1 \\ A_1B_2 \\ A_2B_1 \\ A_2B_2 \end{array} \begin{array}{ccc} X_0 & X_1 & X_2 \\ \left[\begin{array}{c} 1 \\ 1 \\ 1 \\ 1 \end{array}\right. & \begin{array}{c} 1 \\ 1 \\ -1 \\ -1 \end{array} & \left.\begin{array}{c} 1 \\ -1 \\ 1 \\ -1 \end{array}\right] \end{array}$$

Note that if there are equal numbers of cases in each group, the sum of the cross-products of values for the X_1 and X_2 columns is 0, for example, with 1 case in each group (1*1)+(1*-1)+(-1*1)+(-1*-1)=0. Using the overparameterized model, the matrix ***X*** defining the between design is

$$\mathbf{X} = \begin{array}{c|ccccc|} & X_0 & X_1 & X_2 & X_3 & X_4 \\ A_1B_1 & 1 & 1 & 0 & 1 & 0 \\ A_1B_2 & 1 & 1 & 0 & 0 & 1 \\ A_2B_1 & 1 & 0 & 1 & 1 & 0 \\ A_2B_2 & 1 & 0 & 1 & 0 & 1 \end{array}$$

Comparing the two types of coding, it can be seen that the overparameterized coding takes almost twice as many values as the sigma-restricted coding to convey the same information.

Factorial ANOVA. Factorial ANOVA designs contain *X* variables representing combinations of the levels of 2 or more categorical predictors (e.g., a study of boys and girls in four age groups, resulting in a *2 (Gender) x 4 (Age Group)* design). In particular, full-factorial designs represent all possible combinations of the levels of the categorical predictors. A full-factorial design with 2 categorical predictor variables *A* and *B* each with 2 levels each would be called a 2 x 2 full-factorial design. Using the sigma-restricted coding, the ***X*** matrix for this design would be

$$\mathbf{X} = \begin{array}{c|cccc|} & X_0 & X_1 & X_2 & X_3 \\ A_1B_1 & 1 & 1 & 1 & 1 \\ A_1B_2 & 1 & 1 & -1 & -1 \\ A_2B_1 & 1 & -1 & 1 & -1 \\ A_2B_2 & 1 & -1 & -1 & 1 \end{array}$$

Several features of this ***X*** matrix deserve comment. Note that the X_1 and X_2 columns represent main effect contrasts for one variable, (i.e., *A* and *B*, respectively) collapsing across the levels of the other variable. The X_3 column instead represents a contrast between different combinations of the levels of *A* and *B*. Note also that the values for X_3 are products of the corresponding values for X_1 and X_2. Product variables such as X_3 represent the multiplicative or interaction effects of their factors, so X_3 would be said to represent the 2-way interaction of *A* and *B*. The relationship of such product variables to the dependent variables indicate the interactive influences of the factors on responses above and beyond their independent (i.e., main effect) influences on responses. Thus, factorial designs provide more information about the relationships between categorical predictor variables and responses on the dependent variables than is provided by corresponding one-way or main effect designs.

When many factors are being investigated, however, full-factorial designs sometimes require more data than reasonably can be collected to represent all possible combinations of levels of the factors, and high-order interactions between many factors can become difficult to interpret. With many factors, a useful alternative to the full-factorial design is the fractional factorial design. As an example, consider a 2 x 2 x 2 fractional factorial design to degree 2 with 3 categorical predictor variables each with 2 levels. The design would include the main effects for each variable, and all 2-way interactions between the three variables, but would not include the 3-way interaction between all three variables. Using the overparameterized model, the ***X*** matrix for this design is

$$\mathbf{X} = \begin{array}{c} \\ A_1B_1C_1 \\ A_1B_2C_2 \\ A_1B_2C_1 \\ A_1B_2C_2 \\ A_2B_1C_1 \\ A_2B_1C_2 \\ A_2B_2C_1 \\ A_2B_2C_2 \end{array}$$

$$\mathbf{X} \;=\; \begin{array}{c|ccccccccccccccccccc|}
 & \multicolumn{7}{c}{\text{......main effects......}} & \multicolumn{12}{c}{\text{...............2 - way interactio ns..........}} \\
A_1B_1C_1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
A_1B_1C_2 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
A_1B_2C_1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
A_1B_2C_2 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
A_2B_1C_1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
A_2B_1C_2 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\
A_2B_2C_1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
A_2B_2C_2 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1
\end{array}$$

The 2-way interactions are the highest degree effects included in the design. These types of designs are discussed in detail the *2**(k-p) Fractional Factorial Designs* section of the *Experimental Design* chapter.

Nested ANOVA Designs. Nested designs are similar to fractional factorial designs in that all possible combinations of the levels of the categorical predictor variables are not represented in the design. In nested designs, however, the omitted effects are lower-order effects. Nested effects are effects in which the nested variables never appear as main effects. Suppose that for 2 variables *A* and *B* with 3 and 2 levels, respectively, the design includes the main effect for *A* and the effect of *B* nested within the levels of *A*. The ***X*** matrix for this design using the overparameterized model is

$$\mathbf{X} \;=\; \begin{array}{c|cccccccccc|}
 & X_0 & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 & X_8 & X_9 \\
A_1B_1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
A_1B_2 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
A_2B_1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
A_2B_2 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
A_3B_1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
A_3B_2 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}$$

Note that if the sigma-restricted coding were used, there would be only 2 columns in the ***X*** matrix for the *B* nested within *A* effect instead of the 6 columns in the ***X*** matrix for this effect when the overparameterized model coding is used (i.e., columns X_4 through X_9). The sigma-restricted coding method is overly-restrictive for nested designs, so only the overparameterized model is used to represent nested designs.

Simple Regression. Simple regression designs involve a single continuous predictor variable. If there were 3 cases with values on a predictor variable *P* of, say, 7, 4, and 9, and the design is for the first-order effect of *P*, the ***X*** matrix would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{cc} X_0 & X_1 \end{array} \\ \begin{bmatrix} 1 & 7 \\ 1 & 4 \\ 1 & 9 \end{bmatrix} \end{array}$$

and using *P* for X_1 the regression equation would be

$$Y = b_0 + b_1P$$

If the simple regression design is for a higher-order effect of *P*, say the quadratic effect, the values in the X_1 column of the design matrix would be raised to the 2nd power, that is, squared

$$\mathbf{X} = \begin{array}{c} \begin{array}{cc} X_0 & X_1 \end{array} \\ \begin{bmatrix} 1 & 49 \\ 1 & 16 \\ 1 & 81 \end{bmatrix} \end{array}$$

and using P^2 for X_1 the regression equation would be

$$Y = b_0 + b_1P^2$$

The sigma-restricted and overparameterized coding methods do not apply to simple regression designs and any other design containing only continuous predictors (since there are no categorical predictors to code). Regardless of which coding method is chosen, values on the continuous predictor variables are raised to the desired power and used as the values for the *X* variables. No recoding is performed. It is therefore sufficient, in describing regression designs, to simply describe the regression equation without explicitly describing the design matrix ***X***.

Multiple Regression. Multiple regression designs are to continuous predictor variables as main effect ANOVA designs are to categorical predictor variables, that is, multiple regression designs contain the separate simple regression designs for 2 or more continuous predictor variables. The regression equation for a multiple regression design for the first-order effects of 3 continuous predictor variables *P*, *Q*, and *R* would be

$$Y = b_0 + b_1P + b_2Q + b_3R$$

Factorial Regression. Factorial regression designs are similar to factorial ANOVA designs, in which combinations of the levels of the factors are represented in the design. In factorial regression designs, however, there may be many more such possible combinations of distinct levels for the continuous predictor variables than there are cases in the data set. To simplify matters, full-factorial regression designs are defined as designs in which all possible products of the continuous predictor variables are represented in the design. For example, the full-factorial regression design for two continuous predictor variables *P* and *Q* would include the main effects (i.e., the first-order effects) of *P* and *Q* and their 2-way *P* by *Q* interaction effect, which is represented by the product of *P* and *Q* scores for each case. The regression equation would be

$$Y = b_0 + b_1P + b_2Q + b_3P*Q$$

Factorial regression designs can also be fractional, that is, higher-order effects can be omitted from the design. A fractional factorial design to degree 2 for 3 continuous predictor variables *P*, *Q*, and *R* would include the main effects and all 2-way interactions between the predictor variables

$$Y = b_0 + b_1P + b_2Q + b_3R + b_4P*Q + b_5P*R + b_6Q*R$$

Polynomial Regression. Polynomial regression designs are designs which contain main effects and higher-order effects for the continuous predictor variables but do not include interaction effects between predictor variables. For example, the polynomial regression design to degree 2 for three continuous predictor variables *P, Q,* and *R* would include the main effects (i.e., the first-order effects) of *P, Q,* and *R* and their quadratic (i.e., second-order) effects, but not the 2-way interaction effects or the *P* by *Q* by *R* 3-way interaction effect.

$$Y = b_0 + b_1P + b_2P^2 + b_3Q + b_4Q^2 + b_5R + b_6R^2$$

Polynomial regression designs do not have to contain all effects up to the same degree for every predictor variable. For example, main, quadratic, and cubic effects could be included in the design for some predictor variables, and effects up the fourth degree could be included in the design for other predictor variables.

Response Surface Regression. Quadratic response surface regression designs are a hybrid type of design with characteristics of both polynomial regression designs and fractional factorial regression designs. Quadratic response surface regression designs contain all the same effects of polynomial regression designs to degree 2 and additionally the 2-way interaction effects of the predictor variables. The regression equation for a quadratic response surface regression design for 3 continuous predictor variables *P, Q,* and *R* would be

$$Y = b_0 + b_1P + b_2P^2 + b_3Q + b_4Q^2 + b_5R + b_6R^2 + b_7P*Q + b_8P*R + b_9Q*R$$

These types of designs are commonly employed in applied research (e.g., in industrial experimentation), and a detailed discussion of these types of designs is also presented in the *Experimental Design* chapter (see *Central composite designs*).

Mixture Surface Regression. Mixture surface regression designs are identical to factorial regression designs to degree 2 except for the omission of the intercept. Mixtures, as the name implies, add up to a constant value; the sum of the proportions of ingredients in different recipes for some material all must add up 100%. Thus, the proportion of one ingredient in a material is redundant with the remaining ingredients. Mixture surface regression designs deal with this redundancy by omitting the intercept from the design. The design matrix for a mixture surface regression design for 3 continuous predictor variables *P*, *Q*, and *R* would be

$$Y = b_1P + b_2Q + b_3R + b_4P*Q + b_5P*R + b_6Q*R$$

These types of designs are commonly employed in applied research (e.g., in industrial experimation), and a detailed discussion of these types of designs is also presented in the *Experimental Design* chapter (see *Mixture designs and triangular surfaces*).

Analysis of Covariance. In general, between designs which contain both categorical and continuous predictor variables can be called ANCOVA designs. Traditionally, however, ANCOVA designs have referred more specifically to designs in which the first-order effects of one or more continuous predictor variables are taken into account when assessing the effects of one or more categorical predictor variables. A basic introduction to analysis of covariance can also be found in the *Analysis of covariance (ANCOVA)* topic of the *ANOVA/MANOVA* chapter.

To illustrate, suppose a researcher wants to assess the influences of a categorical predictor variable *A* with 3 levels on some outcome, and that measurements on a continuous predictor variable *P*, known to covary with the outcome, are available. If the data for the analysis are

$$\begin{array}{cc} P & \text{Group} \\ \begin{bmatrix} 7 \\ 4 \\ 9 \\ 3 \\ 6 \\ 8 \end{bmatrix} & \begin{bmatrix} A_1 \\ A_1 \\ A_2 \\ A_2 \\ A_3 \\ A_3 \end{bmatrix} \end{array}$$

then the sigma-restricted ***X*** matrix for the design that includes the separate first-order effects of *P* and *A* would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{cccc} X_0 & X_1 & X_2 & X_3 \end{array} \\ \begin{bmatrix} 1 & 7 & 1 & 0 \\ 1 & 4 & 1 & 0 \\ 1 & 9 & 0 & 1 \\ 1 & 3 & 0 & 1 \\ 1 & 6 & -1 & -1 \\ 1 & 8 & -1 & -1 \end{bmatrix} \end{array}$$

The b_2 and b_3 coefficients in the regression equation

$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3$

represent the influences of group membership on the *A* categorical predictor variable, controlling for the influence of scores on the *P* continuous predictor variable. Similarly, the b_1 coefficient represents the influence of scores on *P* controlling for the influences of group membership on *A*. This traditional ANCOVA analysis gives a more sensitive test of the influence of *A* to the extent that *P* reduces the prediction error, that is, the residuals for the outcome variable.

The ***X*** matrix for the same design using the overparameterized model would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{ccccc} X_0 & X_1 & X_2 & X_3 & X_4 \end{array} \\ \left[\begin{array}{ccccc} 1 & 7 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ 1 & 9 & 0 & 1 & 0 \\ 1 & 3 & 0 & 1 & 0 \\ 1 & 6 & 0 & 0 & 1 \\ 1 & 8 & 0 & 0 & 1 \end{array}\right] \end{array}$$

The interpretation is unchanged except that the influences of group membership on the *A* categorical predictor variables are represented by the b_2, b_3 and b_4 coefficients in the regression equation

$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4$

Separate Slope Designs. The traditional analysis of covariance (ANCOVA) design for categorical and continuous predictor variables is inappropriate when the categorical and continuous predictors interact in influencing responses on the outcome. The appropriate design for modeling the influences of the predictors in this situation is called the separate slope design. For the same example data used to illustrate traditional ANCOVA, the overparameterized ***X*** matrix for the design that includes the main effect of the three-level categorical predictor *A* and the 2-way interaction of *P* by *A* would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{ccccccc} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 \end{array} \\ \left[\begin{array}{ccccccc} 1 & 1 & 0 & 0 & 7 & 0 & 0 \\ 1 & 1 & 0 & 0 & 4 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 9 & 0 \\ 1 & 0 & 1 & 0 & 0 & 3 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 6 \\ 1 & 0 & 0 & 1 & 0 & 0 & 8 \end{array}\right] \end{array}$$

The b_4, b_5, and b_6 coefficients in the regression equation

$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + b_5X_5 + b_6X_6$

give the separate slopes for the regression of the outcome on *P* within each group on *A*, controlling for the main effect of *A*.

As with nested ANOVA designs, the sigma-restricted coding of effects for separate slope designs is overly restrictive, so only the overparameterized model is used to represent separate slope designs. In fact, separate slope designs are identical in form to nested ANOVA designs, since the main effects for continuous predictors are omitted in separate slope designs.

Homogeneity of Slopes. The appropriate design for modeling the influences of continuous and categorical predictor variables depends on whether the continuous and categorical predictors interact in influencing the outcome. The traditional analysis of covariance (ANCOVA) design for continuous and categorical predictor variables is appropriate when the continuous and categorical predictors do not interact in influencing responses on the outcome, and the separate slope design is appropriate when the continuous and categorical predictors do interact in influencing responses. The homogeneity of slopes designs can be used to test whether the continuous and categorical predictors interact in influencing responses, and thus, whether the traditional ANCOVA design or the separate slope design is appropriate for modeling the effects of the predictors. For the same example data used to illustrate the traditional ANCOVA and separate slope designs, the overparameterized ***X*** matrix for the design that includes the main effect of *P*, the main effect of the three-level categorical predictor *A*, and the 2-way interaction of *P* by *A* would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{cccccccc} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 \end{array} \\ \left[\begin{array}{cccccccc} 1 & 7 & 1 & 0 & 0 & 7 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 & 4 & 0 & 0 \\ 1 & 9 & 0 & 1 & 0 & 0 & 9 & 0 \\ 1 & 3 & 0 & 1 & 0 & 0 & 3 & 0 \\ 1 & 6 & 0 & 0 & 1 & 0 & 0 & 6 \\ 1 & 8 & 0 & 0 & 1 & 0 & 0 & 8 \end{array}\right] \end{array}$$

If the b_5, b_6, or b_7 coefficient in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + b_5X_5 + b_6X_6 + b_7X_7$$

is non-zero, the separate slope model should be used. If instead all 3 of these regression coefficients are zero the traditional ANCOVA design should be used.

The sigma-restricted ***X*** matrix for the homogeneity of slopes design would be

$$\mathbf{X} = \begin{array}{c} \begin{matrix} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 \end{matrix} \\ \begin{bmatrix} 1 & 7 & 1 & 0 & 7 & 0 \\ 1 & 4 & 1 & 0 & 4 & 0 \\ 1 & 9 & 0 & 1 & 0 & 9 \\ 1 & 3 & 0 & 1 & 0 & 3 \\ 1 & 6 & -1 & -1 & -6 & -6 \\ 1 & 8 & -1 & -1 & -8 & -8 \end{bmatrix} \end{array}$$

Using this ***X*** matrix, if the b_4, or b_5 coefficient in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + b_5X_5$$

is non-zero, the separate slope model should be used. If instead both of these regression coefficients are zero the traditional ANCOVA design should be used.

<>Model Building

In addition to fitting the whole model for the specified type of analysis, different methods for automatic model building can be employed in analyses using the generalized linear model. Specifically, forward entry, backward removal, forward stepwise, and backward stepwise procedures can be performed, as well as best-subset search procedures. In forward methods of selection of effects to include in the model (i.e., forward entry and forward stepwise methods), score statistics are compared to select new (significant) effects. The Wald statistic can be used for backward removal methods (i.e., backward removal and backward stepwise, when effects are selected for removal from the model).

The best subsets search method can be based on three different test statistics: the score statistic, the model likelihood, and the AIC (Akaike Information Criterion, see Akaike, 1973). Note that, since the score statistic does not require iterative computations, best subset selection based on the score statistic is computationally fastest, while selection based on the other two statistics usually provides more accurate results; see McCullagh and Nelder(1989), for additional details.

Interpretation of Results and Diagnostics

Simple estimation and test statistics may not be sufficient for adequate interpretation of the effects in an analysis. Especially for higher order (e.g., interaction) effects, inspection of the observed and predicted means can be invaluable for understanding the nature of an effect. Plots of these means (with error bars) can be useful for quickly grasping the role of the effects in the model.

Inspection of the distributions of variables is critically important when using the generalized linear model. Histograms and probability plots for variables, and scatterplots showing the relationships between observed values, predicted values, and residuals (e.g., Pearson residuals, deviance residuals, studentized residuals, differential *Chi-square* statistics, differential deviance statistics, and generalized Cook's D) provide invaluable model-checking tools.

Chapter 21
General Regression Models (GRM)

- Basic Ideas: The Need for Simple Models
- Model Building in GSR
- Types of Analyses
    - Between Subject Designs
    - Multivariate Designs
- Building the Whole Model
    - Partitioning Sums of Squares
    - Testing the Whole Model
    - Limitations of Whole Models
- Building Models via Stepwise Regression
- Building Models via Best-Subset Regression

This chapter describes the use of the general linear model for finding the "best" linear model from a number of possible models. If you are unfamiliar with the basic methods of ANOVA and regression in linear models, it may be useful to first review the basic information on these topics in *Elementary Concepts*. A detailed discussion of univariate and multivariate ANOVA techniques can also be found in the *ANOVA/MANOVA* chapter; a discussion of multiple regression methods is also provided in the *Multiple Regression* chapter. Discussion of the ways in which the linear regression model is extended by the general linear model can be found in the General Linear Models chapter.

Basic Ideas: The Need for Simple Models

A good theory is the end result of a winnowing process. We start with a comprehensive model that includes all conceivable, testable influences on the phenomena under investigation. Then we test the components of the initial comprehensive model, to identify the less comprehensive submodels that adequately account for the phenomena under investigation. Finally from these candidate submodels, we single out the simplest submodel, which by the principle of parsimony we take to be the "best" explanation for the phenomena under investigation.

We prefer simple models not just for philosophical but also for practical reasons. Simple models are easier to put to test again in replication and cross-validation studies. Simple models are less costly to put into practice in predicting and controlling the outcome in the future. The philosophical reasons for preferring simple models should not be downplayed, however. Simpler models are easier to understand and appreciate, and therefore have a "beauty" that their more complicated counterparts often lack.

The entire winnowing process described above is encapsulated in the model-building techniques of stepwise and best-subset regression. The use of these model-building techniques begins with the specification of the design for a comprehensive "whole model." Less comprehensive submodels are then tested to determine if they adequately account for the outcome under investigation. Finally, the simplest of the adequate is adopted as the "best."

Model Building in GSR

Unlike the multiple regression model, which is used to analyze designs with continuous predictor variables, the general linear model can be used to analyze any ANOVA design with categorical predictor variables, any ANCOVA design with both categorical and continuous predictor variables, as well as any regression design with continuous predictor variables. Effects for categorical predictor variables can be coded in the design matrix ***X*** using either the overparameterized model or the sigma-restricted model.

Only the sigma-restricted parameterization can be used for model-building. True to its description as general, the general linear model can be used to analyze designs with effects for categorical predictor variables which are coded using either parameterization method. In many uses of the general linear model, it is arbitrary whether categorical predictors are coded using the sigma-restricted or the overparameterized coding. When one desires to build models, however, the use of the overparameterized model is unsatisfactory; lower-order effects for categorical predictor variables are redundant with higher-order *containing* interactions, and therefore cannot be fairly evaluated for inclusion in the model when higher-order *containing* interactions are already in the model.

This problem does not occur when categorical predictors are coded using the sigma-restricted parameterization, so only the sigma-restricted parameterization is necessary in general stepwise regression.

Designs which cannot be represented using the sigma-restricted parameterization. The sigma-restricted parameterization can be used to represent most, but not all types of designs. Specifically, the designs which cannot be represented using the sigma-restricted parameterization are designs with nested effects, such as *nested ANOVA* and *separate slope*, and random effects. Any other type of ANOVA, ANCOVA, or regression design can be represented using the sigma-restricted parameterization, and can therefore be analyzed with general stepwise regression.

Model building for designs with multiple dependent variables. *Stepwise* and *best-subset* model-building techniques are well-developed for regression designs with a single dependent variable (e.g., see Cooley and Lohnes, 1971; Darlington, 1990; Hocking Lindeman, Merenda, and Gold, 1980; Morrison, 1967; Neter, Wasserman, and Kutner, 1985; Pedhazur, 1973; Stevens, 1986; Younger, 1985). Using the sigma-restricted parameterization and general linear model methods, these model-building techniques can be readily applied to any ANOVA design with categorical predictor variables, any ANCOVA design with both categorical and continuous predictor variables, as well as any regression design with continuous predictor variables. Building models for designs with multiple dependent variables, however, involves considerations that are not typically addressed by the general linear model. Model-building techniques for designs with multiple dependent variables are available with *Structural Equation Modeling*.

Types of Analyses

A wide variety of types of designs can be represented using the sigma-restricted coding of the design matrix ***X***, and any such design can be analyzed using the general linear model. The following topics describe these different types of designs and how they differ. Some general ways in which designs might differ can be suggested, but keep in mind that any particular design can be a "hybrid" in the sense that it could have combinations of features of a number of different types of designs.

Between-subject designs

- Overview
- Simple regression
- Multiple regression
- Factorial regression
- Polynomial regression
- Response surface regression
- Mixture surface regression
- One-way ANOVA
- Main effect ANOVA
- Factorial ANOVA
- Analysis of covariance (ANCOVA)
- Homogeneity of slopes

Overview. The levels or values of the predictor variables in an analysis describe the differences between the *n* subjects or the *n* valid cases that are analyzed. Thus, when we speak of the between subject design (or simply the between design) for an analysis, we are referring to the nature, number, and arrangement of the predictor variables.

Concerning the nature or type of predictor variables, between designs which contain only categorical predictor variables can be called ANOVA (analysis of variance) designs, between designs which contain only continuous predictor variables can be called regression designs, and between designs which contain both categorical and continuous predictor variables can be called ANCOVA (analysis of covariance) designs.

Between designs may involve only a single predictor variable and therefore be described as simple (e.g., simple regression) or may employ numerous predictor variables (e.g., multiple regression).

Concerning the arrangement of predictor variables, some between designs employ only "main effect" or first-order terms for predictors, that is, the values for different predictor variables are independent and raised only to the first power. Other between designs may employ higher-order terms for predictors by raising the values for the original predictor variables to a power greater than 1 (e.g., in polynomial regression designs), or by forming products of different predictor variables (i.e., interaction terms). A common arrangement for ANOVA designs is the full-factorial design, in which every combination of levels for each of the categorical predictor variables is represented in the design. Designs with some but not all combinations of levels for each of the categorical predictor variables are aptly called fractional factorial designs.

These basic distinctions about the nature, number, and arrangement of predictor variables can be used in describing a variety of different types of between designs. Some of the more common between designs can now be described.

Simple Regression. Simple regression designs involve a single continuous predictor variable. If there were 3 cases with values on a predictor variable *P* of, say, 7, 4, and 9, and the design is for the first-order effect of *P*, the ***X*** matrix would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{cc} X_0 & X_1 \end{array} \\ \begin{bmatrix} 1 & 7 \\ 1 & 4 \\ 1 & 9 \end{bmatrix} \end{array}$$

and using *P* for X_1 the regression equation would be

$$Y = b_0 + b_1P$$

If the simple regression design is for a higher-order effect of *P*, say the quadratic effect, the values in the X_1 column of the design matrix would be raised to the 2nd power, that is, squared

$$\mathbf{X} = \begin{array}{c} \begin{array}{cc} X_0 & X_1 \end{array} \\ \begin{bmatrix} 1 & 49 \\ 1 & 16 \\ 1 & 81 \end{bmatrix} \end{array}$$

and using P^2 for X_1 the regression equation would be

$Y = b_0 + b_1P^2$

In regression designs, values on the continuous predictor variables are raised to the desired power and used as the values for the *X* variables. No recoding is performed. It is therefore sufficient, in describing regression designs, to simply describe the regression equation without explicitly describing the design matrix ***X***.

Multiple Regression. Multiple regression designs are to continuous predictor variables as main effect ANOVA designs are to categorical predictor variables, that is, multiple regression designs contain the separate simple regression designs for 2 or more continuous predictor variables. The regression equation for a multiple regression design for the first-order effects of 3 continuous predictor variables *P*, *Q*, and *R* would be

$Y = b_0 + b_1P + b_2Q + b_3R$

A discussion of multiple regression methods is also provided in the *Multiple Regression* chapter.

Factorial Regression. Factorial regression designs are similar to factorial ANOVA designs, in which combinations of the levels of the factors are represented in the design. In factorial regression designs, however, there may be many more such possible combinations of distinct levels for the continuous predictor variables than there are cases in the data set. To simplify matters, full-factorial regression designs are defined as designs in which all possible products of the continuous predictor variables are represented in the design. For example, the full-factorial regression design for two continuous predictor variables *P* and *Q* would include the main effects (i.e., the first-order effects) of *P* and *Q* and their 2-way *P* by *Q* interaction effect, which is represented by the product of *P* and *Q* scores for each case. The regression equation would be

$Y = b_0 + b_1P + b_2Q + b_3P*Q$

Factorial regression designs can also be fractional, that is, higher-order effects can be omitted from the design. A fractional factorial design to degree 2 for 3 continuous predictor variables *P*, *Q*, and *R* would include the main effects and all 2-way interactions between the predictor variables

$Y = b_0 + b_1P + b_2Q + b_3R + b_4P*Q + b_5P*R + b_6Q*R$

Polynomial Regression. Polynomial regression designs are designs which contain main effects and higher-order effects for the continuous predictor variables but do not include interaction effects between predictor variables. For example, the polynomial regression design to degree 2 for three continuous predictor variables *P*, *Q*, and *R* would include the main effects (i.e., the first-order effects) of *P*, *Q*, and *R* and their quadratic (i.e., second-order) effects, but not the 2-way interaction effects or the *P* by *Q* by *R* 3-way interaction effect.

$$Y = b_0 + b_1P + b_2P^2 + b_3Q + b_4Q^2 + b_5R + b_6R^2$$

Polynomial regression designs do not have to contain all effects up to the same degree for every predictor variable. For example, main, quadratic, and cubic effects could be included in the design for some predictor variables, and effects up the fourth degree could be included in the design for other predictor variables.

Response Surface Regression. Quadratic response surface regression designs are a hybrid type of design with characteristics of both polynomial regression designs and fractional factorial regression designs. Quadratic response surface regression designs contain all the same effects of polynomial regression designs to degree 2 and additionally the 2-way interaction effects of the predictor variables. The regression equation for a quadratic response surface regression design for 3 continuous predictor variables *P, Q,* and *R* would be

$$Y = b_0 + b_1P + b_2P^2 + b_3Q + b_4Q^2 + b_5R + b_6R^2 + b_7P*Q + b_8P*R + b_9Q*R$$

These types of designs are commonly employed in applied research (e.g., in industrial experimentation), and a detailed discussion of these types of designs is also presented in the *Experimental Design* chapter (see *Central composite designs*).

Mixture Surface Regression. Mixture surface regression designs are identical to factorial regression designs to degree 2 except for the omission of the intercept. Mixtures, as the name implies, add up to a constant value; the sum of the proportions of ingredients in different recipes for some material all must add up 100%. Thus, the proportion of one ingredient in a material is redundant with the remaining ingredients. Mixture surface regression designs deal with this redundancy by omitting the intercept from the design. The design matrix for a mixture surface regression design for 3 continuous predictor variables *P*, *Q,* and *R* would be

$$Y = b_1P + b_2P^2 + b_3Q + b_4P*Q + b_5P*R + b_6Q*R$$

These types of designs are commonly employed in applied research (e.g., in industrial experimentation), and a detailed discussion of these types of designs is also presented in the *Experimental Design* chapter (see *Mixture designs and triangular surfaces*).

One-Way ANOVA. A design with a single categorical predictor variable is called a one-way ANOVA design. For example, a study of 4 different fertilizers used on different individual plants could be analyzed via one-way ANOVA, with four levels for the factor *Fertilizer.*

Consider a single categorical predictor variable *A* with 1 case in each of its 3 categories. Using the sigma-restricted coding of *A* into 2 quantitative contrast variables, the matrix ***X*** defining the between design is

$$\mathbf{X} = \begin{array}{c|ccc|} & X_0 & X_1 & X_2 \\ A_1 & 1 & 1 & 0 \\ A_2 & 1 & 0 & 1 \\ A_3 & 1 & -1 & -1 \end{array}$$

That is, cases in groups A_1, A_2, and A_3 are all assigned values of 1 on X_0 (the intercept), the case in group A_1 is assigned a value of 1 on X_1 and a value 0 on X_2, the case in group A_2 is assigned a value of 0 on X_1 and a value 1 on X_2, and the case in group A_3 is assigned a value of -1 on X_1 and a value -1 on X_2. Of course, any additional cases in any of the 3 groups would be coded similarly. If there were 1 case in group A_1, 2 cases in group A_2, and 1 case in group A_3, the ***X*** matrix would be

$$\mathbf{X} = \begin{array}{c|ccc|} & X_0 & X_1 & X_2 \\ A_{11} & 1 & 1 & 0 \\ A_{12} & 1 & 0 & 1 \\ A_{22} & 1 & 0 & 1 \\ A_{13} & 1 & -1 & -1 \end{array}$$

where the first subscript for *A* gives the replicate number for the cases in each group. For brevity, replicates usually are not shown when describing ANOVA design matrices.

Note that in one-way designs with an equal number of cases in each group, sigma-restricted coding yields $X_1 \ldots X_k$ variables all of which have means of 0.

These simple examples show that the ***X*** matrix actually serves two purposes. It specifies (1) the coding for the levels of the original predictor variables on the *X* variables used in the analysis as well as (2) the nature, number, and arrangement of the X variables, that is, the between design.

Main Effect ANOVA. Main effect ANOVA designs contain separate one-way ANOVA designs for 2 or more categorical predictors. A good example of main effect ANOVA would be the typical analysis performed on *screening designs* as described in the context of the *Experimental Design* chapter.

Consider 2 categorical predictor variables *A* and *B* each with 2 categories. Using the sigma-restricted coding, the ***X*** matrix defining the between design is

$$\mathbf{X} = \begin{array}{c|ccc|} & X_0 & X_1 & X_2 \\ A_1B_1 & 1 & 1 & 1 \\ A_1B_2 & 1 & 1 & -1 \\ A_2B_1 & 1 & -1 & 1 \\ A_2B_2 & 1 & -1 & -1 \end{array}$$

Note that if there are equal numbers of cases in each group, the sum of the cross-products of values for the X_1 and X_2 columns is 0, for example, with 1 case in each group (1*1)+(1*-1)+(-1*1)+(-1*-1)=0.

Factorial ANOVA. Factorial ANOVA designs contain *X* variables representing combinations of the levels of 2 or more categorical predictors (e.g., a study of boys and girls in four age groups, resulting in a *2 (Gender) x 4 (Age Group)* design). In particular, full-factorial designs represent all possible combinations of the levels of the categorical predictors. A full-factorial design with 2 categorical predictor variables *A* and *B* each with 2 levels would be called a 2 x 2 full-factorial design. Using the sigma-restricted coding, the ***X*** matrix for this design would be

$$\mathbf{X} = \begin{array}{c} \\ A_1B_1 \\ A_1B_2 \\ A_2B_1 \\ A_2B_2 \end{array} \begin{array}{cccc} X_0 & X_1 & X_2 & X_3 \\ \hline 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{array}$$

Several features of this ***X*** matrix deserve comment. Note that the X_1 and X_2 columns represent main effect contrasts for one variable, (i.e., *A* and *B*, respectively) collapsing across the levels of the other variable. The X_3 column instead represents a contrast between different combinations of the levels of *A* and *B*. Note also that the values for X_3 are products of the corresponding values for X_1 and X_2. Product variables such as X_3 represent the multiplicative or interaction effects of their factors, so X_3 would be said to represent the 2-way interaction of *A* and *B*. The relationship of such product variables to the dependent variables indicate the interactive influences of the factors on responses above and beyond their independent (i.e., main effect) influences on responses. Thus, factorial designs provide more information about the relationships between categorical predictor variables and responses on the dependent variables than is provided by corresponding one-way or main effect designs.

When many factors are being investigated, however, full-factorial designs sometimes require more data than reasonably can be collected to represent all possible combinations of levels of the factors, and high-order interactions between many factors can become difficult to interpret. With many factors, a useful alternative to the full-factorial design is the fractional factorial design. As an example, consider a 2 x 2 x 2 fractional factorial design to degree 2 with 3 categorical predictor variables each with 2 levels. The design would include the main effects for each variable, and all 2-way interactions between the three variables, but would not include the 3-way interactions between all three variables. These types of designs are discussed in detail in the *2**(k-p) Fractional Factorial Designs* section of the *Experimental Design* chapter.

Analysis of Covariance. In general, between designs which contain both categorical and continuous predictor variables can be called ANCOVA designs. Traditionally, however, ANCOVA designs have referred more specifically to designs in which the first-order effects of one or more continuous predictor variables are taken into account when assessing the effects of one or more categorical predictor variables. A basic introduction to analysis of covariance can also be found in the *Analysis of covariance (ANCOVA)* topic of the *ANOVA/MANOVA* chapter.

To illustrate, suppose a researcher wants to assess the influences of a categorical predictor variable *A* with 3 levels on some outcome, and that measurements on a continuous predictor variable *P*, known to covary with the outcome, are available. If the data for the analysis are

$$\begin{array}{cc} P & \text{Group} \\ \begin{bmatrix} 7 \\ 4 \\ 9 \\ 3 \\ 6 \\ 8 \end{bmatrix} & \begin{bmatrix} A_1 \\ A_1 \\ A_2 \\ A_2 \\ A_3 \\ A_3 \end{bmatrix} \end{array}$$

then the sigma-restricted ***X*** matrix for the design that includes the separate first-order effects of *P* and *A* would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{cccc} X_0 & X_1 & X_2 & X_3 \end{array} \\ \begin{bmatrix} 1 & 7 & 1 & 0 \\ 1 & 4 & 1 & 0 \\ 1 & 9 & 0 & 1 \\ 1 & 3 & 0 & 1 \\ 1 & 6 & -1 & -1 \\ 1 & 8 & -1 & -1 \end{bmatrix} \end{array}$$

The b_2 and b_3 coefficients in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3$$

represent the influences of group membership on the *A* categorical predictor variable, controlling for the influence of scores on the *P* continuous predictor variable. Similarly, the b_1 coefficient represents the influence of scores on *P* controlling for the influences of group membership on *A*. This traditional ANCOVA analysis gives a more sensitive test of the influence of *A* to the extent that *P* reduces the prediction error, that is, the residuals for the outcome variable.

Homogeneity of Slopes. The appropriate design for modeling the influences of continuous and categorical predictor variables depends on whether the continuous and categorical predictors interact in influencing the outcome. The traditional analysis of covariance (ANCOVA) design for continuous and categorical predictor variables is appropriate when the continuous and categorical predictors do not interact in influencing responses on the outcome. The homogeneity of slopes designs can be used to test whether the continuous and categorical predictors interact in influencing responses. For the same example data used to illustrate the traditional ANCOVA design, the sigma-restricted ***X*** matrix for the homogeneity of slopes design would be

$$\mathbf{X} = \begin{array}{c} \\ \end{array}\begin{array}{cccccc} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 \\ \end{array}$$

$$\mathbf{X} \;=\; \left[\begin{array}{cccccc} 1 & 7 & 1 & 0 & 7 & 0 \\ 1 & 4 & 1 & 0 & 4 & 0 \\ 1 & 9 & 0 & 1 & 0 & 9 \\ 1 & 3 & 0 & 1 & 0 & 3 \\ 1 & 6 & -1 & -1 & -6 & -6 \\ 1 & 8 & -1 & -1 & -8 & -8 \end{array}\right]$$

Using this design matrix ***X***, if the b_4 and b_5 coefficients in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + b_5X_5$$

are zero, the simpler traditional ANCOVA design should be used.

Multivariate Designs Overview

When there are multiple dependent variables in a design, the design is said to be multivariate. Multivariate measures of association are by nature more complex than their univariate counterparts (such as the correlation coefficient, for example). This is because multivariate measures of association must take into account not only the relationships of the predictor variables with responses on the dependent variables, but also the relationships among the multiple dependent variables. By doing so, however, these measures of association provide information about the strength of the relationships between predictor and dependent variables independent of the dependent variables interrelationships. A basic discussion of multivariate designs is also presented in the *Multivariate Designs* topic in the *ANOVA/MANOVA* chapter.

The most commonly used multivariate measures of association all can be expressed as functions of the eigenvalues of the product matrix

$$\mathbf{E}^{-1}\mathbf{H}$$

where ***E*** is the error SSCP matrix (i.e., the matrix of sums of squares and cross-products for the dependent variables that are not accounted for by the predictors in the between design), and ***H*** is a hypothesis SSCP matrix (i.e., the matrix of sums of squares and cross-products for the dependent variables that are accounted for by all the predictors in the between design, or the sums of squares and cross-products for the dependent variables that are accounted for by a particular effect). If

$\square_i$ = the ordered eigenvalues of $\mathbf{E}^{-1}\mathbf{H}$, if $\mathbf{E}^{-1}$ exists then the 4 commonly used multivariate measures of association are

Wilks' lambda = $\square[1/(1+\square_i)]$

Pillai's trace = $\square\,\square_i/(1+\square_i)$

Hotelling-Lawley trace = $\square\,\square_i$

Roy's largest root = $\square_1$

These 4 measures have different upper and lower bounds, with Wilks' lambda perhaps being the most easily interpretable of the four measures. Wilks' lambda can range from 0 to 1, with 1 indicating no relationship of predictors to responses and 0 indicating a perfect relationship of predictors to responses. 1 - Wilks' lambda can be interpreted as the multivariate counterpart of a univariate R-squared, that is, it indicates the proportion of generalized variance in the dependent variables that is accounted for by the predictors.

The 4 measures of association are also used to construct multivariate tests of significance. These multivariate tests are covered in detail in a number of sources (e.g., Finn, 1974; Tatsuoka, 1971).

Building the Whole Model

The following sections discuss details for building and testing hypotheses about the "whole model", for example, how sums of squares are partitioned and how the overall fit for the whole model is tested.

Partitioning Sums of Squares

A fundamental principle of least squares methods is that variation on a dependent variable can be partitioned, or divided into parts, according to the sources of the variation. Suppose that a dependent variable is regressed on one or more predictor variables, and that for convenience the dependent variable is scaled so that its mean is 0. Then a basic least squares identity is that the total sum of squared values on the dependent variable equals the sum of squared predicted values plus the sum of squared residual values. Stated more generally,

$\square(\text{y - y-bar})^2 = \square(\text{y-hat - y-bar})^2 + \square(\text{y - y-hat})^2$

where the term on the left is the total sum of squared deviations of the observed values on the dependent variable from the dependent variable mean, and the respective terms on the right are (1) the sum of squared deviations of the predicted values for the dependent variable from the dependent variable mean and (2) the sum of the squared deviations of the observed values on the dependent variable from the predicted values, that is, the sum of the squared residuals. Stated yet another way,

Total SS = Model SS + Error SS

Note that the Total SS is always the same for any particular data set, but that the Model SS and the Error SS depend on the regression equation. Assuming again that the dependent variable is scaled so that its mean is 0, the Model SS and the Error SS can be computed using

Model SS = **b'X'Y**

Error SS = **Y'Y - b'X'Y**

Testing the Whole Model

Given the Model SS and the Error SS, one can perform a test that all the regression coefficients for the *X* variables (b_1 through b_k, excluding the b_0 coefficient for the intercept) are zero. This test is equivalent to a comparison of the fit of the regression surface defined by the predicted values (computed from the whole model regression equation) to the fit of the regression surface defined solely by the dependent variable mean (computed from the reduced regression equation containing only the intercept). Assuming that ***X'X*** is full-rank, the whole model hypothesis mean square

MSH = (Model SS)/k

where *k* is the number of columns of ***X*** (excluding the intercept column), is an estimate of the variance of the predicted values. The error mean square

s^2 = MSE = (Error SS)/(n-k-1)

where *n* is the number of observations, is an unbiased estimate of the residual or error variance. The test statistic is

F = MSH/MSE

where *F* has (*k*, *n* - *k* - 1) degrees of freedom.

If ***X'X*** is not full rank, *r* + 1 is substituted for *k*, where *r* is the rank or the number of non-redundant columns of ***X'X***.

If the whole model test is not significant the analysis is complete; the whole model is concluded to fit the data no better than the reduced model using the dependent variable mean alone. It is futile to seek a submodel which adequately fits the data when the whole model is inadequate.

Note that in the case of non-intercept models, some multiple regression programs will only compute the full model test based on the proportion of variance around 0 (zero) accounted for by the predictors; for more information (see Kvålseth, 1985; Okunade, Chang, and Evans, 1993). Other programs will actually compute both values (i.e., based on the residual variance around 0, and around the respective dependent variable means.

Limitations of Whole Models

For designs such as one-way ANOVA or simple regression designs, the whole model test by itself may be sufficient for testing general hypotheses about whether or not the single predictor variable is related to the outcome. In complex designs, however, finding a statistically significant test of whole model fit is often just the first step in the analysis; one then seeks to identify simpler submodels that fit the data equally well (see the section on *Basic ideas: The need for simple models*). It is to this task, the search for submodels that fit the data well, that stepwise and best-subset regression are devoted.

Building Models via Stepwise Regression

Stepwise model-building techniques for regression designs with a single dependent variable are described in numerous sources (e.g., see Darlington, 1990; Hocking, 1966, Lindeman, Merenda, and Gold, 1980; Morrison, 1967; Neter, Wasserman, and Kutner, 1985; Pedhazur, 1973; Stevens, 1986; Younger, 1985). The basic procedures involve (1) identifying an initial model, (2) iteratively "stepping," that is, repeatedly altering the model at the previous step by adding or removing a predictor variable in accordance with the "stepping criteria," and (3) terminating the search when stepping is no longer possible given the stepping criteria, or when a specified maximum number of steps has been reached. The following topics provide details on the use of stepwise model-building procedures.

The Initial Model in Stepwise Regression. The initial model is designated the model at *Step 0*. The initial model always includes the regression intercept (unless the *No intercept* option has been specified.). For the *backward stepwise* and *backward removal* methods, the initial model also includes all effects specified to be included in the *design* for the analysis. The initial model for these methods is therefore the whole model.

For the *forward stepwise* and *forward entry* methods, the initial model always includes the regression intercept (unless the *No intercept* option has been specified.). The initial model may also include 1 or more effects specified to be *forced* into the model. If *j* is the number of effects specified to be *forced* into the model, the first *j* effects specified to be included in the *design* are entered into the model at *Step 0* . Any such effects are *not eligible to be removed* from the model during subsequent *Steps.*

Effects may also be specified to be *forced* into the model when the *backward stepwise* and *backward removal* methods are used. As in the *forward stepwise* and *forward entry* methods, any such effects are *not eligible to be removed* from the model during subsequent *Steps*.

The Forward Entry Method. The *forward entry* method is a simple model-building procedure. At each *Step* after *Step 0,* the *entry statistic* is computed for each effect eligible for entry in the model. If no effect has a value on the *entry statistic* which exceeds the specified critical value for model entry, then stepping is terminated, otherwise the effect with the largest value on the *entry statistic* is entered into the model. Stepping is also terminated if the maximum number of steps is reached.

The Backward Removal Method. The *backward removal* method is also a simple model-building procedure. At each *Step* after *Step 0*, the *removal statistic* is computed for each effect eligible to be removed from the model. If no effect has a value on the *removal statistic* which is less than the critical value for removal from the model, then stepping is terminated, otherwise the effect with the smallest value on the *removal statistic* is removed from the model. Stepping is also terminated if the maximum number of steps is reached.

The Forward Stepwise Method. The *forward stepwise* method employs a combination of the procedures used in the *forward entry* and *backward removal* methods. At *Step 1* the procedures for *forward entry* are performed. At any subsequent step where 2 or more effects have been selected for entry into the model, *forward entry* is performed if possible, and *backward removal* is performed if possible, until neither procedure can be performed and stepping is terminated. Stepping is also terminated if the maximum number of steps is reached.

The Backward Stepwise Method. The *backward stepwise* method employs a combination of the procedures used in the *forward entry* and *backward removal* methods. At *Step 1* the procedures for *backward removal* are performed. At any subsequent step where 2 or more effects have been selected for entry into the model, *forward entry* is performed if possible, and *backward removal* is performed if possible, until neither procedure can be performed and stepping is terminated. Stepping is also terminated if the maximum number of steps is reached.

Entry and Removal Criteria. Either critical *F* values or critical *p* values can be specified to be used to control entry and removal of effects from the model. If *p* values are specified, the actual values used to control entry and removal of effects from the model are 1 minus the specified *p* values. The critical value for model entry must exceed the critical value for removal from the model. A maximum number of *Steps* can also be specified. If not previously terminated, stepping stops when the specified maximum number of *Steps* is reached.

Building Models via Best-Subset Regression

All-possible-subset regression can be used as an alternative to or in conjunction with *stepwise* methods for finding the "best" possible submodel.

Neter, Wasserman, and Kutner (1985) discuss the use of all-possible-subset regression in conjunction with *stepwise regression* "A limitation of the stepwise regression search approach is that it presumes there is a single "best" subset of *X* variables and seeks to identify it. As noted earlier, there is often no unique "best" subset. Hence, some statisticians suggest that all possible regression models with a similar number of *X* variables as in the stepwise regression solution be fitted subsequently to study whether some other subsets of *X* variables might be better." (p. 435). This reasoning suggests that after finding a stepwise solution, the "best" of all the possible subsets of the same number of effects should be examined to determine if the stepwise solution is among the "best." If not, the stepwise solution is suspect.

All-possible-subset regression can also be used as an alternative to *stepwise regression*. Using this approach, one first decides on the range of subset sizes that could be considered to be useful. For example, one might expect that inclusion of at least 3 effects in the model is necessary to adequately account for responses, and also might expect there is no advantage to considering models with more than 6 effects. Only the "best" of all possible subsets of 3, 4, 5, and 6 effects are then considered.

Note that several different criteria can be used for ordering subsets in terms of "goodness." The most often used criteria are the subset multiple *R-square*, *adjusted R-square*, and *Mallow's Cp* statistics. When all-possible-subset regression is used in conjunction with *stepwise* methods, the subset multiple *R-square* statistic allows direct comparisons of the "best" subsets identified using each approach.

The number of possible submodels increases very rapidly as the number of effects in the whole model increases, and as subset size approaches half of the number of effects in the whole model. The amount of computation required to perform all-possible-subset regression increases as the number of possible submodels increases, and holding all else constant, also increases very rapidly as the number of levels for effects involving categorical predictors increases, thus resulting in more columns in the design matrix ***X***. For example, all possible subsets of up to a dozen or so effects could certainly theoretically be computed for a design that includes two dozen or so effects all of which have many levels, but the computation would be very time consuming (e.g., there are about 2.7 million different ways to select 12 predictors from 24 predictors, i.e., 2.7

million models to evaluate just for subset size 12). Simpler is generally better when using all-possible-subset regression.

Chapter 22
Selected Topics in Graphical Analytic Techniques

Brief Overviews of Types of Graphs

2D Graphs

- Bar/Column
- Bar Dev
- Bar Left Y
- Bar Right Y
- Bar Top
- Bar X
- Box
- Detrended Probability
- Half-Normal Probability
- Hanging Bar Histograms
- Histograms
- Line
- Pie Charts
- Probability
- Probability-Probability
- Quantile-Quantile
- Range
- Scatterplots
- Sequential/Stacked
- Voronoi Scatterplot

3D XYZ Graphs

- Contour
- Deviation
- Scatterplots
- Space
- Spectral
- Trace

3D Sequential Graphs

- Bivariate Histograms
- Box
- Range
- Raw Data Contour/Discrete
- Sequential Contour
- Sequential Surface
- Raw Data Spikes
- Raw Data Surface

4D/Ternary Graphs

- Scatterplots
- 3D Ternary
- Contour/Area
- Contour/Line
- 3D Deviation
- 3D Space

2D Categorized Graphs

- Detrended Probability
- Half-Normal Probability
- Normal Probability
- Probability-Probability
- Quantile-Quantile

3D Categorized Graphs

- Contour
- Deviation
- Scatterplots
- Space
- Spectral
- Surface

Ternary Categorized Graphs

- Ternary Contour/Area
- Ternary Contour/Line
- Ternary Scatterplot

nD/Icon Graphs

- Chernoff Faces
- Columns
- Lines
- Pies
- Polygons
- Profiles
- Stars
- Sun Rays

Matrix Graphs

- Columns
- Lines
- Scatterplot

Representative Visualization Techniques

- Categorized Graphs
  - What are Categorized Graphs?
  - Categorization Methods

    - Histograms
    - Scatterplots
    - Probability Plots
    - Quantile-Quantile Plots
    - Probability-Probability Plots
    - Line Plots
    - Box Plots
    - Pie Charts
    - Missing/Range Data Points Plots
    - 3D Plots
    - Ternary Plots
- Brushing
- Smoothing Bivariate Distributions
- Layered Compression
- Projections of 3D data sets
- Icon Plots
    - Analyzing Icon Plots
    - Taxonomy of Icon Plots
    - Standardization of Values
    - Applications
    - Related Graphs
    - Graph Type
    - Mark Icons
- Data Reduction
- Data Rotation (in 3D space)

Categorized Graphs

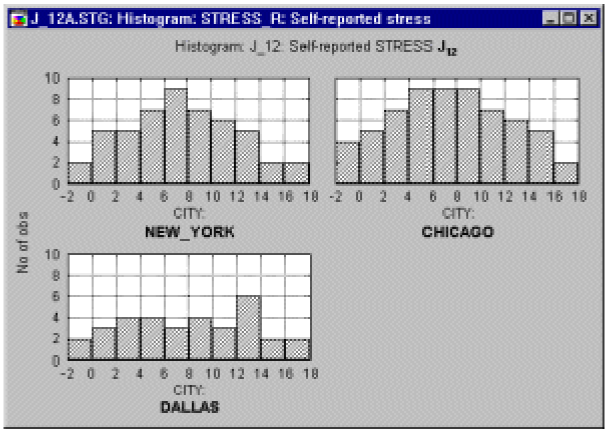
One of the most important, general, and also powerful analytic methods involves dividing ("splitting") the data set into categories in order compare the patterns of data between the resulting subsets. This common technique is known under a variety of terms (such as *breaking down*, *grouping*, *categorizing*, *splitting*, *slicing*, *drilling-down*, or *conditioning*) and it is used both in exploratory data analyses and hypothesis testing. For example: A positive relation between the age and the risk of a heart attack may be different in males and females (it may be stronger in males). A promising relation between taking a drug and a decrease of the cholesterol level may be present only in women with a low blood pressure and only in their thirties and forties. The process capability indices or capability histograms can be different for periods of time supervised by different operators. The regression slopes can be different in different experimental groups.

There are many computational techniques that capitalize on *grouping* and that are designed to quantify the differences that the grouping will reveal (e.g., ANOVA/MANOVA). However, graphical techniques (such as *categorized graphs* discussed in this section) offer unique advantages that cannot be substituted by any computational method alone: they can reveal patterns that cannot be easily quantified (e.g., complex interactions, exceptions, anomalies) and they provide unique, multidimensional, global analytic perspectives to explore or "mine" the data.

What are Categorized Graphs?

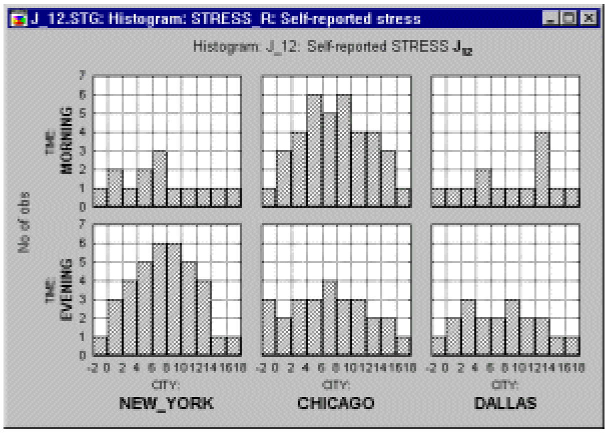
Categorized graphs (the term first used in *STATISTICA* software by StatSoft in 1990; also recently called *Trellis graphs*, by Becker, Cleveland, and Clark, at Bell Labs) produce a series of 2D, 3D, ternary, or nD graphs (such as histograms, scatterplots, line plots, surface plots, ternary scatterplots, etc.), one for each selected *category* of cases (i.e., subset of cases), for example, respondents from New York, Chicago, Dallas, etc. These "component" graphs are placed sequentially in one display, allowing for comparisons between the patterns of data shown in graphs for each of the requested groups (e.g., cities).

A variety of methods can be used to select the subsets; the simplest of them is using a categorical variable (e.g., a variable *City*, with three values *New York*, *Chicago*, and *Dallas*). For example, the following graph shows histograms of a variable representing self-reported stress levels in each of the three cities.


J_12A.STG: Histogram: STRESS_R: Self-reported stress
Histogram: J_12: Self-reported STRESS J_{12}
No of obs
10
8
6
4
2
0
-2 0 2 4 6 8 10 12 14 16 18
CITY:
NEW_YORK
CITY:
CHICAGO
CITY:
DALLAS

One could conclude that the data suggest that people who live in Dallas are less likely to report being stressed, while the patterns (distributions) of stress reporting in New York and Chicago are quite similar.

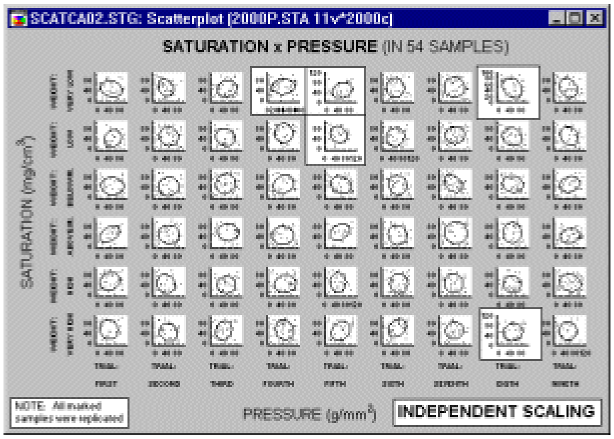
Categorized graphs in some software systems (e.g., in *STATISTICA*) also support two-way or multi-way categorizations, where not one criterion (e.g., *City*) but two or more criteria (e.g., *City* and *Time* of the day) are used to create the subsets. Two-way categorized graphs can be thought of as "crosstabulations of graphs" where each component graph represents a cross-section of one level of one grouping variable (e.g., *City*) and one level of the other grouping variable (e.g., *Time*).



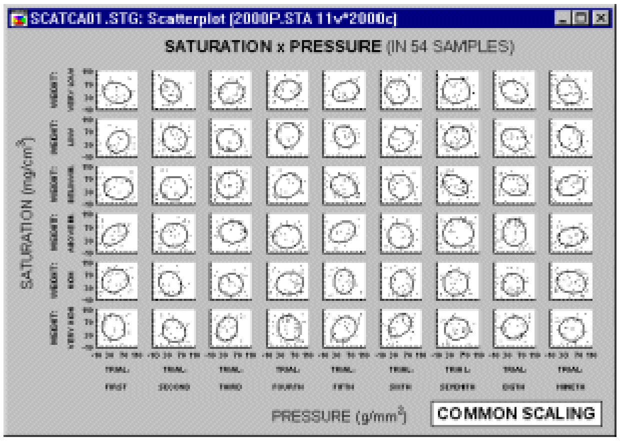
Adding this second factor reveals that the patterns of stress reporting in New York and Chicago are actually quite different when the *Time* of questioning is taken into consideration, whereas the *Time* factor makes little difference in Dallas.

Categorized graphs vs. matrix graphs. Matrix graphs also produce displays containing multiple component graphs; however, each of those component graphs are (or can be) based on the same set of cases and the graphs are generated for all combinations of variables from one or two lists. Categorized graphs require a selection of variables that normally would be selected for non-categorized graphs of the respective type (e.g., two variables for a scatterplot). However, in categorized plots, you also need to specify at least one *grouping variable* (or some criteria to be used for sorting the observations into the categories) that contains information on group membership of each case (e.g., *Chicago*, *Dallas*). That grouping variable will not be included in the graph directly (i.e., it will not be plotted) but it will serve as a criterion for dividing all analyzed cases into separate graphs. As illustrated above, one graph will be created for each group (category) identified by the grouping variable.

Common vs. Independent scaling. Each individual category graph may be scaled according to its own range of values (*independent* scaling),



or all graphs may be scaled to a *common* scale wide enough to accommodate all values in all of the category graphs.

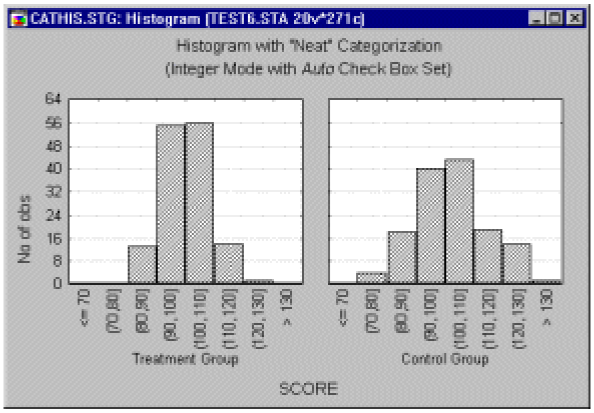


Common scaling allows the analyst to make comparisons of ranges and distributions of values among categories. However, if the ranges of values in graph categories are considerably different (causing a very wide common scale), then some of the graphs may be difficult to examine. The use of *independent* scaling may make it easier to spot trends and specific patterns within categories, but it may be more difficult to make comparisons of ranges of values among categories.

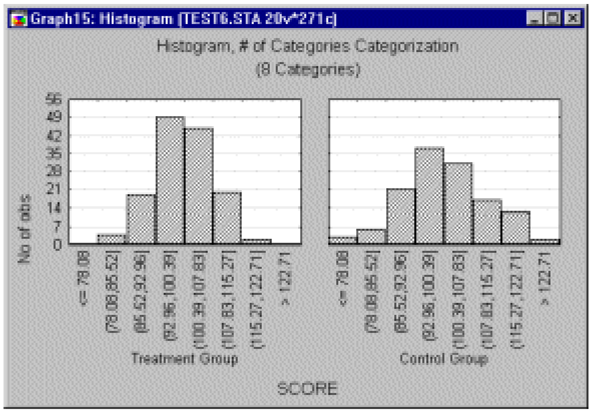
Categorization Methods

There are five general methods of categorization of values and they will be reviewed briefly in this section: Integer mode, Categories, Boundaries, Codes, and Multiple subsets. Note that the same methods of categorization can be used to categorize cases into component graphs and to categorize cases within component graphs (e.g., in histograms or box plots).

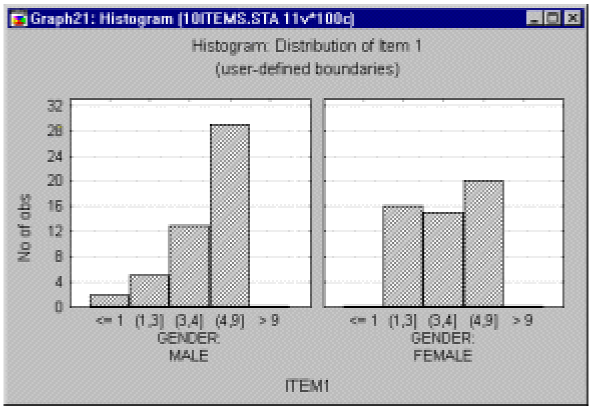
Integer Mode. When you use Integer Mode, integer values of the selected grouping variable will be used to define the categories, and one graph will be created for all cases that belong each category (defined by those integer values). If the selected grouping variable contains non-integer values, the software will usually truncate each encountered value of the selected grouping variable to an integer value.



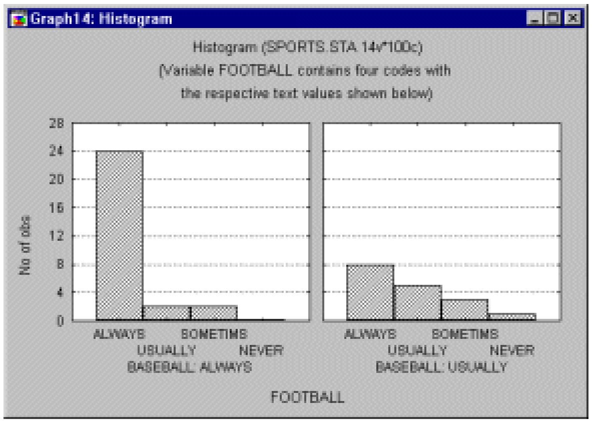
Categories. With this mode of categorization, you will specify the number of categories which you wish to use. The software will divide the entire range of values of the selected grouping variable (from minimum to maximum) into the requested number of equal length intervals.



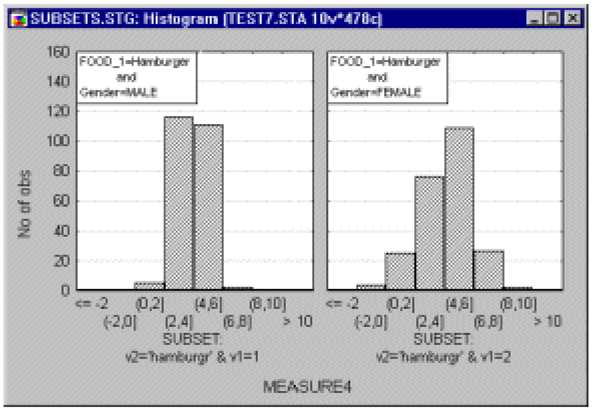
Boundaries. The Boundaries method will also create interval categorization, however, the intervals can be of arbitrary (e.g., uneven) width as defined by custom interval boundaries (for example, "less than –10," "greater than or equal to –10 but less than 0," "greater than or equal to 0 but less than 10," and "equal to or greater than 10").



Codes. Use this method if the selected grouping variable contains "codes" (i.e., specific, meaningful values such as *Male*, *Female*) from which you want to specify the categories.



Multiple subsets. This method allows you to custom-define the categories and enables you to use more than one variable to define the category. In other words, categorizations based on multiple subset definitions of categories may not represent distributions of specific (individual) variables but distributions of frequencies of specific "events" defined by particular combinations of values of several variables (and defined by conditions which may involve any number of variables from the current data set). For example, you might specify six categories based on combinations of three variables *Gender*, *Age*, and *Employment*.



Histograms

In general, histograms are used to examine frequency distributions of values of variables. For example, the frequency distribution plot shows which specific values or ranges of values of the examined variable are most frequent, how differentiated the values are, whether most observations are concentrated around the mean, whether the distribution is symmetrical or skewed, whether it is multimodal (i.e., has two or more peaks) or unimodal, etc. Histograms are also useful for evaluating the similarity of an observed distribution with theoretical or expected distributions.

Categorized Histograms allow you to produce histograms broken down by one or more categorical variables, or by any other one or more sets of logical categorization rules (see Categorization Methods).

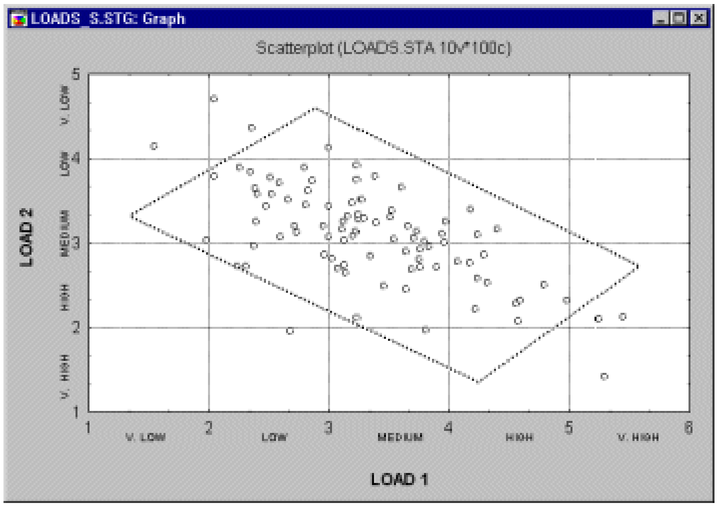
There are two major reasons why frequency distributions are of interest.

- One may learn from the shape of the distribution about the nature of the examined variable (e.g., a bimodal distribution may suggest that the sample is not homogeneous and consists of observations that belong to two populations that are more or less normally distributed).
- Many statistics are based on assumptions about the distributions of analyzed variables; histograms help one to test whether those assumptions are met.

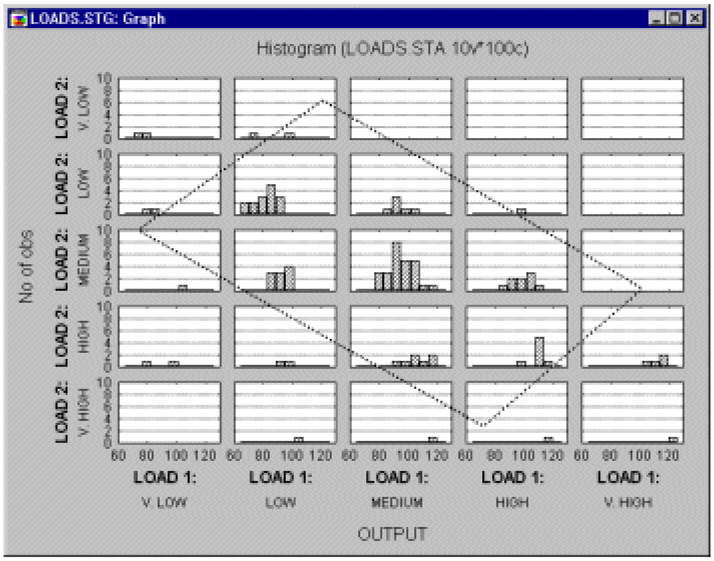
Often, the first step in the analysis of a new data set is to run histograms on all variables.

Histograms vs. Breakdown. Categorized Histograms provide information similar to breakdowns (e.g., mean, median, minimum, maximum, differentiation of values, etc.; see the Basic Statistics and Tables chapter). Although specific (numerical) descriptive statistics are easier to read in a table, the overall shape and global descriptive characteristics of a distribution are much easier to examine in a graph. Moreover, the graph provides qualitative information about the distribution that cannot be fully represented by any single index. For example, the overall skewed distribution of income may indicate that the majority of people have an income that is much closer to the minimum than maximum of the range of income. Moreover, when broken down by gender and ethnic background, this characteristic of the income distribution may be found to be more pronounced in certain subgroups. Although this information will be contained in the index of skewness (for each sub-group), when presented in the graphical form of a histogram, the information is usually more easily recognized and remembered. The histogram may also reveal "bumps" that may represent important facts about the specific social stratification of the investigated population or anomalies in the distribution of income in a particular group caused by a recent tax reform.

Categorized histograms and scatterplots. A useful application of the categorization methods for continuous variables is to represent the simultaneous relationships between three variables. Shown below is a scatterplot for two variables Load 1 and Load 2.



Now suppose you would like to add a third variable (Output) and examine how it is distributed at different levels of the joint distribution of Load 1 and Load 2. The following graph could be produced:



In this graph, Load 1 and Load 2 are both categorized into 5 intervals, and within each combination of intervals the distribution for variable Output is computed. Note that the "box" (parallelogram) encloses approximately the same observations (cases) in both graphs shown above.

Scatterplots

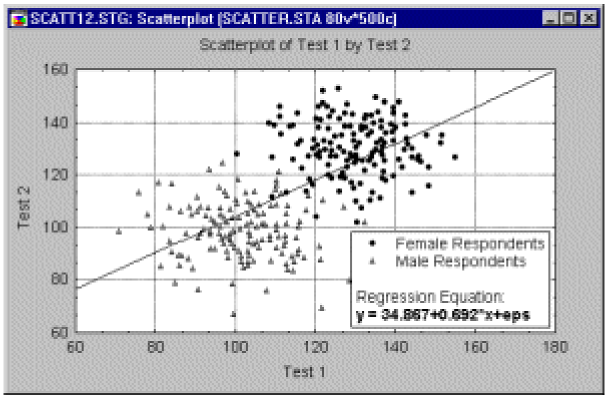
In general, two-dimensional scatterplots are used to visualize relations between two variables X and Y (e.g., weight and height). In scatterplots, individual data points are represented by point markers in two-dimensional space, where axes represent the variables. The two coordinates (X and Y) which determine the location of each point correspond to its specific values on the two variables. If the two variables are strongly related, then the data points form a systematic shape (e.g., a straight line or a clear curve). If the variables are not related, then the points form a round "cloud."

The categorized scatterplot option allows you to produce scatterplots categorized by one or more variables. Via the Multiple Subsets method (see Categorization Methods), you can also categorize the scatterplot based on logical selection conditions that define each category or group of observations.

Categorized scatterplots offer a powerful exploratory and analytic technique for investigating relationships between two or more variables within different sub-groups.

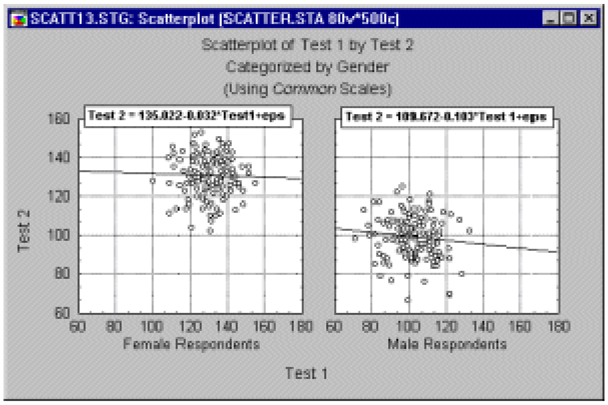
Homogeneity of Bivariate Distributions (Shapes of Relations). Scatterplots are typically used to identify the nature of relations between two variables (e.g., blood pressure and cholesterol level), because they can provide much more information than a correlation coefficient.

For example, a lack of homogeneity in the sample from which a correlation was calculated can bias the value of the correlation. Imagine a case where a correlation coefficient is calculated from data points which came from two different experimental groups, but this fact was ignored when the correlation was calculated. Suppose the experimental manipulation in one of the groups increased the values of both correlated variables, and thus the data from each group form a distinctive "cloud" in the scatterplot (as shown in the following illustration).



In this example, the high correlation is entirely due to the arrangement of the two groups, and it does not represent the "true" relation between the two variables, which is practically equal to 0 (as could be seen if one looked at each group separately).

If you suspect that such pattern may exist in your data and you know how to identify the possible "subsets" of data, then producing a categorized scatterplot



may yield a more accurate picture of the strength of the relationship between the X and Y variable, within each group (i.e., after controlling for group membership).

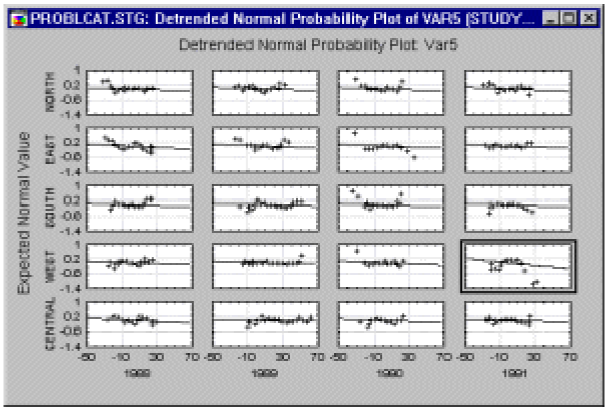
Curvilinear Relations. Curvilinearity is another aspect of the relationships between variables which can be examined in scatterplots. There are no "automatic" or easy-to-use tests to measure curvilinear relationships between variables: The standard Pearson r coefficient measures only linear relations; some nonparametric correlations such as the Spearman R can measure curvilinear relations, but not non-monotonous relations. Examining scatterplots allows one to identify the shape of relations, so that later an appropriate data transformation can be chosen to "straighten" the data or choose an appropriate nonlinear estimation equation to be fit.

For more information, refer to the chapters on Basic Statistics, Nonparametrics and Distributions, Multiple Regression, and Nonlinear Estimation.

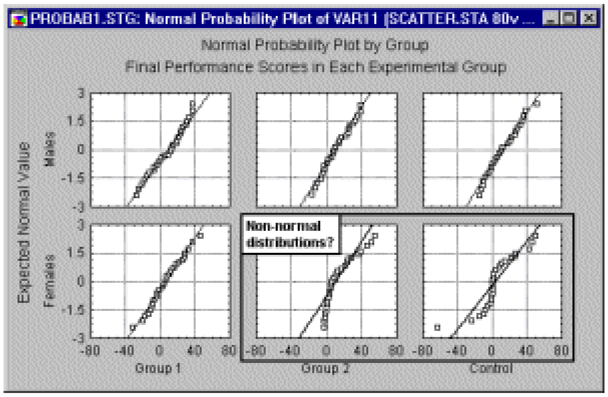
Probability Plots

Three types of categorized probability plots are Normal, Half-Normal, and Detrended. Normal probability plots provide a quick way to visually inspect to what extent the pattern of data follows a normal distribution.

Via categorized probability plots, one can examine how closely the distribution of a variable follows the normal distribution in different sub-groups.

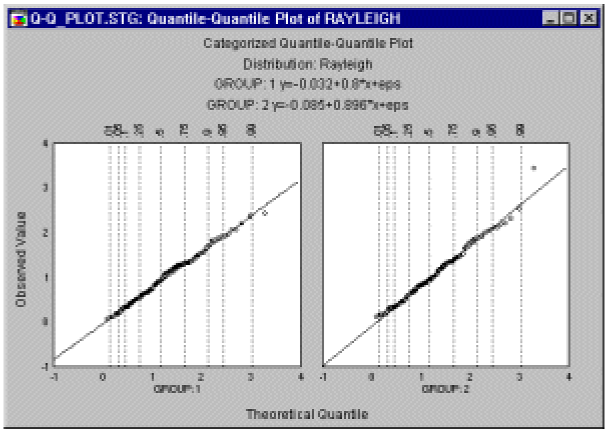


Categorized normal probability plots provide an efficient tool to examine the normality aspect of group homogeneity.



Quantile-Quantile Plots

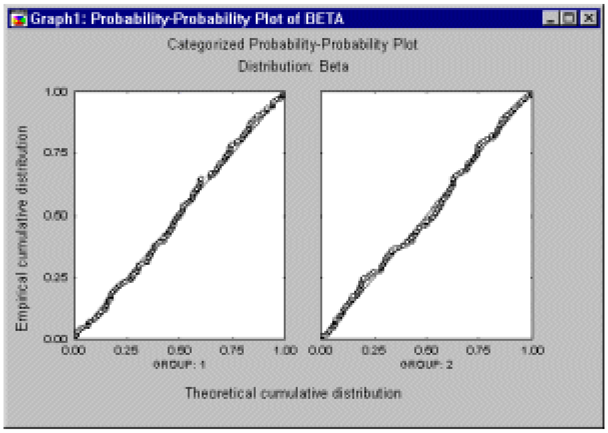
The categorized Quantile-Quantile (or Q-Q) plot is useful for finding the best fitting distribution within a family of distributions.



With Categorized Q-Q plots, a series of Quantile-Quantile (or Q-Q) plots, one for each category of cases identified by the X or X and Y category variables (or identified by the Multiple Subset criteria, see Categorization Methods) are produced. Examples of distributions which are used for Q-Q plots are the Exponential Distribution, Extreme Distribution, Normal, Rayleigh, Beta, Gamma, Lognormal, and Weibull distributions.

Probability-Probability Plots

The categorized Probability-Probability (or P-P) plot is useful for determining how well a specific theoretical distribution fits the observed data. This type of graph includes a series of Probability-Probability (or P-P) plots, one for each category of cases identified by the X or X and Y category variables (or identified by the Multiple Subset criteria, see Categorization Methods).

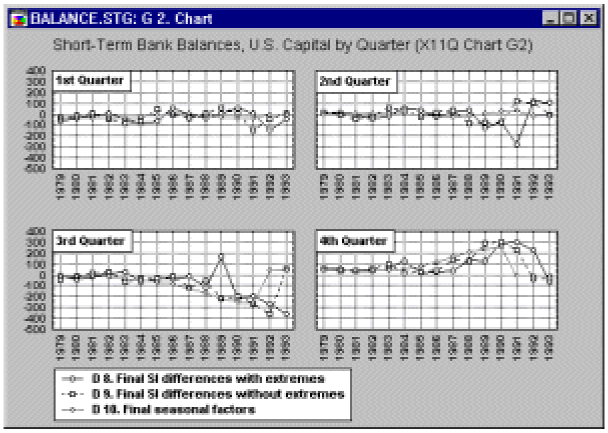


In the P-P plot, the observed cumulative distribution function (the proportion of non-missing values ≤ x) is plotted against a theoretical cumulative distribution function in order to assess the fit of the theoretical distribution to the observed data. If all points in this plot fall onto a diagonal line (with intercept 0 and slope 1), then you can conclude that the theoretical cumulative distribution adequately approximates the observed distribution.

If the data points do not all fall on the diagonal line, then you can use this plot to visually assess where the data do and do not follow the distribution (e.g., if the points form an S shape along the diagonal line, then the data may need to be transformed in order to bring them to the desired distribution pattern).

Line Plots

In line plots, individual data points are connected by a line. Line plots provide a simple way to visually present a sequence of many values (e.g., stock market quotes over a number of days). The categorized Line Plots graph is useful when one wants to view such data broken down (categorized) by a grouping variable (e.g., closing stock quotes on Mondays, Tuesdays, etc.) or some other logical criteria involving one or more other variables (e.g., closing quotes only for those days when two other stocks and the Dow Jones index went up, versus all other closing quotes; see Categorization Methods).

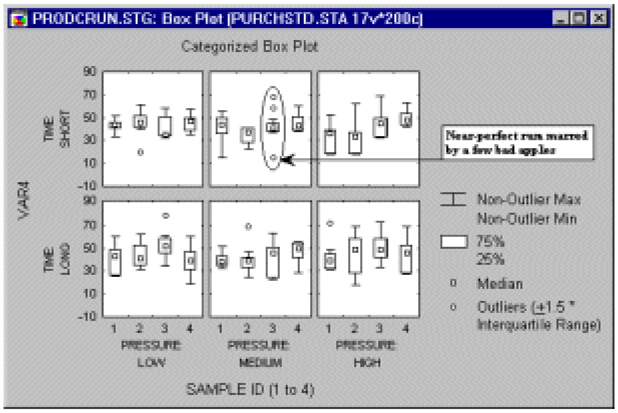


Box Plots

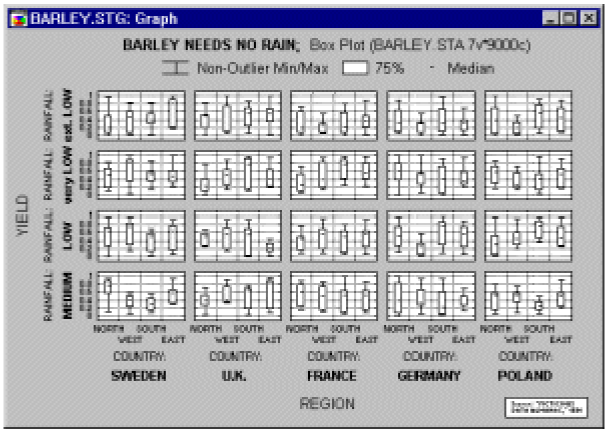
In Box Plots (the term first used by Tukey, 1970), ranges of values of a selected variable (or variables) are plotted separately for groups of cases defined by values of up to three categorical (grouping) variables, or as defined by Multiple Subsets categories.

The central tendency (e.g., median or mean), and range or variation statistics (e.g., quartiles, standard errors, or standard deviations) are computed for each group of cases, and the selected values are presented in one of five styles (Box Whiskers, Whiskers, Boxes, Columns, or High-Low Close). Outlier data points can also be plotted (see the sections on outliers and extremes).

For example, in the following graph, outliers (in this case, points greater or less than 1.5 times the inter-quartile range) indicate a particularly "unfortunate" flaw in an otherwise nearly perfect combination of factors:



However, in the following graph, no outliers or extreme values are evident.



There are two typical applications for box plots: (a) showing ranges of values for individual items, cases or samples (e.g., a typical MIN-MAX plot for stocks or commodities or aggregated sequence data plots with ranges), and (b) showing variation of scores in individual groups or samples (e.g., box and whisker plots presenting the mean for each sample as a point inside the box, standard errors as the box, and standard deviations around the mean as a narrower box or a pair of "whiskers").

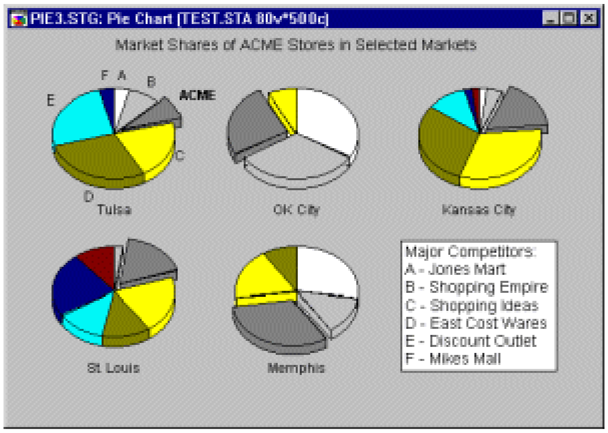
Box plots showing variation of scores allow one to quickly evaluate and "intuitively envision" the strength of the relation between the grouping and dependent variable. Specifically, assuming that the dependent variable is normally distributed, and knowing what proportion of observations fall, for example, within ±1 or ±2 standard deviations from the mean (see Elementary Concepts), one can easily evaluate the results of an experiment and say that, for example, the scores in about 95% of cases in experimental group 1 belong to a different range than scores in about 95% of cases in group 2.

In addition, so-called trimmed means (this term was first used by Tukey, 1962) may be plotted by excluding a user-specified percentage of cases from the extremes (i.e., tails) of the distribution of cases.

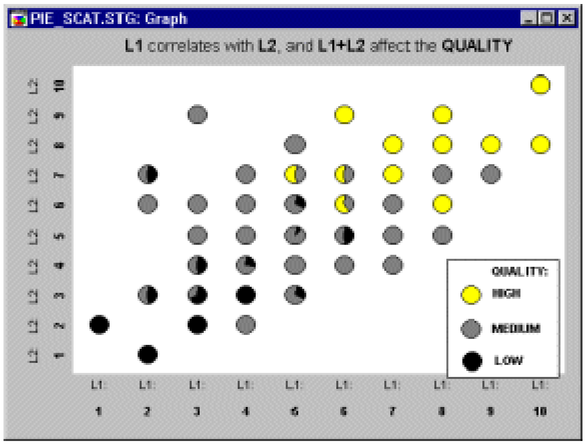
Pie Charts

The pie chart is one of the most common graph formats used for representing proportions or values of variables. This graph allows you to produce pie charts broken down by one or more other variables (e.g., grouping variables such as gender) or categorized according to some logical selection conditions that identify Multiple Subsets (see Categorization Methods).

For purposes of this discussion, categorized pie charts will always be interpreted as frequency pie charts (as opposed to data pie charts). This type of pie chart (sometimes called a frequency pie chart) interprets data like a histogram. It categorizes all values of the selected variable following the selected categorization technique and then displays the relative frequencies as pie slices of proportional sizes. Thus, these pie charts offer an alternative method to display frequency histogram data (see the section on Categorized Histograms).



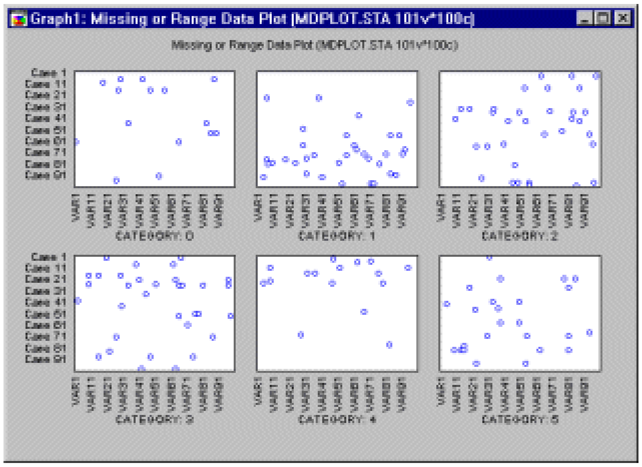
Pie-Scatterplots. Another useful application of categorized pie charts is to represent the relative frequency distribution of a variable at each "location" of the joint distribution of two other variables. Here is an example:



Note that pies are only drawn in "places" where there are data. Thus, the graph shown above takes on the appearance of a scatterplot (of variables L1 and L2), with the individual pies as point markers. However, in addition to the information contained in a simple scatterplot, each pie shows the relative distribution of a third variable at the respective location (i.e., Low, Medium, and High Quality).

Missing/Range Data Points Plots

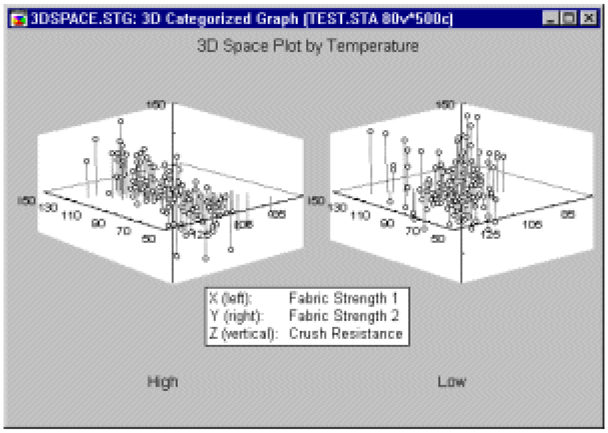
This graph produces a series of 2D graphs (one for each category of cases identified by the grouping variables or by the Multiple Subset criteria; see Categorization Methods) of missing data points and/or user-specified "out of range" points from which you can visualize the pattern or distribution of missing data (and/or user-specified "out of range" points) within each subset of cases (category).



This graph is useful in exploratory data analysis to determine the extent of missing (and/or "out of range") data and whether the patterns of those data occur randomly.

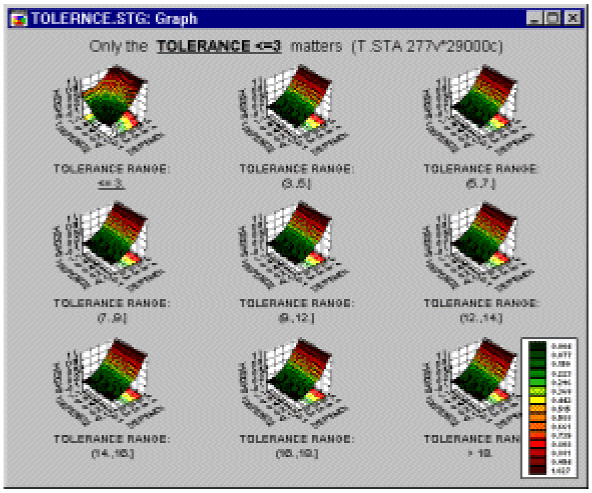
3D Plots

This type of graph allows you to produce 3D scatterplots (space plots, spectral plots, deviation plots, and trace plots), contour plots, and surface plots for subsets of cases defined by the specified categories of a selected variable or categories determined by user-defined case selection conditions (see Categorization Methods). Thus, the general purpose of this plot is to facilitate comparisons between groups or categories regarding the relationships between three or more variables.

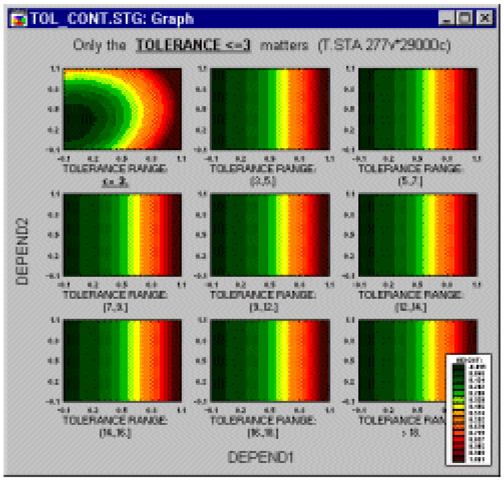

3DSPACE.STG: 3D Categorized Graph (TEST.STA 80v*500c)
3D Space Plot by Temperature
150
150
130
110
90
70
50
125
105
85
X (left): Fabric Strength 1
Y (right): Fabric Strength 2
Z (vertical): Crush Resistance
High
Low

Applications. In general, 3D XYZ graphs summarize the interactive relationships between three variables. The different ways in which data can be categorized (in a Categorized Graph) allow one to review those relationships contingent on some other criterion (e.g., group membership).

For example, from the categorized surface plot shown below, one can conclude that the setting of the tolerance level in an apparatus does not affect the investigated relationship between the measurements (Depend1, Depend2, and Height) unless the setting is ≤ 3.

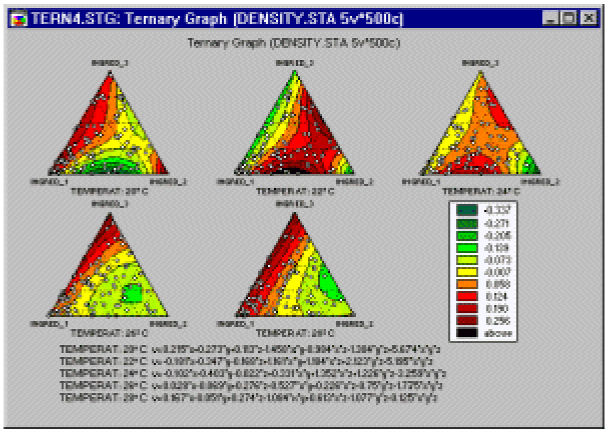


The effect is more salient when you switch to the contour plot representation.



Ternary Plots

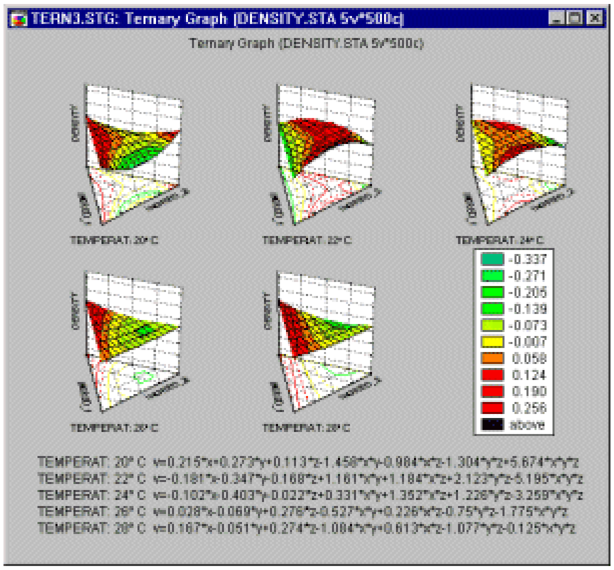
A categorized ternary plot can be used to examine relations between three or more dimensions where three of those dimensions represent components of a mixture (i.e., the relations between them is constrained such that the values of the three variables add up to the same constant for each case) for each level of a grouping variable.



In ternary plots, the triangular coordinate systems are used to plot four (or more) variables (the components X, Y, and Z, and the responses V1, V2, etc.) in two dimensions (ternary scatterplots or contours) or three dimensions (ternary surface plots). In order to produce ternary graphs, the relative proportions of each component within each case are constrained to add up to the same value (e.g., 1).

In a categorized ternary plot, one component graph is produced for each level of the grouping variable (or user-defined subset of data) and all the component graphs are arranged in one display to allow for comparisons between the subsets of data (categories).

Applications. A typical application of this graph is when the measured response(s) from an experiment depends on the relative proportions of three components (e.g., three different chemicals) which are varied in order to determine an optimal combination of those components (e.g., in mixture designs). This type of graph can also be used for other applications where relations between constrained variables need to be compared across categories or subsets of data.

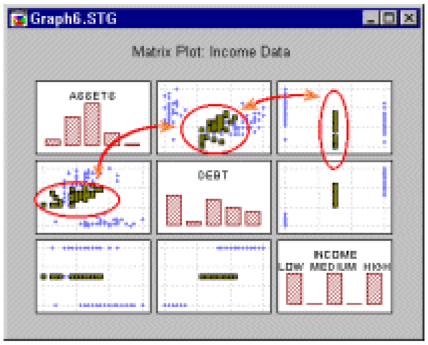


```
TEMPERAT: 20° C  v=0.215*x+0.273*y+0.113*z-1.458*x*y-0.984*x*z-1.304*y*z+5.674*x*y*z
TEMPERAT: 22° C  v=-0.181*x-0.347*y-0.168*z+1.161*x*y+1.184*x*z+2.123*y*z-5.195*x*y*z
TEMPERAT: 24° C  v=-0.102*x-0.403*y-0.022*z+0.331*x*y+1.352*x*z+1.226*y*z-3.259*x*y*z
TEMPERAT: 26° C  v=0.028*x-0.069*y+0.276*z-0.527*x*y+0.226*x*z-0.75*y*z-1.775*x*y*z
TEMPERAT: 28° C  v=0.167*x-0.051*y+0.274*z-1.084*x*y+0.613*x*z-1.077*y*z-0.125*x*y*z
```

Brushing

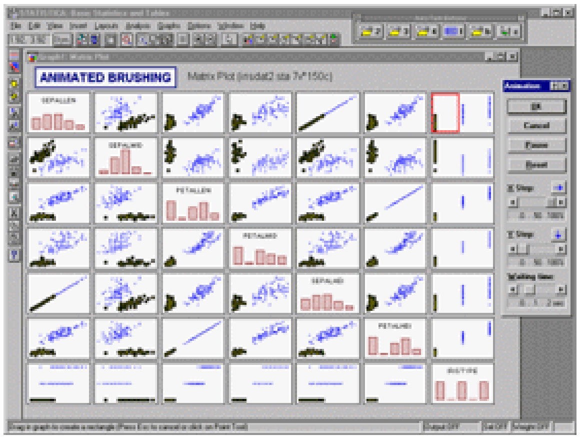
Perhaps the most common and historically first widely used technique explicitly identified as *graphical exploratory data analysis* is *brushing*, an interactive method allowing one to select on-screen specific data points or subsets of data and identify their (e.g., common) characteristics, or to examine their effects on relations between relevant variables (e.g., in scatterplot matrices) or to identify (e.g., label) outliers.

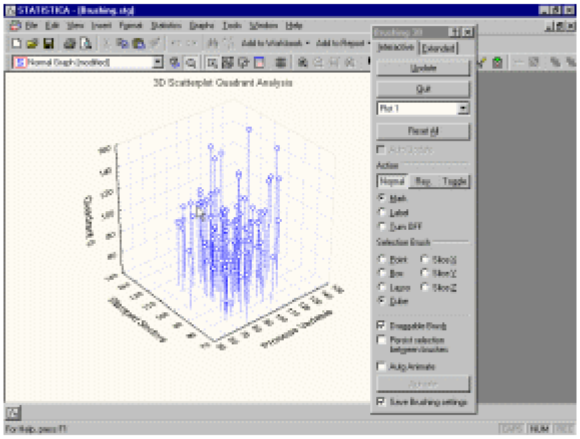
Those relations between variables can be visualized by fitted functions (e.g., 2D lines or 3D surfaces) and their confidence intervals, thus, for example, one can examine changes in those functions by interactively (temporarily) removing or adding specific subsets of data. For example, one of many applications of the brushing technique is to select (i.e., highlight) in a matrix scatterplot all data points that belong to a certain category (e.g., a "medium" income level, see the highlighted subset in the upper right component graph in illustration below):



in order to examine how those specific observations contribute to relations between other variables in the same data set (e.g, the correlation between the "debt" and "assets" in the current example).

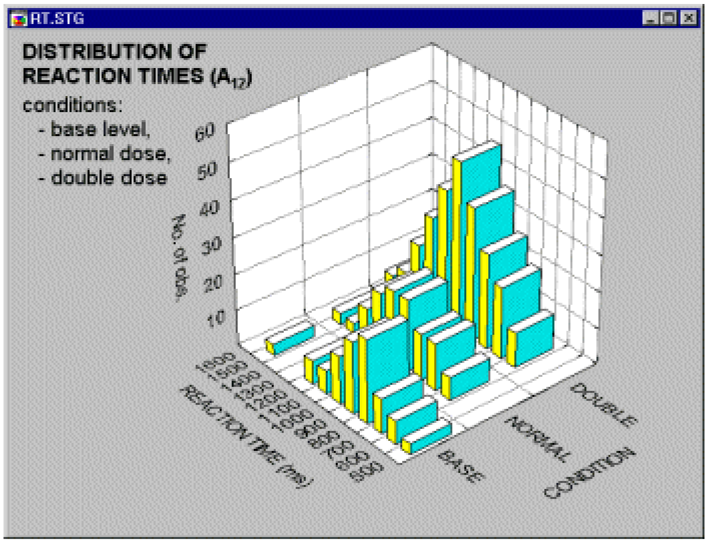
If the brushing facility supports features like "animated brushing" (see example below) or "automatic function re-fitting," one can define a dynamic brush that would move over the consecutive ranges of a criterion variable (e.g., "income" measured on a continuous scale and not a discrete scale as in the illustration to the above) and examine the dynamics of the contribution of the criterion variable to the relations between other relevant variables in the same data set.



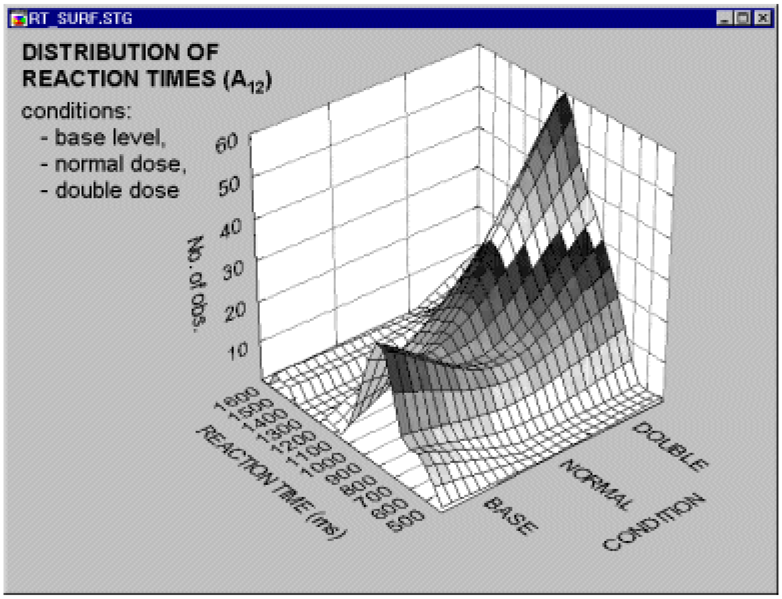


Smoothing Bivariate Distributions

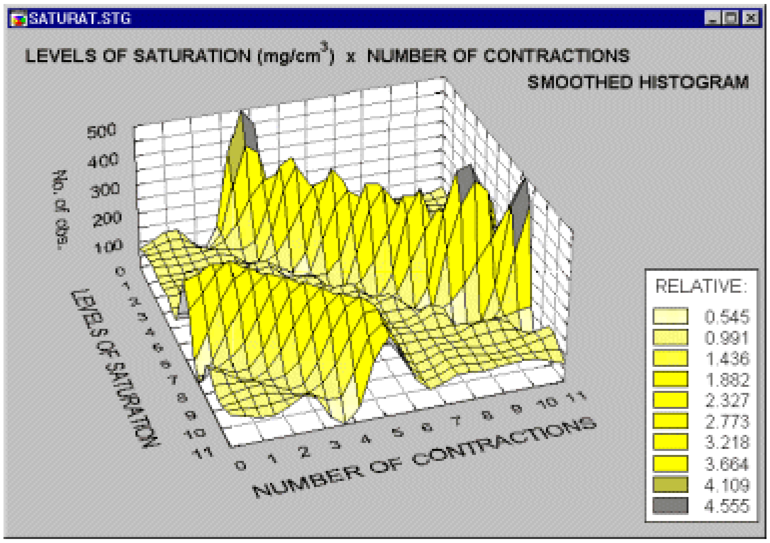
Three-dimensional histograms are used to visualize crosstabulations of values in two variables. They can be considered to be a conjunction of two simple (i.e., univariate) histograms, combined such that the frequencies of co-occurrences of values on the two analyzed variables can be examined. In a most common format of this graph, a 3D bar is drawn for each "cell" of the crosstabulation table and the height of the bar represents the frequency of values for the respective cell of the table. Different methods of categorization can be used for each of the two variables for which the bivariate distribution is visualized (see below).



If the software provides smoothing facilities, you can fit surfaces to 3D representations of bivariate frequency data. Thus, every 3D histogram can be turned into a smoothed surface. This technique is of relatively little help if applied to a simple pattern of categorized data (such as the histogram that was shown above).



However, if applied to more complex patterns of frequencies, it may provide a valuable exploratory technique,

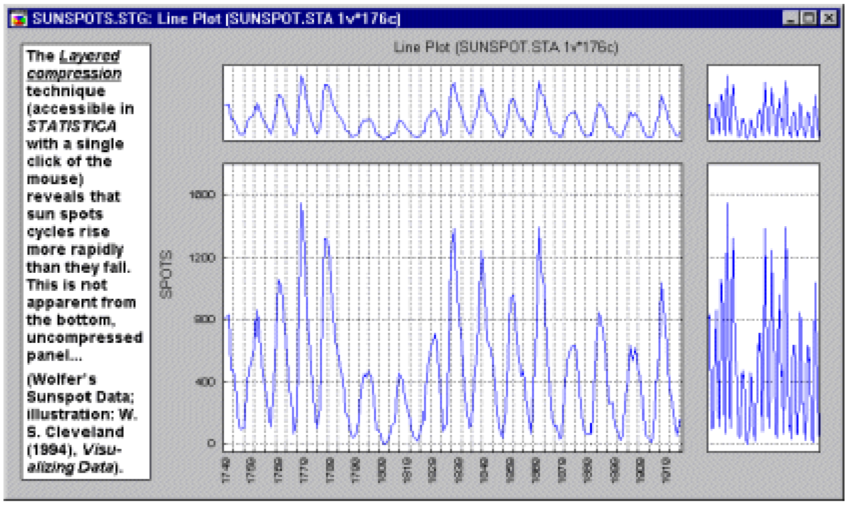


allowing identification of regularities which are less salient when examining the standard 3D histogram representations (e.g., see the systematic surface "wave-patterns" shown on the smoothed histogram above).

Layered Compression

When layered compression is used, the main graph plotting area is reduced in size to leave space for *Margin Graphs* in the upper and right side of the display (and a miniature graph in the corner). These smaller *Margin Graphs* represent vertically and horizontally compressed images (respectively) of the main graph.

In 2D graphs, layered compression is an exploratory data analysis technique that may facilitate the identification of otherwise obscured trends and patterns in 2-dimensional data sets. For example, in the following illustration


SUNSPOTS.STG: Line Plot (SUNSPOT.STA 1v*176c)

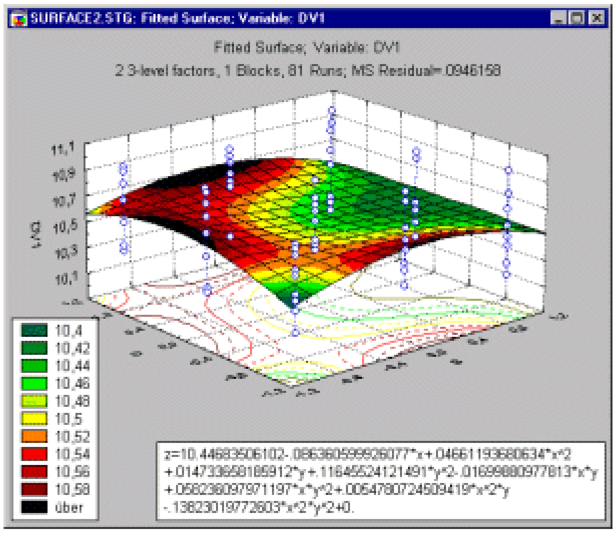
The *Layered compression* technique (accessible in *STATISTICA* with a single click of the mouse) reveals that sun spots cycles rise more rapidly than they fall. This is not apparent from the bottom, uncompressed panel...

(Wolfer's Sunspot Data; illustration: W. S. Cleveland (1994), *Visualizing Data*).

(based on an example discussed by Cleveland, 1993), it can be seen that the number of sunspots in each cycle decays more slowly than it rises at the onset of each cycle. This tendency is not readily apparent when examining the standard line plot; however, the compressed graph uncovers the hidden pattern.

Projections of 3D data sets

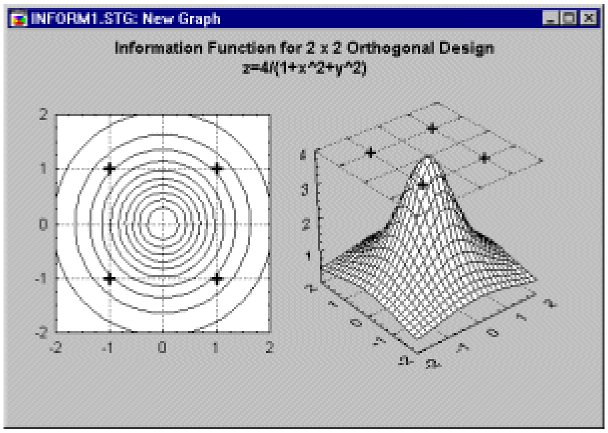
Contour plots generated by projecting surfaces (created from multivariate, typically three-variable, data sets) offer a useful method to explore and analytically examine the shapes of surfaces.


SURFACE2.STG: Fitted Surface; Variable: DV1

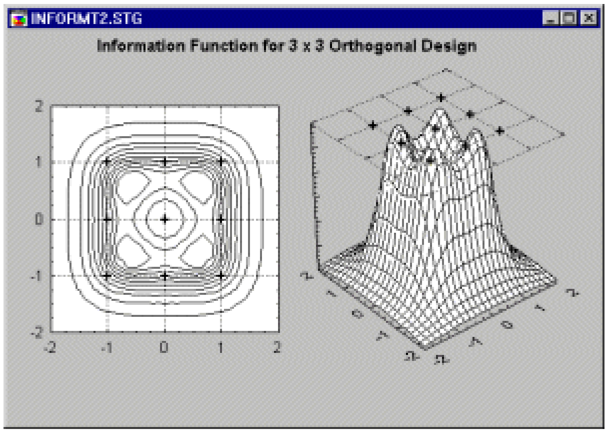
Fitted Surface; Variable: DV1
2 3-level factors, 1 Blocks, 81 Runs; MS Residual=.0946158

z=10.44683506102-.086360599926077*x+.04661193680634*x^2
+.014733658185912*y+.11645524121491*y^2-.016998809778 13*x*y
+.058236097971197*x*y^2+.0054780724509419*x^2*y
-.13823019772803*x^2*y^2+0.

As compared to surface plots, they may be less effective to quickly visualize the overall shape of 3D data structures,



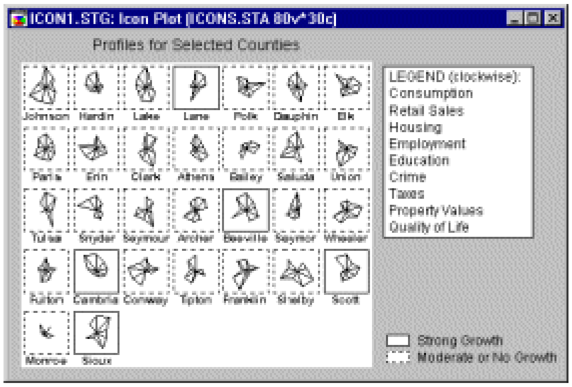
however, their main advantage is that they allow for precise examination and analysis of the shape of the surface



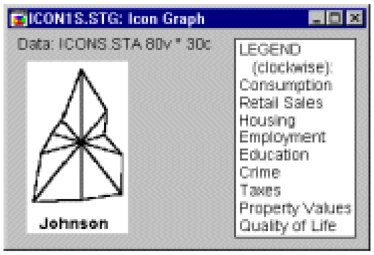
(*Contour Plots* display a series of undistorted horizontal "cross sections" of the surface).

Icon Plots

Icon Graphs represent cases or units of observation as multidimensional symbols and they offer a powerful although not easy to use exploratory technique. The general idea behind this method capitalizes on the human ability to "automatically" spot complex (sometimes interactive) relations between multiple variables if those relations are consistent across a set of instances (in this case "icons"). Sometimes the observation (or a "feeling") that certain instances are "somehow similar" to each other comes before the observer (in this case an analyst) can articulate which specific variables are responsible for the observed conisistency (Lewicki, Hill, & Czyzewska, 1992). However, further analysis that focuses on such intuitively spotted consistencies can reveal the specific nature of the relevant relations between variables.



The basic idea of icon plots is to represent individual units of observation as particular graphical objects where values of variables are assigned to specific features or dimensions of the objects (usually one case = one object). The assignment is such that the overall appearance of the object changes as a function of the configuration of values.



Thus, the objects are given visual "identities" that are unique for configurations of values and that can be identified by the observer. Examining such icons may help to discover specific clusters of both simple relations and interactions between variables.

Analyzing Icon Plots

The "ideal" design of the analysis of icon plots consists of five phases:

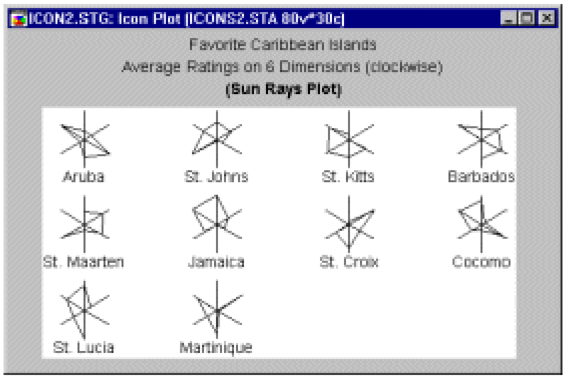
1. Select the order of variables to be analyzed. In many cases a random starting sequence is the best solution. You may also try to enter variables based on the order in a multiple regression equation, factor loadings on an interpretable factor (see the Factor Analysis chapter), or a similar multivariate technique. That method may simplify and "homogenize" the general appearance of the icons which may facilitate the identification of non-salient patterns. It may also, however, make some interactive patterns more difficult to find. No universal recommendations can be given at this point, other than to try the quicker (random order) method before getting involved in the more time-consuming method.

2. Look for any potential regularities, such as similarities between groups of icons, outliers, or specific relations between aspects of icons (e.g., "if the first two rays of the star icon are long, then one or two rays on the other side of the icon are usually short"). The Circular type of icon plots is recommended for this phase.
3. If any regularities are found, try to identify them in terms of the specific variables involved.
4. Reassign variables to features of icons (or switch to one of the sequential icon plots) to verify the identified structure of relations (e.g., try to move the related aspects of the icon closer together to facilitate further comparisons). In some cases, at the end of this phase it is recommended to drop the variables that appear not to contribute to the identified pattern.
5. Finally, use a quantitative method (such as a regression method, nonlinear estimation, discriminant function analysis, or cluster analysis) to test and quantify the identified pattern or at least some aspects of the pattern.

Taxonomy of Icon Plots

Most icon plots can be assigned to one of two categories: circular and sequential.

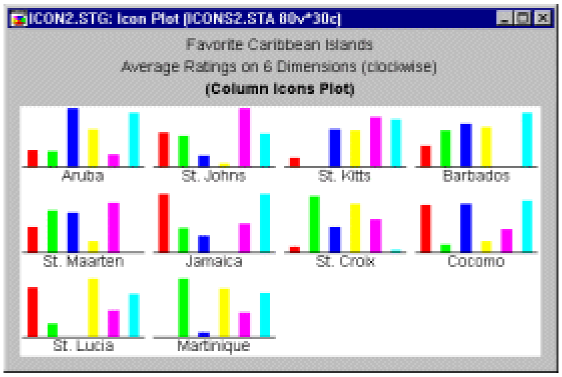
Circular icons. Circular icon plots (star plots, sun ray plots, polygon icons) follow a "spoked wheel" format where values of variables are represented by distances between the center ("hub") of the icon and its edges.


ICON2.STG: Icon Plot (ICONS2.STA 80v*30c)
Favorite Caribbean Islands
Average Ratings on 6 Dimensions (clockwise)
(Sun Rays Plot)
Aruba
St. Johns
St. Kitts
Barbados
St. Maarten
Jamaica
St. Croix
Cocomo
St. Lucia
Martinique

Those icons may help to identify interactive relations between variables because the overall shape of the icon may assume distinctive and identifiable overall patterns depending on multivariate configurations of values of input variables.

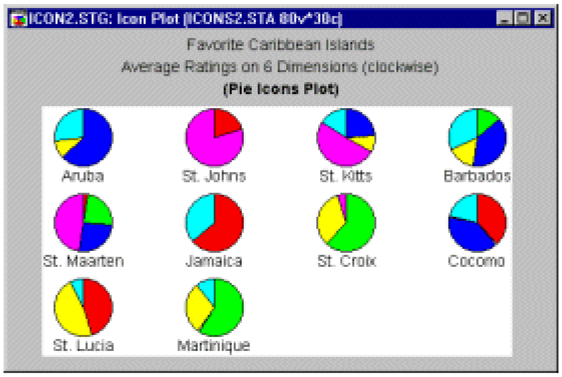
In order to translate such "overall patterns" into specific models (in terms of relations between variables) or verify specific observations about the pattern, it is helpful to switch to one of the sequential icon plots which may prove more efficient when one already knows what to look for.

Sequential icons. Sequential icon plots (column icons, profile icons, line icons) follow a simpler format where individual symbols are represented by small sequence plots (of different types).



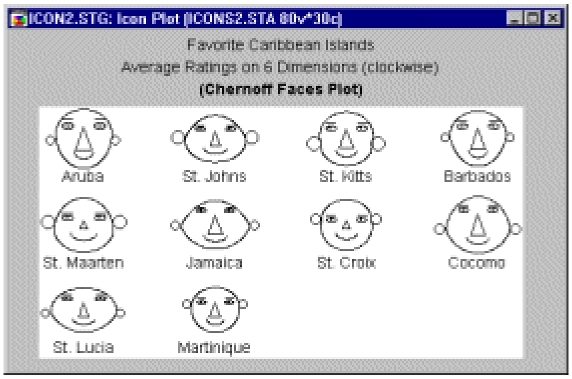
The values of consecutive variables are represented in those plots by distances between the base of the icon and the consecutive break points of the sequence (e.g., the height of the columns shown above). Those plots may be less efficient as a tool for the initial exploratory phase of icon analysis because the icons may look alike. However, as mentioned before, they may be helpful in the phase when some hypothetical pattern has already been revealed and one needs to verify it or articulate it in terms of relations between individual variables.

Pie icons. Pie icon plots fall somewhere in-between the previous two categories; all icons have the same shape (pie) but are sequentially divided in a different way according to the values of consecutive variables.



From a functional point of view, they belong rather to the sequential than circular category, although they can be used for both types of applications.

Chernoff faces. This type of icon is a category by itself. Cases are visualized by schematic faces such that relative values of variables selected for the graph are represented by variations of specific facial features.


ICON2.STG: Icon Plot (ICONS2.STA 80v*30c)
Favorite Caribbean Islands
Average Ratings on 6 Dimensions (clockwise)
(Chernoff Faces Plot)
Aruba
St. Johns
St. Kitts
Barbados
St. Maarten
Jamaica
St. Croix
Cocomo
St. Lucia
Martinique

Due to its unique features, it is considered by some researchers as an ultimate exploratory multivariate technique that is capable of revealing hidden patterns of interrelations between variables that cannot be uncovered by any other technique. This statement may be an exaggeration, however. Also, it must be admitted that Chernoff Faces is a method that is difficult to use, and it requires a great deal of experimentation with the assignment of variables to facial features. See also Data Mining Techniques.

Standardization of Values

Except for unusual cases when you intend for the icons to reflect the global differences in ranges of values between the selected variables, the values of the variables should be standardized once to assure within-icon compatibility of value ranges. For example, because the largest value sets the global scaling reference point for the icons, then if there are variables that are in a range of much smaller order, they may not appear in the icon at all, e.g., in a star plot, the rays that represent them will be too short to be visible.

Applications

Icon plots are generally applicable (1) to situations where one wants to find systematic patterns or clusters of observations, and (2) when one wants to explore possible complex relationships between several variables. The first type of application is similar to cluster analysis; that is, it can be used to classify observations.

For example, suppose you studied the personalities of artists, and you recorded the scores for several artists on a number of personality questionnaires. The icon plot may help you determine whether there are natural clusters of artists distinguished by particular patterns of scores on different questionnaires (e.g., you may find that some artists are very creative, undisciplined, and independent, while a second group is particularly intelligent, disciplined, and concerned with publicly-acknowledged success).

The second type of application -- the exploration of relationships between several variables -- is more similar to factor analysis; that is, it can be used to detect which variables tend to "go together." For example, suppose you were studying the structure of people’s perception of cars. Several subjects completed detailed questionnaires rating different cars on numerous dimensions. In the data file, the average ratings on each dimension (entered as the variables) for each car (entered as cases or observations) are recorded.

When you now study the Chernoff faces (each face representing the perceptions for one car), it may occur to you that smiling faces tend to have big ears; if price was assigned to the amount of smile and acceleration to the size of ears, then this "discovery" means that fast cars are more expensive. This, of course, is only a simple example; in real-life exploratory data analyses, non-obvious complex relationships between variables may become apparent.

Related Graphs

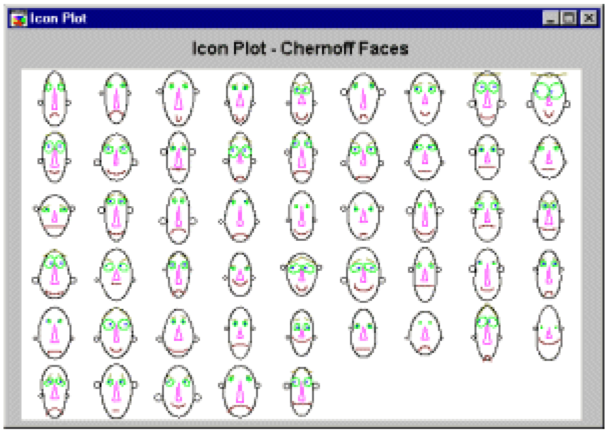
Matrix plots visualize relations between variables from one or two lists. If the software allows you to mark selected subsets, matrix plots may provide information similar to that in icon plots.

If the software allows you to create and identify user-defined subsets in scatterplots, simple 2D scatterplots can be used to explore the relationships between two variables; likewise, when exploring the relationships between three variables, 3D scatterplots provide an alternative to icon plots.

Graph Type

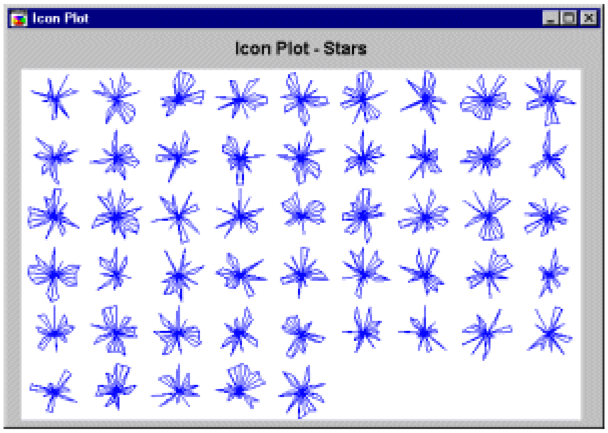
There are various types of Icon Plots.

Chernoff Faces. A separate "face" icon is drawn for each case; relative values of the selected variables for each case are assigned to shapes and sizes of individual facial features (e.g., length of nose, angle of eyebrows, width of face).

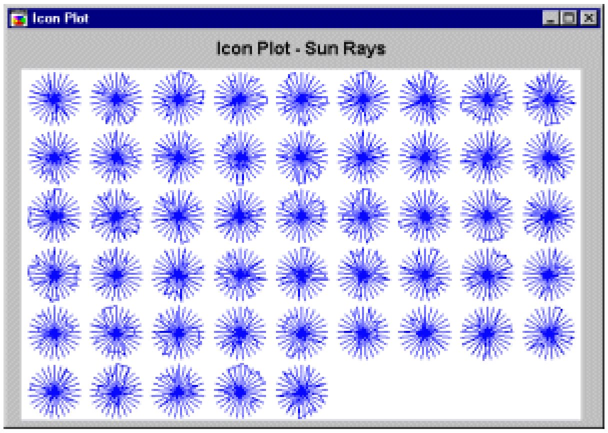


For more information see Chernoff Faces in Taxonomy of Icon Plots.

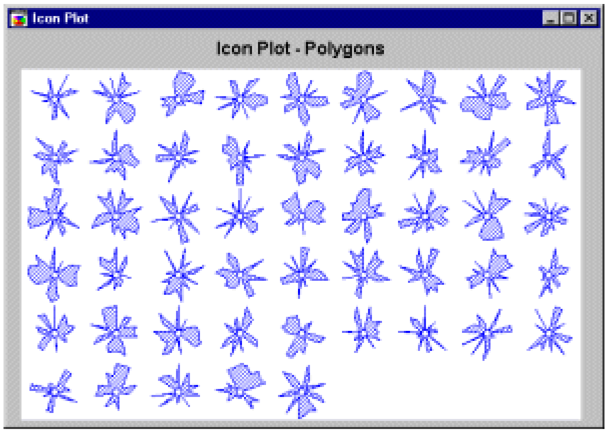
Stars. Star Icons is a circular type of icon plot. A separate star-like icon is plotted for each case; relative values of the selected variables for each case are represented (clockwise, starting at 12:00) by the length of individual rays in each star. The ends of the rays are connected by a line.



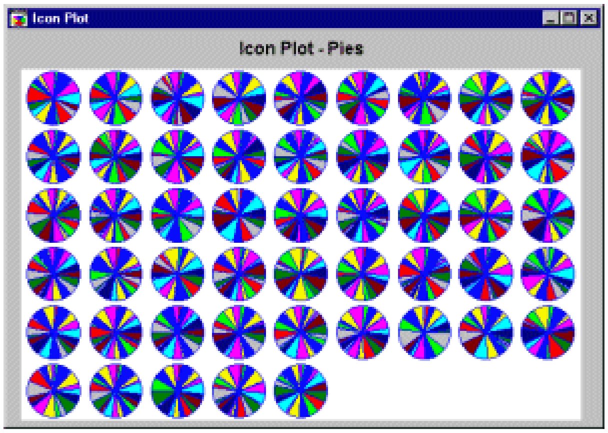
Sun Rays. Sun Ray Icons is a circular type of icon plot. A separate sun-like icon is plotted for each case; each ray represents one of the selected variables (clockwise, starting at 12:00), and the length of the ray represents the relative value of the respective variable. Data values of the variables for each case are connected by a line.



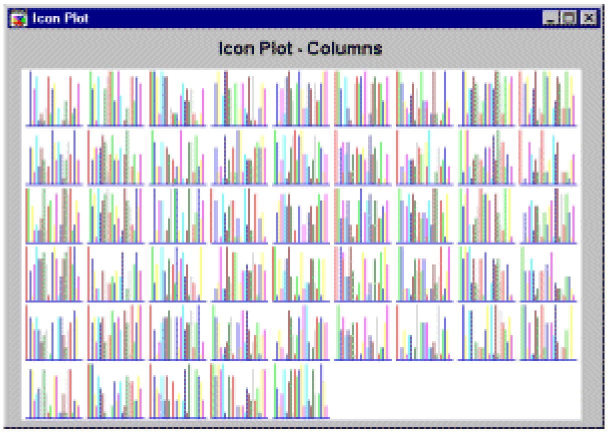
Polygons. Polygon Icons is a circular type of icon plot. A separate polygon icon is plotted for each case; relative values of the selected variables for each case are represented by the distance from the center of the icon to consecutive corners of the polygon (clockwise, starting at 12:00).



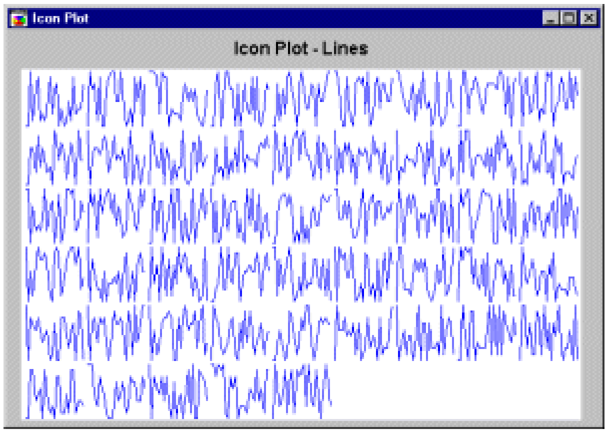
Pies. Pie Icons is a circular type of icon plot. Data values for each case are plotted as a pie chart (clockwise, starting at 12:00); relative values of selected variables are represented by the size of the pie slices.



Columns. Column Icons is a sequential type of icon plot. An individual column graph is plotted for each case; relative values of the selected variables for each case are represented by the height of consecutive columns.

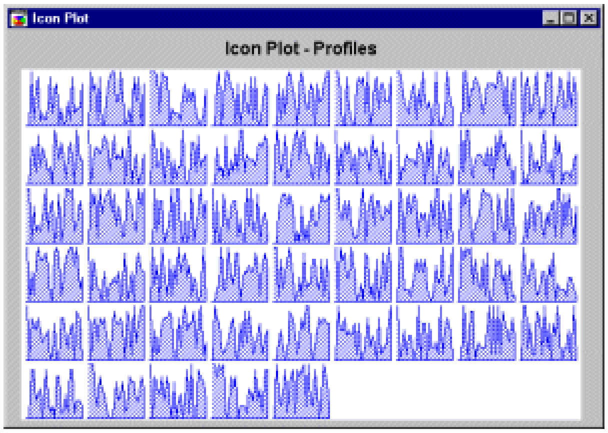


Lines. Line Icons is a sequential type of icon plot.



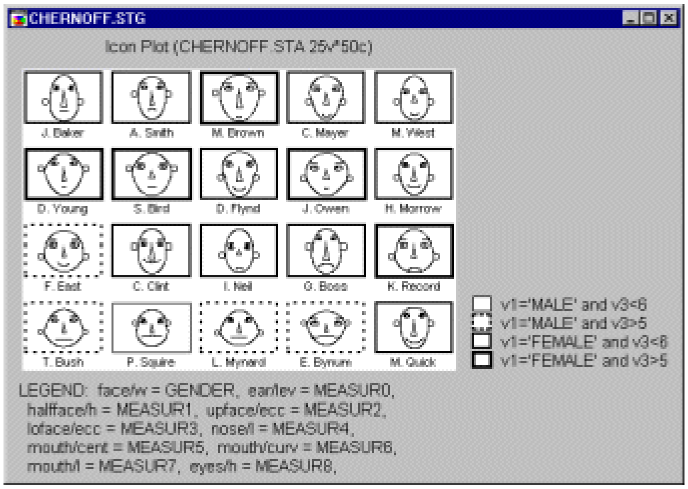
An individual line graph is plotted for each case; relative values of the selected variables for each case are represented by the height of consecutive break points of the line above the baseline.

Profiles. Profile Icons is a sequential type of icon plot. An individual area graph is plotted for each case; relative values of the selected variables for each case are represented by the height of consecutive peaks of the profile above the baseline.



Mark Icons

If the software allows you to specify multiple subsets, it is useful to specify the cases (subjects) whose icons will be marked (i.e., frames will be placed around the selected icons) in the plot.

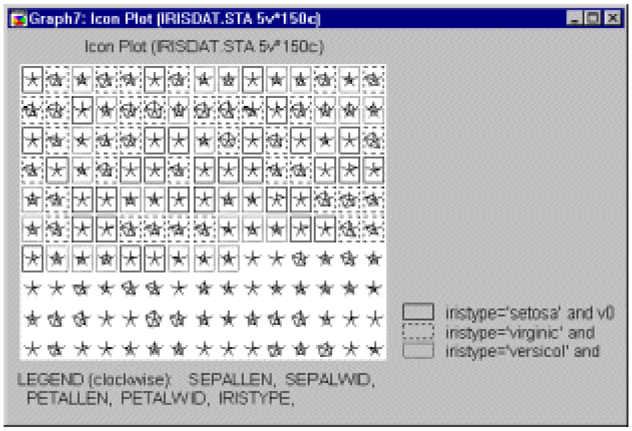

CHERNOFF.STG

Icon Plot (CHERNOFF.STA 25v*50c)

□ v1='MALE' and v3<6
⬚ v1='MALE' and v3>5
□ v1='FEMALE' and v3<6
▣ v1='FEMALE' and v3>5

LEGEND: face/w = GENDER, ear/lev = MEASUR0, halfface/h = MEASUR1, upface/ecc = MEASUR2, loface/ecc = MEASUR3, nose/l = MEASUR4, mouth/cent = MEASUR5, mouth/curv = MEASUR6, mouth/l = MEASUR7, eyes/h = MEASUR8,

The line patterns of frames which identify specific subsets should be listed in the legend along with the case selection conditions. The following graph shows an example of marked subsets.


Graph7: Icon Plot (IRISDAT.STA 5v*150c)

Icon Plot (IRISDAT.STA 5v*150c)

□ iristype='setosa' and v0
⬚ iristype='virginic' and
□ iristype='versicol' and

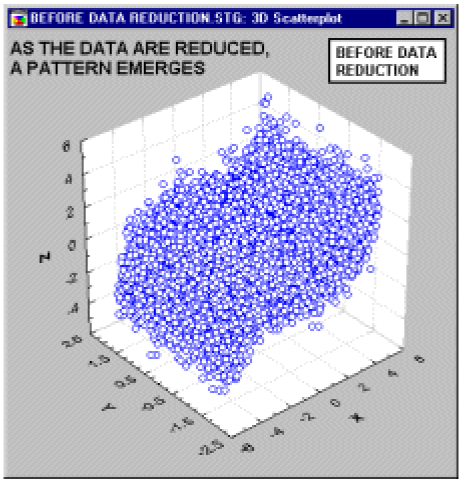
LEGEND (clockwise): SEPALLEN, SEPALWID, PETALLEN, PETALWID, IRISTYPE,

All cases (observations) which meet the condition specified in Subset 1 (i.e., cases for which the value of variable Iristype is equal to Setosa and for which the case number is less than 100) are marked with a specific frame around the selected icons.

All cases which meet the condition outlined in Subset 2 (i.e., cases for which the value of Iristype is equal to Virginic and for which the case number is less than 100) are assigned a different frame around the selected icons.

Data Reduction

Sometimes plotting an extremely large data set, can obscure an existing pattern (see the animation below). When you have a very large data file, it can be useful to plot only a subset of the data, so that the pattern is not hidden by the number of point markers.

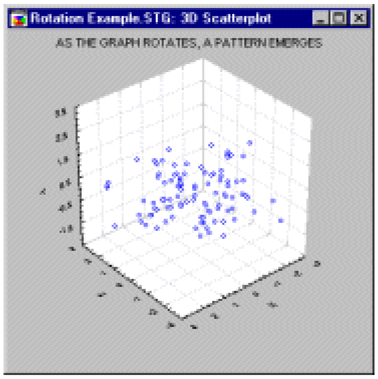

BEFORE DATA REDUCTION.STG: 3D Scatterplot
AS THE DATA ARE REDUCED, A PATTERN EMERGES
BEFORE DATA REDUCTION
Z
6
4
2
0
-2
-4
Y
2.5
1.5
0.5
-0.5
-1.5
-2.5
X
-6
-4
-2
0
2
4
6

Some software products offer methods for data reduction (or optimizing) which can be useful in these instances. Ideally, a data reduction option will allow you to specify an integer value *n* less than the number of cases in the data file. Then the software will randomly select approximately *n* cases from the available cases and create the plot based on these cases only.

Note that such data set (or sample size) reduction methods effectively draw a random sample from the current data set. Obviously, the nature of such data reduction is entirely different than when data are selectively reduced only to a specific subset or split into subgroups based on certain criteria (e.g., such as gender, region, or cholesterol level). The latter methods can be implemented interactively (e.g., using animated brushing facilities), or other techniques (e.g., categorized graphs or case selection conditions). All these methods can further aid in identifying patterns in large data sets.

Data Rotation (in 3D space)

Changing the viewpoint for 3D scatterplots (e.g., simple, spectral, or space plots) may prove to be an effective exploratory technique since it can reveal patterns that are easily obscured unless you look at the "cloud" of data points from an appropriate angle (see the animation below).


Rotation Example.STG: 3D Scatterplot
AS THE GRAPH ROTATES, A PATTERN EMERGES

Some software products offer interactive perspective, rotation, and continuous spinning controls which can be useful in these instances. Ideally, these controls will allow you to adjust the graph's angle and perspective to find the most informative location of the "viewpoint" for the graph as well as allowing you to control the vertical and horizontal rotation of the graph.

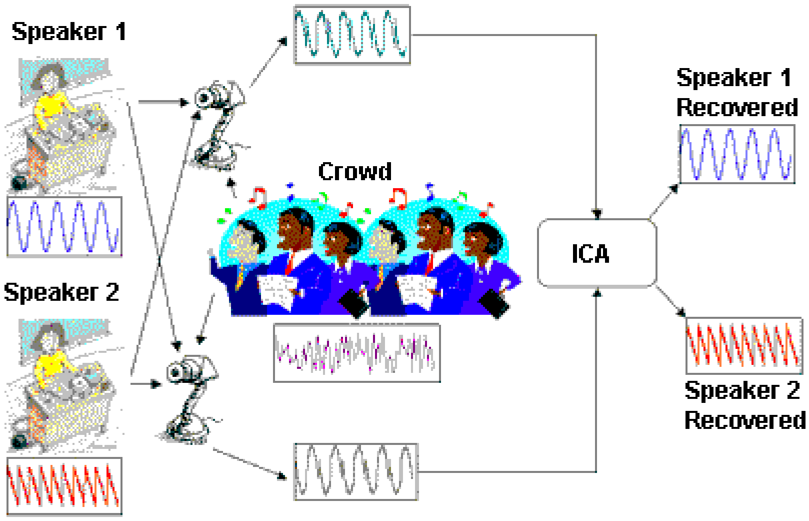
While these facilities are useful for initial exploratory data analysis, they can also be quite beneficial in exploring the factorial space (see Factor Analysis) and exploring the dimensional space (see Multidimensional Scaling).

Chapter 23
Independent Components Analysis

Introductory Overview

Independent Component Analysis is a well established and reliable statistical method that performs signal separation. Signal separation is a frequently occurring problem and is central to Statistical Signal Processing, which has a wide range of applications in many areas of technology ranging from Audio and Image Processing to Biomedical Signal Processing, Telecommunications, and Econometrics.

Imagine being in a room with a crowd of people and two speakers giving presentations at the same time. The crowed is making comments and noises in the background. We are interested in what the speakers say and not the comments emanating from the crowd. There are two microphones at different locations, recording the speakers' voices as well as the noise coming from the crowed. Our task is to separate the voice of each speaker while ignoring the background noise (see illustration below).



This is a classic example of the Independent Component Analysis, a well established stochastic technique. ICA can be used as a method of Blind Source Separation, meaning that it can separate independent signals from linear mixtures with virtually no prior knowledge on the signals. An example is decomposition of Electro or Magnetoencephalographic signals. In computational Neuroscience, ICA has been used for Feature Extraction, in which case it seems to adequately model the basic cortical processing of visual and auditory information. New application areas are being discovered at an increasing pace.

Chapter 24
Multiple Regression

- General Purpose
- Computational Approach
    - Least Squares
    - The Regression Equation
    - Unique Prediction and Partial Correlation
    - Predicted and Residual Scores
    - Residual Variance and R-square
    - Interpreting the Correlation Coefficient R
- Assumptions, Limitations, and Practical Considerations
    - Assumption of Linearity
    - Normality Assumption
    - Limitations
    - Choice of the number of variables
    - Multicollinearity and matrix ill-conditioning
    - Fitting centered polynomial models
    - The importance of residual analysis

General Purpose

The general purpose of multiple regression (the term was first used by Pearson, 1908) is to learn more about the relationship between several independent or predictor variables and a dependent or criterion variable. For example, a real estate agent might record for each listing the size of the house (in square feet), the number of bedrooms, the average income in the respective neighborhood according to census data, and a subjective rating of appeal of the house. Once this information has been compiled for various houses it would be interesting to see whether and how these measures relate to the price for which a house is sold. For example, one might learn that the number of bedrooms is a better predictor of the price for which a house sells in a particular neighborhood than how "pretty" the house is (subjective rating). One may also detect "outliers," that is, houses that should really sell for more, given their location and characteristics.

Personnel professionals customarily use multiple regression procedures to determine equitable compensation. One can determine a number of factors or dimensions such as "amount of responsibility" (*Resp*) or "number of people to supervise" (*No_Super*) that one believes to contribute to the value of a job. The personnel analyst then usually conducts a salary survey among comparable companies in the market, recording the salaries and respective characteristics (i.e., values on dimensions) for different positions. This information can be used in a multiple regression analysis to build a regression equation of the form:

Salary = .5*Resp + .8*No_Super

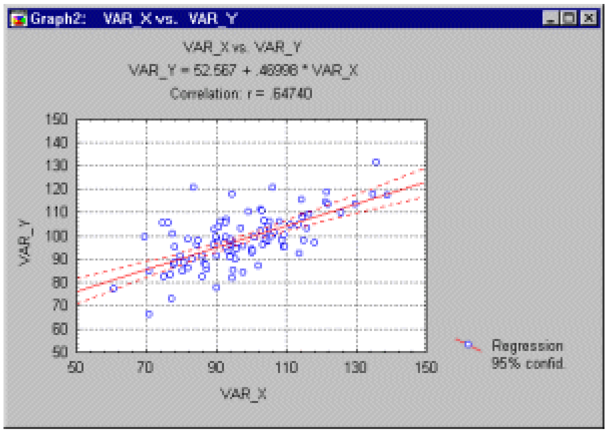
Once this so-called regression line has been determined, the analyst can now easily construct a graph of the expected (predicted) salaries and the actual salaries of job incumbents in his or her company. Thus, the analyst is able to determine which position is underpaid (below the regression line) or overpaid (above the regression line), or paid equitably.

In the social and natural sciences multiple regression procedures are very widely used in research. In general, multiple regression allows the researcher to ask (and hopefully answer) the general question "what is the best predictor of ...". For example, educational researchers might want to learn what are the best predictors of success in high-school. Psychologists may want to determine which personality variable best predicts social adjustment. Sociologists may want to find out which of the multiple social indicators best predict whether or not a new immigrant group will adapt and be absorbed into society.

See also Exploratory Data Analysis and Data Mining Techniques, the *General Stepwise Regression* chapter, and the *General Linear Models* chapter.

Computational Approach

The general computational problem that needs to be solved in multiple regression analysis is to fit a straight line to a number of points.



In the simplest case -- one dependent and one independent variable -- one can visualize this in a scatterplot.

- Least Squares
- The Regression Equation
- Unique Prediction and Partial Correlation

- Predicted and Residual Scores
- Residual Variance and R-square
- Interpreting the Correlation Coefficient R

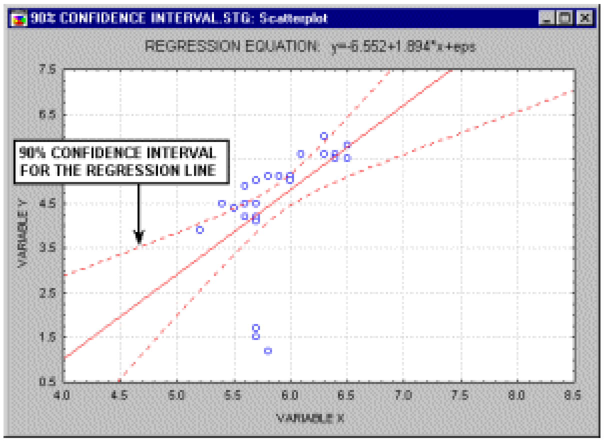
See also Exploratory Data Analysis and Data Mining Techniques, the *General Stepwise Regression* chapter, and the *General Linear Models* chapter.

Least Squares. In the scatterplot, we have an independent or *X* variable, and a dependent or *Y* variable. These variables may, for example, represent IQ (intelligence as measured by a test) and school achievement (grade point average; GPA), respectively. Each point in the plot represents one student, that is, the respective student's IQ and GPA. The goal of linear regression procedures is to fit a line through the points. Specifically, the program will compute a line so that the squared deviations of the observed points from that line are minimized. Thus, this general procedure is sometimes also referred to as *least squares estimation*.

See also Exploratory Data Analysis and Data Mining Techniques, the *General Stepwise Regression* chapter, and the *General Linear Models* chapter.

The Regression Equation. A line in a two dimensional or two-variable space is defined by the equation *Y=a+b*X*; in full text: the *Y* variable can be expressed in terms of a constant (*a*) and a slope (*b*) times the *X* variable. The constant is also referred to as the *intercept*, and the slope as the *regression coefficient* or *B coefficient*. For example, GPA may best be predicted as *1+.02*IQ*. Thus, knowing that a student has an *IQ* of 130 would lead us to predict that her GPA would be 3.6 (since, 1+.02*130=3.6).

For example, the animation below shows a two dimensional regression equation plotted with three different confidence intervals (90%, 95% and 99%).



In the multivariate case, when there is more than one independent variable, the regression line cannot be visualized in the two dimensional space, but can be computed just as easily. For example, if in addition to *IQ* we had additional predictors of achievement (e.g., *Motivation, Self- discipline*) we could construct a linear equation containing all those variables. In general then, multiple regression procedures will estimate a linear equation of the form:

$$Y = a + b_1*X_1 + b_2*X_2 + ... + b_p*X_p$$

Unique Prediction and Partial Correlation. Note that in this equation, the regression coefficients (or *B* coefficients) represent the *independent* contributions of each independent variable to the prediction of the dependent variable. Another way to express this fact is to say that, for example, variable X_1 is correlated with the *Y* variable, after controlling for all other independent variables. This type of correlation is also referred to as a *partial correlation* (this term was first used by Yule, 1907). Perhaps the following example will clarify this issue. One would probably find a significant negative correlation between hair length and height in the population (i.e., short people have longer hair). At first this may seem odd; however, if we were to add the variable *Gender* into the multiple regression equation, this correlation would probably disappear. This is because women, on the average, have longer hair than men; they also are shorter on the average than men. Thus, after we remove this gender difference by entering *Gender* into the equation, the relationship between hair length and height disappears because hair length does *not* make any unique contribution to the prediction of height, above and beyond what it shares in the prediction with variable *Gender*. Put another way, after controlling for the variable *Gender*, the partial correlation between hair length and height is zero.

Predicted and Residual Scores. The regression line expresses the best prediction of the dependent variable (*Y*), given the independent variables (*X*). However, nature is rarely (if ever) perfectly predictable, and usually there is substantial variation of the observed points around the fitted regression line (as in the scatterplot shown earlier). The deviation of a particular point from the regression line (its predicted value) is called the *residual* value.

Residual Variance and R-square. The smaller the variability of the residual values around the regression line relative to the overall variability, the better is our prediction. For example, if there is no relationship between the *X* and *Y* variables, then the ratio of the residual variability of the *Y* variable to the original variance is equal to 1.0. If *X* and *Y* are perfectly related then there is no residual variance and the ratio of variance would be 0.0. In most cases, the ratio would fall somewhere between these extremes, that is, between 0.0 and 1.0. 1.0 minus this ratio is referred to as *R-square* or the *coefficient of determination*. This value is immediately interpretable in the following manner. If we have an *R-square* of 0.4 then we know that the variability of the *Y* values around the regression line is 1-0.4 times the original variance; in other words we have explained 40% of the original variability, and are left with 60% residual variability. Ideally, we would like to explain most if not all of the original variability. The *R-square* value is an indicator of how well the model fits the data (e.g., an *R-square* close to 1.0 indicates that we have accounted for almost all of the variability with the variables specified in the model).

Interpreting the Correlation Coefficient R. Customarily, the degree to which two or more predictors (independent or *X* variables) are related to the dependent (*Y*) variable is expressed in the correlation coefficient *R*, which is the square root of *R-square*. In multiple regression, *R* can assume values between 0 and 1. To interpret the direction of the relationship between variables, one looks at the signs (plus or minus) of the regression or *B* coefficients. If a *B* coefficient is positive, then the relationship of this variable with the dependent variable is positive (e.g., the greater the IQ the better the grade point average); if the *B* coefficient is negative then the relationship is negative (e.g., the lower the class size the better the average test scores). Of course, if the *B* coefficient is equal to 0 then there is no relationship between the variables.

Assumptions, Limitations, Practical Considerations

- Assumption of Linearity
- Normality Assumption
- Limitations
- Choice of the number of variables
- Multicollinearity and matrix ill-conditioning
- The importance of residual analysis

Assumption of Linearity. First of all, as is evident in the name multiple *linear* regression, it is assumed that the relationship between variables is linear. In practice this assumption can virtually never be confirmed; fortunately, multiple regression procedures are not greatly affected by minor deviations from this assumption. However, as a rule it is prudent to *always* look at bivariate scatterplot of the variables of interest. If curvature in the relationships is evident, one may consider either transforming the variables, or explicitly allowing for nonlinear components.

See also Exploratory Data Analysis and Data Mining Techniques, the *General Stepwise Regression* chapter, and the *General Linear Models* chapter.

Normality Assumption. It is assumed in multiple regression that the residuals (predicted minus observed values) are distributed normally (i.e., follow the normal distribution). Again, even though most tests (specifically the *F*-test) are quite robust with regard to violations of this assumption, it is *always* a good idea, before drawing final conclusions, to review the distributions of the major variables of interest. You can produce histograms for the residuals as well as normal probability plots, in order to inspect the distribution of the residual values.

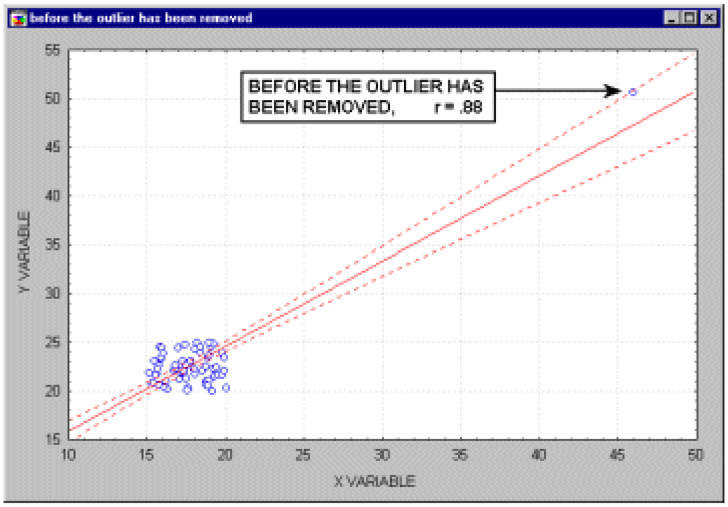
Limitations. The major conceptual limitation of all regression techniques is that one can only ascertain *relationships*, but never be sure about underlying *causal* mechanism. For example, one would find a strong positive relationship (correlation) between the damage that a fire does and the number of firemen involved in fighting the blaze. Do we conclude that the firemen cause the damage? Of course, the most likely explanation of this correlation is that the size of the fire (an external variable that we forgot to include in our study) caused the damage as well as the involvement of a certain number of firemen (i.e., the bigger the fire, the more firemen are called to fight the blaze). Even though this example is fairly obvious, in real correlation research, alternative causal explanations are often not considered.

Choice of the Number of Variables. Multiple regression is a seductive technique: "plug in" as many predictor variables as you can think of and usually at least a few of them will come out significant. This is because one is capitalizing on chance when simply including as many variables as one can think of as predictors of some other variable of interest. This problem is compounded when, in addition, the number of observations is relatively low. Intuitively, it is clear that one can hardly draw conclusions from an analysis of 100 questionnaire items based on 10 respondents. Most authors recommend that one should have at least 10 to 20 times as many observations (cases, respondents) as one has variables, otherwise the estimates of the regression line are probably very unstable and unlikely to replicate if one were to do the study over.

Multicollinearity and Matrix Ill-Conditioning. This is a common problem in many correlation analyses. Imagine that you have two predictors (*X* variables) of a person's height: (1) weight in pounds and (2) weight in ounces. Obviously, our two predictors are completely redundant; weight is one and the same variable, regardless of whether it is measured in pounds or ounces. Trying to decide which one of the two measures is a better predictor of height would be rather silly; however, this is exactly what one would try to do if one were to perform a multiple regression analysis with height as the dependent (*Y*) variable and the two measures of weight as the independent (X) variables. When there are very many variables involved, it is often not immediately apparent that this problem exists, and it may only manifest itself after several variables have already been entered into the regression equation. Nevertheless, when this problem occurs it means that at least one of the predictor variables is (practically) completely redundant with other predictors. There are many statistical indicators of this type of redundancy (tolerances, semi-partial *R*, etc., as well as some remedies (e.g., *Ridge regression*).

Fitting Centered Polynomial Models. The fitting of higher-order polynomials of an independent variable with a mean not equal to zero can create difficult multicollinearity problems. Specifically, the polynomials will be highly correlated due to the mean of the primary independent variable. With large numbers (e.g., Julian dates), this problem is very serious, and if proper protections are not put in place, can cause wrong results! The solution is to "center" the independent variable (sometimes, this procedures is referred to as "centered polynomials"), i.e., to subtract the mean, and then to compute the polynomials. See, for example, the classic text by Neter, Wasserman, & Kutner (1985, Chapter 9), for a detailed discussion of this issue (and analyses with polynomial models in general).

The Importance of Residual Analysis. Even though most assumptions of multiple regression cannot be tested explicitly, gross violations can be detected and should be dealt with appropriately. In particular outliers (i.e., extreme cases) can seriously bias the results by "pulling" or "pushing" the regression line in a particular direction (see the animation below), thereby leading to biased regression coefficients. Often, excluding just a single extreme case can yield a completely different set of results.



Chapter 25
Log-Linear Analysis of Frequency Tables

- General Purpose
- Two-way Frequency Tables
- Multi-Way Frequency Tables
- The Log-Linear Model
- Goodness-of-fit
- Automatic Model Fitting

General Purpose

One basic and straightforward method for analyzing data is via crosstabulation. For example, a medical researcher may tabulate the frequency of different symptoms by patients' age and gender; an educational researcher may tabulate the number of high school drop-outs by age, gender, and ethnic background; an economist may tabulate the number of business failures by industry, region, and initial capitalization; a market researcher may tabulate consumer preferences by product, age, and gender; etc. In all of these cases, the major results of interest can be summarized in a multi-way frequency table, that is, in a crosstabulation table with two or more factors.

Log-Linear provides a more "sophisticated" way of looking at crosstabulation tables. Specifically, you can test the different factors that are used in the crosstabulation (e.g., gender, region, etc.) and their interactions for statistical significance (see Elementary Concepts for a discussion of statistical significance testing). The following text will present a brief introduction to these methods, their logic, and interpretation.

Correspondence analysis is a descriptive/exploratory technique designed to analyze two-way and multi-way tables containing some measure of correspondence between the rows and columns. The results provide information which is similar in nature to those produced by Factor Analysis techniques, and they allow one to explore the structure of the categorical variables included in the table.

Two-way Frequency Tables

Let us begin with the simplest possible crosstabulation, the 2 by 2 table. Suppose we were interested in the relationship between age and the graying of people's hair. We took a sample of 100 subjects, and determined who does and does not have gray hair. We also recorded the approximate age of the subjects. The results of this study may be summarized as follows:

Gray Hair	Age		Total
	Below 40	40 or older	
No	40	5	45
Yes	20	35	55
Total	60	40	100

While interpreting the results of our little study, let us introduce the terminology that will allow us to generalize to complex tables more easily.

Design variables and response variables. In multiple regression (Multiple Regression) or analysis of variance (ANOVA/MANOVA) one customarily distinguishes between independent and dependent variables. Dependent variables are those that we are trying to explain, that is, that we hypothesize to depend on the independent variables. We could classify the factors in the 2 by 2 table accordingly: we may think of hair color (gray, not gray) as the dependent variable, and age as the independent variable. Alternative terms that are often used in the context of frequency tables are response variables and design variables, respectively. Response variables are those that vary in response to the design variables. Thus, in the example table above, hair color can be considered to be the response variable, and age the design variable.

Fitting marginal frequencies. Let us now turn to the analysis of our example table. We could ask ourselves what the frequencies would look like if there were no relationship between variables (the null hypothesis). Without going into details, intuitively one could expect that the frequencies in each cell would proportionately reflect the marginal frequencies (Totals). For example, consider the following table:

Gray Hair	Age		Total
	Below 40	40 or older	
No	27	18	45
Yes	33	22	55
Total	60	40	100

In this table, the proportions of the marginal frequencies are reflected in the individual cells. Thus, 27/33=18/22=45/55 and 27/18=33/22=60/40. Given the marginal frequencies, these are the cell frequencies that we would expect if there were no relationship between age and graying. If you compare this table with the previous one you will see that the previous table does reflect a relationship between the two variables: There are more than expected (under the null hypothesis) cases below age 40 without gray hair, and more cases above age 40 with gray hair.

This example illustrates the general principle on which the log-linear analysis is based: Given the marginal totals for two (or more) factors, we can compute the cell frequencies that would be expected if the two (or more) factors are unrelated. Significant deviations of the observed frequencies from those expected frequencies reflect a relationship between the two (or more) variables.

Model fitting approach. Let us now rephrase our discussion of the 2 by 2 table so far. We can say that fitting the model of two variables that are not related (age and hair color) amounts to computing the cell frequencies in the table based on the respective marginal frequencies (totals). Significant deviations of the observed table from those fitted frequencies reflect the lack of fit of the independence (between two variables) model. In that case we would reject that model for our data, and instead accept the model that allows for a relationship or association between age and hair color.

Multi-way Frequency Tables

The reasoning presented for the analysis of the 2 by 2 table can be generalized to more complex tables. For example, suppose we had a third variable in our study, namely whether or not the individuals in our sample experience stress at work. Because we are interested in the effect of stress on graying, we will consider Stress as another design variable. (Note that, if our study were concerned with the effect of gray hair on subsequent stress, variable stress would be the response variable, and hair color would be the design variable.). The resultant table is a three- way frequency table.

Fitting models. We can apply our previous reasoning to analyze this table. Specifically, we could fit different models that reflect different hypotheses about the data. For example, we could begin with a model that hypothesizes independence between all factors. As before, the expected frequencies in that case would reflect the respective marginal frequencies. If any significant deviations occur, we would reject this model.

Interaction effects. Another conceivable model would be that age is related to hair color, and stress is related to hair color, but the two (age and stress) factors do not interact in their effect. In that case, we would need to simultaneously fit the marginal totals for the two-way table of age by hair color collapsed across levels of stress, and the two-way table of stress by hair color collapsed across the levels of age. If this model does not fit the data, we would have to conclude that age, stress, and hair color all are interrelated. Put another way, we would conclude that age and stress interact in their effect on graying.

The concept of interaction here is analogous to that used in analysis of variance (ANOVA /MANOVA). For example, the age by stress interaction could be interpreted such that the relationship of age to hair color is modified by stress. While age brings about only little graying in the absence of stress, age is highly related when stress is present. Put another way, the effects of age and stress on graying are not additive, but interactive.

If you are not familiar with the concept of interaction, we recommend that you read the Introductory Overview to ANOVA/MANOVA. Many aspects of the interpretation of results from a log-linear analysis of a multi-way frequency table are very similar to ANOVA.

Iterative proportional fitting. The computation of expected frequencies becomes increasingly complex when there are more than two factors in the table. However, they can be computed, and, therefore, we can easily apply the reasoning developed for the 2 by 2 table to complex tables. The commonly used method for computing the expected frequencies is the so-called iterative proportional fitting procedure.

The Log-Linear Model

The term log-linear derives from the fact that one can, through logarithmic transformations, restate the problem of analyzing multi-way frequency tables in terms that are very similar to ANOVA. Specifically, one may think of the multi-way frequency table to reflect various main effects and interaction effects that add together in a linear fashion to bring about the observed table of frequencies. Bishop, Fienberg, and Holland (1974) provide details on how to derive log- linear equations to express the relationship between factors in a multi-way frequency table.

Goodness-of-Fit

In the previous discussion we have repeatedly made reference to the "significance" of deviations of the observed frequencies from the expected frequencies. One can evaluate the statistical significance of the goodness-of-fit of a particular model via a Chi-square test. You can compute two types of Chi-squares, the traditional Pearson Chi-square statistic and the maximum likelihood ratio Chi-square statistic (the term likelihood ratio was first introduced by Neyman and Pearson, 1931; the term maximum likelihood was first used by Fisher, 1922a). In practice, the interpretation and magnitude of those two Chi-square statistics are essentially identical. Both tests evaluate whether the expected cell frequencies under the respective model are significantly different from the observed cell frequencies. If so, the respective model for the table is rejected.

Reviewing and plotting residual frequencies. After one has chosen a model for the observed table, it is always a good idea to inspect the residual frequencies, that is, the observed minus the expected frequencies. If the model is appropriate for the table, then all residual frequencies should be "random noise," that is, consist of positive and negative values of approximately equal magnitudes that are distributed evenly across the cells of the table.

Statistical significance of effects. The Chi-squares of models that are hierarchically related to each other can be directly compared. For example, if we first fit a model with the age by hair color interaction and the stress by hair color interaction, and then fit a model with the age by stress by hair color (three-way) interaction, then the second model is a superset of the previous model. We could evaluate the difference in the Chi-square statistics, based on the difference in the degrees of freedom; if the differential Chi-square statistic is significant, then we would conclude that the three-way interaction model provides a significantly better fit to the observed table than the model without this interaction. Therefore, the three-way interaction is statistically significant.

In general, two models are hierarchically related to each other if one can be produced from the other by either adding terms (variables or interactions) or deleting terms (but not both at the same time).

Automatic Model Fitting

When analyzing four- or higher-way tables, finding the best fitting model can become increasingly difficult. You can use automatic model fitting options to facilitate the search for a "good model" that fits the data. The general logic of this algorithm is as follows. First, fit a model with no relationships between factors; if that model does not fit (i.e., the respective Chi- square statistic is significant), then it will fit a model with all two-way interactions. If that model does not fit either, then the program will fit all three-way interactions, and so on. Let us assume that this process found the model with all two-way interactions to fit the data. The program will then proceed to eliminate all two-way interactions that are not statistically significant. The resulting model will be the one that includes the least number of interactions necessary to fit the observed table.

Chapter 26
Multivariate Adaptive Regression Splines (MARSplines)

- Introductory Overview
    - Regression Problems
    - Multivariate Adaptive Regression Splines
    - Model Selection and Pruning
    - Applications
- Technical Notes: The MARSplines Algorithm
- Technical Notes: The MARSplines Model

Introductory Overview

Multivariate Adaptive Regression Splines (MARSplines) is an implementation of techniques popularized by Friedman (1991) for solving regression-type problems (see also, Multiple Regression), with the main purpose to predict the values of a continuous dependent or outcome variable from a set of independent or predictor variables. There are a large number of methods available for fitting models to continuous variables, such as a linear regression [e.g., *Multiple Regression*, *General Linear Model (GLM)*], nonlinear regression *(Generalized Linear/Nonlinear Models)*, regression trees (see *Classification and Regression Trees*), *CHAID*, *Neural Networks*, etc. (see also Hastie, Tishirani, and Friedman, 2001, for an overview).

Multivariate Adaptive Regression Splines (*MARSplines*) is a nonparametric regression procedure that makes no assumption about the underlying functional relationship between the dependent and independent variables. Instead, *MARSplines* constructs this relation from a set of coefficients and basis functions that are entirely "driven" from the regression data. In a sense, the method is based on the "divide and conquer" strategy, which partitions the input space into regions, each with its own regression equation. This makes *MARSplines* particularly suitable for problems with higher input dimensions (i.e., with more than 2 variables), where the curse of dimensionality would likely create problems for other techniques.

The *MARSplines* technique has become particularly popular in the area of data mining because it does not assume or impose any particular type or class of relationship (e.g., linear, logistic, etc.) between the predictor variables and the dependent (outcome) variable of interest. Instead, useful models (i.e., models that yield accurate predictions) can be derived even in situations where the relationship between the predictors and the dependent variables is non-monotone and difficult to approximate with parametric models. For more information about this technique and how it compares to other methods for nonlinear regression (or regression trees), see Hastie, Tishirani, and Friedman (2001).

Regression Problems

Regression problems are used to determine the relationship between a set of dependent variables (also called output, outcome, or response variables) and one or more independent variables (also known as input or predictor variables). The dependent variable is the one whose values you want to predict, based on the values of the independent (predictor) variables. For instance, one might be interested in the number of car accidents on the roads, which can be caused by 1) bad weather and 2) drunk driving. In this case one might write, for example,

Number_of_Accidents = Some Constant + 0.5*Bad_Weather + 2.0*Drunk_Driving

The variable *Number of Accidents* is the dependent variable that is thought to be caused by (among other variables) *Bad Weather* and *Drunk Driving* (hence the name dependent variable). Note that the independent variables are multiplied by factors, i.e., *0.5* and *2.0*. These are known as regression coefficients. The larger these coefficients, the stronger the influence of the independent variables on the dependent variable. If the two predictors in this simple (fictitious) example were measured on the same scale (e.g., if the variables were standardized to a mean of *0.0* and standard deviation *1.0*), then *Drunk Driving* could be inferred to contribute 4 times more to car accidents than *Bad Weather*. (If the variables are not measured on the same scale, then direct comparisons between these coefficients are not meaningful, and, usually, some other standardized measure of predictor "importance" is included in the results.)

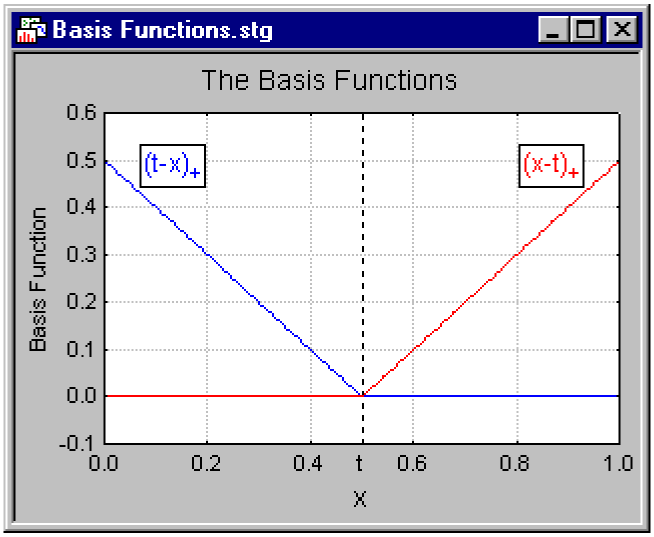
For additional details regarding these types of statistical models, refer to *Multiple Regression* or *General Linear Models (GLM)*, as well as *General Regression Models (GRM)*. In general, the social and natural sciences regression procedures are widely used in research. Regression allows the researcher to ask (and hopefully answer) the general question "what is the best predictor of ..." For example, educational researchers might want to learn what the best predictors of success in high-school are. Psychologists may want to determine which personality variable best predicts social adjustment. Sociologists may want to find out which of the multiple social indicators best predict whether a new immigrant group will adapt and be absorbed into society.

Multivariate Adaptive Regression Splines

The car accident example we considered previously is a typical application for linear regression, where the response variable is hypothesized to depend linearly on the predictor variables. Linear regression also falls into the category of so-called parametric regression, which assumes that the nature of the relationships (but not the specific parameters) between the dependent and independent variables is known *a priori* (e.g., is linear). By contrast, nonparametric regression (see *Nonparametrics*) does not make any such assumption as to how the dependent variables are related to the predictors. Instead it allows the regression function to be "driven" directly from data.

Multivariate Adaptive Regression Splines (MARSplines) is a nonparametric regression procedure that makes no assumption about the underlying functional relationship between the dependent and independent variables. Instead, *MARSplines* constructs this relation from a set of coefficients and so-called basis functions that are entirely determined from the regression data. You can think of the general "mechanism" by which the *MARSplines* algorithm operates as multiple piecewise linear regression (see *Nonlinear Estimation*), where each breakpoint (estimated from the data) defines the "region of application" for a particular (very simple) linear regression equation.

Basis functions. Specifically, *MARSplines* uses two-sided truncated functions of the form (as shown below) as basis functions for linear or nonlinear expansion, which approximates the relationships between the response and predictor variables.



Shown above is a simple example of two basis functions (t-x)+ and (x-t)+ (adapted from Hastie, et al., 2001, Figure 9.9). Parameter *t* is the knot of the basis functions (defining the "pieces" of the piecewise linear regression); these knots (parameters) are also determined from the data. The "+" signs next to the terms *(t-x)* and *(x-t)* simply denote that only positive results of the respective equations are considered; otherwise the respective functions evaluate to zero. This can also be seen in the illustration.

The MARSplines model. The basis functions together with the model parameters (estimated via least squares estimation) are combined to produce the predictions given the inputs. The general *MARSplines* model equation (see Hastie et al., 2001, equation 9.19) is given as:

$$y = f(X) = \beta_o + \sum_{m=1}^{M} \beta_m h_m(X)$$

where the summation is over the *M* nonconstant terms in the model (further details regarding the model are also provided in Technical Notes). To summarize, *y* is predicted as a function of the predictor variables *X* (and their interactions); this function consists of an intercept parameter (β_0) and the weighted (by β_m) sum of one or more basis functions $h_m(X)$, of the kind illustrated earlier. You can also think of this model as "selecting" a weighted sum of basis functions from the set of (a large number of) basis functions that span all values of each predictor (i.e., that set would consist of one basis function, and parameter *t*, for each distinct value for each predictor variable). The *MARSplines* algorithm then searches over the space of all inputs and predictor values (knot locations t) as well as interactions between variables. During this search, an increasingly larger number of basis functions are added to the model (selected from the set of possible basis functions), to maximize an overall least squares goodness-of-fit criterion. As a result of these operations, *MARSplines* automatically determines the most important independent variables as well as the most significant interactions among them. The details of this algorithm are further described in Technical Notes, as well as in Hastie et al., 2001).

Categorical predictors. In practice, both continuous and categorical predictors could be used, and will often yield useful results. However, the basic *MARSplines* algorithm assumes that the predictor variables are continuous in nature, and, for example, the computed knots program will usually not coincide with actual class codes found in the categorical predictors. For a detailed discussion of categorical predictor variables in *MARSplines*, see Friedman (1993).

Multiple dependent (outcome) variables. The *MARSplines* algorithm can be applied to multiple dependent (outcome) variables. In this case, the algorithm will determine a common set of basis functions in the predictors, but estimate different coefficients for each dependent variable. This method of treating multiple outcome variables is not unlike some neural networks architectures, where multiple outcome variables can be predicted from common neurons and hidden layers; in the case of *MARSplines*, multiple outcome variables are predicted from common basis functions, with different coefficients.

MARSplines and classification problems. Because *MARSplines* can handle multiple dependent variables, it is easy to apply the algorithm to classification problems as well. First, code the classes in the categorical response variable into multiple indicator variables (e.g., 1 = observation belongs to class k, 0 = observation does not belong to class k); then apply the *MARSplines* algorithm to fit a model, and compute predicted (continuous) values or scores; finally, for prediction, assign each case to the class for which the highest score is predicted (see also Hastie, Tibshirani, and Freedman, 2001, for a description of this procedure). Note that this type of application will yield heuristic classifications that may work very well in practice, but is not based on a statistical model for deriving classification probabilities.

Model Selection and Pruning

In general, nonparametric models are adaptive and can exhibit a high degree of flexibility that may ultimately result in overfitting if no measures are taken to counteract it. Although such models can achieve zero error on training data, they have the tendency to perform poorly when presented with new observations or instances (i.e., they do not generalize well to the prediction of "new" cases). *MARSplines*, like most methods of this kind, tend to overfit the data as well. To combat this problem, *MARSplines* uses a pruning technique (similar to pruning in classification trees) to limit the complexity of the model by reducing the number of its basis functions.

MARSplines as a predictor (feature) selection method. This feature - the selection of and pruning of basis functions - makes this method a very powerful tool for predictor selection. The *MARSplines* algorithm will pick up only those basis functions (and those predictor variables) that make a "sizeable" contribution to the prediction (refer to Technical Notes for details).

Applications

Multivariate Adaptive Regression Splines (MARSplines) have become very popular recently for finding predictive models for "difficult" data mining problems, i.e., when the predictor variables do not exhibit simple and/or monotone relationships to the dependent variable of interest. Alternative models or approaches that you can consider for such cases are *CHAID*, *Classification and Regression Trees*, or any of the many *Neural Networks* architectures available. Because of the specific manner in which *MARSplines* selects predictors (basis functions) for the model, it does generally "well" in situations where regression-tree models are also appropriate, i.e., where hierarchically organized successive splits on the predictor variables yield good (accurate) predictions. In fact, instead of considering this technique as a generalization of multiple regression (as it was presented in this introduction), you may consider *MARSplines* as a generalization of regression trees, where the "hard" binary splits are replaced by "smooth" basis functions. Refer to Hastie, Tibshirani, and Friedman (2001) for additional details.

Technical Notes: The MARSplines Algorithm

Implementing *MARSplines* involves a two step procedure that is applied successively until a desired model is found. In the first step, we build the model, i.e. increase its complexity by adding basis functions until a preset (user-defined) maximum level of complexity has been reached. Then we begin a backward procedure to remove the least significant basis functions from the model, i.e. those whose removal will lead to the least reduction in the (least-squares) goodness of fit. This algorithm is implemented as follows:

1. Start with the simplest model involving only the constant basis function.
2. Search the space of basis functions, for each variable and for all possible knots, and add those which maximize a certain measure of goodness of fit (minimize prediction error).
3. Step 2 is recursively applied until a model of pre-determined maximum complexity is derived.
4. Finally, in the last stage, a pruning procedure is applied where those basis functions are removed that contribute least to the overall (least squares) goodness of fit.

Technical Notes: The Multivariate Adaptive Regression Splines (MARSplines) Model

The *MARSplines* algorithm builds models from two sided truncated functions of the predictors (x) of the form:

$$(x-t)_+ = \begin{Bmatrix} x-t & x>t \\ 0 & \text{otherwise} \end{Bmatrix}$$

These serve as basis functions for linear or nonlinear expansion that approximates some true underlying function *f(x)*.

The *MARSplines* model for a dependent (outcome) variable *y*, and *M* terms , can be summarized in the following equation:

$$y = f(x) = \beta_o + \sum_{m=1}^{M} \beta_m H_{km}\left(x_{v(k,m)}\right)$$

where the summation is over the *M* terms in the model, and $\square_o$ and $\square_m$ are parameters of the model (along with the knots *t* for each basis function, which are also estimated from the data). Function *H* is defined as:

$$H_{km}\left(x_{v(k,m)}\right) = \prod_{k=1}^{K} h_{km}$$

where *xv(k,m)* is the predictor in the k'th of the m'th product. For order of interactions *K=1*, the model is additive and for *K=2* the model pairwise interactive.

During forward stepwise, a number of basis functions are added to the model according to a pre-determined maximum which should be considerably larger (twice as much at least) than the optimal (best least-squares fit).

After implementing the forward stepwise selection of basis functions, a backward procedure is applied in which the model is pruned by removing those basis functions that are associated with the smallest increase in the (least squares) goodness-of-fit. A least squares error function (inverse of goodness-of-fit) is computed. The so-called Generalized Cross Validation error is a measure of the goodness of fit that takes into account not only the residual error but also the model complexity as well. It is given by

$$\mathrm{GCV} = \frac{\sum_{i=1}^{N}\left(\mathrm{y}_i - f(x_i)\right)^2}{\left(1-\frac{C}{N}\right)^2}$$

with

$$C = 1 + cd$$

where *N* is the number of cases in the data set, *d* is the effective degrees of freedom, which is equal to the number of independent basis functions. The quantity *c* is the penalty for adding a basis function. Experiments have shown that the best value for *C* can be found somewhere in the range $2 < d < 3$ (see Hastie et al., 2001).

Chapter 27
Machine Learning

- Introductory Overview
- Support Vector Machines (SVM)
- Naive Bayes
- *k*-Nearest Neighbors (KNN)

Machine Learning Introductory Overview

Machine Learning includes a number of advanced statistical methods for handling regression and classification tasks with multiple dependent and independent variables. These methods include Support Vector Machines (SVM) for regression and classification, Naive Bayes for classification, and *k*-Nearest Neighbours (KNN) for regression and classification. Detailed discussions of these techniques can be found in Hastie, Tibshirani, & Freedman (2001); a specialized comprehensive introduction to support vector machines can also be found in Cristianini and Shawe-Taylor (2000).

Support Vector Machines (SVM)

This method performs regression and classification tasks by constructing nonlinear decision boundaries. Because of the nature of the feature space in which these boundaries are found, Support Vector Machines can exhibit a large degree of flexibility in handling classification and regression tasks of varied complexities. There are several types of Support Vector models including linear, polynomial, RBF, and sigmoid.

Naive Bayes

This is a well established Bayesian method primarily formulated for performing classification tasks. Given its simplicity, i.e., the assumption that the independent variables are statistically independent, Naive Bayes models are effective classification tools that are easy to use and interpret. Naive Bayes is particularly appropriate when the dimensionality of the independent space (i.e., number of input variables) is high (a problem known as the curse of dimensionality). For the reasons given above, Naive Bayes can often outperform other more sophisticated classification methods. A variety of methods exist for modeling the conditional distributions of the inputs including normal, lognormal, gamma, and Poisson.

k-Nearest Neighbors

k-Nearest Neighbors is a memory-based method that, in contrast to other statistical methods, requires no training (i.e., no model to fit). It falls into the category of Prototype Methods. It functions on the intuitive idea that close objects are more likely to be in the same category. Thus, in KNN, predictions are based on a set of prototype examples that are used to predict new (i.e., unseen) data based on the majority vote (for classification tasks) and averaging (for regression) over a set of *k*-nearest prototypes (hence the name *k*-nearest neighbors).

Chapter 28
Multidimensional Scaling

- General Purpose
- Logic of MDS
- Computational Approach
- How many dimensions to specify?
- Interpreting the Dimensions
- Applications
- MDS and Factor Analysis

General Purpose

Multidimensional scaling (*MDS*) can be considered to be an alternative to factor analysis (see *Factor Analysis*). In general, the goal of the analysis is to detect meaningful underlying dimensions that allow the researcher to explain observed similarities or dissimilarities (distances) between the investigated objects. In factor analysis, the similarities between objects (e.g., variables) are expressed in the correlation matrix. With MDS one may analyze any kind of similarity or dissimilarity matrix, in addition to correlation matrices.

Logic of MDS

The following simple example may demonstrate the logic of an MDS analysis. Suppose we take a matrix of distances between major US cities from a map. We then analyze this matrix, specifying that we want to reproduce the distances based on two dimensions. As a result of the MDS analysis, we would most likely obtain a two-dimensional representation of the locations of the cities, that is, we would basically obtain a two-dimensional map.

In general then, MDS attempts to arrange "objects" (major cities in this example) in a space with a particular number of dimensions (two-dimensional in this example) so as to reproduce the observed distances. As a result, we can "explain" the distances in terms of underlying dimensions; in our example, we could explain the distances in terms of the two geographical dimensions: north/south and east/west.

Orientation of axes. As in factor analysis, the actual orientation of axes in the final solution is arbitrary. To return to our example, we could rotate the map in any way we want, the distances between cities remain the same. Thus, the final orientation of axes in the plane or space is mostly the result of a subjective decision by the researcher, who will choose an orientation that can be most easily explained. To return to our example, we could have chosen an orientation of axes other than north/south and east/west; however, that orientation is most convenient because it "makes the most sense" (i.e., it is easily interpretable).

Computational Approach

MDS is not so much an exact procedure as rather a way to "rearrange" objects in an efficient manner, so as to arrive at a configuration that best approximates the observed distances. It actually moves objects around in the space defined by the requested number of dimensions, and checks how well the distances between objects can be reproduced by the new configuration. In more technical terms, it uses a function minimization algorithm that evaluates different configurations with the goal of maximizing the goodness-of-fit (or minimizing "lack of fit").

Measures of goodness-of-fit: Stress. The most common measure that is used to evaluate how well (or poorly) a particular configuration reproduces the observed distance matrix is the stress measure. The raw stress value *Phi* of a configuration is defined by:

$$\text{Phi} = \Sigma[d_{ij} - f(\delta_{ij})]^2$$

In this formula, d_{ij} stands for the reproduced distances, given the respective number of dimensions, and δ_{ij} ($delta_{ij}$) stands for the input data (i.e., observed distances). The expression $f(\delta_{ij})$ indicates a *nonmetric*, monotone transformation of the observed input data (distances). Thus, it will attempt to reproduce the general rank-ordering of distances between the objects in the analysis.

There are several similar related measures that are commonly used; however, most of them amount to the computation of the sum of squared deviations of observed distances (or some monotone transformation of those distances) from the reproduced distances. Thus, the smaller the stress value, the better is the fit of the reproduced distance matrix to the observed distance matrix.

Shepard diagram. One can plot the reproduced distances for a particular number of dimensions against the observed input data (distances). This scatterplot is referred to as a *Shepard* diagram. This plot shows the reproduced distances plotted on the vertical (*Y*) axis versus the original similarities plotted on the horizontal (*X*) axis (hence, the generally negative slope). This plot also shows a step-function. This line represents the so- called *D-hat* values, that is, the result of the monotone transformation $f(\delta_{ij})$ of the input data. If all reproduced distances fall onto the step-line, then the rank-ordering of distances (or similarities) would be perfectly reproduced by the respective solution (dimensional model). Deviations from the step-line indicate lack of fit.

How Many Dimensions to Specify?

If you are familiar with factor analysis, you will be quite aware of this issue. If you are not familiar with factor analysis, you may want to read the *Factor Analysis* section in the manual; however, this is not necessary in order to understand the following discussion. In general, the more dimensions we use in order to reproduce the distance matrix, the better is the fit of the reproduced matrix to the observed matrix (i.e., the smaller is the stress). In fact, if we use as many dimensions as there are variables, then we can perfectly reproduce the observed distance matrix. Of course, our goal is to *reduce* the observed complexity of nature, that is, to explain the distance matrix in terms of fewer underlying dimensions. To return to the example of distances between cities, once we have a two-dimensional map it is much easier to visualize the location of and navigate between cities, as compared to relying on the distance matrix only.

Sources of misfit. Let us consider for a moment why fewer factors may produce a worse representation of a distance matrix than would more factors. Imagine the three cities *A*, *B*, and *C*, and the three cities *D*, *E*, and *F*; shown below are their distances from each other.

	A	**B**	**C**
A	0		
B	90	0	
C	90	90	0

	D	**E**	**F**
D	0		
E	90	0	
F	180	90	0

In the first matrix, all cities are exactly 90 miles apart from each other; in the second matrix, cities ***D*** and ***F*** are 180 miles apart. Now, can we arrange the three cities (objects) on one dimension (line)? Indeed, we can arrange cities ***D***, ***E***, and ***F*** on one dimension:

D---90 miles---**E**---90 miles---**F**

D is 90 miles away from ***E***, and ***E*** is 90 miles away form ***F***; thus, ***D*** is 90+90=180 miles away from ***F***. If you try to do the same thing with cities ***A***, ***B***, and ***C*** you will see that there is no way to arrange the three cities on one line so that the distances can be reproduced. However, we can arrange those cities in two dimensions, in the shape of a triangle:

A

90 miles 90 miles

B 90 miles **C**

Arranging the three cities in this manner, we can perfectly reproduce the distances between them. Without going into much detail, this small example illustrates how a particular distance matrix implies a particular number of dimensions. Of course, "real" data are never this "clean," and contain a lot of noise, that is, random variability that contributes to the differences between the reproduced and observed matrix.

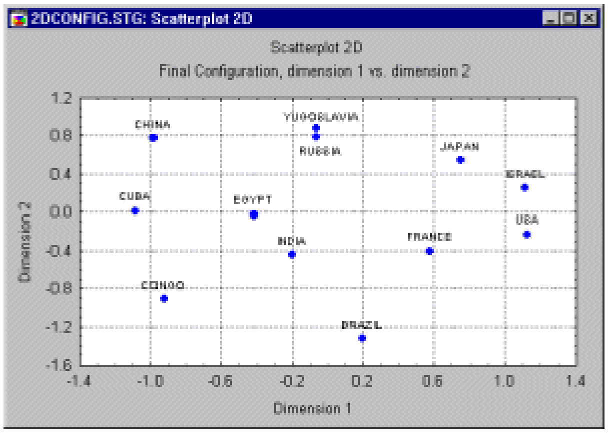
Scree test. A common way to decide how many dimensions to use is to plot the stress value against different numbers of dimensions. This test was first proposed by Cattell (1966) in the context of the number-of-factors problem in factor analysis (see *Factor Analysis*); Kruskal and Wish (1978; pp. 53-60) discuss the application of this plot to MDS.

Cattell suggests to find the place where the smooth decrease of stress values (eigenvalues in factor analysis) appears to level off to the right of the plot. To the right of this point one finds, presumably, only "factorial scree" -- "scree" is the geological term referring to the debris which collects on the lower part of a rocky slope.

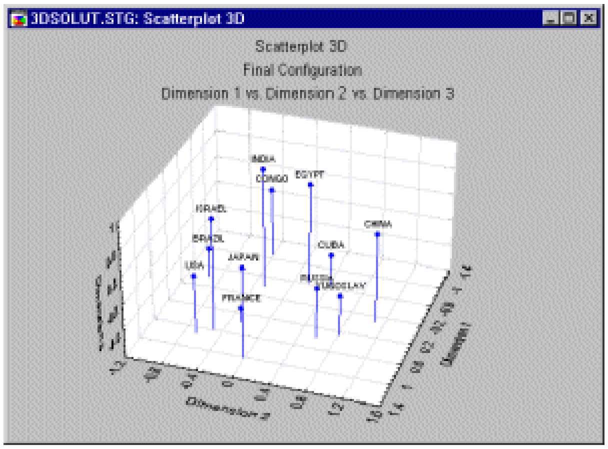
Interpretability of configuration. A second criterion for deciding how many dimensions to interpret is the clarity of the final configuration. Sometimes, as in our example of distances between cities, the resultant dimensions are easily interpreted. At other times, the points in the plot form a sort of "random cloud," and there is no straightforward and easy way to interpret the dimensions. In the latter case one should try to include more or fewer dimensions and examine the resultant final configurations. Often, more interpretable solutions emerge. However, if the data points in the plot do not follow any pattern, and if the stress plot does not show any clear "elbow," then the data are most likely random "noise."

Interpreting the Dimensions

The interpretation of dimensions usually represents the final step of the analysis. As mentioned earlier, the actual orientations of the axes from the MDS analysis are arbitrary, and can be rotated in any direction. A first step is to produce scatterplots of the objects in the different two-dimensional planes.



Three-dimensional solutions can also be illustrated graphically, however, their interpretation is somewhat more complex.



In addition to "meaningful dimensions," one should also look for clusters of points or particular patterns and configurations (such as circles, manifolds, etc.). For a detailed discussion of how to interpret final configurations, see Borg and Lingoes (1987), Borg and Shye (in press), or Guttman (1968).

Use of multiple regression techniques. An analytical way of interpreting dimensions (described in Kruskal & Wish, 1978) is to use multiple regression techniques to regress some meaningful variables on the coordinates for the different dimensions. Note that this can easily be done via Multiple Regression.

Applications

The "beauty" of MDS is that we can analyze any kind of distance or similarity matrix. These similarities can represent people's ratings of similarities between objects, the percent agreement between judges, the number of times a subjects fails to discriminate between stimuli, etc. For example, MDS methods used to be very popular in psychological research on person perception where similarities between trait descriptors were analyzed to uncover the underlying dimensionality of people's perceptions of traits (see, for example Rosenberg, 1977). They are also very popular in marketing research, in order to detect the number and nature of dimensions underlying the perceptions of different brands or products & Carmone, 1970).

In general, MDS methods allow the researcher to ask relatively unobtrusive questions ("how similar is brand A to brand B") and to derive from those questions underlying dimensions without the respondents ever knowing what is the researcher's real interest.

MDS and Factor Analysis

Even though there are similarities in the type of research questions to which these two procedures can be applied, MDS and factor analysis are fundamentally different methods. Factor analysis requires that the underlying data are distributed as multivariate normal, and that the relationships are linear. MDS imposes no such restrictions. As long as the rank-ordering of distances (or similarities) in the matrix is meaningful, MDS can be used. In terms of resultant differences, factor analysis tends to extract more factors (dimensions) than MDS; as a result, MDS often yields more readily, interpretable solutions. Most importantly, however, MDS can be applied to any kind of distances or similarities, while factor analysis requires us to first compute a correlation matrix. MDS can be based on subjects' direct assessment of similarities between stimuli, while factor analysis requires subjects to rate those stimuli on some list of attributes (for which the factor analysis is performed).

In summary, MDS methods are applicable to a wide variety of research designs because distance measures can be obtained in any number of ways (for different examples, refer to the references provided at the beginning of this section).

Chapter 29
Neural Networks

- Preface
- Applications for Neural Networks
- The Biological Inspiration
- The Basic Artificial Model
- Using a Neural Network
- Gathering Data for Neural Networks
    - Summary
- Pre- and Post-processing
- Multilayer Perceptrons
    - Training Multilayer Perceptrons
    - The Back Propagation Algorithm
    - Over-learning and Generalization
    - Data Selection
    - Insights into MLP Training
    - Other MLP Training Algorithms
- Radial Basis Function Networks
- Probabilistic Neural Networks
- Generalized Regression Neural Networks
- Linear Networks
- SOFM Networks
- Classification in Neural Networks
    - Classification Statistics
- Regression Problems in Neural Networks
- Time Series Prediction in Neural Networks
- Variable Selection and Dimensionality Reduction
- Ensembles and Resampling
- Recommended Textbooks

Many concepts related to the neural networks methodology are best explained if they are illustrated with applications of a specific neural network program. Therefore, this chapter contains many references to *STATISTICA Neural Networks* (in short, *ST Neural Networks*, a neural networks application available from StatSoft), a particularly comprehensive neural network tool.

Preface

Neural networks have seen an explosion of interest over the last few years, and are being successfully applied across an extraordinary range of problem domains, in areas as diverse as finance, medicine, engineering, geology and physics. Indeed, anywhere that there are problems of prediction, classification or control, neural networks are being introduced. This sweeping success can be attributed to a few key factors:

- **Power.** Neural networks are very sophisticated modeling techniques capable of modeling extremely complex functions. In particular, neural networks are *nonlinear* (a term which is discussed in more detail later in this section). For many years linear modeling has been the commonly used technique in most modeling domains since linear models have well-known optimization strategies. Where the linear approximation was not valid (which was frequently the case) the models suffered accordingly. Neural networks also keep in check the *curse of dimensionality* problem that bedevils attempts to model nonlinear functions with large numbers of variables.
- **Ease of use.** Neural networks *learn by example*. The neural network user gathers representative data, and then invokes *training algorithms* to automatically learn the structure of the data. Although the user does need to have some heuristic knowledge of how to select and prepare data, how to select an appropriate neural network, and how to interpret the results, the level of user knowledge needed to successfully apply neural networks is much lower than would be the case using (for example) some more traditional nonlinear statistical methods.

Neural networks are also intuitively appealing, based as they are on a crude low-level model of biological neural systems. In the future, the development of this neurobiological modeling may lead to genuinely intelligent computers.

Applications for Neural Networks

Neural networks are applicable in virtually every situation in which a relationship between the predictor variables (independents, inputs) and predicted variables (dependents, outputs) exists, even when that relationship is very complex and not easy to articulate in the usual terms of "correlations" or "differences between groups." A few representative examples of problems to which neural network analysis has been applied successfully are:

- **Detection of medical phenomena.** A variety of health-related indices (e.g., a combination of heart rate, levels of various substances in the blood, respiration rate) can be monitored. The onset of a particular medical condition could be associated with a very complex (e.g., nonlinear and interactive) combination of changes on a subset of the variables being monitored. Neural networks have been used to recognize this predictive pattern so that the appropriate treatment can be prescribed.
- **Stock market prediction.** Fluctuations of stock prices and stock indices are another example of a complex, multidimensional, but in some circumstances at least partially-deterministic phenomenon. Neural networks are being used by many technical analysts to make predictions about stock prices based upon a large number of factors such as past performance of other stocks and various economic indicators.

- **Credit assignment.** A variety of pieces of information are usually known about an applicant for a loan. For instance, the applicant's age, education, occupation, and many other facts may be available. After training a neural network on historical data, neural network analysis can identify the most relevant characteristics and use those to classify applicants as good or bad credit risks.
- **Monitoring the condition of machinery.** Neural networks can be instrumental in cutting costs by bringing additional expertise to scheduling the preventive maintenance of machines. A neural network can be trained to distinguish between the sounds a machine makes when it is running normally ("false alarms") versus when it is on the verge of a problem. After this training period, the expertise of the network can be used to warn a technician of an upcoming breakdown, before it occurs and causes costly unforeseen "downtime."
- **Engine management.** Neural networks have been used to analyze the input of sensors from an engine. The neural network controls the various parameters within which the engine functions, in order to achieve a particular goal, such as minimizing fuel consumption.

The Biological Inspiration

Neural networks grew out of research in Artificial Intelligence; specifically, attempts to mimic the fault-tolerance and capacity to learn of biological neural systems by modeling the low-level structure of the brain (see Patterson, 1996). The main branch of Artificial Intelligence research in the 1960s -1980s produced Expert Systems. These are based upon a high-level model of reasoning processes (specifically, the concept that our reasoning processes are built upon manipulation of symbols). It became rapidly apparent that these systems, although very useful in some domains, failed to capture certain key aspects of human intelligence. According to one line of speculation, this was due to their failure to mimic the underlying structure of the brain. In order to reproduce intelligence, it would be necessary to build systems with a similar architecture.

The brain is principally composed of a very large number (circa 10,000,000,000) of *neurons*, massively interconnected (with an average of several thousand interconnects per neuron, although this varies enormously). Each neuron is a specialized cell which can propagate an electrochemical signal. The neuron has a branching input structure (the dendrites), a cell body, and a branching output structure (the axon). The axons of one cell connect to the dendrites of another via a synapse. When a neuron is activated, it *fires* an electrochemical signal along the axon. This signal crosses the synapses to other neurons, which may in turn fire. A neuron fires only if the total signal received at the cell body from the dendrites exceeds a certain level (the firing threshold).

The strength of the signal received by a neuron (and therefore its chances of firing) critically depends on the efficacy of the synapses. Each synapse actually contains a gap, with neurotransmitter chemicals poised to transmit a signal across the gap. One of the most influential researchers into neurological systems (Donald Hebb) postulated that learning consisted principally in altering the "strength" of synaptic connections. For example, in the classic Pavlovian conditioning experiment, where a bell is rung just before dinner is delivered to a dog, the dog rapidly learns to associate the ringing of a bell with the eating of food. The synaptic connections between the appropriate part of the auditory cortex and the salivation glands are strengthened, so that when the auditory cortex is stimulated by the sound of the bell the dog starts to salivate. Recent research in cognitive science, in particular in the area of nonconscious information processing, have further demonstrated the enormous capacity of the human mind to infer ("learn") simple input-output covariations from extremely complex stimuli (e.g., see Lewicki, Hill, and Czyzewska, 1992).

Thus, from a very large number of extremely simple processing units (each performing a weighted sum of its inputs, and then firing a binary signal if the total input exceeds a certain level) the brain manages to perform extremely complex tasks. Of course, there is a great deal of complexity in the brain which has not been discussed here, but it is interesting that artificial neural networks can achieve some remarkable results using a model not much more complex than this.

The Basic Artificial Model

To capture the essence of biological neural systems, an artificial *neuron* is defined as follows:

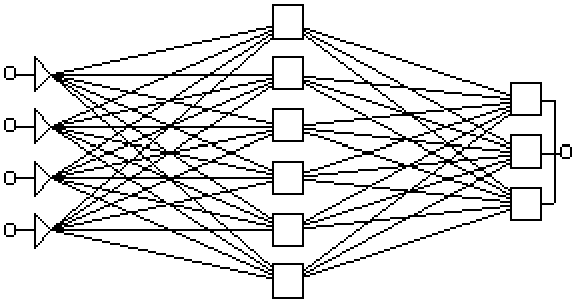
- It receives a number of inputs (either from original data, or from the output of other neurons in the neural network). Each input comes via a connection that has a strength (or *weight*); these weights correspond to synaptic efficacy in a biological neuron. Each neuron also has a single threshold value. The weighted sum of the inputs is formed, and the threshold subtracted, to compose the *activation* of the neuron (also known as the post-synaptic potential, or PSP, of the neuron).
- The activation signal is passed through an activation function (also known as a transfer function) to produce the output of the neuron.

If the step activation function is used (i.e., the neuron's output is 0 if the input is less than zero, and 1 if the input is greater than or equal to 0) then the neuron acts just like the biological neuron described earlier (subtracting the threshold from the weighted sum and comparing with zero is equivalent to comparing the weighted sum to the threshold). Actually, the step function is rarely used in artificial neural networks, as will be discussed. Note also that weights can be negative, which implies that the synapse has an inhibitory rather than excitatory effect on the neuron: inhibitory neurons are found in the brain.

This describes an individual neuron. The next question is: how should neurons be connected together? If a network is to be of any use, there must be inputs (which carry the values of variables of interest in the outside world) and outputs (which form predictions, or control signals). Inputs and outputs correspond to sensory and motor nerves such as those coming from the eyes and leading to the hands. However, there also can be hidden neurons that play an internal role in the network. The input, hidden and output neurons need to be connected together.

The key issue here is *feedback* (Haykin, 1994). A simple network has a *feedforward* structure: signals flow from inputs, forwards through any hidden units, eventually reaching the output units. Such a structure has stable behavior. However, if the network is *recurrent* (contains connections back from later to earlier neurons) it can be unstable, and has very complex dynamics. Recurrent networks are very interesting to researchers in neural networks, but so far it is the feedforward structures that have proved most useful in solving real problems.

A typical feedforward network has neurons arranged in a distinct layered topology. The input layer is not really neural at all: these units simply serve to introduce the values of the input variables. The hidden and output layer neurons are each connected to all of the units in the preceding layer. Again, it is possible to define networks that are partially-connected to only some units in the preceding layer; however, for most applications fully-connected networks are better.



When the network is executed (used), the input variable values are placed in the input units, and then the hidden and output layer units are progressively executed. Each of them calculates its activation value by taking the weighted sum of the outputs of the units in the preceding layer, and subtracting the threshold. The activation value is passed through the activation function to produce the output of the neuron. When the entire network has been executed, the outputs of the output layer act as the output of the entire network.

Using a Neural Network

The previous section describes in simplified terms how a neural network turns inputs into outputs. The next important question is: how do you apply a neural network to solve a problem?

The type of problem amenable to solution by a neural network is defined by the way they *work* and the way they are *trained*. Neural networks work by feeding in some input variables, and producing some output variables. They can therefore be used where you have some known information, and would like to infer some unknown information (see Patterson, 1996; Fausett, 1994). Some examples are:

Stock market prediction. You know last week's stock prices and today's DOW, NASDAQ, or FTSE index; you want to know tomorrow's stock prices.

Credit assignment. You want to know whether an applicant for a loan is a good or bad credit risk. You usually know applicants' income, previous credit history, etc. (because you ask them these things).

Control. You want to know whether a robot should turn left, turn right, or move forwards in order to reach a target; you know the scene that the robot's camera is currently observing.

Needless to say, not every problem can be solved by a neural network. You may wish to know next week's lottery result, and know your shoe size, but there is no relationship between the two. Indeed, if the lottery is being run correctly, there is no fact you could possibly know that would allow you to infer next week's result. Many financial institutions use, or have experimented with, neural networks for stock market prediction, so it is likely that any trends predictable by neural techniques are already discounted by the market, and (unfortunately), unless you have a sophisticated understanding of that problem domain, you are unlikely to have any success there either!

Therefore, another important requirement for the use of a neural network therefore is that you know (or at least strongly suspect) that there is a relationship between the proposed known inputs and unknown outputs. This relationship may be noisy (you certainly would not expect that the factors given in the stock market prediction example above could give an exact prediction, as prices are clearly influenced by other factors not represented in the input set, and there may be an element of pure randomness) but it must exist.

In general, if you use a neural network, you won't know the exact nature of the relationship between inputs and outputs - if you knew the relationship, you would model it directly. The other key feature of neural networks is that they learn the input/output relationship through training. There are two types of training used in neural networks, with different types of networks using different types of training. These are supervised and unsupervised training, of which supervised is the most common and will be discussed in this section (unsupervised learning is described in a later section).

In supervised learning, the network user assembles a set of *training data*. The training data contains examples of inputs together with the corresponding outputs, and the network learns to infer the relationship between the two. Training data is usually taken from historical records. In the above examples, this might include previous stock prices and DOW, NASDAQ, or FTSE indices, records of previous successful loan applicants, including questionnaires and a record of whether they defaulted or not, or sample robot positions and the correct reaction.

The neural network is then trained using one of the supervised learning algorithms (of which the best known example is *back propagation*, devised by Rumelhart et. al., 1986), which uses the data to adjust the network's weights and thresholds so as to minimize the error in its predictions on the training set. If the network is properly trained, it has then learned to model the (unknown) function that relates the input variables to the output variables, and can subsequently be used to make predictions where the output is *not* known.

Gathering Data for Neural Networks

Once you have decided on a problem to solve using neural networks, you will need to gather data for training purposes. The training data set includes a number of *cases*, each containing values for a range of input and output *variables*. The first decisions you will need to make are: which variables to use, and how many (and which) cases to gather.

The choice of variables (at least initially) is guided by intuition. Your own expertise in the problem domain will give you some idea of which input variables are likely to be influential. As a first pass, you should include any variables that you think could have an influence - part of the design process will be to whittle this set down.

Neural networks process numeric data in a fairly limited range. This presents a problem if data is in an unusual range, if there is missing data, or if data is non-numeric. Fortunately, there are methods to deal with each of these problems. Numeric data is scaled into an appropriate range for the network, and missing values can be substituted for using the mean value (or other statistic) of that variable across the other available training cases (see Bishop, 1995).

Handling non-numeric data is more difficult. The most common form of non-numeric data consists of nominal-value variables such as *Gender*={*Male*, *Female*}. Nominal-valued variables can be represented numerically. However, neural networks do not tend to perform well with nominal variables that have a large number of possible values.

For example, consider a neural network being trained to estimate the value of houses. The price of houses depends critically on the area of a city in which they are located. A particular city might be subdivided into dozens of named locations, and so it might seem natural to use a nominal-valued variable representing these locations. Unfortunately, it would be very difficult to train a neural network under these circumstances, and a more credible approach would be to assign ratings (based on expert knowledge) to each area; for example, you might assign ratings for the quality of local schools, convenient access to leisure facilities, etc.

Other kinds of non-numeric data must either be converted to numeric form, or discarded. Dates and times, if important, can be converted to an offset value from a starting date/time. Currency values can easily be converted. Unconstrained text fields (such as names) cannot be handled and should be discarded.

The number of cases required for neural network training frequently presents difficulties. There are some heuristic guidelines, which relate the number of cases needed to the size of the network (the simplest of these says that there should be ten times as many cases as connections in the network). Actually, the number needed is also related to the (unknown) complexity of the underlying function which the network is trying to model, and to the variance of the additive noise. As the number of variables increases, the number of cases required increases nonlinearly, so that with even a fairly small number of variables (perhaps fifty or less) a huge number of cases are required. This problem is known as "the curse of dimensionality," and is discussed further later in this chapter.

For most practical problem domains, the number of cases required will be hundreds or thousands. For very complex problems more may be required, but it would be a rare (even trivial) problem which required less than a hundred cases. If your data is sparser than this, you really don't have enough information to train a network, and the best you can do is probably to fit a linear model. If you have a larger, but still restricted, data set, you can compensate to some extent by forming an ensemble of networks, each trained using a different resampling of the available data, and then average across the predictions of the networks in the ensemble.

Many practical problems suffer from data that is unreliable: some variables may be corrupted by noise, or values may be missing altogether. Neural networks are also noise tolerant. However, there is a limit to this tolerance; if there are occasional outliers far outside the range of normal values for a variable, they may bias the training. The best approach to such outliers is to identify and remove them (either discarding the case, or converting the outlier into a missing value). If outliers are difficult to detect, a city block error function (see Bishop, 1995) may be used, but this outlier-tolerant training is generally less effective than the standard approach.

Summary

Choose variables that you believe may be influential

Numeric and nominal variables can be handled. Convert other variables to one of these forms, or discard.

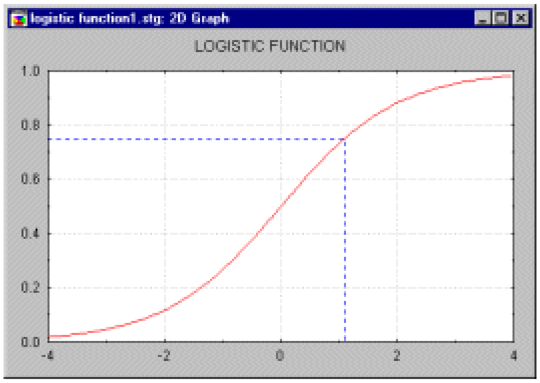
Hundreds or thousands of cases are required; the more variables, the more cases.

Cases with missing values can be used, if necessary, but outliers may cause problems - check your data. Remove outliers if possible. If you have sufficient data, discard cases with missing values.

If the volume of the data available is small, consider using ensembles and resampling.

Pre- and Post-processing

All neural networks take numeric input and produce numeric output. The transfer function of a unit is typically chosen so that it can accept *input in any* range, and produces *output in a strictly limited* range (it has a squashing effect). Although the input can be in any range, there is a saturation effect so that the unit is only sensitive to inputs within a fairly limited range. The illustration below shows one of the most common transfer functions, the logistic function (also sometimes referred to as the sigmoid function, although strictly speaking it is only one example of a sigmoid - S-shaped - function). In this case, the output is in the range (0,1), and the input is sensitive in a range not much larger than (-1,+1). The function is also smooth and easily differentiable, facts that are critical in allowing the network training algorithms to operate (this is the reason why the step function is not used in practice).



The limited numeric response range, together with the fact that information has to be in numeric form, implies that neural solutions require preprocessing and post-processing stages to be used in real applications (see Bishop, 1995). Two issues need to be addressed:

Scaling. Numeric values have to be scaled into a range that is appropriate for the network. Typically, raw variable values are scaled linearly. In some circumstances, non-linear scaling may be appropriate (for example, if you know that a variable is exponentially distributed, you might take the logarithm). Non-linear scaling is not supported in *ST Neural Networks*. Instead, you should scale the variable using *STATISTICA*'s data transformation facilities before transferring the data to *ST Neural Networks*.

Nominal variables. Nominal variables may be two-state (e.g., *Gender*={*Male*,*Female*}) or many-state (i.e., more than two states). A two-state nominal variable is easily represented by transformation into a numeric value (e.g., *Male=0*, *Female=1*). Many-state nominal variables are more difficult to handle. They can be represented using an ordinal encoding (e.g., *Dog=0,Budgie=1,Cat=2*) but this implies a (probably) false ordering on the nominal values - in this case, that *Budgies* are in some sense midway between *Dogs* and *Cats*. A better approach, known as *one-of-N* encoding, is to use a number of numeric variables to represent the single nominal variable. The number of numeric variables equals the number of possible values; one of the *N* variables is set, and the others cleared (e.g., *Dog*={*1,0,0*}, *Budgie*={*0,1,0*}, *Cat*={*0,0,1*}). *ST Neural Networks* has facilities to convert both two-state and many-state nominal variables for use in the neural network. Unfortunately, a nominal variable with a large number of states would require a prohibitive number of numeric variables for *one-of-N* encoding, driving up the network size and making training difficult. In such a case it is possible (although unsatisfactory) to model the nominal variable using a single numeric ordinal; a better approach is to look for a different way to represent the information.

Prediction problems may be divided into two main categories:

Classification. In classification, the objective is to determine to which of a number of discrete classes a given input case belongs. Examples include credit assignment (is this person a good or bad credit risk), cancer detection (tumor, clear), signature recognition (forgery, true). In all these cases, the output required is clearly a single nominal variable. The most common classification tasks are (as above) two-state, although many-state tasks are also not unknown.

Regression. In regression, the objective is to predict the value of a (usually) continuous variable: tomorrow's stock price, the fuel consumption of a car, next year's profits. In this case, the output required is a single numeric variable.

Neural networks can actually perform a number of regression and/or classification tasks at once, although commonly each network performs only one. In the vast majority of cases, therefore, the network will have a single output variable, although in the case of many-state classification problems, this may correspond to a number of output units (the post-processing stage takes care of the mapping from output units to output variables). If you do define a single network with multiple output variables, it may suffer from cross-talk (the hidden neurons experience difficulty learning, as they are attempting to model at least two functions at once). The best solution is usually to train separate networks for each output, then to combine them into an ensemble so that they can be run as a unit.

Multilayer Perceptrons

This is perhaps the most popular network architecture in use today, due originally to Rumelhart and McClelland (1986) and discussed at length in most neural network textbooks (e.g., Bishop, 1995). This is the type of network discussed briefly in previous sections: the units each perform a biased weighted sum of their inputs and pass this activation level through a transfer function to produce their output, and the units are arranged in a layered feedforward topology. The network thus has a simple interpretation as a form of input-output model, with the weights and thresholds (biases) the free parameters of the model. Such networks can model functions of almost arbitrary complexity, with the number of layers, and the number of units in each layer, determining the function complexity. Important issues in Multilayer Perceptrons (MLP) design include specification of the number of hidden layers and the number of units in these layers (see Haykin, 1994; Bishop, 1995).

The number of input and output units is defined by the problem (there may be some uncertainty about precisely which inputs to use, a point to which we will return later. However, for the moment we will assume that the input variables are intuitively selected and are all meaningful). The number of hidden units to use is far from clear. As good a starting point as any is to use one hidden layer, with the number of units equal to half the sum of the number of input and output units. Again, we will discuss how to choose a sensible number later.

Training Multilayer Perceptrons

Once the number of layers, and number of units in each layer, has been selected, the network's weights and thresholds must be set so as to minimize the prediction error made by the network. This is the role of the *training algorithms*. The historical cases that you have gathered are used to automatically adjust the weights and thresholds in order to minimize this error. This process is equivalent to fitting the model represented by the network to the training data available. The error of a particular configuration of the network can be determined by running all the training cases through the network, comparing the actual output generated with the desired or target outputs. The differences are combined together by an *error function* to give the network error. The most common error functions are the *sum squared error* (used for regression problems), where the individual errors of output units on each case are squared and summed together, and the cross entropy functions (used for maximum likelihood classification).

In traditional modeling approaches (e.g., linear modeling) it is possible to algorithmically determine the model configuration that absolutely minimizes this error. The price paid for the greater (non-linear) modeling power of neural networks is that although we can adjust a network to lower its error, we can never be sure that the error could not be lower still.

A helpful concept here is the error surface. Each of the *N* weights and thresholds of the network (i.e., the free parameters of the model) is taken to be a dimension in space. The *N+1*th dimension is the network error. For any possible configuration of weights the error can be plotted in the *N+1*th dimension, forming an *error surface*. The objective of network training is to find the lowest point in this many-dimensional surface.

In a linear model with *sum squared error* function, this error surface is a parabola (a quadratic), which means that it is a smooth bowl-shape with a single minimum. It is therefore "easy" to locate the minimum.

Neural network error surfaces are much more complex, and are characterized by a number of unhelpful features, such as local minima (which are lower than the surrounding terrain, but above the global minimum), flat-spots and plateaus, saddle-points, and long narrow ravines.

It is not possible to analytically determine where the global minimum of the error surface is, and so neural network training is essentially an exploration of the error surface. From an initially random configuration of weights and thresholds (i.e., a random point on the error surface), the training algorithms incrementally seek for the global minimum. Typically, the gradient (slope) of the error surface is calculated at the current point, and used to make a downhill move. Eventually, the algorithm stops in a low point, which may be a local minimum (but hopefully is the global minimum).

The Back Propagation Algorithm

The best-known example of a neural network training algorithm is *back propagation* (see Patterson, 1996; Haykin, 1994; Fausett, 1994). Modern second-order algorithms such as *conjugate gradient descent* and *Levenberg-Marquardt* (see Bishop, 1995; Shepherd, 1997) (both included in *ST Neural Networks*) are substantially faster (e.g., an order of magnitude faster) for many problems, but *back propagation* still has advantages in some circumstances, and is the easiest algorithm to understand. We will introduce this now, and discuss the more advanced algorithms later. There are also heuristic modifications of *back propagation* which work well for some problem domains, such as *quick propagation* (Fahlman, 1988) and *Delta-Bar-Delta* (Jacobs, 1988) and are also included in *ST Neural Networks*.

In *back propagation*, the gradient vector of the error surface is calculated. This vector points along the line of steepest descent from the current point, so we know that if we move along it a "short" distance, we will decrease the error. A sequence of such moves (slowing as we near the bottom) will eventually find a minimum of some sort. The difficult part is to decide how large the steps should be.

Large steps may converge more quickly, but may also overstep the solution or (if the error surface is very eccentric) go off in the wrong direction. A classic example of this in neural network training is where the algorithm progresses very slowly along a steep, narrow, valley, bouncing from one side across to the other. In contrast, very small steps may go in the correct direction, but they also require a large number of iterations. In practice, the step size is proportional to the slope (so that the algorithms settles down in a minimum) and to a special constant: the *learning rate*. The correct setting for the learning rate is application-dependent, and is typically chosen by experiment; it may also be time-varying, getting smaller as the algorithm progresses.

The algorithm is also usually modified by inclusion of a momentum term: this encourages movement in a fixed direction, so that if several steps are taken in the same direction, the algorithm "picks up speed", which gives it the ability to (sometimes) escape local minimum, and also to move rapidly over flat spots and plateaus.

The algorithm therefore progresses iteratively, through a number of *epochs*. On each epoch, the training cases are each submitted in turn to the network, and target and actual outputs compared and the error calculated. This error, together with the error surface gradient, is used to adjust the weights, and then the process repeats. The initial network configuration is random, and training stops when a given number of epochs elapses, or when the error reaches an acceptable level, or when the error stops improving (you can select which of these stopping conditions to use).

Over-learning and Generalization

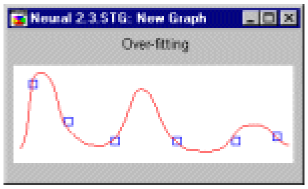
One major problem with the approach outlined above is that it doesn't actually minimize the error that we are really interested in - which is the expected error the network will make when *new* cases are submitted to it. In other words, the most desirable property of a network is its ability to *generalize* to new cases. In reality, the network is trained to minimize the error on the training set, and short of having a perfect and infinitely large training set, this is not the same thing as minimizing the error on the real error surface - the error surface of the underlying and unknown model (see Bishop, 1995).

The most important manifestation of this distinction is the problem of over-learning, or over-fitting. It is easiest to demonstrate this concept using polynomial curve fitting rather than neural networks, but the concept is precisely the same.

A polynomial is an equation with terms containing only constants and powers of the variables. For example:

$y=2x+3$
$y=3x^2+4x+1$

Different polynomials have different shapes, with larger powers (and therefore larger numbers of terms) having steadily more eccentric shapes. Given a set of data, we may want to fit a polynomial curve (i.e., a model) to explain the data. The data is probably noisy, so we don't necessarily expect the best model to pass exactly through all the points. A low-order polynomial may not be sufficiently flexible to fit close to the points, whereas a high-order polynomial is actually too flexible, fitting the data exactly by adopting a highly eccentric shape that is actually unrelated to the underlying function. See illustration below.



Neural networks have precisely the same problem. A network with more weights models a more complex function, and is therefore prone to over-fitting. A network with less weights may not be sufficiently powerful to model the underlying function. For example, a network with no hidden layers actually models a simple linear function.

How then can we select the right complexity of network? A larger network will almost invariably achieve a lower error eventually, but this may indicate over-fitting rather than good modeling.

The answer is to check progress against an independent data set, the selection set. Some of the cases are reserved, and not actually used for training in the *back propagation* algorithm. Instead, they are used to keep an independent check on the progress of the algorithm. It is invariably the case that the initial performance of the network on training and selection sets is the same (if it is not at least approximately the same, the division of cases between the two sets is probably biased). As training progresses, the training error naturally drops, and providing training is minimizing the true error function, the selection error drops too. However, if the selection error stops dropping, or indeed starts to rise, this indicates that the network is starting to overfit the data, and training should cease. When over-fitting occurs during the training process like this, it is called over-learning. In this case, it is usually advisable to decrease the number of hidden units and/or hidden layers, as the network is over-powerful for the problem at hand. In contrast, if the network is not sufficiently powerful to model the underlying function, over-learning is not likely to occur, and neither training nor selection errors will drop to a satisfactory level.

The problems associated with local minima, and decisions over the size of network to use, imply that using a neural network typically involves experimenting with a large number of different networks, probably training each one a number of times (to avoid being fooled by local minima), and observing individual performances. The key guide to performance here is the selection error. However, following the standard scientific precept that, all else being equal, a simple model is always preferable to a complex model, you can also select a smaller network in preference to a larger one with a negligible improvement in selection error.

A problem with this approach of repeated experimentation is that the selection set plays a key role in selecting the model, which means that it is actually part of the training process. Its reliability as an independent guide to performance of the model is therefore compromised - with sufficient experiments, you may just hit upon a lucky network that happens to perform well on the selection set. To add confidence in the performance of the final model, it is therefore normal practice (at least where the volume of training data allows it) to reserve a third set of cases - the test set. The final model is tested with the test set data, to ensure that the results on the selection and training set are real, and not artifacts of the training process. Of course, to fulfill this role properly the test set should be used only once - if it is in turn used to adjust and reiterate the training process, it effectively becomes selection data!

This division into multiple subsets is very unfortunate, given that we usually have less data than we would ideally desire even for a single subset. We can get around this problem by resampling. Experiments can be conducted using different divisions of the available data into training, selection, and test sets. There are a number of approaches to this subset, including random (monte-carlo) resampling, cross-validation, and bootstrap. If we make design decisions, such as the best configuration of neural network to use, based upon a number of experiments with different subset examples, the results will be much more reliable. We can then either use those experiments solely to guide the decision as to which network types to use, and train such networks from scratch with new samples (this removes any sampling bias); or, we can retain the best networks found during the sampling process, but average their results in an ensemble, which at least mitigates the sampling bias.

To summarize, network design (once the input variables have been selected) follows a number of stages:

- Select an initial configuration (typically, one hidden layer with the number of hidden units set to half the sum of the number of input and output units).
- Iteratively conduct a number of experiments with each configuration, retaining the best network (in terms of selection error) found. A number of experiments are required with each configuration to avoid being fooled if training locates a local minimum, and it is also best to resample.
- On each experiment, if under-learning occurs (the network doesn't achieve an acceptable performance level) try adding more neurons to the hidden layer(s). If this doesn't help, try adding an extra hidden layer.
- If over-learning occurs (selection error starts to rise) try removing hidden units (and possibly layers).
- Once you have experimentally determined an effective configuration for your networks, resample and generate new networks with that configuration.

Data Selection

All the above stages rely on a key assumption. Specifically, the training, verification and test data must be representative of the underlying model (and, further, the three sets must be independently representative). The old computer science adage "garbage in, garbage out" could not apply more strongly than in neural modeling. If training data is not representative, then the model's worth is at best compromised. At worst, it may be useless. It is worth spelling out the kind of problems which can corrupt a training set:

The future is not the past. Training data is typically historical. If circumstances have changed, relationships which held in the past may no longer hold.

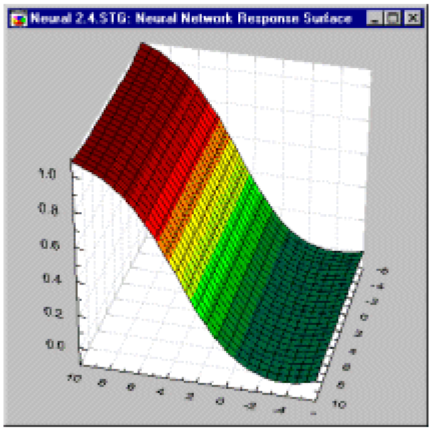
All eventualities must be covered. A neural network can only learn from cases that are present. If people with incomes over $100,000 per year are a bad credit risk, and your training data includes nobody over $40,000 per year, you cannot expect it to make a correct decision when it encounters one of the previously-unseen cases. Extrapolation is dangerous with any model, but some types of neural network may make particularly poor predictions in such circumstances.

A network learns the easiest features it can. A classic (possibly apocryphal) illustration of this is a vision project designed to automatically recognize tanks. A network is trained on a hundred pictures including tanks, and a hundred not. It achieves a perfect 100% score. When tested on new data, it proves hopeless. The reason? The pictures of tanks are taken on dark, rainy days; the pictures without on sunny days. The network learns to distinguish the (trivial matter of) differences in overall light intensity. To work, the network would need training cases including all weather and lighting conditions under which it is expected to operate - not to mention all types of terrain, angles of shot, distances...

Unbalanced data sets. Since a network minimizes an overall error, the proportion of types of data in the set is critical. A network trained on a data set with 900 good cases and 100 bad will bias its decision towards good cases, as this allows the algorithm to lower the overall error (which is much more heavily influenced by the good cases). If the representation of good and bad cases is different in the real population, the network's decisions may be wrong. A good example would be disease diagnosis. Perhaps 90% of patients routinely tested are clear of a disease. A network is trained on an available data set with a 90/10 split. It is then used in diagnosis on patients complaining of specific problems, where the likelihood of disease is 50/50. The network will react over-cautiously and fail to recognize disease in some unhealthy patients. In contrast, if trained on the "complainants" data, and then tested on "routine" data, the network may raise a high number of false positives. In such circumstances, the data set may need to be crafted to take account of the distribution of data (e.g., you could replicate the less numerous cases, or remove some of the numerous cases), or the network's decisions modified by the inclusion of a *loss matrix* (Bishop, 1995). Often, the best approach is to ensure even representation of different cases, then to interpret the network's decisions accordingly.

Insights into MLP Training

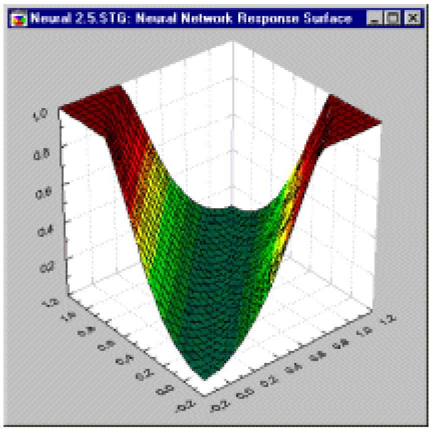
More key insights into MLP behavior and training can be gained by considering the type of functions they model. Recall that the activation level of a unit is the weighted sum of the inputs, plus a threshold value. This implies that the activation level is actually a simple linear function of the inputs. The activation is then passed through a sigmoid (S-shaped) curve. The combination of the multi-dimensional linear function and the one-dimensional sigmoid function gives the characteristic sigmoid cliff response of a first hidden layer MLP unit (the figure below illustrates the shape plotted across two inputs. An MLP unit with more inputs has a higher-dimensional version of this functional shape). Altering the weights and thresholds alters this response surface. In particular, both the orientation of the surface, and the steepness of the sloped section, can be altered. A steep slope corresponds to large weight values: doubling all weight values gives the same orientation but a different slope.



A multi-layered network combines a number of these response surfaces together, through repeated linear combination and non-linear activation functions. The next figure illustrates a typical response surface for a network with only one hidden layer, of two units, and a single output unit, on the classic XOR problem. Two separate sigmoid surfaces have been combined into a single U-shaped surface.

During network training, the weights and thresholds are first initialized to small, random values. This implies that the units' response surfaces are each aligned randomly with low slope: they are effectively uncommitted. As training progresses, the units' response surfaces are rotated and shifted into appropriate positions, and the magnitudes of the weights grow as they commit to modeling particular parts of the target response surface.

In a classification problem, an output unit's task is to output a strong signal if a case belongs to its class, and a weak signal if it doesn't. In other words, it is attempting to model a function that has magnitude one for parts of the pattern-space that contain its cases, and magnitude zero for other parts.

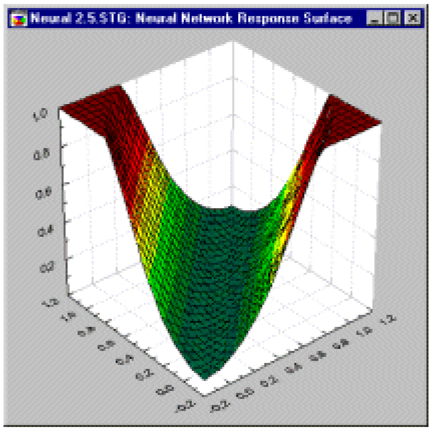


This is known as a *discriminant function* in pattern recognition problems. An ideal discriminant function could be said to have a plateau structure, where all points on the function are either at height zero or height one.

If there are no hidden units, then the output can only model a single sigmoid-cliff with areas to one side at low height and areas to the other high. There will always be a region in the middle (on the cliff) where the height is in-between, but as weight magnitudes are increased, this area shrinks.

A sigmoid-cliff like this is effectively a linear discriminant. Points to one side of the cliff are classified as belonging to the class, points to the other as not belonging to it. This implies that a network with no hidden layers can only classify linearly-separable problems (those where a line - or, more generally in higher dimensions, a hyperplane - can be drawn which separates the points in pattern space).

A network with a single hidden layer has a number of sigmoid-cliffs (one per hidden unit) represented in that hidden layer, and these are in turn combined into a plateau in the output layer. The plateau has a convex hull (i.e., there are no dents in it, and no holes inside it). Although the plateau is convex, it may extend to infinity in some directions (like an extended peninsular). Such a network is in practice capable of modeling adequately most real-world classification problems.



The figure above shows the plateau response surface developed by an MLP to solve the XOR problem: as can be seen, this neatly sections the space along a diagonal.

A network with two hidden layers has a number of plateaus combined together - the number of plateaus corresponds to the number of units in the second layer, and the number of sides on each plateau corresponds to the number of units in the first hidden layer. A little thought shows that you can represent any shape (including concavities and holes) using a sufficiently large number of such plateaus.

A consequence of these observations is that an MLP with two hidden layers is theoretically sufficient to model any problem (there is a more formal proof, the Kolmogorov Theorem). This does not necessarily imply that a network with more layers might not more conveniently or easily model a particular problem. In practice, however, most problems seem to yield to a single hidden layer, with two an occasional resort and three practically unknown.

A key question in classification is how to interpret points on or near the cliff. The standard practice is to adopt some confidence levels (the accept and reject thresholds) that must be exceeded before the unit is deemed to have made a decision. For example, if accept/reject thresholds of 0.95/0.05 are used, an output unit with an output level in excess of 0.95 is deemed to be on, below 0.05 it is deemed to be off, and in between it is deemed to be undecided.

A more subtle (and perhaps more useful) interpretation is to treat the network outputs as probabilities. In this case, the network gives more information than simply a decision: it tells us how sure (in a formal sense) it is of that decision. There are modifications to MLPs that allow the neural network outputs to be interpreted as probabilities, which means that the network effectively learns to model the probability density function of the class. However, the probabilistic interpretation is only valid under certain assumptions about the distribution of the data (specifically, that it is drawn from the family of exponential distributions; see Bishop, 1995). Ultimately, a classification decision must still be made, but a probabilistic interpretation allows a more formal concept of minimum cost decision making to be evolved.

Other MLP Training Algorithms

Earlier in this section, we discussed how the *back propagation* algorithm performs gradient descent on the error surface. Speaking loosely, it calculates the direction of steepest descent on the surface, and jumps down the surface a distance proportional to the learning rate and the slope, picking up momentum as it maintains a consistent direction. As an analogy, it behaves like a blindfold kangaroo hopping in the most obvious direction. Actually, the descent is calculated independently on the error surface for each training case, and in random order, but this is actually a good approximation to descent on the composite error surface. Other MLP training algorithms work differently, but all use a strategy designed to travel towards a minimum as quickly as possible.

More sophisticated techniques for non-linear function optimization have been in use for some time. These methods include *conjugate gradient descent*, quasi-Newton, and *Levenberg-Marquardt* (see Bishop, 1995; Shepherd, 1997), which are very successful forms of two types of algorithm: line search and model-trust region approaches. They are collectively known as second order training algorithms.

A line search algorithm works as follows: pick a sensible direction to move in the multi-dimensional landscape. Then project a line in that direction, locate the minimum along that line (it is relatively trivial to locate a minimum along a line, by using some form of bisection algorithm), and repeat. What is a sensible direction in this context? An obvious choice is the direction of steepest descent (the same direction that would be chosen by *back propagation*). Actually, this intuitively obvious choice proves to be rather poor. Having minimized along one direction, the next line of steepest descent may spoil the minimization along the initial direction (even on a simple surface like a parabola a large number of line searches may be necessary). A better approach is to select conjugate or non-interfering directions - hence *conjugate gradient descent* (Bishop, 1995).

The idea here is that, once the algorithm has minimized along a particular direction, the second derivative along that direction should be kept at zero. Conjugate directions are selected to maintain this zero second derivative on the assumption that the surface is parabolic (speaking roughly, a nice smooth surface). If this condition holds, *N* epochs are sufficient to reach a minimum. In reality, on a complex error surface the conjugacy deteriorates, but the algorithm still typically requires far less epochs than *back propagation*, and also converges to a better minimum (to settle down thoroughly, *back propagation* must be run with an extremely low learning rate).

Quasi-Newton training is based on the observation that the direction pointing directly towards the minimum on a quadratic surface is the so-called Newton direction. This is very expensive to calculate analytically, but quasi-Newton iteratively builds up a good approximation to it. Quasi-Newton is usually a little faster than conjugate gradient descent, but has substantially larger memory requirements and is occasionally numerically unstable.

A model-trust region approach works as follows: instead of following a search direction, assume that the surface is a simple shape such that the minimum can be located (and jumped to) directly - if the assumption is true. Try the model out and see how good the suggested point is. The model typically assumes that the surface is a nice well-behaved shape (e.g., a parabola), which will be true if sufficiently close to a minima. Elsewhere, the assumption may be grossly violated, and the model could choose wildly inappropriate points to move to. The model can only be trusted within a region of the current point, and the size of this region isn't known. Therefore, choose new points to test as a compromise between that suggested by the model and that suggested by a standard gradient-descent jump. If the new point is good, move to it, and strengthen the role of the model in selecting a new point; if it is bad, don't move, and strengthen the role of the gradient descent step in selecting a new point (and make the step smaller). *Levenberg-Marquardt* uses a model that assumes that the underlying function is locally linear (and therefore has a parabolic error surface).

Levenberg-Marquardt (Levenberg, 1944; Marquardt, 1963; Bishop, 1995) is typically the fastest of the training algorithms, although unfortunately it has some important limitations, specifically: it can only be used on single output networks, can only be used with the *sum squared error* function, and has memory requirements proportional to W_2 (where *W* is the number of weights in the network; this makes it impractical for reasonably big networks). *Conjugate gradient descent* is nearly as good, and doesn't suffer from these restrictions.

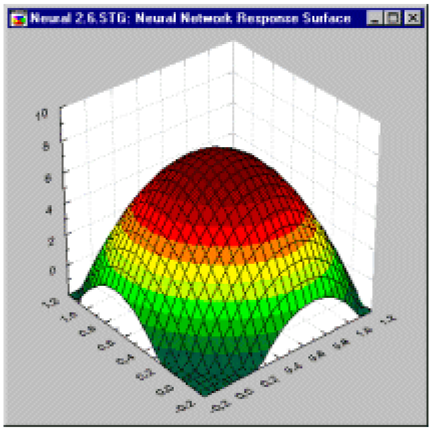
Back propagation can still be useful, not least in providing a quick (if not overwhelmingly accurate) solution. It is also a good choice if the data set is very large, and contains a great deal of redundant data. *Back propagation*'s case-by-case error adjustment means that data redundancy does it no harm (for example, if you double the data set size by replicating every case, each epoch will take twice as long, but have the same effect as two of the old epochs, so there is no loss). In contrast, *Levenberg-Marquardt*, quasi-Newton, and conjugate gradient descent all perform calculations using the entire data set, so increasing the number of cases can significantly slow each epoch, but does not necessarily improve performance on that epoch (not if data is redundant; if data is sparse, then adding data will make each epoch better). *Back propagation* can also be equally good if the data set is very small, for there is then insufficient information to make a highly fine-tuned solution appropriate (a more advanced algorithm may achieve a lower training error, but the selection error is unlikely to improve in the same way). Finally, the second order training algorithms seem to be very prone to stick in local minima in the early phases - for this reason, we recommend the practice of starting with a short burst of back propagation, before switching to a second order algorithm.

There are variations on *back propagation* (*quick propagation*, Fahlman, 1988, and *Delta-bar-Delta*, Jacobs, 1988) that are designed to deal with some of the limitations on this technique. In most cases, they are not significantly better than *back propagation*, and sometimes they are worse (relative performance is application-dependent). They also require more control parameters than any of the other algorithms, which makes them more difficult to use, so they are not described in further detail in this section.

Radial Basis Function Networks

We have seen in the last section how an MLP models the response function using the composition of sigmoid-cliff functions - for a classification problem, this corresponds to dividing the pattern space up using hyperplanes. The use of hyperplanes to divide up space is a natural approach - intuitively appealing, and based on the fundamental simplicity of lines.

An equally appealing and intuitive approach is to divide up space using circles or (more generally) hyperspheres. A hypersphere is characterized by its center and radius. More generally, just as an MLP unit responds (non-linearly) to the distance of points from the line of the sigmoid-cliff, in a radial basis function network (Broomhead and Lowe, 1988; Moody and Darkin, 1989; Haykin, 1994) units respond (non-linearly) to the distance of points from the center represented by the radial unit. The response surface of a single radial unit is therefore a Gaussian (bell-shaped) function, peaked at the center, and descending outwards. Just as the steepness of the MLP's sigmoid curves can be altered, so can the slope of the radial unit's Gaussian. See the next illustration below.



MLP units are defined by their weights and threshold, which together give the equation of the defining line, and the rate of fall-off of the function from that line. Before application of the sigmoid activation function, the activation level of the unit is determined using a weighted sum, which mathematically is the dot product of the input vector and the weight vector of the unit; these units are therefore referred to as dot product units. In contrast, a *radial unit* is defined by its center point and a radius. A point in *N* dimensional space is defined using *N* numbers, which exactly corresponds to the number of weights in a dot product unit, so the center of a radial unit is stored as weights. The radius (or deviation) value is stored as the threshold. It is worth emphasizing that the weights and thresholds in a radial unit are actually entirely different to those in a dot product unit, and the terminology is dangerous if you don't remember this: Radial weights really form a point, and a radial threshold is really a deviation.

A *radial basis function* network (RBF), therefore, has a hidden layer of radial units, each actually modeling a Gaussian response surface. Since these functions are nonlinear, it is not actually necessary to have more than one hidden layer to model any shape of function: sufficient radial units will always be enough to model any function. The remaining question is how to combine the hidden radial unit outputs into the network outputs? It turns out to be quite sufficient to use a linear combination of these outputs (i.e., a weighted sum of the Gaussians) to model any nonlinear function. The standard RBF has an output layer containing dot product units with indentity activation function (see Haykin, 1994; Bishop, 1995).

RBF networks have a number of advantages over MLPs. First, as previously stated, they can model any nonlinear function using a single hidden layer, which removes some design-decisions about numbers of layers. Second, the simple linear transformation in the output layer can be optimized fully using traditional linear modeling techniques, which are fast and do not suffer from problems such as local minima which plague MLP training techniques. RBF networks can therefore be trained extremely quickly (i.e., orders of magnitude faster than MLPs).

On the other hand, before linear optimization can be applied to the output layer of an RBF network, the number of radial units must be decided, and then their centers and deviations must be set. Although faster than MLP training, the algorithms to do this are equally prone to discover sub-optimal combinations. Other features that distinguish RBF performance from MLPs are due to the differing approaches to modeling space, with RBFs "clumpy" and MLPs "planey."

Other features which distinguish RBF performance from MLPs are due to the differing approaches to modeling space, with RBFs "clumpy" and MLPs "planey."

Experience indicates that the RBF's more eccentric response surface requires a *lot* more units to adequately model most functions. Of course, it is always possible to draw shapes that are most easily represented one way or the other, but the balance does not favor RBFs. Consequently, an RBF solution will tend to be slower to execute and more space consuming than the corresponding MLP (but it was much faster to train, which is sometimes more of a constraint).

The clumpy approach also implies that RBFs are not inclined to extrapolate beyond known data: the response drops off rapidly towards zero if data points far from the training data are used. Often the RBF output layer optimization will have set a bias level, hopefully more or less equal to the mean output level, so in fact the extrapolated output is the observed mean - a reasonable working assumption. In contrast, an MLP becomes more certain in its response when far-flung data is used. Whether this is an advantage or disadvantage depends largely on the application, but on the whole the MLP's uncritical extrapolation is regarded as a bad point: extrapolation far from training data is usually dangerous and unjustified.

RBFs are also more sensitive to the curse of dimensionality, and have greater difficulties if the number of input units is large: this problem is discussed further in a later section.

As mentioned earlier, training of RBFs takes place in distinct stages. First, the centers and deviations of the radial units must be set; then the linear output layer is optimized.

Centers should be assigned to reflect the natural clustering of the data. The two most common methods are:

Sub-sampling. Randomly-chosen training points are copied to the radial units. Since they are randomly selected, they will represent the distribution of the training data in a statistical sense. However, if the number of radial units is not large, the radial units may actually be a poor representation (Haykin, 1994).

K-Means algorithm. This algorithm (Bishop, 1995) tries to select an optimal set of points that are placed at the centroids of clusters of training data. Given *K* radial units, it adjusts the positions of the centers so that:

- Each training point belongs to a cluster center, and is nearer to this center than to any other center;
- Each cluster center is the centroid of the training points that belong to it.

Once centers are assigned, deviations are set. The size of the deviation (also known as a smoothing factor) determines how spiky the Gaussian functions are. If the Gaussians are too spiky, the network will not interpolate between known points, and the network loses the ability to generalize. If the Gaussians are very broad, the network loses fine detail. This is actually another manifestation of the over/under-fitting dilemma. Deviations should typically be chosen so that Gaussians overlap with a few nearby centers. Methods available are:

Explicit. Choose the deviation yourself.

Isotropic. The deviation (same for all units) is selected heuristically to reflect the number of centers and the volume of space they occupy (Haykin, 1994).

K-Nearest Neighbor. Each unit's deviation is individually set to the mean distance to its *K* nearest neighbors (Bishop, 1995). Hence, deviations are smaller in tightly packed areas of space, preserving detail, and higher in sparse areas of space (interpolating where necessary).

Once centers and deviations have been set, the output layer can be optimized using the standard linear optimization technique: the pseudo-inverse (singular value decomposition) algorithm (Haykin, 1994; Golub and Kahan, 1965).

However, RBFs as described above suffer similar problems to Multilayer Perceptrons if they are used for classification - the output of the network is a measure of distance from a decision hyperplane, rather than a probabilistic confidence level. We may therefore choose to modify the RBF by including an output layer with logistic or softmax (normalized exponential) outputs, which is capable of probability estimation. We lose the advantage of fast linear optimization of the output layer; however, the non-linear output layer still has a relatively well-behaved error surface, and can be optimized quite quickly using a fast iterative algorithm such as conjugate gradient descent.

Radial basis functions can also be hybridized in a number of ways. The radial layer (the hidden layer) can be trained using the Kohonen and Learned Vector Quantization training algorithms, which are alternative methods of assigning centers to reflect the spread of data, and the output layer (whether linear or otherwise) can be trained using any of the iterative dot product algorithms.

Probabilistic Neural Networks

Elsewhere, we briefly mentioned that, in the context of classification problems, a useful interpretation of network outputs was as estimates of probability of class membership, in which case the network was actually learning to estimate a probability density function (p.d.f.). A similar useful interpretation can be made in regression problems if the output of the network is regarded as the expected value of the model at a given point in input-space. This expected value is related to the joint probability density function of the output and inputs.

Estimating probability density functions from data has a long statistical history (Parzen, 1962), and in this context fits into the area of Bayesian statistics. Conventional statistics can, given a known model, inform us what the chances of certain outcomes are (e.g., we know that a unbiased die has a 1/6th chance of coming up with a six). Bayesian statistics turns this situation on its head, by estimating the validity of a model given certain data. More generally, Bayesian statistics can estimate the probability density of model parameters given the available data. To minimize error, the model is then selected whose parameters maximize this p.d.f.

In the context of a classification problem, if we can construct estimates of the p.d.f.s of the possible classes, we can compare the probabilities of the various classes, and select the most-probable. This is effectively what we ask a neural network to do when it learns a classification problem - the network attempts to learn (an approximation to) the p.d.f.

A more traditional approach is to construct an estimate of the p.d.f. from the data. The most traditional technique is to assume a certain form for the p.d.f. (typically, that it is a normal distribution), and then to estimate the model parameters. The normal distribution is commonly used as the model parameters (mean and standard deviation) can be estimated using analytical techniques. The problem is that the assumption of normality is often not justified.

An alternative approach to p.d.f. estimation is *kernel-based approximation* (see Parzen, 1962; Speckt, 1990; Speckt, 1991; Bishop, 1995; Patterson, 1996). We can reason loosely that the presence of particular case indicates some probability density at that point: a cluster of cases close together indicate an area of high probability density. Close to a case, we can have high confidence in some probability density, with a lesser and diminishing level as we move away. In kernel-based estimation, simple functions are located at each available case, and added together to estimate the overall p.d.f. Typically, the kernel functions are each Gaussians (bell-shapes). If sufficient training points are available, this will indeed yield an arbitrarily good approximation to the true p.d.f.

This kernel-based approach to p.d.f. approximation is very similar to radial basis function networks, and motivates the probabilistic neural network (PNN) and generalized regression neural network (GRNN), both devised by Speckt (1990 and 1991). PNNs are designed for classification tasks, and GRNNs for regression. These two types of network are really kernel-based approximation methods cast in the form of neural networks.

In the PNN, there are at least three layers: input, radial, and output layers. The radial units are copied directly from the training data, one per case. Each models a Gaussian function centered at the training case. There is one output unit per class. Each is connected to all the radial units belonging to its class, with zero connections from all other radial units. Hence, the output units simply add up the responses of the units belonging to their own class. The outputs are each proportional to the kernel-based estimates of the p.d.f.s of the various classes, and by normalizing these to sum to 1.0 estimates of class probability are produced.

The basic PNN can be modified in two ways.

First, the basic approach assumes that the proportional representation of classes in the training data matches the actual representation in the population being modeled (the so-called prior probabilities). For example, in a disease-diagnosis network, if 2% of the population has the disease, then 2% of the training cases should be positives. If the prior probability is different from the level of representation in the training cases, then the network's estimate will be invalid. To compensate for this, prior probabilities can be given (if known), and the class weightings are adjusted to compensate.

Second, any network making estimates based on a noisy function will inevitably produce some misclassifications (there may be disease victims whose tests come out normal, for example). However, some forms of misclassification may be regarded as more expensive mistakes than others (for example, diagnosing somebody healthy as having a disease, which simply leads to exploratory surgery may be inconvenient but not life-threatening; whereas failing to spot somebody who is suffering from disease may lead to premature death). In such cases, the raw probabilities generated by the network can be weighted by loss factors, which reflect the costs of misclassification. A fourth layer can be specified in PNNs which includes a loss matrix. This is multiplied by the probability estimates in the third layer, and the class with lowest estimated cost is selected. (Loss matrices may also be attached to other types of classification network).

The only control factor that needs to be selected for probabilistic neural network training is the smoothing factor (i.e., the radial deviation of the Gaussian functions). As with RBF networks, this factor needs to be selected to cause a reasonable amount of overlap - too small deviations cause a very spiky approximation which cannot generalize, too large deviations smooth out detail. An appropriate figure is easily chosen by experiment, by selecting a number which produces a low selection error, and fortunately PNNs are not too sensitive to the precise choice of smoothing factor.

The greatest advantages of PNNs are the fact that the output is probabilistic (which makes interpretation of output easy), and the training speed. Training a PNN actually consists mostly of copying training cases into the network, and so is as close to instantaneous as can be expected.

The greatest disadvantage is network size: a PNN network actually contains the entire set of training cases, and is therefore space-consuming and slow to execute.

PNNs are particularly useful for prototyping experiments (for example, when deciding which input parameters to use), as the short training time allows a great number of tests to be conducted in a short period of time.

Generalized Regression Neural Networks

Generalized regression neural networks (GRNNs) work in a similar fashion to PNNs, but perform regression rather than classification tasks (see Speckt, 1991; Patterson, 1996; Bishop, 1995). As with the PNN, Gaussian kernel functions are located at each training case. Each case can be regarded, in this case, as evidence that the response surface is a given height at that point in input space, with progressively decaying evidence in the immediate vicinity. The GRNN copies the training cases into the network to be used to estimate the response on new points. The output is estimated using a weighted average of the outputs of the training cases, where the weighting is related to the distance of the point from the point being estimated (so that points nearby contribute most heavily to the estimate).

The first hidden layer in the GRNN contains the radial units. A second hidden layer contains units that help to estimate the weighted average. This is a specialized procedure. Each output has a special unit assigned in this layer that forms the weighted sum for the corresponding output. To get the weighted average from the weighted sum, the weighted sum must be divided through by the sum of the weighting factors. A single special unit in the second layer calculates the latter value. The output layer then performs the actual divisions (using special division units). Hence, the second hidden layer always has exactly one more unit than the output layer. In regression problems, typically only a single output is estimated, and so the second hidden layer usually has two units.

The GRNN can be modified by assigning radial units that represent clusters rather than each individual training case: this reduces the size of the network and increases execution speed. Centers can be assigned using any appropriate algorithm (i.e., sub-sampling, *K*-means or Kohonen).

GRNNs have advantages and disadvantages broadly similar to PNNs - the difference being that GRNNs can only be used for regression problems, whereas PNNs are used for classification problems. A GRNN trains almost instantly, but tends to be large and slow (although, unlike PNNs, it is not necessary to have one radial unit for each training case, the number still needs to be large). Like an RBF network, a GRNN does not extrapolate.

Linear Networks

A general scientific principal is that a simple model should always be chosen in preference to a complex model if the latter does not fit the data better. In terms of function approximation, the simplest model is the linear model, where the fitted function is a hyperplane. In classification, the hyperplane is positioned to divide the two classes (a linear discriminant function); in regression, it is positioned to pass through the data. A linear model is typically represented using an *NxN* matrix and an *Nx1* bias vector.

A neural network with no hidden layers, and an output with dot product synaptic function and identity activation function, actually implements a linear model. The weights correspond to the matrix, and the thresholds to the bias vector. When the network is executed, it effectively multiplies the input by the weights matrix then adds the bias vector.

The linear network provides a good benchmark against which to compare the performance of your neural networks. It is quite possible that a problem that is thought to be highly complex can actually be solved as well by linear techniques as by neural networks. If you have only a small number of training cases, you are probably anyway not justified in using a more complex model.

SOFM Networks

Self Organizing Feature Map (SOFM, or Kohonen) networks are used quite differently to the other networks. Whereas all the other networks are designed for supervised learning tasks, SOFM networks are designed primarily for *unsupervised learning* (see Kohonen, 1982; Haykin, 1994; Patterson, 1996; Fausett, 1994).

Whereas in supervised learning the training data set contains cases featuring input variables together with the associated outputs (and the network must infer a mapping from the inputs to the outputs), in unsupervised learning the training data set contains only input variables.

At first glance this may seem strange. Without outputs, what can the network learn? The answer is that the SOFM network attempts to learn the structure of the data.

One possible use is therefore in exploratory data analysis. The SOFM network can learn to recognize clusters of data, and can also relate similar classes to each other. The user can build up an understanding of the data, which is used to refine the network. As classes of data are recognized, they can be labeled, so that the network becomes capable of classification tasks. SOFM networks can also be used for classification when output classes are immediately available - the advantage in this case is their ability to highlight similarities between classes.

A second possible use is in novelty detection. SOFM networks can learn to recognize clusters in the training data, and respond to it. If new data, unlike previous cases, is encountered, the network fails to recognize it and this indicates novelty.

A SOFM network has only two layers: the input layer, and an output layer of radial units (also known as the *topological map* layer). The units in the topological map layer are laid out in space - typically in two dimensions (although *ST Neural Networks* also supports one-dimensional Kohonen networks).

SOFM networks are trained using an iterative algorithm. Starting with an initially-random set of radial centers, the algorithm gradually adjusts them to reflect the clustering of the training data. At one level, this compares with the sub-sampling and *K*-Means algorithms used to assign centers in RBF and GRNN networks, and indeed the SOFM algorithm can be used to assign centers for these types of networks. However, the algorithm also acts on a different level.

The iterative training procedure also arranges the network so that units representing centers close together in the input space are also situated close together on the topological map. You can think of the network's topological layer as a crude two-dimensional grid, which must be folded and distorted into the N-dimensional input space, so as to preserve as far as possible the original structure. Clearly any attempt to represent an N-dimensional space in two dimensions will result in loss of detail; however, the technique can be worthwhile in allowing the user to visualize data which might otherwise be impossible to understand.

The basic iterative Kohonen algorithm simply runs through a number of epochs, on each epoch executing each training case and applying the following algorithm:

- Select the winning neuron (the one who's center is nearest to the input case);
- Adjust the winning neuron to be more like the input case (a weighted sum of the old neuron center and the training case).

The algorithm uses a time-decaying learning rate, which is used to perform the weighted sum and ensures that the alterations become more subtle as the epochs pass. This ensures that the centers settle down to a compromise representation of the cases which cause that neuron to win.

The topological ordering property is achieved by adding the concept of a neighborhood to the algorithm. The neighborhood is a set of neurons surrounding the winning neuron. The neighborhood, like the learning rate, decays over time, so that initially quite a large number of neurons belong to the neighborhood (perhaps almost the entire topological map); in the latter stages the neighborhood will be zero (i.e., consists solely of the winning neuron itself). In the Kohonen algorithm, the adjustment of neurons is actually applied not just to the winning neuron, but to all the members of the current neighborhood.

The effect of this neighborhood update is that initially quite large areas of the network are "dragged towards" training cases - and dragged quite substantially. The network develops a crude topological ordering, with similar cases activating clumps of neurons in the topological map. As epochs pass the learning rate and neighborhood both decrease, so that finer distinctions within areas of the map can be drawn, ultimately resulting in fine-tuning of individual neurons. Often, training is deliberately conducted in two distinct phases: a relatively short phase with high learning rates and neighborhood, and a long phase with low learning rate and zero or near-zero neighborhood.

Once the network has been trained to recognize structure in the data, it can be used as a visualization tool to examine the data. The *Win Frequencies Datasheet* (counts of the number of times each neuron wins when training cases are executed) can be examined to see if distinct clusters have formed on the map. Individual cases are executed and the topological map observed, to see if some meaning can be assigned to the clusters (this usually involves referring back to the original application area, so that the relationship between clustered cases can be established). Once clusters are identified, neurons in the topological map are labeled to indicate their meaning (sometimes individual cases may be labeled, too). Once the topological map has been built up in this way, new cases can be submitted to the network. If the winning neuron has been labeled with a class name, the network can perform classification. If not, the network is regarded as undecided.

SOFM networks also make use of the accept threshold, when performing classification. Since the activation level of a neuron in a SOFM network is the distance of the neuron from the input case, the accept threshold acts as a maximum recognized distance. If the activation of the winning neuron is greater than this distance, the SOFM network is regarded as undecided. Thus, by labeling all neurons and setting the accept threshold appropriately, a SOFM network can act as a novelty detector (it reports undecided only if the input case is sufficiently dissimilar to all radial units).

SOFM networks are inspired by some known properties of the brain. The cerebral cortex is actually a large flat sheet (about 0.5m squared; it is folded up into the familiar convoluted shape only for convenience in fitting into the skull!) with known topological properties (for example, the area corresponding to the hand is next to the arm, and a distorted human frame can be topologically mapped out in two dimensions on its surface).

Classification in ST Neural Networks

In classification problems, the purpose of the network is to assign each case to one of a number of classes (or, more generally, to estimate the probability of membership of the case in each class). Nominal output variables are used to indicate a classification problem. The nominal values correspond to the various classes.

Nominal variables are normally represented in networks using one of two techniques, the first of which is only available for two-state variables; these techniques are: *two-state*, *one-of-N*. In two-state representation, a single node corresponds to the variable, and a value of 0.0 is interpreted as one state, and a value of 1.0 as the other. In *one-of-N* encoding, one unit is allocated for each state, with a particular state represented by 1.0 on that particular unit, and 0.0 on the others.

Input nominal variables are easily converted using the above methods, both during training and during execution. Target outputs for units corresponding to nominal variables are also easily determined during training. However, more effort is required to determine the output class assigned by a network during execution.

The output units each have continuous activation values between 0.0 and 1.0. In order to definitely assign a class from the outputs, the network must decide if the outputs are reasonably close to 0.0 and 1.0. If they are not, the class is regarded as undecided.

Confidence levels (the accept and reject thresholds) decide how to interpret the network outputs. These thresholds can be adjusted to make the network more or less fussy about when to assign a classification. The interpretation differs slightly for *two-state* and *one-of-N* representation:

Two-state. If the unit output is above the accept threshold, the 1.0 class is deemed to be chosen. If the output is below the reject threshold, the 0.0 class is chosen. If the output is between the two thresholds, the class is undecided.

One-of-N. A class is selected if the corresponding output unit is above the accept threshold and all the other output units are below the reject threshold. If this condition is not met, the class is undecided.

For one-of-N encoding, the use of thresholds is optional. If not used, the "winner-takes-all" algorithm is used (the highest activation unit gives the class, and the network is never undecided). There is one peculiarity when dealing with *one-of-N* encoding. On first reading, you might expect a network with accept and reject thresholds set to 0.5 is equivalent to a "winner takes all" network. Actually, this is not the case for *one-of-N* encoded networks (it **is** the case for two-state). You can actually set the accept threshold lower than the reject threshold, and only a network with accept 0.0 and reject 1.0 is equivalent to a winner-takes-all network. This is true since the algorithm for assigning a class is actually:

- Select the unit with the highest output. If this unit has output greater than or equal to the accept threshold, and all other units have output less than the reject threshold, assign the class represented by that unit.

With an accept threshold of 0.0, the winning unit is bound to be accepted, and with a reject threshold of 1.0, none of the other units can possibly be rejected, so the algorithm reduces to a simple selection of the winning unit. In contrast, if both accept and reject are set to 0.5, the network may return undecided (if the winner is below 0.5, or any of the losers are above 0.5).

Although this concept takes some getting used to, it does allow you to set some subtle conditions. For example, accept/reject 0.3/0.7 can be read as: "select the class using the winning unit, provided it has an output level at least 0.3, and none of the other units have activation above 0.7" - in other words, the winner must show some significant level of activation, and the losers mustn't, for a decision to be reached.

If the network's output unit activations are probabilities, the range of possible output patterns is of course restricted, as they must sum to 1.0. In that case, winner-takes-all is equivalent to setting accept and reject both to 1/N, where N is the number of classes. The above discussion covers the assignment of classifications in most types of network: MLPs, RBFs, linear and Cluster. However, SOFM networks work quite differently.

In a SOFM network, the winning node in the topological map (output) layer is the one with the lowest activation level (which measures the distance of the input case from the point stored by the unit). Some or all of the units in the topological map may be labeled, indicating an output class. If the distance is small enough, then the case is assigned to the class (if one is given). The accept threshold indicates the largest distance which will result in a positive classification. If an input case is further than this distance away from the winning unit, or if the winning unit is unlabelled (or its label doesn't match one of the output variable's nominal values) then the case is unclassified. The reject threshold is not used in SOFM networks.

The discussion on non-SOFM networks has assumed that a positive classification is indicated by a figure close to 1.0, and a negative classification by a figure close to 0.0. This is true if the logistic output activation function is used, and is convenient as probabilities range from 0.0 to 1.0. However, in some circumstances a different range may be used. Also, sometimes ordering is reversed, with smaller outputs indicating higher confidence.

First, the range values used are actually the min/mean and max/SD values stored for each variable. With a logistic output activation function, the default values 0.0 and 1.0 are fine. Some authors actually recommend using the hyperbolic tangent activation function, which has the range (-1.0,+1.0) . Training performance may be enhanced because this function (unlike the logistic function) is symmetrical. Alternatively (and we recommend this practice) use hyperbolic tangent activation function in hidden layers, but not in the output layer.

Ordering is typically reversed in two situations. We have just discussed one of these: SOFM networks, where the output is a distance measure, with a small value indicating greater confidence. The same is true in the closely-related Cluster networks. The second circumstance is the use of a loss matrix (which may be added at creation time to PNNs, and also manually joined to other types of network). When a loss matrix is used, the network outputs indicate the expected cost if each class is selected, and the objective is to select the class with the lowest cost. In this case, we would normally expect the accept threshold to be smaller than the reject threshold.

Classification Statistics

When selecting accept/reject thresholds, and assessing the classification ability of the network, the most important indicator is the *classification summary spreadsheet*. This shows how many cases were correctly classified, incorrectly classified, or unclassified. You can also use the confusion matrix spreadsheet to break down how many cases belonging to each class were assigned to another class. All these figures can be independently reported for the training, selection and test sets.

Regression Problems in ST Neural Networks

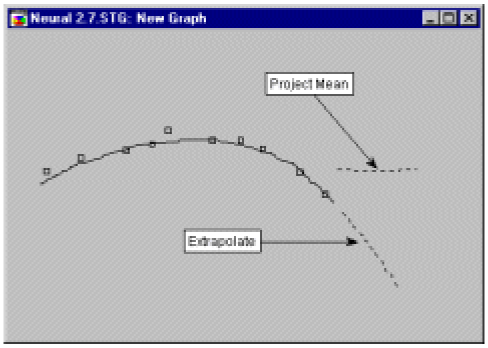
In regression problems, the objective is to estimate the value of a continuous output variable, given the known input variables. Regression problems can be solved using the following network types: MLP, RBF, GRNN and Linear. Regression problems are represented by data sets with non-nominal (standard numeric) output(s).

A particularly important issue in regression is output scaling, and extrapolation effects.

The most common neural network architectures have outputs in a limited range (e.g., (0,1) for the logistic activation function). This presents no difficulty for classification problems, where the desired output is in such a range. However, for regression problems there clearly is an issue to be resolved, and some of the consequences are quite subtle.

This subject is discussed below.

As a first pass, we can apply a scaling algorithm to ensure that the network's output will be in a sensible range. The simplest scaling function is minimax: this finds the minimum and maximum values of a variable in the training data, and performs a linear transformation (using a shift and a scale factor) to convert the values into the target range (typically [0.0,1.0]). If this is used on a continuous output variable, then we can guarantee that all training values will be converted into the range of possible outputs of the network, and so the network can be trained. We also know that the network's output will be constrained to lie within this range. This may or may not be regarded as a good thing, which brings us to the subject of extrapolation.



Consider the figure above. Here, we are trying to estimate the value of y from the value of x. A curve has to be fitted that passes through the available data points. We can probably easily agree on the illustrated curve, which is approximately the right shape, and this will allow us to estimate y given inputs in the range represented by the solid line where we can interpolate.

However, what about a point well to the right of the data points? There are two possible approaches to estimating y for this point. First, we might decide to extrapolate: projecting the trend of the fitted curve onwards. Second, we might decide that we don't really have sufficient evidence to assign any value, and therefore assign the mean output value (which is probably the best estimate we have lacking any other evidence).

Let us assume that we are using an MLP. Using minimax as suggested above is highly restrictive. First, the curve is not extrapolated, however close to the training data we may be (if we are only a little bit outside the training data, extrapolation may well be justified). Second, it does not estimate the mean either - it actually saturates at either the minimum or maximum, depending on whether the estimated curve was rising or falling as it approached this region.

There are a number of approaches to correct this deficiency in an MLP:

First, we can replace the logistic output activation function with a linear activation function, which simply passes on the activation level unchanged (N.B. only the activation functions in the output layer are changed; the hidden layers still use logistic or hyperbolic activation functions). The linear activation function does not saturate, and so can extrapolate further (the network will still saturate eventually as the hidden units saturate). A linear activation function in an MLP can cause some numerical difficulties for the *back propagation* algorithm, however, and if this is used a low learning rate (below 0.1) must be used. This approach may be appropriate if you want to extrapolate.

Second, you can alter the target range for the minimax scaling function (for example, to [0.1,0.9]). The training cases are then all mapped to levels that correspond to only the middle part of the output units' output range. Interestingly, if this range is small, with both figures close to 0.5, it corresponds to the middle part of the sigmoid curve that is nearly linear, and the approach is then quite similar to using a linear output layer. Such a network can then perform limited extrapolation, but eventually saturates. This has quite a nice intuitive interpretation: extrapolation is justified for a certain distance, and then should be curtailed.

If may have occurred to you that if the first approach is used, and linear units are placed in the output layer, there is no need to use a scaling algorithm at all, since the units can achieve any output level without scaling. However, in reality the entire removal of scaling presents difficulties to the training algorithms. It implies that different weights in the network operate on very different scales, which makes both initialization of weights and (some) training more complex. It is therefore not recommended that you turn off scaling unless the output range is actually very small and close to zero. The same argument actually justifies the use of scaling during preprocessing for MLPs (where, in principal, the first hidden layer weights could simply be adjusted to perform any scaling required).

The above discussion focused on the performance of MLPs in regression, and particularly their behavior with respect to extrapolation. Networks using radial units (RBFs and GRNNs) perform quite differently, and need different treatment.

Radial networks are inherently incapable of extrapolation. As the input case gets further from the points stored in the radial units, so the activation of the radial units decays and (ultimately) the output of the network decays. An input case located far from the radial centers will generate a zero output from all hidden units. The tendency not to extrapolate can be regarded as good (depending on your problem-domain and viewpoint), but the tendency to decay to a zero output (at first sight) is not. If we decide to eschew extrapolation, then what we would like to see reported at highly novel input points is the mean. In fact, the RBF has a bias value on the output layer, and sets this to a convenient value, which hopefully approximates the sample mean. Then, the RBF will always output the mean if asked to extrapolate.

Using the mean/SD scaling function with radial networks in regression problems, the training data is scaled so that its output mean corresponds to 0.0, with other values scaled according to the output standard deviation, and the bias is expected to be approximately zero. As input points are executed outside the range represented in the radial units, the output of the network tends back towards the mean.

The performance of a regression network can be examined in a number of ways.

1. The output of the network for each case (or any new case you choose to test) can be submitted to the network. If part of the data set, the residual errors can also be generated.
2. Summary statistics can be generated. These include the mean and standard deviation of both the training data values and the prediction error. One would

generally expect to see a prediction error mean extremely close to zero (it is, after all, possible to get a zero prediction error mean simply by estimating the mean training data value, without any recourse to the input variables or a neural network at all). The most significant value is the prediction error standard deviation. If this is no better than the training data standard deviation, then the network has performed no better than a simple mean estimator. A ratio of the prediction error SD to the training data SD significantly below 1.0 indicates good regression performance, with a level below 0.1 often said (heuristically) to indicate good regression. This regression ratio (or, more accurately, one minus this ratio) is sometimes referred to as the explained variance of the model.

The regression statistics also include the Pearson-R correlation coefficient between the network's prediction and the observed values. In linear modeling, the Pearson-R correlation between the predictor variable and the predicted is often used to express correlation - if a linear model is fitted, this is identical to the correlation between the model's prediction and the observed values (or, to the negative of it). Thus, this gives you a convenient way to compare the neural network's accuracy with that of your linear models.

3. A view of the response surface can be generated. The network's actual response surface is, of course, constructed in *N+1* dimensions, where *N* is the number of input units, and the last dimension plots the height. It is clearly impossible to directly visualize this surface where *N* is anything greater than two (which it invariably is).

Time Series Prediction in ST Neural Networks

In time series problems, the objective is to predict ahead the value of a variable that varies in time, using previous values of that and/or other variables (see Bishop, 1995)

Typically the predicted variable is continuous, so that time series prediction is usually a specialized form of regression. However, without this restriction, time series can also do prediction of nominal variables (i.e,. classification).

It is also usual to predict the next value in a series from a fixed number of previous values (looking ahead a single time step). When the next value in a series is generated, further values can be estimated by feeding the newly-estimated value back into the network together with other previous values: time series projection. Obviously, the reliability of projection drops the more steps ahead one tries to predict, and if a particular distance ahead is required, it is probably better to train a network specifically for that degree of lookahead.

Any type of network can be used for time series prediction (the network type must, however, be appropriate for regression or classification, depending on the problem type). The network can also have any number of input and output variables. However, most commonly there is a single variable that is both the input and (with the lookahead taken into account) the output. Configuring a network for time series usage alters the way that data is pre-processed (i.e., it is drawn from a number of sequential cases, rather than a single case), but the network is executed and trained just as for any other problem.

The time series training data set therefore typically has a single variable, and this has type input/output (i.e., it is used both for network input and network output).

The most difficult concept in time series handling is the interpretation of training, selection, test and ignored cases. For standard data sets, each case is independent, and these meanings are clear. However, with a time series network each pattern of inputs and outputs is actually drawn from a number of cases, determined by the network's *Steps* and *Lookahead* parameters. There are two consequences of this:

The input pattern's type is taken from the type of the output case. For example, in a data set containing some cases, the first two ignored and the third test, with *Steps=2* and *Lookahead=1*, the first usable pattern has type Test, and draws its inputs from the first two cases, and its output from the third. Thus, the first two cases are used in the test set even though they are marked Ignore. Further, any given case may be used in three patterns, and these may be any of training, selection and test patterns. In some sense, data actually leaks between training, selection and test sets. To isolate the three sets entirely, contiguous blocks of train, verify or test cases would need to be constructed, separated by the appropriate number of ignore cases.

The first few cases can only be used as inputs for patterns. When selecting cases for time series use, the case number selected is always the output case. The first few clearly cannot be selected (as this would require further cases before the beginning of the data set), and are not available.

Variable Selection and Dimensionality Reduction

The most common approach to dimensionality reduction is principal components analysis (see Bishop, 1995; Bouland and Kamp, 1988). This is a linear transformation that locates directions of maximum variance in the original input data, and rotates the data along these axes. Typically, the first principal components contain most information. Principal component analysis can be represented in a linear network. PCA can often extract a very small number of components from quite high-dimensional original data and still retain the important structure.

The preceding sections on network design and training have all assumed that the input and output layers are fixed; that is, that we know what variables will be input to the network, and what output is expected. The latter is always (at least, for supervised learning problems) known. However, the selection of inputs is far more difficult (see Bishop, 1995). Often, we do not know which of a set of candidate input variables are actually useful, and the selection of a good set of inputs is complicated by a number of important considerations:

Curse of dimensionality. Each additional input unit in a network adds another dimension to the space in which the data cases reside. We are attempting to fit a response surface to this data. Thought of in this way, there must be sufficient data points to populate an *N* dimensional space sufficiently densely to be able to see the structure. The number of points needed to do this properly grows very rapidly with the dimensionality (roughly, in proportion to 2N for most modelling techniques). Most forms of neural network (in particular, MLPs) actually suffer less from the curse of dimensionality than some other methods, as they can concentrate on a lower-dimensional section of the high-dimensional space (for example, by setting the outgoing weights from a particular input to zero, an MLP can entirely ignore that input). Nevertheless, the curse of dimensionality is still a problem, and the performance of a network can certainly be improved by eliminating unnecessary input variables. Indeed, even input variables that carry a small amount of information may sometimes be better eliminated if this reduces the curse of dimensionality.

Inter-dependency of variables. It would be extremely useful if each candidate input variable could be independently assessed for usefulness, so that the most useful ones could be extracted. Unfortunately, it is seldom possible to do this, and two or more interdependent variables may together carry significant information that a subset would not. A classic example is the two-spirals problem, where two classes of data are laid out in an interlocking spiral pattern in two dimensions. Either variable alone carries no useful information (the two classes appear wholly intermixed), but with the two variables together the two classes can be perfectly distinguished. Thus, variables cannot, in general, be independently selected.

Redundancy of variables. Often a number of variables can carry to some extent or other the same information. For example, the height and weight of people might in many circumstances carry similar information, as these two variables are correlated. It may be sufficient to use as inputs some subset of the correlated variables, and the choice of subset may be arbitrary. The superiority of a subset of correlated variables over the full set is a consequence of the curse of dimensionality.

Selection of input variables is therefore a critical part of neural network design. You can use a combination of your own expert knowledge of the problem domain, and standard statistical tests to make some selection of variables before starting to use Neural Networks. Once you begin using Neural Networks, various combinations of inputs can be tried. You can experimentally add and remove various combinations, building new networks for each. You can also conduct Sensitivity Analysis, which rates the importance of variable with respect to a particular model.

When experimenting in this fashion, the probabilistic and generalized regression networks are extremely useful. Although slow to execute, compared with the more compact MLPs and RBFs, they train almost instantaneously - and when iterating through a large number of input variable combinations, you will need to repeatedly build networks. Moreover, PNNs and GRNNs are both (like RBFs) examples of radially-based networks (i.e., they have radial units in the first layer, and build functions from a combination of Gaussians). This is an advantage when selecting input variables because radially-based networks actually suffer *more* from the curse of dimensionality than linearly-based networks.

To explain this statement, consider the effect of adding an extra, perfectly spurious input variable to a network. A linearly-based network such as an MLP can learn to set the outgoing weights of the spurious input unit to 0, thus ignoring the spurious input (in practice, the initially-small weights will just stay small, while weights from relevant inputs diverge). A radially-based network such as a PNN or GRNN has no such luxury: clusters in the relevant lower-dimensional space get smeared out through the irrelevant dimension, requiring larger numbers of units to encompass the irrelevant variability. A network that suffers from poor inputs actually has an advantage when trying to eliminate such inputs.

This form of experimentation is time-consuming, and several feature selection algorithms exist, including the genetic algorithm (Goldberg, 1989). Genetic Algorithms are very good at this kind of problem, having a capability to search through large numbers of combinations where there may be interdependencies between variables.

Another approach to dealing with dimensionality problems, which may be an alternative or a complement to variable selection, is *dimensionality reduction*. In dimensionality reduction, the original set of variables is processed to produce a new and smaller set of variables that contains (one hopes) as much information as possible from the original set. As an example, consider a data set where all the points lie on a plane in a three dimensional space. The *intrinsic dimensionality* of the data is said to be two (as all the information actually resides in a two-dimensional sub-space). If this plane can be discovered, the neural network can be presented with a lower dimensionality input, and stands a better chance of working correctly.

Ensembles and Resampling

We have already discussed the problem of over-learning, which can compromise the ability of neural networks to generalize successfully to new data. An important approach to improve performance is to form ensembles of neural networks. The member networks' predictions are averaged (or combined by voting) to form the ensemble's prediction. Frequently, ensemble formation is combined with resampling of the data set. This approach can significantly improve generalization performance. Resampling can also be useful for improved estimation of network generalization performance.

To explain why resampling and ensembles are so useful, it is helpful to formulate the neural network training process in statistical terms (Bishop, 1995). We regard the problem as that of estimating an unknown nonlinear function, which has additive noise, on the basis of a limited data set of examples, D. There are several sources of error in our neural network's predictions. First, and unavoidably, even a "perfect" network that exactly modeled the underlying function would make errors due to the noise. However, there is also error due to the fact that we need to fit the neural network model using the finite sample data set, D. This remaining error can be split into two components, the model bias and variance. The bias is the average error that a particular model training procedure will make across different particular data sets (drawn from the unknown function's distribution). The variance reflects the sensitivity of the modeling procedure to a particular choice of data set.

We can trade off bias versus variance. At one extreme, we can arbitrarily select a function that entirely ignores the data. This has zero variance, but presumably high bias, since we have not actually taken into account the known aspects of the problem at all. At the opposite extreme, we can choose a highly complex function that can fit every point in a particular data set, and thus has zero bias, but high variance as this complex function changes shape radically to reflect the exact points in a given data set. The high bias, low variance solutions can have low complexity (e.g., linear models), whereas the low bias, high variance solutions have high complexity. In neural networks, the low complexity models have smaller numbers of units.

How does this relate to ensembles and resampling? We necessarily divide the data set into subsets for training, selection, and test. Intuitively, this is a shame, as not all the data gets used for training. If we resample, using a different split of data each time, we can build multiple neural networks, and all the data gets used for training at least some of them. If we then form the networks into an ensemble, and average the predictions, an extremely useful result occurs. Averaging across the models reduces the variance, without increasing the bias. Arguably, we can afford to build higher bias models than we would otherwise tolerate (i.e., higher complexity models), on the basis that ensemble averaging can then mitigate the resulting variance.

The generalization performance of an ensemble can be better than that of the best member network, although this does depend on how good the other networks in the ensemble are. Unfortunately, it is not possible to show whether this is actually the case for a given ensemble. However, there are some reassuring pieces of theory to back up the use of ensembles.

First, it can be shown (Bishop, 1995) that, on the assumption that the ensemble members' errors have zero mean and are uncorrelated, the ensemble reduces the error by a factor of N, where N is the number of members. In practice, of course, these errors are not uncorrelated. An important corollary is that an ensemble is more effective when the members are less correlated, and we might intuitively expect that to be the case if diverse network types and structures are used.

Second, and perhaps more significantly, it can be shown that the expected error of the ensemble is at least as good as the average expected error of the members, and usually better. Typically, some useful reduction in error does occur. There is of course a cost in processing speed, but for many applications this is not particularly problematic.

There are a number of approaches to resampling available.

The simplest approach is random (monte carlo) resampling, where the training, selection and test sets are simply drawn at random from the data set, keeping the sizes of the subsets constant. Alternatively, you CAN sometimes resample the training and selection set, but keep the test set the same, to support a simple direct comparison of results. The second approach is the popular cross-validation algorithm. Here, the data set is divided into a number of equal sized divisions. A number of neural networks are created. For each of these, one division is used for the test data, and the others are used for training and selection. In the most extreme version of this algorithm, leave-one-out cross validation, N divisions are made, where N is the number of cases in the data set, and on each division the network is trained on all bar one of the cases, and tested on the single case that is omitted. This allows the training algorithm to use virtually the entire data set for training, but is obviously very intensive.

The third approach is bootstrap sampling. In the bootstrap, a new training set is formed by sampling with replacement from the available data set. In sampling with replacement, cases are drawn at random from the data set, with equal probability, and any one case may be selected any number of times. Typically the bootstrap set has the same number of cases as the data set, although this is not a necessity. Due to the sampling process, it is likely that some of the original cases will not be selected, and these can be used to form a test set, whereas other cases will have been duplicated.

The bootstrap procedure replicates, insofar as is possible with limited data, the idea of drawing multiple data sets from the original distribution. Once again, the effect can be to generate a number of models with low bias, and to average out the variance. Ensembles can also be beneficial at averaging out bias. If we include different network types and configurations in an ensemble, it may be that different networks make systematic errors in different parts of the input space. Averaging these differently configured networks may iron out some of this bias.

Recommended Textbooks

Bishop, C. (1995). *Neural Networks for Pattern Recognition.* Oxford: University Press. Extremely well-written, up-to-date. Requires a good mathematical background, but rewards careful reading, putting neural networks firmly into a statistical context.

Carling, A. (1992). *Introducing Neural Networks.* Wilmslow, UK: Sigma Press. A relatively gentle introduction. Starting to show its age a little, but still a good starting point.

Fausett, L. (1994). *Fundamentals of Neural Networks.* New York: Prentice Hall. A well-written book, with very detailed worked examples to explain how the algorithms function.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation.* New York: Macmillan Publishing. A comprehensive book, with an engineering perspective. Requires a good mathematical background, and contains a great deal of background theory.

Patterson, D. (1996). *Artificial Neural Networks.* Singapore: Prentice Hall. Good wide-ranging coverage of topics, although less detailed than some other books.

Ripley, B.D. (1996). *Pattern Recognition and Neural Networks.* Cambridge University Press. A very good advanced discussion of neural networks, firmly putting them in the wider context of statistical modeling.

Chapter 30
Nonlinear Estimation

- General Purpose
- Estimating Linear and Nonlinear Models
- Common Nonlinear Regression Models
    - Intrinsically Linear Regression Models
    - Intrinsically Nonlinear Regression Models
- Nonlinear Estimation Procedures
    - Least Squares Estimation
    - Loss Functions
    - Weighted Least Squares
    - Maximum Likelihood
    - Maximum likelihood and probit/logit models
    - Function Minimization Algorithms
    - Start Values, Step Sizes, Convergence Criteria
    - Penalty Functions, Constraining Parameters
    - Local Minima
    - Quasi-Newton Method
    - Simplex Procedure
    - Hooke-Jeeves Pattern Moves
    - Rosenbrock Pattern Search
    - Hessian Matrix and Standard Errors
- Evaluating the Fit of the Model
    - Proportion of Variance Explained
    - Goodness-of-fit Chi-square
    - Plot of Observed vs. Predicted Values
    - Normal and Half-Normal Probability Plots
    - Plot of the Fitted Function
    - Variance/Covariance Matrix for Parameters

General Purpose

In the most general terms, *Nonlinear Estimation* will compute the relationship between a set of independent variables and a dependent variable. For example, we may want to compute the relationship between the dose of a drug and its effectiveness, the relationship between training and subsequent performance on a task, the relationship between the price of a house and the time it takes to sell it, etc. You may recognize research issues in these examples that are commonly addressed by such techniques as multiple regression (see, *Multiple Regression*) or analysis of variance (see, *ANOVA/MANOVA*). In fact, you may think of *Nonlinear Estimation* as a *generalization* of those methods. Specifically, multiple regression (and ANOVA) assumes that the relationship between the independent variable(s) and the dependent variable is *linear* in nature. *Nonlinear Estimation* leaves it up to you to specify the nature of the relationship; for example, you may specify the dependent variable to be a logarithmic function of the independent variable(s), an exponential function, a function of some complex ratio of independent measures, etc. (However, if all variables of interest are categorical in nature, or can be converted into categorical variables, you may also consider *Correspondence Analysis*.)

When allowing for any type of relationship between the independent variables and the dependent variable, two issues raise their heads. First, what types of relationships "make sense", that is, are interpretable in a meaningful manner? Note that the simple linear relationship is very convenient in that it allows us to make such straightforward interpretations as "the more of *x* (e.g., the higher the price of a house), the more there is of *y* (the longer it takes to sell it); and given a particular increase in *x*, a proportional increase in *y* can be expected." Nonlinear relationships cannot usually be interpreted and verbalized in such a simple manner. The second issue that needs to be addressed is how to exactly compute the relationship, that is, how to arrive at results that allow us to say whether or not there is a nonlinear relationship as predicted.

Let us now discuss the nonlinear regression problem in a somewhat more formal manner, that is, introduce the common terminology that will allow us to examine the nature of these techniques more closely, and how they are used to address important questions in various research domains (medicine, social sciences, physics, chemistry, pharmacology, engineering, etc.).

Estimating Linear and Nonlinear Models

Technically speaking, *Nonlinear Estimation* is a general fitting procedure that will estimate any kind of relationship between a dependent (or response variable), and a list of independent variables. In general, all regression models may be stated as:

$$y = F(x_1, x_2, ... , x_n)$$

In most general terms, we are interested in whether and how a dependent variable is related to a list of independent variables; the term *F(x...)* in the expression above means that *y*, the dependent or response variable, is a function of the *x*'s, that is, the independent variables.

An example of this type of model would be the linear multiple regression model as described in *Multiple Regression*. For this model, we assume the dependent variable to be a *linear* function of the independent variables, that is:

$$y = a + b_1*x_1 + b_2*x_2 + ... + b_n*x_n$$

If you are not familiar with multiple linear regression, you may want to read the introductory section to *Multiple Regression* at this point (however, it is not necessary to understand all of the nuances of multiple linear regression techniques in order to understand the methods discussed here).

Nonlinear Estimation allows you to specify essentially any type of continuous or discontinuous regression model. Some of the most common nonlinear models are *probit, logit*, *exponential growth*, and *breakpoint regression*. However, you can also define any type of regression equation to fit to your data. Moreover, you can specify either standard *least squares* estimation, *maximum likelihood* estimation (where appropriate), or, again, define your own "loss function" (see below) by defining the respective equation.

In general, whenever the simple linear regression model does not appear to adequately represent the relationships between variables, then the nonlinear regression model approach is appropriate. See the following topics for overviews of the common nonlinear regression models, nonlinear estimation procedures, and evaluation of the fit of the data to the nonlinear model.

Common Nonlinear Regression Models

- Intrinsically Linear Regression Models
- Intrinsically Nonlinear Regression Models

Intrinsically Linear Regression Models

Polynomial Regression. A common "nonlinear" model is polynomial regression. We put the term *nonlinear* in quotes here because the nature of this model is actually linear. For example, suppose we measure in a learning experiment subjects' physiological arousal and their performance on a complex tracking task. Based on the well-known Yerkes-Dodson law we could expect a curvilinear relationship between arousal and performance; this expectation can be expressed in the regression equation:

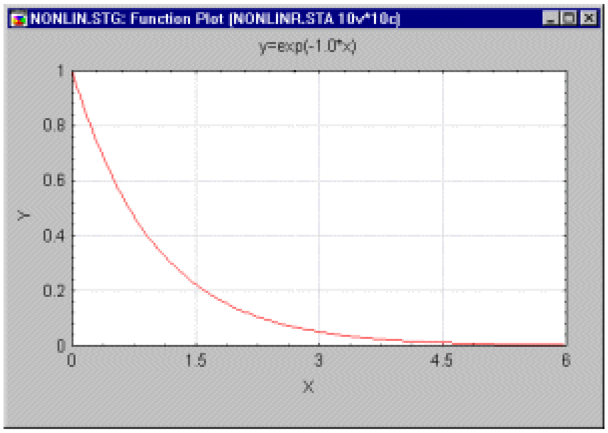
$$\text{Performance} = a + b_1 * \text{Arousal} + b_2 * \text{Arousal}^2$$

In this equation, a represents the intercept, and b_1 and b_2 are regression coefficients. The non-linearity of this model is expressed in the term $Arousal^2$. However, the *nature* of the model is still linear, except that when estimating it, we would square the measure of arousal. These types of models, where we include some transformation of the independent variables in a linear equation, are also referred to as models that are *nonlinear in the variables*.

Models that are nonlinear in the parameters. To contrast the example above, consider the relationship between a human's age from birth (the *x* variable) and his or her growth rate (the *y* variable). Clearly, the relationship between these two variables in the first year of a person's life (when most growth occurs) is very different than during adulthood (when almost no growth occurs). Thus, the relationship could probably best be expressed in terms of some negative exponential function:

Growth = exp(-b_1*Age)

If you plotted this relationship for a particular estimate of the regression coefficient you would obtain a curve that looks something like this.



Note that the *nature* of this model is no longer linear, that is, the expression shown above does not simply represent a linear regression model, with some transformation of the independent variable. This type of model is said to be *nonlinear in the parameters*.

Making nonlinear models linear. In general, whenever a regression model can be "made" into a linear model, this is the preferred route to pursue (for estimating the respective model). The linear multiple regression model (see *Multiple Regression*) is very well understood mathematically, and, from a pragmatic standpoint, is most easily interpreted. Therefore, returning to the simple exponential regression model of *Growth* as a function of *Age* shown above, we could convert this nonlinear regression equation into a linear one by simply taking the logarithm of both sides of the equations, so that:

log(Growth) = -b_1*Age

If we now substitute *log(Growth)* with *y*, we have the standard linear regression model as shown earlier (without the intercept which was ignored here to simplify matters). Thus, we could log-transform the *Growth* rate data and then use *Multiple Regression* to estimate the relationship between *Age* and *Growth*, that is, compute the regression coefficient b_1.

Model adequacy. Of course, by using the "wrong" transformation, one could end up with an inadequate model. Therefore, after "linearizing" a model such as the one shown above, it is particularly important to use extensive residual statistics in *Multiple Regression*.

Intrinsically Nonlinear Regression Models

Some regression models which cannot be transformed into linear ones, can only be estimated via *Nonlinear Estimation*. In the growth rate example above, we purposely "forgot" about the random error in the dependent variable. Of course, the growth rate is affected by very many other variables (other than time), and we can expect a considerable amount of random (*residual*) fluctuation around the fitted line. If we add this *error* or residual variability to the model, we could rewrite it as follows:

Growth = exp(-b_1*Age) + error

Additive error. In this model we assume that the error variability is independent of age, that is, that the amount of residual error variability is the same at any age. Because the error term in this model is additive, you can no longer linearize this model by taking the logarithm of both sides. If for a given data set, you were to log-transform variable *Growth* anyway and fit the simple linear model, then you would find that the residuals from the analysis would no longer be evenly distributed over the range of variable Age; and thus, the standard linear regression analysis (via *Multiple Regression*) would no longer be appropriate. Therefore, the only way to estimate the parameters for this model is via *Nonlinear Estimation*.

Multiplicative error. To "defend" our previous example, in this particular instance it is not likely that the error variability is constant at all ages, that is, that the error is additive. Most likely, there is more random and unpredictable fluctuation of the growth rate at the earlier ages than the later ages, when growth comes to a virtual standstill anyway. Thus, a more realistic model including the error would be:

Growth = exp(-b_1*Age) * error

Put in words, the greater the age, the smaller the term *exp(-b_1*Age)*, and, consequently, the smaller the resultant error variability. If we now take the log of both sides of the equation, the residual error term will become an additive factor in a linear equation, and we can go ahead and estimate b1 via standard multiple regression.

Log (Growth) = -b_1*Age + error

Let us now consider some regression models (that are nonlinear in their parameters) which cannot be "made into" linear models through simple transformations of the raw data.

General Growth Model. The general growth model, is similar to the example that we previously considered:

y = b_0 + b_1*exp(b_2*x) + error

This model is commonly used in studies of any kind of growth (*y*), when the rate of growth at any given point in time (*x*) is proportional to the amount of growth remaining. The parameter *b_0* in this model represents the maximum growth value. A typical example where this model would be adequate is when one wants to describe the concentration of a substance (e.g., in water) as a function of elapsed time.

Models for Binary Responses: Probit & Logit. It is not uncommon that a dependent or response variable is binary in nature, that is, that it can have only two possible values. For example, patients either do or do not recover from an injury; job applicants either succeed or fail at an employment test, subscribers to a journal either do or do not renew a subscription, coupons may or may not be returned, etc. In all of these cases, one may be interested in estimating a model that describes the relationship between one or more continuous independent variable(s) to the binary dependent variable.

Using linear regression. Of course, one could use standard multiple regression procedures to compute standard regression coefficients. For example, if one studied the renewal of journal subscriptions, one could create a y variable with *1*'s and *0*'s, where *1* indicates that the respective subscriber renewed, and *0* indicates that the subscriber did not renew. However, there is a problem: *Multiple Regression* does not "know" that the response variable is binary in nature. Therefore, it will inevitably fit a model that leads to predicted values that are greater than *1* or less than *0*. However, predicted values that are greater than *1* or less than *0* are not valid; thus, the restriction in the range of the binary variable (e.g., between *0* and *1*) is ignored if one uses the standard multiple regression procedure.

Continuous response functions. We could rephrase the regression problem so that, rather than predicting a binary variable, we are predicting a *continuous* variable that naturally stays within the 0-1 bounds. The two most common regression models that accomplish exactly this are the *logit* and the *probit* regression models.

Logit regression. In the logit regression model, the predicted values for the dependent variable will never be less than (or equal to) *0*, or greater than (or equal to) *1*, regardless of the values of the independent variables. This is accomplished by applying the following regression equation, which actually has some "deeper meaning" as we will see shortly (the term *logit* was first used by Berkson, 1944):

$$y = \exp(b_0 + b_1*x_1 + ... + b_n*x_n)/\{1 + \exp(b_0 + b_1*x_1 + ... + b_n*x_n)\}$$

One can easily recognize that, regardless of the regression coefficients or the magnitude of the x values, this model will always produce predicted values (predicted y's) in the range of 0 to 1.

The name *logit* stems from the fact that one can easily linearize this model via the *logit* transformation. Suppose we think of the binary dependent variable y in terms of an underlying continuous probability p, ranging from 0 to 1. We can then transform that probability p as:

$$p' = \log_e\{p/(1-p)\}$$

This transformation is referred to as the *logit* or *logistic* transformation. Note that *p'* can theoretically assume any value between minus and plus infinity. Since the logit transform solves the issue of the 0/1 boundaries for the original dependent variable (probability), we could use those (logit transformed) values in an ordinary linear regression equation. In fact, if we perform the logit transform on both sides of the logit regression equation stated earlier, we obtain the standard linear regression model:

$p' = b_0 + b_1*x_1 + b_2*x_2 + ... + b_n*x_n$

Probit regression. One may consider the binary response variable to be the result of a normally distributed underlying variable that actually ranges from minus infinity to positive infinity. For example, a subscriber to a journal can feel very strongly about not renewing a subscription, be almost undecided, "tend towards" renewing the subscription, or feel very much in favor of renewing the subscription. In any event, all that we (the publisher of the journal) will see is the binary response of renewal or failure to renew the subscription. However, if we set up the standard linear regression equation based on the underlying "feeling" or attitude we could write:

$feeling... = b_0 + b_1*x_1 + ...$

which is, of course, the standard regression model. It is reasonable to assume that these feelings are normally distributed, and that the probability *p* of renewing the subscription is about equal to the relative *space* under the normal curve. Therefore, if we transform each side of the equation so as to reflect normal probabilities, we obtain:

$NP(feeling...) = NP(b_0 + b_1*x_1 + ...)$

where *NP* stands for *normal probability* (space under the normal curve), as tabulated in practically all statistics texts. The equation shown above is also referred to as the *probit* regression model. (The term *probit* was first used by Bliss, 1934.)

General Logistic Regression Model. The general logistic model can be stated as:

$y = b_0/\{1 + b_1*exp(b_2*x)\}$

You can think of this model as an extension of the logit or logistic model for binary responses. However, while the logit model restricts the dependent response variable to only two values, this model allows the response to vary within a particular lower and upper limit. For example, suppose we are interested in the population growth of a species that is introduced to a new habitat, as a function of time. The dependent variable would be the number of individuals of that species in the respective habitat. Obviously, there is a lower limit on the dependent variable, since fewer than 0 individuals cannot exist in the habitat; however, there also is most likely an upper limit that will be reached at some point in time.

Drug Responsiveness and Half-Maximal Response. In pharmacology, the following model is often used to describe the effects of different dose levels of a drug:

$$y = b_0 - b_0/\{1 + (x/b_2)^{b}{}_{1}\}$$

In this model, x is the dose level (usually in some coded form, so that $x \geq 1$) and y is the responsiveness, in terms of the percent of maximum possible responsiveness. The parameter b_0 then denotes the expected response at the level of dose saturation and b_2 is the concentration that produces a half- maximal response; the parameter b_1 determines the slope of the function.

Discontinuous Regression Models

Piecewise linear regression. It is not uncommon that the *nature* of the relationship between one or more independent variables and a dependent variable changes over the range of the independent variables. For example, suppose we monitor the per-unit manufacturing cost of a particular product as a function of the number of units manufactured (output) per month. In general, the more units per month we produce, the lower is our per-unit cost, and this linear relationship may hold over a wide range of different levels of production output. However, it is conceivable that above a certain point, there is a discontinuity in the relationship between these two variables. For example, the per-unit cost may decrease relatively less quickly when older (less efficient) machines have to be put on-line in order to cope with the larger volume. Suppose that the older machines go on-line when the production output rises above 500 units per month; we may specify a regression model for cost-per-unit as:

$$y = b_0 + b_1*x*(x \leq 500) + b_2*x*(x > 500)$$

In this formula, y stands for the estimated per-unit cost; x is the output per month. The expressions ($x \leq 500$) and ($x > 500$) denote logical conditions that evaluate to *0* if false, and to *1* if true. Thus, this model specifies a common intercept (b_0), and a slope that is either equal to b_1 (if $x \leq 500$ is true, that is, equal to 1) or b_2 (if $x > 500$ is true, that is, equal to 1).

Instead of *specifying* the point where the discontinuity in the regression line occurs (at 500 units per months in the example above), one could also *estimate* that point. For example, one might have noticed or suspected that there is a discontinuity in the cost-per-unit at one particular point; however, one may not know where that point is. In that case, simply replace the 500 in the equation above with an additional parameter (e.g., b_3).

Breakpoint regression. One could also adjust the equation above to reflect a "jump" in the regression line. For example, imagine that, after the older machines are put on-line, the per-unit-cost jumps to a higher level, and then slowly goes down as volume continues to increase. In that case, simply specify an additional intercept (b_3), so that:

$$y = (b_0 + b_1*x)*(x \leq 500) + (b_3 + b_2*x)*(x > 500)$$

Comparing groups. The method described here to estimate different regression equations in different domains of the independent variable can also be used to distinguish between groups. For example, suppose in the example above, there are three different plants; to simplify the example, let us ignore the breakpoint for now. If we coded the three plants in a grouping variable by using the values *1*, *2*, and *3*, we could simultaneously estimate three different regression equations by specifying:

$$y = (x_p=1)*(b_{10} + b_{11}*x) + (x_p=2)*(b_{20} + b_{21}*x) + (x_p=3)*(b_{30} + b_{31}*x)$$

In this equation, x_p denotes the grouping variable containing the codes that identify each plant, b_{10}, b_{20}, and b_{30} are the three different intercepts, and b_{11}, b_{21}, and b_{31} refer to the slope parameters (regression coefficients) for each plant. One could compare the fit of the common regression model without considering the different groups (plants) with this model in order to determine which model is more appropriate.

Nonlinear Estimation Procedures

- Least Squares Estimation
- Loss Functions
- Weighted Least Squares
- Maximum Likelihood
- Maximum likelihood and probit/logit models
- Function Minimization Algorithms
- Start Values, Step Sizes, Convergence Criteria
- Penalty Functions, Constraining Parameters
- Local Minima
- Quasi-Newton Method
- Simplex Procedure
- Hooke-Jeeves Pattern Moves
- Rosenbrock Pattern Search
- Hessian Matrix and Standard Errors

Least Squares Estimation. Some of the more common nonlinear regression models are reviewed in *Common Nonlinear Regression Models*. Now, the question arises as to how these models are estimated. If you are familiar with linear regression techniques (as described in *Multiple Regression*) or analysis of variance (ANOVA) techniques (as described in *ANOVA/MANOVA*), then you may be aware of the fact that all of those methods use so-called least squares estimation procedures. In the most general terms, least squares estimation is aimed at minimizing the sum of squared deviations of the observed values for the dependent variable from those predicted by the model. (The term *least squares* was first used by Legendre, 1805.)

Loss Functions. In standard multiple regression we estimate the regression coefficients by "finding" those coefficients that minimize the residual variance (sum of squared residuals) around the regression line. Any deviation of an observed score from a predicted score signifies some *loss* in the accuracy of our prediction, for example, due to random noise (error). Therefore, we can say that the goal of least squares estimation is to minimize a *loss function*; specifically, this loss function is defined as the sum of the squared deviation about the predicted values (the term *loss* was first used by Wald, 1939). When this function is at its minimum, then we get the same parameter estimates (intercept, regression coefficients) as we would in *Multiple Regression*; because of the particular loss functions that yielded those estimates, we can call the estimates *least squares estimates*.

Phrased in this manner, there is no reason why you cannot consider other loss functions. For example, rather than minimizing the sum of *squared* deviations, why not minimize the sum of *absolute* deviations? Indeed, this is sometimes useful in order to "de-emphasize" outliers. Relative to all other residuals, a large residual will become much larger when squared. However, if one only takes the absolute value of the deviations, then the resulting regression line will most likely be less affected by outliers.

There are several function minimization methods that can be used to minimize any kind of loss function. For more information, see:

- Weighted Least Squares
- Maximum Likelihood
- Maximum likelihood and probit/logit models
- Function Minimization Algorithms
- Start Values, Step Sizes, Convergence Criteria
- Penalty Functions, Constraining Parameters
- Local Minima
- Quasi-Newton Method
- Simplex Procedure
- Hooke-Jeeves Pattern Moves
- Rosenbrock Pattern Search
- Hessian Matrix and Standard Errors

Weighted Least Squares. In addition to least squares and absolute deviation regression (see above), weighted least squares estimation is probably the most commonly used technique. Ordinary least squares techniques assume that the residual variance around the regression line is the same across all values of the independent variable(s). Put another way, it is assumed that the error variance in the measurement of each case is identical. Often, this is not a realistic assumption; in particular, violations frequently occur in business, economic, or biological applications.

For example, suppose we wanted to study the relationship between the projected cost of construction projects, and the actual cost. This may be useful in order to gage the expected cost overruns. In this case it is reasonable to assume that the absolute magnitude (dollar amount) by which the estimates are off, is proportional to the size of the project. Thus, we would use a weighted least squares loss function to fit a linear regression model. Specifically, the loss function would be (see, for example, Neter, Wasserman, & Kutner, 1985, p. 168):

$$\text{Loss} = (\text{Obs-Pred})^2 * (1/x^2)$$

In this equation, the loss function first specifies the standard least squares loss function (Observed minus *Predicted* squared; i.e., the squared residual), and then weighs this loss by the inverse of the squared value of the independent variable (x) for each case. In the actual estimation, you sum up the value of the loss function for each case (e.g., construction project), as specified above, and estimate the parameters that minimize that sum. To return to our example, the larger the project (x) the less weight is placed on the deviation from the predicted value (cost). This method will yield more stable estimates of the regression parameters (for more details, see Neter, Wasserman, & Kutner, 1985).

Maximum Likelihood. An alternative to the least squares loss function (see above) is to maximize the *likelihood* or *log*-likelihood function (or to minimize the negative log-likelihood function; the term *maximum likelihood* was first used by Fisher, 1922a). In most general terms, the likelihood function is defined as:

$$L = F(Y,\text{Model}) = \prod_{i=1}^{n} \{p\ [y_i, \text{Model Parameters}(x_i)]\}$$

In theory, we can compute the probability (now called L, the *likelihood*) of the specific dependent variable values to occur in our sample, given the respective regression model. Provided that all observations are independent of each other, this likelihood is the geometric sum ($\prod$, across $i = 1$ to n cases) of probabilities for each individual observation (i) to occur, given the respective model and parameters for the x values. (The geometric sum means that we would *multiply* out the individual probabilities across cases.) It is also customary to express this function as a natural logarithm, in which case the geometric sum becomes a regular arithmetic sum (Σ, across i = 1 to n cases).

Given the respective model, the larger the likelihood of the model, the larger is the probability of the dependent variable values to occur in the sample. Therefore, the greater the likelihood, the better is the fit of the model to the data. The actual computations for particular models here can become quite complicated because we need to "track" (compute) the probabilities of the y-values to occur (given the model and the respective x-values). As it turns out, if all assumptions for standard multiple regression are met (as described in the Multiple Regression chapter in the manual), then the standard least squares estimation method (see above) will yield results identical to the maximum likelihood method. If the assumption of equal error variances across the range of the x variable(s) is violated, then the weighted least squares method described earlier will yield maximum likelihood estimates.

Maximum Likelihood and Probit/Logit Models. The maximum likelihood function has been "worked out" for probit and logit regression models. Specifically, the loss function for these models is computed as the sum of the natural log of the logit or probit likelihood L_1 so that:

$$\log(L_1) = \Sigma_{i=1}^{n} [y_i*\log(p_i) + (1-y_i)*\log(1-p_i)]$$

where
$\log(L_1)$ is the natural log of the (logit or probit) likelihood (log-likelihood) for the current model
y_i is the observed value for case *i*
p_i is the expected (predicted or fitted) probability (between 0 and 1)

The log-likelihood of the null model (L_0), that is, the model containing the intercept only (and no regression coefficients) is computed as:

$$\log(L_0) = n_0*(\log(n_0/n)) + n_1*(\log(n_1/n))$$

where
$\log(L_0)$ is the natural log of the (logit or probit) likelihood of the null model (intercept only)
n_0 is the number of observations with a value of 0 (zero)
n_1 is the number of observations with a value of 1
n is the total number of observations

Function Minimization Algorithms. Now that we have discussed different regression models, and the loss functions that can be used to estimate them, the only "mystery" that is left is how to minimize the loss functions (to find the best fitting set of parameters), and how to estimate the standard errors of the parameter estimates. There is one very efficient algorithm (*quasi-Newton*) that approximates the second-order derivatives of the loss function to guide the search for the minimum (i.e., for the best parameter estimates, given the respective loss function). In addition, there are several more general function minimization algorithms that follow different search strategies (which do not depend on the second-order derivatives). These strategies are sometimes more effective for estimating loss functions with local minima; therefore, these methods are often particularly useful to find appropriate *start values* for the estimation via the quasi-Newton method.

In all cases, you can compute the standard errors of the parameter estimates. These standard errors are based on the second-order partial derivatives for the parameters, which are computed via finite difference approximation.

If you are not interested in how the minimization of the loss function is done, only that it *can* be done, you may skip the following paragraphs. However, you may find it useful to know a little about these procedures in case your regression model "refuses" to be fit to the data. In that case, the iterative estimation procedure will fail to converge, producing ever "stranger" (e.g., very large or very small) parameter estimates.

In the following paragraphs we will first discuss some general issues involved in unconstrained optimization, and then briefly review the methods used. For more detailed discussions of these procedures you may refer to Brent (1973), Gill and Murray (1974), Peressini, Sullivan, and Uhl (1988), and Wilde and Beightler (1967). For specific algorithms, see Dennis and Schnabel (1983), Eason and Fenton (1974), Fletcher (1969), Fletcher and Powell (1963), Fletcher and Reeves (1964), Hooke and Jeeves (1961), Jacoby, Kowalik, and Pizzo (1972), and Nelder and Mead (1964).

Start Values, Step Sizes, Convergence Criteria. A common aspect of all estimation procedures is that they require the user to specify some start values, initial step sizes, and a criterion for convergence . All methods will begin with a particular set of initial estimates (*start values*), which will be changed in some systematic manner from iteration to iteration; in the first iteration, the *step size* determines by how much the parameters will be moved. Finally, the *convergence criterion* determines when the iteration process will stop. For example, the process may stop when the improvements in the loss function from iteration to iteration are less than a specific amount.

Penalty Functions, Constraining Parameters. These estimation procedures are unconstrained in nature. When this happens, it will move parameters around without any regard for whether or not permissible values result. For example, in the course of logit regression we may get estimated values that are equal to 0.0, in which case the logarithm cannot be computed (since the log of 0 is undefined). When this happens, it will assign a *penalty* to the loss function, that is, a very large value. As a result, the various estimation procedures usually move away from the regions that produce those functions. However, in some circumstances, the estimation will "get stuck," and as a result, you would see a very large value of the loss function. This could happen, if, for example, the regression equation involves taking the logarithm of an independent variable which has a value of zero for some cases (in which case the logarithm cannot be computed).

If you wish to constrain a procedure, then this constraint must be specified in the loss function as a penalty function (assessment). By doing this, you may control what permissible values of the parameters to be estimated may be manipulated. For example, if two parameters (*a* and *b*) are to be constrained to be greater than or equal to zero, then one must assess a large penalty to these parameters if this condition is not met. Below is an example of a user-specified regression and loss function, including a penalty assessment designed to "penalize" the parameters *a* and/or *b* if either one is not greater than or equal to zero:

Estimated function: v3 = a + b*v1 + (c*v2)
Loss function: L = (obs - pred)**2 + (a<0)*100000 + (b<0)*100000

Local Minima. The most "treacherous" threat to unconstrained function minimization is *local minima*. For example, a particular loss function may become slightly larger, regardless of how a particular parameter is moved. However, if the parameter were to be moved into a completely different place, the loss function may actually become smaller. You can think of such local minima as local "valleys" or minor "dents" in the loss function. However, in most practical applications, local minima will produce "outrageous" and extremely large or small parameter estimates with very large standard errors. In those cases, specify different start values and try again. Also note, that the *Simplex* method (see below) is particularly "smart" in avoiding such minima; therefore, this method may be particularly suited in order to find appropriate start values for complex functions.

Quasi-Newton Method. As you may remember, the slope of a function at a particular point can be computed as the first- order derivative of the function (at that point). The "slope of the slope" is the second-order derivative, which tells us how fast the slope is changing at the respective point, and in which direction. The quasi-Newton method will, at each step, evaluate the function at different points in order to estimate the first-order derivatives and second-order derivatives. It will then use this information to follow a path towards the minimum of the loss function.

Simplex Procedure. This algorithm does not rely on the computation or estimation of the derivatives of the loss function. Instead, at each iteration the function will be evaluated at *m+1* points in the *m* dimensional parameter space. For example, in two dimensions (i.e., when there are two parameters to be estimated), it will evaluate the function at three points around the current optimum. These three points would define a triangle; in more than two dimensions, the "figure" produced by these points is called a *Simplex*. Intuitively, in two dimensions, three points will allow us to determine "which way to go," that is, in which direction in the two dimensional space to proceed in order to minimize the function. The same principle can be applied to the multidimensional parameter space, that is, the Simplex will "move" downhill; when the current step sizes become too "crude" to detect a clear downhill direction, (i.e., the Simplex is too large), the Simplex will "contract" and try again.

An additional strength of this method is that when a minimum appears to have been found, the Simplex will again be expanded to a larger size to see whether the respective minimum is a local minimum. Thus, in a way, the Simplex moves like a smooth single cell organism down the loss function, contracting and expanding as local minima or significant ridges are encountered.

Hooke-Jeeves Pattern Moves. In a sense this is the simplest of all algorithms. At each iteration, this method first defines a pattern of points by moving each parameter one by one, so as to optimize the current loss function. The entire pattern of points is then shifted or moved to a new location; this new location is determined by extrapolating the line from the old base point in the *m* dimensional parameter space to the new base point. The step sizes in this process are constantly adjusted to "zero in" on the respective optimum. This method is usually quite effective, and should be tried if both the quasi-Newton and Simplex methods (see above) fail to produce reasonable estimates.

Rosenbrock Pattern Search. Where all other methods fail, the Rosenbrock Pattern Search method often succeeds. This method will rotate the parameter space and align one axis with a ridge (this method is also called the *method of rotating coordinates*); all other axes will remain orthogonal to this axis. If the loss function is unimodal and has detectable ridges pointing towards the minimum of the function, then this method will proceed with sure-footed accuracy towards the minimum of the function. However, note that this search algorithm may terminate early when there are several constraint boundaries (resulting in the penalty value; see above) that intersect, leading to a discontinuity in the ridges.

Hessian Matrix and Standard Errors. The matrix of second-order (partial) derivatives is also called the *Hessian* matrix. It turns out that the inverse of the Hessian matrix approximates the variance/covariance matrix of parameter estimates. Intuitively, there *should* be an inverse relationship between the second-order derivative for a parameter and its standard error: If the change of the slope around the minimum of the function is very sharp, then the second-order derivative will be large; however, the parameter estimate will be quite stable in the sense that the minimum with respect to the parameter is clearly identifiable. If the second-order derivative is nearly zero, then the change in the slope around the minimum is zero, meaning that we can practically move the parameter in any direction without greatly affecting the loss function. Thus, the standard error of the parameter will be very large.

The Hessian matrix (and asymptotic standard errors for the parameters) can be computed via finite difference approximation. This procedure yields very precise asymptotic standard errors for all estimation methods.

Evaluating the Fit of the Model

After estimating the regression parameters, an essential aspect of the analysis is to test the appropriateness of the overall model. For example, if one specified a linear regression model, but the relationship is intrinsically non-linear, then the parameter estimates (regression coefficients) and the estimated standard errors of those estimates may be significantly "off." Let us review some of the ways to evaluate the appropriateness of a model.

- Proportion of Variance Explained
- Goodness-of-fit Chi-square
- Plot of Observed vs. Predicted Values
- Normal and Half-Normal Probability Plots
- Plot of the Fitted Function
- Variance/Covariance Matrix for Parameters

Proportion of Variance Explained. Regardless of the model, one can always compute the total variance of the dependent variable (total sum of squares, SST), the proportion of variance due to the residuals (error sum of squares, SSE), and the proportion of variance due to the regression model (regression sum of squares, SSR=SST-SSE). The ratio of the regression sum of squares to the total sum of squares (SSR/SST) explains the proportion of variance accounted for in the dependent variable (*y*) by the model; thus, this ratio is equivalent to the *R-square* (0 ≤ *R-square* ≤ 1, the *coefficient of determination*). Even when the dependent variable is not normally distributed across cases, this measure may help evaluate how well the model fits the data.

Goodness-of-fit *Chi-square*. For probit and logit regression models, you may use maximum likelihood estimation (i.e., maximize the likelihood function). As it turns out, one can directly compare the likelihood L_0 for the null model where all slope parameters are zero, with the likelihood L_1 of the fitted model. Specifically, one can compute the *Chi-square* statistic for this comparison as:

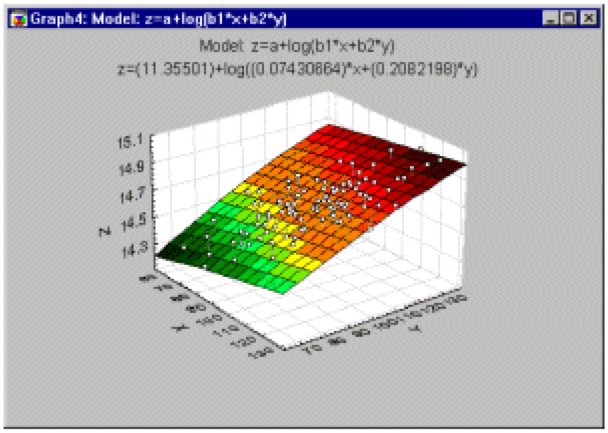
Chi-square = -2 * (log(L_0) - log(L_1))

The degrees of freedom for this *Chi-square* value are equal to the difference in the number of parameters for the null and the fitted model; thus, the degrees of freedom will be equal to the number of independent variables in the logit or probit regression. If the *p*-level associated with this *Chi-square* is significant, then we can say that the estimated model yields a significantly better fit to the data than the null model, that is, that the regression parameters are statistically significant.

Plot of Observed vs. Predicted Values. It is always a good idea to inspect a scatterplot of predicted vs. observed values. If the model is appropriate for the data, then we would expect the points to roughly follow a straight line; if the model is incorrectly specified, then this plot will indicate a non-linear pattern.

Normal and Half-Normal Probability Plots. The *normal probability plot* of residual will give us an indication of whether or not the residuals (i.e., errors) are normally distributed.

Plot of the Fitted Function. For models involving two or three variables (one or two predictors) it is useful to plot the fitted function using the final parameter estimates. Here is an example of a 3D plot with two predictor variables:


Graph4: Model: z=a+log(b1*x+b2*y)
Model: z=a+log(b1*x+b2*y)
z=(11.35501)+log((0.07430864)*x+(0.2082198)*y)
15.1
14.9
14.7
14.5
14.3
Z
X
110
120
Y
70
80
90
100

This type of plot represents the most direct visual check of whether or not a model fits the data, and whether there are apparent outliers.

Variance/Covariance Matrix for Parameters. When a model is grossly misspecified, or the estimation procedure gets "hung up" in a local minimum, the standard errors for the parameter estimates can become very large. This means that regardless of how the parameters were moved around the final values, the resulting loss function did not change much. Also, the correlations between parameters may become very large, indicating that parameters are very redundant; put another way, when the estimation algorithm moved one parameter away from the final value, then the increase in the loss function could be almost entirely compensated for by moving another parameter. Thus, the effect of those two parameters on the loss function was very redundant.

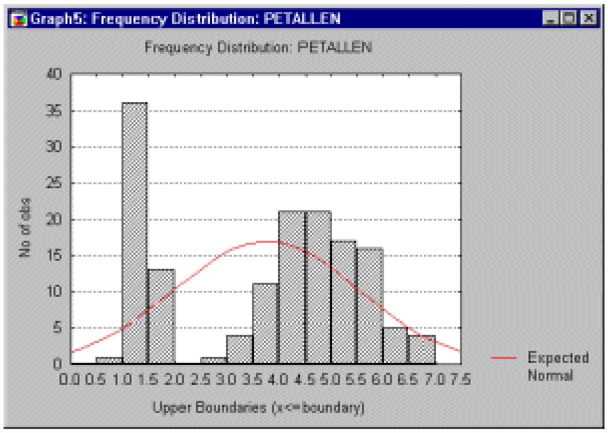
Chapter 31
Nonparametric Statistics

- General Purpose
- Brief Overview of Nonparametric Procedures
- When to Use Which Method
- Nonparametric Correlations

General Purpose

Brief review of the idea of significance testing. To understand the idea of *nonparametric* statistics (the term *nonparametric* was first used by Wolfowitz, 1942) first requires a basic understanding of parametric statistics. The *Elementary Concepts* chapter of the manual introduces the concept of statistical significance testing based on the sampling distribution of a particular statistic (you may want to review that chapter before reading on). In short, if we have a basic knowledge of the underlying distribution of a variable, then we can make predictions about how, in repeated samples of equal size, this particular statistic will "behave," that is, how it is distributed. For example, if we draw 100 random samples of 100 adults each from the general population, and compute the mean height in each sample, then the distribution of the standardized means across samples will likely approximate the normal distribution (to be precise, Student's *t* distribution with 99 degrees of freedom; see below). Now imagine that we take an additional sample in a particular city ("Tallburg") where we suspect that people are taller than the average population. If the mean height in that sample falls outside the upper 95% tail area of the *t* distribution then we conclude that, indeed, the people of Tallburg are taller than the average population.

Are most variables normally distributed? In the above example we relied on our knowledge that, in repeated samples of equal size, the standardized means (for height) will be distributed following the *t* distribution (with a particular mean and variance). However, this will only be true if in the population the variable of interest (height in our example) is normally distributed, that is, if the distribution of people of particular heights follows the normal distribution (the bell-shape distribution).



For many variables of interest, we simply do not know for sure that this is the case. For example, is income distributed normally in the population? -- probably not. The incidence rates of rare diseases are not normally distributed in the population, the number of car accidents is also not normally distributed, and neither are very many other variables in which a researcher might be interested.

For more information on the normal distribution, see Elementary Concepts; for information on tests of normality, see Normality tests.

Sample size. Another factor that often limits the applicability of tests based on the assumption that the sampling distribution is normal is the size of the sample of data available for the analysis (*sample size*; *n*). We can assume that the sampling distribution is normal even if we are not sure that the distribution of the variable in the population is normal, as long as our sample is large enough (e.g., 100 or more observations). However, if our sample is very small, then those tests can be used only if we are sure that the variable is normally distributed, and there is no way to test this assumption if the sample is small.

Problems in measurement. Applications of tests that are based on the normality assumptions are further limited by a lack of precise measurement. For example, let us consider a study where grade point average (*GPA*) is measured as the major variable of interest. Is an A average twice as good as a C average? Is the difference between a B and an A average comparable to the difference between a D and a C average? Somehow, the GPA is a crude measure of scholastic accomplishments that only allows us to establish a rank ordering of students from "good" students to "poor" students. This general measurement issue is usually discussed in statistics textbooks in terms of *types of measurement* or *scale of measurement*. Without going into too much detail, most common statistical techniques such as analysis of variance (and *t*- tests), regression, etc. assume that the underlying measurements are at least of *interval*, meaning that equally spaced intervals on the scale can be compared in a meaningful manner (e.g, *B* minus *A* is equal to *D* minus *C*). However, as in our example, this assumption is very often not tenable, and the data rather represent a *rank* ordering of observations (ordinal) rather than precise measurements.

Parametric and nonparametric methods. Hopefully, after this somewhat lengthy introduction, the need is evident for statistical procedures that allow us to process data of "low quality," from small samples, on variables about which nothing is known (concerning their distribution). Specifically, nonparametric methods were developed to be used in cases when the researcher knows nothing about the parameters of the variable of interest in the population (hence the name *nonparametric*). In more technical terms, nonparametric methods do not rely on the estimation of parameters (such as the mean or the standard deviation) describing the distribution of the variable of interest in the population. Therefore, these methods are also sometimes (and more appropriately) called *parameter-free* methods or *distribution-free* methods.

Brief Overview of Nonparametric Methods

Basically, there is at least one nonparametric equivalent for each parametric general type of test. In general, these tests fall into the following categories:

- Tests of differences between groups (independent samples);
- Tests of differences between variables (dependent samples);
- Tests of relationships between variables.

Differences between independent groups. Usually, when we have two samples that we want to compare concerning their mean value for some variable of interest, we would use the *t*-test for independent samples); nonparametric alternatives for this test are the *Wald-Wolfowitz runs test*, the *Mann-Whitney U test*, and the *Kolmogorov-Smirnov two-sample test*. If we have multiple groups, we would use analysis of variance (see *ANOVA/MANOVA*; the nonparametric equivalents to this method are the *Kruskal-Wallis analysis of ranks* and the *Median test*.

Differences between dependent groups. If we want to compare two variables measured in the same sample we would customarily use the *t-test for dependent samples* (in Basic Statistics for example, if we wanted to compare students' math skills at the beginning of the semester with their skills at the end of the semester). Nonparametric alternatives to this test are the *Sign* test and *Wilcoxon's matched pairs* test. If the variables of interest are dichotomous in nature (i.e., "pass" vs. "no pass") then *McNemar's Chi-square* test is appropriate. If there are more than two variables that were measured in the same sample, then we would customarily use repeated measures ANOVA. Nonparametric alternatives to this method are *Friedman's two-way analysis of variance* and *Cochran Q* test (if the variable was measured in terms of categories, e.g., "passed" vs. "failed"). Cochran Q is particularly useful for measuring changes in frequencies (proportions) across time.

Relationships between variables. To express a relationship between two variables one usually computes the correlation coefficient. Nonparametric equivalents to the standard correlation coefficient are *Spearman R*, *Kendall Tau*, and coefficient Gamma (see *Nonparametric correlations*). If the two variables of interest are categorical in nature (e.g., "passed" vs. "failed" by "male" vs. "female") appropriate nonparametric statistics for testing the relationship between the two variables are the *Chi-square* test, the *Phi* coefficient, and the *Fisher exact* test. In addition, a simultaneous test for relationships between multiple cases is available: *Kendall coefficient of concordance*. This test is often used for expressing inter-rater agreement among independent judges who are rating (ranking) the same stimuli.

Descriptive statistics. When one's data are not normally distributed, and the measurements at best contain rank order information, then computing the standard descriptive statistics (e.g., mean, standard deviation) is sometimes not the most informative way to summarize the data. For example, in the area of psychometrics it is well known that the *rated* intensity of a stimulus (e.g., perceived brightness of a light) is often a logarithmic function of the actual intensity of the stimulus (brightness as measured in objective units of *Lux*). In this example, the simple mean rating (sum of ratings divided by the number of stimuli) is not an adequate summary of the average actual intensity of the stimuli. (In this example, one would probably rather compute the geometric mean.) Nonparametrics and Distributions will compute a wide variety of measures of location (mean, median, mode, etc.) and dispersion (variance, average deviation, quartile range, etc.) to provide the "complete picture" of one's data.

When to Use Which Method

It is not easy to give simple advice concerning the use of nonparametric procedures. Each nonparametric procedure has its peculiar sensitivities and blind spots. For example, the Kolmogorov-Smirnov two-sample test is not only sensitive to differences in the location of distributions (for example, differences in means) but is also greatly affected by differences in their shapes. The Wilcoxon matched pairs test assumes that one can rank order the magnitude of differences in matched observations in a meaningful manner. If this is not the case, one should rather use the Sign test. In general, if the result of a study is important (e.g., does a very expensive and painful drug therapy help people get better?), then it is always advisable to run different nonparametric tests; should discrepancies in the results occur contingent upon which test is used, one should try to understand why some tests give different results. On the other hand, nonparametric statistics are less statistically powerful (sensitive) than their parametric counterparts, and if it is important to detect even small effects (e.g., is this food additive harmful to people?) one should be very careful in the choice of a test statistic.

Large data sets and nonparametric methods. Nonparametric methods are most appropriate when the sample sizes are small. When the data set is large (e.g., n > 100) it often makes little sense to use nonparametric statistics at all. The Elementary Concepts chapter of the manual briefly discusses the idea of the *central limit theorem*. In a nutshell, when the samples become very large, then the sample means will follow the normal distribution even if the respective variable is not normally distributed in the population, or is not measured very well. Thus, parametric methods, which are usually much more sensitive (i.e., have more *statistical power*) are in most cases appropriate for large samples. However, the tests of significance of many of the nonparametric statistics described here are based on asymptotic (large sample) theory; therefore, meaningful tests can often not be performed if the sample sizes become too small. Please refer to the descriptions of the specific tests to learn more about their power and efficiency.

Nonparametric Correlations

The following are three types of commonly used nonparametric correlation coefficients (Spearman R, Kendall Tau, and Gamma coefficients). Note that the chi-square statistic computed for two-way frequency tables, also provides a careful measure of a relation between the two (tabulated) variables, and unlike the correlation measures listed below, it can be used for variables that are measured on a simple nominal scale.

Spearman R. Spearman R (Siegel & Castellan, 1988) assumes that the variables under consideration were measured on at least an ordinal (rank order) scale, that is, that the individual observations can be ranked into two ordered series. Spearman R can be thought of as the regular Pearson product moment correlation coefficient, that is, in terms of proportion of variability accounted for, except that Spearman R is computed from ranks.

Kendall tau. Kendall tau is equivalent to Spearman R with regard to the underlying assumptions. It is also comparable in terms of its statistical power. However, Spearman R and Kendall tau are usually not identical in magnitude because their underlying logic as well as their computational formulas are very different. Siegel and Castellan (1988) express the relationship of the two measures in terms of the inequality:

-1 ≤ 3 * Kendall tau - 2 * Spearman R ≤ 1

More importantly, Kendall tau and Spearman R imply different interpretations: Spearman R can be thought of as the regular Pearson product moment correlation coefficient, that is, in terms of proportion of variability accounted for, except that Spearman R is computed from ranks. Kendall tau, on the other hand, represents a probability, that is, it is the difference between the probability that in the observed data the two variables are in the same order versus the probability that the two variables are in different orders.

Gamma. The Gamma statistic (Siegel & Castellan, 1988) is preferable to Spearman R or Kendall tau when the data contain many tied observations. In terms of the underlying assumptions, Gamma is equivalent to Spearman R or Kendall tau; in terms of its interpretation and computation it is more similar to Kendall tau than Spearman R. In short, Gamma is also a probability; specifically, it is computed as the difference between the probability that the rank ordering of the two variables agree minus the probability that they disagree, divided by 1 minus the probability of ties. Thus, Gamma is basically equivalent to Kendall tau, except that ties are explicitly taken into account.

Chapter 32
Partial Least Squares (PLS)

- Basic Ideas
- Computational Approach
    - Basic Model
    - NIPALS Algorithm
    - SIMPLS Algorithm
- Training and Verification (Crossvalidation) Samples
- Types of Analyses
    - Between-subject Designs
    - Distance Graphs

This chapter describes the use of partial least squares regression analysis. If you are unfamiliar with the basic methods of regression in linear models, it may be useful to first review the information on these topics in *Elementary Concepts*. The different designs discussed in this chapter are also described in the context of General Linear Models, Generalized Linear Models, and General Stepwise Regression.

Basic Ideas

Partial least squares regression is an extension of the multiple linear regression model (see, e.g., *Multiple Regression* or *General Stepwise Regression*). In its simplest form, a linear model specifies the (linear) relationship between a dependent (response) variable *Y*, and a set of predictor variables, the *X*'s, so that

$$Y = b_0 + b_1X_1 + b_2X_2 + ... + b_pX_p$$

In this equation b_0 is the regression coefficient for the intercept and the b_i values are the regression coefficients (for variables 1 through *p*) computed from the data.

So for example, one could estimate (i.e., predict) a person's weight as a function of the person's height and gender. You could use linear regression to estimate the respective regression coefficients from a sample of data, measuring height, weight, and observing the subjects' gender. For many data analysis problems, estimates of the linear relationships between variables are adequate to describe the observed data, and to make reasonable predictions for new observations (see *Multiple Regression* or *General Stepwise Regression* for additional details).

The multiple linear regression model has been extended in a number of ways to address more sophisticated data analysis problems. The multiple linear regression model serves as the basis for a number of multivariate methods such as *discriminant analysis* (i.e., the prediction of group membership from the levels of continuous predictor variables), *principal components regression* (i.e., the prediction of responses on the dependent variables from factors underlying the levels of the predictor variables), and *canonical correlation* (i.e., the prediction of factors underlying responses on the dependent variables from factors underlying the levels of the predictor variables). These multivariate methods all have two important properties in common. These methods impose restrictions such that (1) factors underlying the *Y* and *X* variables are extracted from the ***Y'Y*** and ***X'X*** matrices, respectively, and never from cross-product matrices involving both the *Y* and *X* variables, and (2) the number of prediction functions can never exceed the minimum of the number of *Y* variables and *X* variables.

Partial least squares regression extends multiple linear regression without imposing the restrictions employed by *discriminant analysis*, *principal components regression*, and *canonical correlation*. In partial least squares regression, prediction functions are represented by factors extracted from the ***Y'XX'Y*** matrix. The number of such prediction functions that can be extracted typically will exceed the maximum of the number of *Y* and *X* variables.

In short, partial least squares regression is probably the least restrictive of the various multivariate extensions of the multiple linear regression model. This flexibility allows it to be used in situations where the use of traditional multivariate methods is severely limited, such as when there are fewer observations than predictor variables. Furthermore, partial least squares regression can be used as an exploratory analysis tool to select suitable predictor variables and to identify outliers before classical linear regression.

Partial least squares regression has been used in various disciplines such as chemistry, economics, medicine, psychology, and pharmaceutical science where predictive linear modeling, especially with a large number of predictors, is necessary. Especially in chemometrics, partial least squares regression has become a standard tool for modeling linear relations between multivariate measurements (de Jong, 1993).

Computational Approach

Basic Model

As in multiple linear regression, the main purpose of partial least squares regression is to build a linear model, ***Y=XB+E***, where ***Y*** is an *n* cases by *m* variables response matrix, ***X*** is an *n* cases by *p* variables predictor (design) matrix, ***B*** is a *p* by *m* regression coefficient matrix, and $\boldsymbol{E}$ is a noise term for the model which has the same dimensions as $\boldsymbol{Y}$. Usually, the variables in $\boldsymbol{X}$ and $\boldsymbol{Y}$ are centered by subtracting their means and scaled by dividing by their standard deviations. For more information about centering and scaling in partial least squares regression, you can refer to Geladi and Kowalsky(1986).

Both principal components regression and partial least squares regression produce factor scores as linear combinations of the original predictor variables, so that there is no correlation between the factor score variables used in the predictive regression model. For example, suppose we have a data set with response variables $\boldsymbol{Y}$ (in matrix form) and a large number of predictor variables $\boldsymbol{X}$ (in matrix form), some of which are highly correlated. A regression using factor extraction for this type of data computes the factor score matrix $\boldsymbol{T}=\boldsymbol{XW}$ for an appropriate weight matrix $\boldsymbol{W}$, and then considers the linear regression model $\boldsymbol{Y}=\boldsymbol{TQ}+\boldsymbol{E}$, where $\boldsymbol{Q}$ is a matrix of regression coefficients (loadings) for $\boldsymbol{T}$, and $\boldsymbol{E}$ is an error (noise) term. Once the loadings $\boldsymbol{Q}$ are computed, the above regression model is equivalent to $\boldsymbol{Y}=\boldsymbol{XB}+\boldsymbol{E}$, where $\boldsymbol{B}=\boldsymbol{WQ}$, which can be used as a predictive regression model.

Principal components regression and partial least squares regression differ in the methods used in extracting factor scores. In short, principal components regression produces the weight matrix $\boldsymbol{W}$ reflecting the covariance structure between the predictor variables, while partial least squares regression produces the weight matrix $\boldsymbol{W}$ reflecting the covariance structure between the predictor and response variables.

For establishing the model, partial least squares regression produces a p by c weight matrix $\boldsymbol{W}$ for $\boldsymbol{X}$ such that $\boldsymbol{T}=\boldsymbol{XW}$, i.e., the columns of $\boldsymbol{W}$ are weight vectors for the $\boldsymbol{X}$ columns producing the corresponding n by c factor score matrix $\boldsymbol{T}$. These weights are computed so that each of them maximizes the covariance between responses and the corresponding factor scores. Ordinary least squares procedures for the regression of $\boldsymbol{Y}$ on $\boldsymbol{T}$ are then performed to produce $\boldsymbol{Q}$, the loadings for $\boldsymbol{Y}$ (or weights for $\boldsymbol{Y}$) such that $\boldsymbol{Y}=\boldsymbol{TQ}+\boldsymbol{E}$. Once $\boldsymbol{Q}$ is computed, we have $\boldsymbol{Y}=\boldsymbol{XB}+\boldsymbol{E}$, where $\boldsymbol{B}=\boldsymbol{WQ}$, and the prediction model is complete.

One additional matrix which is necessary for a complete description of partial least squares regression procedures is the p by c factor loading matrix $\boldsymbol{P}$ which gives a factor model $\boldsymbol{X}=\boldsymbol{TP}+\boldsymbol{F}$, where $\boldsymbol{F}$ is the unexplained part of the $\boldsymbol{X}$ scores. We now can describe the algorithms for computing partial least squares regression.

NIPALS Algorithm

The standard algorithm for computing partial least squares regression components (i.e., factors) is nonlinear iterative partial least squares (NIPALS). There are many variants of the NIPALS algorithm which normalize or do not normalize certain vectors. The following algorithm, which assumes that the X and Y variables have been transformed to have means of zero, is considered to be one of most efficient NIPALS algorithms.

For each h=1,…,c, where $\boldsymbol{A_0=X'Y}$, $\boldsymbol{M_0=X'X}$, $\boldsymbol{C_0=I}$, and c given,

1. compute $\boldsymbol{q_h}$, the dominant eigenvector of $\boldsymbol{A_h'A_h}$
2. $\boldsymbol{w_h=G_hA_hq_h}$, $\boldsymbol{w_h=w_h}/||\boldsymbol{w_h}||$, and store $\boldsymbol{w_h}$ into $\boldsymbol{W}$ as a column
3. $\boldsymbol{p_h=M_hw_h}$, $\boldsymbol{c_h=w_h'M_hw_h}$, $\boldsymbol{p_h=p_h/c_h}$, and store $\boldsymbol{p_h}$ into $\boldsymbol{P}$ as a column
4. $\boldsymbol{q_h=A_h'w_h/c_h}$, and store $\boldsymbol{q_h}$ into $\boldsymbol{Q}$ as a column
5. $\boldsymbol{A_{h+1}=A_h - c_hp_hq_h'}$ and $\boldsymbol{B_{h+1}=M_h - c_hp_hp_h'}$
6. $\boldsymbol{C_{h+1}=C_h - w_hp_h'}$

The factor scores matrix $\boldsymbol{T}$ is then computed as $\boldsymbol{T}=\boldsymbol{XW}$ and the partial least squares regression coefficients $\boldsymbol{B}$ of $\boldsymbol{Y}$ on $\boldsymbol{X}$ are computed as $\boldsymbol{B}=\boldsymbol{WQ}$.

SIMPLS Algorithm

An alternative estimation method for partial least squares regression components is the SIMPLS algorithm (de Jong, 1993), which can be described as follows.

For each h=1,…,c, where $\boldsymbol{A_0}=\boldsymbol{X'Y}$, $\boldsymbol{M_0}=\boldsymbol{X'X}$, $\boldsymbol{C_0}=\boldsymbol{I}$, and c given,

1. compute $\boldsymbol{q_h}$, the dominant eigenvector of $\boldsymbol{A_h'A_h}$
2. $\boldsymbol{w_h=A_hq_h}$, $\boldsymbol{c_h=w_h'M_hw_h}$, $\boldsymbol{w_h=w_h/sqrt(c_h)}$, and store $\boldsymbol{w_h}$ into $\boldsymbol{W}$ as a column
3. $\boldsymbol{p_h=M_hw_h}$, and store $\boldsymbol{p_h}$ into $\boldsymbol{P}$ as a column
4. $\boldsymbol{q_h=A_h'w_h}$, and store $\boldsymbol{q_h}$ into $\boldsymbol{Q}$ as a column
5. $\boldsymbol{v_h=C_hp_h}$, and $\boldsymbol{v_h=v_h}/||\boldsymbol{v_h}||$
6. $\boldsymbol{C_{h+1}=C_h - v_hv_h'}$ and $\boldsymbol{M_{h+1}=M_h - p_hp_h'}$
7. $\boldsymbol{A_{h+1}=C_hA_h}$

Similarly to NIPALS, the $\boldsymbol{T}$ of SIMPLS is computed as $\boldsymbol{T}=\boldsymbol{XW}$ and $\boldsymbol{B}$ for the regression of $\boldsymbol{Y}$ on $\boldsymbol{X}$ is computed as $\boldsymbol{B}=\boldsymbol{WQ'}$.

Training (Analysis) and Verification (Cross-Validation) Samples

A very important step when fitting models to be used for prediction of future observation is to verify (cross-validate) the results, i.e., to apply the current results to a new set of observations that was not used to compute those results (estimate the parameters). Some software programs offer very flexible methods for computing detailed predicted value and residual statistics for observations (1) that were not used in the computations for fitting the current model and have observed values for the dependent variables (the so-called *cross-validation sample*), and (2) that were not used in the computations for fitting the current model, and have missing data for the dependent variables (prediction sample).

Types of analyses

The design for an analysis can include effects for continuous as well as categorical predictor variables. Designs may include polynomials for continuous predictors (e.g., squared or cubic terms) as well as interaction effects (i.e., product terms) for continuous predictors. For categorical predictor, one can fit ANOVA-like designs, including full factorial, nested, and fractional factorial designs, etc. Designs can be incomplete (i.e., involve missing cells), and effects for categorical predictor variables can be represented using either the sigma-restricted parameterization or the overparameterized (i.e., indicator variable) representation of effects.

The topics below give complete descriptions of the types of designs that can be analyzed using partial least squares regression, as well as types of designs that can be analyzed using the general linear model.

Between-Subject Designs

- Overview
- One-way ANOVA
- Main effect ANOVA
- Factorial ANOVA
- Nested designs
- Simple regression
- Multiple regression
- Factorial regression
- Polynomial regression
- Response surface regression
- Analysis of covariance (ANCOVA)
- Separate slopes designs
- Homogeneity of slopes

Overview. The levels or values of the predictor variables in an analysis describe the differences between the *n* subjects or the *n* valid cases that are analyzed. Thus, when we speak of the between subject design (or simply the between design) for an analysis, we are referring to the nature, number, and arrangement of the predictor variables.

Concerning the nature or type of predictor variables, between designs which contain only categorical predictor variables can be called ANOVA (analysis of variance) designs, between designs which contain only continuous predictor variables can be called regression designs, and between designs which contain both categorical and continuous predictor variables can be called ANCOVA (analysis of covariance) designs. Further, continuous predictors are always considered to have fixed values, but the levels of categorical predictors can be considered to be fixed or to vary randomly. Designs which contain random categorical factors are called mixed-model designs (see the *Variance Components and Mixed Model ANOVA/ANCOVA* chapter).

Between designs may involve only a single predictor variable and therefore be described as simple (e.g., simple regression) or may employ numerous predictor variables (e.g., multiple regression).

Concerning the arrangement of predictor variables, some between designs employ only "main effect" or first-order terms for predictors, that is, the values for different predictor variables are independent and raised only to the first power. Other between designs may employ higher-order terms for predictors by raising the values for the original predictor variables to a power greater than 1 (e.g., in polynomial regression designs), or by forming products of different predictor variables (i.e., interaction terms). A common arrangement for ANOVA designs is the full-factorial design, in which every combination of levels for each of the categorical predictor variables is represented in the design. Designs with some but not all combinations of levels for each of the categorical predictor variables are aptly called fractional factorial designs. Designs with a hierarchy of combinations of levels for the different categorical predictor variables are called nested designs.

These basic distinctions about the nature, number, and arrangement of predictor variables can be used in describing a variety of different types of between designs. Some of the more common between designs can now be described.

One-Way ANOVA. A design with a single categorical predictor variable is called a one-way ANOVA design. For example, a study of 4 different fertilizers used on different individual plants could be analyzed via one-way ANOVA, with four levels for the factor *Fertilizer*.

In genera, consider a single categorical predictor variable *A* with 1 case in each of its 3 categories. Using the sigma-restricted coding of A into 2 quantitative contrast variables, the matrix ***X*** defining the between design is

$$\mathbf{X} = \begin{array}{c|ccc|} & X_0 & X_1 & X_2 \\ A_1 & 1 & 1 & 0 \\ A_2 & 1 & 0 & 1 \\ A_3 & 1 & -1 & -1 \end{array}$$

That is, cases in groups A_1, A_2, and A_3 are all assigned values of 1 on X_0 (the intercept), the case in group A_1 is assigned a value of 1 on X_1 and a value 0 on X_2, the case in group A_2 is assigned a value of 0 on X_1 and a value 1 on X_2, and the case in group A_3 is assigned a value of -1 on X_1 and a value -1 on X_2. Of course, any additional cases in any of the 3 groups would be coded similarly. If there were 1 case in group A_1, 2 cases in group A_2, and 1 case in group A_3, the ***X*** matrix would be

$$\mathbf{X} = \begin{array}{c} \\ A_{11} \\ A_{12} \\ A_{22} \\ A_{13} \end{array} \begin{array}{ccc} X_0 & X_1 & X_2 \\ \left[\begin{matrix} 1 \\ 1 \\ 1 \\ 1 \end{matrix}\right. & \begin{matrix} 1 \\ 0 \\ 0 \\ -1 \end{matrix} & \left.\begin{matrix} 0 \\ 1 \\ 1 \\ -1 \end{matrix}\right] \end{array}$$

where the first subscript for *A* gives the replicate number for the cases in each group. For brevity, replicates usually are not shown when describing ANOVA design matrices.

Note that in one-way designs with an equal number of cases in each group, sigma-restricted coding yields $X_1 \ldots X_k$ variables all of which have means of 0.

Using the overparameterized model to represent A, the ***X*** matrix defining the between design is simply

$$\mathbf{X} = \begin{array}{c} \\ A_1 \\ A_2 \\ A_3 \end{array} \begin{array}{cccc} X_0 & X_1 & X_2 & X_3 \\ \left[\begin{matrix} 1 \\ 1 \\ 1 \end{matrix}\right. & \begin{matrix} 1 \\ 0 \\ 0 \end{matrix} & \begin{matrix} 0 \\ 1 \\ 0 \end{matrix} & \left.\begin{matrix} 0 \\ 0 \\ 1 \end{matrix}\right] \end{array}$$

These simple examples show that the ***X*** matrix actually serves two purposes. It specifies (1) the coding for the levels of the original predictor variables on the *X* variables used in the analysis as well as (2) the nature, number, and arrangement of the X variables, that is, the between design.

Main Effect ANOVA. Main effect ANOVA designs contain separate one-way ANOVA designs for 2 or more categorical predictors. A good example of main effect ANOVA would be the typical analysis performed on *screening designs* as described in the context of the *Experimental Design* chapter.

Consider 2 categorical predictor variables *A* and *B* each with 2 categories. Using the sigma-restricted coding, the ***X*** matrix defining the between design is

$$\mathbf{X} = \begin{array}{c} \\ A_1B_1 \\ A_1B_2 \\ A_2B_1 \\ A_2B_2 \end{array} \begin{array}{ccc} X_0 & X_1 & X_2 \\ \left[\begin{matrix} 1 \\ 1 \\ 1 \\ 1 \end{matrix}\right. & \begin{matrix} 1 \\ 1 \\ -1 \\ -1 \end{matrix} & \left.\begin{matrix} 1 \\ -1 \\ 1 \\ -1 \end{matrix}\right] \end{array}$$

Note that if there are equal numbers of cases in each group, the sum of the cross-products of values for the X_1 and X_2 columns is 0, for example, with 1 case in each group (1*1)+(1*-1)+(-1*1)+(-1*-1)=0. Using the overparameterized model, the matrix ***X*** defining the between design is

$$\mathbf{X} = \begin{array}{c|ccccc|} & X_0 & X_1 & X_2 & X_3 & X_4 \\ \hline A_1B_1 & 1 & 1 & 0 & 1 & 0 \\ A_1B_2 & 1 & 1 & 0 & 0 & 1 \\ A_2B_1 & 1 & 0 & 1 & 1 & 0 \\ A_2B_2 & 1 & 0 & 1 & 0 & 1 \end{array}$$

Comparing the two types of coding, it can be seen that the overparameterized coding takes almost twice as many values as the sigma-restricted coding to convey the same information.

Factorial ANOVA. Factorial ANOVA designs contain *X* variables representing combinations of the levels of 2 or more categorical predictors (e.g., a study of boys and girls in four age groups, resulting in a *2 (Gender) x 4 (Age Group)* design). In particular, full-factorial designs represent all possible combinations of the levels of the categorical predictors. A full-factorial design with 2 categorical predictor variables *A* and *B* each with 2 levels each would be called a 2 x 2 full-factorial design. Using the sigma-restricted coding, the ***X*** matrix for this design would be

$$\mathbf{X} = \begin{array}{c|cccc|} & X_0 & X_1 & X_2 & X_3 \\ \hline A_1B_1 & 1 & 1 & 1 & 1 \\ A_1B_2 & 1 & 1 & -1 & -1 \\ A_2B_1 & 1 & -1 & 1 & -1 \\ A_2B_2 & 1 & -1 & -1 & 1 \end{array}$$

Several features of this ***X*** matrix deserve comment. Note that the X_1 and X_2 columns represent main effect contrasts for one variable, (i.e., *A* and *B*, respectively) collapsing across the levels of the other variable. The X_3 column instead represents a contrast between different combinations of the levels of *A* and *B*. Note also that the values for X_3 are products of the corresponding values for X_1 and X_2. Product variables such as X_3 represent the multiplicative or interaction effects of their factors, so X_3 would be said to represent the 2-way interaction of *A* and *B*. The relationship of such product variables to the dependent variables indicate the interactive influences of the factors on responses above and beyond their independent (i.e., main effect) influences on responses. Thus, factorial designs provide more information about the relationships between categorical predictor variables and responses on the dependent variables than is provided by corresponding one-way or main effect designs.

When many factors are being investigated, however, full-factorial designs sometimes require more data than reasonably can be collected to represent all possible combinations of levels of the factors, and high-order interactions between many factors can become difficult to interpret. With many factors, a useful alternative to the full-factorial design is the fractional factorial design. As an example, consider a 2 x 2 x 2 fractional factorial design to degree 2 with 3 categorical predictor variables each with 2 levels. The design would include the main effects for each variable, and all 2-way interactions between the three variables, but would not include the 3-way interaction between all three variables. Using the overparameterized model, the *X* matrix for this design is

$$\mathbf{X} = \begin{array}{cc} & \begin{array}{cccccccccccccccccc} & \multicolumn{6}{c}{\text{......main effects......}} & \multicolumn{12}{c}{\text{..............2 - way interactio ns..........}} \end{array} \\ \begin{array}{c} A_1B_1C_1 \\ A_1B_1C_2 \\ A_1B_2C_1 \\ A_1B_2C_2 \\ A_2B_1C_1 \\ A_2B_1C_2 \\ A_2B_2C_1 \\ A_2B_2C_2 \end{array} & \left[\begin{array}{ccccccccccccccccccc} 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right] \end{array}$$

The 2-way interactions are the highest degree effects included in the design. These types of designs are discussed in detail the *2**(k-p) Fractional Factorial Designs* section of the *Experimental Design* chapter.

Nested ANOVA Designs. Nested designs are similar to fractional factorial designs in that all possible combinations of the levels of the categorical predictor variables are not represented in the design. In nested designs, however, the omitted effects are lower-order effects. Nested effects are effects in which the nested variables never appear as main effects. Suppose that for 2 variables *A* and *B* with 3 and 2 levels, respectively, the design includes the main effect for *A* and the effect of *B* nested within the levels of *A*. The ***X*** matrix for this design using the overparameterized model is

$$\mathbf{X} = \begin{array}{cc} & \begin{array}{cccccccccc} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 & X_8 & X_9 \end{array} \\ \begin{array}{c} A_1B_1 \\ A_1B_2 \\ A_2B_1 \\ A_2B_2 \\ A_3B_1 \\ A_3B_2 \end{array} & \left[\begin{array}{cccccccccc} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \end{array}$$

Note that if the sigma-restricted coding were used, there would be only 2 columns in the ***X*** matrix for the *B* nested within *A* effect instead of the 6 columns in the ***X*** matrix for this effect when the overparameterized model coding is used (i.e., columns X_4 through X_9). The sigma-restricted coding method is overly-restrictive for nested designs, so only the overparameterized model is used to represent nested designs.

Simple Regression. Simple regression designs involve a single continuous predictor variable. If there were 3 cases with values on a predictor variable *P* of, say, 7, 4, and 9, and the design is for the first-order effect of *P*, the ***X*** matrix would be

$$\mathbf{X} = \begin{array}{c} \begin{array}{cc} X_0 & X_1 \end{array} \\ \begin{bmatrix} 1 & 7 \\ 1 & 4 \\ 1 & 9 \end{bmatrix} \end{array}$$

and using *P* for X_1 the regression equation would be

$$Y = b_0 + b_1P$$

If the simple regression design is for a higher-order effect of *P*, say the quadratic effect, the values in the X_1 column of the design matrix would be raised to the 2nd power, that is, squared

$$\mathbf{X} = \begin{array}{c} \begin{array}{cc} X_0 & X_1 \end{array} \\ \begin{bmatrix} 1 & 49 \\ 1 & 16 \\ 1 & 81 \end{bmatrix} \end{array}$$

and using P^2 for X_1 the regression equation would be

$$Y = b_0 + b_1P^2$$

The sigma-restricted and overparameterized coding methods do not apply to simple regression designs and any other design containing only continuous predictors (since there are no categorical predictors to code). Regardless of which coding method is chosen, values on the continuous predictor variables are raised to the desired power and used as the values for the *X* variables. No recoding is performed. It is therefore sufficient, in describing regression designs, to simply describe the regression equation without explicitly describing the design matrix ***X***.

Multiple Regression. Multiple regression designs are to continuous predictor variables as main effect ANOVA designs are to categorical predictor variables, that is, multiple regression designs contain the separate simple regression designs for 2 or more continuous predictor variables. The regression equation for a multiple regression design for the first-order effects of 3 continuous predictor variables *P*, *Q*, and *R* would be

$$Y = b_0 + b_1P + b_2Q + b_3R$$

Factorial Regression. Factorial regression designs are similar to factorial ANOVA designs, in which combinations of the levels of the factors are represented in the design. In factorial regression designs, however, there may be many more such possible combinations of distinct levels for the continuous predictor variables than there are cases in the data set. To simplify matters, full-factorial regression designs are defined as designs in which all possible products of the continuous predictor variables are represented in the design. For example, the full-factorial regression design for two continuous predictor variables *P* and *Q* would include the main effects (i.e., the first-order effects) of *P* and *Q* and their 2-way *P* by *Q* interaction effect, which is represented by the product of *P* and *Q* scores for each case. The regression equation would be

$$Y = b_0 + b_1P + b_2Q + b_3P*Q$$

Factorial regression designs can also be fractional, that is, higher-order effects can be omitted from the design. A fractional factorial design to degree 2 for 3 continuous predictor variables *P*, *Q*, and *R* would include the main effects and all 2-way interactions between the predictor variables

$$Y = b_0 + b_1P + b_2Q + b_3R + b_4P*Q + b_5P*R + b_6Q*R$$

Polynomial Regression. Polynomial regression designs are designs which contain main effects and higher-order effects for the continuous predictor variables but do not include interaction effects between predictor variables. For example, the polynomial regression design to degree 2 for three continuous predictor variables *P, Q,* and *R* would include the main effects (i.e., the first-order effects) of *P, Q,* and *R* and their quadratic (i.e., second-order) effects, but not the 2-way interaction effects or the *P* by *Q* by *R* 3-way interaction effect.

$$Y = b_0 + b_1P + b_2P^2 + b_3Q + b_4Q^2 + b_5R + b_6R^2$$

Polynomial regression designs do not have to contain all effects up to the same degree for every predictor variable. For example, main, quadratic, and cubic effects could be included in the design for some predictor variables, and effects up the fourth degree could be included in the design for other predictor variables.

Response Surface Regression. Quadratic response surface regression designs are a hybrid type of design with characteristics of both polynomial regression designs and fractional factorial regression designs. Quadratic response surface regression designs contain all the same effects of polynomial regression designs to degree 2 and additionally the 2-way interaction effects of the predictor variables. The regression equation for a quadratic response surface regression design for 3 continuous predictor variables *P, Q,* and *R* would be

$$Y = b_0 + b_1P + b_2P^2 + b_3Q + b_4Q^2 + b_5R + b_6R^2 + b_7P*Q + b_8P*R + b_9Q*R$$

These types of designs are commonly employed in applied research (e.g., in industrial experimentation), and a detailed discussion of these types of designs is also presented in the *Experimental Design* chapter (see *Central composite designs*).

Analysis of Covariance. In general, between designs which contain both categorical and continuous predictor variables can be called ANCOVA designs. Traditionally, however, ANCOVA designs have referred more specifically to designs in which the first-order effects of one or more continuous predictor variables are taken into account when assessing the effects of one or more categorical predictor variables. A basic introduction to analysis of covariance can also be found in the *Analysis of covariance (ANCOVA)* topic of the *ANOVA/MANOVA* chapter.

To illustrate, suppose a researcher wants to assess the influences of a categorical predictor variable *A* with 3 levels on some outcome, and that measurements on a continuous predictor variable *P*, known to covary with the outcome, are available. If the data for the analysis are

$$\begin{array}{cc} P & \text{Group} \\ \begin{bmatrix} 7 \\ 4 \\ 9 \\ 3 \\ 6 \\ 8 \end{bmatrix} & \begin{bmatrix} A_1 \\ A_1 \\ A_2 \\ A_2 \\ A_3 \\ A_3 \end{bmatrix} \end{array}$$

then the sigma-restricted ***X*** matrix for the design that includes the separate first-order effects of *P* and *A* would be

$$\mathbf{X} = \begin{array}{cccc} X_0 & X_1 & X_2 & X_3 \\ \hline 1 & 7 & 1 & 0 \\ 1 & 4 & 1 & 0 \\ 1 & 9 & 0 & 1 \\ 1 & 3 & 0 & 1 \\ 1 & 6 & -1 & -1 \\ 1 & 8 & -1 & -1 \end{array}$$

The b_2 and b_3 coefficients in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3$$

represent the influences of group membership on the *A* categorical predictor variable, controlling for the influence of scores on the *P* continuous predictor variable. Similarly, the b_1 coefficient represents the influence of scores on *P* controlling for the influences of group membership on *A*. This traditional ANCOVA analysis gives a more sensitive test of the influence of *A* to the extent that *P* reduces the prediction error, that is, the residuals for the outcome variable.

The ***X*** matrix for the same design using the overparameterized model would be

$$\mathbf{X} = \begin{array}{c} \begin{matrix} X_0 & X_1 & X_2 & X_3 & X_4 \end{matrix} \\ \begin{bmatrix} 1 & 7 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ 1 & 9 & 0 & 1 & 0 \\ 1 & 3 & 0 & 1 & 0 \\ 1 & 6 & 0 & 0 & 1 \\ 1 & 8 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

The interpretation is unchanged except that the influences of group membership on the *A* categorical predictor variables are represented by the b_2, b_3 and b_4 coefficients in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4$$

Separate Slope Designs. The traditional analysis of covariance (ANCOVA) design for categorical and continuous predictor variables is inappropriate when the categorical and continuous predictors interact in influencing responses on the outcome. The appropriate design for modeling the influences of the predictors in this situation is called the separate slope design. For the same example data used to illustrate traditional ANCOVA, the overparameterized ***X*** matrix for the design that includes the main effect of the three-level categorical predictor *A* and the 2-way interaction of *P* by *A* would be

$$\mathbf{X} = \begin{array}{c} \begin{matrix} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 \end{matrix} \\ \begin{bmatrix} 1 & 1 & 0 & 0 & 7 & 0 & 0 \\ 1 & 1 & 0 & 0 & 4 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 9 & 0 \\ 1 & 0 & 1 & 0 & 0 & 3 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 6 \\ 1 & 0 & 0 & 1 & 0 & 0 & 8 \end{bmatrix} \end{array}$$

The b_4, b_5, and b_6 coefficients in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + b_5X_5 + b_6X_6$$

give the separate slopes for the regression of the outcome on *P* within each group on *A*, controlling for the main effect of *A*.

As with nested ANOVA designs, the sigma-restricted coding of effects for separate slope designs is overly restrictive, so only the overparameterized model is used to represent separate slope designs. In fact, separate slope designs are identical in form to nested ANOVA designs, since the main effects for continuous predictors are omitted in separate slope designs.

Homogeneity of Slopes. The appropriate design for modeling the influences of continuous and categorical predictor variables depends on whether the continuous and categorical predictors interact in influencing the outcome. The traditional analysis of covariance (ANCOVA) design for continuous and categorical predictor variables is appropriate when the continuous and categorical predictors do not interact in influencing responses on the outcome, and the separate slope design is appropriate when the continuous and categorical predictors do interact in influencing responses. The homogeneity of slopes designs can be used to test whether the continuous and categorical predictors interact in influencing responses, and thus, whether the traditional ANCOVA design or the separate slope design is appropriate for modeling the effects of the predictors. For the same example data used to illustrate the traditional ANCOVA and separate slope designs, the overparameterized ***X*** matrix for the design that includes the main effect of *P*, the main effect of the three-level categorical predictor *A*, and the 2-way interaction of *P* by *A* would be

$$\mathbf{X} = \begin{array}{cccccccc} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 \\ \end{array}$$
$$\mathbf{X} = \left[\begin{array}{cccccccc} 1 & 7 & 1 & 0 & 0 & 7 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 & 4 & 0 & 0 \\ 1 & 9 & 0 & 1 & 0 & 0 & 9 & 0 \\ 1 & 3 & 0 & 1 & 0 & 0 & 3 & 0 \\ 1 & 6 & 0 & 0 & 1 & 0 & 0 & 6 \\ 1 & 8 & 0 & 0 & 1 & 0 & 0 & 8 \end{array}\right]$$

If the b_5, b_6, or b_7 coefficient in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + b_5X_5 + b_6X_6 + b_7X_7$$

is non-zero, the separate slope model should be used. If instead all 3 of these regression coefficients are zero the traditional ANCOVA design should be used.

The sigma-restricted ***X*** matrix for the homogeneity of slopes design would be

$$\mathbf{X} = \left[\begin{array}{cccccc} X_0 & X_1 & X_2 & X_3 & X_4 & X_5 \\ 1 & 7 & 1 & 0 & 7 & 0 \\ 1 & 4 & 1 & 0 & 4 & 0 \\ 1 & 9 & 0 & 1 & 0 & 9 \\ 1 & 3 & 0 & 1 & 0 & 3 \\ 1 & 6 & -1 & -1 & -6 & -6 \\ 1 & 8 & -1 & -1 & -8 & -8 \end{array}\right]$$

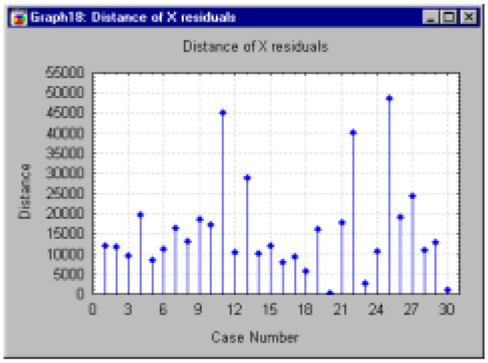
Using this ***X*** matrix, if the b_4, or b_5 coefficient in the regression equation

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_4 + b_5X_5$$

is non-zero, the separate slope model should be used. If instead both of these regression coefficients are zero the traditional ANCOVA design should be used.

Distance Graphs

A graphic technique that is useful in analyzing Partial Least Squares designs is a distance graph. These graphs allow you to compute and plot distances from the origin (zero for all dimensions) for the predicted and residual statistics, loadings, and weights for the respective number of components.



Based on Euclidean distances, these observation plots can be helpful in determining major contributors to the prediction of the conceptual variable(s) (plotting weights) as well as outliers that have a disproportionate influence (relative to the other observation) on the results (plotting residual values).

Chapter 33
Power Analysis

- General Purpose
- Power Analysis and Sample Size Calculation in Experimental Design
    - Sampling Theory
    - Hypothesis Testing Logic
    - Calculating Power
    - Calculating Required Sample Size
    - Graphical Approaches to Power Analysis
- Noncentrality Interval Estimation and the Evaluation of Statistical Models
    - Inadequacies of the Hypothesis Testing Approach
    - Advantages of Interval Estimation
    - Reasons Why Interval Estimates are Seldom Reported
    - Replacing Traditional Hypothesis Tests with Interval Estimates

General Purpose

The techniques of *statistical power analysis*, *sample size estimation*, and advanced techniques for *confidence interval estimation* are discussed here. The main goal of first the two techniques is to allow you to decide, while in the process of designing an experiment, (a) how large a sample is needed to enable statistical judgments that are accurate and reliable and (b) how likely your statistical test will be to detect effects of a given size in a particular situation. The third technique is useful in implementing objectives *a* and *b* and in evaluating the size of experimental effects in practice.

Performing power analysis and sample size estimation is an important aspect of experimental design, because without these calculations, sample size may be too high or too low. If sample size is too low, the experiment will lack the precision to provide reliable answers to the questions it is investigating. If sample size is too large, time and resources will be wasted, often for minimal gain.

In some power analysis software programs, a number of graphical and analytical tools are available to enable precise evaluation of the factors affecting power and sample size in many of the most commonly encountered statistical analyses. This information can be crucial to the design of a study that is cost-effective and scientifically useful.

Noncentrality interval estimation procedures and other sophisticated confidence interval procedures provide some sophisticated confidence interval methods for analyzing the importance of an observed experimental result. An increasing number of influential statisticians are suggesting that confidence interval estimation should augment or replace traditional hypothesis testing approaches in the analysis of experimental data.

Power Analysis and Sample Size Calculation in Experimental Design

There is a growing recognition of the importance of power analysis and sample size calculation in the proper design of experiments. Click on the links below for a discussion of the fundamental ideas behind these methods.

- Sampling Theory
- Hypothesis Testing Logic
- Calculating Power
- Calculating Required Sample Size
- Graphical Approaches to Power Analysis

Sampling Theory.

In most situations in statistical analysis, we do not have access to an entire statistical population of interest, either because the population is too large, is not willing to be measured, or the measurement process is too expensive or time-consuming to allow more than a small segment of the population to be observed. As a result, we often make important decisions about a statistical population on the basis of a relatively small amount of sample data.

Typically, we take a sample and compute a quantity called a *statistic* in order to estimate some characteristic of a population called a *parameter*.

For example, suppose a politician is interested in the proportion of people who currently favor her position on a particular issue. Her constituency is a large city with a population of about 1,500,000 potential voters. In this case, the *parameter* of interest, which we might call π, is the proportion of people *in the entire population* who favor the politician's position. The politician is going to commission an opinion poll, in which a (hopefully) random sample of people will be asked whether or not they favor her position. The number (call it N) of people to be polled will be quite small, relative to the size of the population. Once these people have been polled, the proportion of them favoring the politician's position will be computed. This proportion, which is a *statistic*, can be called p.

One thing is virtually certain before the study is ever performed: p will *not be equal to* π ! Because p involves "the luck of the draw," it will deviate from π. The amount by which p is wrong, i.e., the amount by which it deviates from π, is called *sampling error.*

In any one sample, it is virtually certain there will be *some* sampling error (except in some highly unusual circumstances), and that we will never be certain exactly how large this error is. If we knew the amount of the sampling error, this would imply that we also knew the exact value of the parameter, in which case we would not need to be doing the opinion poll in the first place.

In general, the larger the sample size *N*, the smaller sampling error *tends* to be. (One can never be sure what will happen in a particular experiment, of course.) If we are to make accurate decisions about a parameter like π, we need to have an *N* large enough so that sampling error will tend to be "reasonably small." If *N* is too small, there is not much point in gathering the data, because the results will tend to be too imprecise to be of much use.

On the other hand, there is also a point of diminishing returns beyond which increasing *N* provides little benefit. Once *N* is "large enough" to produce a reasonable level of accuracy, making it larger simply wastes time and money.

So some key decisions in planning any experiment are, "How precise will my parameter estimates tend to be if I select a particular sample size?" and "How big a sample do I need to attain a desirable level of precision?"

The purpose of Power Analysis and Sample Size Estimation is to provide you with the statistical methods to answer these questions quickly, easily, and accurately. A good statistical software program will provide simple dialogs for performing power calculations and sample size estimation for many of the classic statistical procedures as well as special *noncentral distribution* routines to allow the advanced user to perform a variety of additional calculations.

Hypothesis Testing.

Suppose that the politician was interested in showing that more than the majority of people supported her position. Her question, in statistical terms: "Is $\pi > .50$?" Being an optimist, she believes that it is.

In statistics, the following strategy is quite common. State as a "statistical null hypothesis" something that is the logical opposite of what you believe. Call this hypothesis *H0*. Gather data. Then, using statistical theory, show from the data that it is likely *H0* is false, and should be rejected.

By rejecting *H0*, you support what you actually believe. This kind of situation, which is typical in many fields of research, for example, is called "Reject-Support testing," (RS testing) because *rejecting* the null hypothesis *supports* the experimenter's theory.

The null hypothesis is either true or false, and the statistical decision process is set up so that there are no "ties." The null hypothesis is either rejected or not rejected. Consequently, before undertaking the experiment, we can be certain that only 4 possible things can happen. These are summarized in the table below

		State of the World	
		HO	H1
Decision	H0	Correct Acceptance	Type II Error β
	H1	Type I Error	Correct

		α	Rejection

Note that there are two kinds of errors represented in the table. Many statistics textbooks present a point of view that is common in the social sciences, i.e., that α, the Type I error rate, must be kept at or below .05, and that, if at all possible, β, the Type II error rate, must be kept low as well. "Statistical power," which is equal to 1 - β, must be kept correspondingly high. Ideally, power should be at least .80 to detect a reasonable departure from the null hypothesis.

The conventions are, of course, much more rigid with respect to α than with respect to β. For example, in the social sciences seldom, if ever, is α allowed to stray above the magical .05 mark.

Significance Testing (RS/AS). In the context of significance testing, we can define two basic kinds of situations, reject-support (RS) (discussed above) and accept-support (AS). In RS testing, *the null hypothesis is the opposite of what the researcher actually believes*, and rejecting it supports the researcher's theory. In a two group RS experiment involving comparison of the means of an experimental and control group, the experimenter believes the treatment has an effect, and seeks to confirm it through a significance test that rejects the null hypothesis.

In the RS situation, a Type I error represents, in a sense, a "false positive" for the researcher's theory. From society's standpoint, such false positives are particularly undesirable. They result in much wasted effort, especially when the false positive is interesting from a theoretical or political standpoint (or both), and as a result stimulates a substantial amount of research. Such follow-up research will usually not replicate the (incorrect) original work, and much confusion and frustration will result.

In RS testing, a Type II error is a tragedy from the researcher's standpoint, because a theory that is true is, by mistake, not confirmed. So, for example, if a drug designed to improve a medical condition is found (incorrectly) not to produce an improvement relative to a control group, a worthwhile therapy will be lost, at least temporarily, and an experimenter's worthwhile idea will be discounted.

As a consequence, in RS testing, society, in the person of journal editors and reviewers, insists on keeping α low. The statistically well-informed researcher makes it a top priority to keep β low as well. Ultimately, of course, everyone benefits if *both* error probabilities are kept low, but unfortunately there is often, in practice, a trade-off between the two types of error.

The RS situation is by far the more common one, and the conventions relevant to it have come to dominate popular views on statistical testing. As a result, the prevailing views on error rates are that relaxing α beyond a certain level is unthinkable, and that it is up to the researcher to make sure *statistical power* is adequate. One might argue how appropriate these views are in the context of RS testing, but they are not altogether unreasonable.

In AS testing, the common view on error rates we described above is clearly inappropriate. In AS testing, *H0 is what the researcher actually believes*, so *accepting* it *supports* the researcher's theory. In this case, a Type I error is a false negative for the researcher's theory, and a Type II error constitutes a false positive. Consequently, acting in a way that might be construed as highly *virtuous* in the RS situation, for example, maintaining a very low Type I error rate like .001, is actually "stacking the deck" in favor of the researcher's theory in AS testing.

In both AS and RS situations, it is easy to find examples where significance testing seems strained and unrealistic. Consider first the RS situation. In some such situations, it is simply not possible to have very large samples. An example that comes to mind is social or clinical psychological field research. Researchers in these fields sometimes spend several days interviewing a single subject. A year's research may only yield valid data from 50 subjects. Correlational tests, in particular, have very low power when samples are that small. In such a case, it probably makes sense to relax α beyond .05, if it means that reasonable power can be achieved.

On the other hand, it is possible, in an important sense, to have power that is too high. For example, one might be testing the hypothesis that two population means are equal (i.e., Mu1 = Mu2) with sample sizes of a million in each group. In this case, even with trivial differences between groups, the null hypothesis would virtually always be rejected.

The situation becomes even more unnatural in AS testing. Here, if *N* is too high, the researcher almost inevitably decides against the theory, even when it turns out, in an important sense, to be an excellent approximation to the data. It seems paradoxical indeed that in this context experimental precision seems to work against the researcher.

To summarize:

In Reject-Support research:

1. The researcher wants to reject *H0*.
2. Society wants to control Type I error.
3. The researcher must be very concerned about Type II error.
4. High sample size works for the researcher.
5. If there is "too much power," trivial effects become "highly significant."

In Accept-Support research:

1. The researcher wants to accept *H0*.
2. "Society" should be worrying about controlling Type II error, although it sometimes gets confused and retains the conventions applicable to RS testing.
3. The researcher must be very careful to control Type I error.
4. High sample size works against the researcher.
5. If there is "too much power," the researcher's theory can be "rejected" by a significance test even though it fits the data almost perfectly.

Calculating Power. Properly designed experiments must ensure that power will be reasonably high to detect reasonable departures from the null hypothesis. Otherwise, an experiment is hardly worth doing. Elementary textbooks contain detailed discussions of the factors influencing power in a statistical test. These include

1. What kind of statistical test is being performed. Some statistical tests are inherently more powerful than others.
2. Sample size. In general, the larger the sample size, the larger the power. However, generally increasing sample size involves tangible costs, both in time, money, and effort. Consequently, it is important to make sample size "large enough," but not wastefully large.
3. The size of experimental effects. If the null hypothesis is wrong by a substantial amount, power will be higher than if it is wrong by a small amount.
4. The level of error in experimental measurements. Measurement error acts like "noise" that can bury the "signal" of real experimental effects. Consequently, anything that enhances the accuracy and consistency of measurement can increase *statistical power*.

Calculating Required Sample Size To ensure a statistical test will have adequate power, one usually must perform special analyses prior to running the experiment, to calculate how large an *N* is required.

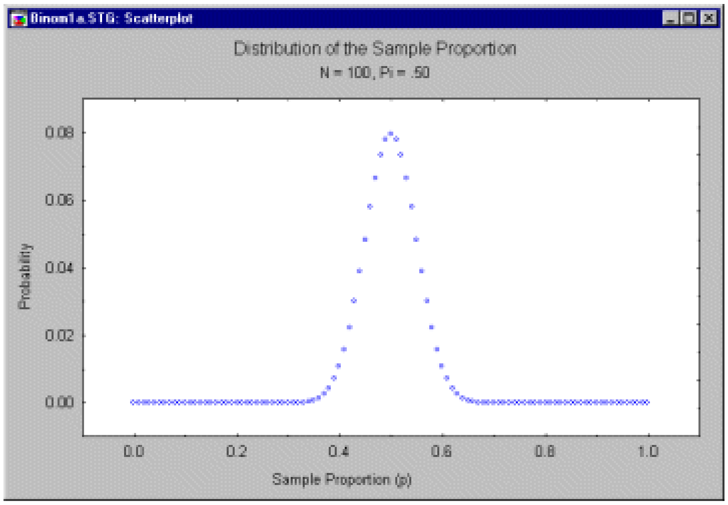
Let's briefly examine the kind of statistical theory that lies at the foundation of the calculations used to estimate power and sample size. Return to the original example of the politician, contemplating how large an opinion poll should be taken to suit her purposes.

Statistical theory, of course, cannot tell us what *will* happen with any particular opinion poll. However, through the concept of a *sampling distribution*, it can tell us *what will tend to happen in the long run,* over many opinion polls of a particular size.

A sampling distribution is the distribution of a statistic over repeated samples. Consider the sample proportion *p* resulting from an opinion poll of size *N*, in the situation where the population proportion π is exactly .50. Sampling distribution theory tells us that *p* will have a distribution that can be calculated from the binomial theorem. This distribution, for reasonably large *N*, and for values of *p* not too close to 0 or 1, looks very much like a normal distribution with a mean of π and a standard deviation (called the "standard error of the proportion") of

$\square_p = (\square(1-\square)/N)**1/2$

Suppose, for example, the politician takes an opinion poll based on an *N* of 100. Then the distribution of *p*, over repeated samples, will look like this if π = .5.

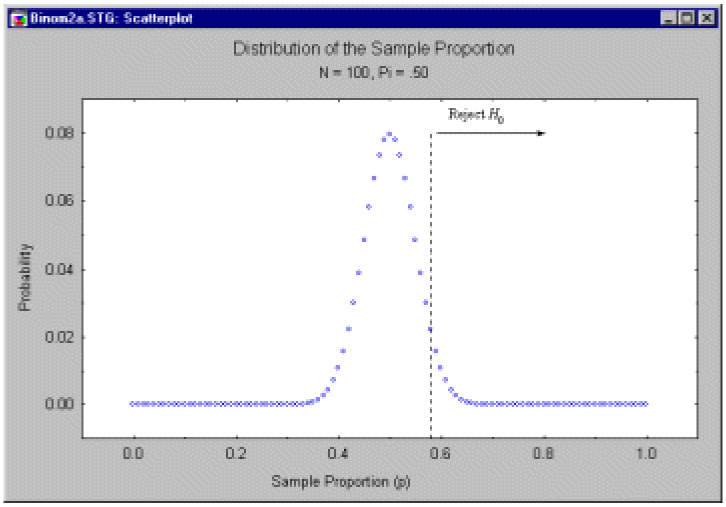


The values are centered around .5, but a small percentage of values are greater than .6 or less than .4. This *distribution* of values reflects the fact that an opinion poll based on a sample of 100 is an imperfect indicator of the population proportion π.

If *p* were a "perfect" estimate of π, the standard error of the proportion would be zero, and the sampling distribution would be a spike located at 0.5. The spread of the sampling distribution indicates how much "noise" is mixed in with the "signal" generated by the parameter.

Notice from the equation for the standard error of the proportion that, as *N* increases, the standard error of the proportion gets smaller. If *N* becomes large enough, we can be very certain that our estimate *p* will be a very accurate one.

Suppose the politician uses a decision criterion as follows. If the observed value of *p* is greater than .58, she will decide that the null hypothesis that π is less than or equal to .50 is false. This rejection rule is diagrammed below.

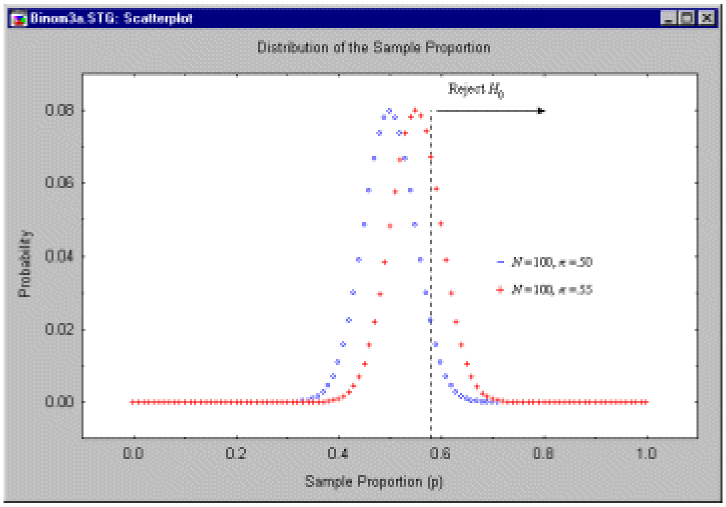


One may, by adding up all the probabilities (computable from the binomial distribution), determine that the probability of rejecting the null hypothesis when *p* = .50 is .044. Hence, this decision rule controls the Type I Error rate, α, at or below .044. It turns out, this is the lowest decision criterion that maintains α at or below .05.

However, the politician is also concerned about *power* in this situation, because it is by *rejecting* the null hypothesis that she is able to support the notion that she has public opinion on her side.

Suppose that 55% of the people support the politician, that is, that π = .55 and the null hypothesis is actually false. In this case, the correct decision is to reject the null hypothesis. What is the probability that she will obtain a sample proportion greater than the "cut-off" value of .58 required to reject the null hypothesis?

In the figure below, we have superimposed the sampling distribution for *p* when π = .55. Clearly, only a small percentage of the time will the politician reach the correct decision that she has majority support. The probability of obtaining a *p* greater than .58 is only .241.



Needless to say, there is no point in conducting an experiment in which, if your position is correct, it will only be verified 24.1% of the time! In this case a statistician would say that the significance test has "inadequate power to detect a departure of 5 percentage points from the null hypothesized value."

The crux of the problem lies in the width of the two distributions in the preceding figure. If the sample size were larger, the standard error of the proportion would be smaller, and there would be little overlap between the distributions. Then it would be possible to find a decision criterion that provides a low α and high power.

The question is, "How large an N is necessary to produce a power that is reasonably high" in this situation, while maintaining α at a reasonably low value.

One could, of course, go through laborious, repetitive calculations in order to arrive at such a sample size. However, a good software program will perform them automatically, with just a few clicks of the mouse. Moreover, for each analytic situation that it handles, it will provide extensive capabilities for analyzing and graphing the theoretical relationships between power, sample size, and the variables that affect them. Assuming that the user will be employing the well known chi-square test, rather than the exact binomial test, suppose that the politician decides that she requires a power of .80 to detect a p of .80. It turns out, a sample size of 607 will yield a power of exactly .8009. (The actual alpha of this test, which has a nominal level of .05, is .0522 in this situation.)

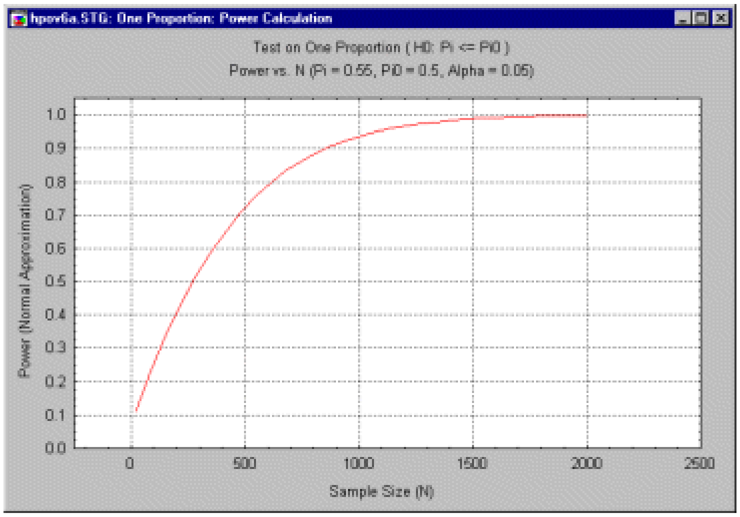
Graphical Approaches to Power Analysis.

In the preceding discussion, we arrived at a necessary sample size of 607 under the assumption that p is precisely .80. In practice, of course, we would be foolish to perform only one power calculation, based on one hypothetical value. For example, suppose the function relating required sample size to p is particularly steep in this case. It might then be that the sample size required for a p of .70 is much different than that required to reliably detect a p of .80.

Intelligent analysis of power and sample size requires the construction, and careful evaluation, of graphs relating power, sample size, the amount by which the null hypothesis is wrong (i.e., the experimental effect), and other factors such as Type I error rate.

In the example discussed in the preceding section, the goal, from the standpoint of the politician, is to plan a study that can decide, with a low probability of error, whether the support level is greater than .50. Graphical analysis can shed a considerable amount of light on the capabilities of a statistical test to provide the desired information under such circumstances.

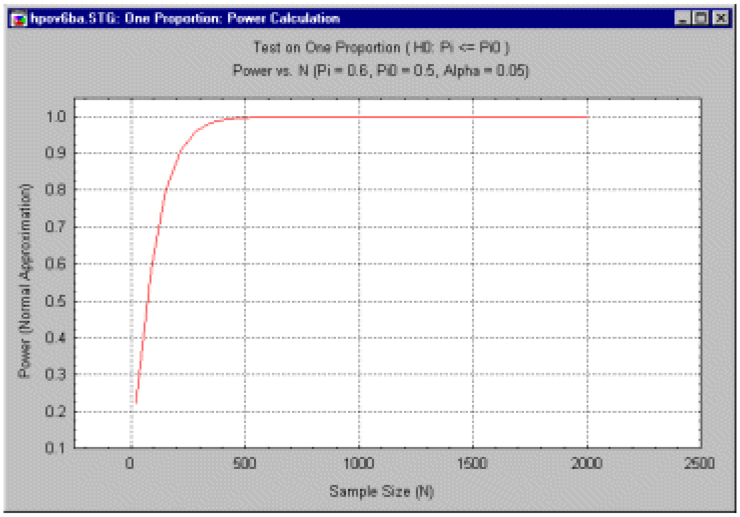
For example, the researcher could plot power against sample size, under the assumption that the true level is .55, i.e., 55%. The user might start with a graph that covers a very wide range of sample sizes, to get a general idea of how the statistical test behaves. The following graph shows power as a function of sample sizes ranging from 20 to 2000, using a "normal approximation" to the exact binomial distribution.



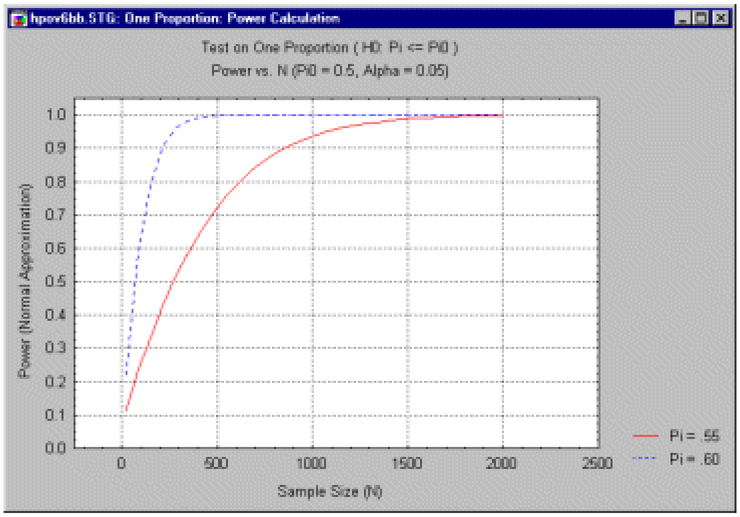
The previous graph demonstrates that power reaches an acceptable level (often considered to be between .80 and .90) at a sample size of approximately 600.

Remember, however, that this calculation is based on the supposition that the true value of *p* is .55. It may be that the shape of the curve relating power and sample size is very sensitive to this value. The question immediately arises, "how sensitive is the slope of this graph to changes in the actual value of *p*?

There are a number of ways to address this question. One can plot power vs. sample size for other values of *p*, for example. Below is a graph of power vs. sample size for *p* = .6.



One can see immediately in the preceding graph that the improvement in power for increases in *N* occurs *much* more rapidly for *p* = .6 than for *p* = .55. The difference is striking if you merge the two graphs into one, as shown below.

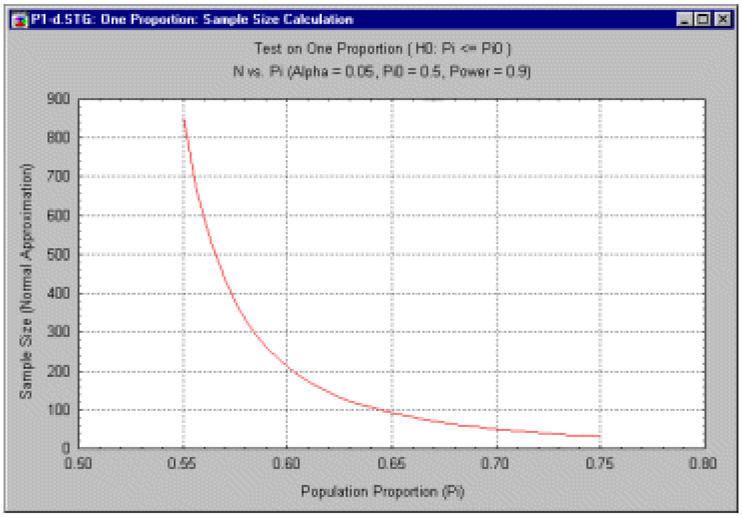


In planning a study, particularly when a grant proposal must be submitted with a proposed sample size, one must estimate what constitutes a reasonable minimum effect that one wishes to detect, a minimum power to detect that effect, and the sample size that will achieve that desired level of power. This sample size can be obtained by analyzing the above graphs (additionally, some software packages can calculate it directly). For example, if the user requests the minimum sample size required to achieve a power of .90 when *p* = .55, some programs can calculate this directly. The result is reported in a spreadsheet, as below,

	One Proportion, Z (or Chi-Square) Test H0: Pi < = Pi0

	Value
Null Hypothesized Proportion (Pi0)	.5000
Population Proportion (Pi)	.5500
Alpha (Nominal)	.0500
Required Power	.9000
Required Sample Size (N)	853.0000
Actual Alpha (Exact)	.0501
Power (Normal Approximation)	.9001
Power (Exact)	.9002

For a given level of power, a graph of sample size vs. *p* can show how sensitive the required sample size is to the actual value of *p*. This can be important in gauging how sensitive the estimate of a required sample size is. For example, the following graph shows values of *N* needed to achieve a power of .90 for various values of *p*, when the null hypothesis is that *p* = .50



The preceding graph demonstrates how the required *N* drops off rapidly as *p* varies from .55 to .60. To be able to reliably detect a difference of .05 (from the null hypothesized value of .50) requires an *N* greater than 800, but reliable detection of a difference of .10 requires an *N* of only around 200. Obviously, then, required sample size is somewhat difficult to pinpoint in this situation. It is much better to be aware of the overall performance of the statistical test against a range of possibilities *before* beginning an experiment, than to be informed of an unpleasant reality after the fact. For example, imagine that the experimenter had estimated the required sample size on the basis of reliably (with power of .90) detecting a *p* of .6. The experimenter budgets for a sample size of, say, 220, and imagines that minor departures of *p* from .6 will not require substantial differences in *N*. Only later does the experimenter realize that a small change in requires a huge increase in *N* , and that the planning for the experiment was optimistic.

In some such situations, a "window of opportunity" may close before the sample size can be adjusted upward.

Across a wide variety of analytic situations, Power analysis and sample size estimation involve steps that are fundamentally the same.

1. The type of analysis and null hypothesis are specified
2. Power and required sample size for a reasonable range of effects is investigated.
3. The sample size required to detect a reasonable experimental effect (i.e., departure from the null hypothesis), with a reasonable level of power, is calculated, while allowing for a reasonable margin of error.

Noncentrality Interval Estimation and the Evaluation of Statistical Models

Power Analysis and Interval Estimation includes a number of confidence intervals that are not widely available in general purpose statistics packages. Several of these are discussed within a common theoretical framework, called "noncentrality interval estimation," by Steiger and Fouladi (1997). In this section, we briefly review some of the basic rationale behind the emerging popularity of confidence intervals.

Inadequacies of the Hypothesis Testing Approach.

Strictly speaking, the outcome of a significance test is the dichotomous decision whether or not to reject the null hypothesis. This dichotomy is inherently dissatisfying to many scientists who use the null hypothesis as a statement of no effect, and are more interested in knowing how big an effect is than whether it is (precisely) zero. This has led to behavior like putting one, two, or three asterisks next to results in tables, or listing *p* levels next to results, when, in fact, such numbers, across (or sometimes even within!) studies need not be monotonically related to the best estimates of strength of experimental effects, and hence can be extremely misleading. Some writers (e.g., Guttman, 1977) view asterisk-placing behavior as inconsistent with the foundations of significance testing logic.

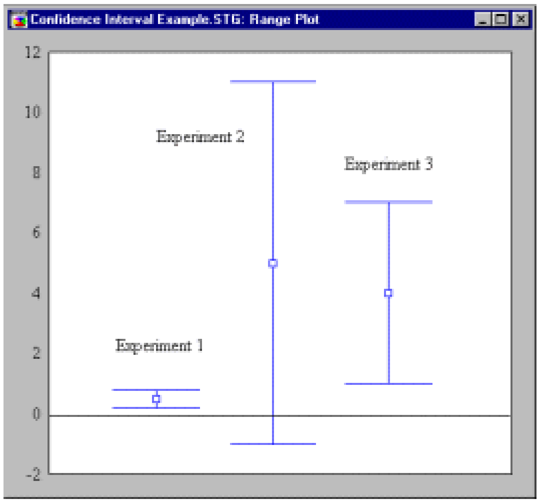
Probability levels can deceive about the "strength" of a result, especially when presented without supporting information. For example, if, in an ANOVA table, one effect had a *p* level of .019, and the other a *p* level of .048, *it might be an error* to conclude that the statistical evidence supported the view that the first effect was stronger than the second. A meaningful interpretation would require additional information. To see why, suppose someone reports a *p* level of .001. This *could* be representative of a trivial population effect combined with a huge sample size, or a powerful population effect combined with a moderate sample size, or a huge population effect with a small sample. Similarly a *p* level of .075 *could* represent a powerful effect operating with a small sample, or a tiny effect with a huge sample. Clearly then, we need to be careful when comparing *p* levels.

In *Accept-Support testing*, which occurs frequently in the context of model fitting in factor analysis or "causal modeling," significance testing logic is basically inappropriate. Rejection of an "almost true" null hypothesis in such situations frequently has been followed by vague statements that the rejection shouldn't be taken too seriously. Failure to reject a null hypothesis usually results in a demand by a vigilant journal editor for cumbersome power calculations. Such problems can be avoided to some extent by using confidence intervals.

Advantages of Interval Estimation. Much research is exploratory. The fundamental questions in exploratory research are "What is our best guess for the size of the population effect?" and "How precisely have we determined the population effect size from our sample data?" Significance testing fails to answer these questions directly. Many a researcher, faced with an "overwhelming rejection" of a null hypothesis, cannot resist the temptation to report that it was "significant *well beyond* the .001 level." Yet it is widely agreed that a *p* level following a significance test can be a poor vehicle for conveying what we have learned about the strength of population effects.

Confidence interval estimation provides a convenient alternative to significance testing in most situations. Consider the 2-tailed hypothesis of no difference between means. Recall first that the significance test rejects at the α significance level if and only if the $1 - \alpha$ confidence interval for the mean difference excludes the value zero. Thus the significance test can be performed with the confidence interval. Most undergraduate texts in behavioral statistics show how to compute such a confidence interval. The interval is exact under the assumptions of the standard *t* test. However, the confidence interval contains information about experimental precision that is not available from the result of a significance test. Assuming we are reasonably confident about the metric of the data, it is much more informative to state a confidence interval on Mu1 - Mu2 than it is to give the *p* level for the *t* test of the hypothesis that Mu1 - Mu2 = 0 In summary, we might say that, in general, a confidence interval conveys more information, in a more naturally usable form, than a significance test.

This is seen most clearly when confidence intervals from several studies are graphed alongside one another, as in the figure below



The figure shows confidence intervals for the difference between means for 3 experiments, all performed in the same domain, using measures with approximately the same variability. Experiments 1 and 3 yield a confidence interval that fails to include zero. For these experiments, the null hypothesis was rejected. The second experiment yields a confidence interval that includes zero, so the null hypothesis of no difference is not rejected. A significance testing approach would yield the impression that the second experiment did not agree with the first and the third.

The confidence intervals suggest a different interpretation, however. The first experiment had a very large sample size, and very high precision of measurement, reflected in a very narrow confidence interval. In this experiment, a small effect was found, and determined with such high precision that the null hypothesis of no difference could be rejected at a stringent significance level.

The second experiment clearly lacked precision, and this is reflected in the very wide confidence interval. Evidently, the sample size was too small. It may well be that the actual effect in conditions assessed in the second experiment was larger than that in the first experiment, but the experimental precision was simply inadequate to detect it.

The third experiment found an effect that was statistically significant, and perhaps substantially higher than the first experiment, although this is partly masked by the lower level of precision, reflected in a confidence interval that, though narrower than Experiment 2, is substantially wider than Experiment 1.

Suppose the 3 experiments involved testing groups for differences in IQ. In the final analysis, we may have had *too much power* in Experiment 1, as we are declaring "highly significant" a rather miniscule effect substantially less than a single IQ point. We had far too little power in Experiment 2. Experiment 3 seems about right.

Many of the arguments we have made on behalf of confidence intervals have been made by others as cogently as we have made them here. Yet, confidence intervals are seldom reported in the literature. Most important, as we demonstrate in the succeeding sections, there are several extremely useful confidence intervals that virtually *never* are reported. In what follows, we discuss *why* the intervals are seldom reported.

Reasons Why Interval Estimates are Seldom Reported.

In spite of the obvious advantages of interval estimates, they are seldom employed in published articles in many areas of science. On those infrequent occasions when interval estimates are reported, they are often not the optimal ones. There are several reasons for this status quo:

Tradition. Traditional approaches to statistics emphasize significance testing much more than interval estimation.

Pragmatism. In RS situations, interval estimates are sometimes embarrassing. When they are narrow but close to zero, they suggest that a "highly significant" result may be statistically significant but trivial. When they are wide, they betray a lack of experimental precision.

Ignorance. Many people are simply unaware of some of the very valuable interval estimation procedures that are available. For example, many textbooks on multivariate analysis never mention that it is possible to compute a confidence interval on the squared multiple correlation coefficient.

Lack of availability. Some of the most desirable interval estimation procedures are computer intensive, and have not been implemented in major statistical packages. This has made it less likely that anyone will try the procedure.

Replacing Traditional Hypothesis Tests with Interval Estimates.

There are a number of confidence interval procedures that can replace and/or augment the traditional hypothesis tests used in classical testing situations. For a review of these techniques, see Steiger & Fouladi (1997).

Analysis of Variance. One area where confidence intervals have seldom been employed is in assessing strength of effects in the Analysis of Variance (ANOVA).

For example, suppose you are reading a paper, which reports that, in a 1-Way ANOVA, with 4 groups, and $N = 60$ per group, an F statistic was found that is significant at the .05 level ("$F = 2.70$, $p =.0464$"). This result is statistically significant, but how *meaningful* is it in a practical sense? What have we learned about the size of the experimental effects?

Fleischman (1980) discusses a technique for setting a confidence interval on the overall effect size in the Analysis of Variance. This technique allows one to set a confidence interval on the RMSSE, the root-mean-square *standardized* effect. Standardized effects are reported in standard deviation units, and are hence remain constant when the unit of measurement changes. So, for example, an experimental effect reported in pounds would be different from the same effect reported in kilograms, whereas the standardized effect would be the same in each case. In the case of the data mentioned above, the *F* statistic that is significant at the .05 level yields a 90% confidence interval for the RMSSE that ranges from .0190 to .3139. The lower limit of this interval stands for a truly mediocre effect, less than 1/50th of a standard deviation. The upper limit of the interval represents effects on the order of 1/3 of a standard deviation, moderate but not overwhelming. It seems, then, that the results from this study need not imply really strong experimental effects, even though the effects are statistically "significant."

Multiple Regression. The squared multiple correlation is reported frequently as an index of the overall strength of a prediction equation. After fitting a regression equation, the most natural questions to ask are, (a) "How effective is the regression equation at predicting the criterion?" and (b) "How precisely has this effectiveness been determined?"

Hence, one very common statistical application that practically cries out for a confidence interval is multiple regression analysis. Publishing an observed squared multiple R together with the result of a hypothesis test that the population squared multiple correlation is zero, conveys little of the available statistical information. A confidence interval on the populations squared multiple correlation is much more informative.

One software package computes exact confidence intervals for the population squared multiple correlation, following the approach of Steiger and Fouladi (1992). As an example, suppose a criterion is predicted from 45 independent observations on 5 variables and the observed squared multiple correlation is .40. In this case a 95% confidence interval for the population squared multiple correlation ranges from .095 to .562! A 95% lower confidence limit is at .129. On the other hand the sample multiple correlation value is significant "beyond the .001 level," because the *p* level is .0009, and the shrunken estimator is .327. Clearly, it is far more impressive to state that "the squared multiple R value is significant at the .001 level" than it is to state that "we are 95% confident that the population squared multiple correlation is between .095 and .562." But we believe the latter statement conveys the quality and meaning of the statistical result more accurately than the former.

Some writers, like Lee (1972), prefer a lower confidence limit, or "statistical lower bound" on the squared multiple correlation to a confidence interval. The rationale, apparently, is that one is primarily interested in assuring that the percentage of variance "accounted for" in the regression equation exceeds some value. Although we understand the motivation behind this view, we hesitate to accept it. The confidence interval, in fact, contains a lower bound, but also includes an upper bound, and, in the interval width, a measure of precision of estimation. It seems to us that adoption of a lower confidence limit can lead to a false sense of security, and reduces that amount of information available in the model assessment process.

Chapter 34
Process Analysis

- Sampling Plans
    - General Purpose
    - Computational Approach
    - Means for H0 and H1
    - Alpha and Beta Error Probabilities
    - Fixed Sampling Plans
    - Sequential Sampling Plans
    - Summary
- Process (Machine) Capability Analysis
    - Introductory Overview
    - Computational Approach
    - Process Capability Indices
    - Process Performance vs. Process Capability
    - Using Experiments to Improve Process Capability
    - Testing the Normality Assumption
    - Tolerance Limits
- Gage Repeatability and Reproducibility
    - Introductory Overview
    - Computational Approach
    - Plots of Repeatability and Reproducibility
    - Components of Variance
    - Summary
- Non-Normal Distributions
    - Introductory Overview
    - Fitting Distributions by Moments
    - Assessing the Fit: Quantile and Probability Plots
    - Non-Normal Process Capability Indices (Percentile Method)
- Weibull and Reliability/Failure Time Analysis
    - General Purpose
    - The Weibull Distribution
    - Censored Observations
    - Two- and three-parameter Weibull Distribution
    - Parameter Estimation
    - Goodness of Fit Indices
    - Interpreting Results
    - Grouped Data
    - Modified Failure Order for Multiple-Censored Data
    - Weibull CDF, Reliability, and Hazard Functions

Sampling plans are discussed in detail in Duncan (1974) and Montgomery (1985); most process capability procedures (and indices) were only recently introduced to the US from Japan (Kane, 1986), however, they are discussed in three excellent recent hands-on books by Bohte (1988), Hart and Hart (1989), and Pyzdek (1989); detailed discussions of these methods can also be found in Montgomery (1991).

Step-by-step instructions for the computation and interpretation of capability indices are also provided in the *Fundamental Statistical Process Control Reference Manual* published by the ASQC (*American Society for Quality Control*) and AIAG (*Automotive Industry Action Group*, 1991; referenced as *ASQC/AIAG, 1991*). Repeatability and reproducibility (*R & R*) methods are discussed in Grant and Leavenworth (1980), Pyzdek (1989) and Montgomery (1991); a more detailed discussion of the subject (of variance estimation) is also provided in Duncan (1974).

Step-by-step instructions on how to conduct and analyze R & R experiments are presented in the *Measurement Systems Analysis Reference Manual* published by ASQC/AIAG (1990). In the following topics, we will briefly introduce the purpose and logic of each of these procedures. For more information on analyzing designs with random effects and for estimating components of variance, see the Variance Components chapter.

Sampling Plans

- General Purpose
- Computational Approach
- Means for H0 and H1
- Alpha and Beta Error Probabilities
- Fixed Sampling Plans
- Sequential Sampling Plans
- Summary

General Purpose

A common question that quality control engineers face is to determine how many items from a batch (e.g., shipment from a supplier) to inspect in order to ensure that the items (products) in that batch are of acceptable quality. For example, suppose we have a supplier of piston rings for small automotive engines that our company produces, and our goal is to establish a sampling procedure (of piston rings from the delivered batches) that ensures a specified quality. In principle, this problem is similar to that of on-line quality control discussed in *Quality Control*. In fact, you may want to read that section at this point to familiarize yourself with the issues involved in industrial statistical quality control.

Acceptance sampling. The procedures described here are useful whenever we need to decide whether or not a batch or lot of items complies with specifications, without having to inspect 100% of the items in the batch. Because of the nature of the problem -- whether or not to accept a batch -- these methods are also sometimes discussed under the heading of *acceptance sampling*.

Advantages over 100% inspection. An obvious advantage of acceptance sampling over 100% inspection of the batch or lot is that reviewing only a sample requires less time, effort, and money. In some cases, inspection of an item is destructive (e.g., stress testing of steel), and testing 100% would destroy the entire batch. Finally, from a managerial standpoint, rejecting an entire batch or shipment (based on acceptance sampling) from a supplier, rather than just a certain percent of defective items (based on 100% inspection) often provides a stronger incentive to the supplier to adhere to quality standards.

Computational Approach

In principle, the computational approach to the question of how large a sample to take is straightforward. *Elementary Concepts* discusses the concept of the sampling distribution. Briefly, if we were to take repeated samples of a particular size from a population of, for example, piston rings and compute their average diameters, then the distribution of those averages (means) would approach the normal distribution with a particular mean and standard deviation (or *standard error*; in sampling distributions the term standard error is preferred, in order to distinguish the variability of the means from the variability of the items in the population). Fortunately, we do not need to take repeated samples from the population in order to estimate the location (mean) and variability (standard error) of the sampling distribution. If we have a good idea (estimate) of what the variability (standard deviation or *sigma*) is in the population, then we can infer the sampling distribution of the mean. In principle, this information is sufficient to estimate the sample size that is needed in order to detect a certain change in quality (from target specifications). Without going into the details about the computational procedures involved, let us next review the particular information that the engineer must supply in order to estimate required sample sizes.

Means for H_0 and H_1

To formalize the inspection process of, for example, a shipment of piston rings, we can formulate two alternative hypotheses: First, we may hypothesize that the average piston ring diameters comply with specifications. This hypothesis is called the *null hypothesis* (H_0). The second and *alternative hypothesis* (H_1) is that the diameters of the piston rings delivered to us deviate from specifications by more than a certain amount. Note that we may specify these types of hypotheses not just for measurable variables such as diameters of piston rings, but also for attributes. For example, we may hypothesize (H_1) that the number of defective parts in the batch exceeds a certain percentage. Intuitively, it should be clear that the larger the difference between H_0 and H_1, the smaller the sample necessary to detect this difference (see *Elementary Concepts*).

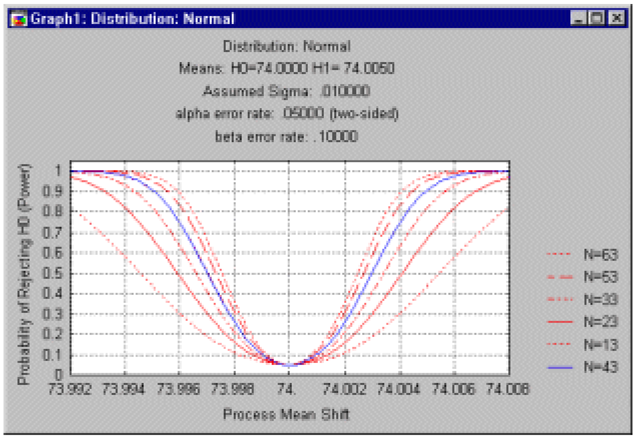
Alpha and *Beta* Error Probabilities

To return to the piston rings example, there are two types of mistakes that we can make when inspecting a batch of piston rings that has just arrived at our plant. First, we may *erroneously reject H_0*, that is, reject the batch because we erroneously conclude that the piston ring diameters deviate from target specifications. The probability of committing this mistake is usually called the *alpha* error probability. The second mistake that we can make is to *erroneously not reject H_0* (accept the shipment of piston rings), when, in fact, the mean piston ring diameter deviates from the target specification by a certain amount. The probability of committing this mistake is usually called the *beta* error probability. Intuitively, the more certain we want to be, that is, the lower we set the *alpha* and *beta* error probabilities, the larger the sample will have to be; in fact, in order to be 100% certain, we would have to measure every single piston ring delivered to our company.

Fixed Sampling Plans

To construct a simple sampling plan, we would first decide on a sample size, based on the means under H_0/H_1 and the particular *alpha* and *beta* error probabilities. Then, we would take a single sample of this *fixed* size and, based on the mean in this sample, decide whether to accept or reject the batch. This procedure is referred to as a *fixed sampling plan*.

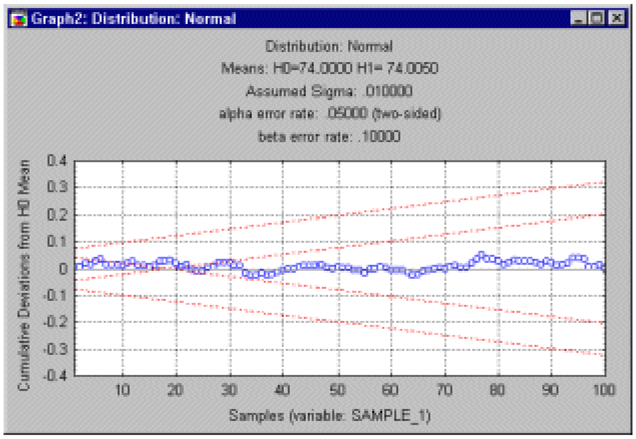
Operating characteristic (OC) curve. The *power* of the fixed sampling plan can be summarized via the operating characteristic curve. In that plot, the probability of rejecting H_0 (and accepting H_1) is plotted on the Y axis, as a function of an actual shift from the target (nominal) specification to the respective values shown on the *X* axis of the plot (see example below). This probability is, of course, one minus the *beta* error probability of erroneously rejecting H_1 and accepting H_0; this value is referred to as the *power* of the fixed sampling plan to detect deviations. Also indicated in this plot are the power functions for smaller sample sizes.



Sequential Sampling Plans

As an alternative to the fixed sampling plan, we could randomly choose individual piston rings and record their deviations from specification. As we continue to measure each piston ring, we could keep a running total of the *sum of deviations* from specification. Intuitively, if H_1 is true, that is, if the average piston ring diameter in the batch is not on target, then we would expect to observe a slowly increasing or decreasing cumulative sum of deviations, depending on whether the average diameter in the batch is larger or smaller than the specification, respectively. It turns out that this kind of *sequential sampling* of individual items from the batch is a more sensitive procedure than taking a fixed sample. In practice, we continue sampling until we either accept or reject the batch.

Using a sequential sampling plan. Typically, we would produce a graph in which the cumulative deviations from specification (plotted on the *Y*-axis) are shown for successively sampled items (e.g., piston rings, plotted on the *X*-axis). Then two sets of lines are drawn in this graph to denote the "corridor" along which we will continue to draw samples, that is, as long as the cumulative sum of deviations from specifications stays within this corridor, we continue sampling.



Distribution: Normal
Means: H0=74.0000 H1= 74.0050
Assumed Sigma: .010000
alpha error rate: .05000 (two-sided)
beta error rate: .10000

If the cumulative sum of deviations steps outside the corridor we stop sampling. If the cumulative sum moves above the upper line or below the lowest line, we reject the batch. If the cumulative sum steps out of the corridor to the *inside*, that is, if it moves closer to the center line, we accept the batch (since this indicates zero deviation from specification). Note that the *inside* area starts only at a certain sample number; this indicates the minimum number of samples necessary to accept the batch (with the current error probability).

Summary

To summarize, the idea of (acceptance) sampling is to use statistical "inference" to accept or reject an entire batch of items, based on the inspection of only relatively few items from that batch. The advantage of applying statistical reasoning to this decision is that we can be explicit about the probabilities of making a wrong decision.

Whenever possible, sequential sampling plans are preferable to fixed sampling plans because they are more powerful. In most cases, relative to the fixed sampling plan, using sequential plans requires fewer items to be inspected in order to arrive at a decision with the same degree of certainty.

Process (Machine) Capability Analysis

- Introductory Overview
- Computational Approach
- Process Capability Indices
- Process Performance vs. Process Capability
- Using Experiments to Improve Process Capability
- Testing the Normality Assumption
- Tolerance Limits

Introductory Overview

See also, Non-Normal Distributions.

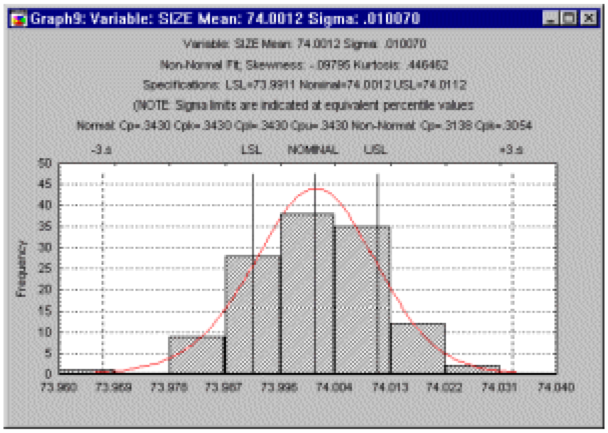
Quality Control describes numerous methods for monitoring the quality of a production process. However, once a process is under control the question arises, "to what extent does the long-term performance of the process comply with engineering requirements or managerial goals?" For example, to return to our piston ring example, how many of the piston rings that we are using fall within the design specification limits? In more general terms, the question is, "how capable is our process (or supplier) in terms of producing items within the specification limits?" Most of the procedures and indices described here were only recently introduced to the US by Ford Motor Company (Kane, 1986). They allow us to summarize the process capability in terms of meaningful percentages and indices.

In this topic, the computation and interpretation of process capability indices will first be discussed for the normal distribution case. If the distribution of the quality characteristic of interest does not follow the normal distribution, modified capability indices can be computed based on the percentiles of a fitted non-normal distribution.

Order of business. Note that it makes little sense to examine the process capability if the process is not in control. If the means of successively taken samples fluctuate widely, or are clearly off the target specification, then those quality problems should be addressed first. Therefore, the first step towards a high-quality process is to bring the process under control, using the charting techniques available in *Quality Control*.

Computational Approach

Once a process is in control, we can ask the question concerning the process capability. Again, the approach to answering this question is based on "statistical" reasoning, and is actually quite similar to that presented earlier in the context of sampling plans. To return to the piston ring example, given a sample of a particular size, we can estimate the standard deviation of the process, that is, the resultant ring diameters. We can then draw a histogram of the distribution of the piston ring diameters. As we discussed earlier, if the distribution of the diameters is normal, then we can make inferences concerning the proportion of piston rings within specification limits.


Graph9: Variable: SIZE Mean: 74.0012 Sigma: .010070

Variable: SIZE Mean: 74.0012 Sigma: .010070

Non-Normal Fit; Skewness: -.09796 Kurtosis: .446462

Specifications: LSL=73.9911 Nominal=74.0012 USL=74.0112

(NOTE: Sigma limits are indicated at equivalent percentile values

Normal: Cp=.3430 Cpk=.3430 Cpl=.3430 Cpu=.3430 Non-Normal: Cp=.3138 Cpk=.3054

(For non-normal distributions, see *Percentile Method*. Let us now review some of the major indices that are commonly used to describe process capability.

Capability Analysis - Process Capability Indices

Process range. First, it is customary to establish the ± 3 *sigma* limits around the nominal specifications. Actually, the *sigma* limits should be the same as the ones used to bring the process under control using Shewhart control charts (see *Quality Control*). These limits denote the *range* of the process (i.e., *process range*). If we use the ± 3 *sigma* limits then, based on the normal distribution, we can estimate that approximately 99% of all piston rings fall within these limits.

Specification limits LSL, USL. Usually, engineering requirements dictate a range of acceptable values. In our example, it may have been determined that acceptable values for the piston ring diameters would be 74.0 ± .02 millimeters. Thus, the *lower specification limit (LSL)* for our process is 74.0 - 0.02 = 73.98; the *upper specification limit (USL)* is 74.0 + 0.02 = 74.02. The difference between USL and LSL is called the *specification range*.

Potential capability (C_p). This is the simplest and most straightforward indicator of process capability. It is defined as the ratio of the specification range to the process range; using ± 3 *sigma* limits we can express this index as:

C_p = (USL-LSL)/(6*Sigma)

Put into words, this ratio expresses the proportion of the range of the normal curve that falls within the engineering specification limits (provided that the mean is on target, that is, that the process is *centered*, see below).

Bhote (1988) reports that prior to the widespread use of statistical quality control techniques (prior to 1980), the normal quality of US manufacturing processes was approximately C_p = .67. This means that the two 33/2 percent tail areas of the normal curve fall outside specification limits. As of 1988, only about 30% of US processes are at or below this level of quality (see Bhote, 1988, p. 51). Ideally, of course, we would like this index to be greater than 1, that is, we would like to achieve a process capability so that no (or almost no) items fall outside specification limits. Interestingly, in the early 1980's the Japanese manufacturing industry adopted as their standard C_p = 1.33! The process capability required to manufacture high-tech products is usually even higher than this; Minolta has established a C_p index of 2.0 as their minimum standard (Bhote, 1988, p. 53), and as the standard for its suppliers. Note that high process capability usually implies lower, not higher costs, taking into account the costs due to poor quality. We will return to this point shortly.

Capability ratio (C_r). This index is equivalent to C_p; specifically, it is computed as 1/C_p (the inverse of C_p).

Lower/upper potential capability: C_{pl}, C_{pu}. A major shortcoming of the C_p (and C_r) index is that it may yield erroneous information if the process is not on target, that is, if it is not *centered*. We can express non-centering via the following quantities. First, upper and lower potential capability indices can be computed to reflect the deviation of the observed process mean from the LSL and USL.. Assuming ± 3 *sigma* limits as the process range, we compute:

C_{pl} = (Mean - LSL)/3*Sigma
and
C_{pu} = (USL - Mean)/3*Sigma

Obviously, if these values are not identical to each other, then the process is not centered.

Non-centering correction (K). We can correct C_p for the effects of non-centering. Specifically, we can compute:

K=abs(D - Mean)/(1/2*(USL - LSL))

where

D = (USL+LSL)/2.

This correction factor expresses the non-centering (target specification minus mean) relative to the specification range.

Demonstrated excellence (C_{pk}). Finally, we can adjust C_p for the effect of non-centering by computing:

$C_{pk} = (1-k)*C_p$

If the process is perfectly centered, then *k* is equal to zero, and C_{pk} is equal to C_p. However, as the process drifts from the target specification, *k* increases and C_{pk} becomes smaller than C_p.

Potential Capability II: C_{pm}. A recent modification (Chan, Cheng, & Spiring, 1988) to C_p is directed at adjusting the estimate of *sigma* for the effect of (random) non-centering. Specifically, we may compute the alternative *sigma (Sigma₂*) as:

$\text{Sigma}_2 = \{\Sigma (x_i - TS)^2/(n-1)\}^{½}$

where:
Sigma_2 is the alternative estimate of *sigma*
x_i is the value of the *i*'th observation in the sample
TS is the target or nominal specification
n is the number of observations in the sample

We may then use this alternative estimate of *sigma* to compute C_p as before; however, we will refer to the resultant index as C_{pm}.

Process Performance vs. Process Capability

When monitoring a process via a quality control chart (e.g., the X-bar and R-chart; *Quality Control*) it is often useful to compute the capability indices for the process. Specifically, when the data set consists of multiple samples, such as data collected for the quality control chart, then one can compute two different indices of variability in the data. One is the regular standard deviation for all observations, ignoring the fact that the data consist of multiple samples; the other is to estimate the process's inherent variation from the within-sample variability. For example, when plotting X-bar and R-charts one may use the common estimator R-bar/d_2 for the process sigma (e.g., see Duncan, 1974; Montgomery, 1985, 1991). Note however, that this estimator is only valid if the process is statistically stable. For a detailed discussion of the difference between the total process variation and the inherent variation refer to ASQC/AIAG reference manual (ASQC/AIAG, 1991, page 80).

When the total process variability is used in the standard capability computations, the resulting indices are usually referred to as process performance indices (as they describe the actual performance of the process), while indices computed from the inherent variation (within- sample sigma) are referred to as capability indices (since they describe the inherent capability of the process).

Using Experiments to Improve Process Capability

As mentioned before, the higher the C_p index, the better the process -- and there is virtually no upper limit to this relationship. The issue of quality costs, that is, the losses due to poor quality, is discussed in detail in the context of Taguchi robust design methods (see *Experimental Design*). In general, higher quality usually results in lower costs overall; even though the costs of production may increase, the losses due to poor quality, for example, due to customer complaints, loss of market share, etc. are usually much greater. In practice, two or three well-designed experiments carried out over a few weeks can often achieve a C_p of 5 or higher. If you are not familiar with the use of designed experiments, but are concerned with the quality of a process, we strongly recommend that you review the methods detailed in *Experimental Design*.

Testing the Normality Assumption

The indices we have just reviewed are only meaningful if, in fact, the quality characteristic that is being measured is normally distributed. A specific test of the normality assumption (*Kolmogorov-Smirnov* and *Chi-square* test of goodness-of-fit) is available; these tests are described in most statistics textbooks, and they are also discussed in greater detail in *Nonparametrics and Distribution Fitting*.

A visual check for normality is to examine the *probability-probability* and *quantile-quantile* plots for the normal distribution. For more information, see Process Analysis and Non-Normal Distributions.

Tolerance Limits

Before the introduction of process capability indices in the early 1980's, the common method for estimating the characteristics of a production process was to estimate and examine the *tolerance limits* of the process (see, for example, Hald, 1952). The logic of this procedure is as follows. Let us assume that the respective quality characteristic is normally distributed in the population of items produced; we can then estimate the lower and upper interval limits that will ensure with a certain *level of confidence* (probability) that a certain *percent of the population* is included in those limits. Put another way, given:

1. a specific sample size (*n*),
2. the process mean,
3. the process standard deviation (sigma),
4. a confidence level, and
5. the percent of the population that we want to be included in the interval,

we can compute the corresponding *tolerance* limits that will satisfy all these parameters. You can also compute parameter-free tolerance limits that are not based on the assumption of normality (Scheffe & Tukey, 1944, p. 217; Wilks, 1946, p. 93; see also Duncan, 1974, or Montgomery, 1985, 1991).

See also, *Non-Normal Distributions*.

Gage Repeatability and Reproducibility

- Introductory Overview
- Computational Approach
- Plots of Repeatability and Reproducibility
- Components of Variance
- Summary

Introductory Overview

Gage repeatability and reproducibility analysis addresses the issue of precision of measurement. The purpose of repeatability and reproducibility experiments is to determine the proportion of measurement variability that is due to (1) the items or parts being measured (*part-to-part variation*), (2) the operator or appraiser of the gages (*reproducibility*), and (3) errors (unreliabilities) in the measurements over several trials by the same operators of the same parts (*repeatability*). In the ideal case, all variability in measurements will be due to the part-to- part variation, and only a negligible proportion of the variability will be due to operator reproducibility and trial-to-trial repeatability.

To return to the piston ring example , if we require detection of deviations from target specifications of the magnitude of .01 millimeters, then we obviously need to use gages of sufficient precision. The procedures described here allow the engineer to evaluate the precision of gages and different operators (users) of those gages, relative to the variability of the items in the population.

You can compute the standard indices of repeatability, reproducibility, and part-to-part variation, based either on ranges (as is still common in these types of experiments) or from the analysis of variance (*ANOVA*) table (as, for example, recommended in ASQC/AIAG, 1990, page 65). The ANOVA table will also contain an *F* test (statistical significance test) for the operator-by-part interaction, and report the estimated variances, standard deviations, and confidence intervals for the components of the ANOVA model.

Finally, you can compute the respective percentages of total variation, and report so-called percent-of-tolerance statistics. These measures are briefly discussed in the following sections of this introduction. Additional information can be found in Duncan (1974), Montgomery (1991), or the DataMyte *Handbook* (1992); step-by-step instructions and examples are also presented in the ASQC/AIAG *Measurement systems analysis reference manual* (1990) and the ASQC/AIAG *Fundamental statistical process control reference manual* (1991).

Note that there are several other statistical procedures which may be used to analyze these types of designs; see the section on Methods for Analysis of Variance for details. In particular the methods discussed in the Variance Components and Mixed Model ANOVA/ANCOVA chapter are very efficient for analyzing very large nested designs (e.g., with more than 200 levels overall), or hierarchically nested designs (with or without random factors).

Computational Approach

One may think of each measurement as consisting of the following components:

1. a component due to the characteristics of the part or item being measured,
2. a component due to the reliability of the gage, and
3. a component due to the characteristics of the operator (user) of the gage.

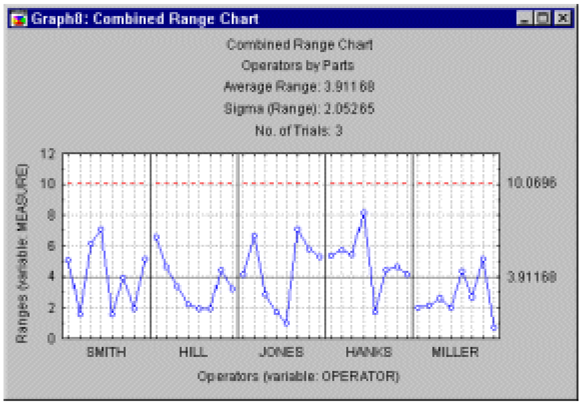
The method of measurement (measurement system) is *reproducible* if different users of the gage come up with identical or very similar measurements. A measurement method is *repeatable* if repeated measurements of the same part produces identical results. Both of these characteristics -- repeatability and reproducibility -- will affect the *precision* of the measurement system.

We can design an experiment to estimate the magnitudes of each component, that is, the repeatability, reproducibility, and the variability between parts, and thus assess the precision of the measurement system. In essence, this procedure amounts to an analysis of variance (*ANOVA*) on an experimental design which includes as factors different parts, operators, and repeated measurements (trials). We can then estimate the corresponding variance components (the term was first used by Daniels, 1939) to assess the *repeatability* (variance due to differences across trials), *reproducibility* (variance due to differences across operators), and variability between parts (variance due to differences across parts). If you are not familiar with the general idea of ANOVA, you may want to refer to *ANOVA/MANOVA*. In fact, the extensive features provided there can also be used to analyze repeatability and reproducibility studies.

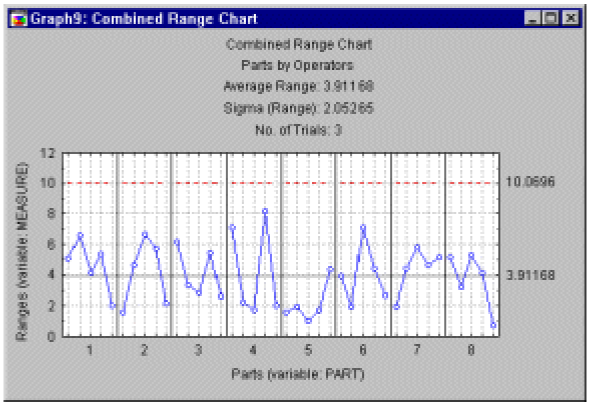
Plots of Repeatability and Reproducibility

There are several ways to summarize via graphs the findings from a repeatability and reproducibility experiment. For example, suppose we are manufacturing small kilns that are used for drying materials for other industrial production processes. The kilns should operate at a target temperature of around 100 degrees Celsius. In this study, 5 different engineers (operators) measured the same sample of 8 kilns (parts), three times each (three trials). We can plot the mean ratings of the 8 parts by operator. If the measurement system is reproducible, then the pattern of means across parts should be quite consistent across the 5 engineers who participated in the study.

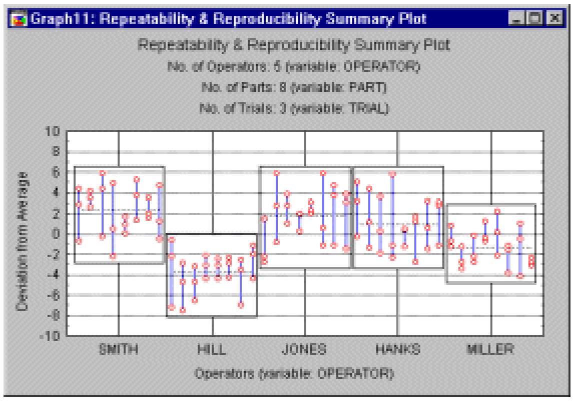
R and S charts. *Quality Control* discusses in detail the idea of *R* (range) and *S* (*sigma*) plots for controlling process variability. We can apply those ideas here and produce a plot of ranges (or *sigmas*) by operators or by parts; these plots will allow us to identify outliers among operators or parts. If one operator produced particularly wide ranges of measurements, we may want to find out why that particular person had problems producing reliable measurements (e.g., perhaps he or she failed to understand the instructions for using the measurement gage).



Analogously, producing an *R* chart by parts may allow us to identify parts that are particularly difficult to measure reliably; again, inspecting that particular part may give us some insights into the weaknesses in our measurement system.



Repeatability and reproducibility summary plot. The summary plot shows the individual measurements by each operator; specifically, the measurements are shown in terms of *deviations* from the respective average rating for the respective part. Each trial is represented by a point, and the different measurement trials for each operator for each part are connected by a vertical line. Boxes drawn around the measurements give us a general idea of a particular operator's bias (see graph below).



Components of Variance (see also the Variance Components chapter)

Percent of Process Variation and Tolerance. The *Percent Tolerance* allows you to evaluate the performance of the measurement system with regard to the overall process variation, and the respective tolerance range. You can specify the tolerance range (*Total tolerance for parts*) and the *Number of sigma intervals*. The latter value is used in the computations to define the range (spread) of the respective (repeatability, reproducibility, part-to- part, etc.) variability. Specifically, the default value (*5.15*) defines *5.15* times the respective *sigma* estimate as the respective range of values; if the data are normally distributed, then this range defines 99% of the space under the normal curve, that is, the range that will include 99% of all values (or reproducibility/repeatability errors) due to the respective source of variation.

Percent of process variation. This value reports the variability due to different sources relative to the total variability (range) in the measurements.

Analysis of Variance. Rather than computing variance components estimates based on ranges, an accurate method for computing these estimates is based on the ANOVA mean squares (see Duncan, 1974, ASQC/AIAG, 1990).

One may treat the three factors in the R & R experiment (*Operator, Parts, Trials*) as random factors in a three-way ANOVA model (see also General *ANOVA/MANOVA*). For details concerning the different models that are typically considered, refer to ASQC/AIAG (1990, pages 92-95), or to Duncan (1974, pages 716-734). Customarily, it is assumed that all interaction effects by the trial factor are non-significant. This assumption seems reasonable, since, for example, it is difficult to imagine how the measurement of some parts will be systematically different in successive trials, in particular when parts and trials are randomized.

However, the *Operator by Parts* interaction may be important. For example, it is conceivable that certain less experienced operators will be more prone to particular biases, and hence will arrive at systematically different measurements for particular parts. If so, then one would expect a significant two-way interaction (again, refer to General *ANOVA/MANOVA* if you are not familiar with ANOVA terminology).

In the case when the two-way interaction is statistically significant, then one can separately estimate the variance components due to operator variability, and due to the operator by parts variability

In the case of significant interactions, the combined repeatability and reproducibility variability is defined as the sum of three components: repeatability (gage error), operator variability, and the operator-by-part variability.

If the *Operator by Part* interaction is not statistically significant a simpler additive model can be used without interactions.

Summary

To summarize, the purpose of the repeatability and reproducibility procedures is to allow the quality control engineer to assess the precision of the measurement system (gages) used in the quality control process. Obviously, if the measurement system is not repeatable (large variability across trials) or reproducible (large variability across operators) relative to the variability between parts, then the measurement system is not sufficiently precise to be used in the quality control efforts. For example, it should not be used in charts produced via *Quality Control*, or product capability analyses and acceptance sampling procedures via *Process Analysis*.

Non-Normal Distributions

- Introductory Overview
- Fitting Distributions by Moments
- Assessing the Fit: Quantile and Probability Plots
- Non-Normal Process Capability Indices (Percentile Method)

Introductory Overview

General Purpose. The concept of process capability is described in detail in the *Process Capability Overview*. To reiterate, when judging the quality of a (e.g., production) process it is useful to estimate the proportion of items produced that fall outside a predefined acceptable specification range. For example, the so-called C_p index is computed as:

C_p - (USL-LSL)/(6*sigma)

where sigma is the estimated process standard deviation, and *USL* and *LSL* are the upper and lower specification limits, respectively. If the distribution of the respective quality characteristic or variable (e.g., size of piston rings) is normal, and the process is perfectly centered (i.e., the mean is equal to the design center), then this index can be interpreted as the proportion of the range of the standard normal curve (the process width) that falls within the engineering specification limits. If the process is not centered, an adjusted index C_{pk} is used instead.

Non-Normal Distributions. You can fit non-normal distributions to the observed histogram, and compute capability indices based on the respective fitted non-normal distribution (via the *percentile method*). In addition, instead of computing capability indices by fitting specific distributions, you can compute capability indices based on two different general families of distributions -- the Johnson distributions (Johnson, 1965; see also Hahn and Shapiro, 1967) and Pearson distributions (Johnson, Nixon, Amos, and Pearson, 1963; Gruska, Mirkhani, and Lamberson, 1989; Pearson and Hartley, 1972) -- which allow the user to approximate a wide variety of continuous distributions. For all distributions, the user can also compute the table of expected frequencies, the expected number of observations beyond specifications, and *quantile-quantile* and *probability-probability* plots. The specific method for computing process capability indices from these distributions is described in Clements (1989).

Quantile-quantile plots and probability-probability plots. There are various methods for assessing the quality of respective fit to the observed data. In addition to the table of observed and expected frequencies for different intervals, and the Kolmogorov-Smirnov and *Chi-square* goodness-of-fit tests, you can compute quantile and probability plots for all distributions. These scatterplots are constructed so that if the observed values follow the respective distribution, then the points will form a straight line in the plot. These plots are described further below.

Fitting Distributions by Moments

In addition to the specific continuous distributions described above, you can fit general "families" of distributions -- the so-called Johnson and Pearson curves -- with the goal to match the first four moments of the observed distribution.

General approach. The shapes of most continuous distributions can be sufficiently summarized in the first four moments. Put another way, if one fits to a histogram of observed data a distribution that has the same mean (first moment), variance (second moment), skewness (third moment) and kurtosis (fourth moment) as the observed data, then one can usually approximate the overall shape of the distribution very well. Once a distribution has been fitted, one can then calculate the expected percentile values under the (standardized) fitted curve, and estimate the proportion of items produced by the process that fall within the specification limits.

Johnson curves. Johnson (1949) described a system of frequency curves that represents transformations of the standard normal curve (see Hahn and Shapiro, 1967, for details). By applying these transformations to a standard normal variable, a wide variety of non-normal distributions can be approximated, including distributions which are bounded on either one or both sides (e.g., *U*-shaped distributions). The advantage of this approach is that once a particular Johnson curve has been fit, the normal integral can be used to compute the expected percentage points under the respective curve. Methods for fitting Johnson curves, so as to approximate the first four moments of an empirical distribution, are described in detail in Hahn and Shapiro, 1967, pages 199-220; and Hill, Hill, and Holder, 1976.

Pearson curves. Another system of distributions was proposed by Karl Pearson (e.g., see Hahn and Shapiro, 1967, pages 220-224). The system consists of seven solutions (of 12 originally enumerated by Pearson) to a differential equation which also approximate a wide range of distributions of different shapes. Gruska, Mirkhani, and Lamberson (1989) describe in detail how the different Pearson curves can be fit to an empirical distribution. A method for computing specific Pearson percentiles is also described in Davis and Stephens (1983).

Assessing the Fit: Quantile and Probability Plots

For each distribution, you can compute the table of expected and observed frequencies and the respective *Chi-square* goodness-of-fit test, as well as the Kolmogorov-Smirnov *d* test. However, the best way to assess the quality of the fit of a theoretical distribution to an observed distribution is to review the plot of the observed distribution against the theoretical fitted distribution. There are two standard types of plots used for this purpose: *Quantile- quantile* plots and *probability-probability* plots.

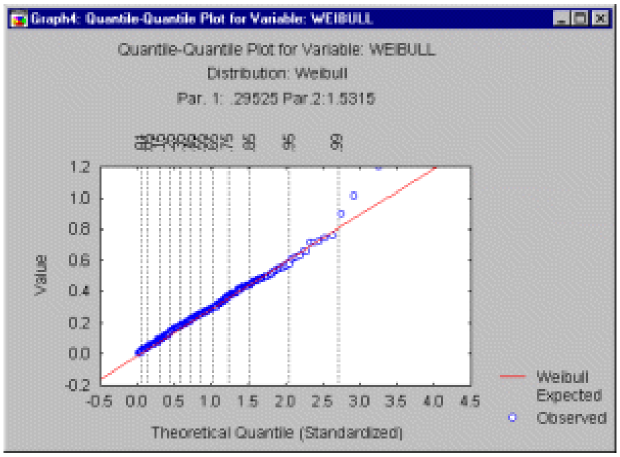
Quantile-quantile plots. In quantile-quantile plots (or Q-Q plots for short), the observed values of a variable are plotted against the theoretical quantiles. To produce a Q-Q plot, you first sort the *n* observed data points into ascending order, so that:

$x_1 \leq x_2 \leq ... \leq x_n$

These observed values are plotted against one axis of the graph; on the other axis the plot will show:

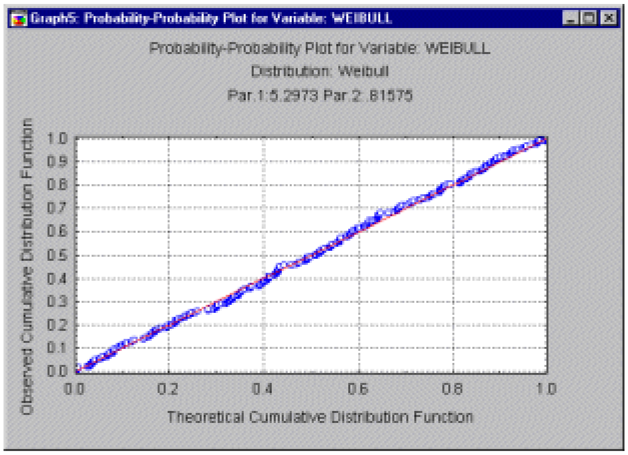
$F^{-1} ((i-r_{adj})/(n+n_{adj}))$

where *i* is the rank of the respective observation, r_{adj} and n_{adj} are adjustment factors ($\leq$ 0.5) and F^{-1} denotes the inverse of the probability integral for the respective standardized distribution. The resulting plot (see example below) is a scatterplot of the observed values against the (standardized) expected values, given the respective distribution. Note that, in addition to the inverse probability integral value, you can also show the respective cumulative probability values on the opposite axis, that is, the plot will show not only the standardized values for the theoretical distribution, but also the respective *p*-values.



A good fit of the theoretical distribution to the observed values would be indicated by this plot if the plotted values fall onto a straight line. Note that the adjustment factors r_{adj} and n_{adj} ensure that the *p*-value for the inverse probability integral will fall between 0 and 1, but not including 0 and 1 (see Chambers, Cleveland, Kleiner, and Tukey, 1983).

Probability-probability plots. In probability-probability plots (or P-P plots for short) the observed cumulative distribution function is plotted against the theoretical cumulative distribution function. As in the Q-Q plot, the values of the respective variable are first sorted into ascending order. The *i*'th observation is plotted against one axis as *i/n* (i.e., the observed cumulative distribution function), and against the other axis as $F(x_{(i)})$, where $F(x_{(i)})$ stands for the value of the theoretical cumulative distribution function for the respective observation $x_{(i)}$. If the theoretical cumulative distribution approximates the observed distribution well, then all points in this plot should fall onto the diagonal line (as in the graph below).



Non-Normal Process Capability Indices (Percentile Method)

As described earlier, process capability indices are generally computed to evaluate the quality of a process, that is, to estimate the relative range of the items manufactured by the process (process width) with regard to the engineering specifications. For the standard, normal- distribution-based, process capability indices, the process width is typically defined as 6 times *sigma*, that is, as plus/minus 3 times the estimated process standard deviation. For the standard normal curve, these limits (z_l = -3 and *zu* = +3) translate into the *0.135* percentile and *99.865* percentile, respectively. In the non-normal case, the 3 times *sigma* limits as well as the mean (z_M = 0.0) can be replaced by the corresponding standard values, given the same percentiles, under the non- normal curve. This procedure is described in detail by Clements (1989).

Process capability indices. Shown below are the formulas for the non-normal process capability indices:

$C_p = (USL-LSL)/(U_p-L_p)$

$C_{pL} = (M-LSL)/(M-L_p)$

$C_{pU} = (USL-M)/(U_p-M)$

$C_{pk} = Min(C_{pU}, C_{pL})$

In these equations, *M* represents the 50'th percentile value for the respective fitted distribution, and U_p and L_p are the *99.865* and *.135* percentile values, respectively, if the computations are based on a process width of ±3 times *sigma*. Note that the values for U_p and L_p may be different, if the process width is defined by different *sigma* limits (e.g., ±2 times *sigma*).

Weibull and Reliability/Failure Time Analysis

- General Purpose
- The Weibull Distribution
- Censored Observations
- Two- and three-parameter Weibull Distribution
- Parameter Estimation
- Goodness of Fit Indices
- Interpreting Results
- Grouped Data
- Modified Failure Order for Multiple-Censored Data
- Weibull CDF, reliability, and hazard functions

A key aspect of product quality is product reliability. A number of specialized techniques have been developed to quantify reliability and to estimate the "life expectancy" of a product. Standard references and textbooks describing these techniques include Lawless (1982), Nelson (1990), Lee (1980, 1992), and Dodson (1994); the relevant functions of the Weibull distribution (hazard, CDF, reliability) are also described in the *Weibull CDF, reliability, and hazard functions* section. Note that very similar statistical procedures are used in the analysis of survival data (see also the description of *Survival Analysis*), and, for example, the descriptions in Lee's book (Lee, 1992) are primarily addressed to biomedical research applications. An excellent overview with many examples of engineering applications is provided by Dodson (1994).

General Purpose

The reliability of a product or component constitutes an important aspect of product quality. Of particular interest is the quantification of a product's reliability, so that one can derive estimates of the product's expected useful life. For example, suppose you are flying a small single engine aircraft. It would be very useful (in fact vital) information to know what the probability of engine failure is at different stages of the engine's "life" (e.g., after 500 hours of operation, 1000 hours of operation, etc.). Given a good estimate of the engine's reliability, and the confidence limits of this estimate, one can then make a rational decision about when to swap or overhaul the engine.

The Weibull Distribution

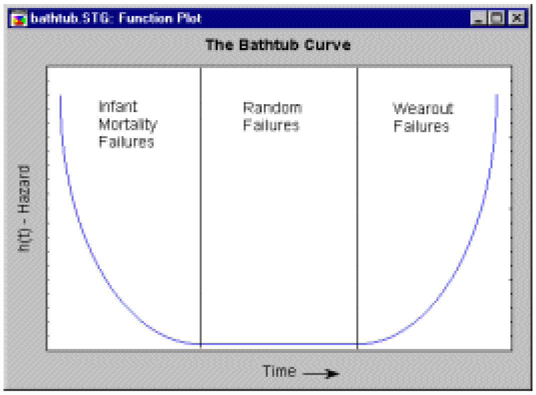
A useful general distribution for describing failure time data is the Weibull distribution (see also *Weibull CDF, reliability, and hazard functions*). The distribution is named after the Swedish professor Waloddi Weibull, who demonstrated the appropriateness of this distribution for modeling a wide variety of different data sets (see also Hahn and Shapiro, 1967; for example, the Weibull distribution has been used to model the life times of electronic components, relays, ball bearings, or even some businesses).

Hazard function and the bathtub curve. It is often meaningful to consider the function that describes the probability of failure during a very small time increment (assuming that no failures have occurred prior to that time). This function is called the hazard function (or, sometimes, also the conditional failure, intensity, or force of mortality function), and is generally defined as:

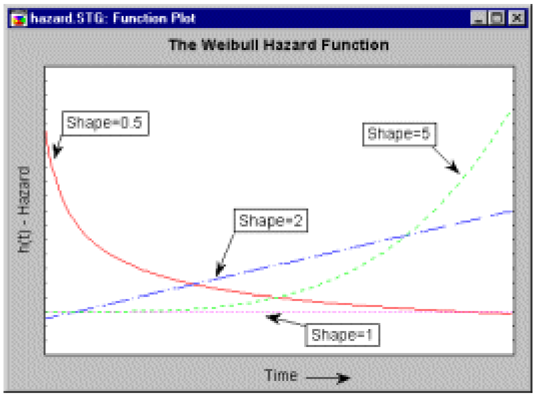
$$h(t) = f(t)/(1-F(t))$$

where *h(t)* stands for the hazard function (of time *t*), and *f(t)* and *F(t)* are the probability density and cumulative distribution functions, respectively. The hazard (conditional failure) function for most machines (components, devices) can best be described in terms of the "bathtub" curve: Very early during the life of a machine, the rate of failure is relatively high (so-called Infant Mortality Failures); after all components settle, and the electronic parts are burned in, the failure rate is relatively constant and low. Then, after some time of operation, the failure rate again begins to increase (so-called Wear-out Failures), until all components or devices will have failed.

For example, new automobiles often suffer several small failures right after they were purchased. Once these have been "ironed out," a (hopefully) long relatively trouble-free period of operation will follow. Then, as the car reaches a particular age, it becomes more prone to breakdowns, until finally, after 20 years and 250000 miles, practically all cars will have failed. A typical bathtub hazard function is shown below.



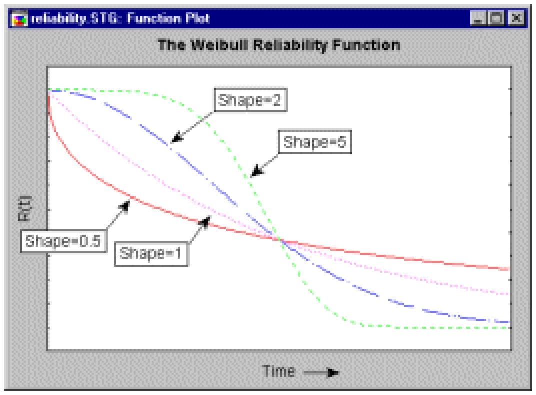
The Weibull distribution is flexible enough for modeling the key stages of this typical bathtub-shaped hazard function. Shown below are the hazard functions for shape parameters *c=.5*, *c=1*, *c=2*, and *c=5.*



Clearly, the early ("infant mortality") "phase" of the bathtub can be approximated by a Weibull hazard function with shape parameter *c<1*; the constant hazard phase of the bathtub can be modeled with a shape parameter *c=1*, and the final ("wear-out") stage of the bathtub with *c>1*.

Cumulative distribution and reliability functions. Once a Weibull distribution (with a particular set of parameters) has been fit to the data, a number of additional important indices and measures can be estimated. For example, you can compute the cumulative distribution function (commonly denoted as *F(t)*) for the fitted distribution, along with the standard errors for this function. Thus, you can determine the percentiles of the cumulative survival (and failure) distribution, and, for example, predict the time at which a predetermined percentage of components can be expected to have failed.

The reliability function (commonly denoted as *R(t)*) is the complement to the cumulative distribution function (i.e., *R(t)=1-F(t)*); the reliability function is also sometimes referred to as the survivorship or survival function (since it describes the probability of not failing or of surviving until a certain time *t*; e.g., see Lee, 1992). Shown below is the reliability function for the Weibull distribution, for different shape parameters.



For shape parameters less than 1, the reliability decreases sharply very early in the respective product's life, and then slowly thereafter. For shape parameters greater than 1, the initial drop in reliability is small, and then the reliability drops relatively sharply at some point later in time. The point where all curves intersect is called the *characteristic life*: regardless of the shape parameter, 63.2 percent of the population will have failed at or before this point (i.e., *R(t)* = 1-0.632 = .368). This point in time is also equal to the respective scale parameter *b* of the two-parameter Weibull distribution (with θ= 0; otherwise it is equal to b+θ).

The formulas for the Weibull cumulative distribution, reliability, and hazard functions are shown in the *Weibull CDF, reliability, and hazard functions* section.

Censored Observations

In most studies of product reliability, not all items in the study will fail. In other words, by the end of the study the researcher only knows that a certain number of items have not failed for a particular amount of time, but has no knowledge of the exact failure times (i.e., "when the items would have failed"). Those types of data are called *censored observations*. The issue of censoring, and several methods for analyzing censored data sets, are also described in great detail in the context of *Survival Analysis*. Censoring can occur in many different ways.

Type I and II censoring. So-called *Type I* censoring describes the situation when a test is terminated at a particular point in time, so that the remaining items are only known not to have failed up to that time (e.g., we start with 100 light bulbs, and terminate the experiment after a certain amount of time). In this case, the censoring time is often fixed, and the number of items failing is a random variable. In *Type II* censoring the experiment would be continued until a fixed proportion of items have failed (e.g., we stop the experiment after exactly 50 light bulbs have failed). In this case, the number of items failing is fixed, and time is the random variable.

Left and right censoring. An additional distinction can be made to reflect the "side" of the time dimension at which censoring occurs. In the examples described above, the censoring always occurred on the right side (*right censoring*), because the researcher knows when exactly the experiment started, and the censoring always occurs on the right side of the time continuum. Alternatively, it is conceivable that the censoring occurs on the left side (*left censoring*). For example, in biomedical research one may know that a patient entered the hospital at a particular date, and that s/he survived for a certain amount of time thereafter; however, the researcher does not know when exactly the symptoms of the disease first occurred or were diagnosed.

Single and multiple censoring. Finally, there are situations in which censoring can occur at different times (*multiple censoring*), or only at a particular point in time (*single censoring*). To return to the light bulb example, if the experiment is terminated at a particular point in time, then a single point of censoring exists, and the data set is said to be single-censored. However, in biomedical research multiple censoring often exists, for example, when patients are discharged from a hospital after different amounts (times) of treatment, and the researcher knows that the patient survived up to those (differential) points of censoring.

The methods described in this section are applicable primarily to right censoring, and single- as well as multiple-censored data.

Two- and three-parameter Weibull distribution

The Weibull distribution is bounded on the left side. If you look at the probability density function, you can see that that the term *x*- θmust be greater than 0. In most cases, the location parameter θ(*theta*) is known (usually 0): it identifies the smallest possible failure time. However, sometimes the probability of failure of an item is 0 (zero) for some time after a study begins, and in that case it may be necessary to estimate a location parameter that is greater than 0. There are several methods for estimating the location parameter of the three-parameter Weibull distribution. To identify situations when the location parameter is greater than 0, Dodson (1994) recommends to look for downward of upward sloping tails on a probability plot (see below), as well as large (>6) values for the shape parameter after fitting the two-parameter Weibull distribution, which may indicate a non-zero location parameter.

Parameter Estimation

Maximum likelihood estimation. Standard iterative function minimization methods can be used to compute maximum likelihood parameter estimates for the two- and three parameter Weibull distribution. The specific methods for estimating the parameters are described in Dodson (1994); a detailed description of a Newton-Raphson iterative method for estimating the maximum likelihood parameters for the two-parameter distribution is provided in Keats and Lawrence (1997).

The estimation of the location parameter for the three-parameter Weibull distribution poses a number of special problems, which are detailed in Lawless (1982). Specifically, when the shape parameter is less than 1, then a maximum likelihood solution does not exist for the parameters. In other instances, the likelihood function may contain more than one maximum (i.e., multiple local maxima). In the latter case, Lawless basically recommends using the smallest failure time (or a value that is a little bit less) as the estimate of the location parameter.

Nonparametric (rank-based) probability plots. One can derive a descriptive estimate of the cumulative distribution function (regardless of distribution) by first rank-ordering the observations, and then computing any of the following expressions:

Median rank:

F(t) = (j-0.3)/(n+0.4)

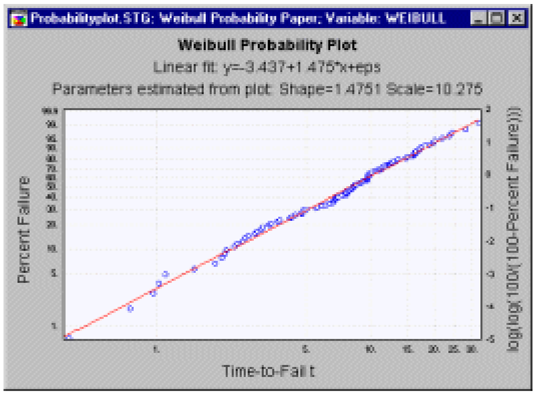
Mean rank:

F(t) = j/(n+1)

White's plotting position:

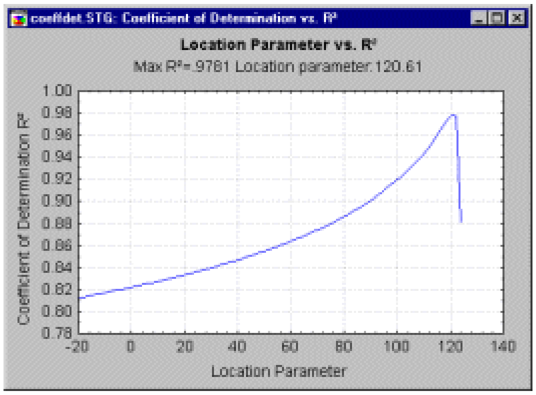
F(t) = (j-3/8)/(n+1/4)

where *j* denotes the failure order (rank; for multiple-censored data a weighted average ordered failure is computed; see Dodson, p. 21), and *n* is the total number of observations. One can then construct the following plot.



Note that the horizontal *Time* axis is scaled logarithmically; on the vertical axis the quantity *log(log(100/(100-F(t)))* is plotted (a probability scale is shown on the left-y axis). From this plot the parameters of the two-parameter Weibull distribution can be estimated; specifically, the shape parameter is equal to the slope of the linear fit-line, and the scale parameter can be estimated as *exp(-intercept/slope).*

Estimating the location parameter from probability plots. It is apparent in the plot shown above that the regression line provides a good fit to the data. When the location parameter is misspecified (e.g., not equal to zero), then the linear fit is worse as compared to the case when it is appropriately specified. Therefore, one can compute the probability plot for several values of the location parameter, and observe the quality of the fit. These computations are summarized in the following plot.



Here the common *R-square* measure (correlation squared) is used to express the quality of the linear fit in the probability plot, for different values of the location parameter shown on the horizontal *x* axis (this plot is based on the example data set in Dodson, 1994, Table 2.9). This plot is often very useful when the maximum likelihood estimation procedure for the three-parameter Weibull distribution fails, because it shows whether or not a unique (single) optimum value for the location parameter exists (as in the plot shown above).

Hazard plotting. Another method for estimating the parameters for the two-parameter Weibull distribution is via hazard plotting (as discussed earlier, the hazard function describes the probability of failure during a very small time increment, assuming that no failures have occurred prior to that time). This method is very similar to the probability plotting method. First plot the cumulative hazard function against the logarithm of the survival times; then fit a linear regression line and compute the slope and intercept of that line. As in probability plotting, the shape parameter can then be estimated as the slope of the regression line, and the scale parameter as *exp(-intercept/slope).* See Dodson (1994) for additional details; see also *Weibull CDF, reliability, and hazard functions*.

Method of moments. This method -- to approximate the moments of the observed distribution by choosing the appropriate parameters for the Weibull distribution -- is also widely described in the literature. In fact, this general method is used for fitting the Johnson curves general non-normal distribution to the data, to compute non-normal process capability indices (see *Fitting Distributions by Moments*). However, the method is not suited for censored data sets, and is therefore not very useful for the analysis of failure time data.

Comparing the estimation methods. Dodson (1994) reports the result of a Monte Carlo simulation study, comparing the different methods of estimation. In general, the maximum likelihood estimates proved to be best for large sample sizes (e.g., $n>15$), while probability plotting and hazard plotting appeared to produce better (more accurate) estimates for smaller samples.

A note of caution regarding maximum likelihood based confidence limits. Many software programs will compute confidence intervals for maximum likelihood estimates, and for the reliability function based on the standard errors of the maximum likelihood estimates. Dodson (1994) cautions against the interpretation of confidence limits computed from maximum likelihood estimates, or more precisely, estimates that involve the information matrix for the estimated parameters. When the shape parameter is less than 2, the variance estimates computed for maximum likelihood estimates lack accuracy, and it is advisable to compute the various results graphs based on nonparametric confidence limits as well.

Goodness of Fit Indices

A number of different tests have been proposed for evaluating the quality of the fit of the Weibull distribution to the observed data. These tests are discussed and compared in detail in Lawless (1982).

Hollander-Proschan. This test compares the theoretical reliability function to the Kaplan-Meier estimate. The actual computations for this test are somewhat complex, and you may refer to Dodson (1994, Chapter 4) for a detailed description of the computational formulas. The Hollander-Proschan test is applicable to complete, single-censored, and multiple-censored data sets; however, Dodson (1994) cautions that the test may sometimes indicate a poor fit when the data are heavily single-censored. The Hollander-Proschan *C* statistic can be tested against the normal distribution (*z*).

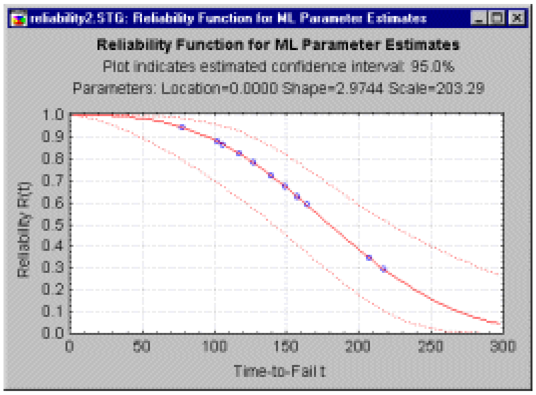
Mann-Scheuer-Fertig. This test, proposed by Mann, Scheuer, and Fertig (1973), is described in detail in, for example, Dodson (1994) or Lawless (1982). The null hypothesis for this test is that the population follows the Weibull distribution with the estimated parameters. Nelson (1982) reports this test to have reasonably good power, and this test can be applied to Type II censored data. For computational details refer to Dodson (1994) or Lawless (1982); the critical values for the test statistic have been computed based on Monte Carlo studies, and have been tabulated for *n* (sample sizes) between 3 and 25.

Anderson-Darling. The Anderson-Darling procedure is a general test to compare the fit of an observed cumulative distribution function to an expected cumulative distribution function. However, this test is only applicable to complete data sets (without censored observations). The critical values for the Anderson-Darling statistic have been tabulated (see, for example, Dodson, 1994, Table 4.4) for sample sizes between 10 and 40; this test is not computed for *n* less than 10 and greater than 40.

Interpreting Results

Once a satisfactory fit of the Weibull distribution to the observed failure time data has been obtained, there are a number of different plots and tables that are of interest to understand the reliability of the item under investigation. If a good fit for the Weibull cannot be established, distribution-free reliability estimates (and graphs) should be reviewed to determine the shape of the reliability function.

Reliability plots. This plot will show the estimated reliability function along with the confidence limits.



Note that nonparametric (distribution-free) estimates and their standard errors can also be computed and plotted.

Hazard plots. As mentioned earlier, the hazard function describes the probability of failure during a very small time increment (assuming that no failures have occurred prior to that time). The plot of hazard as a function of time gives valuable information about the conditional failure probability.

Percentiles of the reliability function. Based on the fitted Weibull distribution, one can compute the percentiles of the reliability (survival) function, along with the confidence limits for these estimates (for maximum likelihood parameter estimates). These estimates are particularly valuable for determining the percentages of items that can be expected to have failed at particular points in time.

Grouped Data

In some cases, failure time data are presented in grouped form. Specifically, instead of having available the precise failure time for each observation, only aggregate information is available about the number of items that failed or were censored in a particular time interval. Such life-table data input is also described in the context of the *Survival Analysis* chapter. There are two general approaches for fitting the Weibull distribution to grouped data.

First, one can treat the tabulated data as if they were continuous. In other words, one can "expand" the tabulated values into continuous data by assuming (1) that each observation in a given time interval failed exactly at the interval mid-point (interpolating out "half a step" for the last interval), and (2) that censoring occurred after the failures in each interval (in other words, censored observations are sorted after the observed failures). Lawless (1982) advises that this method is usually satisfactory if the class intervals are relatively narrow.

Alternatively, you may treat the data explicitly as a tabulated life table, and use a weighted least squares methods algorithm (based on Gehan and Siddiqui, 1973; see also Lee, 1992) to fit the Weibull distribution (Lawless, 1982, also describes methods for computing maximum likelihood parameter estimates from grouped data).

Modified Failure Order for Multiple-Censored Data

For multiple-censored data a weighted average ordered failure is calculated for each failure after the first censored data point. These failure orders are then used to compute the median rank, to estimate the cumulative distribution function.

The modified failure order *j* is computed as (see Dodson 1994):

$$I_j = ((n+1)-O_p)/(1+c)$$

where:

I_j is the increment for the j'th failure
n is the total number of data points
O_p is the failure order of the previous observation (and $O_j = O_p + I_j$)
c is the number of data points remaining in the data set, including the current data point

The median rank is then computed as:

$$F(t) = (I_j -0.3)/(n+0.4)$$

where I_j denotes the modified failure order, and n is the total number of observations.

Weibull CDF, Reliability, and Hazard

Density function. The Weibull distribution (Weibull, 1939, 1951; see also Lieblein, 1955) has density function (for positive parameters *b*, *c*, and θ):

$$f(x) = c/b*[(x-\theta)/b]^{c-1} * e\wedge\{-[(x-\theta)/b]^c\}$$
$$\theta < x,\ \ b > 0,\ \ c > 0$$

where
b is the scale parameter of the distribution
c is the shape parameter of the distribution
θ is the location parameter of the distribution
e is the base of the natural logarithm, sometimes called Euler's e (2.71...)

Cumulative distribution function (CDF). The Weibull distribution has the cumulative distribution function (for positive parameters *b*, *c*, and θ):

$F(x) = 1 - \exp\{-[(x-\theta)/b]^{c}\}$

using the same notation and symbols as described above for the density function.

Reliability function. The Weibull reliability function is the complement of the cumulative distribution function:

$R(x) = 1 - F(x)$

Hazard function. The hazard function describes the probability of failure during a very small time increment, assuming that no failures have occurred prior to that time. The Weibull distribution has the hazard function (for positive parameters *b*, *c*, and θ):

$h(t) = f(t)/R(t) = [c*(x-\theta)^{(c-1)}] / b^{c}$

using the same notation and symbols as described above for the density and reliability functions.

Cumulative hazard function. The Weibull distribution has the cumulative hazard function (for positive parameters *b*, *c*, and θ):

$H(t) = (x-\theta) / b^{c}$

using the same notation and symbols as described above for the density and reliability functions.

Chapter 35
Quality Control Charts

- General Purpose
- General Approach
- Establishing Control Limits
- Common Types of Charts
- Short Run Control Charts
  - Short Run Charts for Variables
  - Short Run Charts for Attributes
- Unequal Sample Sizes
- Control Charts for Variables vs. Charts for Attributes
- Control Charts for Individual Observations
- Out-of-Control Process: Runs Tests
- Operating Characteristic (OC) Curves
- Process Capability Indices
- Other Specialized Control Charts

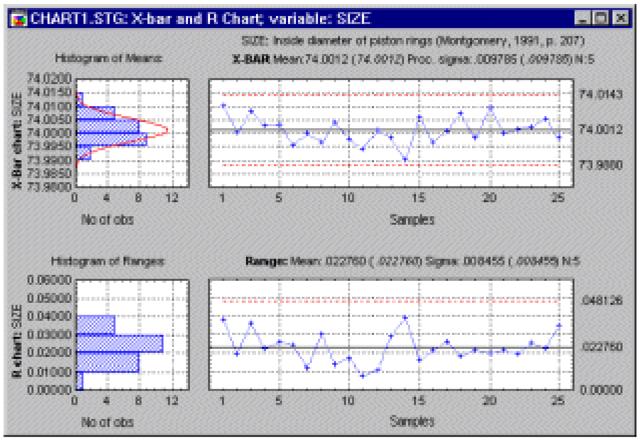
General Purpose

In all production processes, we need to monitor the extent to which our products meet specifications. In the most general terms, there are two "enemies" of product quality: (1) deviations from target specifications, and (2) excessive variability around target specifications. During the earlier stages of developing the production process, designed experiments are often used to optimize these two quality characteristics (see *Experimental Design*); the methods provided in *Quality Control* are *on-line* or *in-process* quality control procedures to monitor an on-going production process. For detailed descriptions of these charts and extensive annotated examples, see Buffa (1972), Duncan (1974) Grant and Leavenworth (1980), Juran (1962), Juran and Gryna (1970), Montgomery (1985, 1991), Shirland (1993), or Vaughn (1974). Two recent excellent introductory texts with a "how-to" approach are Hart & Hart (1989) and Pyzdek (1989); two recent German language texts on this subject are Rinne and Mittag (1995) and Mittag (1993).

General Approach

The general approach to on-line quality control is straightforward: We simply extract samples of a certain size from the ongoing production process. We then produce line charts of the variability in those samples, and consider their closeness to target specifications. If a trend emerges in those lines, or if samples fall outside pre-specified limits, then we declare the process to be out of control and take action to find the cause of the problem. These types of charts are sometimes also referred to as Shewhart control charts (named after W. A. Shewhart who is generally credited as being the first to introduce these methods; see Shewhart, 1931).

Interpreting the chart. The most standard display actually contains two charts (and two histograms); one is called an *X-bar chart*, the other is called an *R chart*.



In both line charts, the horizontal axis represents the different samples; the vertical axis for the X-bar chart represents the means for the characteristic of interest; the vertical axis for the R chart represents the ranges. For example, suppose we wanted to control the diameter of piston rings that we are producing. The center line in the X-bar chart would represent the desired standard size (e.g., diameter in millimeters) of the rings, while the center line in the R chart would represent the acceptable (within-specification) range of the rings within samples; thus, this latter chart is a chart of the variability of the process (the larger the variability, the larger the range). In addition to the center line, a typical chart includes two additional horizontal lines to represent the upper and lower control limits (*UCL*, *LCL*, respectively); we will return to those lines shortly. Typically, the individual points in the chart, representing the samples, are connected by a line. If this line moves outside the upper or lower control limits or exhibits systematic patterns across consecutive samples (see Runs Tests), then a quality problem may potentially exist.

Establishing Control Limits

Even though one could arbitrarily determine when to declare a process out of control (that is, outside the UCL-LCL range), it is common practice to apply statistical principles to do so. *Elementary Concepts* discusses the concept of the *sampling distribution*, and the characteristics of the *normal distribution*. The method for constructing the upper and lower control limits is a straightforward application of the principles described there.

Example. Suppose we want to control the mean of a variable, such as the size of piston rings. Under the assumption that the mean (and variance) of the process does *not* change, the successive sample means will be distributed normally around the actual mean. Moreover, without going into details regarding the derivation of this formula, we also know (because of the central limit theorem, and thus approximate normal distribution of the means; see, for example, Hoyer and Ellis, 1996) that the distribution of sample means will have a standard deviation of *Sigma* (the standard deviation of individual data points or measurements) over the square root of *n* (the sample size). It follows that approximately 95% of the sample means will fall within the limits μ± *1.96 * Sigma/Square Root(n)* (refer to Elementary Concepts for a discussion of the characteristics of the normal distribution and the central limit theorem). In practice, it is common to replace the 1.96 with 3 (so that the interval will include approximately 99% of the sample means), and to define the upper and lower control limits as plus and minus 3 *sigma limits*, respectively.

General case. The general principle for establishing control limits just described applies to all control charts. After deciding on the characteristic we want to control, for example, the standard deviation, we estimate the expected variability of the respective characteristic in samples of the size we are about to take. Those estimates are then used to establish the control limits on the chart.

Common Types of Charts

The types of charts are often classified according to the type of quality characteristic that they are supposed to monitor: there are quality control charts for *variables* and control charts for *attributes*. Specifically, the following charts are commonly constructed for controlling variables:

- **X-bar chart.** In this chart the sample *means* are plotted in order to control the mean value of a variable (e.g., size of piston rings, strength of materials, etc.).
- **R chart.** In this chart, the sample *ranges* are plotted in order to control the variability of a variable.
- **S chart.** In this chart, the sample *standard deviations* are plotted in order to control the variability of a variable.
- **S**2 chart.** In this chart, the sample *variances* are plotted in order to control the variability of a variable.

For controlling quality characteristics that represent *attributes* of the product, the following charts are commonly constructed:

- **C chart.** In this chart (see example below), we plot the *number of defectives* (per batch, per day, per machine, per 100 feet of pipe, etc.). This chart assumes that defects of the quality attribute are *rare*, and the control limits in this chart are computed based on the *Poisson* distribution (distribution of rare events).

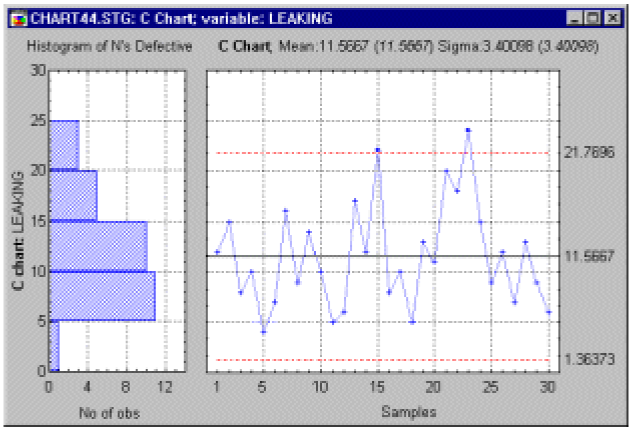


CHART44.STG: C Chart: variable: LEAKING
Histogram of N's Defective
C Chart, Mean:11.5667 (11.5667) Sigma:3.40098 (3.40098)
C chart: LEAKING
30
25
20
15
10
5
0
0
4
8
12
No of obs
21.7696
11.5667
1.36373
1
5
10
15
20
25
30
Samples

- **U chart.** In this chart we plot the *rate of defectives*, that is, the number of defectives divided by the number of units inspected (the *n*; e.g., feet of pipe, number of batches). Unlike the C chart, this chart does not require a constant number of units, and it can be used, for example, when the batches (samples) are of different sizes.
- **Np chart.** In this chart, we plot the number of defectives (per batch, per day, per machine) as in the C chart. However, the control limits in this chart are not based on the distribution of rare events, but rather on the binomial distribution. Therefore, this chart should be used if the occurrence of defectives is not rare (e.g., they occur in more than 5% of the units inspected). For example, we may use this chart to control the number of units produced with minor flaws.
- **P chart.** In this chart, we plot the percent of defectives (per batch, per day, per machine, etc.) as in the U chart. However, the control limits in this chart are not based on the distribution of rare events but rather on the binomial distribution (of proportions). Therefore, this chart is most applicable to situations where the occurrence of defectives is not rare (e.g., we expect the percent of defectives to be more than 5% of the total number of units produced).

All of these charts can be adapted for short production runs (short run charts), and for multiple process streams.

Short Run Charts

The short run control chart, or control chart for short production runs, plots observations of variables or attributes for multiple parts on the same chart. Short run control charts were developed to address the requirement that several dozen measurements of a process must be collected before control limits are calculated. Meeting this requirement is often difficult for operations that produce a limited number of a particular part during a production run.

For example, a paper mill may produce only three or four (huge) rolls of a particular kind of paper (i.e., *part*) and then shift production to another kind of paper. But if variables, such as paper thickness, or attributes, such as blemishes, are monitored for several dozen rolls of paper of, say, a dozen different kinds, control limits for thickness and blemishes could be calculated for the *transformed* (within the short production run) variable values of interest. Specifically, these *transformations* will rescale the variable values of interest such that they are of compatible magnitudes across the different short production runs (or parts). The control limits computed for those transformed values could then be applied in monitoring thickness, and blemishes, regardless of the types of paper (parts) being produced. Statistical process control procedures could be used to determine if the production process is in control, to monitor continuing production, and to establish procedures for continuous quality improvement.

For additional discussions of short run charts refer to Bothe (1988), Johnson (1987), or Montgomery (1991).

Short Run Charts for Variables

Nominal chart, target chart. There are several different types of short run charts. The most basic are the nominal short run chart, and the target short run chart. In these charts, the measurements for each part are transformed by subtracting a part-specific constant. These constants can either be the nominal values for the respective parts (*nominal* short run chart), or they can be target values computed from the (historical) means for each part (*Target X-bar and R chart*). For example, the diameters of piston bores for different engine blocks produced in a factory can only be meaningfully compared (for determining the consistency of bore sizes) if the mean differences between bore diameters for different sized engines are first removed. The nominal or target short run chart makes such comparisons possible. Note that for the nominal or target chart it is assumed that the variability across parts is identical, so that control limits based on a common estimate of the process sigma are applicable.

Standardized short run chart. If the variability of the process for different parts cannot be assumed to be identical, then a further transformation is necessary before the sample means for different parts can be plotted in the same chart. Specifically, in the standardized short run chart the plot points are further transformed by dividing the deviations of sample means from part means (or nominal or target values for parts) by part-specific constants that are proportional to the variability for the respective parts. For example, for the short run X-bar and R chart, the plot points (that are shown in the X-bar chart) are computed by first subtracting from each sample mean a part specific constant (e.g., the respective part mean, or nominal value for the respective part), and then dividing the difference by another constant, for example, by the average range for the respective chart. These transformations will result in comparable scales for the sample means for different parts.

Short Run Charts for Attributes

For attribute control charts (C, U, Np, or P charts), the estimate of the variability of the process (proportion, rate, etc.) is a function of the process average (average proportion, rate, etc.; for example, the standard deviation of a proportion *p* is equal to the square root of *p*(1- p)/n)*. Hence, only standardized short run charts are available for attributes. For example, in the short run P chart, the plot points are computed by first subtracting from the respective sample *p* values the average part *p*'s, and then dividing by the standard deviation of the average *p*'s.

Unequal Sample Sizes

When the samples plotted in the control chart are not of equal size, then the control limits around the center line (target specification) cannot be represented by a straight line. For example, to return to the formula *Sigma/Square Root(n)* presented earlier for computing control limits for the X-bar chart, it is obvious that unequal *n*'s will lead to different control limits for different sample sizes. There are three ways of dealing with this situation.

Average sample size. If one wants to maintain the straight-line control limits (e.g., to make the chart easier to read and easier to use in presentations), then one can compute the average *n* per sample across all samples, and establish the control limits based on the average sample size. This procedure is not "exact," however, as long as the sample sizes are reasonably similar to each other, this procedure is quite adequate.

Variable control limits. Alternatively, one may compute different control limits for each sample, based on the respective sample sizes. This procedure will lead to *variable* control limits, and result in step-chart like control lines in the plot. This procedure ensures that the correct control limits are computed for each sample. However, one loses the simplicity of straight-line control limits.

Stabilized (normalized) chart. The best of two worlds (straight line control limits that are accurate) can be accomplished by standardizing the quantity to be controlled (mean, proportion, etc.) according to units of *sigma*. The control limits can then be expressed in straight lines, while the location of the sample points in the plot depend not only on the characteristic to be controlled, but also on the respective sample *n*'s. The disadvantage of this procedure is that the values on the vertical (*Y*) axis in the control chart are in terms of *sigma* rather than the original units of measurement, and therefore, those numbers cannot be taken at face value (e.g., a sample with a value of 3 is 3 times *sigma* away from specifications; in order to express the value of this sample in terms of the original units of measurement, we need to perform some computations to convert this number back).

Control Charts for Variables vs. Charts for Attributes

Sometimes, the quality control engineer has a choice between variable control charts and attribute control charts.

Advantages of attribute control charts. Attribute control charts have the advantage of allowing for quick summaries of various aspects of the quality of a product, that is, the engineer may simply classify products as acceptable or unacceptable, based on various quality criteria. Thus, attribute charts sometimes bypass the need for expensive, precise devices and time-consuming measurement procedures. Also, this type of chart tends to be more easily understood by managers unfamiliar with quality control procedures; therefore, it may provide more persuasive (to management) evidence of quality problems.

Advantages of variable control charts. Variable control charts are more sensitive than attribute control charts (see Montgomery, 1985, p. 203). Therefore, variable control charts may alert us to quality problems before any actual "unacceptables" (as detected by the attribute chart) will occur. Montgomery (1985) calls the variable control charts *leading indicators* of trouble that will sound an alarm before the number of rejects (scrap) increases in the production process.

Control Chart for Individual Observations

Variable control charts can by constructed for individual observations taken from the production line, rather than samples of observations. This is sometimes necessary when testing samples of multiple observations would be too expensive, inconvenient, or impossible. For example, the number of customer complaints or product returns may only be available on a monthly basis; yet, one would like to chart those numbers to detect quality problems. Another common application of these charts occurs in cases when automated testing devices inspect every single unit that is produced. In that case, one is often primarily interested in detecting small shifts in the product quality (for example, gradual deterioration of quality due to machine wear). The *CUSUM*, *MA*, and *EWMA* charts of cumulative sums and weighted averages discussed below may be most applicable in those situations.

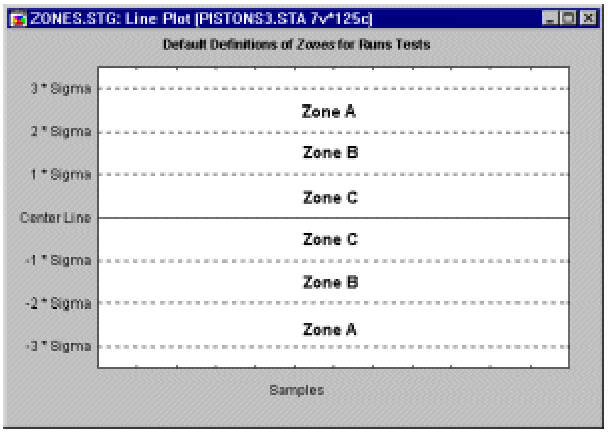
Out-Of-Control Process: Runs Tests

As mentioned earlier in the introduction, when a sample point (e.g., mean in an X-bar chart) falls outside the control lines, one has reason to believe that the process may no longer be in control. In addition, one should look for systematic patterns of points (e.g., means) across samples, because such patterns may indicate that the process average has shifted. These tests are also sometimes referred to as *AT&T runs rules* (see AT&T, 1959) or *tests for special causes* (e.g., see Nelson, 1984, 1985; Grant and Leavenworth, 1980; Shirland, 1993). The term *special or assignable causes* as opposed to *chance or common* causes was used by Shewhart to distinguish between a process that is in control, with variation due to random (chance) causes only, from a process that is out of control, with variation that is due to some non-chance or special (*assignable*) factors (cf. Montgomery, 1991, p. 102).

As the *sigma* control limits discussed earlier, the runs rules are based on "statistical" reasoning. For example, the probability of any sample mean in an X-bar control chart falling above the center line is equal to 0.5, provided (1) that the process is in control (i.e., that the center line value is equal to the population mean), (2) that consecutive sample means are independent (i.e., not auto-correlated), and (3) that the distribution of means follows the normal distribution. Simply stated, under those conditions there is a 50-50 chance that a mean will fall above or below the center line. Thus, the probability that two consecutive means will fall above the center line is equal to 0.5 times 0.5 = 0.25.

Accordingly, the probability that 9 consecutive samples (or a *run* of 9 samples) will fall on the same side of the center line is equal to 0.5**9 = .00195. Note that this is approximately the probability with which a sample mean can be expected to fall outside the 3- times *sigma* limits (given the normal distribution, and a process in control). Therefore, one could look for 9 consecutive sample means on the same side of the center line as another indication of an out-of-control condition. Refer to Duncan (1974) for details concerning the "statistical" interpretation of the other (more complex) tests.

Zone A, B, C. Customarily, to define the runs tests, the area above and below the chart center line is divided into three "zones."



By default, *Zone A* is defined as the area between 2 and 3 times *sigma* above and below the center line; *Zone B* is defined as the area between 1 and 2 times *sigma*, and *Zone C* is defined as the area between the center line and 1 times *sigma*.

9 points in Zone C or beyond (on one side of central line). If this test is positive (i.e., if this pattern is detected), then the process average has probably changed. Note that it is assumed that the distribution of the respective quality characteristic in the plot is symmetrical around the mean. This is, for example, not the case for R charts, S charts, or most attribute charts. However, this is still a useful test to alert the quality control engineer to potential shifts in the process. For example, successive samples with less-than-average variability may be worth investigating, since they may provide hints on how to decrease the variation in the process.

6 points in a row steadily increasing or decreasing. This test signals a drift in the process average. Often, such drift can be the result of tool wear, deteriorating maintenance, improvement in skill, etc. (Nelson, 1985).

14 points in a row alternating up and down. If this test is positive, it indicates that two systematically alternating causes are producing different results. For example, one may be using two alternating suppliers, or monitor the quality for two different (alternating) shifts.

2 out of 3 points in a row in Zone A or beyond. This test provides an "early warning" of a process shift. Note that the probability of a false-positive (test is positive but process is in control) for this test in X-bar charts is approximately 2%.

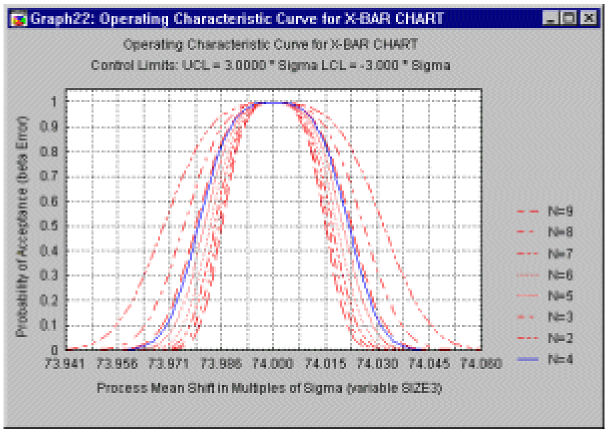
4 out of 5 points in a row in Zone B or beyond. Like the previous test, this test may be considered to be an "early warning indicator" of a potential process shift. The false-positive error rate for this test is also about 2%.

15 points in a row in Zone C (above and below the center line). This test indicates a smaller variability than is expected (based on the current control limits).

8 points in a row in Zone B, A, or beyond, on either side of the center line (without points in Zone C). This test indicates that different samples are affected by different factors, resulting in a bimodal distribution of means. This may happen, for example, if different samples in an X-bar chart where produced by one of two different machines, where one produces above average parts, and the other below average parts.

Operating Characteristic (OC) Curves

A common supplementary plot to standard quality control charts is the so-called *operating characteristic* or *OC* curve (see example below). One question that comes to mind when using standard variable or attribute charts is how sensitive is the current quality control procedure? Put in more specific terms, how likely is it that you will not find a sample (e.g., mean in an X-bar chart) outside the control limits (i.e., *accept* the production process as "in control"), when, in fact, it has shifted by a certain amount? This probability is usually referred to as the β(beta) error probability, that is, the probability of erroneously accepting a process (mean, mean proportion, mean rate defectives, etc.) as being "in control." Note that operating characteristic curves pertain to the false-acceptance probability using the sample-outside-of- control-limits criterion only, and not the runs tests described earlier.



Operating characteristic curves are extremely useful for exploring the power of our quality control procedure. The actual decision concerning sample sizes should depend not only on the cost of implementing the plan (e.g., cost per item sampled), but also on the costs resulting from *not* detecting quality problems. The OC curve allows the engineer to estimate the probabilities of not detecting shifts of certain sizes in the production quality.

Process Capability Indices

For variable control charts, it is often desired to include so-called *process capability* indices in the summary graph. In short, process capability indices express (as a ratio) the proportion of parts or items produced by the current process that fall within user-specified limits (e.g., engineering tolerances).

For example, the so-called Cp index is computed as:

C_p = (USL-LSL)/(6*sigma)

where *sigma* is the estimated process standard deviation, and *USL* and *LSL* are the upper and lower specification (engineering) limits, respectively. If the distribution of the respective quality characteristic or variable (e.g., size of piston rings) is normal, and the process is perfectly centered (i.e., the mean is equal to the design center), then this index can be interpreted as the proportion of the range of the standard normal curve (the process width) that falls within the engineering specification limits. If the process is not centered, an adjusted index C_{pk} is used instead. For a "capable" process, the C_p index should be greater than 1, that is, the specification limits would be larger than *6* times the *sigma* limits, so that over 99% of all items or parts produced could be expected to fall inside the acceptable engineering specifications. For a detailed discussion of this and other indices, refer to *Process Analysis*.

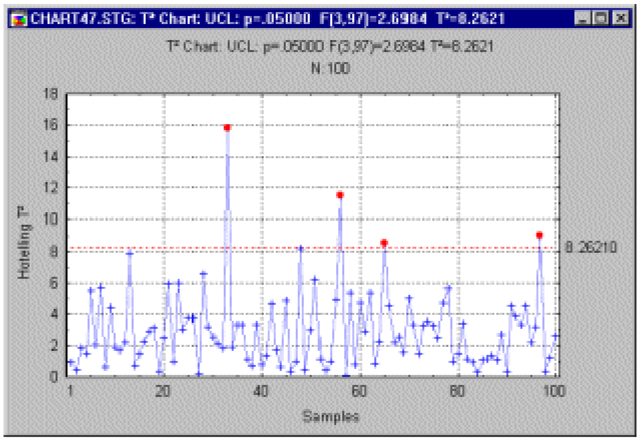
Other Specialized Control Charts

The types of control charts mentioned so far are the "workhorses" of quality control, and they are probably the most widely used methods. However, with the advent of inexpensive desktop computing, procedures requiring more computational effort have become increasingly popular.

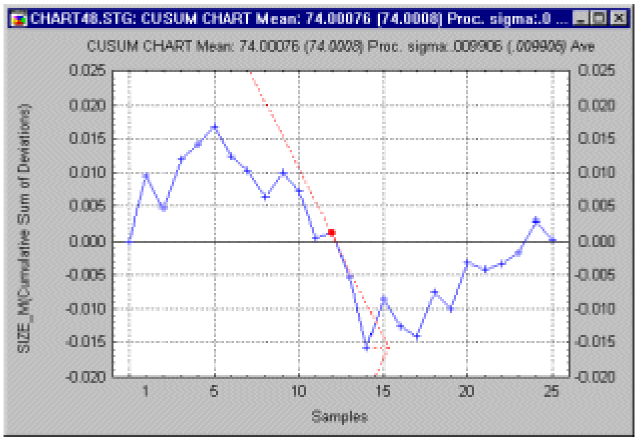
X-bar Charts For Non-Normal Data. The control limits for standard X-bar charts are constructed based on the assumption that the sample means are approximately normally distributed. Thus, the underlying individual observations do not have to be normally distributed, since, as the sample size increases, the distribution of the means will become approximately normal (i.e., see discussion of the central limit theorem in the *Elementary Concepts*; however, note that for *R, S¸ and S**2* charts, it *is* assumed that the individual observations are normally distributed). Shewhart (1931) in his original work experimented with various non-normal distributions for individual observations, and evaluated the resulting distributions of means for samples of size four. He concluded that, indeed, the standard normal distribution-based control limits for the means are appropriate, as long as the underlying distribution of observations are approximately normal. (See also Hoyer and Ellis, 1996, for an introduction and discussion of the distributional assumptions for quality control charting.)

However, as Ryan (1989) points out, when the distribution of observations is highly skewed and the sample sizes are small, then the resulting standard control limits may produce a large number of false alarms (increased alpha error rate), as well as a larger number of false negative ("process-is-in-control") readings (increased beta-error rate). You can compute control limits (as well as process capability indices) for X-bar charts based on so-called Johnson curves(Johnson, 1949), which allow to approximate the skewness and kurtosis for a large range of non-normal distributions (see also Fitting Distributions by Moments, in *Process Analysis*). These non- normal X-bar charts are useful when the distribution of means across the samples is clearly skewed, or otherwise non-normal.

Hotelling T2 Chart.** When there are multiple related quality characteristics (recorded in several variables), we can produce a simultaneous plot (see example below) for all means based on Hotelling multivariate *T**2* statistic (first proposed by Hotelling, 1947).



Cumulative Sum (CUSUM) Chart. The CUSUM chart was first introduced by Page (1954); the mathematical principles involved in its construction are discussed in Ewan (1963), Johnson (1961), and Johnson and Leone (1962).



If one plots the cumulative sum of deviations of successive sample means from a target specification, even minor, permanent shifts in the process mean will eventually lead to a sizable cumulative sum of deviations. Thus, this chart is particularly well-suited for detecting such small permanent shifts that may go undetected when using the X-bar chart. For example, if, due to machine wear, a process slowly "slides" out of control to produce results above target specifications, this plot would show a steadily increasing (or decreasing) cumulative sum of deviations from specification.

To establish control limits in such plots, Barnhard (1959) proposed the so-called *V- mask*, which is plotted after the last sample (on the right). The V-mask can be thought of as the upper and lower control limits for the cumulative sums. However, rather than being parallel to the center line; these lines converge at a particular angle to the right, producing the appearance of a *V* rotated on its side. If the line representing the cumulative sum crosses either one of the two lines, the process is out of control.

Moving Average (MA) Chart. To return to the piston ring example, suppose we are mostly interested in detecting small trends across successive sample means. For example, we may be particularly concerned about machine wear, leading to a slow but constant deterioration of quality (i.e., deviation from specification). The CUSUM chart described above is one way to monitor such trends, and to detect small permanent shifts in the process average. Another way is to use some weighting scheme that summarizes the means of several successive samples; moving such a *weighted mean* across the samples will produce a moving average chart (as shown in the following graph).

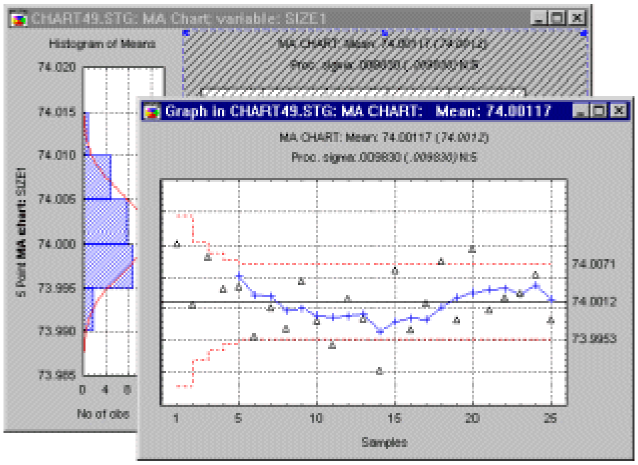


CHART49.STG: MA Chart: variable: SIZE1
Histogram of Means
MA CHART: Mean: 74.00117 (74.0012)
Proc. sigma: .009830 (.009830) N:5
Graph in CHART49.STG: MA CHART: Mean: 74.00117
74.020
74.015
74.010
74.005
74.000
73.995
73.990
73.985
5 Point MA chart: SIZE1
0 4 8
No of obs
74.0071
74.0012
73.9953
1 5 10 15 20 25
Samples

Exponentially-weighted Moving Average (EWMA) Chart. The idea of *moving averages* of successive (adjacent) samples can be generalized. In principle, in order to detect a trend we need to weight successive samples to form a moving average; however, instead of a simple arithmetic moving average, we could compute a geometric moving average (this chart (see graph below) is also called *Geometric Moving Average chart*, see Montgomery, 1985, 1991).

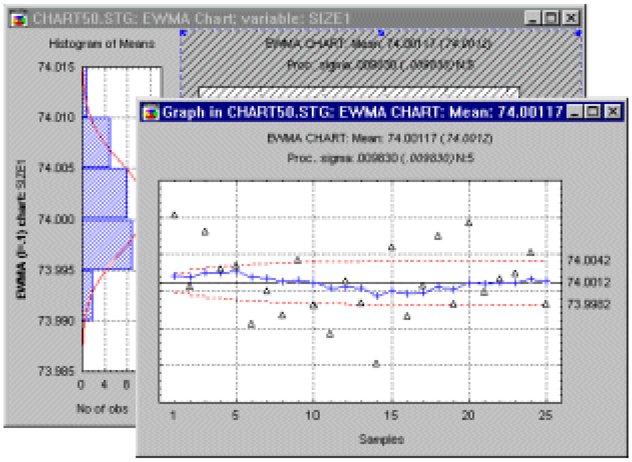


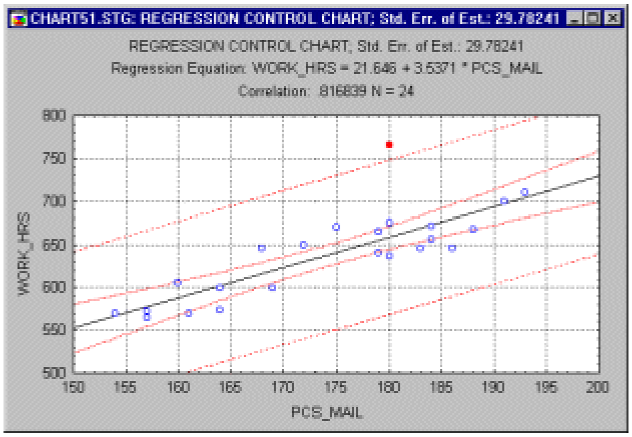
CHART50.STG: EWMA Chart: variable: SIZE1
Histogram of Means
EWMA CHART: Mean: 74.00117 (74.0012)
Proc. sigma: .009830 (.009830) N:5
Graph in CHART50.STG: EWMA CHART: Mean: 74.00117
74.015
74.010
74.005
74.000
73.995
73.990
73.985
EWMA (R=.1) chart: SIZE1
0 4 8
No of obs
74.0042
74.0012
73.9982
1 5 10 15 20 25
Samples

Specifically, we could compute each data point for the plot as:

$z_t = \lambda * \text{x-bar}_t + (1- \lambda)*z_{t-1}$

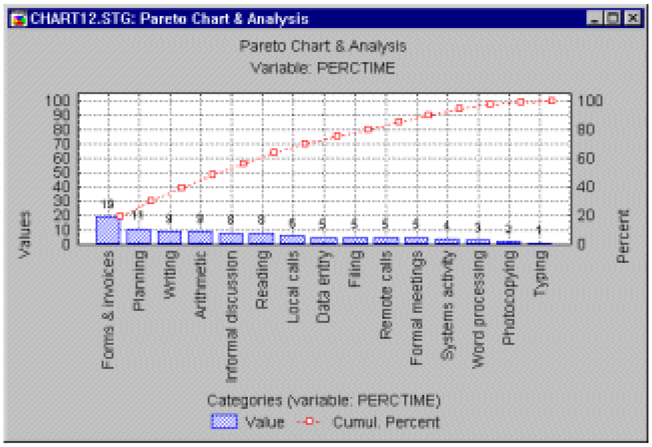
In this formula, each point z_t is computed as λ (lambda) times the respective mean $x\text{-}bar_t$, plus one minus λ times the previous (computed) point in the plot. The parameter λ (lambda) here should assume values greater than 0 and less than 1. Without going into detail (see Montgomery, 1985, p. 239), this method of averaging specifies that the weight of historically "old" sample means decreases geometrically as one continues to draw samples. The interpretation of this chart is much like that of the moving average chart, and it allows us to detect small shifts in the means, and, therefore, in the quality of the production process.

Regression Control Charts. Sometimes we want to monitor the relationship between two aspects of our production process. For example, a post office may want to monitor the number of worker-hours that are spent to process a certain amount of mail. These two variables should roughly be linearly correlated with each other, and the relationship can probably be described in terms of the well-known Pearson product-moment correlation coefficient *r*. This statistic is also described in *Basic Statistics*. The regression control chart contains a regression line that summarizes the linear relationship between the two variables of interest. The individual data points are also shown in the same graph. Around the regression line we establish a confidence interval within which we would expect a certain proportion (e.g., 95%) of samples to fall. Outliers in this plot may indicate samples where, for some reason, the common relationship between the two variables of interest does not hold.



Applications. There are many useful applications for the regression control chart. For example, professional auditors may use this chart to identify retail outlets with a greater than expected number of cash transactions given the overall volume of sales, or grocery stores with a greater than expected number of coupons redeemed, given the total sales. In both instances, outliers in the regression control charts (e.g., too many cash transactions; too many coupons redeemed) may deserve closer scrutiny.

Pareto Chart Analysis. Quality problems are rarely spread evenly across the different aspects of the production process or different plants. Rather, a few "bad apples" often account for the majority of problems. This principle has come to be known as the *Pareto principle*, which basically states that quality losses are mal-distributed in such a way that a small percentage of possible causes are responsible for the majority of the quality problems. For example, a relatively small number of "dirty" cars are probably responsible for the majority of air pollution; the majority of losses in most companies result from the failure of only one or two products. To illustrate the "bad apples", one plots the Pareto chart,



which simply amounts to a histogram showing the distribution of the quality loss (e.g., dollar loss) across some meaningful categories; usually, the categories are sorted into descending order of importance (frequency, dollar amounts, etc.). Very often, this chart provides useful guidance as to where to direct quality improvement efforts.

Chapter 36
Reliability and Item Analysis

- General Introduction
- Basic Ideas
- Classical Testing Model
- Reliability
- Sum Scales
- Cronbach's Alpha
- Split-Half Reliability
- Correction for Attenuation
- Designing a Reliable Scale

This chapter discusses the concept of *reliability of measurement* as used in social sciences (but not in industrial statistics or biomedical research). The term *reliability* used in industrial statistics denotes a function describing the probability of failure (as a function of time). For a discussion of the concept of *reliability* as applied to product quality (e.g., in industrial statistics), please refer to the section on Reliability/Failure Time Analysis in the *Process Analysis* chapter (see also the section Repeatability and Reproducibility in the same chapter and the chapter Survival/Failure Time Analysis). For a comparison between these two (very different) concepts of *reliability*, see Reliability.

General Introduction

In many areas of research, the precise measurement of hypothesized processes or variables (theoretical *constructs*) poses a challenge by itself. For example, in psychology, the precise measurement of personality variables or attitudes is usually a necessary first step before any theories of personality or attitudes can be considered. In general, in all social sciences, unreliable measurements of people's beliefs or intentions will obviously hamper efforts to predict their behavior. The issue of precision of measurement will also come up in applied research, whenever variables are difficult to observe. For example, reliable measurement of employee performance is usually a difficult task; yet, it is obviously a necessary precursor to any performance-based compensation system.

In all of these cases, *Reliability & Item Analysis* may be used to construct reliable measurement scales, to improve existing scales, and to evaluate the reliability of scales already in use. Specifically, *Reliability & Item Analysis* will aid in the design and evaluation of *sum scales*, that is, scales that are made up of multiple individual measurements (e.g., different items, repeated measurements, different measurement devices, etc.). You can compute numerous statistics that allows you to build and evaluate scales following the so-called *classical testing theory* model.

The assessment of scale reliability is based on the correlations between the individual items or measurements that make up the scale, relative to the variances of the items. If you are not familiar with the *correlation coefficient* or the variance statistic, we recommend that you review the respective discussions provided in the Basic Statistics section.

The classical testing theory model of scale construction has a long history, and there are many textbooks available on the subject. For additional detailed discussions, you may refer to, for example, Carmines and Zeller (1980), De Gruitjer and Van Der Kamp (1976), Kline (1979, 1986), or Thorndyke and Hagen (1977). A widely acclaimed "classic" in this area, with an emphasis on psychological and educational testing, is Nunally (1970).

Testing hypotheses about relationships between items and tests. Using Structural Equation Modeling and Path Analysis (*SEPATH*), you can test specific hypotheses about the relationship between sets of items or different tests (e.g., test whether two sets of items measure the same construct, analyze multi-trait, multi-method matrices, etc.).

Basic Ideas

Suppose we want to construct a questionnaire to measure people's prejudices against foreign- made cars. We could start out by generating a number of items such as: "Foreign cars lack personality," "Foreign cars all look the same," etc. We could then submit those questionnaire items to a group of subjects (for example, people who have never owned a foreign-made car). We could ask subjects to indicate their agreement with these statements on 9-point scales, anchored at *1=disagree* and *9=agree*.

True scores and error. Let us now consider more closely what we mean by precise measurement in this case. We hypothesize that there is such a thing (theoretical construct) as "prejudice against foreign cars," and that each item "taps" into this concept to some extent. Therefore, we may say that a subject's response to a particular item reflects two aspects: first, the response reflects the prejudice against foreign cars, and second, it will reflect some esoteric aspect of the respective question. For example, consider the item "Foreign cars all look the same." A subject's agreement or disagreement with that statement will partially depend on his or her general prejudices, and partially on some other aspects of the question or person. For example, the subject may have a friend who just bought a very different looking foreign car.

Testing hypotheses about relationships between items and tests. To test specific hypotheses about the relationship between sets of items or different tests (e.g., whether two sets of items measure the same construct, analyze multi- trait, multi-method matrices, etc.) use *Structural Equation Modeling (SEPATH*).

Classical Testing Model

To summarize, each measurement (response to an item) reflects to some extent the true score for the intended concept (prejudice against foreign cars), and to some extent esoteric, random error. We can express this in an equation as:
X = tau + error
In this equation, *X* refers to the respective actual measurement, that is, subject's response to a particular item; *tau* is commonly used to refer to the *true score*, and *error* refers to the random error component in the measurement.

Reliability

In this context the definition of *reliability* is straightforward: a measurement is reliable if it reflects mostly true score, relative to the error. For example, an item such as "Red foreign cars are particularly ugly" would likely provide an unreliable measurement of prejudices against foreign- made cars. This is because there probably are ample individual differences concerning the likes and dislikes of colors. Thus, this item would "capture" not only a person's prejudice but also his or her color preference. Therefore, the proportion of true score (for prejudice) in subjects' response to that item would be relatively small.

Measures of reliability. From the above discussion, one can easily infer a measure or statistic to describe the reliability of an item or scale. Specifically, we may define an *index of reliability* in terms of the proportion of true score variability that is captured across subjects or respondents, relative to the total observed variability. In equation form, we can say:

$$\text{Reliability} = \sigma^2_{\text{(true score)}} / \sigma^2_{\text{(total observed)}}$$

Sum Scales

What will happen when we sum up several more or less reliable items designed to measure prejudice against foreign-made cars? Suppose the items were written so as to cover a wide range of possible prejudices against foreign-made cars. If the error component in subjects' responses to each question is truly random, then we may expect that the different components will cancel each other out across items. In slightly more technical terms, the expected value or mean of the error component across items will be zero. The true score component remains the same when summing across items. Therefore, the more items are added, the more true score (relative to the error score) will be reflected in the sum scale.

Number of items and reliability. This conclusion describes a basic principle of test design. Namely, the more items there are in a scale designed to measure a particular concept, the more reliable will the measurement (sum scale) be. Perhaps a somewhat more practical example will further clarify this point. Suppose you want to measure the height of 10 persons, using only a crude stick as the measurement device. Note that we are not interested in this example in the absolute correctness of measurement (i.e., in inches or centimeters), but rather in the ability to distinguish reliably between the 10 individuals in terms of their height. If you measure each person only once in terms of multiples of lengths of your crude measurement stick, the resultant measurement may not be very reliable. However, if you measure each person 100 times, and then take the average of those 100 measurements as the summary of the respective person's height, then you will be able to make very precise and reliable distinctions between people (based solely on the crude measurement stick).

Let us now look at some of the common statistics that are used to estimate the reliability of a sum scale.

Cronbach's Alpha

To return to the prejudice example, if there are several subjects who respond to our items, then we can compute the variance for each item, and the variance for the sum scale. The variance of the sum scale will be smaller than the sum of item variances if the items measure the *same* variability between subjects, that is, if they measure some true score. Technically, the variance of the sum of two items is equal to the sum of the two variances *minus* (two times) the covariance, that is, the amount of true score variance common to the two items.

We can estimate the proportion of true score variance that is captured by the items by comparing the sum of item variances with the variance of the sum scale. Specifically, we can compute:

$$\alpha = (k/(k-1)) * [1- \Sigma(s^2_i)/s^2_{sum}]$$

This is the formula for the most common index of reliability, namely, Cronbach's coefficient *alpha* (α). In this formula, the s_i**2's denote the variances for the k individual items; $s_{sum}**2$ denotes the variance for the sum of all items. If there is no true score but only error in the items (which is esoteric and unique, and, therefore, uncorrelated across subjects), then the variance of the sum will be the same as the sum of variances of the individual items. Therefore, coefficient *alpha* will be equal to zero. If all items are perfectly reliable and measure the same thing (true score), then coefficient alpha is equal to 1. (Specifically, $1-\Sigma(s_i**2)/s_{sum}**2$ will become equal to $(k-1)/k$; if we multiply this by $k/(k-1)$ we obtain 1.)

Alternative terminology. Cronbach's *alpha*, when computed for binary (e.g., true/false) items, is identical to the so-called *Kuder-Richardson-20* formula of reliability for sum scales. In either case, because the reliability is actually estimated from the consistency of all items in the sum scales, the reliability coefficient computed in this manner is also referred to as the *internal-consistency reliability*.

Split-Half Reliability

An alternative way of computing the reliability of a sum scale is to divide it in some random manner into two halves. If the sum scale is perfectly reliable, we would expect that the two halves are perfectly correlated (i.e., $r = 1.0$). Less than perfect reliability will lead to less than perfect correlations. We can estimate the reliability of the sum scale via the *Spearman-Brown split half* coefficient:

$$r_{sb} = 2r_{xy} /(1+r_{xy})$$

In this formula, r_{sb} is the split-half reliability coefficient, and r_{xy} represents the correlation between the two halves of the scale.

Correction for Attenuation

Let us now consider some of the consequences of less than perfect reliability. Suppose we use our scale of prejudice against foreign-made cars to predict some other criterion, such as subsequent actual purchase of a car. If our scale correlates with such a criterion, it would raise our confidence in the *validity* of the scale, that is, that it really measures prejudices against foreign-made cars, and not something completely different. In actual test design, the *validation* of a scale is a lengthy process that requires the researcher to correlate the scale with various external criteria that, in theory, should be related to the concept that is supposedly being measured by the scale.

How will validity be affected by less than perfect scale reliability? The random error portion of the scale is unlikely to correlate with some external criterion. Therefore, if the proportion of true score in a scale is only 60% (that is, the reliability is only .60), then the correlation between the scale and the criterion variable will be *attenuated*, that is, it will be smaller than the actual correlation of true scores. In fact, the validity of a scale is always limited by its reliability.

Given the reliability of the two measures in a correlation (i.e., the scale and the criterion variable), we can estimate the actual correlation of true scores in both measures. Put another way, we can *correct* the correlation *for attenuation*:

$$r_{xy,corrected} = r_{xy} /(r_{xx}*r_{yy})^{½}$$

In this formula, $r_{xy,corrected}$ stands for the corrected correlation coefficient, that is, it is the estimate of the correlation between the true scores in the two measures x and y. The term r_{xy} denotes the uncorrected correlation, and r_{xx} and r_{yy} denote the reliability of measures (scales) x and y. You can compute the attenuation correction based on specific values, or based on actual raw data (in which case the reliabilities of the two measures are estimated from the data).

Designing a Reliable Scale

After the discussion so far, it should be clear that, the more reliable a scale, the better (e.g., more valid) the scale. As mentioned earlier, one way to make a sum scale more valid is by adding items. You can compute how many items would have to be added in order to achieve a particular reliability, or how reliable the scale would be if a certain number of items were added. However, in practice, the number of items on a questionnaire is usually limited by various other factors (e.g., respondents get tired, overall space is limited, etc.). Let us return to our prejudice example, and outline the steps that one would generally follow in order to design the scale so that it will be reliable:

Step 1: Generating items. The first step is to write the items. This is essentially a creative process where the researcher makes up as many items as possible that seem to relate to prejudices against foreign-made cars. In theory, one should "sample items" from the domain defined by the concept. In practice, for example in marketing research, *focus groups* are often utilized to illuminate as many aspects of the concept as possible. For example, we could ask a small group of highly committed American car buyers to express their general thoughts and feelings about foreign-made cars. In educational and psychological testing, one commonly looks at other similar questionnaires at this stage of the scale design, again, in order to gain as wide a perspective on the concept as possible.

Step 2: Choosing items of optimum difficulty. In the first draft of our prejudice questionnaire, we will include as many items as possible. We then administer this questionnaire to an initial sample of typical respondents, and examine the results for each item. First, we would look at various characteristics of the items, for example, in order to identify *floor* or *ceiling* effects. If all respondents agree or disagree with an item, then it obviously does not help us discriminate between respondents, and thus, it is useless for the design of a reliable scale. In test construction, the proportion of respondents who agree or disagree with an item, or who answer a test item correctly, is often referred to as the *item difficulty*. In essence, we would look at the item means and standard deviations and eliminate those items that show extreme means, and zero or nearly zero variances.

Step 3: Choosing internally consistent items. Remember that a reliable scale is made up of items that proportionately measure mostly true score; in our example, we would like to select items that measure mostly prejudice against foreign-made cars, and few esoteric aspects we consider random error. To do so, we would look at the following:

STATISTICA RELIABL. ANALYSIS	Summary for scale: Mean=46.1100 Std.Dv.=8.26444 Valid n:100 Cronbach alpha: .794313 Standardized alpha: .800491 Average inter-item corr.: .297818					
variable	Mean if deleted	Var. if deleted	StDv. if deleted	Itm-Totl Correl.	Squared Multp. R	Alpha if deleted
ITEM1	41.61000	51.93790	7.206795	.656298	.507160	.752243
ITEM2	41.37000	53.79310	7.334378	.666111	.533015	.754692
ITEM3	41.41000	54.86190	7.406882	.549226	.363895	.766778
ITEM4	41.63000	56.57310	7.521509	.470852	.305573	.776015
ITEM5	41.52000	64.16961	8.010593	.054609	.057399	.824907
ITEM6	41.56000	62.68640	7.917474	.118561	.045653	.817907
ITEM7	41.46000	54.02840	7.350401	.587637	.443563	.762033
ITEM8	41.33000	53.32110	7.302130	.609204	.446298	.758992
ITEM9	41.44000	55.06640	7.420674	.502529	.328149	.772013
ITEM10	41.66000	53.78440	7.333785	.572875	.410561	.763314

Shown above are the results for 10 items. Of most interest to us are the three right-most columns. They show us the correlation between the respective item and the total sum score (without the respective item), the squared multiple correlation between the respective item and all others, and the internal consistency of the scale (coefficient *alpha*) if the respective item would be deleted. Clearly, items *5* and *6* "stick out," in that they are not consistent with the rest of the scale. Their correlations with the sum scale are *.05* and *.12*, respectively, while all other items correlate at *.45* or better. In the right-most column, we can see that the reliability of the scale would be about *.82* if either of the two items were to be deleted. Thus, we would probably delete the two items from this scale.

Step 4: Returning to Step 1. After deleting all items that are not consistent with the scale, we may not be left with enough items to make up an overall reliable scale (remember that, the fewer items, the less reliable the scale). In practice, one often goes through several rounds of generating items and eliminating items, until one arrives at a final set that makes up a reliable scale.

Tetrachoric correlations. In educational and psychological testing, it is common to use *yes/no* type items, that is, to prompt the respondent to answer either yes or no to a question. An alternative to the regular correlation coefficient in that case is the so-called *tetrachoric* correlation coefficient. Usually, the tetrachoric correlation coefficient is larger than the standard correlation coefficient, therefore, Nunally (1970, p. 102) discourages the use of this coefficient for estimating reliabilities. However, it is a widely used statistic (e.g., in mathematical modeling).

Chapter 37
Structural Equation Modeling

- A Conceptual Overview
- The Basic Idea Behind Structural Modeling
- Structural Equation Modeling and the Path Diagram

A Conceptual Overview

Structural Equation Modeling is a very general, very powerful multivariate analysis technique that includes specialized versions of a number of other analysis methods as special cases. We will assume that you are familiar with the basic logic of statistical reasoning as described in *Elementary Concepts*. Moreover, we will also assume that you are familiar with the concepts of variance, covariance, and correlation; if not, we advise that you read the *Basic Statistics* section at this point. Although it is not absolutely necessary, it is highly desirable that you have some background in *factor analysis* before attempting to use structural modeling.

Major applications of structural equation modeling include:

1. *causal modeling*, or *path analysis*, which hypothesizes causal relationships among variables and tests the causal models with a linear equation system. Causal models can involve either manifest variables, latent variables, or both;
2. *confirmatory factor analysis*, an extension of factor analysis in which specific hypotheses about the structure of the factor loadings and intercorrelations are tested;
3. *second order factor analysis*, a variation of factor analysis in which the correlation matrix of the common factors is itself factor analyzed to provide second order factors;
4. *regression models*, an extension of linear regression analysis in which regression weights may be constrained to be equal to each other, or to specified numerical values;
5. *covariance structure models*, which hypothesize that a covariance matrix has a particular form. For example, you can test the hypothesis that a set of variables all have equal variances with this procedure;
6. *correlation structure models*, which hypothesize that a correlation matrix has a particular form. A classic example is the hypothesis that the correlation matrix has the structure of a circumplex (Guttman, 1954; Wiggins, Steiger, & Gaelick, 1981).

Many different kinds of models fall into each of the above categories, so structural modeling as an enterprise is very difficult to characterize.

Most structural equation models can be expressed as path diagrams. Consequently even beginners to structural modeling can perform complicated analyses with a minimum of training.

The Basic Idea Behind Structural Modeling

One of the fundamental ideas taught in intermediate applied statistics courses is the effect of additive and multiplicative transformations on a list of numbers. Students are taught that, if you multiply every number in a list by some constant K, you multiply the *mean* of the numbers by K. Similarly, you multiply the *standard deviation* by the absolute value of K.

For example, suppose you have the list of numbers 1,2,3. These numbers have a *mean* of 2 and a *standard deviation* of 1. Now, suppose you were to take these 3 numbers and multiply them by 4. Then the mean would become 8, and the standard deviation would become 4, the variance thus 16.

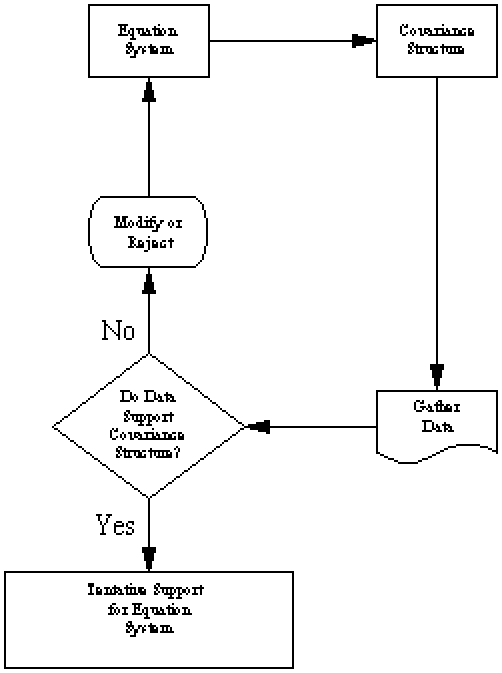
The point is, if you have a set of numbers X related to another set of numbers Y by the equation Y = 4X, then the variance of Y *must* be 16 times that of X, so you can test the hypothesis that Y and X are related by the equation Y = 4X *indirectly* by comparing the variances of the Y and X variables.

This idea generalizes, in various ways, to several variables inter-related by a group of linear equations. The rules become more complex, the calculations more difficult, but the basic message remains the same -- *you can test whether variables are interrelated through a set of linear relationships by examining the variances and covariances of the variables.*

Statisticians have developed procedures for testing whether a set of variances and covariances in a covariance matrix fits a specified structure. The way structural modeling works is as follows:

1. You state the way that you believe the variables are inter-related, often with the use of a *path diagram*.
2. You work out, via some complex internal rules, what the implications of this are for the variances and covariances of the variables.
3. You test whether the variances and covariances fit this model of them.
4. Results of the statistical testing, and also parameter estimates and standard errors for the numerical coefficients in the linear equations are reported.
5. On the basis of this information, you decide whether the model seems like a good fit to your data.

There are some important, and very basic logical points to remember about this process. First, although the mathematical machinery required to perform structural equations modeling is extremely complicated, the basic logic is embodied in the above 5 steps. Below, we diagram the process.


Equation System
Covariance Structure
Modify or Reject
No
Do Data Support Covariance Structure?
Gather Data
Yes
Tentative Support for Equation System

Second, we must remember that it is unreasonable to expect a structural model to fit perfectly — for a number of reasons. A structural model with linear relations is only an approximation. The world is unlikely to be linear. Indeed, the true relations between variables are probably nonlinear. Moreover, many of the statistical assumptions are somewhat questionable as well. The real question is not so much, "Does the model fit perfectly?" but rather, "Does it fit well enough to be a useful approximation to reality, and a reasonable explanation of the trends in our data?"

Third, we must remember that simply because a model fits the data well does not mean that the model is necessarily correct. One cannot prove that a model is true — to assert this is the fallacy of affirming the consequent. For example, we could say "If Joe is a cat, Joe has hair." However, "Joe has hair" does not imply Joe is a cat. Similarly, we can say that "If a certain causal model is true, it will fit the data." However, the model fitting the data does not necessarily imply the model is the correct one. There may be another model that fits the data equally well.

Structural Equation Modeling and the Path Diagram

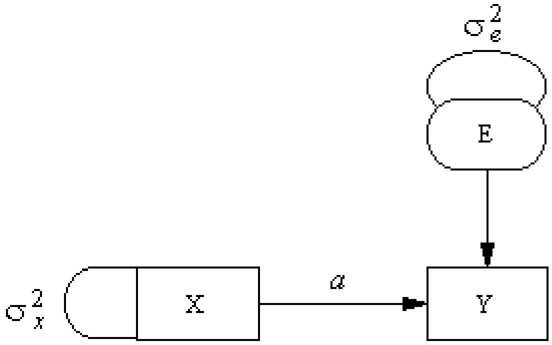
Path Diagrams play a fundamental role in structural modeling. Path diagrams are like flowcharts. They show variables interconnected with lines that are used to indicate causal flow.

One can think of a path diagram as a device for showing which variables cause changes in other variables. However, path diagrams *need not* be thought of strictly in this way. They may also be given a narrower, more specific interpretation.

Consider the classic linear regression equation

$Y = aX + e$

Any such equation may be represented in a path diagram as follows:



Such diagrams establish a simple isomorphism. All variables in the equation system are placed in the diagram, either in boxes or ovals. Each equation is represented on the diagram as follows: All independent variables (the variables on the right side of an equation) have *arrows* pointing to the dependent variable. The weighting coefficient is placed above the arrow. The above diagram shows a simple linear equation system and its path diagram representation.

Notice that, besides representing the linear equation relationships with arrows, the diagrams also contain some additional aspects. First, the variances of the independent variables, which we must know in order to test the structural relations model, are shown on the diagrams using curved lines without arrowheads attached. We refer to such lines as wires. Second, some variables are represented in ovals, others in rectangular boxes. Manifest variables are placed in boxes in the path diagram. Latent variables are placed in an oval or circle. For example, the variable *E* in the above diagram can be thought of as a linear regression residual when Y is predicted from X. Such a residual is not observed directly, but calculated from Y and X, so we treat it as a *latent* variable and place it in an oval.

The example discussed above is an extremely simple one. Generally, we are interested in testing models that are much more complicated than these. As the equation systems we examine become increasingly complicated, so do the *covariance structures* they imply.

Ultimately, the complexity can become so bewildering that we lose sight of some very basic principles. For one thing the train of reasoning which supports testing causal models with linear structural equations testing has several weak links. The variables may be non-linear. They may be linearly related for reasons unrelated to what we commonly view as causality. The ancient adage, "correlation is not causation" remains true, even if the correlation is complex and multivariate. What causal modeling *does* allow us to do is examine the extent to which data fail to agree with one reasonably viable consequence of a model of causality. If the linear equations system isomorphic to the path diagram does fit the data well, it is encouraging, but hardly proof of the truth of the causal model.

Although path diagrams can be used to represent causal flow in a system of variables, they need not imply such a causal flow. Such diagrams may be viewed as simply an isomorphic representation of a linear equations system. As such, they can convey linear relationships when no causal relations are assumed. Hence, although one *might* interpret the diagram in the above figure to mean that "X causes Y," the diagram can also be interpreted as a visual representation of the linear regression relationship between X and Y.

Chapter 38
Survival/Failure Time Analysis

- General Information
- Censored Observations
- Analytic Techniques
- Life Table Analysis
    - Number of Cases at Risk
    - Proportion Failing
    - Proportion surviving
    - Cumulative Proportion Surviving (Survival Function)
    - Probability Density
    - Hazard rate
    - Median survival time
    - Required sample sizes
- Distribution Fitting
    - General Introduction
    - Estimation
    - Goodness-of-fit
    - Plots
- Kaplan-Meier Product-Limit Estimator
- Comparing Samples
    - General Introduction
    - Available tests
    - Choosing a two-sample test
    - Multiple sample test
    - Unequal proportions of censored data
- Regression Models
    - General Introduction
    - Cox's Proportional Hazard Model
    - Cox's Proportional Hazard Model with Time-Dependent Covariates
    - Exponential Regression
    - Normal and Log-Normal Regression
    - Stratified Analyses

General Information

These techniques were primarily developed in the medical and biological sciences, but they are also widely used in the social and economic sciences, as well as in engineering (reliability and failure time analysis).

Imagine that you are a researcher in a hospital who is studying the effectiveness of a new treatment for a generally terminal disease. The major variable of interest is the number of days that the respective patients survive. In principle, one could use the standard parametric and nonparametric statistics for describing the average survival, and for comparing the new treatment with traditional methods (see *Basic Statistics* and *Nonparametrics and Distribution Fitting*). However, at the end of the study there will be patients who survived over the entire study period, in particular among those patients who entered the hospital (and the research project) late in the study; there will be other patients with whom we will have lost contact. Surely, one would not want to exclude all of those patients from the study by declaring them to be missing data (since most of them are "survivors" and, therefore, they reflect on the success of the new treatment method). Those observations, which contain only partial information are called censored observations (e.g., "patient A survived at least 4 months before he moved away and we lost contact;" the term censoring was first used by Hald, 1949).

Censored Observations

In general, censored observations arise whenever the dependent variable of interest represents the time to a terminal event, and the duration of the study is limited in time. Censored observations may occur in a number of different areas of research. For example, in the social sciences we may study the "survival" of marriages, high school drop-out rates (time to drop-out), turnover in organizations, etc. In each case, by the end of the study period, some subjects will still be married, will not have dropped out, or are still working at the same company; thus, those subjects represent censored observations.

In economics we may study the "survival" of new businesses or the "survival" times of products such as automobiles. In quality control research, it is common practice to study the "survival" of parts under stress (failure time analysis).

Analytic Techniques

Essentially, the methods offered in *Survival Analysis* address the same research questions as many of the other procedures; however, all methods in *Survival Analysis* will handle censored data. The *life table, survival distribution*, and *Kaplan-Meier* survival function estimation are all descriptive methods for estimating the distribution of survival times from a sample. Several techniques are available for comparing the survival in two or more groups. Finally, *Survival Analysis* offers several regression models for estimating the relationship of (multiple) continuous variables to survival times.

Life Table Analysis

The most straightforward way to describe the survival in a sample is to compute the *Life Table*. The life table technique is one of the oldest methods for analyzing survival (failure time) data (e.g., see Berkson & Gage, 1950; Cutler & Ederer, 1958; Gehan, 1969). This table can be thought of as an "enhanced" frequency distribution table. The distribution of survival times is divided into a certain number of intervals. For each interval we can then compute the number and proportion of cases or objects that entered the respective interval "alive," the number and proportion of cases that failed in the respective interval (i.e., number of terminal events, or number of cases that "died"), and the number of cases that were lost or censored in the respective interval.

Based on those numbers and proportions, several additional statistics can be computed:

- Number of Cases at Risk
- Proportion Failing
- Proportion surviving
- Cumulative Proportion Surviving (Survival Function)
- Probability Density
- Hazard rate
- Median survival time
- Required sample sizes

Number of Cases at Risk. This is the number of cases that entered the respective interval alive, minus half of the number of cases lost or censored in the respective interval.

Proportion Failing. This proportion is computed as the ratio of the number of cases failing in the respective interval, divided by the number of cases at risk in the interval.

Proportion Surviving. This proportion is computed as 1 minus the proportion failing.

Cumulative Proportion Surviving (Survival Function). This is the cumulative proportion of cases surviving up to the respective interval. Since the probabilities of survival are assumed to be independent across the intervals, this probability is computed by multiplying out the probabilities of survival across all previous intervals. The resulting function is also called the *survivorship* or *survival function*.

Probability Density. This is the estimated probability of failure in the respective interval, computed per unit of time, that is:

$$F_i = (P_i-P_{i+1}) /h_i$$

In this formula, F_i is the respective probability density in the i'th interval, P_i is the estimated cumulative proportion surviving at the beginning of the i'th interval (at the end of interval i-1), P_{i+1} is the cumulative proportion surviving at the end of the i'th interval, and h_i is the width of the respective interval.

Hazard Rate. The hazard rate (the term was first used by Barlow, 1963) is defined as the probability per time unit that a case that has survived to the beginning of the respective interval will fail in that interval. Specifically, it is computed as the number of failures per time units in the respective interval, divided by the average number of surviving cases at the mid-point of the interval.

Median Survival Time. This is the survival time at which the cumulative survival function is equal to *0.5*. Other percentiles (25th and 75th percentile) of the cumulative survival function can be computed accordingly. Note that the 50th percentile (median) for the cumulative survival function is usually not the same as the point in time up to which 50% of the sample survived. (This would only be the case if there were no censored observations prior to this time).

Required Sample Sizes. In order to arrive at reliable estimates of the three major functions (survival, probability density, and hazard) and their standard errors at each time interval the minimum recommended sample size is 30.

Distribution Fitting

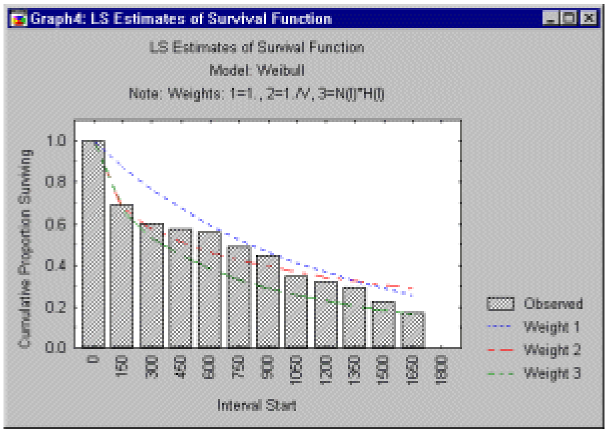
- General Introduction
- Estimation
- Goodness-of-fit
- Plots

General Introduction. In summary, the life table gives us a good indication of the distribution of failures over time. However, for predictive purposes it is often desirable to understand the shape of the underlying survival function in the population. The major distributions that have been proposed for modeling survival or failure times are the exponential (and linear exponential) distribution, the Weibull distribution of extreme events, and the Gompertz distribution.

Estimation. The parameter estimation procedure (for estimating the parameters of the theoretical survival functions) is essentially a least squares linear regression algorithm (see Gehan & Siddiqui, 1973). A linear regression algorithm can be used because all four theoretical distributions can be "made linear" by appropriate transformations. Such transformations sometimes produce different variances for the residuals at different times, leading to biased estimates.

Goodness-of-Fit. Given the parameters for the different distribution functions and the respective model, we can compute the likelihood of the data. One can also compute the likelihood of the data under the null model, that is, a model that allows for different hazard rates in each interval. Without going into details, these two likelihoods can be compared via an incremental *Chi-square* test statistic. If this *Chi-square* is statistically significant, then we conclude that the respective theoretical distribution fits the data significantly worse than the null model; that is, we reject the respective distribution as a model for our data.

Plots. You can produce plots of the survival function, hazard, and probability density for the observed data and the respective theoretical distributions. These plots provide a quick visual check of the goodness-of-fit of the theoretical distribution. The example plot below shows an observed survivorship function and the fitted Weibull distribution.



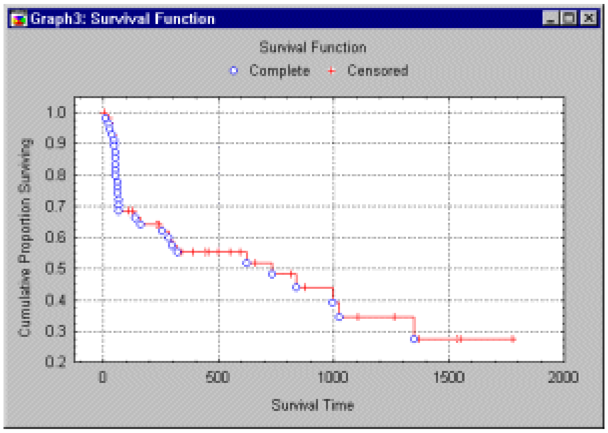
Specifically, the three lines in this plot denote the theoretical distributions that resulted from three different estimation procedures (least squares and two methods of weighted least squares).

Kaplan-Meier Product-Limit Estimator

Rather than classifying the observed survival times into a life table, we can estimate the survival function directly from the continuous survival or failure times. Intuitively, imagine that we create a life table so that each time interval contains exactly one case. Multiplying out the survival probabilities across the "intervals" (i.e., for each single observation) we would get for the survival function:

$$S(t) = \prod_{j=1}^{t} [(n-j)/(n-j+1)]^{\delta(j)}$$

In this equation, *$S(t)$* is the estimated survival function, *n* is the total number of cases, and $\prod$ denotes the multiplication (geometric sum) across all cases less than or equal to *t*; δ *(j)* is a constant that is either *1* if the *j*'th case is uncensored (complete), and *0* if it is censored. This estimate of the survival function is also called the *product-limit estimator*, and was first proposed by Kaplan and Meier (1958). An example plot of this function is shown below.



The advantage of the Kaplan-Meier Product-Limit method over the life table method for analyzing survival and failure time data is that the resulting estimates do not depend on the grouping of the data (into a certain number of time intervals). Actually, the Product-Limit method and the life table method are identical if the intervals of the life table contain at most one observation.

Comparing Samples

- General Introduction
- Available tests
- Choosing a two-sample test
- Multiple sample test
- Unequal proportions of censored data

General Introduction. One can compare the survival or failure times in two or more samples. In principle, because survival times are not normally distributed, nonparametric tests that are based on the *rank ordering* of survival times should be applied. A wide range of nonparametric tests can be used in order to compare survival times; however, the tests cannot "handle" censored observations.

Available Tests. The following five different (mostly nonparametric) tests for censored data are available: Gehan's generalized Wilcoxon test, the Cox-Mantel test, the Cox's *F* test , the log-rank test, and Peto and Peto's generalized Wilcoxon test. A nonparametric test for the comparison of multiple groups is also available. Most of these tests are accompanied by appropriate *z*- values (values of the standard normal distribution); these *z*-values can be used to test for the statistical significance of any differences between groups. However, note that most of these tests will only yield reliable results with fairly large samples sizes; the small sample "behavior" is less well understood.

Choosing a Two Sample Test. There are no widely accepted guidelines concerning which test to use in a particular situation. Cox's *F* test tends to be more powerful than Gehan's generalized Wilcoxon test when:

1. Sample sizes are small (i.e., *n* per group less than 50);
2. If samples are from an exponential or Weibull;
3. If there are no censored observations (see Gehan & Thomas, 1969).

Lee, Desu, and Gehan (1975) compared Gehan's test to several alternatives and showed that the Cox-Mantel test and the log-rank test are more powerful (regardless of censoring) when the samples are drawn from a population that follows an exponential or Weibull distribution; under those conditions there is little difference between the Cox-Mantel test and the log-rank test. Lee (1980) discusses the power of different tests in greater detail.

Multiple Sample Test. There is a multiple-sample test that is an extension (or generalization) of Gehan's generalized Wilcoxon test, Peto and Peto's generalized Wilcoxon test, and the log-rank test. First, a score is assigned to each survival time using Mantel's procedure (Mantel, 1967); next a *Chi- square* value is computed based on the sums (for each group) of this score. If only two groups are specified, then this test is equivalent to Gehan's generalized Wilcoxon test, and the computations will default to that test in this case.

Unequal Proportions of Censored Data. When comparing two or more groups it is very important to examine the number of censored observations in each group. Particularly in medical research, censoring can be the result of, for example, the application of different treatments: patients who get better faster or get worse as the result of a treatment may be more likely to drop out of the study, resulting in different numbers of censored observations in each group. Such systematic censoring may greatly bias the results of comparisons.

Regression Models

- General Introduction
- Cox's Proportional Hazard Model
- Cox's Proportional Hazard Model with Time-Dependent Covariates
- Exponential Regression
- Normal and Log-Normal Regression
- Stratified Analyses

General Introduction

A common research question in medical, biological, or engineering (failure time) research is to determine whether or not certain continuous (independent) variables are correlated with the survival or failure times. There are two major reasons why this research issue cannot be addressed via straightforward multiple regression techniques (as available in Multiple Regression): First, the dependent variable of interest (survival/failure time) is most likely not normally distributed -- a serious violation of an assumption for ordinary least squares multiple regression. Survival times usually follow an exponential or Weibull distribution. Second, there is the problem of censoring, that is, some observations will be incomplete.

Cox's Proportional Hazard Model

The proportional hazard model is the most general of the regression models because it is not based on any assumptions concerning the nature or shape of the underlying survival distribution. The model assumes that the underlying hazard *rate* (rather than survival time) is a function of the independent variables (covariates); no assumptions are made about the nature or shape of the hazard function. Thus, in a sense, Cox's regression model may be considered to be a nonparametric method. The model may be written as:

$$h\{(t), (z_1, z_2, ..., z_m)\} = h_0(t)*\exp(b_1*z_1 + ... + b_m*z_m)$$

where $h(t,...)$ denotes the resultant hazard, given the values of the m covariates for the respective case (z_1, z_2, ..., z_m) and the respective survival time (t). The term $h_0(t)$ is called the *baseline hazard*; it is the hazard for the respective individual when all independent variable values are equal to zero. We can linearize this model by dividing both sides of the equation by $h_0(t)$ and then taking the natural logarithm of both sides:

$$\log[h\{(t), (z...)\}/h_0(t)] = b_1*z_1 + ... + b_m*z_m$$

We now have a fairly "simple" linear model that can be readily estimated.

Assumptions. While no assumptions are made about the shape of the underlying hazard function, the model equations shown above do imply two assumptions. First, they specify a multiplicative relationship between the underlying hazard function and the log-linear function of the covariates. This assumption is also called the *proportionality assumption*. In practical terms, it is assumed that, given two observations with different values for the independent variables, the ratio of the hazard functions for those two observations does not depend on time. The second assumption of course, is that there is a log-linear relationship between the independent variables and the underlying hazard function.

Cox's Proportional Hazard Model with Time-Dependent Covariates

An assumption of the proportional hazard model is that the hazard function for an individual (i.e., observation in the analysis) depends on the values of the covariates and the value of the baseline hazard. Given two individuals with particular values for the covariates, the ratio of the estimated hazards over time will be constant -- hence the name of the method: the *proportional hazard* model. The validity of this assumption may often be questionable. For example, age is often included in studies of physical health. Suppose you studied survival after surgery. It is likely, that age is a more important predictor of risk immediately after surgery, than some time after the surgery (after initial recovery). In accelerated life testing one sometimes uses a stress covariate (e.g., amount of voltage) that is slowly increased over time until failure occurs (e.g., until the electrical insulation fails; see Lawless, 1982, page 393). In this case, the impact of the covariate is clearly dependent on time. The user can specify arithmetic expressions to define covariates as functions of several variables and survival time.

Testing the Proportionality Assumption. As indicated by the previous examples, there are many applications where it is likely that the proportionality assumption does not hold. In that case, one can explicitly define covariates as functions of time. For example, the analysis of a data set presented by Pike (1966) consists of survival times for two groups of rats that had been exposed to a carcinogen (see also Lawless, 1982, page 393, for a similar example). Suppose that *z* is a grouping variable with codes *1* and *0* to denote whether or not the respective rat was exposed. One could then fit the proportional hazard model:

$$h(t,z) = h_0(t)*\exp\{b_1*z + b_2*[z*\log(t)-5.4]\}$$

Thus, in this model the conditional hazard at time t is a function of (1) the baseline hazard h_0, (2) the covariate z, and (3) of z times the logarithm of time. Note that the constant *5.4* is used here for scaling purposes only: the mean of the logarithm of the survival times in this data set is equal to *5.4*. In other words, the conditional hazard at each point in time is a function of the covariate and time; thus, the effect of the covariate on survival is dependent on time; hence the name *time-dependent covariate*. This model allows one to specifically test the proportionality assumption. If parameter b_2 is statistically significant (e.g., if it is at least twice as large as its standard error), then one can conclude that, indeed, the effect of the covariate z on survival is dependent on time, and, therefore, that the proportionality assumption does not hold.

Exponential Regression

Basically, this model assumes that the survival time distribution is exponential, and contingent on the values of a set of independent variables (*zi*). The rate parameter of the exponential distribution can then be expressed as:

$$S(z) = \exp(a + b_1*z_1 + b_2*z_2 + ... + b_m*z_m)$$

$S(z)$ denotes the survival times, a is a constant, and the b_i's are the regression parameters.

Goodness-of-fit. The *Chi-square* goodness-of-fit value is computed as a function of the log-likelihood for the model with all parameter estimates (*L1*), and the log-likelihood of the model in which all covariates are forced to 0 (zero; *L0*). If this *Chi-square* value is significant, we reject the null hypothesis and assume that the independent variables are significantly related to survival times.

Standard exponential order statistic. One way to check the exponentiality assumption of this model is to plot the residual survival times against the standard exponential order statistic *theta*. If the exponentiality assumption is met, then all points in this plot will be arranged roughly in a straight line.

Normal and Log-Normal Regression

In this model, it is assumed that the survival times (or log survival times) come from a normal distribution; the resulting model is basically identical to the ordinary multiple regression model, and may be stated as:

$t = a + b_1*z_1 + b_2*z_2 + ... + b_m*z_m$

where *t* denotes the survival times. For log-normal regression, *t* is replaced by its natural logarithm. The normal regression model is particularly useful because many data sets can be transformed to yield approximations of the normal distribution. Thus, in a sense this is the most general fully parametric model (as opposed to Cox's proportional hazard model which is non-parametric), and estimates can be obtained for a variety of different underlying survival distributions.

Goodness-of-fit. The *Chi-square* value is computed as a function of the log-likelihood for the model with all independent variables (L1), and the log-likelihood of the model in which all independent variables are forced to 0 (zero, L0).

Stratified Analyses

The purpose of a stratified analysis is to test the hypothesis whether identical regression models are appropriate for different groups, that is, whether the relationships between the independent variables and survival are identical in different groups. To perform a stratified analysis, one must first fit the respective regression model separately within each group. The sum of the log-likelihoods from these analyses represents the log-likelihood of the model with different regression coefficients (and intercepts where appropriate) in different groups. The next step is to fit the requested regression model to all data in the usual manner (i.e., ignoring group membership), and compute the log-likelihood for the overall fit. The difference between the log-likelihoods can then be tested for statistical significance (via the *Chi-square* statistic).

Chapter 39
Text Mining

- Introductory Overview
- Some Typical Applications for Text Mining
- Approaches to Text Mining
- Issues and Considerations for "Numericizing" Text
- Transforming Word Frequencies
- Latent Semantic Indexing via Singular Value Decomposition
- Incorporating Text Mining Results in Data Mining Projects

Text Mining Introductory Overview

The purpose of Text Mining is to process unstructured (textual) information, extract meaningful numeric indices from the text, and, thus, make the information contained in the text accessible to the various data mining (statistical and machine learning) algorithms. Information can be extracted to derive summaries for the words contained in the documents or to compute summaries for the documents based on the words contained in them. Hence, you can analyze words, clusters of words used in documents, etc., or you could analyze documents and determine similarities between them or how they are related to other variables of interest in the data mining project. In the most general terms, text mining will "turn text into numbers" (meaningful indices), which can then be incorporated in other analyses such as predictive data mining projects, the application of unsupervised learning methods (clustering), etc. These methods are described and discussed in great detail in the comprehensive overview work by Manning and Schütze (2002), and for an in-depth treatment of these and related topics as well as the history of this approach to text mining, we highly recommend that source.

Some Typical Applications for Text Mining

Unstructured text is very common, and in fact may represent the majority of information available to a particular research or data mining project.

Analyzing open-ended survey responses. In survey research (e.g., marketing), it is not uncommon to include various open-ended questions pertaining to the topic under investigation. The idea is to permit respondents to express their "views" or opinions without constraining them to particular dimensions or a particular response format. This may yield insights into customers' views and opinions that might otherwise not be discovered when relying solely on structured questionnaires designed by "experts." For example, you may discover a certain set of words or terms that are commonly used by respondents to describe the pro's and con's of a product or service (under investigation), suggesting common misconceptions or confusion regarding the items in the study.

Automatic processing of messages, emails, etc. Another common application for text mining is to aid in the automatic classification of texts. For example, it is possible to "filter" out automatically most undesirable "junk email" based on certain terms or words that are not likely to appear in legitimate messages, but instead identify undesirable electronic mail. In this manner, such messages can automatically be discarded. Such automatic systems for classifying electronic messages can also be useful in applications where messages need to be routed (automatically) to the most appropriate department or agency; e.g., email messages with complaints or petitions to a municipal authority are automatically routed to the appropriate departments; at the same time, the emails are screened for inappropriate or obscene messages, which are automatically returned to the sender with a request to remove the offending words or content.

Analyzing warranty or insurance claims, diagnostic interviews, etc. In some business domains, the majority of information is collected in open-ended, textual form. For example, warranty claims or initial medical (patient) interviews can be summarized in brief narratives, or when you take your automobile to a service station for repairs, typically, the attendant will write some notes about the problems that you report and what you believe needs to be fixed. Increasingly, those notes are collected electronically, so those types of narratives are readily available for input into text mining algorithms. This information can then be usefully exploited to, for example, identify common clusters of problems and complaints on certain automobiles, etc. Likewise, in the medical field, open-ended descriptions by patients of their own symptoms might yield useful clues for the actual medical diagnosis.

Investigating competitors by crawling their web sites. Another type of potentially very useful application is to automatically process the contents of Web pages in a particular domain. For example, you could go to a Web page, and begin "crawling" the links you find there to process all Web pages that are referenced. In this manner, you could automatically derive a list of terms and documents available at that site, and hence quickly determine the most important terms and features that are described. It is easy to see how these capabilities could efficiently deliver valuable business intelligence about the activities of competitors.

Approaches to Text Mining

To reiterate, text mining can be summarized as a process of "numericizing" text. At the simplest level, all words found in the input documents will be indexed and counted in order to compute a table of documents and words, i.e., a matrix of frequencies that enumerates the number of times that each word occurs in each document. This basic process can be further refined to exclude certain common words such as "the" and "a" (stop word lists) and to combine different grammatical forms of the same words such as "traveling," "traveled," "travel," etc. (stemming). However, once a table of (unique) words (terms) by documents has been derived, all standard statistical and data mining techniques can be applied to derive dimensions or clusters of words or documents, or to identify "important" words or terms that best predict another outcome variable of interest.

Using well-tested methods and understanding the results of text mining. Once a data matrix has been computed from the input documents and words found in those documents, various well-known analytic techniques can be used for further processing those data including methods for clustering, factoring, or predictive data mining (see, for example, Manning and Schütze, 2002).

"Black-box" approaches to text mining and extraction of concepts. There are text mining applications which offer "black-box" methods to extract "deep meaning" from documents with little human effort (to first read and understand those documents). These text mining applications rely on proprietary algorithms for presumably extracting "concepts" from text, and may even claim to be able to summarize large numbers of text documents automatically, retaining the core and most important meaning of those documents. While there are numerous algorithmic approaches to extracting "meaning from documents," this type of technology is very much still in its infancy, and the aspiration to provide meaningful automated summaries of large numbers of documents may forever remain elusive. We urge skepticism when using such algorithms because 1) if it is not clear to the user how those algorithms work, it cannot possibly be clear how to interpret the results of those algorithms, and 2) the methods used in those programs are not open to scrutiny, for example by the academic community and peer review and, hence, one simply doesn't know how well they might perform in different domains. As a final thought on this subject, you may consider this concrete example: Try the various automated translation services available via the Web that can translate entire paragraphs of text from one language into another. Then translate some text, even simple text, from your native language to some other language and back, and review the results. Almost every time, the attempt to translate even short sentences to other languages and back while retaining the original meaning of the sentence produces humorous rather than accurate results. This illustrates the difficulty of automatically interpreting the meaning of text.

Text mining as document search. There is another type of application that is often described and referred to as "text mining" - the automatic search of large numbers of documents based on key words or key phrases. This is the domain of, for example, the popular internet search engines that have been developed over the last decade to provide efficient access to Web pages with certain content. While this is obviously an important type of application with many uses in any organization that needs to search very large document repositories based on varying criteria, it is very different from what has been described here.

Issues and Considerations for "Numericizing" Text

Large numbers of small documents vs. small numbers of large documents. Examples of scenarios using large numbers of small or moderate sized documents were given earlier (e.g., analyzing warranty or insurance claims, diagnostic interviews, etc.). On the other hand, if your intent is to extract "concepts" from only a few documents that are very large (e.g., two lengthy books), then statistical analyses are generally less powerful because the "number of cases" (documents) in this case is very small while the "number of variables" (extracted words) is very large.

Excluding certain characters, short words, numbers, etc. Excluding numbers, certain characters, or sequences of characters, or words that are shorter or longer than a certain number of letters can be done before the indexing of the input documents starts. You may also want to exclude "rare words," defined as those that only occur in a small percentage of the processed documents.

Include lists, exclude lists (stop-words). Specific list of words to be indexed can be defined; this is useful when you want to search explicitly for particular words, and classify the input documents based on the frequencies with which those words occur. Also, "stop-words," i.e., terms that are to be excluded from the indexing can be defined. Typically, a default list of English stop words includes "the", "a", "of", "since," etc, i.e., words that are used in the respective language very frequently, but communicate very little unique information about the contents of the document.

Synonyms and phrases. Synonyms, such as "sick" or "ill", or words that are used in particular phrases where they denote unique meaning can be combined for indexing. For example, "Microsoft Windows" might be such a phrase, which is a specific reference to the computer operating system, but has nothing to do with the common use of the term "Windows" as it might, for example, be used in descriptions of home improvement projects.

Stemming algorithms. An important pre-processing step before indexing of input documents begins is the stemming of words. The term "stemming" refers to the reduction of words to their roots so that, for example, different grammatical forms or declinations of verbs are identified and indexed (counted) as the same word. For example, stemming will ensure that both "traveling" and "traveled" will be recognized by the text mining program as the same word.

Support for different languages. Stemming, synonyms, the letters that are permitted in words, etc. are highly language dependent operations. Therefore, support for different languages is important.

Transforming Word Frequencies

Once the input documents have been indexed and the initial word frequencies (by document) computed, a number of additional transformations can be performed to summarize and aggregate the information that was extracted.

Log-frequencies. First, various transformations of the frequency counts can be performed. The raw word or term frequencies generally reflect on how salient or important a word is in each document. Specifically, words that occur with greater frequency in a document are better descriptors of the contents of that document. However, it is not reasonable to assume that the word counts themselves are proportional to their importance as descriptors of the documents. For example, if a word occurs 1 time in document A, but 3 times in document B, then it is not necessarily reasonable to conclude that this word is 3 times as important a descriptor of document B as compared to document A. Thus, a common transformation of the raw word frequency counts (wf) is to compute:

$f(wf) = 1 + \log(wf)$, for $wf > 0$

This transformation will "dampen" the raw frequencies and how they will affect the results of subsequent computations.

Binary frequencies. Likewise, an even simpler transformation can be used that enumerates whether a term is used in a document; i.e.:

$f(wf) = 1$, for $wf > 0$

The resulting documents-by-words matrix will contain only 1s and 0s to indicate the presence or absence of the respective words. Again, this transformation will dampen the effect of the raw frequency counts on subsequent computations and analyses.

Inverse document frequencies. Another issue that you may want to consider more carefully and reflect in the indices used in further analyses are the relative document frequencies (df) of different words. For example, a term such as "guess" may occur frequently in all documents, while another term such as "software" may only occur in a few. The reason is that one might make "guesses" in various contexts, regardless of the specific topic, while "software" is a more semantically focused term that is only likely to occur in documents that deal with computer software. A common and very useful transformation that reflects both the specificity of words (document frequencies) as well as the overall frequencies of their occurrences (word frequencies) is the so-called inverse document frequency (for the i'th word and j'th document):

$$idf(i,j) = \begin{cases} 0 & \text{if } wf_{i,j} = 0 \\ (1 + \log(wf_{i,j}))\log \dfrac{N}{df_i} & \text{if } wf_{i,j} \geq 1 \end{cases}$$

In this formula (see also formula 15.5 in Manning and Schütze, 2002), *N* is the total number of documents, and *dfi* is the document frequency for the *i*'th word (the number of documents that include this word). Hence, it can be seen that this formula includes both the dampening of the simple word frequencies via the log function (described above), and also includes a weighting factor that evaluates to 0 if the word occurs in all documents *(log(N/N=1)=0)*, and to the maximum value when a word only occurs in a single document *(log(N/1)=log(N))*. It can easily be seen how this transformation will create indices that both reflect the relative frequencies of occurrences of words, as well as their semantic specificities over the documents included in the analysis.

Latent Semantic Indexing via Singular Value Decomposition

As described above, the most basic result of the initial indexing of words found in the input documents is a frequency table with simple counts, i.e., the number of times that different words occur in each input document. Usually, one would transform those raw counts to indices that better reflect the (relative) "importance" of words and/or their semantic specificity in the context of the set of input documents (see the discussion of inverse document frequencies, above).

A common analytic tool for interpreting the "meaning" or "semantic space" described by the words that were extracted, and hence by the documents that were analyzed, is to create a mapping of the word and documents into a common space, computed from the word frequencies or transformed word frequencies (e.g., inverse document frequencies). In general, here is how it works:

Suppose you indexed a collection of customer reviews of their new automobiles (e.g., for different makes and models). You may find that every time a review includes the word "gas-mileage," it also includes the term "economy." Further, when reports include the word "reliability" they also include the term "defects" (e.g., make reference to "no defects"). However, there is no consistent pattern regarding the use of the terms "economy" and "reliability," i.e., some documents include either one or both. In other words, these four words "gas-mileage" and "economy," and "reliability" and "defects," describe two independent dimensions - the first having to do with the overall operating cost of the vehicle, the other with the quality and workmanship. The idea of latent semantic indexing is to identify such underlying dimensions (of "meaning"), into which the words and documents can be mapped. As a result, we may identify the underlying (latent) themes described or discussed in the input documents, and also identify the documents that mostly deal with economy, reliability, or both. Hence, we want to map the extracted words or terms and input documents into a common latent semantic space.

Singular value decomposition. The use of singular value decomposition in order to extract a common space for the variables and cases (observations) is used in various statistical techniques, most notably in *Correspondence Analysis*. The technique is also closely related to Principal Components Analysis and *Factor Analysis*. In general, the purpose of this technique is to reduce the overall dimensionality of the input matrix (number of input documents by number of extracted words) to a lower-dimensional space, where each consecutive dimension represents the largest degree of variability (between words and documents) possible. Ideally, you might identify the two or three most salient dimensions, accounting for most of the variability (differences) between the words and documents and, hence, identify the latent semantic space that organizes the words and documents in the analysis. In some way, once such dimensions can be identified, you have extracted the underlying "meaning" of what is contained (discussed, described) in the documents.

Incorporating Text Mining Results in Data Mining Projects

After significant (e.g., frequent) words have been extracted from a set of input documents, and/or after singular value decomposition has been applied to extract salient semantic dimensions, typically the next and most important step is to use the extracted information in a data mining project.

Graphics (visual data mining methods). Depending on the purpose of the analyses, in some instances the extraction of semantic dimensions alone can be a useful outcome if it clarifies the underlying structure of what is contained in the input documents. For example, a study of new car owners' comments about their vehicles may uncover the salient dimensions in the minds of those drivers when they think about or consider their automobile (or how they "feel" about it). For marketing research purposes, that in itself can be a useful and significant result. You can use the graphics (e.g., 2D scatterplots or 3D scatterplots) to help you visualize and identify the semantic space extracted from the input documents.

Clustering and factoring. You can use cluster analysis methods to identify groups of documents (e.g., vehicle owners who described their new cars), to identify groups of similar input texts. This type of analysis also could be extremely useful in the context of market research studies, for example of new car owners. You can also use *Factor Analysis* and Principal Components and Classification Analysis (to factor analyze words or documents).

Predictive data mining. Another possibility is to use the raw or transformed word counts as predictor variables in predictive data mining projects.

Chapter 40
Time Series Analysis

- General Introduction
- Two Main Goals
- Identifying Patterns in Time Series Data
  - Systematic pattern and random noise
  - Two general aspects of time series patterns
  - Trend Analysis
  - Analysis of Seasonality
- ARIMA (Box & Jenkins) and Autocorrelations
  - General Introduction
  - Two Common Processes
  - ARIMA Methodology
  - Identification Phase
  - Parameter Estimation
  - Evaluation of the Model
- Interrupted Time Series
- Exponential Smoothing
  - General Introduction
  - Simple Exponential Smoothing
  - Choosing the Best Value for Parameter a (alpha)
  - Indices of Lack of Fit (Error)
  - Seasonal and Non-seasonal Models With or Without Trend
- Seasonal Decomposition (Census I)
  - General Introduction
  - Computations
- X-11 Census method II seasonal adjustment
  - Seasonal Adjustment: Basic Ideas and Terms
  - The Census II Method
  - Results Tables Computed by the X-11 Method
  - Specific Description of all Results Tables Computed by the X-11 Method
- Distributed Lags Analysis
  - General Purpose
  - General Model
  - Almon Distributed Lag
- Single Spectrum (Fourier) Analysis
- Cross-spectrum Analysis
  - General Introduction
  - Basic Notation and Principles
  - Results for Each Variable
  - The Cross-periodogram, Cross-density, Quadrature-density, and Cross-amplitude

    - Squared Coherency, Gain, and Phase Shift
    - How the Example Data were Created
- Spectrum Analysis - Basic Notations and Principles
    - Frequency and Period
    - The General Structural Model
    - A Simple Example
    - Periodogram
    - The Problem of Leakage
    - Padding the Time Series
    - Tapering
    - Data Windows and Spectral Density Estimates
    - Preparing the Data for Analysis
    - Results when no Periodicity in the Series Exists
- Fast Fourier Transformations
    - General Introduction
    - Computation of FFT in Time Series

In the following topics, we will first review techniques used to identify patterns in time series data (such as smoothing and curve fitting techniques and autocorrelations), then we will introduce a general class of models that can be used to represent time series data and generate predictions (autoregressive and moving average models). Finally, we will review some simple but commonly used modeling and forecasting techniques based on linear regression. For more information on these topics, see the topic name below.

General Introduction

In the following topics, we will review techniques that are useful for analyzing time series data, that is, sequences of measurements that follow non-random orders. Unlike the analyses of random samples of observations that are discussed in the context of most other statistics, the analysis of time series is based on the assumption that successive values in the data file represent consecutive measurements taken at equally spaced time intervals.

Detailed discussions of the methods described in this section can be found in Anderson (1976), Box and Jenkins (1976), Kendall (1984), Kendall and Ord (1990), Montgomery, Johnson, and Gardiner (1990), Pankratz (1983), Shumway (1988), Vandaele (1983), Walker (1991), and Wei (1989).

Two Main Goals

There are two main goals of time series analysis: (a) identifying the nature of the phenomenon represented by the sequence of observations, and (b) forecasting (predicting future values of the time series variable). Both of these goals require that the pattern of observed time series data is identified and more or less formally described. Once the pattern is established, we can interpret and integrate it with other data (i.e., use it in our theory of the investigated phenomenon, e.g., sesonal commodity prices). Regardless of the depth of our understanding and the validity of our interpretation (theory) of the phenomenon, we can extrapolate the identified pattern to predict future events.

Identifying Patterns in Time Series Data

- Systematic pattern and random noise
- Two general aspects of time series patterns
- Trend Analysis
- Analysis of Seasonality

For more information on simple autocorrelations (introduced in this section) and other auto correlations, see Anderson (1976), Box and Jenkins (1976), Kendall (1984), Pankratz (1983), and Vandaele (1983). See also:

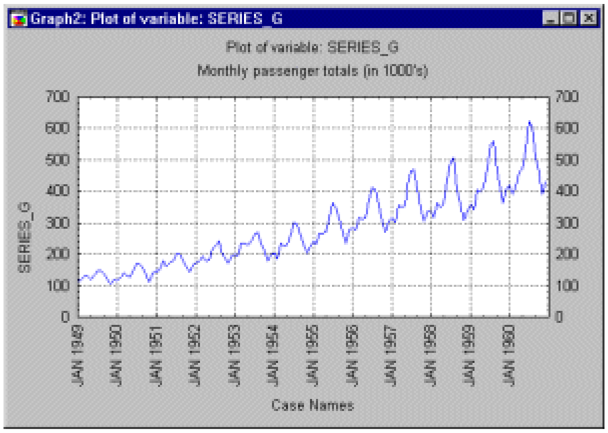
- ARIMA (Box & Jenkins) and Autocorrelations
- Interrupted Time Series
- Exponential Smoothing
- Seasonal Decomposition (Census I)
- X-11 Census method II seasonal adjustment
- X-11 Census method II result tables
- Distributed Lags Analysis
- Single Spectrum (Fourier) Analysis
- Cross-spectrum Analysis
- Basic Notations and Principles
- Fast Fourier Transformations

Systematic Pattern and Random Noise

As in most other analyses, in time series analysis it is assumed that the data consist of a systematic pattern (usually a set of identifiable components) and random noise (error) which usually makes the pattern difficult to identify. Most time series analysis techniques involve some form of filtering out noise in order to make the pattern more salient.

Two General Aspects of Time Series Patterns

Most time series patterns can be described in terms of two basic classes of components: trend and seasonality. The former represents a general systematic linear or (most often) nonlinear component that changes over time and does not repeat or at least does not repeat within the time range captured by our data (e.g., a plateau followed by a period of exponential growth). The latter may have a formally similar nature (e.g., a plateau followed by a period of exponential growth), however, it repeats itself in systematic intervals over time. Those two general classes of time series components may coexist in real-life data. For example, sales of a company can rapidly grow over years but they still follow consistent seasonal patterns (e.g., as much as 25% of yearly sales each year are made in December, whereas only 4% in August).



This general pattern is well illustrated in a "classic" *Series G* data set (Box and Jenkins, 1976, p. 531) representing monthly international airline passenger totals (measured in thousands) in twelve consecutive years from 1949 to 1960 (see example data file *G.sta* and graph above). If you plot the successive observations (months) of airline passenger totals, a clear, almost linear trend emerges, indicating that the airline industry enjoyed a steady growth over the years (approximately 4 times more passengers traveled in 1960 than in 1949). At the same time, the monthly figures will follow an almost identical pattern each year (e.g., more people travel during holidays then during any other time of the year). This example data file also illustrates a very common general type of pattern in time series data, where the amplitude of the seasonal changes increases with the overall trend (i.e., the variance is correlated with the mean over the segments of the series). This pattern which is called *multiplicative seasonality* indicates that the relative amplitude of seasonal changes is constant over time, thus it is related to the trend.

Trend Analysis

There are no proven "automatic" techniques to identify trend components in the time series data; however, as long as the trend is monotonous (consistently increasing or decreasing) that part of data analysis is typically not very difficult. If the time series data contain considerable error, then the first step in the process of trend identification is smoothing.

Smoothing. Smoothing always involves some form of local averaging of data such that the nonsystematic components of individual observations cancel each other out. The most common technique is *moving average* smoothing which replaces each element of the series by either the simple or weighted average of *n* surrounding elements, where *n* is the width of the smoothing "window" (see Box & Jenkins, 1976; Velleman & Hoaglin, 1981). Medians can be used instead of means. The main advantage of median as compared to moving average smoothing is that its results are less biased by outliers (within the smoothing window). Thus, if there are outliers in the data (e.g., due to measurement errors), median smoothing typically produces smoother or at least more "reliable" curves than moving average based on the same window width. The main disadvantage of median smoothing is that in the absence of clear outliers it may produce more "jagged" curves than moving average and it does not allow for weighting.

In the relatively less common cases (in time series data), when the measurement error is very large, the *distance weighted least squares smoothing* or *negative exponentially weighted smoothing* techniques can be used. All those methods will filter out the noise and convert the data into a smooth curve that is relatively unbiased by outliers (see the respective sections on each of those methods for more details). Series with relatively few and systematically distributed points can be smoothed with *bicubic splines*.

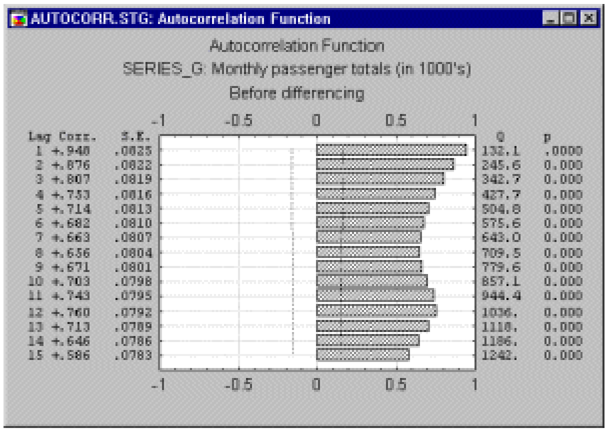
Fitting a function. Many monotonous time series data can be adequately approximated by a linear function; if there is a clear monotonous nonlinear component, the data first need to be transformed to remove the nonlinearity. Usually a logarithmic, exponential, or (less often) polynomial function can be used.

Analysis of Seasonality

Seasonal dependency (seasonality) is another general component of the time series pattern. The concept was illustrated in the example of the airline passengers data above. It is formally defined as correlational dependency of order *k* between each *i*'th element of the series and the (*i*-*k*)'th element (Kendall, 1976) and measured by autocorrelation (i.e., a correlation between the two terms); *k* is usually called the *lag*. If the measurement error is not too large, seasonality can be visually identified in the series as a pattern that repeats every *k* elements.

Autocorrelation correlogram. Seasonal patterns of time series can be examined via correlograms. The correlogram (autocorrelogram) displays graphically and numerically the autocorrelation function (*ACF*), that is, serial correlation coefficients (and their standard errors) for consecutive lags in a specified range of lags (e.g., 1 through 30). Ranges of two standard errors for each lag are usually marked in correlograms but typically the size of auto correlation is of more interest than its reliability (see *Elementary Concepts*) because we are usually interested only in very strong (and thus highly significant) autocorrelations.

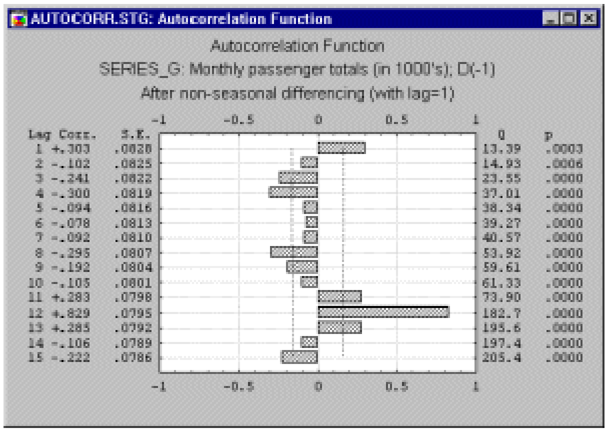
Examining correlograms. While examining correlograms one should keep in mind that autocorrelations for consecutive lags are formally dependent. Consider the following example. If the first element is closely related to the second, and the second to the third, then the first element must also be somewhat related to the third one, etc. This implies that the pattern of serial dependencies can change considerably after removing the first order auto correlation (i.e., after differencing the series with a lag of 1).



AUTOCORR.STG: Autocorrelation Function

Autocorrelation Function
SERIES_G: Monthly passenger totals (in 1000's)
Before differencing

```
Lag Corr.   S.E.       Q       p
 1 +.940   .0825   132.1   .0000
 2 +.876   .0822   245.6   0.000
 3 +.807   .0819   342.7   0.000
 4 +.753   .0816   427.7   0.000
 5 +.714   .0813   504.8   0.000
 6 +.682   .0810   575.6   0.000
 7 +.663   .0807   643.0   0.000
 8 +.656   .0804   709.5   0.000
 9 +.671   .0801   779.6   0.000
10 +.703   .0798   857.1   0.000
11 +.743   .0795   944.4   0.000
12 +.760   .0792   1036.   0.000
13 +.713   .0789   1118.   0.000
14 +.646   .0786   1186.   0.000
15 +.586   .0783   1242.   0.000
```



AUTOCORR.STG: Autocorrelation Function

Autocorrelation Function
SERIES_G: Monthly passenger totals (in 1000's); D(-1)
After non-seasonal differencing (with lag=1)

```
Lag Corr.   S.E.       Q       p
 1 +.303   .0828   13.39   .0003
 2 -.102   .0825   14.93   .0006
 3 -.241   .0822   23.55   .0000
 4 -.300   .0819   37.01   .0000
 5 -.094   .0816   38.34   .0000
 6 -.078   .0813   39.27   .0000
 7 -.092   .0810   40.57   .0000
 8 -.295   .0807   53.92   .0000
 9 -.192   .0804   59.61   .0000
10 -.105   .0801   61.33   .0000
11 +.283   .0798   73.90   .0000
12 +.829   .0795   182.7   .0000
13 +.285   .0792   195.6   .0000
14 -.106   .0789   197.4   .0000
15 -.222   .0786   205.4   .0000
```

Partial autocorrelations. Another useful method to examine serial dependencies is to examine the partial autocorrelation function (*PACF*) - an extension of autocorrelation, where the dependence on the intermediate elements (those *within* the lag) is removed. In other words the partial autocorrelation is similar to autocorrelation, except that when calculating it, the (auto) correlations with all the elements within the lag are partialled out (Box & Jenkins, 1976; see also McDowall, McCleary, Meidinger, & Hay, 1980). If a lag of 1 is specified (i.e., there are no intermediate elements within the lag), then the partial autocorrelation is equivalent to auto correlation. In a sense, the partial autocorrelation provides a "cleaner" picture of serial dependencies for individual lags (not confounded by other serial dependencies).

Removing serial dependency. Serial dependency for a particular lag of *k* can be removed by differencing the series, that is converting each *i*'th element of the series into its difference from the (*i*-*k*)"th element. There are two major reasons for such transformations.

First, one can identify the hidden nature of seasonal dependencies in the series. Remember that, as mentioned in the previous paragraph, autocorrelations for consecutive lags are interdependent. Therefore, removing some of the autocorrelations will change other auto correlations, that is, it may eliminate them or it may make some other seasonalities more apparent.

The other reason for removing seasonal dependencies is to make the series *stationary* which is necessary for ARIMA and other techniques.

ARIMA

- General Introduction
- Two Common Processes
- ARIMA Methodology
- Identification Phase
- Parameter Estimation
- Evaluation of the Model

For more information on *Time Series* methods, see also:

□ Identifying Patterns in Time Series Data
□ Interrupted Time Series
□ Exponential Smoothing
□ Seasonal Decomposition (Census I)
□ X-11 Census method II seasonal adjustment
□ X-11 Census method II result tables
□ Distributed Lags Analysis
□ Single Spectrum (Fourier) Analysis
□ Cross-spectrum Analysis
□ Basic Notations and Principles
□ Fast Fourier Transformations

General Introduction

The modeling and forecasting procedures discussed in the *Identifying Patterns in Time Series Data*, involved knowledge about the mathematical model of the process. However, in real-life research and practice, patterns of the data are unclear, individual observations involve considerable error, and we still need not only to uncover the hidden patterns in the data but also generate forecasts. The ARIMA methodology developed by Box and Jenkins (1976) allows us to do just that; it has gained enormous popularity in many areas and research practice confirms its power and flexibility (Hoff, 1983; Pankratz, 1983;

Vandaele, 1983). However, because of its power and flexibility, ARIMA is a complex technique; it is not easy to use, it requires a great deal of experience, and although it often produces satisfactory results, those results depend on the researcher's level of expertise (Bails & Peppers, 1982). The following sections will introduce the basic ideas of this methodology. For those interested in a brief, applications-oriented (non- mathematical), introduction to ARIMA methods, we recommend McDowall, McCleary, Meidinger, and Hay (1980).

Two Common Processes

Autoregressive process. Most time series consist of elements that are serially dependent in the sense that one can estimate a coefficient or a set of coefficients that describe consecutive elements of the series from specific, time-lagged (previous) elements. This can be summarized in the equation:

$$x_t = \xi + \phi_1 * x_{(t-1)} + \phi_2 * x_{(t-2)} + \phi_3 * x_{(t-3)} + \ldots + \varepsilon$$

Where:

ξ is a constant (intercept), and
ϕ_1, ϕ_2, ϕ_3 are the autoregressive model parameters.

Put in words, each observation is made up of a random error component (random shock, ε) and a linear combination of prior observations.

Stationarity requirement. Note that an autoregressive process will only be stable if the parameters are within a certain range; for example, if there is only one autoregressive parameter then is must fall within the interval of $-1 < \phi < 1$. Otherwise, past effects would accumulate and the values of successive x_t' s would move towards infinity, that is, the series would not be stationary. If there is more than one autoregressive parameter, similar (general) restrictions on the parameter values can be defined (e.g., see Box & Jenkins, 1976; Montgomery, 1990).

Moving average process. Independent from the autoregressive process, each element in the series can also be affected by the past error (or random shock) that cannot be accounted for by the autoregressive component, that is:

$$x_t = \mu + \varepsilon_t - \theta_1 * \varepsilon_{(t-1)} - \theta_2 * \varepsilon_{(t-2)} - \theta_3 * \varepsilon_{(t-3)} - \ldots$$

Where:

μ is a constant, and
θ_1, θ_2, θ_3 are the moving average model parameters.

Put in words, each observation is made up of a random error component (random shock, ε) and a linear combination of prior random shocks.

Invertibility requirement. Without going into too much detail, there is a "duality" between the moving average process and the autoregressive process (e.g., see Box & Jenkins, 1976; Montgomery, Johnson, & Gardiner, 1990), that is, the moving average equation above can be rewritten (*inverted*) into an autoregressive form (of infinite order). However, analogous to the stationarity condition described above, this can only be done if the moving average parameters follow certain conditions, that is, if the model is *invertible*. Otherwise, the series will not be stationary.

ARIMA Methodology

Autoregressive moving average model. The general model introduced by Box and Jenkins (1976) includes autoregressive as well as moving average parameters, and explicitly includes differencing in the formulation of the model. Specifically, the three types of parameters in the model are: the autoregressive parameters (*p*), the number of differencing passes (*d*), and moving average parameters (*q*). In the notation introduced by Box and Jenkins, models are summarized as ARIMA (*p, d, q*); so, for example, a model described as (0, 1, 2) means that it contains 0 (zero) autoregressive (*p*) parameters and 2 moving average (*q*) parameters which were computed for the series after it was differenced once.

Identification. As mentioned earlier, the input series for ARIMA needs to be stationary, that is, it should have a constant mean, variance, and autocorrelation through time. Therefore, usually the series first needs to be differenced until it is stationary (this also often requires log transforming the data to stabilize the variance). The number of times the series needs to be differenced to achieve stationarity is reflected in the *d* parameter (see the previous paragraph). In order to determine the necessary level of differencing, one should examine the plot of the data and autocorrelogram. Significant changes in level (strong upward or downward changes) usually require first order non seasonal (lag=1) differencing; strong changes of slope usually require second order non seasonal differencing. Seasonal patterns require respective seasonal differencing (see below). If the estimated autocorrelation coefficients decline slowly at longer lags, first order differencing is usually needed. However, one should keep in mind that some time series may require little or no differencing, and that *over differenced* series produce less stable coefficient estimates.

At this stage (which is usually called *Identification* phase, see below) we also need to decide how many autoregressive (*p*) and moving average (*q*) parameters are necessary to yield an effective but still *parsimonious* model of the process (*parsimonious* means that it has the fewest parameters and greatest number of degrees of freedom among all models that fit the data). In practice, the numbers of the *p* or *q* parameters very rarely need to be greater than 2 (see below for more specific recommendations).

Estimation and Forecasting. At the next step (*Estimation*), the parameters are estimated (using function minimization procedures, see below; for more information on minimization procedures see also *Nonlinear Estimation*), so that the sum of squared residuals is minimized. The estimates of the parameters are used in the last stage

(*Forecasting*) to calculate new values of the series (beyond those included in the input data set) and confidence intervals for those predicted values. The estimation process is performed on transformed (differenced) data; before the forecasts are generated, the series needs to be *integrated* (integration is the inverse of differencing) so that the forecasts are expressed in values compatible with the input data. This automatic integration feature is represented by the letter I in the name of the methodology (ARIMA = Auto-Regressive Integrated Moving Average).

The constant in ARIMA models. In addition to the standard autoregressive and moving average parameters, ARIMA models may also include a constant, as described above. The interpretation of a (statistically significant) constant depends on the model that is fit. Specifically, (1) if there are no autoregressive parameters in the model, then the expected value of the constant is μ, the mean of the series; (2) if there are autoregressive parameters in the series, then the constant represents the intercept. If the series is differenced, then the constant represents the mean or intercept of the differenced series; For example, if the series is differenced once, and there are no autoregressive parameters in the model, then the constant represents the mean of the differenced series, and therefore the *linear trend slope* of the un-differenced series.

Identification

Number of parameters to be estimated. Before the estimation can begin, we need to decide on (identify) the specific number and type of ARIMA parameters to be estimated. The major tools used in the identification phase are plots of the series, correlograms of auto correlation (ACF), and partial autocorrelation (PACF). The decision is not straightforward and in less typical cases requires not only experience but also a good deal of experimentation with alternative models (as well as the technical parameters of ARIMA). However, a majority of empirical time series patterns can be sufficiently approximated using one of the 5 basic models that can be identified based on the shape of the autocorrelogram (ACF) and partial auto correlogram (PACF). The following brief summary is based on practical recommendations of Pankratz (1983); for additional practical advice, see also Hoff (1983), McCleary and Hay (1980), McDowall, McCleary, Meidinger, and Hay (1980), and Vandaele (1983). Also, note that since the number of parameters (to be estimated) of each kind is almost never greater than 2, it is often practical to try alternative models on the same data.

1. *One autoregressive (p) parameter*: ACF - exponential decay; PACF - spike at lag 1, no correlation for other lags.
2. *Two autoregressive (p) parameters*: ACF - a sine-wave shape pattern or a set of exponential decays; PACF - spikes at lags 1 and 2, no correlation for other lags.
3. *One moving average (q) parameter*: ACF - spike at lag 1, no correlation for other lags; PACF - damps out exponentially.
4. *Two moving average (q) parameters*: ACF - spikes at lags 1 and 2, no correlation for other lags; PACF - a sine-wave shape pattern or a set of exponential decays.
5. *One autoregressive (p) and one moving average (q) parameter*: ACF - exponential decay starting at lag 1; PACF - exponential decay starting at lag 1.

Seasonal models. Multiplicative seasonal ARIMA is a generalization and extension of the method introduced in the previous paragraphs to series in which a pattern repeats seasonally over time. In addition to the non-seasonal parameters, seasonal parameters for a specified lag (established in the identification phase) need to be estimated. Analogous to the simple ARIMA parameters, these are: seasonal autoregressive (*ps*), seasonal differencing (*ds*), and seasonal moving average parameters (*qs*). For example, the model (*0,1,2*)(*0,1,1*) describes a model that includes no autoregressive parameters, 2 regular moving average parameters and 1 seasonal moving average parameter, and these parameters were computed for the series after it was differenced once with lag 1, and once seasonally differenced. The seasonal lag used for the seasonal parameters is usually determined during the identification phase and must be explicitly specified.

The general recommendations concerning the selection of parameters to be estimated (based on ACF and PACF) also apply to seasonal models. The main difference is that in seasonal series, ACF and PACF will show sizable coefficients at multiples of the seasonal lag (in addition to their overall patterns reflecting the non seasonal components of the series).

Parameter Estimation

There are several different methods for estimating the parameters. All of them should produce very similar estimates, but may be more or less efficient for any given model. In general, during the parameter estimation phase a function minimization algorithm is used (the so-called *quasi-Newton* method; refer to the description of the *Nonlinear Estimation*method) to maximize the likelihood (probability) of the observed series, given the parameter values. In practice, this requires the calculation of the (conditional) sums of squares (SS) of the residuals, given the respective parameters. Different methods have been proposed to compute the SS for the residuals: (1) the approximate maximum likelihood method according to McLeod and Sales (1983), (2) the approximate maximum likelihood method with backcasting, and (3) the exact maximum likelihood method according to Melard (1984).

Comparison of methods. In general, all methods should yield very similar parameter estimates. Also, all methods are about equally efficient in most real-world time series applications. However, method *1* above, (approximate maximum likelihood, no backcasts) is the fastest, and should be used in particular for very long time series (e.g., with more than 30,000 observations). Melard's exact maximum likelihood method (number *3* above) may also become inefficient when used to estimate parameters for seasonal models with long seasonal lags (e.g., with yearly lags of 365 days). On the other hand, you should always use the approximate maximum likelihood method first in order to establish initial parameter estimates that are very close to the actual final values; thus, usually only a few iterations with the exact maximum likelihood method (*3*, above) are necessary to finalize the parameter estimates.

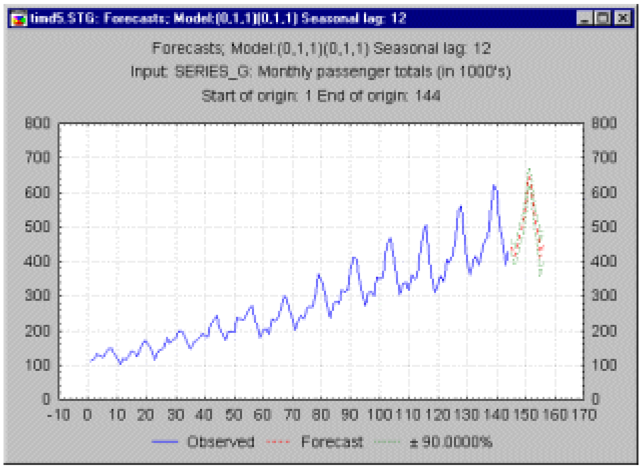
Parameter standard errors. For all parameter estimates, you will compute so-called *asymptotic standard errors*. These are computed from the matrix of second-order partial derivatives that is approximated via finite differencing (see also the respective discussion in *Nonlinear Estimation*).

Penalty value. As mentioned above, the estimation procedure requires that the (conditional) sums of squares of the ARIMA residuals be minimized. If the model is inappropriate, it may happen during the iterative estimation process that the parameter estimates become very large, and, in fact, invalid. In that case, it will assign a very large value (a so-called *penalty value*) to the SS. This usually "entices" the iteration process to move the parameters away from invalid ranges. However, in some cases even this strategy fails, and you may see on the screen (during the *Estimation procedure*) very large values for the SS in consecutive iterations. In that case, carefully evaluate the appropriateness of your model. If your model contains many parameters, and perhaps an intervention component (see below), you may try again with different parameter start values.

Evaluation of the Model

Parameter estimates. You will report approximate *t* values, computed from the parameter standard errors (see above). If not significant, the respective parameter can in most cases be dropped from the model without affecting substantially the overall fit of the model.

Other quality criteria. Another straightforward and common measure of the reliability of the model is the accuracy of its forecasts generated based on partial data so that the forecasts can be compared with known (original) observations.



However, a good model should not only provide sufficiently accurate forecasts, it should also be parsimonious and produce statistically independent residuals that contain only noise and no systematic components (e.g., the correlogram of residuals should not reveal any serial dependencies). A good test of the model is (a) to plot the residuals and inspect them for any systematic trends, and (b) to examine the autocorrelogram of residuals (there should be no serial dependency between residuals).

Analysis of residuals. The major concern here is that the residuals are systematically distributed across the series (e.g., they could be negative in the first part of the series and approach zero in the second part) or that they contain some serial dependency which may suggest that the ARIMA model is inadequate. The analysis of ARIMA residuals constitutes an important test of the model. The estimation procedure assumes that the residual are not (auto-) correlated and that they are normally distributed.

Limitations. The ARIMA method is appropriate only for a time series that is stationary (i.e., its mean, variance, and autocorrelation should be approximately constant through time) and it is recommended that there are at least 50 observations in the input data. It is also assumed that the values of the estimated parameters are constant throughout the series.

Interrupted Time Series ARIMA

A common research questions in time series analysis is whether an outside event affected subsequent observations. For example, did the implementation of a new economic policy improve economic performance; did a new anti-crime law affect subsequent crime rates; and so on. In general, we would like to evaluate the impact of one or more discrete events on the values in the time series. This type of interrupted time series analysis is described in detail in McDowall, McCleary, Meidinger, & Hay (1980). McDowall, et. al., distinguish between three major types of impacts that are possible: (1) permanent abrupt, (2) permanent gradual, and (3) abrupt temporary. See also:

- Identifying Patterns in Time Series Data
- ARIMA
- Exponential Smoothing
- Seasonal Decomposition (Census I)
- X-11 Census method II seasonal adjustment
- X-11 Census method II result tables
- Distributed Lags Analysis
- Single Spectrum (Fourier) Analysis
- Cross-spectrum Analysis
- Basic Notations and Principles
- Fast Fourier Transformations

Exponential Smoothing

- General Introduction
- Simple Exponential Smoothing
- Choosing the Best Value for Parameter a (alpha)
- Indices of Lack of Fit (Error)

- Seasonal and Non-seasonal Models With or Without Trend

See also:

- Identifying Patterns in Time Series Data
- ARIMA (Box & Jenkins) and Autocorrelations
- Interrupted Time Series
- Seasonal Decomposition (Census I)
- X-11 Census method II seasonal adjustment
- X-11 Census method II result tables
- Distributed Lags Analysis
- Single Spectrum (Fourier) Analysis
- Cross-spectrum Analysis
- Basic Notations and Principles
- Fast Fourier Transformations

General Introduction

Exponential smoothing has become very popular as a forecasting method for a wide variety of time series data. Historically, the method was independently developed by Brown and Holt. Brown worked for the US Navy during World War II, where his assignment was to design a tracking system for fire-control information to compute the location of submarines. Later, he applied this technique to the forecasting of demand for spare parts (an inventory control problem). He described those ideas in his 1959 book on inventory control. Holt's research was sponsored by the Office of Naval Research; independently, he developed exponential smoothing models for constant processes, processes with linear trends, and for seasonal data.

Gardner (1985) proposed a "unified" classification of exponential smoothing methods. Excellent introductions can also be found in Makridakis, Wheelwright, and McGee (1983), Makridakis and Wheelwright (1989), Montgomery, Johnson, & Gardiner (1990).

Simple Exponential Smoothing

A simple and pragmatic model for a time series would be to consider each observation as consisting of a constant (*b*) and an error component ε(epsilon), that is: $X_t = b + \varepsilon_t$. The constant *b* is relatively stable in each segment of the series, but may change slowly over time. If appropriate, then one way to isolate the true value of *b*, and thus the systematic or predictable part of the series, is to compute a kind of moving average, where the current and immediately preceding ("younger") observations are assigned greater weight than the respective older observations. Simple exponential smoothing accomplishes exactly such weighting, where exponentially smaller weights are assigned to older observations. The specific formula for simple exponential smoothing is:

$$S_t = \alpha *X_t + (1-\alpha)*S_{t-1}$$

When applied recursively to each successive observation in the series, each new smoothed value (forecast) is computed as the weighted average of the current observation and the previous smoothed observation; the previous smoothed observation was computed in turn from the previous observed value and the smoothed value before the previous observation, and so on. Thus, in effect, each smoothed value is the weighted average of the previous observations, where the weights decrease exponentially depending on the value of parameter α (alpha). If α is equal to 1 (one) then the previous observations are ignored entirely; if α is equal to 0 (zero), then the current observation is ignored entirely, and the smoothed value consists entirely of the previous smoothed value (which in turn is computed from the smoothed observation before it, and so on; thus all smoothed values will be equal to the initial smoothed value S_0). Values of α in-between will produce intermediate results.

Even though significant work has been done to study the theoretical properties of (simple and complex) exponential smoothing (e.g., see Gardner, 1985; Muth, 1960; see also McKenzie, 1984, 1985), the method has gained popularity mostly because of its usefulness as a forecasting tool. For example, empirical research by Makridakis *et al*. (1982, Makridakis, 1983), has shown simple exponential smoothing to be the best choice for one-period-ahead forecasting, from among 24 other time series methods and using a variety of accuracy measures (see also Gross and Craig, 1974, for additional empirical evidence). Thus, regardless of the theoretical model for the process underlying the observed time series, simple exponential smoothing will often produce quite accurate forecasts.

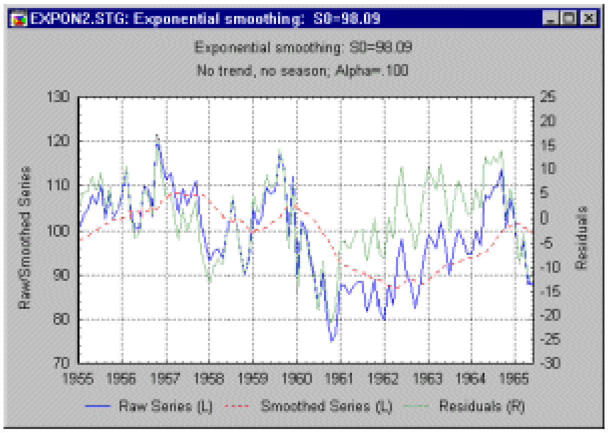
Choosing the Best Value for Parameter α (alpha)

Gardner (1985) discusses various theoretical and empirical arguments for selecting an appropriate smoothing parameter. Obviously, looking at the formula presented above, α should fall into the interval between 0 (zero) and 1 (although, see Brenner *et al.*, 1968, for an ARIMA perspective, implying $0<\alpha<2$). Gardner (1985) reports that among practitioners, an α smaller than .30 is usually recommended. However, in the study by Makridakis *et al*. (1982), α values above .30 frequently yielded the best forecasts. After reviewing the literature on this topic, Gardner (1985) concludes that it is best to estimate an optimum α from the data (see below), rather than to "guess" and set an artificially low value.

Estimating the best α value from the data. In practice, the smoothing parameter is often chosen by a *grid search* of the parameter space; that is, different solutions for α are tried starting, for example, with $\alpha = 0.1$ to $\alpha = 0.9$, with increments of 0.1. Then α is chosen so as to produce the smallest sums of squares (or mean squares) for the residuals (i.e., observed values minus one-step-ahead forecasts; this mean squared error is also referred to as *ex post* mean squared error, *ex post* MSE for short).

Indices of Lack of Fit (Error)

The most straightforward way of evaluating the accuracy of the forecasts based on a particular α value is to simply plot the observed values and the one-step-ahead forecasts. This plot can also include the residuals (scaled against the right *Y*-axis), so that regions of better or worst fit can also easily be identified.



This visual check of the accuracy of forecasts is often the most powerful method for determining whether or not the current exponential smoothing model fits the data. In addition, besides the *ex post* MSE criterion (see previous paragraph), there are other statistical measures of error that can be used to determine the optimum α parameter (see Makridakis, Wheelwright, and McGee, 1983):

Mean error: The mean error (ME) value is simply computed as the average error value (average of observed minus one-step-ahead forecast). Obviously, a drawback of this measure is that positive and negative error values can cancel each other out, so this measure is not a very good indicator of overall fit.

Mean absolute error: The mean absolute error (MAE) value is computed as the average *absolute* error value. If this value is 0 (zero), the fit (forecast) is perfect. As compared to the mean *squared* error value, this measure of fit will "de-emphasize" outliers, that is, unique or rare large error values will affect the MAE less than the MSE value.

Sum of squared error (SSE), Mean squared error. These values are computed as the sum (or average) of the squared error values. This is the most commonly used lack-of-fit indicator in statistical fitting procedures.

Percentage error (PE). All the above measures rely on the actual error value. It may seem reasonable to rather express the lack of fit in terms of the *relative* deviation of the one-step-ahead forecasts from the observed values, that is, relative to the magnitude of the observed values. For example, when trying to predict monthly sales that may fluctuate widely (e.g., seasonally) from month to month, we may be satisfied if our prediction "hits the target" with about ±10% accuracy. In other words, the absolute errors may be not so much of interest as are the relative errors in the forecasts. To assess the relative error, various indices have been proposed (see Makridakis, Wheelwright, and McGee, 1983). The first one, the percentage error value, is computed as:

$PE_t = 100*(X_t - F_t)/X_t$

where X_t is the observed value at time t, and F_t is the forecasts (smoothed values).

Mean percentage error (MPE). This value is computed as the average of the PE values.

Mean absolute percentage error (MAPE). As is the case with the mean error value (ME, see above), a mean percentage error near 0 (zero) can be produced by large positive and negative percentage errors that cancel each other out. Thus, a better measure of relative overall fit is the mean *absolute* percentage error. Also, this measure is usually more meaningful than the mean squared error. For example, knowing that the average forecast is "off" by ±5% is a useful result in and of itself, whereas a mean squared error of 30.8 is not immediately interpretable.

Automatic search for best parameter. A quasi-Newton function minimization procedure (the same as in ARIMA is used to minimize either the mean squared error, mean absolute error, or mean absolute percentage error. In most cases, this procedure is more efficient than the grid search (particularly when more than one parameter must be determined), and the optimum α parameter can quickly be identified.

The first smoothed value S_0. A final issue that we have neglected up to this point is the problem of the initial value, or how to start the smoothing process. If you look back at the formula above, it is evident that one needs an S_0 value in order to compute the smoothed value (forecast) for the first observation in the series. Depending on the choice of the α parameter (i.e., when α is close to zero), the initial value for the smoothing process can affect the quality of the forecasts for many observations. As with most other aspects of exponential smoothing it is recommended to choose the initial value that produces the best forecasts. On the other hand, in practice, when there are many leading observations prior to a crucial actual forecast, the initial value will not affect that forecast by much, since its effect will have long "faded" from the smoothed series (due to the exponentially decreasing weights, the older an observation the less it will influence the forecast).

Seasonal and Non-seasonal Models With or Without Trend

The discussion above in the context of simple exponential smoothing introduced the basic procedure for identifying a smoothing parameter, and for evaluating the goodness-of-fit of a model. In addition to simple exponential smoothing, more complex models have been developed to accommodate time series with seasonal and trend components. The general idea here is that forecasts are not only computed from consecutive previous observations (as in simple exponential smoothing), but an independent (smoothed) trend and seasonal component can be added. Gardner (1985) discusses the different models in terms of seasonality (none, additive, or multiplicative) and trend (none, linear, exponential, or damped).

Additive and multiplicative seasonality. Many time series data follow recurring seasonal patterns. For example, annual sales of toys will probably peak in the months of

November and December, and perhaps during the summer (with a much smaller peak) when children are on their summer break. This pattern will likely repeat every year, however, the relative amount of increase in sales during December may slowly change from year to year. Thus, it may be useful to smooth the seasonal component independently with an extra parameter, usually denoted as δ (*delta*). Seasonal components can be additive in nature or multiplicative. For example, during the month of December the sales for a particular toy may increase by 1 million dollars every year. Thus, we could *add* to our forecasts for every December the amount of 1 million dollars (over the respective annual average) to account for this seasonal fluctuation. In this case, the seasonality is *additive*. Alternatively, during the month of December the sales for a particular toy may increase by 40%, that is, increase by a *factor* of 1.4. Thus, when the sales for the toy are generally weak, than the absolute (dollar) increase in sales during December will be relatively weak (but the percentage will be constant); if the sales of the toy are strong, than the absolute (dollar) increase in sales will be proportionately greater. Again, in this case the sales increase by a certain *factor*, and the seasonal component is thus *multiplicative* in nature (i.e., the multiplicative seasonal component in this case would be 1.4). In plots of the series, the distinguishing characteristic between these two types of seasonal components is that in the additive case, the series shows steady seasonal fluctuations, regardless of the overall level of the series; in the multiplicative case, the size of the seasonal fluctuations vary, depending on the overall level of the series.

The seasonal smoothing parameter δ. In general the one-step-ahead forecasts are computed as (for no trend models, for linear and exponential trend models a trend component is added to the model; see below):

Additive model:

$\text{Forecast}_t = S_t + I_{t-p}$

Multiplicative model:

$\text{Forecast}_t = S_t * I_{t-p}$

In this formula, S_t stands for the (simple) exponentially smoothed value of the series at time t, and I_{t-p} stands for the smoothed seasonal factor at time t minus p (the length of the season). Thus, compared to simple exponential smoothing, the forecast is "enhanced" by adding or multiplying the simple smoothed value by the predicted seasonal component. This seasonal component is derived analogous to the S_t value from simple exponential smoothing as:

Additive model:

$I_t = I_{t-p} + \delta * (1-\alpha) * e_t$

Multiplicative model:

$I_t = I_{t-p} + \delta*(1-\alpha)*e_t/S_t$

Put in words, the predicted seasonal component at time *t* is computed as the respective seasonal component in the last seasonal cycle plus a portion of the error (e_t; the observed minus the forecast value at time *t*). Considering the formulas above, it is clear that parameter δ can assume values between 0 and 1. If it is zero, then the seasonal component for a particular point in time is predicted to be identical to the predicted seasonal component for the respective time during the previous seasonal cycle, which in turn is predicted to be identical to that from the previous cycle, and so on. Thus, if δ is zero, a constant unchanging seasonal component is used to generate the one-step-ahead forecasts. If the δ parameter is equal to 1, then the seasonal component is modified "maximally" at every step by the respective forecast error (times $(1-\alpha)$, which we will ignore for the purpose of this brief introduction). In most cases, when seasonality is present in the time series, the optimum δ parameter will fall somewhere between 0 (zero) and 1(one).

Linear, exponential, and damped trend. To remain with the toy example above, the sales for a toy can show a linear upward trend (e.g., each year, sales increase by 1 million dollars), exponential growth (e.g., each year, sales increase by a factor of 1.3), or a damped trend (during the first year sales increase by 1 million dollars; during the second year the increase is only 80% over the previous year, i.e., $800,000; during the next year it is again 80% less than the previous year, i.e., $800,000 * .8 = $640,000; etc.). Each type of trend leaves a clear "signature" that can usually be identified in the series; shown below in the brief discussion of the different models are icons that illustrate the general patterns. In general, the trend factor may change slowly over time, and, again, it may make sense to smooth the trend component with a separate parameter (denoted γ [*gamma*] for linear and exponential trend models, and ϕ[*phi*] for damped trend models).

The trend smoothing parameters γ (linear and exponential trend) and ϕ(damped trend). Analogous to the seasonal component, when a trend component is included in the exponential smoothing process, an independent trend component is computed for each time, and modified as a function of the forecast error and the respective parameter. If the γ parameter is 0 (zero), than the trend component is constant across all values of the time series (and for all forecasts). If the parameter is 1, then the trend component is modified "maximally" from observation to observation by the respective forecast error. Parameter values that fall in-between represent mixtures of those two extremes. Parameter $\phi$is a trend modification parameter, and affects how strongly changes in the trend will affect estimates of the trend for subsequent forecasts, that is, how quickly the trend will be "damped" or increased.

Classical Seasonal Decomposition (Census Method 1)

- General Introduction
- Computations

See also:

- Identifying Patterns in Time Series Data
- ARIMA (Box & Jenkins) and Autocorrelations
- Interrupted Time Series
- Exponential Smoothing
- X-11 Census method II seasonal adjustment
- X-11 Census method II result tables
- Distributed Lags Analysis
- Single Spectrum (Fourier) Analysis
- Cross-spectrum Analysis
- Basic Notations and Principles
- Fast Fourier Transformations

General Introduction

Suppose you recorded the monthly passenger load on international flights for a period of 12 years (see Box & Jenkins, 1976). If you plot those data, it is apparent that (1) there appears to be a linear upwards trend in the passenger loads over the years, and (2) there is a recurring pattern or *seasonality* within each year (i.e., most travel occurs during the summer months, and a minor peak occurs during the December holidays). The purpose of the seasonal decomposition method is to isolate those components, that is, to de-compose the series into the trend effect, seasonal effects, and remaining variability. The "classic" technique designed to accomplish this decomposition is known as the *Census I* method. This technique is described and discussed in detail in Makridakis, Wheelwright, and McGee (1983), and Makridakis and Wheelwright (1989).

General model. The general idea of seasonal decomposition is straightforward. In general, a time series like the one described above can be thought of as consisting of four different components: (1) A seasonal component (denoted as S_t, where t stands for the particular point in time) (2) a trend component (T_t), (3) a cyclical component (C_t), and (4) a random, error, or irregular component (I_t). The difference between a cyclical and a seasonal component is that the latter occurs at regular (seasonal) intervals, while cyclical factors have usually a longer duration that varies from cycle to cycle. In the Census I method, the trend and cyclical components are customarily combined into a *trend-cycle component* (TC_t). The specific functional relationship between these components can assume different forms. However, two straightforward possibilities are that they combine in an *additive* or a *multiplicative* fashion:

Additive model:

$X_t = TC_t + S_t + I_t$

Multiplicative model:

$X_t = T_t * C_t * S_t * I_t$

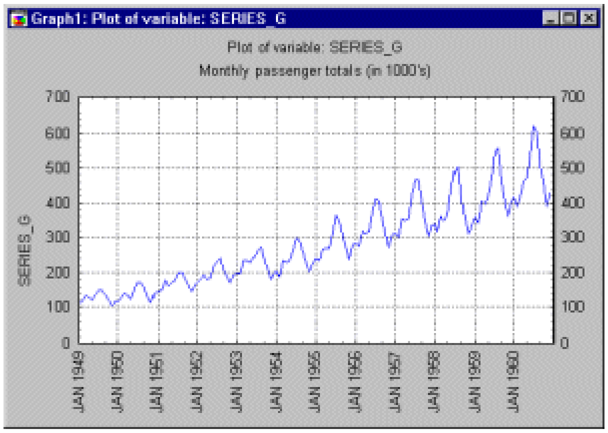
Here X_t stands for the observed value of the time series at time t. Given some *a priori* knowledge about the cyclical factors affecting the series (e.g., business cycles), the estimates for the different components can be used to compute forecasts for future observations. (However, the *Exponential smoothing* method, which can also incorporate seasonality and trend components, is the preferred technique for forecasting purposes.)

Additive and multiplicative seasonality. Let us consider the difference between an additive and multiplicative seasonal component in an example: The annual sales of toys will probably peak in the months of November and December, and perhaps during the summer (with a much smaller peak) when children are on their summer break. This seasonal pattern will likely repeat every year. Seasonal components can be additive or multiplicative in nature. For example, during the month of December the sales for a particular toy may increase by 3 million dollars every year. Thus, we could *add* to our forecasts for every December the amount of 3 million to account for this seasonal fluctuation. In this case, the seasonality is *additive*. Alternatively, during the month of December the sales for a particular toy may increase by 40%, that is, increase by a factor of 1.4. Thus, when the sales for the toy are generally weak, then the absolute (dollar) increase in sales during December will be relatively weak (but the percentage will be constant); if the sales of the toy are strong, then the absolute (dollar) increase in sales will be proportionately greater. Again, in this case the sales increase by a certain *factor*, and the seasonal component is thus *multiplicative* in nature (i.e., the multiplicative seasonal component in this case would be 1.4). In plots of series, the distinguishing characteristic between these two types of seasonal components is that in the additive case, the series shows steady seasonal fluctuations, regardless of the overall level of the series; in the multiplicative case, the size of the seasonal fluctuations vary, depending on the overall level of the series.

Additive and multiplicative trend-cycle. We can extend the previous example to illustrate the additive and multiplicative trend-cycle components. In terms of our toy example, a "fashion" *trend* may produce a steady increase in sales (e.g., a trend towards more educational toys in general); as with the seasonal component, this trend may be additive (sales increase by 3 million dollars per year) or multiplicative (sales increase by 30%, or by a factor of 1.3, annually) in nature. In addition, cyclical components may impact sales; to reiterate, a cyclical component is different from a seasonal component in that it usually is of longer duration, and that it occurs at irregular intervals. For example, a particular toy may be particularly "hot" during a summer season (e.g., a particular doll which is tied to the release of a major children's movie, and is promoted with extensive advertising). Again such a cyclical component can effect sales in an additive manner or multiplicative manner.

Computations

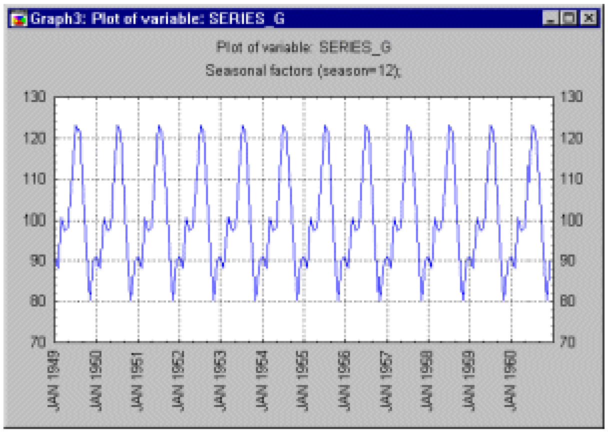
The *Seasonal Decomposition (Census I)* standard formulas are shown in Makridakis, Wheelwright, and McGee (1983), and Makridakis and Wheelwright (1989).



Moving average. First a moving average is computed for the series, with the moving average window width equal to the length of one season. If the length of the season is even, then the user can choose to use either equal weights for the moving average or unequal weights can be used, where the first and last observation in the moving average window are averaged.

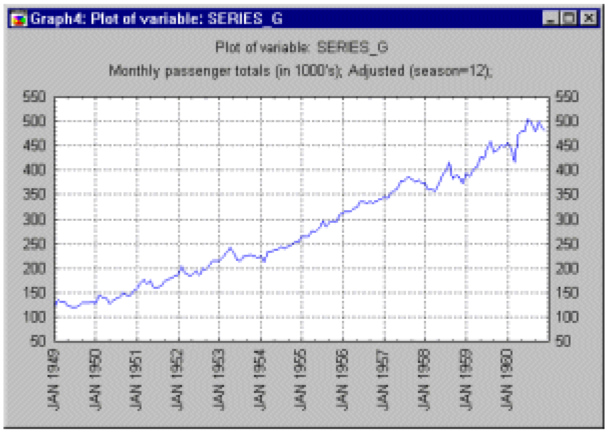
Ratios or differences. In the moving average series, all seasonal (within-season) variability will be eliminated; thus, the differences (in additive models) or ratios (in multiplicative models) of the observed and smoothed series will isolate the seasonal component (plus irregular component). Specifically, the moving average is subtracted from the observed series (for additive models) or the observed series is divided by the moving average values (for multiplicative models).

Seasonal components. The seasonal component is then computed as the average (for additive models) or medial average (for multiplicative models) for each point in the season.



(The medial average of a set of values is the mean after the smallest and largest values are excluded). The resulting values represent the (average) seasonal component of the series.

Seasonally adjusted series. The original series can be adjusted by subtracting from it (additive models) or dividing it by (multiplicative models) the seasonal component.



The resulting series is the seasonally adjusted series (i.e., the seasonal component will be removed).

Trend-cycle component. Remember that the cyclical component is different from the seasonal component in that it is usually longer than one season, and different cycles can be of different lengths. The combined trend and cyclical component can be approximated by applying to the seasonally adjusted series a 5 point (centered) weighed moving average smoothing transformation with the weights of 1, 2, 3, 2, 1.

Random or irregular component. Finally, the random or irregular (error) component can be isolated by subtracting from the seasonally adjusted series (additive models) or dividing the adjusted series by (multiplicative models) the trend-cycle component.

X-11 Census Method II Seasonal Adjustment

The general ideas of seasonal decomposition and adjustment are discussed in the context of the Census I seasonal adjustment method (*Seasonal Decomposition (Census I)*). The Census method II (2) is an extension and refinement of the simple adjustment method. Over the years, different versions of the Census method II evolved at the Census Bureau; the method that has become most popular and is used most widely in government and business is the so-called *X-11* variant of the Census method II (see Hiskin, Young, & Musgrave, 1967). Subsequently, the term *X-11* has become synonymous with this refined version of the Census method II. In addition to the documentation that can be obtained from the Census Bureau, a detailed summary of this method is also provided in Makridakis, Wheelwright, and McGee (1983) and Makridakis and Wheelwright (1989).

For more information on this method, see the following topics:

- Seasonal Adjustment: Basic Ideas and Terms
- The Census II Method
- Results Tables Computed by the X-11 Method
- Specific Description of all Results Tables Computed by the X-11 Method

For more information on other Time Series methods, see Time Series Analysis - Index and the following topics:

- Identifying Patterns in Time Series Data
- ARIMA (Box & Jenkins) and Autocorrelations
- Interrupted Time series
- Exponential Smoothing
- Seasonal Decomposition (Census I)
- Distributed Lags Analysis
- Single Spectrum (Fourier) Analysis
- Cross-spectrum Analysis
- Basic Notations and Principles
- Fast Fourier Transformations

Seasonal Adjustment: Basic Ideas and Terms.

Suppose you recorded the monthly passenger load on international flights for a period of 12 years (see Box & Jenkins, 1976). If you plot those data, it is apparent that (1) there appears to be an upwards linear trend in the passenger loads over the years, and (2) there is a recurring pattern or *seasonality* within each year (i.e., most travel occurs during the summer months, and a minor peak occurs during the December holidays). The purpose of seasonal decomposition and adjustment is to isolate those components, that is, to de-compose the series into the trend effect, seasonal effects, and remaining variability. The "classic" technique designed to accomplish this decomposition was developed in the 1920's and is also known as the *Census I* method (see the Census I overview section). This technique is also described and discussed in detail in Makridakis, Wheelwright, and McGee (1983), and Makridakis and Wheelwright (1989).

General model. The general idea of seasonal decomposition is straightforward. In general, a time series like the one described above can be thought of as consisting of four different components: (1) A seasonal component (denoted as S_t, where t stands for the particular point in time) (2) a trend component (T_t), (3) a cyclical component (C_t), and (4) a random, error, or irregular component (I_t). The difference between a cyclical and a seasonal component is that the latter occurs at regular (seasonal) intervals, while cyclical factors usually have a longer duration that varies from cycle to cycle. The trend and cyclical components are customarily combined into a *trend-cycle component* (TC_t). The specific functional relationship between these components can assume different forms. However, two straightforward possibilities are that they combine in an *additive* or a *multiplicative* fashion:

Additive Model:

$X_t = TC_t + S_t + I_t$

Multiplicative Model:

$X_t = T_t*C_t*S_t*I_t$

Where:

X_t represents the observed value of the time series at time t.

Given some *a priori* knowledge about the cyclical factors affecting the series (e.g., business cycles), the estimates for the different components can be used to compute forecasts for future observations. (However, the *Exponential smoothing* method, which can also incorporate seasonality and trend components, is the preferred technique for forecasting purposes.)

Additive and multiplicative seasonality. Consider the difference between an additive and multiplicative seasonal component in an example: The annual sales of toys will probably peak in the months of November and December, and perhaps during the summer (with a much smaller peak) when children are on their summer break. This seasonal pattern will likely repeat every year. Seasonal components can be additive or multiplicative in nature. For example, during the month of December the sales for a particular toy may increase by 3 million dollars every year. Thus, you could *add* to your forecasts for every December the amount of 3 million to account for this seasonal fluctuation. In this case, the seasonality is *additive*. Alternatively, during the month of December the sales for a particular toy may increase by 40%, that is, increase by a *factor* of 1.4. Thus, when the sales for the toy are generally weak, then the absolute (dollar) increase in sales during December will be relatively weak (but the percentage will be constant); if the sales of the toy are strong, then the absolute (dollar) increase in sales will be proportionately greater. Again, in this case the sales increase by a certain *factor*, and the seasonal component is thus *multiplicative* in nature (i.e., the multiplicative seasonal component in this case would be 1.4). In plots of series, the distinguishing characteristic between these two types of seasonal components is that in the additive case, the series shows steady seasonal fluctuations, regardless of the overall level of the series; in the multiplicative case, the size of the seasonal fluctuations vary, depending on the overall level of the series.

Additive and multiplicative trend-cycle. The previous example can be extended to illustrate the additive and multiplicative trend-cycle components. In terms of the toy example, a "fashion" *trend* may produce a steady increase in sales (e.g., a trend towards more educational toys in general); as with the seasonal component, this trend may be additive (sales increase by 3 million dollars per year) or multiplicative (sales increase by 30%, or by a factor of 1.3, annually) in nature. In addition, cyclical components may impact sales. To reiterate, a cyclical component is different from a seasonal component in that it usually is of longer duration, and that it occurs at irregular intervals. For example, a particular toy may be particularly "hot" during a summer season (e.g., a particular doll which is tied to the release of a major children's movie, and is promoted with extensive advertising). Again such a cyclical component can effect sales in an additive manner or multiplicative manner.

The Census II Method

The basic method for seasonal decomposition and adjustment outlined in the Basic Ideas and Terms topic can be refined in several ways. In fact, unlike many other time-series modeling techniques (e.g., ARIMA) which are grounded in some theoretical model of an underlying process, the *X-11* variant of the Census II method simply contains many *ad hoc* features and refinements, that over the years have proven to provide excellent estimates for many real-world applications (see Burman, 1979, Kendal & Ord, 1990, Makridakis & Wheelwright, 1989; Wallis, 1974). Some of the major refinements are listed below.

Trading-day adjustment. Different months have different numbers of days, and different numbers of trading-days (i.e., Mondays, Tuesdays, etc.). When analyzing, for example, monthly revenue figures for an amusement park, the fluctuation in the different numbers of Saturdays and Sundays (peak days) in the different months will surely contribute significantly to the variability in monthly revenues. The *X-11* variant of the Census II method allows the user to test whether such trading-day variability exists in the series, and, if so, to adjust the series accordingly.

Extreme values. Most real-world time series contain outliers, that is, extreme fluctuations due to rare events. For example, a strike may affect production in a particular month of one year. Such extreme outliers may bias the estimates of the seasonal and trend components. The *X-11* procedure includes provisions to deal with extreme values through the use of "statistical control principles," that is, values that are above or below a certain range (expressed in terms of multiples of *sigma*, the standard deviation) can be modified or dropped before final estimates for the seasonality are computed.

Multiple refinements. The refinement for outliers, extreme values, and different numbers of trading-days can be applied more than once, in order to obtain successively improved estimates of the components. The *X-11* method applies a series of successive refinements of the estimates to arrive at the final trend-cycle, seasonal, and irregular components, and the seasonally adjusted series.

Tests and summary statistics. In addition to estimating the major components of the series, various summary statistics can be computed. For example, analysis of variance tables can be prepared to test the significance of seasonal variability and trading-day variability (see above) in the series; the *X-11* procedure will also compute the percentage change from month to month in the random and trend-cycle components. As the duration or span in terms of months (or quarters for quarterly *X-11*) increases, the change in the trend-cycle component will likely also increase, while the change in the random component should remain about the same. The width of the average span at which the changes in the random component are about equal to the changes in the trend-cycle component is called the *month (quarter) for cyclical dominance*, or MCD (QCD) for short. For example, if the MCD is equal to 2 then one can infer that over a 2 month span the trend-cycle will dominate the fluctuations of the irregular (random) component. These and various other results are discussed in greater detail below.

Result Tables Computed by the X-11 Method

The computations performed by the *X-11* procedure are best discussed in the context of the results tables that are reported. The adjustment process is divided into seven major steps, which are customarily labeled with consecutive letters A through G.

A. **Prior adjustment (monthly seasonal adjustment only).** Before any seasonal adjustment is performed on the monthly time series, various prior user- defined adjustments can be incorporated. The user can specify a second series that contains prior adjustment factors; the values in that series will either be subtracted (additive model) from the original series, or the original series will be divided by these values (multiplicative model). For multiplicative models, user-specified trading-day adjustment weights can also be specified. These weights will be used to adjust the monthly observations depending on the number of respective trading-days represented by the observation.
B. **Preliminary estimation of trading-day variation (monthly X-11) and weights.** Next, preliminary trading-day adjustment factors (monthly *X-11* only) and weights for reducing the effect of extreme observations are computed.
C. **Final estimation of trading-day variation and irregular weights (monthly *X-11*).** The adjustments and weights computed in *B* above are then used to derive improved trend-cycle and seasonal estimates. These improved estimates are used to compute the final trading-day factors (monthly *X-11* only) and weights.
D. **Final estimation of seasonal factors, trend-cycle, irregular, and seasonally adjusted series.** The final trading-day factors and weights computed in *C* above are used to compute the final estimates of the components.
E. **Modified original, seasonally adjusted, and irregular series.** The original and final seasonally adjusted series, and the irregular component are modified for extremes. The resulting modified series allow the user to examine the stability of the seasonal adjustment.
F. **Month (quarter) for cyclical dominance (MCD, QCD), moving average, and summary measures.** In this part of the computations, various summary measures (see below) are computed to allow the user to examine the relative importance of the different components, the average fluctuation from month-to-month (quarter-

to-quarter), the average number of consecutive changes in the same direction (average number of runs), etc.

G. **Charts.** Finally, you will compute various charts (graphs) to summarize the results. For example, the final seasonally adjusted series will be plotted, in chronological order, or by month (see below).

Specific Description of all Result Tables Computed by the _X-11_ Method

In each part _A_ through _G_ of the analysis (see Results Tables Computed by the _X-11_ Method), different result tables are computed. Customarily, these tables are numbered, and also identified by a letter to indicate the respective part of the analysis. For example, table _B 11_ shows the initial seasonally adjusted series; _C 11_ is the refined seasonally adjusted series, and _D 11_ is the final seasonally adjusted series. Shown below is a list of all available tables. Those tables identified by an asterisk (*) are not available (applicable) when analyzing quarterly series. (Also, for quarterly adjustment, some of the computations outlined below are slightly different; for example instead of a 12-term [monthly] moving average, a 4-term [quarterly] moving average is applied to compute the seasonal factors; the initial trend-cycle estimate is computed via a centered 4-term moving average, the final trend-cycle estimate in each part is computed by a 5-term Henderson average.)

Following the convention of the Bureau of the Census version of the _X-11_ method, three levels of printout detail are offered: _Standard_ (17 to 27 tables), _Long_ (27 to 39 tables), and _Full_ (44 to 59 tables). In the description of each table below, the letters _S_, _L_, and _F_ are used next to each title to indicate, which tables will be displayed and/or printed at the respective setting of the output option. (For the charts, two levels of detail are available: _Standard_ and _All_.)

See the table name below, to obtain more information about that table.

*A 1. Original Series_(S)_

* A 2. Prior Monthly Adjustment _(S)_Factors

* A 3. Original Series Adjusted by Prior Monthly Adjustment Factors_(S)_

* A 4. Prior Trading Day Adjustment Factors_(S)_

B 1. Prior Adjusted Series or Original Series_(S)_

B 2. Trend-cycle _(L)_

B 3. Unmodified S-I Differences or Ratios_(F)_

B 4. Replacement Values for Extreme S-I Differences (Ratios)_(F)_

B 5. Seasonal Factors_(F)_

B 6. Seasonally Adjusted Series_(F)_

B 7. Trend-cycle_(L)_

B 8. Unmodified S-I Differences (Ratios)_(F)_

B 9. Replacement Values for Extreme S-I Differences (Ratios)_(F)_

B 10. Seasonal Factors*(L)*

B 11. Seasonally Adjusted Series*(F)*

B 12. (not used)

B 13. Irregular Series *(L)*

Tables B 14 through B 16, B18, and B19: Adjustment for trading-day variation. These tables are only available when analyzing monthly series. Different months contain different numbers of days of the week (i.e., Mondays, Tuesdays, etc.). In some series, the variation in the different numbers of trading-days may contribute significantly to monthly fluctuations (e.g., the monthly revenues of an amusement park will be greatly influenced by the number of Saturdays/Sundays in each month). The user can specify initial weights for each trading-day (see *A 4*), and/or these weights can be estimated from the data (the user can also choose to apply those weights conditionally, i.e., only if they explain a significant proportion of variance).

* B 14. Extreme Irregular Values Excluded from Trading-day Regression *(L)*

* B 15. Preliminary Trading-day Regression *(L)*

* B 16. Trading-day Adjustment Factors Derived from Regression Coefficients *(F)*

B 17. Preliminary Weights for Irregular Component*(L)*

* B 18. Trading-day Factors Derived from Combined Daily Weights *(F)*

* B 19. Original Series Adjusted for Trading-day and Prior Variation*(F)*

C 1. Original Series Modified by Preliminary Weights and Adjusted for Trading-day and Prior Variation *(L)*

C 2. Trend-cycle *(F)*

C 3. (not used)

C 4. Modified S-I Differences (Ratios) *(F)*

C 5. Seasonal Factors*(F)*

C 6. Seasonally Adjusted Series*(F)*

C 7. Trend-cycle*(L)*

C 8. (not used)

C 9. Modified S-I Differences (Ratios)*(F*

C 10. Seasonal Factors *(L)*

C 11. Seasonally Adjusted Series *(F>*

C 12. (not used)

C 13. Irregular Series *(S)*

Tables C 14 through C 16, C 18, and C 19: Adjustment for trading-day variation. These tables are only available when analyzing monthly series, and when adjustment for trading-day variation is requested. In that case, the trading-day adjustment factors are computed from the refined adjusted series, analogous to the adjustment performed in part *B* (*B 14* through *B 16*, *B 18* and *B 19*).

* C 14. Extreme Irregular Values Excluded from Trading-day Regression *(S)*

* C 15. Final Trading-day Regression *(S)*

* C 16. Final Trading-day Adjustment Factors Derived from Regression X11 output: Coefficients *(S)*

C 17. Final Weights for Irregular Component *(S)*

* C 18. Final Trading-day Factors Derived From Combined Daily Weights *(S)*

* C 19. Original Series Adjusted for Trading-day and Prior Variation *(S)*

D 1. Original Series Modified by Final Weights and Adjusted for Trading-day and Prior Variation *(L)*

D 2. Trend-cycle

D 3. (not used)

D 4. Modified S-I Differences (Ratios) *(F)*

D 5. Seasonal Factors *(F)*

D 6. Seasonally Adjusted Series *(F)*

D 7. Trend-cycle *(L)*

D 8. Final Unmodified S-I Differences (Ratios) *(S)*

D 9. Final Replacement Values for Extreme S-I Differences (Ratios) *(S)*

D 10. Final Seasonal Factors *(S)*

D 11. Final Seasonally Adjusted Series *(S)*

D 12. Final Trend-cycle *(S)*

D 13. Final Irregular *(S)*

E 1. Modified Original Series *(S)*

E 2. Modified Seasonally Adjusted Series *(S)*

E 3. Modified Irregular Series *(S)*

E 4. Differences (Ratios) of Annual Totals *(S)*

E 5. Differences (Percent Changes) in Original Series *(S)*

E 6. Differences (Percent Changes) in Final Seasonally Adjusted Series *(S)*

F 1. MCD (QCD) Moving Average *(S)*

F 2. Summary Measures *(S)*

G 1. Chart *(S)*

G 2. Chart *(S)*

G 3. Chart *(A)*

G 4. Chart *(A)*

Distributed Lags Analysis Introductory Overview

- General Purpose
- General Model

- Almon Distributed Lag

For more information on other Time Series methods, see Time Series Analysis - Index and the following topics:

- Identifying Patterns in Time Series Data
- ARIMA (Box & Jenkins) and Autocorrelations ARIMA Introductory Overview
- Interrupted Time Series
- Exponential Smoothing
- Seasonal Decomposition (Census I)
- X-11 Census method II seasonal adjustment
- X-11 Census method II result tables
- Single Spectrum (Fourier) Analysis
- Cross-spectrum Analysis
- Basic Notations and Principles
- Fast Fourier Transformations

General Purpose

Distributed lags analysis is a specialized technique for examining the relationships between variables that involve some delay. For example, suppose that you are a manufacturer of computer software, and you want to determine the relationship between the number of inquiries that are received, and the number of orders that are placed by your customers. You could record those numbers monthly for a one year period, and then correlate the two variables. However, obviously inquiries will precede actual orders, and one can expect that the number of orders will follow the number of inquiries with some delay. Put another way, there will be a (time) *lagged* correlation between the number of inquiries and the number of orders that are received.

Time-lagged correlations are particularly common in econometrics. For example, the benefits of investments in new machinery usually only become evident after some time. Higher income will change people's choice of rental apartments, however, this relationship will be lagged because it will take some time for people to terminate their current leases, find new apartments, and move. In general, the relationship between capital appropriations and capital expenditures will be lagged, because it will require some time before investment decisions are actually acted upon.

In all of these cases, we have an independent or *explanatory* variable that affects the *dependent* variables with some lag. The distributed lags method allows you to investigate those lags.

Detailed discussions of distributed lags correlation can be found in most econometrics textbooks, for example, in Judge, Griffith, Hill, Luetkepohl, and Lee (1985), Maddala (1977), and Fomby, Hill, and Johnson (1984). In the following paragraphs we will present a brief description of these methods. We will assume that you are familiar with the concept of correlation (see *Basic Statistics*), and the basic ideas of multiple regression (see *Multiple Regression*).

General Model

Suppose we have a dependent variable *y* and an independent or explanatory variable *x* which are both measured repeatedly over time. In some textbooks, the dependent variable is also referred to as the *endogenous* variable, and the independent or explanatory variable the *exogenous* variable. The simplest way to describe the relationship between the two would be in a simple linear relationship:

$$Y_t = \sum \beta_i * x_{t-i}$$

In this equation, the value of the dependent variable at time t is expressed as a linear function of x measured at times t, $t-1$, $t-2$, etc. Thus, the dependent variable is a linear function of x, and x is *lagged* by *1*, *2*, etc. time periods. The beta weights (β_i) can be considered slope parameters in this equation. You may recognize this equation as a special case of the general linear regression equation (see the *Multiple Regression*overview). If the weights for the lagged time periods are statistically significant, we can conclude that the y variable is predicted (or explained) with the respective lag.

Almon Distributed Lag

A common problem that often arises when computing the weights for the multiple linear regression model shown above is that the values of adjacent (in time) values in the *x* variable are highly correlated. In extreme cases, their independent contributions to the prediction of y may become so redundant that the correlation matrix of measures can no longer be inverted, and thus, the *beta* weights cannot be computed. In less extreme cases, the computation of the *beta* weights and their standard errors can become very imprecise, due to round-off error. In the context of *Multiple Regression* this general computational problem is discussed as the *multicollinearity* or *matrix ill-conditioning* issue.

Almon (1965) proposed a procedure that will reduce the multicollinearity in this case. Specifically, suppose we express each weight in the linear regression equation in the following manner:

$$\beta_i = \alpha_0 + \alpha_1 * i + ... + \alpha_q * i^q$$

Almon could show that in many cases it is easier (i.e., it avoids the multicollinearity problem) to estimate the *alpha* values than the *beta* weights directly. Note that with this method, the precision of the beta weight estimates is dependent on the degree or order of the *polynomial approximation*.

Misspecifications. A general problem with this technique is that, of course, the lag length and correct polynomial degree are not known *a priori*. The effects of misspecifications of these parameters are potentially serious (in terms of biased estimation). This issue is discussed in greater detail in Frost (1975), Schmidt and Waud (1973), Schmidt and Sickles (1975), and Trivedi and Pagan (1979).

Single Spectrum (Fourier) Analysis

Spectrum analysis is concerned with the exploration of cyclical patterns of data. The purpose of the analysis is to decompose a complex time series with cyclical components into a few underlying sinusoidal (sine and cosine) functions of particular wavelengths. The term "spectrum" provides an appropriate metaphor for the nature of this analysis: Suppose you study a beam of white sun light, which at first looks like a random (white noise) accumulation of light of different wavelengths. However, when put through a prism, we can separate the different wave lengths or cyclical components that make up white sun light. In fact, via this technique we can now identify and distinguish between different sources of light. Thus, by identifying the important underlying cyclical components, we have learned something about the phenomenon of interest. In essence, performing spectrum analysis on a time series is like putting the series through a prism in order to identify the wave lengths and importance of underlying cyclical components. As a result of a successful analysis one might uncover just a few recurring cycles of different lengths in the time series of interest, which at first looked more or less like random noise.

A much cited example for spectrum analysis is the cyclical nature of sun spot activity (e.g., see Bloomfield, 1976, or Shumway, 1988). It turns out that sun spot activity varies over 11 year cycles. Other examples of celestial phenomena, weather patterns, fluctuations in commodity prices, economic activity, etc. are also often used in the literature to demonstrate this technique. To contrast this technique with ARIMA or Exponential Smoothing, the purpose of spectrum analysis is to identify the seasonal fluctuations of different lengths, while in the former types of analysis, the length of the seasonal component is usually known (or guessed) *a priori* and then included in some theoretical model of moving averages or autocorrelations.

The classic text on spectrum analysis is Bloomfield (1976); however, other detailed discussions can be found in Jenkins and Watts (1968), Brillinger (1975), Brigham (1974), Elliott and Rao (1982), Priestley (1981), Shumway (1988), or Wei (1989).

For more information, see Time Series Analysis - Index and the following topics:

- Basic Notations and Principles
- Fast Fourier Transformations
- Identifying Patterns in Time Series Data
- ARIMA (Box & Jenkins) and Autocorrelations ARIMA Introductory Overview
- Interrupted Time Series
- Distributed Lags Analysis
- Seasonal Decomposition (Census I)
- Exponential Smoothing

- Cross-spectrum Analysis

Cross-spectrum Analysis

- General Introduction
- Basic Notation and Principles
- Results for Each Variable
- The Cross-periodogram, Cross-density, Quadrature-density, and Cross-amplitude
- Squared Coherency, Gain, and Phase Shift
- How the Example Data were Created

For more information, see Time Series Analysis - Index and the following topics:

- Identifying Patterns in Time Series Data
- ARIMA (Box & Jenkins) and Autocorrelations ARIMA Introductory Overview
- Interrupted Time Series
- Exponential Smoothing Seasonal Decomposition (Census I)
- X-11 Census method II seasonal adjustment
- X-11 Census method II result tables
- Distributed Lags analysis
- Single Spectrum (Fourier) Analysis
- Basic Notations and Principles
- Fast Fourier Transformations

General Introduction

Cross-spectrum analysis is an extension of *Single Spectrum (Fourier) Analysis* to the simultaneous analysis of two series. In the following paragraphs, we will assume that you have already read the introduction to single spectrum analysis. Detailed discussions of this technique can be found in Bloomfield (1976), Jenkins and Watts (1968), Brillinger (1975), Brigham (1974), Elliott and Rao (1982), Priestley (1981), Shumway (1988), or Wei (1989).

Strong periodicity in the series at the respective frequency. A much cited example for spectrum analysis is the cyclical nature of sun spot activity (e.g., see Bloomfield, 1976, or Shumway, 1988). It turns out that sun spot activity varies over 11 year cycles. Other examples of celestial phenomena, weather patterns, fluctuations in commodity prices, economic activity, etc. are also often used in the literature to demonstrate this technique.

The purpose of cross-spectrum analysis is to uncover the correlations between two series at different frequencies. For example, sun spot activity may be related to weather phenomena here on earth. If so, then if we were to record those phenomena (e.g., yearly average temperature) and submit the resulting series to a cross-spectrum analysis together with the sun spot data, we may find that the weather indeed correlates with the sunspot activity at the 11 year cycle. That is, we may find a periodicity in the weather data that is "in-sync" with the sun spot cycles. One can easily think of other areas of research where such knowledge could be very useful; for example, various economic indicators may show similar (correlated) cyclical behavior; various physiological measures likely will also display "coordinated" (i.e., correlated) cyclical behavior, and so on.

Basic Notation and Principles

A simple example
Consider the following two series with 16 cases:

	VAR1	VAR2
1	1.000	-.058
2	1.637	-.713
3	1.148	-.383
4	-.058	.006
5	-.713	-.483
6	-.383	-1.441
7	.006	-1.637
8	-.483	-.707
9	-1.441	.331
10	-1.637	.441
11	-.707	-.058
12	.331	-.006
13	.441	.924
14	-.058	1.713
15	-.006	1.365
16	.924	.266

At first sight it is not easy to see the relationship between the two series. However, as shown below the series were created so that they would contain two strong correlated periodicities. Shown below are parts of the summary from the cross-spectrum analysis (the spectral estimates were smoothed with a Parzen window of width 3).

Indep.(X): VAR1 **Dep.(Y): VAR2**						
Frequncy	**Period**	**X Density**	**Y Density**	**Cross Density**	**Cross Quad**	**Cross Amplit.**
0.000000		.000000	.024292	-.00000	0.00000	.000000
.062500	16.00000	8.094709	7.798284	2.35583	-7.58781	7.945114
.125000	8.00000	.058771	.100936	-.04755	.06059	.077020
.187500	5.33333	3.617294	3.845154	-2.92645	2.31191	3.729484
.250000	4.00000	.333005	.278685	-.26941	.14221	.304637
.312500	3.20000	.091897	.067630	-.07435	.02622	.078835
.375000	2.66667	.052575	.036056	-.04253	.00930	.043539
.437500	2.28571	.040248	.026633	-.03256	.00342	.032740
.500000	2.00000	.037115	0.000000	0.00000	0.00000	0.000000

Results for Each Variable

The complete summary contains all spectrum statistics computed for each variable, as described in the *Single Spectrum (Fourier) Analysis* overview section. Looking at the results shown above, it is clear that both variables show strong periodicities at the frequencies .0625 and .1875.

Cross-periodogram, Cross-Density, Quadrature-density, Cross-amplitude

Analogous to the results for the single variables, the complete summary will also display periodogram values for the cross periodogram. However, the cross-spectrum consists of complex numbers that can be divided into a real and an imaginary part. These can be smoothed to obtain the cross-density and quadrature density (quad density for short) estimates, respectively. (The reasons for smoothing, and the different common weight functions for smoothing are discussed in the *Single Spectrum (Fourier) Analysis*.) The square root of the sum of the squared cross-density and quad-density values is called the *cross- amplitude*. The cross-amplitude can be interpreted as a measure of covariance between the respective frequency components in the two series. Thus we can conclude from the results shown in the table above that the .0625 and .1875 frequency components in the two series covary.

Squared Coherency, Gain, and Phase Shift

There are additional statistics that can be displayed in the complete summary.

Squared coherency. One can standardize the cross-amplitude values by squaring them and dividing by the product of the spectrum density estimates for each series. The result is called the *squared coherency*, which can be interpreted similar to the squared correlation coefficient (see *Correlations - Overview*), that is, the coherency value is the squared correlation between the cyclical components in the two series at the respective frequency. However, the coherency values should not be interpreted by themselves; for example, when the spectral density estimates in both series are very small, large coherency values may result (the divisor in the computation of the coherency values will be very small), even though there are no strong cyclical components in either series at the respective frequencies.

Gain. The gain value is computed by dividing the cross-amplitude value by the spectrum density estimates for one of the two series in the analysis. Consequently, two gain values are computed, which can be interpreted as the standard least squares regression coefficients for the respective frequencies.

Phase shift. Finally, the phase shift estimates are computed as tan**-1 of the ratio of the quad density estimates over the cross-density estimate. The phase shift estimates (usually denoted by the Greek letter) are measures of the extent to which each frequency component of one series leads the other.

How the Example Data were Created

Now, let us return to the example data set presented above. The large spectral density estimates for both series, and the cross-amplitude values at frequencies ν = 0.0625 and ν = .1875 suggest two strong synchronized periodicities in both series at those frequencies. In fact, the two series were created as:

v1 = cos(2*π*.0625*(v0-1)) + .75*sin(2*π*.2*(v0-1))

v2 = cos(2*π*.0625*(v0+2)) + .75*sin(2*π*.2*(v0+2))

(where *v0* is the case number). Indeed, the analysis presented in this overview reproduced the periodicity "inserted" into the data very well.

Spectrum Analysis - Basic Notation and Principles

- Frequency and Period
- The General Structural Model
- A Simple Example
- Periodogram
- The Problem of Leakage
- Padding the Time Series
- Tapering
- Data Windows and Spectral Density Estimates
- Preparing the Data for Analysis
- Results when no Periodicity in the Series Exists

For more information, see Time Series Analysis - Index and the following topics:

- Identifying Patterns in Time Series Data
- ARIMA (Box & Jenkins) and Autocorrelations ARIMA Introductory Overview
- Interrupted Time Series
- Exponential Smoothing
- Seasonal Decomposition (Census I)
- X-11 Census method II seasonal adjustment
- X-11 Census method II result tables
- Distributed Lags Analysis
- Single Spectrum (Fourier) Analysis
- Cross-spectrum Analysis
- Fast Fourier Transformations

Frequency and Period

The "wave length" of a sine or cosine function is typically expressed in terms of the number of cycles per unit time (*Frequency*), often denoted by the Greek letter *nu* (ν; some text books also use *f*). For example, the number of letters handled in a post office may show 12 cycles per year: On the first of every month a large amount of mail is sent (many bills come due on the first of the month), then the amount of mail decreases in the middle of the month, then it increases again towards the end of the month. Therefore, every month the fluctuation in the amount of mail handled by the post office will go through a full cycle. Thus, if the unit of analysis is one year, then n would be equal to 12, as there would be 12 cycles per year. Of course, there will likely be other cycles with different frequencies. For example, there might be annual cycles (ν=1), and perhaps weekly cycles <(ν=52 weeks per year).

The *period T* of a sine or cosine function is defined as the length of time required for one full cycle. Thus, it is the reciprocal of the frequency, or: $T = 1/\nu$. To return to the mail example in the previous paragraph, the monthly cycle, expressed in yearly terms, would be equal to 1/12 = 0.0833. Put into words, there is a period in the series of length 0.0833 years.

The General Structural Model

As mentioned before, the purpose of spectrum analysis is to decompose the original series into underlying sine and cosine functions of different frequencies, in order to determine those that appear particularly strong or important. One way to do so would be to cast the issue as a linear *Multiple Regression* problem, where the dependent variable is the observed time series, and the independent variables are the sine functions of all possible (discrete) frequencies. Such a linear multiple regression model may be written as:

$$x_t = a_0 + \Sigma[a_k*\cos(\lambda_k*t) + b_k*\sin(\lambda_k*t)] \quad \text{(for k = 1 to q)}$$

Following the common notation from classical harmonic analysis, in this equation λ (lambda) is the frequency expressed in terms of radians per unit time, that is: $\lambda = 2*\pi*\nu_k$, where π is the constant *pi*=3.14... and ν_k = k/q. What is important here is to recognize that the computational problem of fitting sine and cosine functions of different lengths to the data can be considered in terms of multiple linear regression. Note that the cosine parameters a_k and sine parameters b_k are regression coefficients that tell us the degree to which the respective functions are correlated with the data. Overall there are q different sine and cosine functions; intuitively (as also discussed in *Multiple Regression*), it should be clear that we cannot have more sine and cosine functions than there are data points in the series. Without going into detail, if there are *N* data points in the series, then there will be *N/2+1* cosine functions and *N/2-1* sine functions. In other words, there will be as many different sinusoidal waves as there are data points, and we will be able to completely reproduce the series from the underlying functions. (Note that if the number of cases in the series is odd, then the last data point will usually be ignored; in order for a sinusoidal function to be identified, you need at least two points: the high peak and the low peak.)

To summarize, spectrum analysis will identify the correlation of sine and cosine functions of different frequency with the observed data. If a large correlation (sine or cosine coefficient) is identified, one can conclude that there is a strong periodicity of the respective frequency (or period) in the data.

Complex numbers (real and imaginary numbers). In many text books on spectrum analysis, the structural model shown above is presented in terms of complex numbers, that is, the parameter estimation process is described in terms of the Fourier transform of a series into real and imaginary parts. Complex numbers are the superset that includes all real and imaginary numbers. Imaginary numbers, by definition, are numbers that are multiplied by the constant *i*, where *i* is defined as the square root of -1. Obviously, the square root of -1 does not exist, hence the term *imaginary* number; however, meaningful arithmetic operations on imaginary numbers can still be performed (e.g., [i*2]**2= -4). It is useful to think of real and imaginary numbers as forming a two dimensional plane, where the horizontal or *X*-axis represents all real numbers, and the vertical or *Y*-axis represents all imaginary numbers. Complex numbers can then be represented as points in the two- dimensional plane. For example, the complex number 3+i*2 can be represented by a point with coordinates {3,2} in this plane. One can also think of complex numbers as angles, for example, one can connect the point representing a complex number in the plane with the origin (complex number 0+i*0), and measure the angle of that vector to the horizontal line. Thus, intuitively one can see how the spectrum decomposition formula shown above, consisting of sine and cosine functions, can be rewritten in terms of operations on complex numbers. In fact, in this manner the mathematical discussion and required computations are often more elegant and easier to perform; which is why many text books prefer the presentation of spectrum analysis in terms of complex numbers.

A Simple Example

Shumway (1988) presents a simple example to clarify the underlying "mechanics" of spectrum analysis. Let us create a series with 16 cases following the equation shown above, and then see how we may "extract" the information that was put in it. First, create a variable and define it as:

x = 1*cos(2*π*.0625*(v0-1)) + .75*sin(2*π*.2*(v0-1))

This variable is made up of two underlying periodicities: The first at the frequency of ν =.0625 (or period 1/ν=16; one observation completes 1/16'th of a full cycle, and a full cycle is completed every 16 observations) and the second at the frequency of ν=.2 (or period of 5). The cosine coefficient (1.0) is larger than the sine coefficient (.75). The spectrum analysis summary is shown below.

	Spectral analysis:VAR1 (shumex.sta) No. of cases: 16				
t	Freq-uency	Period	Cosine Coeffs	Sine Coeffs	Period-ogram
0	.0000		.000	0.000	.000
1	.0625	16.00	1.006	.028	8.095
2	.1250	8.00	.033	.079	.059
3	.1875	5.33	.374	.559	3.617
4	.2500	4.00	-.144	-.144	.333
5	.3125	3.20	-.089	-.060	.092
6	.3750	2.67	-.075	-.031	.053
7	.4375	2.29	-.070	-.014	.040
8	.5000	2.00	-.068	0.000	.037

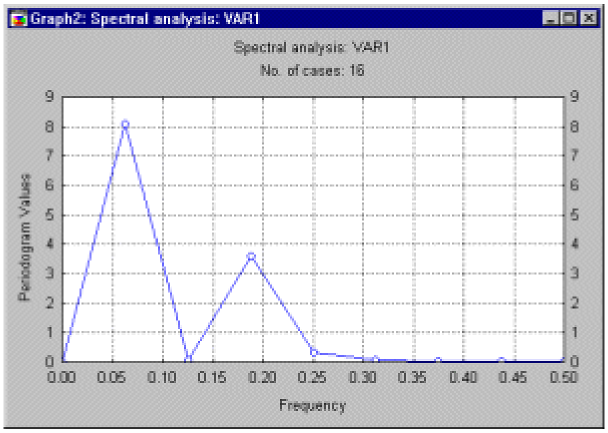
Let us now review the columns. Clearly, the largest cosine coefficient can be found for the .0625 frequency. A smaller sine coefficient can be found at frequency = .1875. Thus, clearly the two sine/cosine frequencies which were "inserted" into the example data file are reflected in the above table.

Periodogram

The sine and cosine functions are mutually independent (or orthogonal); thus we may sum the squared coefficients for each frequency to obtain the *periodogram*. Specifically, the periodogram values above are computed as:

P_k = sine coefficient$_k^2$ + cosine coefficient$_k^2$ * N/2

where P_k is the periodogram value at frequency ν_k and N is the overall length of the series. The periodogram values can be interpreted in terms of variance (sums of squares) of the data at the respective frequency or period. Customarily, the periodogram values are plotted against the frequencies or periods.



The Problem of Leakage

In the example above, a sine function with a frequency of 0.2 was "inserted" into the series. However, because of the length of the series (16), none of the frequencies reported exactly "hits" on that frequency. In practice, what often happens in those cases is that the respective frequency will "leak" into adjacent frequencies. For example, one may find large periodogram values for two adjacent frequencies, when, in fact, there is only one strong underlying sine or cosine function at a frequency that falls in-between those implied by the length of the series. There are three ways in which one can approach the problem of leakage:

- By padding the series one may apply a finer frequency "roster" to the data,
- By *tapering* the series prior to the analysis one may reduce leakage, or
- By smoothing the periodogram one may identify the general frequency "regions" or (*spectral densities*) that significantly contribute to the cyclical behavior of the series.

See below for descriptions of each of these approaches.

Padding the Time Series

Because the frequency values are computed as *N/t* (the number of units of times) one may simply *pad* the series with a constant (e.g., zeros) and thereby introduce smaller increments in the frequency values. In a sense, padding allows one to apply a finer roster to the data. In fact, if we padded the example data file described in the example above with ten zeros, the results would not change, that is, the largest periodogram peaks would still occur at the frequency values closest to .0625 and .2. (Padding is also often desirable for computational efficiency reasons; see below.)

Tapering

The so-called process of *split-cosine-bell tapering* is a recommended transformation of the series prior to the spectrum analysis. It usually leads to a reduction of leakage in the periodogram. The rationale for this transformation is explained in detail in Bloomfield (1976, p. 80-94). In essence, a proportion (*p*) of the data at the beginning and at the end of the series is transformed via multiplication by the weights:

$w_t = 0.5*\{1-\cos[\pi*(t - 0.5)/m]\}$ (for t=0 to m-1)
$w_t = 0.5*\{1-\cos[\pi*(N - t + 0.5)/m]\}$ (for t=N-m to N-1)

where m is chosen so that $2*m/N$ is equal to the proportion of data to be tapered (*p*).

Data Windows and Spectral Density Estimates

In practice, when analyzing actual data, it is usually not of crucial importance to identify exactly the frequencies for particular underlying sine or cosine functions. Rather, because the periodogram values are subject to substantial random fluctuation, one is faced with the problem of very many "chaotic" periodogram spikes. In that case, one would like to find the frequencies with the greatest *spectral densities*, that is, the frequency regions, consisting of many adjacent frequencies, that contribute most to the overall periodic behavior of the series. This can be accomplished by smoothing the periodogram values via a weighted moving average transformation. Suppose the moving average window is of width *m* (which must be an odd number); the following are the most commonly used smoothers (note: *p* = *(m-1)/2*).

Daniell (or equal weight) window. The Daniell window (Daniell 1946) amounts to a simple (equal weight) moving average transformation of the periodogram values, that is, each spectral density estimate is computed as the mean of the *m/2* preceding and subsequent periodogram values.

Tukey window. In the Tukey (Blackman and Tukey, 1958) or Tukey-Hanning window (named after Julius Von Hann), for each frequency, the weights for the weighted moving average of the periodogram values are computed as:

$w_j = 0.5 + 0.5*\cos(\pi *j/p)$ (for j=0 to p)
$w_{-j} = w_j$ (for $j \neq 0$)

Hamming window. In the Hamming (named after R. W. Hamming) window or Tukey-Hamming window (Blackman and Tukey, 1958), for each frequency, the weights for the weighted moving average of the periodogram values are computed as:

$w_j = 0.54 + 0.46*\cos(\pi *j/p)$ (for j=0 to p)
$w_{-j} = w_j$ (for $j \neq 0$)

Parzen window. In the Parzen window (Parzen, 1961), for each frequency, the weights for the weighted moving average of the periodogram values are computed as:

$w_j = 1-6*(j/p)^2 + 6*(j/p)^3$ (for j = 0 to p/2)
$w_j = 2*(1-j/p)^3$ (for j = p/2 + 1 to p)
$w_{-j} = w_j$ (for $j \neq 0$)

Bartlett window. In the Bartlett window (Bartlett, 1950) the weights are computed as:

$w_j = 1-(j/p)$ (for j = 0 to p)
$w_{-j} = w_j$ (for $j \neq 0$)

With the exception of the Daniell window, all weight functions will assign the greatest weight to the observation being smoothed in the center of the window, and increasingly smaller weights to values that are further away from the center. In many cases, all of these data windows will produce very similar results

Preparing the Data for Analysis

Let us now consider a few other practical points in spectrum analysis. Usually, one wants to subtract the mean from the series, and detrend the series (so that it is stationary) prior to the analysis. Otherwise the periodogram and density spectrum will mostly be "overwhelmed" by a very large value for the first cosine coefficient (for frequency 0.0). In a sense, the mean is a cycle of frequency 0 (zero) per unit time; that is, it is a constant. Similarly, a trend is also of little interest when one wants to uncover the periodicities in the series. In fact, both of those potentially strong effects may mask the more interesting periodicities in the data, and thus both the mean and the trend (linear) should be removed from the series prior to the analysis. Sometimes, it is also useful to smooth the data prior to the analysis, in order to "tame" the random noise that may obscure meaningful periodic cycles in the periodogram.

Results when no Periodicity in the Series Exists

Finally, what if there are no recurring cycles in the data, that is, if each observation is completely independent of all other observations? If the distribution of the observations follows the normal distribution, such a time series is also referred to as a *white noise* series (like the white noise one hears on the radio when tuned in-between stations). A white noise input series will result in periodogram values that follow an exponential distribution. Thus, by testing the distribution of periodogram values against the exponential distribution, one may test whether the input series is different from a white noise series. In addition, the you can also request to compute the Kolmogorov-Smirnov one-sample *d* statistic (see also *Nonparametrics and Distributions* for more details).

Testing for white noise in certain frequency bands. Note that you can also plot the periodogram values for a particular frequency range only. Again, if the input is a white noise series with respect to those frequencies (i.e., it there are no significant periodic cycles of those frequencies), then the distribution of the periodogram values should again follow an exponential distribution.

Fast Fourier Transforms (FFT)

- General Introduction
- Computation of FFT in Time Series

For more information, see Time Series Analysis - Index and the following topics:

- Identifying Patterns in Time Series Data
- ARIMA (Box & Jenkins) and Autocorrelations ARIMA Introductory Overview
- Interrupted Time Series
- Exponential Smoothing

- Seasonal Decomposition (Census I)
- X-11 Census method II seasonal adjustment
- X-11 Census method II result tables
- Distributed Lags Analysis
- Single Spectrum (Fourier) Analysis
- Cross-spectrum Analysis
- Basic Notations and Principles

General Introduction

The interpretation of the results of spectrum analysis is discussed in the *Basic Notation and Principles* topic, however, we have not described how it is done computationally. Up until the mid-1960s the standard way of performing the spectrum decomposition was to use explicit formulae to solve for the sine and cosine parameters. The computations involved required at least N**2 (complex) multiplications. Thus, even with today's high-speed computers , it would be very time consuming to analyze even small time series (e.g., 8,000 observations would result in at least 64 million multiplications).

The time requirements changed drastically with the development of the so-called *fast Fourier transform* algorithm, or *FFT* for short. In the mid-1960s, J.W. Cooley and J.W. Tukey (1965) popularized this algorithm which, in retrospect, had in fact been discovered independently by various individuals. Various refinements and improvements of this algorithm can be found in Monro (1975) and Monro and Branch (1976). Readers interested in the computational details of this algorithm may refer to any of the texts cited in the overview. Suffice it to say that via the FFT algorithm, the time to perform a spectral analysis is proportional to *N**log2(*N*) -- a huge improvement.

However, a draw-back of the standard FFT algorithm is that the number of cases in the series must be equal to a power of 2 (i.e., 16, 64, 128, 256, ...). Usually, this necessitated padding of the series, which, as described above, will in most cases not change the characteristic peaks of the periodogram or the spectral density estimates. In cases, however, where the time units are meaningful, such padding may make the interpretation of results more cumbersome.

Computation of FFT in Time Series

The implementation of the FFT algorithm allows you to take full advantage of the savings afforded by this algorithm. On most standard computers, series with over 100,000 cases can easily be analyzed. However, there are a few things to remember when analyzing series of that size.

As mentioned above, the standard (and most efficient) FFT algorithm requires that the length of the input series is equal to a power of 2. If this is not the case, additional computations have to be performed. It will use the simple explicit computational formulas as long as the input series is relatively small, and the number of computations can be performed in a relatively short amount of time. For long time series, in order to still utilize the FFT algorithm, an implementation of the general approach described by Monro and Branch (1976) is used. This method requires significantly more storage space, however, series of considerable length can still be analyzed very quickly, even if the number of observations is not equal to a power of 2.

For time series of lengths not equal to a power of 2, we would like to make the following recommendations: If the input series is small to moderately sized (e.g., only a few thousand cases), then do not worry. The analysis will typically only take a few seconds anyway. In order to analyze moderately large and large series (e.g., over 100,000 cases), pad the series to a power of 2 and then taper the series during the exploratory part of your data analysis.

Chapter 41
Variance Components and Mixed Model ANOVA/ANCOVA

- Basic Ideas
    - Properties of Random Effects
- Estimation of Variance Components (Technical Overview)
    - Estimating the Variation of Random Factors
    - Estimating Components of Variation
    - Testing the Significance of Variance Components
    - Estimating the Population Intraclass Correlation

The *Variance Components and Mixed Model ANOVA/ANCOVA* chapter describes a comprehensive set of techniques for analyzing research designs that include random effects; however, these techniques are also well suited for analyzing large main effect designs (e.g., designs with over 200 levels per factor), designs with many factors where the higher order interactions are not of interest, and analyses involving case weights.

There are several chapters in this textbook that will discuss Analysis of Variance for factorial or specialized designs. For a discussion of these chapters and the types of designs for which they are best suited refer to the section on Methods for Analysis of Variance. Note, however, that the General Linear Models chapter describes how to analyze designs with any number and type of between effects and compute ANOVA-based variance component estimates for any effect in a mixed-model analysis.

Basic Ideas

Experimentation is sometimes mistakenly thought to involve only the manipulation of levels of the independent variables and the observation of subsequent responses on the dependent variables. Independent variables whose levels are determined or set by the experimenter are said to have *fixed effects*. There is a second class of effects, however, which is often of great interest to the researcher, *Random effects* are classification effects where the levels of the effects are assumed to be randomly selected from an infinite population of possible levels. Many independent variables of research interest are not fully amenable to experimental manipulation, but nevertheless can be studied by considering them to have *random effects*. For example, the genetic makeup of individual members of a species cannot at present be (fully) experimentally manipulated, yet it is of great interest to the geneticist to assess the genetic contribution to individual variation on outcomes such as health, behavioral characteristics, and the like. As another example, a manufacturer might want to estimate the components of variation in the characteristics of a product for a random sample of machines operated by a random sample of operators. The statistical analysis of *random effects* is accomplished by using the *random effect* *model*, if all of the independent variables are assumed to have *random effects*, or by using the *mixed model*, if some of the independent variables are assumed to have *random effects* and other independent variables are assumed to have *fixed effects*.

Properties of random effects. To illustrate some of the properties of *random effects*, suppose you collected data on the amount of insect damage done to different varieties of wheat. It is impractical to study insect damage for every possible variety of wheat, so to conduct the experiment, you randomly select four varieties of wheat to study. Plant damage is rated for up to a maximum of four plots per variety. Ratings are on a 0 (no damage) to 10 (great damage) scale. The following data for this example are presented in Milliken and Johnson (1992, p. 237).

DATA: wheat.sta 3v		
VARIETY	**PLOT**	**DAMAGE**
A	1	3.90
A	2	4.05
A	3	4.25
B	4	3.60
B	5	4.20
B	6	4.05
B	7	3.85
C	8	4.15
C	9	4.60
C	10	4.15
C	11	4.40
D	12	3.35
D	13	3.80

To determine the components of variation in resistance to insect damage for *Variety* and *Plot*, an ANOVA can first be performed. Perhaps surprisingly, in the ANOVA, *Variety* can be treated as a fixed or as a random factor without influencing the results (provided that *Type I Sums of squares* are used and that *Variety* is always entered first in the model). The Spreadsheet below shows the ANOVA results of a mixed model analysis treating *Variety* as a *fixed effect* and ignoring *Plot*, i.e., treating the plot-to-plot variation as a measure of random error.

ANOVA Results: DAMAGE (wheat.sta)							
	Effect	**df**	**MS**	**df**	**MS**		
Effect	**(F/R)**	**Effect**	**Effect**	**Error**	**Error**	**F**	**p**
{1}VARIETY	Fixed	3	.270053	9	.056435	4.785196	.029275

Another way to perform the same mixed model analysis is to treat *Variety* as a *fixed effect* and *Plot* as a *random effect*. The Spreadsheet below shows the ANOVA results for this mixed model analysis.

ANOVA Results for Synthesized Errors: DAMAGE (wheat.sta)							
	df error computed using Satterthwaite method						
Effect	Effect (F/R)	df Effect	MS Effect	df Error	MS Error	F	p
{1}VARIETY	Fixed	3	.270053	9	.056435	4.785196	.029275
{2}PLOT	Random	9	.056435	-----	-----	-----	-----

The Spreadsheet below shows the ANOVA results for a *random effect* model treating *Plot* as a *random effect* nested within *Variety*, which is also treated as a *random effect*.

ANOVA Results for Synthesized Errors: DAMAGE (wheat.sta)							
	df error computed using Satterthwaite method						
Effect	Effect (F/R)	df Effect	MS Effect	df Error	MS Error	F	p
{1}VARIETY	Random	3	.270053	9	.056435	4.785196	.029275
{2}PLOT	Random	9	.056435	-----	-----	-----	-----

As can be seen, the tests of significance for the *Variety* effect are identical in all three analyses (and in fact, there are even more ways to produce the same result). When components of variance are estimated, however, the difference between the mixed model (treating *Variety* as fixed) and the random model (treating *Variety* as random) becomes apparent. The Spreadsheet below shows the *variance component* estimates for the mixed model treating *Variety* as a *fixed effect*.

Components of Variance (wheat.sta)	
	Mean Squares Type: 1
Source	DAMAGE
{2}PLOT	.056435
Error	0.000000

The Spreadsheet below shows the *variance component* estimates for the *random effects* model treating *Variety* and *Plot* as *random effects*.

Components of Variance (wheat.sta)	
	Mean Squares Type: 1
Source	DAMAGE
{1}VARIETY	.067186
{2}PLOT	.056435
Error	0.000000

As can be seen, the difference in the two sets of estimates is that a *variance component* is estimated for *Variety* only when it is considered to be a *random effect*. This reflects the basic distinction between *fixed* and *random effects*. The variation in the levels of random factors is assumed to be representative of the variation of the whole population of possible levels. Thus, variation in the levels of a random factor can be used to estimate the population variation. Even more importantly, covariation between the levels of a random factor and responses on a dependent variable can be used to estimate the population component of variance in the dependent variable attributable to the random factor. The variation in the levels of fixed factors is instead considered to be arbitrarily determined by the experimenter (i.e., the experimenter can make the levels of a fixed factor vary as little or as much as desired). Thus, the variation of a fixed factor cannot be used to estimate its population variance, nor can the population covariance with the dependent variable be meaningfully estimated. With this basic distinction between *fixed effects* and *random effects* in mind, we now can look more closely at the properties of *variance components*.

Estimation of Variance Components (Technical Overview)

The basic goal of *variance component* estimation is to estimate the population covariation between random factors and the dependent variable. Depending on the method used to estimate *variance components*, the population variances of the random factors can also be estimated, and significance tests can be performed to test whether the population covariation between the random factors and the dependent variable are nonzero.

Estimating the variation of random factors. The ANOVA method provides an integrative approach to estimating *variance components*, because ANOVA techniques can be used to estimate the variance of random factors, to estimate the components of variance in the dependent variable attributable to the random factors, and to test whether the *variance components* differ significantly from zero. The ANOVA method for estimating the variance of the random factors begins by constructing the *Sums of squares* *and cross products* (*SSCP*) matrix for the independent variables. The *sums of squares and cross products* for the *random effects* are then residualized on the *fixed effects*, leaving the *random effects* independent of the *fixed effects*, as required in the mixed model (see, for example, Searle, Casella, & McCulloch, 1992). The residualized *Sums of squares and cross products* for each random factor are then divided by their degrees of freedom to produce the coefficients in the *Expected mean squares* matrix. Nonzero off-diagonal coefficients for the *random effects* in this matrix indicate confounding, which must be taken into account when estimating the population variance for each factor. For the *wheat.sta* data, treating both *Variety* and *Plot* as *random effects*, the coefficients in the *Expected mean squares* matrix show that the two factors are at least somewhat confounded. The *Expected mean squares* Spreadsheet is shown below.

Expected Mean Squares (wheat.sta)				
	Mean Squares Type: 1			
Source	**Effect (F/R)**	**VARIETY**	**PLOT**	**Error**
{1}VARIETY	Random	3.179487	1.000000	1.000000
{2}PLOT	Random		1.000000	1.000000
Error				1.000000

The coefficients in the *Expected mean squares* matrix are used to estimate the population variation of the *random effects* by equating their variances to their expected mean squares. For example, the estimated population variance for *Variety* using *Type I Sums of squares* would be 3.179487 times the *Mean square* for *Variety* plus 1 times the *Mean square* for *Plot* plus 1 times the *Mean square* for *Error*.

The ANOVA method provides an integrative approach to estimating *variance components*, but it is not without problems (i.e., ANOVA estimates of *variance components* are generally *biased*, and can be *negative*, even though variances, by definition, must be either zero or positive). An alternative to ANOVA estimation is provided by *maximum likelihood* estimation. *Maximum likelihood* methods for estimating *variance components* are based on quadratic forms, and typically, but not always, require iteration to find a solution. Perhaps the simplest form of *maximum likelihood* estimation is *MIVQUE(0)* estimation. *MIVQUE(0)* produces Minimum Variance Quadratic Unbiased Estimators (i.e., *MIVQUE*). In *MIVQUE(0)* estimation, there is no weighting of the *random effects* (thus the 0 [zero] after *MIVQUE*), so an iterative solution for estimating *variance components* is not required. *MIVQUE(0)* estimation begins by constructing the *Quadratic sums of squares (SSQ)* matrix. The elements for the *random effects* in the *SSQ* matrix can most simply be described as the sums of squares of the sums of squares and cross products for each *random effect* in the model (after residualization on the *fixed effects*). The elements of this matrix provide coefficients, similar to the elements of the *Expected Mean Squares* matrix, which are used to estimate the covariances among the random factors and the dependent variable. The *SSQ* matrix for the *wheat.sta* data is shown below. Note that the nonzero off-diagonal element for *Variety* and *Plot* again shows that the two random factors are at least somewhat confounded.

MIVQUE(0) Variance Component Estimation (wheat.sta)				
	SSQ Matrix			
Source	VARIETY	PLOT	Error	DAMAGE
{1}VARIETY	31.90533	9.53846	9.53846	2.418964
{2}PLOT	9.53846	12.00000	12.00000	1.318077
Error	9.53846	12.00000	12.00000	1.318077

Restricted Maximum Likelihood (REML) and *Maximum Likelihood (ML) variance component* estimation methods are closely related to *MIVQUE(0)*. In fact, in the program, *REML* and *ML* use *MIVQUE(0)* estimates as start values for an iterative solution for the *variance components*, so the elements of the *SSQ* matrix serve as initial estimates of the covariances among the random factors and the dependent variable for both *REML* and *ML*.

Estimating components of variation. For ANOVA methods for estimating *variance components*, a solution is found for the system of equations relating the estimated population variances and covariances among the random factors to the estimated population covariances between the random factors and the dependent variable. The solution then defines the *variance components*. The Spreadsheet below shows the *Type I Sums of squares* estimates of the *variance components* for the *wheat.sta* data.

Components of Variance (wheat.sta)	
	Mean Squares Type: 1
Source	DAMAGE
{1}VARIETY	0.067186
{2}PLOT	0.056435
Error	0.000000

MIVQUE(0) variance components are estimated by inverting the partition of the *SSQ* matrix that does not include the dependent variable (or finding the generalized inverse, for singular matrices), and postmultiplying the inverse by the dependent variable column vector. This amounts to solving the system of equations that relates the dependent variable to the random independent variables, taking into account the covariation among the independent variables. The *MIVQUE(0)* estimates for the *wheat.sta* data are listed in the Spreadsheet shown below.

MIVQUE(0) Variance Component Estimation (wheat.sta)	
	Variance Components
Source	**DAMAGE**
{1}VARIETY	0.056376
{2}PLOT	0.065028
Error	0.000000

REML and *ML variance components* are estimated by iteratively optimizing the parameter estimates for the effects in the model. *REML* differs from *ML* in that the likelihood of the data is maximized only for the *random effects*, thus *REML* is a *restricted* solution. In both *REML*and*ML*estimation, an iterative solution is found for the weights for the *random effects* in the model that maximize the likelihood of the data. The program uses *MIVQUE(0)*) estimates as the start values for both *REML* and *ML* estimation, so the relation between these three techniques is close indeed. The statistical theory underlying *maximum likelihood variance component* estimation techniques is an advanced topic (Searle, Casella, & McCulloch, 1992, is recommended as an authoritative and comprehensive source). Implementation of *maximum likelihood* estimation algorithms, furthermore, is difficult (see, for example, Hemmerle & Hartley, 1973, and Jennrich & Sampson, 1976, for descriptions of these algorithms), and faulty implementation can lead to *variance component* estimates that lie outside the parameter space, converge prematurely to nonoptimal solutions, or give nonsensical results. Milliken and Johnson (1992) noted all of these problems with the commercial software packages they used to estimate *variance components*.

The basic idea behind both *REML* and *ML* estimation is to find the set of weights for the *random effects* in the model that minimize the negative of the natural logarithm times the likelihood of the data (the likelihood of the data can vary from zero to one, so minimizing the negative of the natural logarithm times the likelihood of the data amounts to maximizing the probability, or the likelihood, of the data). The logarithm of the *REML*likelihood and the *REML variance component* estimates for the *wheat.sta* data are listed in the last row of the *Iteration history* Spreadsheet shown below.

Iteration History (wheat.sta)			
	Variable: DAMAGE		
Iter.	**Log LL**	**Error**	**VARIETY**
1	-2.30618	.057430	.068746
2	-2.25253	.057795	.073744
3	-2.25130	.056977	.072244
4	-2.25088	.057005	.073138
5	-2.25081	.057006	.073160

6 **7**	-2.25081 -2.25081	.057003 .057003	.073155 .073155

The logarithm of the *ML*likelihood and the *ML* estimates for the *variance components* for the *wheat.sta* data are listed in the last row of the *Iteration history* Spreadsheet shown below.

Iteration History (wheat.sta)			
	Variable: DAMAGE		
Iter.	**Log LL**	**Error**	**VARIETY**
1 **2** **3** **4** **5** **6**	-2.53585 -2.48382 -2.48381 -2.48381 -2.48381 -2.48381	.057454 .057427 .057492 .057491 .057492 .057492	.048799 .048541 .048639 .048552 .048552 .048552

As can be seen, the estimates of the *variance components* for the different methods are quite similar. In general, components of variance using different estimation methods tend to agree fairly well (see, for example, Swallow & Monahan, 1984).

Testing the significance of variance components. When *maximum likelihood* estimation techniques are used, standard linear model significance testing techniques may not be applicable. ANOVA techniques such as decomposing sums of squares and testing the significance of effects by taking ratios of mean squares are appropriate for linear methods of estimation, but generally are not appropriate for quadratic methods of estimation. When ANOVA methods are used for estimation, standard significance testing techniques can be employed, with the exception that any confounding among *random effects* must be taken into account.

To test the significance of effects in mixed or random models, error terms must be constructed that contain all the same sources of random variation except for the variation of the respective effect of interest. This is done using Satterthwaite's method of denominator synthesis (Satterthwaite, 1946), which finds the linear combinations of sources of random variation that serve as appropriate error terms for testing the significance of the respective effect of interest. The Spreadsheet below shows the coefficients used to construct these linear combinations for testing the *Variety* and *Plot* effects.

Denominator Synthesis: Coefficients (MS Type: 1) (wheat.sta)				
	The synthesized MS Errors are linear combinations of the resp. MS effects			
Effect	(F/R)	VARIETY	PLOT	Error
{1}VARIETY	Random		1.000000	
{2}PLOT	Random			1.000000

The coefficients show that the *Mean square* for *Variety* should be tested against the *Mean square* for *Plot*, and that the *Mean square* for *Plot* should be tested against the *Mean square* for *Error*. Referring back to the *Expected mean squares* Spreadsheet, it is clear that the denominator synthesis has identified appropriate error terms for testing the *Variety* and *Plot* effects. Although this is a simple example, in more complex analyses with various degrees of confounding among the *random effects*, the denominator synthesis can identify appropriate error terms for testing the *random effects* that would not be readily apparent.

To perform the tests of significance of the *random effects*, ratios of appropriate *Mean squares* are formed to compute *F* statistics and *p* levels for each effect. Note that in complex analyses the degrees of freedom for *random effects* can be fractional rather than integer values, indicating that fractions of sources of variation were used in synthesizing appropriate error terms for testing the *random effects*. The Spreadsheet displaying the results of the ANOVA for the *Variety* and *Plot random effects* is shown below. Note that for this simple design the results are identical to the results presented earlier in the Spreadsheet for the ANOVA treating *Plot* as a *random effect* nested within *Variety*.

ANOVA Results for Synthesized Errors: DAMAGE (wheat.sta)							
	df error computed using Satterthwaite method						
Effect	Effect (F/R)	df Effect	MS Effect	df Error	MS Error	F	p
{1}VARIETY	Fixed	3	.270053	9	.056435	4.785196	.029275
{2}PLOT	Random	9	.056435	-----	-----	-----	-----

As shown in the Spreadsheet, the *Variety* effect is found to be significant at $p < .05$, but as would be expected, the *Plot* effect cannot be tested for significance because plots served as the basic unit of analysis. If data on samples of plants taken within plots were available, a test of the significance of the *Plot* effect could be constructed.

Appropriate tests of significance for *MIVQUE(0) variance component* estimates generally cannot be constructed, except in special cases (see Searle, Casella, &

McCulloch, 1992). Asymptotic (large sample) tests of significance of *REML* and *ML variance component* estimates, however, can be constructed for the parameter estimates from the *final iteration* of the solution. The Spreadsheet below shows the asymptotic (large sample) tests of significance for the *REML* estimates for the *wheat.sta* data.

Restricted Maximum Likelihood Estimates (wheat.sta)				
	Variable: DAMAGE -2*Log(Likelihood)=4.50162399			
Effect	Variance Comp.	Asympt. Std.Err.	Asympt. z	Asympt. p
{1}VARIETY	.073155	.078019	.937656	.348421
Error	.057003	.027132	2.100914	.035648

The Spreadsheet below shows the asymptotic (large sample) tests of significance for the *ML* estimates for the *wheat.sta* data.

Maximum Likelihood Estimates (wheat.sta)				
	Variable: DAMAGE -2*Log(Likelihood)=4.96761616			
Effect	Variance Comp.	Asympt. Std.Err.	Asympt. z	Asympt. p
{1}VARIETY	.048552	.050747	.956748	.338694
Error	.057492	.027598	2.083213	.037232

It should be emphasized that the asymptotic tests of significance for *REML* and *ML variance component* estimates are based on large sample sizes, which certainly is not the case for the *wheat.sta* data. For this data set, the tests of significance from both analyses agree in suggesting that the *Variety variance component* does not differ significantly from zero.

For basic information on ANOVA in linear models, see also *Elementary Concepts*.

Estimating the population intraclass correlation. Note that if the *variance component* estimates for the random effects in the model are divided by the sum of all components (including the error component), the resulting percentages are population intraclass correlation coefficients for the respective effects.

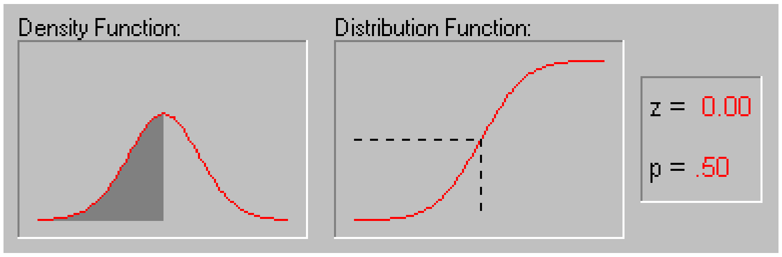
Chapter 42
Distribution Tables

Compared to probability calculators (e.g., the one included in *STATISTICA*), the traditional format of distribution tables such as those presented below, has the advantage of showing many values simultaneously and, thus, enables the user to examine and quickly explore ranges of probabilities.

- Z Table
- t Table
- Chi-Square Table
- F Tables for:
    - alpha=.10
    - alpha=.05
    - alpha=.025
    - alpha=.01

Note that all table values were calculated using the distribution facilities in *STATISTICA BASIC*, and they were verified against other published tables.

Standard Normal (Z) Table



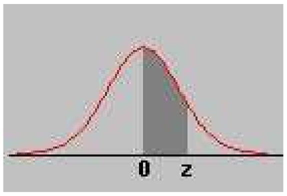
z = 0.00

p = .50

The Standard Normal distribution is used in various hypothesis tests including tests on single means, the difference between two means, and tests on proportions. The Standard Normal distribution has a mean of 0 and a standard deviation of 1. The animation above shows various (left) tail areas for this distribution. For more information on the Normal Distribution as it is used in statistical testing, see the chapter on Elementary Concepts. See also, the Normal Distribution.

As shown in the illustration below, the values inside the given table represent the areas under the standard normal curve for values between 0 and the relative z-score. For example, to determine the area under the curve between 0 and 2.36, look in the intersecting cell for the row labeled 2.30 and the column labeled 0.06. The area under the curve is .4909. To determine the area between 0 and a negative value, look in the intersecting cell of the row and column which sums to the absolute value of the number in question. For example, the area under the curve between -1.3 and 0 is equal to the area under the curve between 1.3 and 0, so look at the cell on the 1.3 row and the 0.00 column (the area is 0.4032).

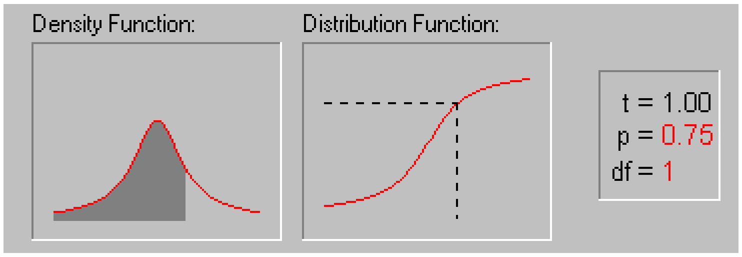
Area between 0 and z



	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.0000	0.0040	0.0080	0.0120	0.0160	0.0199	0.0239	0.0279	0.0319	0.0359
0.1	0.0398	0.0438	0.0478	0.0517	0.0557	0.0596	0.0636	0.0675	0.0714	0.0753
0.2	0.0793	0.0832	0.0871	0.0910	0.0948	0.0987	0.1026	0.1064	0.1103	0.1141
0.3	0.1179	0.1217	0.1255	0.1293	0.1331	0.1368	0.1406	0.1443	0.1480	0.1517
0.4	0.1554	0.1591	0.1628	0.1664	0.1700	0.1736	0.1772	0.1808	0.1844	0.1879
0.5	0.1915	0.1950	0.1985	0.2019	0.2054	0.2088	0.2123	0.2157	0.2190	0.2224
0.6	0.2257	0.2291	0.2324	0.2357	0.2389	0.2422	0.2454	0.2486	0.2517	0.2549
0.7	0.2580	0.2611	0.2642	0.2673	0.2704	0.2734	0.2764	0.2794	0.2823	0.2852
0.8	0.2881	0.2910	0.2939	0.2967	0.2995	0.3023	0.3051	0.3078	0.3106	0.3133
0.9	0.3159	0.3186	0.3212	0.3238	0.3264	0.3289	0.3315	0.3340	0.3365	0.3389
1.0	0.3413	0.3438	0.3461	0.3485	0.3508	0.3531	0.3554	0.3577	0.3599	0.3621
1.1	0.3643	0.3665	0.3686	0.3708	0.3729	0.3749	0.3770	0.3790	0.3810	0.3830
1.2	0.3849	0.3869	0.3888	0.3907	0.3925	0.3944	0.3962	0.3980	0.3997	0.4015
1.3	0.4032	0.4049	0.4066	0.4082	0.4099	0.4115	0.4131	0.4147	0.4162	0.4177
1.4	0.4192	0.4207	0.4222	0.4236	0.4251	0.4265	0.4279	0.4292	0.4306	0.4319
1.5	0.4332	0.4345	0.4357	0.4370	0.4382	0.4394	0.4406	0.4418	0.4429	0.4441
1.6	0.4452	0.4463	0.4474	0.4484	0.4495	0.4505	0.4515	0.4525	0.4535	0.4545
1.7	0.4554	0.4564	0.4573	0.4582	0.4591	0.4599	0.4608	0.4616	0.4625	0.4633
1.8	0.4641	0.4649	0.4656	0.4664	0.4671	0.4678	0.4686	0.4693	0.4699	0.4706
1.9	0.4713	0.4719	0.4726	0.4732	0.4738	0.4744	0.4750	0.4756	0.4761	0.4767
2.0	0.4772	0.4778	0.4783	0.4788	0.4793	0.4798	0.4803	0.4808	0.4812	0.4817
2.1	0.4821	0.4826	0.4830	0.4834	0.4838	0.4842	0.4846	0.4850	0.4854	0.4857
2.2	0.4861	0.4864	0.4868	0.4871	0.4875	0.4878	0.4881	0.4884	0.4887	0.4890
2.3	0.4893	0.4896	0.4898	0.4901	0.4904	0.4906	0.4909	0.4911	0.4913	0.4916
2.4	0.4918	0.4920	0.4922	0.4925	0.4927	0.4929	0.4931	0.4932	0.4934	0.4936
2.5	0.4938	0.4940	0.4941	0.4943	0.4945	0.4946	0.4948	0.4949	0.4951	0.4952

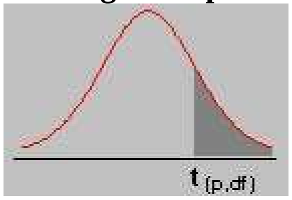
2.6	0.4953	0.4955	0.4956	0.4957	0.4959	0.4960	0.4961	0.4962	0.4963	0.4964
2.7	0.4965	0.4966	0.4967	0.4968	0.4969	0.4970	0.4971	0.4972	0.4973	0.4974
2.8	0.4974	0.4975	0.4976	0.4977	0.4977	0.4978	0.4979	0.4979	0.4980	0.4981
2.9	0.4981	0.4982	0.4982	0.4983	0.4984	0.4984	0.4985	0.4985	0.4986	0.4986
3.0	0.4987	0.4987	0.4987	0.4988	0.4988	0.4989	0.4989	0.4989	0.4990	0.4990

Student's t Table



The Shape of the Student's t distribution is determined by the degrees of freedom. As shown in the animation above, its shape changes as the degrees of freedom increases. For more information on how this distribution is used in hypothesis testing, see t-test for independent samples and t-test for dependent samples in the chapter on Basic Statistics and Tables. See also, Student's t Distribution. As indicated by the chart below, the areas given at the top of this table are the right tail areas for the t-value inside the table. To determine the 0.05 critical value from the t-distribution with 6 degrees of freedom, look in the 0.05 column at the 6 row: $t_{(.05,6)}$ = 1.943180.

t table with right tail probabilities

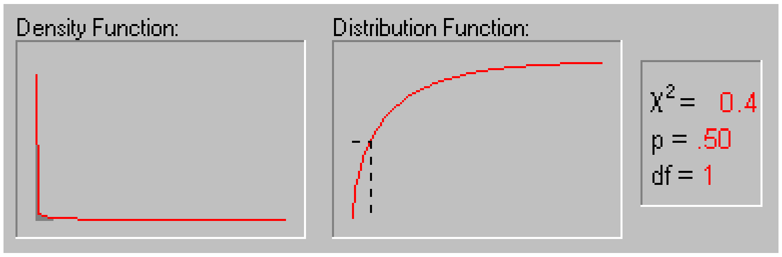


df\p	0.40	0.25	0.10	0.05	0.025	0.01	0.005	0.0005
1	0.324920	1.000000	3.077684	6.313752	12.70620	31.82052	63.65674	636.6192
2	0.288675	0.816497	1.885618	2.919986	4.30265	6.96456	9.92484	31.5991
3	0.276671	0.764892	1.637744	2.353363	3.18245	4.54070	5.84091	12.9240
4	0.270722	0.740697	1.533206	2.131847	2.77645	3.74695	4.60409	8.6103
5	0.267181	0.726687	1.475884	2.015048	2.57058	3.36493	4.03214	6.8688

6	0.264835	0.717558	1.439756	1.943180	2.44691	3.14267	3.70743	5.9588
7	0.263167	0.711142	1.414924	1.894579	2.36462	2.99795	3.49948	5.4079
8	0.261921	0.706387	1.396815	1.859548	2.30600	2.89646	3.35539	5.0413
9	0.260955	0.702722	1.383029	1.833113	2.26216	2.82144	3.24984	4.7809
10	0.260185	0.699812	1.372184	1.812461	2.22814	2.76377	3.16927	4.5869
11	0.259556	0.697445	1.363430	1.795885	2.20099	2.71808	3.10581	4.4370
12	0.259033	0.695483	1.356217	1.782288	2.17881	2.68100	3.05454	4.3178
13	0.258591	0.693829	1.350171	1.770933	2.16037	2.65031	3.01228	4.2208
14	0.258213	0.692417	1.345030	1.761310	2.14479	2.62449	2.97684	4.1405
15	0.257885	0.691197	1.340606	1.753050	2.13145	2.60248	2.94671	4.0728
16	0.257599	0.690132	1.336757	1.745884	2.11991	2.58349	2.92078	4.0150
17	0.257347	0.689195	1.333379	1.739607	2.10982	2.56693	2.89823	3.9651
18	0.257123	0.688364	1.330391	1.734064	2.10092	2.55238	2.87844	3.9216
19	0.256923	0.687621	1.327728	1.729133	2.09302	2.53948	2.86093	3.8834
20	0.256743	0.686954	1.325341	1.724718	2.08596	2.52798	2.84534	3.8495
21	0.256580	0.686352	1.323188	1.720743	2.07961	2.51765	2.83136	3.8193
22	0.256432	0.685805	1.321237	1.717144	2.07387	2.50832	2.81876	3.7921
23	0.256297	0.685306	1.319460	1.713872	2.06866	2.49987	2.80734	3.7676
24	0.256173	0.684850	1.317836	1.710882	2.06390	2.49216	2.79694	3.7454
25	0.256060	0.684430	1.316345	1.708141	2.05954	2.48511	2.78744	3.7251
26	0.255955	0.684043	1.314972	1.705618	2.05553	2.47863	2.77871	3.7066
27	0.255858	0.683685	1.313703	1.703288	2.05183	2.47266	2.77068	3.6896
28	0.255768	0.683353	1.312527	1.701131	2.04841	2.46714	2.76326	3.6739
29	0.255684	0.683044	1.311434	1.699127	2.04523	2.46202	2.75639	3.6594
30	0.255605	0.682756	1.310415	1.697261	2.04227	2.45726	2.75000	3.6460

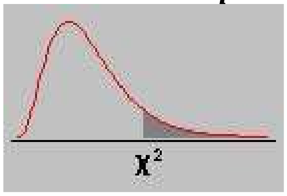
inf	0.253347	0.674490	1.281552	1.644854	1.95996	2.32635	2.57583	3.2905

Chi-Square Table



Like the Student's *t*-Distribution, the *Chi-square* distribtuion's shape is determined by its degrees of freedom. The animation above shows the shape of the *Chi-square* distribution as the degrees of freedom increase (1, 2, 5, 10, 25 and 50). For examples of tests of hypothesis which use the *Chi-square distribution*, see Statistics in crosstabulation tables in the Basic Statistics and Tables chapter as well as the Nonlinear Estimation chapter. See also, Chi-square Distribution. As shown in the illustration below, the values inside this table are critical values of the Chi-square distribution with the corresponding degrees of freedom. To determine the value from a Chi-square distribution (with a specific degree of freedom) which has a given area above it, go to the given area column and the desired degree of freedom row. For example, the .25 critical value for a Chi-square with 4 degrees of freedom is 5.38527. This means that the area to the right of 5.38527 in a Chi-square distribution with 4 degrees of freedom is .25.

Right tail areas for the *Chi-square* Distribution

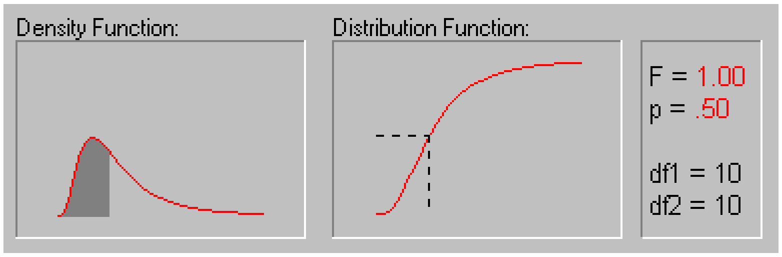


df\area	**.995**	**.990**	**.975**	**.950**	**.900**	**.750**	**.500**	**.250**	**.100**	**.050**	**.025**	**.010**	**.005**
1	0.00004	0.00016	0.00098	0.00393	0.01579	0.10153	0.45494	1.32330	2.70554	3.84146	5.02389	6.63490	7.87944
2	0.01003	0.02010	0.05064	0.10259	0.21072	0.57536	1.38629	2.77259	4.60517	5.99146	7.37776	9.21034	10.59663
3	0.07172	0.11483	0.21580	0.35185	0.58437	1.21253	2.36597	4.10834	6.25139	7.81473	9.34840	11.34487	12.83816
4	0.20699	0.29711	0.48442	0.71072	1.06362	1.92256	3.35669	5.38527	7.77944	9.48773	11.14329	13.27670	14.86026
5	0.411	0.554	0.831	1.145	1.610	2.674	4.351	6.625	9.236	11.07	12.83	15.08	16.74

	74	30	21	48	31	60	46	68	36	050	250	627	960
6	0.67573	0.87209	1.23734	1.63538	2.20413	3.45460	5.34812	7.84080	10.64464	12.59159	14.44938	16.81189	18.54758
7	0.98926	1.23904	1.68987	2.16735	2.83311	4.25485	6.34581	9.03715	12.01704	14.06714	16.01276	18.47531	20.27774
8	1.34441	1.64650	2.17973	2.73264	3.48954	5.07064	7.34412	10.21885	13.36157	15.50731	17.53455	20.09024	21.95495
9	1.73493	2.08790	2.70039	3.32511	4.16816	5.89883	8.34283	11.38875	14.68366	16.91898	19.02277	21.66599	23.58935
10	2.15586	2.55821	3.24697	3.94030	4.86518	6.73720	9.34182	12.54886	15.98718	18.30704	20.48318	23.20925	25.18818
11	2.60322	3.05348	3.81575	4.57481	5.57778	7.58414	10.34100	13.70069	17.27501	19.67514	21.92005	24.72497	26.75685
12	3.07382	3.57057	4.40379	5.22603	6.30380	8.43842	11.34032	14.84540	18.54935	21.02607	23.33666	26.21697	28.29952
13	3.56503	4.10692	5.00875	5.89186	7.04150	9.29907	12.33976	15.98391	19.81193	22.36203	24.73560	27.68825	29.81947
14	4.07467	4.66043	5.62873	6.57063	7.78953	10.16531	13.33927	17.11693	21.06414	23.68479	26.11895	29.14124	31.31935
15	4.60092	5.22935	6.26214	7.26094	8.54676	11.03654	14.33886	18.24509	22.30713	24.99579	27.48839	30.57791	32.80132
16	5.14221	5.81221	6.90766	7.96165	9.31224	11.91222	15.33850	19.36886	23.54183	26.29623	28.84535	31.99993	34.26719
17	5.69722	6.40776	7.56419	8.67176	10.08519	12.79193	16.33818	20.48868	24.76904	27.58711	30.19101	33.40866	35.71847
18	6.26480	7.01491	8.23075	9.39046	10.86494	13.67529	17.33790	21.60489	25.98942	28.86930	31.52638	34.80531	37.15645
19	6.84397	7.63273	8.90652	10.11701	11.65091	14.56200	18.33765	22.71781	27.20357	30.14353	32.85233	36.19087	38.58226
20	7.43384	8.26040	9.59078	10.85081	12.44261	15.45177	19.33743	23.82769	28.41198	31.41043	34.16961	37.56623	39.99685
21	8.03365	8.89720	10.28290	11.59131	13.23960	16.34438	20.33723	24.93478	29.61509	32.67057	35.47888	38.93217	41.40106
22	8.642	9.542	10.98	12.33	14.04	17.23	21.33	26.03	30.81	33.92	36.78	40.28	42.79

	72	49	232	801	149	962	704	927	328	444	071	936	565
23	9.26042	10.19572	11.68855	13.09051	14.84796	18.13730	22.33688	27.14134	32.00690	35.17246	38.07563	41.63840	44.18128
24	9.88623	10.85636	12.40115	13.84843	15.65868	19.03725	23.33673	28.24115	33.19624	36.41503	39.36408	42.97982	45.55851
25	10.51965	11.52398	13.11972	14.61141	16.47341	19.93934	24.33659	29.33885	34.38159	37.65248	40.64647	44.31410	46.92789
26	11.16024	12.19815	13.84390	15.37916	17.29188	20.84343	25.33646	30.43457	35.56317	38.88514	41.92317	45.64168	48.28988
27	11.80759	12.87850	14.57338	16.15140	18.11390	21.74940	26.33634	31.52841	36.74122	40.11327	43.19451	46.96294	49.64492
28	12.46134	13.56471	15.30786	16.92788	18.93924	22.65716	27.33623	32.62049	37.91592	41.33714	44.46079	48.27824	50.99338
29	13.12115	14.25645	16.04707	17.70837	19.76774	23.56659	28.33613	33.71091	39.08747	42.55697	45.72229	49.58788	52.33562
30	13.78672	14.95346	16.79077	18.49266	20.59923	24.47761	29.33603	34.79974	40.25602	43.77297	46.97924	50.89218	53.67196

F Distribution Tables

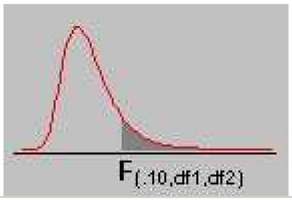


F = 1.00
p = .50

df1 = 10
df2 = 10

The F distribution is a right-skewed distribution used most commonly in Analysis of Variance (see ANOVA/MANOVA). The F distribution is a ratio of two *Chi-square* distributions, and a specific F distribution is denoted by the degrees of freedom for the numerator Chi-square and the degrees of freedom for the denominator Chi-square. An example of the $F_{(10,10)}$ distribution is shown in the animation above. When referencing the F distribution, the numerator degrees of freedom are always given first, as switching the order of degrees of freedom changes the distribution (e.g., $F_{(10,12)}$ does not equal $F_{(12,10)}$). For the four F tables below, the rows represent denominator degrees of freedom and the columns represent numerator degrees of freedom. The right tail area is given in the name of the table. For example, to determine the .05 critical value for an F distribution with 10 and 12 degrees of freedom, look in the 10 column (numerator) and 12 row (denominator) of the F Table for alpha=.05. $F_{(.05, 10, 12)}$ = 2.7534.

F Table for alpha=.10 .

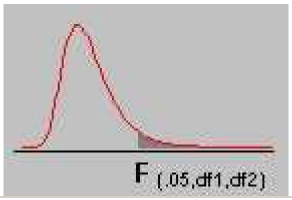


df2/df1	1	2	3	4	5	6	7	8	9	10	12	15	20	24	30	40	60	120	INF
1	39.86346	49.50000	53.59324	55.83296	57.24008	58.20442	58.90595	59.43898	59.85759	60.19498	60.70521	61.22034	61.74029	62.00205	62.26497	62.52905	62.79428	63.06064	63.32812
2	8.52632	9.00000	9.16179	9.24342	9.29263	9.32553	9.34908	9.36677	9.38054	9.39157	9.40813	9.42471	9.44131	9.44962	9.45793	9.46624	9.47456	9.48289	9.49122
3	5.53832	5.46238	5.39077	5.34264	5.30916	5.28473	5.26619	5.25167	5.24000	5.23041	5.21562	5.20031	5.18448	5.17636	5.16811	5.15972	5.15119	5.14251	5.13370
4	4.54477	4.32456	4.19086	4.10725	4.05058	4.00975	3.97897	3.95494	3.93567	3.91988	3.89553	3.87036	3.84434	3.83099	3.81742	3.80361	3.78957	3.77527	3.76073
5	4.06042	3.77972	3.61948	3.52020	3.45298	3.40451	3.36790	3.33928	3.31628	3.29740	3.26824	3.23801	3.20665	3.19052	3.17408	3.15732	3.14023	3.12279	3.10500
6	3.77595	3.46330	3.28876	3.18076	3.10751	3.05455	3.01446	2.98304	2.95774	2.93693	2.90472	2.87122	2.83634	2.81834	2.79996	2.78117	2.76195	2.74229	2.72216
7	3.58943	3.25744	3.07407	2.96053	2.88334	2.82739	2.78493	2.75158	2.72468	2.70251	2.66811	2.63223	2.59473	2.57533	2.55546	2.53510	2.51422	2.49279	2.47079
8	3.45792	3.11312	2.92380	2.80643	2.72645	2.66833	2.62413	2.58935	2.56124	2.53804	2.50196	2.46422	2.42464	2.40410	2.38302	2.36136	2.33910	2.31618	2.29257
9	3.36030	3.00645	2.81286	2.69268	2.61061	2.55086	2.50531	2.46941	2.44034	2.41632	2.37888	2.33962	2.29832	2.27683	2.25472	2.23196	2.20849	2.18427	2.15923

10	3.28502	2.92447	2.72767	2.60534	2.52164	2.46058	2.41397	2.37715	2.34731	2.32260	2.28405	2.24351	2.20074	2.17843	2.15543	2.13169	2.10716	2.08176	2.05542
11	3.22520	2.85951	2.66023	2.53619	2.45118	2.38907	2.34157	2.30400	2.27350	2.24823	2.20873	2.16709	2.12305	2.10001	2.07621	2.05161	2.02612	1.99965	1.97211
12	3.17655	2.80680	2.60552	2.48010	2.39402	2.33102	2.28278	2.24457	2.21352	2.18776	2.14744	2.10485	2.05968	2.03599	2.01149	1.98610	1.95973	1.93228	1.90361
13	3.13621	2.76317	2.56027	2.43371	2.34672	2.28298	2.23410	2.19535	2.16382	2.13763	2.09659	2.05316	2.00698	1.98272	1.95757	1.93147	1.90429	1.87591	1.84620
14	3.10221	2.72647	2.52222	2.39469	2.30694	2.24256	2.19313	2.15390	2.12195	2.09540	2.05371	2.00953	1.96245	1.93766	1.91193	1.88516	1.85723	1.82800	1.79728
15	3.07319	2.69517	2.48979	2.36143	2.27302	2.20808	2.15818	2.11853	2.08621	2.05932	2.01707	1.97222	1.92431	1.89904	1.87277	1.84539	1.81676	1.78672	1.75505
16	3.04811	2.66817	2.46181	2.33274	2.24376	2.17833	2.12800	2.08798	2.05533	2.02815	1.98539	1.93992	1.89127	1.86556	1.83879	1.81084	1.78156	1.75075	1.71817
17	3.02623	2.64464	2.43743	2.30775	2.21825	2.15239	2.10169	2.06134	2.02839	2.00094	1.95772	1.91169	1.86236	1.83624	1.80901	1.78053	1.75063	1.71909	1.68564
18	3.00698	2.62395	2.41601	2.28577	2.19583	2.12958	2.07854	2.03789	2.00467	1.97698	1.93334	1.88681	1.83685	1.81035	1.78269	1.75371	1.72322	1.69099	1.65671
19	2.98990	2.60561	2.39702	2.26630	2.17596	2.10936	2.05802	2.01710	1.98364	1.95573	1.91170	1.86471	1.81416	1.78731	1.75924	1.72979	1.69876	1.66587	1.63077
20	2.97465	2.58925	2.38009	2.24893	2.15823	2.09132	2.03970	1.99853	1.96485	1.93674	1.89236	1.84494	1.79384	1.76667	1.73822	1.70833	1.67678	1.64326	1.60738
21	2.96096	2.57457	2.36489	2.23334	2.14231	2.07512	2.02325	1.98186	1.94797	1.91967	1.87497	1.82715	1.77555	1.74807	1.71927	1.68896	1.65691	1.62278	1.58615
22	2.94858	2.56131	2.35117	2.21927	2.12794	2.06050	2.00840	1.96680	1.93273	1.90425	1.85925	1.81106	1.75899	1.73122	1.70208	1.67138	1.63885	1.60415	1.56678

23	2.93736	2.54929	2.33873	2.20651	2.11491	2.04723	1.99492	1.95312	1.91888	1.89025	1.84497	1.79643	1.74392	1.71588	1.68643	1.65535	1.62237	1.58711	1.54903
24	2.92712	2.53833	2.32739	2.19488	2.10303	2.03513	1.98263	1.94066	1.90625	1.87748	1.83194	1.78308	1.73015	1.70185	1.67210	1.64067	1.60726	1.57146	1.53270
25	2.91774	2.52831	2.31702	2.18424	2.09216	2.02406	1.97138	1.92925	1.89469	1.86578	1.82000	1.77083	1.71752	1.68898	1.65895	1.62718	1.59335	1.55703	1.51760
26	2.90913	2.51910	2.30749	2.17447	2.08218	2.01389	1.96104	1.91876	1.88407	1.85503	1.80902	1.75957	1.70589	1.67712	1.64682	1.61472	1.58050	1.54368	1.50360
27	2.90119	2.51061	2.29871	2.16546	2.07298	2.00452	1.95151	1.90909	1.87427	1.84511	1.79889	1.74917	1.69514	1.66616	1.63560	1.60320	1.56859	1.53129	1.49057
28	2.89385	2.50276	2.29060	2.15714	2.06447	1.99585	1.94270	1.90014	1.86520	1.83593	1.78951	1.73954	1.68519	1.65600	1.62519	1.59250	1.55753	1.51976	1.47841
29	2.88703	2.49548	2.28307	2.14941	2.05658	1.98781	1.93452	1.89184	1.85679	1.82741	1.78081	1.73060	1.67593	1.64655	1.61551	1.58253	1.54721	1.50899	1.46704
30	2.88069	2.48872	2.27607	2.14223	2.04925	1.98033	1.92692	1.88412	1.84896	1.81949	1.77270	1.72227	1.66731	1.63774	1.60648	1.57323	1.53757	1.49891	1.45636
40	2.83535	2.44037	2.22609	2.09095	1.99682	1.92688	1.87252	1.82886	1.79290	1.76269	1.71456	1.66241	1.60515	1.57411	1.54108	1.50562	1.46716	1.42476	1.37691
60	2.79107	2.39325	2.17741	2.04099	1.94571	1.87472	1.81939	1.77483	1.73802	1.70701	1.65743	1.60337	1.54349	1.51072	1.47554	1.43734	1.39520	1.34757	1.29146
120	2.74781	2.34734	2.12999	1.99230	1.89587	1.82381	1.76748	1.72196	1.68425	1.65238	1.60120	1.54500	1.48207	1.44723	1.40938	1.36760	1.32034	1.26457	1.19256
inf	2.70554	2.30259	2.08380	1.94486	1.84727	1.77411	1.71672	1.67020	1.63152	1.59872	1.54578	1.48714	1.42060	1.38318	1.34187	1.29513	1.23995	1.16860	1.00000

F Table for alpha=.05 .

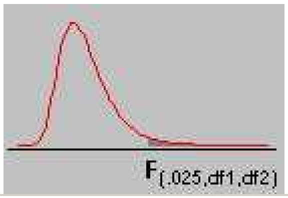


df2/df1	1	2	3	4	5	6	7	8	9	10	12	15	20	24	30	40	60	120	INF
1	161.4476	199.5000	215.7073	224.5832	230.1619	233.9860	236.7684	238.8827	240.5433	241.8817	243.9060	245.9499	248.0131	249.0518	250.0951	251.1432	252.1957	253.2529	254.3144
2	18.5128	19.0000	19.1643	19.2468	19.2964	19.3295	19.3532	19.3710	19.3848	19.3959	19.4125	19.4291	19.4458	19.4541	19.4624	19.4707	19.4791	19.4874	19.4957
3	10.1280	9.5521	9.2766	9.1172	9.0135	8.9406	8.8867	8.8452	8.8123	8.7855	8.7446	8.7029	8.6602	8.6385	8.6166	8.5944	8.5720	8.5494	8.5264
4	7.7086	6.9443	6.5914	6.3882	6.2561	6.1631	6.0942	6.0410	5.9988	5.9644	5.9117	5.8578	5.8025	5.7744	5.7459	5.7170	5.6877	5.6581	5.6281
5	6.6079	5.7861	5.4095	5.1922	5.0503	4.9503	4.8759	4.8183	4.7725	4.7351	4.6777	4.6188	4.5581	4.5272	4.4957	4.4638	4.4314	4.3985	4.3650
6	5.9874	5.1433	4.7571	4.5337	4.3874	4.2839	4.2067	4.1468	4.0990	4.0600	3.9999	3.9381	3.8742	3.8415	3.8082	3.7743	3.7398	3.7047	3.6689
7	5.5914	4.7374	4.3468	4.1203	3.9715	3.8660	3.7870	3.7257	3.6767	3.6365	3.5747	3.5107	3.4445	3.4105	3.3758	3.3404	3.3043	3.2674	3.2298
8	5.3177	4.4590	4.0662	3.8379	3.6875	3.5806	3.5005	3.4381	3.3881	3.3472	3.2839	3.2184	3.1503	3.1152	3.0794	3.0428	3.0053	2.9669	2.9276
9	5.1174	4.2565	3.8625	3.6331	3.4817	3.3738	3.2927	3.2296	3.1789	3.1373	3.0729	3.0061	2.9365	2.9005	2.8637	2.8259	2.7872	2.7475	2.7067
10	4.9646	4.1028	3.7083	3.4780	3.3258	3.2172	3.1355	3.0717	3.0204	2.9782	2.9130	2.8450	2.7740	2.7372	2.6996	2.6609	2.6211	2.5801	2.5379

11	4.8443	3.9823	3.5874	3.3567	3.2039	3.0946	3.0123	2.9480	2.8962	2.8536	2.7876	2.7186	2.6464	2.6090	2.5705	2.5309	2.4901	2.4480	2.4045
12	4.7472	3.8853	3.4903	3.2592	3.1059	2.9961	2.9134	2.8486	2.7964	2.7534	2.6866	2.6169	2.5436	2.5055	2.4663	2.4259	2.3842	2.3410	2.2962
13	4.6672	3.8056	3.4105	3.1791	3.0254	2.9153	2.8321	2.7669	2.7144	2.6710	2.6037	2.5331	2.4589	2.4202	2.3803	2.3392	2.2966	2.2524	2.2064
14	4.6001	3.7389	3.3439	3.1122	2.9582	2.8477	2.7642	2.6987	2.6458	2.6022	2.5342	2.4630	2.3879	2.3487	2.3082	2.2664	2.2229	2.1778	2.1307
15	4.5431	3.6823	3.2874	3.0556	2.9013	2.7905	2.7066	2.6408	2.5876	2.5437	2.4753	2.4034	2.3275	2.2878	2.2468	2.2043	2.1601	2.1141	2.0658
16	4.4940	3.6337	3.2389	3.0069	2.8524	2.7413	2.6572	2.5911	2.5377	2.4935	2.4247	2.3522	2.2756	2.2354	2.1938	2.1507	2.1058	2.0589	2.0096
17	4.4513	3.5915	3.1968	2.9647	2.8100	2.6987	2.6143	2.5480	2.4943	2.4499	2.3807	2.3077	2.2304	2.1898	2.1477	2.1040	2.0584	2.0107	1.9604
18	4.4139	3.5546	3.1599	2.9277	2.7729	2.6613	2.5767	2.5102	2.4563	2.4117	2.3421	2.2686	2.1906	2.1497	2.1071	2.0629	2.0166	1.9681	1.9168
19	4.3807	3.5219	3.1274	2.8951	2.7401	2.6283	2.5435	2.4768	2.4227	2.3779	2.3080	2.2341	2.1555	2.1141	2.0712	2.0264	1.9795	1.9302	1.8780
20	4.3512	3.4928	3.0984	2.8661	2.7109	2.5990	2.5140	2.4471	2.3928	2.3479	2.2776	2.2033	2.1242	2.0825	2.0391	1.9938	1.9464	1.8963	1.8432
21	4.3248	3.4668	3.0725	2.8401	2.6848	2.5727	2.4876	2.4205	2.3660	2.3210	2.2504	2.1757	2.0960	2.0540	2.0102	1.9645	1.9165	1.8657	1.8117
22	4.3009	3.4434	3.0491	2.8167	2.6613	2.5491	2.4638	2.3965	2.3419	2.2967	2.2258	2.1508	2.0707	2.0283	1.9842	1.9380	1.8894	1.8380	1.7831
23	4.2793	3.4221	3.0280	2.7955	2.6400	2.5277	2.4422	2.3748	2.3201	2.2747	2.2036	2.1282	2.0476	2.0050	1.9605	1.9139	1.8648	1.8128	1.7570

24	4.2597	3.4028	3.0088	2.7763	2.6207	2.5082	2.4226	2.3551	2.3002	2.2547	2.1834	2.1077	2.0267	1.9838	1.9390	1.8920	1.8424	1.7896	1.7330
25	4.2417	3.3852	2.9912	2.7587	2.6030	2.4904	2.4047	2.3371	2.2821	2.2365	2.1649	2.0889	2.0075	1.9643	1.9192	1.8718	1.8217	1.7684	1.7110
26	4.2252	3.3690	2.9752	2.7426	2.5868	2.4741	2.3883	2.3205	2.2655	2.2197	2.1479	2.0716	1.9898	1.9464	1.9010	1.8533	1.8027	1.7488	1.6906
27	4.2100	3.3541	2.9604	2.7278	2.5719	2.4591	2.3732	2.3053	2.2501	2.2043	2.1323	2.0558	1.9736	1.9299	1.8842	1.8361	1.7851	1.7306	1.6717
28	4.1960	3.3404	2.9467	2.7141	2.5581	2.4453	2.3593	2.2913	2.2360	2.1900	2.1179	2.0411	1.9586	1.9147	1.8687	1.8203	1.7689	1.7138	1.6541
29	4.1830	3.3277	2.9340	2.7014	2.5454	2.4324	2.3463	2.2783	2.2229	2.1768	2.1045	2.0275	1.9446	1.9005	1.8543	1.8055	1.7537	1.6981	1.6376
30	4.1709	3.3158	2.9223	2.6896	2.5336	2.4205	2.3343	2.2662	2.2107	2.1646	2.0921	2.0148	1.9317	1.8874	1.8409	1.7918	1.7396	1.6835	1.6223
40	4.0847	3.2317	2.8387	2.6060	2.4495	2.3359	2.2490	2.1802	2.1240	2.0772	2.0035	1.9245	1.8389	1.7929	1.7444	1.6928	1.6373	1.5766	1.5089
60	4.0012	3.1504	2.7581	2.5252	2.3683	2.2541	2.1665	2.0970	2.0401	1.9926	1.9174	1.8364	1.7480	1.7001	1.6491	1.5943	1.5343	1.4673	1.3893
120	3.9201	3.0718	2.6802	2.4472	2.2899	2.1750	2.0868	2.0164	1.9588	1.9105	1.8337	1.7505	1.6587	1.6084	1.5543	1.4952	1.4290	1.3519	1.2539
inf	3.8415	2.9957	2.6049	2.3719	2.2141	2.0986	2.0096	1.9384	1.8799	1.8307	1.7522	1.6664	1.5705	1.5173	1.4591	1.3940	1.3180	1.2214	1.0000

F Table for alpha=.025 .



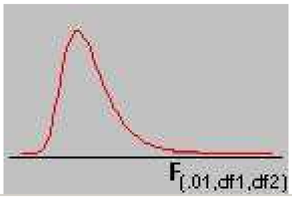
df2/df1	1	2	3	4	5	6	7	8	9	10	12	15	20	24	30	40	60	120	INF
1	647.7890	799.5000	864.1630	899.5833	921.8479	937.1111	948.2169	956.6562	963.2846	968.6274	976.7079	984.8668	993.1028	997.2492	1001.414	1005.598	1009.800	1014.020	1018.258
2	38.5063	39.0000	39.1655	39.2484	39.2982	39.3315	39.3552	39.3730	39.3869	39.3980	39.4146	39.4313	39.4479	39.4562	39.465	39.473	39.481	39.490	39.498
3	17.4434	16.0441	15.4392	15.1010	14.8848	14.7347	14.6244	14.5399	14.4731	14.4189	14.3366	14.2527	14.1674	14.1241	14.081	14.037	13.992	13.947	13.902
4	12.2179	10.6491	9.9792	9.6045	9.3645	9.1973	9.0741	8.9796	8.9047	8.8439	8.7512	8.6565	8.5599	8.5109	8.461	8.411	8.360	8.309	8.257
5	10.0070	8.4336	7.7636	7.3879	7.1464	6.9777	6.8531	6.7572	6.6811	6.6192	6.5245	6.4277	6.3286	6.2780	6.227	6.175	6.123	6.069	6.015
6	8.8131	7.2599	6.5988	6.2272	5.9876	5.8198	5.6955	5.5996	5.5234	5.4613	5.3662	5.2687	5.1684	5.1172	5.065	5.012	4.959	4.904	4.849
7	8.0727	6.5415	5.8898	5.5226	5.2852	5.1186	4.9949	4.8993	4.8232	4.7611	4.6658	4.5678	4.4667	4.4150	4.362	4.309	4.254	4.199	4.142
8	7.5709	6.0595	5.4160	5.0526	4.8173	4.6517	4.5286	4.4333	4.3572	4.2951	4.1997	4.1012	3.9995	3.9472	3.894	3.840	3.784	3.728	3.670
9	7.2093	5.7147	5.0781	4.7181	4.4844	4.3197	4.1970	4.1020	4.0260	3.9639	3.8682	3.7694	3.6669	3.6142	3.560	3.505	3.449	3.392	3.333
10	6.9367	5.4564	4.8256	4.4683	4.2361	4.0721	3.9498	3.8549	3.7790	3.7168	3.6209	3.5217	3.4185	3.3654	3.311	3.255	3.198	3.140	3.080
11	6.7	5.2	4.6	4.2	4.0	3.8	3.7	3.6	3.5	3.5	3.4	3.3	3.2	3.1	3.1	3.0	3.0	2.9	2.8

	241	559	300	751	440	807	586	638	879	257	296	299	261	725	18	61	04	44	83
12	6.5538	5.0959	4.4742	4.1212	3.8911	3.7283	3.6065	3.5118	3.4358	3.3736	3.2773	3.1772	3.0728	3.0187	2.963	2.906	2.848	2.787	2.725
13	6.4143	4.9653	4.3472	3.9959	3.7667	3.6043	3.4827	3.3880	3.3120	3.2497	3.1532	3.0527	2.9477	2.8932	2.837	2.780	2.720	2.659	2.595
14	6.2979	4.8567	4.2417	3.8919	3.6634	3.5014	3.3799	3.2853	3.2093	3.1469	3.0502	2.9493	2.8437	2.7888	2.732	2.674	2.614	2.552	2.487
15	6.1995	4.7650	4.1528	3.8043	3.5764	3.4147	3.2934	3.1987	3.1227	3.0602	2.9633	2.8621	2.7559	2.7006	2.644	2.585	2.524	2.461	2.395
16	6.1151	4.6867	4.0768	3.7294	3.5021	3.3406	3.2194	3.1248	3.0488	2.9862	2.8890	2.7875	2.6808	2.6252	2.568	2.509	2.447	2.383	2.316
17	6.0420	4.6189	4.0112	3.6648	3.4379	3.2767	3.1556	3.0610	2.9849	2.9222	2.8249	2.7230	2.6158	2.5598	2.502	2.442	2.380	2.315	2.247
18	5.9781	4.5597	3.9539	3.6083	3.3820	3.2209	3.0999	3.0053	2.9291	2.8664	2.7689	2.6667	2.5590	2.5027	2.445	2.384	2.321	2.256	2.187
19	5.9216	4.5075	3.9034	3.5587	3.3327	3.1718	3.0509	2.9563	2.8801	2.8172	2.7196	2.6171	2.5089	2.4523	2.394	2.333	2.270	2.203	2.133
20	5.8715	4.4613	3.8587	3.5147	3.2891	3.1283	3.0074	2.9128	2.8365	2.7737	2.6758	2.5731	2.4645	2.4076	2.349	2.287	2.223	2.156	2.085
21	5.8266	4.4199	3.8188	3.4754	3.2501	3.0895	2.9686	2.8740	2.7977	2.7348	2.6368	2.5338	2.4247	2.3675	2.308	2.246	2.182	2.114	2.042
22	5.7863	4.3828	3.7829	3.4401	3.2151	3.0546	2.9338	2.8392	2.7628	2.6998	2.6017	2.4984	2.3890	2.3315	2.272	2.210	2.145	2.076	2.003
23	5.7498	4.3492	3.7505	3.4083	3.1835	3.0232	2.9023	2.8077	2.7313	2.6682	2.5699	2.4665	2.3567	2.2989	2.239	2.176	2.111	2.041	1.968
24	5.7166	4.3187	3.7211	3.3794	3.1548	2.9946	2.8738	2.7791	2.7027	2.6396	2.5411	2.4374	2.3273	2.2693	2.209	2.146	2.080	2.010	1.935

25	5.6864	4.2909	3.6943	3.3530	3.1287	2.9685	2.8478	2.7531	2.6766	2.6135	2.5149	2.4110	2.3005	2.2422	2.182	2.118	2.052	1.981	1.906
26	5.6586	4.2655	3.6697	3.3289	3.1048	2.9447	2.8240	2.7293	2.6528	2.5896	2.4908	2.3867	2.2759	2.2174	2.157	2.093	2.026	1.954	1.878
27	5.6331	4.2421	3.6472	3.3067	3.0828	2.9228	2.8021	2.7074	2.6309	2.5676	2.4688	2.3644	2.2533	2.1946	2.133	2.069	2.002	1.930	1.853
28	5.6096	4.2205	3.6264	3.2863	3.0626	2.9027	2.7820	2.6872	2.6106	2.5473	2.4484	2.3438	2.2324	2.1735	2.112	2.048	1.980	1.907	1.829
29	5.5878	4.2006	3.6072	3.2674	3.0438	2.8840	2.7633	2.6686	2.5919	2.5286	2.4295	2.3248	2.2131	2.1540	2.092	2.028	1.959	1.886	1.807
30	5.5675	4.1821	3.5894	3.2499	3.0265	2.8667	2.7460	2.6513	2.5746	2.5112	2.4120	2.3072	2.1952	2.1359	2.074	2.009	1.940	1.866	1.787
40	5.4239	4.0510	3.4633	3.1261	2.9037	2.7444	2.6238	2.5289	2.4519	2.3882	2.2882	2.1819	2.0677	2.0069	1.943	1.875	1.803	1.724	1.637
60	5.2856	3.9253	3.3425	3.0077	2.7863	2.6274	2.5068	2.4117	2.3344	2.2702	2.1692	2.0613	1.9445	1.8817	1.815	1.744	1.667	1.581	1.482
120	5.1523	3.8046	3.2269	2.8943	2.6740	2.5154	2.3948	2.2994	2.2217	2.1570	2.0548	1.9450	1.8249	1.7597	1.690	1.614	1.530	1.433	1.310

inf	5.0239	3.6889	3.1161	2.7858	2.5665	2.4082	2.2875	2.1918	2.1136	2.0483	1.9447	1.8326	1.7085	1.6402	1.566	1.484	1.388	1.268	1.000

F Table for alpha=.01 .



df2/df1	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**12**	**15**	**20**	**24**	**30**	**40**	**60**	**120**	**INF**
1	4052.181	4999.500	5403.352	5624.583	5763.650	5858.986	5928.356	5981.070	6022.473	6055.847	6106.321	6157.285	6208.730	6234.631	6260.649	6286.782	6313.030	6339.391	6365.864
2	98.503	99.000	99.166	99.249	99.299	99.333	99.356	99.374	99.388	99.399	99.416	99.433	99.449	99.458	99.466	99.474	99.482	99.491	99.499
3	34.116	30.817	29.457	28.710	28.237	27.911	27.672	27.489	27.345	27.229	27.052	26.872	26.690	26.598	26.505	26.411	26.316	26.221	26.125
4	21.198	18.000	16.694	15.977	15.522	15.207	14.976	14.799	14.659	14.546	14.374	14.198	14.020	13.929	13.838	13.745	13.652	13.558	13.463
5	16.258	13.274	12.060	11.392	10.967	10.672	10.456	10.289	10.158	10.051	9.888	9.722	9.553	9.466	9.379	9.291	9.202	9.112	9.020
6	13.745	10.925	9.780	9.148	8.746	8.466	8.260	8.102	7.976	7.874	7.718	7.559	7.396	7.313	7.229	7.143	7.057	6.969	6.880
7	12.246	9.547	8.451	7.847	7.460	7.191	6.993	6.840	6.719	6.620	6.469	6.314	6.155	6.074	5.992	5.908	5.824	5.737	5.650
8	11.259	8.649	7.591	7.006	6.632	6.371	6.178	6.029	5.911	5.814	5.667	5.515	5.359	5.279	5.198	5.116	5.032	4.946	4.859
9	10.561	8.022	6.992	6.422	6.057	5.802	5.613	5.467	5.351	5.257	5.111	4.962	4.808	4.729	4.649	4.567	4.483	4.398	4.311
10	10.044	7.559	6.552	5.994	5.636	5.386	5.200	5.057	4.942	4.849	4.706	4.558	4.405	4.327	4.247	4.165	4.082	3.996	3.909
11	9.6	7.2	6.2	5.6	5.3	5.0	4.8	4.7	4.6	4.5	4.3	4.2	4.0	4.0	3.9	3.8	3.7	3.6	3.6

	46	06	17	68	16	69	86	44	32	39	97	51	99	21	41	60	76	90	02
12	9.330	6.927	5.953	5.412	5.064	4.821	4.640	4.499	4.388	4.296	4.155	4.010	3.858	3.780	3.701	3.619	3.535	3.449	3.361
13	9.074	6.701	5.739	5.205	4.862	4.620	4.441	4.302	4.191	4.100	3.960	3.815	3.665	3.587	3.507	3.425	3.341	3.255	3.165
14	8.862	6.515	5.564	5.035	4.695	4.456	4.278	4.140	4.030	3.939	3.800	3.656	3.505	3.427	3.348	3.266	3.181	3.094	3.004
15	8.683	6.359	5.417	4.893	4.556	4.318	4.142	4.004	3.895	3.805	3.666	3.522	3.372	3.294	3.214	3.132	3.047	2.959	2.868
16	8.531	6.226	5.292	4.773	4.437	4.202	4.026	3.890	3.780	3.691	3.553	3.409	3.259	3.181	3.101	3.018	2.933	2.845	2.753
17	8.400	6.112	5.185	4.669	4.336	4.102	3.927	3.791	3.682	3.593	3.455	3.312	3.162	3.084	3.003	2.920	2.835	2.746	2.653
18	8.285	6.013	5.092	4.579	4.248	4.015	3.841	3.705	3.597	3.508	3.371	3.227	3.077	2.999	2.919	2.835	2.749	2.660	2.566
19	8.185	5.926	5.010	4.500	4.171	3.939	3.765	3.631	3.523	3.434	3.297	3.153	3.003	2.925	2.844	2.761	2.674	2.584	2.489
20	8.096	5.849	4.938	4.431	4.103	3.871	3.699	3.564	3.457	3.368	3.231	3.088	2.938	2.859	2.778	2.695	2.608	2.517	2.421
21	8.017	5.780	4.874	4.369	4.042	3.812	3.640	3.506	3.398	3.310	3.173	3.030	2.880	2.801	2.720	2.636	2.548	2.457	2.360
22	7.945	5.719	4.817	4.313	3.988	3.758	3.587	3.453	3.346	3.258	3.121	2.978	2.827	2.749	2.667	2.583	2.495	2.403	2.305
23	7.881	5.664	4.765	4.264	3.939	3.710	3.539	3.406	3.299	3.211	3.074	2.931	2.781	2.702	2.620	2.535	2.447	2.354	2.256
24	7.823	5.614	4.718	4.218	3.895	3.667	3.496	3.363	3.256	3.168	3.032	2.889	2.738	2.659	2.577	2.492	2.403	2.310	2.211
25	7.770	5.568	4.675	4.177	3.855	3.627	3.457	3.324	3.217	3.129	2.993	2.850	2.699	2.620	2.538	2.453	2.364	2.270	2.169
26	7.721	5.526	4.637	4.140	3.818	3.591	3.421	3.288	3.182	3.094	2.958	2.815	2.664	2.585	2.503	2.417	2.327	2.233	2.131
27	7.677	5.488	4.601	4.106	3.785	3.558	3.388	3.256	3.149	3.062	2.926	2.783	2.632	2.552	2.470	2.384	2.294	2.198	2.097
28	7.636	5.453	4.568	4.074	3.754	3.528	3.358	3.226	3.120	3.032	2.896	2.753	2.602	2.522	2.440	2.354	2.263	2.167	2.064

29	7.598	5.420	4.538	4.045	3.725	3.499	3.330	3.198	3.092	3.005	2.868	2.726	2.574	2.495	2.412	2.325	2.234	2.138	2.034
30	7.562	5.390	4.510	4.018	3.699	3.473	3.304	3.173	3.067	2.979	2.843	2.700	2.549	2.469	2.386	2.299	2.208	2.111	2.006
40	7.314	5.179	4.313	3.828	3.514	3.291	3.124	2.993	2.888	2.801	2.665	2.522	2.369	2.288	2.203	2.114	2.019	1.917	1.805
60	7.077	4.977	4.126	3.649	3.339	3.119	2.953	2.823	2.718	2.632	2.496	2.352	2.198	2.115	2.028	1.936	1.836	1.726	1.601
120	6.851	4.787	3.949	3.480	3.174	2.956	2.792	2.663	2.559	2.472	2.336	2.192	2.035	1.950	1.860	1.763	1.656	1.533	1.381
inf	6.635	4.605	3.782	3.319	3.017	2.802	2.639	2.511	2.407	2.321	2.185	2.039	1.878	1.791	1.696	1.592	1.473	1.325	1.000

REFERENCES CITED

Abraham, B., & Ledolter, J. (1983). *Statistical methods for forecasting*. New York: Wiley.

Adorno, T. W., Frenkel-Brunswik, E., Levinson, D. J., & Sanford, R. N. (1950). *The authoritarian personality.* New York: Harper.

Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. *Proceedings of the 1993 ACM SIGMOD Conference*, Washington, DC.

Agrawal, R. & Srikant, R. (1994). Fast algorithms for mining association rules. *Proceedings of the 20th VLDB Conference*. Santiago, Chile.

Agresti, Alan (1996). *An Introduction to Categorical Data Analysis*. New York: Wiley.

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In B. N. Petrov and F. Csaki (Eds.), *Second International Symposium on Information Theory*. Budapest: Akademiai Kiado.

Akaike, H. (1983). Information measures and model selection. *Bulletin of the International Statistical Institute: Proceedings of the 44th Session, Volume 1*. Pages 277-290.

Aldrich, J. H., & Nelson, F. D. (1984). *Linear probability, logit, and probit models*. Beverly Hills, CA: Sage Publications.

Almon, S. (1965). The distributed lag between capital appropriations and expenditures. *Econometrica*, *33*, 178-196.

American Supplier Institute (1984-1988). *Proceedings of Supplier Symposia on Taguchi Methods*. (April, 1984; November, 1984; October, 1985; October, 1986; October, 1987; October, 1988), Dearborn, MI: American Supplier Institute.

Anderson, O. D. (1976). *Time series analysis and forecasting*. London: Butterworths.

Anderson, S. B., & Maier, M. H. (1963). 34,000 pupils and how they grew. *Journal of Teacher Education*, *14*, 212-216.

Anderson, T. W. (1958). *An introduction to multivariate statistical analysis.* New York: Wiley.

Anderson, T. W. (1984). *An introduction to multivariate statistical analysis* (2nd ed.). New York: Wiley.

Anderson, T. W., & Rubin, H. (1956). Statistical inference in factor analysis. *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability*. Berkeley: The University of California Press.

Andrews, D. F. (1972). Plots of high-dimensional data. *Biometrics*, *28*, 125-136.

ASQC/AIAG (1990). *Measurement systems analysis reference manual*. Troy, MI: AIAG.

ASQC/AIAG (1991). *Fundamental statistical process control reference manual*. Troy, MI: AIAG.

AT&T (1956). *Statistical quality control handbook, Select code 700-444*. Indianapolis, AT&T Technologies.

Auble, D. (1953). Extended tables for the Mann-Whitney statistic. *Bulletin of the Institute of Educational Research, Indiana University*, *1*, No. 2.

Bagozzi, R. P., & Yi, Y. (1989). On the use of structural equation models in experimental design. *Journal of Marketing Research*, *26,* 271-284.

Bagozzi, R. P., Yi, Y., & Singh, S. (1991). On the use of structural equation models in experimental designs: Two extensions. *International Journal of Research in Marketing, 8,* 125-140.

Bailey, A. L. (1931). The analysis of covariance. *Journal of the American Statistical Association*, *26*, 424-435.

Bails, D. G., & Peppers, L. C. (1982). *Business fluctuations: Forecasting techniques and applications*. Englewood Cliffs, NJ: Prentice-Hall.

Bain, L. J. (1978). *Statistical analysis of reliability and life-testing models*. New York: Decker.

Bain, L. J. and Engelhart, M. (1989) *Introduction to Probability and Mathematical Statistics*. Kent, MA: PWS.

Baird, J. C. (1970). *Psychophysical analysis of visual space.* New York: Pergamon Press.

Baird, J. C., & Noma, E. (1978). *Fundamentals of scaling and psychophysics.* New York: Wiley.

Barcikowski, R., & Stevens, J. P. (1975). A Monte Carlo study of the stability of canonical correlations, canonical weights, and canonical variate-variable correlations. *Multivariate Behavioral Research, 10,* 353-364.

Barker, T. B. (1986). Quality engineering by design: Taguchi's philosophy. *Quality Progress*, *19*, 32-42.

Barlow, R. E., & Proschan, F. (1975). *Statistical theory of reliability and life testing*. New York: Holt, Rinehart, & Winston.

Barlow, R. E., Marshall, A. W., & Proschan, F. (1963). Properties of probability distributions with monotone hazard rate. *Annals of Mathematical Statistics*, *34*, 375-389.

Barnard, G. A. (1959). Control charts and stochastic processes. *Journal of the Royal Statistical Society,* Ser. B, *21,* 239.

Bartholomew, D. J. (1984). The foundations of factor analysis. *Biometrika*, *71*, 221-232.

Bates, D. M., & Watts, D. G. (1988). *Nonlinear regression analysis and its applications.* New York: Wiley.

Bayne, C. K., & Rubin, I. B. (1986). *Practical experimental designs and optimization methods for chemists.* Deerfield Beach, FL: VCH Publishers.

Becker, R. A., Denby, L., McGill, R., & Wilks, A. R. (1986). Datacryptanalysis: A case study. *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 92-97.

Bellman, R. (1961). *Adaptive Control Processes: A Guided Tour*. Princeton University Press.

Belsley, D. A., Kuh, E., and Welsch, R. E. (1980). *Regression Diagnostics*. New York: Wiley.

Bendat, J. S. (1990). *Nonlinear system analysis and identification from random data.* New York: Wiley.

Bentler, P. M, & Bonett, D. G. (1980). Significance tests and goodness of fit in the analysis of covariance structures. *Psychological Bulletin, 88*, 588-606.

Bentler, P. M. (1986). Structural modeling and Psychometrika: A historical perspective on growth and achievements. *Psychometrika, 51*, 35-51.

Bentler, P. M. (1989). *EQS Structural equations program manual.* Los Angeles, CA: BMDP Statistical Software.

Bentler, P. M., & Weeks, D. G. (1979). Interrelations among models for the analysis of moment structures. *Multivariate Behavioral Research, 14*, 169-185.

Bentler, P. M., & Weeks, D. G. (1980). Linear structural equations with latent variables. *Psychometrika, 45*, 289-308.

Benzécri, J. P. (1973). *L'Analyse des Données: T. 2, I' Analyse des correspondances.* Paris: Dunod.

Bergeron, B. (2002). *Essentials of CRM: A guide to customer relationship management.* NY: Wiley.

Berkson, J. (1944). Application of the Logistic Function to Bio-Assay. *Journal of the American Statistical Association, 39,* 357-365.

Berkson, J., & Gage, R. R. (1950). The calculation of survival rates for cancer. *Proceedings of Staff Meetings, Mayo Clinic*, *25*, 250.

Berry, M., J., A., & Linoff, G., S., (2000). *Mastering data mining*. New York: Wiley.

Bhote, K. R. (1988). *World class quality*. New York: AMA Membership Publications.

Binns, B., & Clark, N. (1986). The graphic designer's use of visual syntax. *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 36-41.

Birnbaum, Z. W. (1952). Numerical tabulation of the distribution of Kolmogorov's statistic for finite sample values. *Journal of the American Statistical Association*, *47*, 425-441.

Birnbaum, Z. W. (1953). Distribution-free tests of fit for continuous distribution functions. *Annals of Mathematical Statistics, 24,* 1-8.

Bishop, C. (1995). *Neural Networks for Pattern Recognition.* Oxford: University Press.

Bishop, Y. M. M., Fienberg, S. E., & Holland, P. W. (1975). *Discrete multivariate analysis*. Cambridge, MA: MIT Press.

Bjorck, A. (1967). Solving linear least squares problems by Gram-Schmidt orthonormalization. *Bit*, *7,* 1-21.

Blackman, R. B., & Tukey, J. (1958). *The measurement of power spectral from the point of view of communication engineering.* New York: Dover.

Blackwelder, R. A. (1966). *Taxonomy: A text and reference book.* New York: Wiley.

Blalock, H. M. (1972). *Social statistics* (2nd ed.). New York:McGraw-Hill

Bliss, C. I. (1934). The method of probits. *Science*, *79*, 38-39.

Bloomfield, P. (1976). *Fourier analysis of time series: An introduction*. New York: Wiley.

Bock, R. D. (1963). Programming univariate and multivariate analysis of variance. *Technometrics, 5*, 95-117.

Bock, R. D. (1975). *Multivariate statistical methods in behavioral research*. New York: McGraw-Hill.

Bolch, B.W., & Huang, C. J. (1974). *Multivariate statistical methods for business and economics*. Englewood Cliffs, NJ: Prentice-Hall.

Bollen, K. A. (1989). *Structural equations with latent variables.* New York: John Wiley & Sons.

Borg, I., & Lingoes, J. (1987). *Multidimensional similarity structure analysis*. New York: Springer.

Borg, I., & Shye, S. (in press). *Facet Theory*. Newbury Park: Sage.

Bouland, H. and Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics 59*, 291-294.

Bowker, A. G. (1948). A test for symmetry in contingency tables. *Journal of the American Statistical Association*, *43*, 572-574.

Bowley, A. L. (1897). Relations between the accuracy of an average and that of its constituent parts. *Journal of the Royal Statistical Society*, *60*, 855-866.

Bowley, A. L. (1907). *Elements of Statistics*. London: P. S. King and Son.

Box, G. E. P. (1953). Non-normality and tests on variances. *Biometrika*, *40*, 318-335.

Box, G. E. P. (1954a). Some theorems on quadratic forms applied in the study of analysis of variance problems: I. Effect of inequality of variances in the one-way classification. *Annals of Mathematical Statistics*, *25*, 290-302.

Box, G. E. P. (1954b). Some theorems on quadratic forms applied in the study of analysis of variance problems: II. Effect of inequality of variances and of correlation of errors in the two-way classification. *Annals of Mathematical Statistics*, *25*, 484-498.

Box, G. E. P., & Anderson, S. L. (1955). Permutation theory in the derivation of robust criteria and the study of departures from assumptions. *Journal of the Royal Statistical Society*, *17*, 1-34.

Box, G. E. P., & Behnken, D. W. (1960). Some new three level designs for the study of quantitative variables. *Technometrics*, *2*, 455-475.

Box, G. E. P., & Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society*, *26*, 211-253.

Box, G. E. P., & Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society*, *B26*, 211-234.

Box, G. E. P., & Draper, N. R. (1987). *Empirical model-building and response surfaces*. New York: Wiley.

Box, G. E. P., & Jenkins, G. M. (1970). *Time series analysis*. San Francisco: Holden Day.

Box, G. E. P., & Jenkins, G. M. (1976). *Time series analysis: Forecasting and control*. San Francisco: Holden-Day.

Box, G. E. P., & Tidwell, P. W. (1962). Transformation of the independent variables. *Technometrics*, *4*, 531-550.

Box, G. E. P., & Wilson, K. B. (1951). On the experimental attainment of optimum conditions. *Journal of the Royal Statistical Society,* Ser. B, *13,* 1-45.

Box, G. E. P., Hunter, W. G., & Hunter, S. J. (1978). *Statistics for experimenters: An introduction to design, data analysis, and model building.* New York: Wiley.

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and regression trees.* Monterey, CA: Wadsworth & Brooks/Cole Advanced Books & Software.

Brenner, J. L., et al. (1968). Difference equations in forecasting formulas. *Management Science*, *14,* 141-159.

Brent, R. F. (1973). *Algorithms for minimization without derivatives.* Englewood Cliffs, NJ: Prentice-Hall.

Breslow, N. E. (1970). A generalized Kruskal-Wallis test for comparing *K* samples subject to unequal pattern of censorship. *Biometrika*, *57,* 579-594.

Breslow, N. E. (1974). Covariance analysis of censored survival data. *Biometrics*, *30*, 89-99.

Bridle, J.S. (1990). Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In F. Fogelman Soulie and J. Herault (Eds.), *Neurocomputing: Algorithms, Architectures and Applications*, 227-236. New York: Springer-Verlag.

Brigham, E. O. (1974). *The fast Fourier transform.* Englewood Cliffs, NJ: Prentice-Hall.

Brillinger, D. R. (1975). *Time series: Data analysis and theory.* New York: Holt, Rinehart. & Winston.

Broomhead, D.S. and Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems 2*, 321-355.

Brown, D. T. (1959). A note on approximations to discrete probability distributions. *Information and Control*, *2*, 386-392.

Brown, M. B., & Forsythe, A. B. (1974). Robust tests for the equality of variances. *Journal of the American Statistical Association*, *69*, 264-267.

Brown, R. G. (1959). *Statistical forecasting for inventory control*. New York: McGraw-Hill.

Browne, M. W. (1968). A comparison of factor analytic techniques. *Psychometrika*, *33,* 267-334.

Browne, M. W. (1974). Generalized least squares estimators in the analysis of covariance structures. *South African Statistical Journal, 8*, 1-24.

Browne, M. W. (1982). Covariance Structures. In D. M. Hawkins (Ed.) *Topics in Applied Multivariate Analysis.* Cambridge, MA: Cambridge University Press.

Browne, M. W. (1984). Asymptotically distribution free methods for the analysis of covariance structures. *British Journal of Mathematical and Statistical Psychology, 37,* 62-83.

Browne, M. W., & Cudeck, R. (1990). Single sample cross-validation indices for covariance structures. *Multivariate Behavioral Research, 24,* 445-455.

Browne, M. W., & Cudeck, R. (1992). Alternative ways of assessing model fit. In K. A. Bollen and J. S. Long (Eds.), *Testing structural equation models.* Beverly Hills, CA: Sage.

Browne, M. W., & DuToit, S. H. C. (1982). AUFIT (Version 1). A computer programme for the automated fitting of nonstandard models for means and covariances. Research Finding WS-27. Pretoria, South Africa: Human Sciences Research Council.

Browne, M. W., & DuToit, S. H. C. (1987). Automated fitting of nonstandard models. Report WS-39. Pretoria, South Africa: Human Sciences Research Council.

Browne, M. W., & DuToit, S. H. C. (1992). Automated fitting of nonstandard models. *Multivariate Behavioral Research, 27,* 269-300.

Browne, M. W., & Mels, G. (1992). *RAMONA User's Guide.* The Ohio State University: Department of Psychology.

Browne, M. W., & Shapiro, A. (1989). Invariance of covariance structures under groups of transformations. Research Report 89/4. Pretoria, South Africa: University of South Africa Department of Statistics.

Browne, M. W., & Shapiro, A. (1991). Invariance of covariance structures under groups of transformations. *Metrika, 38,* 335-345.

Browne, M.W., & Cudeck, R. (1993). Alternative ways of assessing model fit. In K. A. Bollen & J. S. Long, (Eds.), *Testing structural equation models.* Beverly Hills, CA: Sage.

Brownlee, K. A. (1960). *Statistical Theory and Methodology in Science and Engineering*. New York: John Wiley.

Buffa, E. S. (1972). *Operations management: Problems and models* (3rd. ed.). New York: Wiley.

Buja, A., & Tukey, P. A. (Eds.) (1991). *Computing and Graphics in Statistics*. New York: Springer-Verlag.

Buja, A., Fowlkes, E. B., Keramidas, E. M., Kettenring, J. R., Lee, J. C., Swayne, D. F., & Tukey, P. A. (1986). Discovering features of multivariate data through statistical graphics. *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 98-103.

Burman, J. P. (1979). Seasonal adjustment - a survey. *Forecasting, Studies in Management Science*, *12*, 45-57.

Burns, L. S., & Harman, A. J. (1966). *The complex metropolis, Part V of profile of the Los Angeles metropolis: Its people and its homes*. Los Angeles: University of Chicago Press.

Burt, C. (1950). The factorial analysis of qualitative data. *British Journal of Psychology*, *3*, 166-185.

Campbell D. T., & Fiske, D. W. (1959). Convergent and discriminant validation by the multitrait-multimethod matrix. *Psychological Bulletin*, *56*, 81-105

Carling, A. (1992). *Introducing Neural Networks*. Wilmslow, UK: Sigma Press.

Carmines, E. G., & Zeller, R. A. (1980). *Reliability and validity assessment*. Beverly Hills, CA: Sage Publications.

Carrol, J. D., Green, P. E., and Schaffer, C. M. (1986). Interpoint distance comparisons in correspondence analysis. *Journal of Marketing Research*, *23*, 271-280.

Carroll, J. D., & Wish, M. (1974). Multidimensional perceptual models and measurement methods. In E. C. Carterette and M. P. Friedman (Eds.), *Handbook of perception*. (Vol. 2, pp. 391-447). New York: Academic Press.

Cattell, R. B. (1966). The scree test for the number of factors. *Multivariate Behavioral Research*, *1*, 245-276.

Cattell, R. B., & Jaspers, J. A. (1967). A general plasmode for factor analytic exercises and research. *Multivariate Behavioral Research Monographs.*

Chambers, J. M., Cleveland, W. S., Kleiner, B., & Tukey, P. A. (1983). *Graphical methods for data analysis*. Bellmont, CA: Wadsworth.

Chan, L. K., Cheng, S. W., & Spiring, F. (1988). A new measure of process capability: Cpm. *Journal of Quality Technology*, *20*, 162-175.

Chen, J. (1992). Some results on 2(nk) fractional factorial designs and search for minimum aberration designs. *Annals of Statistics*, *20*, 2124-2141.

Chen, J., & Wu, C. F. J. (1991). Some results on s(nk) fractional factorial designs with minimum aberration or optimal moments. *Annals of Statistics*, *19*, 1028-1041.

Chen, J., Sun, D. X., & Wu, C. F. J. (1993). A catalog of two-level and three-level fractional factorial designs with small runs. *International Statistical Review, 61,* 131-145.

Chernoff, H. (1973). The use of faces to represent points in k-dimensional space graphically. *Journal of American Statistical Association*, *68*, 361-368.

Christ, C. (1966). *Econometric models and methods*. New York: Wiley.

Clarke, G. M., & Cooke, D. (1978). *A basic course in statistics*. London: Edward Arnold.

Clements, J. A. (1989). Process capability calculations for non-normal distributions. *Quality Progress*. September, 95-100.

Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association, 74*, 829-836.

Cleveland, W. S. (1984). Graphs in scientific publications. *The American Statistician, 38*, 270-280.

Cleveland, W. S. (1985). *The elements of graphing data*. Monterey, CA: Wadsworth.

Cleveland, W. S. (1993). *Visualizing data*. Murray Hill, NJ: AT&T.

Cleveland, W. S., Harris, C. S., & McGill, R. (1982). Judgements of circle sizes on statistical maps. *Journal of the American Statistical Association*, *77*, 541-547.

Cliff, N. (1983). Some cautions concerning the application of causal modeling methods. *Multivariate Behavioral Research*, *18*, 115-126.

Cochran, W. G. (1950). The comparison of percentages in matched samples. *Biometrika*, *37*, 256-266.

Cohen, J. (1977). *Statistical power analysis for the behavioral sciences. (*Rev. ed.). New York: Academic Press.

Cohen, J. (1983). *Statistical power analysis for the behavioral sciences. (2nd Ed.).* Mahwah, NJ: Lawrence Erlbaum Associates.

Cohen, J. (1994). The earth is round (*p* < .05). *American Psychologist, 49,* 9971003.

Cole, D. A., Maxwell, S. E., Arvey, R., & Salas, E. (1993). Multivariate group comparisons of variable systems: MANOVA and structural equation modeling. *Psychological Bulletin*, *114*, 174-184.

Connor, W. S., & Young, S. (1984). Fractional factorial experiment designs for experiments with factors at two and three levels. In R. A. McLean & V. L. Anderson (Eds.), *Applied factorial and fractional designs*. New York: Marcel Dekker.

Connor, W. S., & Zelen, M. (1984). Fractional factorial experiment designs for factors at three levels. In R. A. McLean & V. L. Anderson (Eds.), *Applied factorial and fractional designs*. New York: Marcel Dekker.

Conover, W. J. (1974). Some reasons for not using the Yates continuity correction on 2 x 2 contingency tables. *Journal of the American Statistical Association*, *69*, 374-376.

Conover, W. J., Johnson, M. E., & Johnson, M. M. (1981). A comparative study of tests for homogeneity of variances with applications to the outer continental shelf bidding data. *Technometrics*, *23*, 357-361.

Cook, R. D. (1977). Detection of influential observations in linear regression. *Technometrics*, *19*, 15-18.

Cook, R. D., & Nachtsheim, C. J. (1980). A comparison of algorithms for constructing exact D-optimal designs. *Technometrics*, *22*, 315-324.

Cook, R. D., & Weisberg, S. (1982). *Residuals and Influence in Regression*. (Monographs on statistics and applied probability). New York: Chapman and Hall.

Cooke, D., Craven, A. H., & Clarke, G. M. (1982). *Basic statistical computing*. London: Edward Arnold.

Cooley, J. W., & Tukey, J. W. (1965). An algorithm for the machine computation of complex Fourier series. *Mathematics of Computation*, *19*, 297-301.

Cooley, W. W., & Lohnes, P. R. (1971). *Multivariate data analysis.* New York: Wiley.

Cooley, W. W., & Lohnes, P. R. (1976). *Evaluation research in education*. New York: Wiley.

Coombs, C. H. (1950). Psychological scaling without a unit of measurement. *Psychological Review*, *57*, 145-158.

Coombs, C. H. (1964). *A theory of data.* New York: Wiley.

Corballis, M. C., & Traub, R. E. (1970). Longitudinal factor analysis. *Psychometrika*, *35*, 79-98.

Corbeil, R. R., & Searle, S. R. (1976). Restricted maximum likelihood (REML) estimation of variance components in the mixed model. *Technometrics*, *18*, 31-38.

Cormack, R. M. (1971). A review of classification. *Journal of the Royal Statistical Society*, *134*, 321-367.

Cornell, J. A. (1990a). *How to run mixture experiments for product quality*. Milwaukee, Wisconsin: ASQC.

Cornell, J. A. (1990b). *Experiments with mixtures: designs, models, and the analysis of mixture data* (2nd ed.). New York: Wiley.

Cox, D. R. (1957). Note on grouping. *Journal of the American Statistical Association*, *52*, 543-547.

Cox, D. R. (1959). The analysis of exponentially distributed life-times with two types of failures. *Journal of the Royal Statistical Society*, *21*, 411-421.

Cox, D. R. (1964). Some applications of exponential ordered scores. *Journal of the Royal Statistical Society*, *26*, 103-110.

Cox, D. R. (1970). *The analysis of binary data*. New York: Halsted Press.

Cox, D. R. (1972). Regression models and life tables. *Journal of the Royal Statistical Society*, *34*, 187-220.

Cox, D. R., & Oakes, D. (1984). *Analysis of survival data*. New York: Chapman & Hall.

Cramer, H. (1946). *Mathematical methods in statistics*. Princeton, NJ: Princeton University Press.

Cristianini, N., & Shawe-Taylor, J. (2000). *Introduction to support vector machines and other kernel-based learning methods*. Cambridge, UK: Cambridge University Press.

Crowley, J., & Hu, M. (1977). Covariance analysis of heart transplant survival data. *Journal of the American Statistical Association*, *72*, 27-36.

Cudeck, R. (1989). Analysis of correlation matrices using covariance structure models. *Psychological Bulletin*, *105*, 317-327.

Cudeck, R., & Browne, M. W. (1983). Cross-validation of covariance structures. *Multivariate Behavioral Research*, *18*, 147-167.

Cutler, S. J., & Ederer, F. (1958). Maximum utilization of the life table method in analyzing survival. *Journal of Chronic Diseases*, *8*, 699-712.

Dahlquist, G., & Bjorck, A. (1974). *Numerical Methods.* Englewood Cliffs, NJ: Prentice-Hall.

Daniel, C. (1976). *Applications of statistics to industrial experimentation.* New York: Wiley.

Daniell, P. J. (1946). Discussion on symposium on autocorrelation in time series. *Journal of the Royal Statistical Society,* Suppl. *8*, 88-90.

Daniels, H. E. (1939). The estimation of components of variance. *Supplement to the Journal of the Royal Statistical Society*, *6*, 186-197.

Darlington, R. B. (1990). *Regression and linear models*. New York: McGraw-Hill.

Darlington, R. B., Weinberg, S., & Walberg, H. (1973). Canonical variate analysis and related techniques. *Review of Educational Research*, *43*, 433-454.

DataMyte (1992). *DataMyte handbook.* Minnetonka, MN.

David, H. A. (1995). First (?) occurrence of common terms in mathematical statistics. *The American Statistician*, *49*, 121-133.

Davies, P. M., & Coxon, A. P. M. (1982). *Key texts in multidimensional scaling.* Exeter, NH: Heinemann Educational Books.

Davis, C. S., & Stephens, M. A. Approximate percentage points using Pearson curves. *Applied Statistics, 32*, 322-327.

De Boor, C. (1978). *A practical guide to splines*. New York: Springer-Verlag.

DeCarlo, L. T. (1998). Signal detection theory and generalized linear models, *Psychological Methods*, 186-200.

De Gruitjer, P. N. M., & Van Der Kamp, L. J. T. (Eds.). (1976). *Advances in psychological and educational measurement*. New York: Wiley.

de Jong, S (1993) SIMPLS: An Alternative Approach to Partial Least Squares Regression, *Chemometrics and Intelligent Laboratory Systems*, 18, 251-263

de Jong, S and Kiers, H. (1992) Principal Covariates regression, *Chemometrics and Intelligent Laboratory Systems*, 14, 155-164

Deming, S. N., & Morgan, S. L. (1993). *Experimental design: A chemometric approach.* (2nd ed.). Amsterdam, The Netherlands: Elsevier Science Publishers B.V.

Deming, W. E., & Stephan, F. F. (1940). The sampling procedure of the 1940 population census. *Journal of the American Statistical Association*, *35*, 615-630.

Dempster, A. P. (1969). *Elements of Continuous Multivariate Analysis*. San Francisco: Addison-Wesley.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, 39,* 1-38.

Dennis, J. E., & Schnabel, R. B. (1983). *Numerical methods for unconstrained optimization and nonlinear equations.* Englewood Cliffs, NJ: Prentice Hall.

Derringer, G., & Suich, R. (1980). Simultaneous optimization of several response variables. *Journal of Quality Technology*, *12*, 214-219.

Diamond, W. J. (1981). *Practical experimental design.* Belmont, CA: Wadsworth.

Dijkstra, T. K. (1990). Some properties of estimated scale invariant covariance structures. *Psychometrika*, *55*, 327-336.

Dinneen, L. C., & Blakesley, B. C. (1973). A generator for the sampling distribution of the Mann Whitney *U* statistic. *Applied Statistics*, *22*, 269-273.

Dixon, W. J. (1954). Power under normality of several non-parametric tests. *Annals of Mathematical Statistics*, *25*, 610-614.

Dixon, W. J., & Massey, F. J. (1983). *Introduction to statistical analysis* (4th ed.). New York: McGraw-Hill.

Dobson, A. J. (1990). *An introduction to generalized linear models*. New York: Chapman & Hall.

Dodd, B. (1979). Lip reading in infants: Attention to speech presented in- and out-of-synchrony. *Cognitive Psychology*, *11*, 478-484.

Dodge, Y. (1985). *Analysis of experiments with missing data*. New York: Wiley.

Dodge, Y., Fedorov, V. V., & Wynn, H. P. (1988). *Optimal design and analysis of experiments.* New York: North-Holland.

Dodson, B. (1994). *Weibull analysis*. Milwaukee, Wisconsin: ASQC.

Doyle, P. (1973). The use of Automatic Interaction Detection and similar search procedures. *Operational Research Quarterly*, *24*, 465-467.

Duncan, A. J. (1974). *Quality control and industrial statistics*. Homewood, IL: Richard D. Irwin.

Duncan, O. D., Haller, A. O., & Portes, A. (1968). Peer influence on aspiration: a reinterpretation. *American Journal of Sociology*, *74,* 119-137.

Dunnett, C. W. (1955). A multiple comparison procedure for comparing several treatments with a control. *Journal of the American Statistical Association, 50,* 1096-1121.

Durbin, J. (1970). Testing for serial correlation in least-squares regression when some of the regressors are lagged dependent variables. *Econometrica*, *38,* 410-421.

Durbin, J., & Watson, G. S. (1951). Testing for serial correlations in least squares regression. II. *Biometrika*, *38,* 159-178.

Dykstra, O. Jr. (1971). The augmentation of experimental data to maximize |X'X|. *Technometrics, 13,* 682-688.

Eason, E. D., & Fenton, R. G. (1974). A comparison of numerical optimization methods for engineering design. *ASME Paper 73-DET-17.*

Edelstein, H., A. (1999). *Introduction to data mining and knowledge discovery (3rd ed).* Potomac, MD: Two Crows Corp.

Edgeworth, F. Y. (1885). Methods of statistics. In *Jubilee Volume, Royal Statistical Society*, 181-217.

Efron, B. (1982). *The jackknife, the bootstrap, and other resampling plans.* Philadelphia, Pa. Society for Industrial and Applied Mathematics.

Eisenhart, C. (1947). The assumptions underlying the analysis of variance. *Biometrics*, *3,* 1-21.

Elandt-Johnson, R. C., & Johnson, N. L. (1980). *Survival models and data analysis*. New York: Wiley.

Elliott, D. F., & Rao, K. R. (1982). *Fast transforms: Algorithms, analyses, applications.* New York: Academic Press.

Elsner, J. B., Lehmiller, G. S., & Kimberlain, T. B. (1996). Objective classification of Atlantic hurricanes. *Journal of Climate*, *9,* 2880-2889.

Enslein, K., Ralston, A., & Wilf, H. S. (1977). *Statistical methods for digital computers.* New York: Wiley.

Euler, L. (1782). Recherches sur une nouvelle espece de quarres magiques. *Verhandelingen uitgegeven door het zeeuwsch Genootschap der Wetenschappen te Vlissingen*, *9*, 85-239. (Reproduced in *Leonhardi Euleri Opera Omnia*. Sub auspiciis societatis scientiarium naturalium helveticae, 1st series, *7*, 291-392.)

Evans, M., Hastings, N., & Peacock, B. (1993). *Statistical Distributions*. New York: Wiley.

Everitt, B. S. (1977). *The analysis of contingency tables.* London: Chapman & Hall.

Everitt, B. S. (1984). *An introduction to latent variable models*. London: Chapman and Hall.

Ewan, W. D. (1963). When and how to use Cu-sum charts. *Technometrics*, *5*, 1-32.

Fahlman, S.E. (1988). Faster-learning variations on back-propagation: an empirical study. In D. Touretzky, G.E. Hinton and T.J. Sejnowski (Eds.), *Proceedings of the 1988 Connectionist Models Summer School*, 38-51. San Mateo, CA: Morgan Kaufmann.

Fausett, L. (1994). *Fundamentals of Neural Networks.* New York: Prentice Hall.

Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., & Uthurusamy, R. (Eds.). 1996. *Advances in Knowledge Discovery and Data Mining*. Cambridge, MA: The MIT Press.

Fayyad, U. S., & Uthurusamy, R. (Eds.) (1994). *Knowledge Discovery in Databases; Papers from the 1994 AAAI Workshop*. Menlo Park, CA: AAAI Press.

Feigl, P., & Zelen, M. (1965). Estimation of exponential survival probabilities with concomitant information. *Biometrics*, *21*, 826-838.

Feller, W. (1948). On the Kolmogorov-Smirnov limit theorems for empirical distributions. *Annals of Mathematical Statistics*, *19*, 177-189.

Fetter, R. B. (1967). *The quality control system*. Homewood, IL: Richard D. Irwin.

Fienberg, S. E. (1977). *The analysis of cross-classified categorical data*. Cambridge, MA: MIT Press.

Finn, J. D. (1974). *A general model for multivariate analysis.* New York: Holt, Rinehart & Winston.

Finn, J. D. (1977). Multivariate analysis of variance and covariance. In K. Enslein, A. Ralston, and H. S. Wilf (Eds.), *Statistical methods for digital computers. Vol. III*. (pp. 203-264). New York: Wiley.

Finney, D. J. (1944). The application of probit analysis to the results of mental tests. *Psychometrika*, *9*, 31-39.

Finney, D. J. (1971). *Probit analysis*. Cambridge, MA: Cambridge University Press.

Firmin, R. (2002). Advanced time series modeling for semiconductor process control: The fab as a time machine. In Mackulak, G. T., Fowler, J. W., & Schomig, A. (eds.). *Proceedings of the International Conference on Modeling and Analysis of Semiconductor Manufacturing (MASM 2002).*

Fisher, R. A. (1918). The correlation between relatives on the supposition of Mendelian inheritance. *Transactions of the Royal Society of Edinbrugh, 52*, 399-433.

Fisher, R. A. (1922). On the interpretation of *Chi-square* from contingency tables, and the calculation of *p*. *Journal of the Royal Statistical Society*, *85*, 87-94.

Fisher, R. A. (1922). On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London,* Ser. A, *222,* 309-368.

Fisher, R. A. (1926). The arrangement of field experiments. *Journal of the Ministry of Agriculture of Great Britain*, *33*, 503-513.

Fisher, R. A. (1928). The general sampling distribution of the multiple correlation coefficient. *Proceedings of the Royal Society of London*, Ser. A, *121*, 654-673.

Fisher, R. A. (1935). *The Design of Experiments*. Edinburgh: Oliver and Boyd.

Fisher, R. A. (1936). *Statistical Methods for Research Workers (6th ed.)*. Edinburgh: Oliver and Boyd.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, *7*, 179-188.

Fisher, R. A. (1938). The mathematics of experimentation. *Nature*, *142*, 442-443.

Fisher, R. A., & Yates, F. (1934). The 6 x 6 Latin squares. *Proceedings of the Cambridge Philosophical Society*, *30*, 492-507.

Fisher, R. A., & Yates, F. (1938). *Statistical Tables for Biological, Agricultural and Medical Research*. London: Oliver and Boyd.

Fleishman, A. E. (1980). Confidence intervals for correlation ratios. *Educational and Psychological Measurement, 40,* 659670.

Fletcher, R. (1969). *Optimization*. New York: Academic Press.

Fletcher, R., & Powell, M. J. D. (1963). A rapidly convergent descent method for minimization. *Computer Journal, 6,* 163-168.

Fletcher, R., & Reeves, C. M. (1964). Function minimization by conjugate gradients. *Computer Journal*, *7*, 149-154.

Fomby, T.B., Hill, R.C., & Johnson, S.R. (1984). *Advanced econometric methods*. New York: Springer-Verlag.

Ford Motor Company, Ltd. & GEDAS (1991). *Test examples for SPC software*.

Fouladi, R. T. (1991). *A comprehensive examination of procedures for testing the significance of a correlation matrix and its elements.* Unpublished master's thesis, University of British Columbia, Vancouver, British Columbia, Canada.

Franklin, M. F. (1984). Constructing tables of minimum aberration p(nm) designs. *Technometrics, 26*, 225-232.

Fraser, C., & McDonald, R. P. (1988). COSAN: Covariance structure analysis. *Multivariate Behavioral Research, 23,* 263-265.

Freedman, L. S. (1982). Tables of the number of patients required in clinical trials using the logrank test. *Statistics in Medicine*, *1,* 121129.

Friedman, J. (1991). Multivariate adaptive regression splines (with discussion), *Annals of Statistics*, *19*, 1-141.

Friedman, J. H. (1993). Estimating functions of mixed ordinal and categorical variables using adaptive splines. in S. Morgenthaler, E. Ronchetti, & W. A. Stahel (Eds.) (1993, p. 73-113). *New directions in statistical data analysis and robustness.* Berlin: Birkhäuser Verlag.

Friedman, J. H. (1999a). Greedy function approximation: A gradient boosting machine. IMS 1999 Reitz Lecture.

Friedman, J. H. (1999b). Stochastic gradient boosting. Stanford University.

Friedman, M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association, 32,* 675-701.

Friedman, M. (1940). A comparison of alternative tests of significance for the problem of *m* rankings. *Annals of Mathematical Statistics, 11*, 86-92.

Fries, A., & Hunter, W. G. (1980). Minimum aberration 2 (kp) designs. *Technometrics, 22*, 601-608.

Frost, P. A. (1975). Some properties of the Almon lag technique when one searches for degree of polynomial and lag. *Journal of the American Statistical Association, 70*, 606-612.

Fuller, W. A. (1976). *Introduction to statistical time series*. New York: Wiley.

Gaddum, J. H. (1945). Lognormal distributions. *Nature*, *156*, 463-466.

Gale, N., & Halperin, W. C. (1982). A case for better graphics: The unclassed choropleth map. *The American Statistician, 36*, 330-336.

Galil, Z., & Kiefer, J. (1980). Time- and space-saving computer methods, related to Mitchell's DETMAX, for finding D-optimum designs. *Technometrics, 22,* 301-313.

Galton, F. (1882). Report of the anthropometric committee. In *Report of the 51st Meeting of the British Association for the Advancement of Science, 1881*, 245-260.

Galton, F. (1885). Section H. Anthropology. Opening address by Francis Galton. *Nature, 32*, 507- 510.

Galton, F. (1885). Some results of the anthropometric laboratory. *Journal of the Anthropological Institute, 14*, 275-287.

Galton, F. (1888). Co-relations and their measurement. *Proceedings of the Royal Society of London*, *45*, 135-145.

Galton, F. (1889). *Natural Inheritance*. London: Macmillan.

Galton, F. (1889). *Natural Inheritance*. London: Macmillan.

Ganguli, M. (1941). A note on nested sampling. *Sankhya*, *5*, 449-452.

Gara, M. A., & Rosenberg, S. (1979). The identification of persons as supersets and subsets in free-response personality descriptions. *Journal of Personality and Social Psychology, 37*, 2161-2170.

Gara, M. A., & Rosenberg, S. (1981). Linguistic factors in implicit personality theory. *Journal of Personality and Social Psychology*, *41,* 450-457.

Gardner, E. S., Jr. (1985). Exponential smoothing: The state of the art. *Journal of Forecasting*, *4,* 1-28.

Garthwaite, P. H. (1994) An Interpretation of Partial Least Squares, *Journal of the American Statistical Association*, 89 NO. 425, 122-127.

Garvin, D. A. (1987). Competing on the eight dimensions of quality. *Harvard Business Review*, November/December, 101-109.

Gatsonis, C., & Sampson, A. R. (1989). Multiple correlation: exact power and sample size calculations. *Psychological Bulletin, 106*, 516524.

Gbur, E., Lynch, M., & Weidman, L. (1986). An analysis of nine rating criteria on 329 U. S. metropolitan areas. *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 104-109.

Gedye, R. (1968). *A manager's guide to quality and reliability*. New York: Wiley.

Gehan, E. A. (1965a). A generalized Wilcoxon test for comparing arbitrarily singly-censored samples. *Biometrika*, *52,* 203-223.

Gehan, E. A. (1965b). A generalized two-sample Wilcoxon test for doubly-censored data. *Biometrika, 52*, 650-653.

Gehan, E. A., & Siddiqui, M. M. (1973). Simple regression methods for survival time studies. *Journal of the American Statistical Association, 68*, 848-856.

Gehan, E. A., & Thomas, D. G. (1969). The performance of some two sample tests in small samples with and without censoring. *Biometrika*, *56*, 127-132.

Geladi, P. and Kowalski, B. R. (1986) Partial Least Squares Regression: A Tutorial, *Analytica Chimica Acta*, 185, 1-17.

Gerald, C. F., & Wheatley, P. O. (1989). *Applied numerical analysis* (4th ed.). Reading, MA: Addison Wesley.

Gibbons, J. D. (1976). *Nonparametric methods for quantitative analysis.* New York: Holt, Rinehart, & Winston.

Gibbons, J. D. (1985). *Nonparametric statistical inference* (2nd ed.). New York: Marcel Dekker.

Gifi, A. (1981). *Nonlinear multivariate analysis.* Department of Data Theory, The University of Leiden. The Netherlands.

Gifi, A. (1990). *Nonlinear multivariate analysis.* New York: Wiley.

Gill, P. E., & Murray, W. (1972). Quasi-Newton methods for unconstrained optimization. *Journal of the Institute of Mathematics and its Applications, 9*, 91-108.

Gill, P. E., & Murray, W. (1974). *Numerical methods for constrained optimization.* New York: Academic Press.

Gini, C. (1911). Considerazioni sulle probabilita a posteriori e applicazioni al rapporto dei sessi nelle nascite umane. *Studi Economico-Giuridici della Universita de Cagliari*, Anno III, 133-171.

Glass, G V., & Hopkins, K. D. (1996). *Statistical methods in education and psychology*. Needham Heights, MA: Allyn & Bacon.

Glass, G. V., & Stanley, J. (1970). *Statistical methods in education and Psychology*. Englewood Cliffs, NJ: Prentice-Hall.

Glasser, M. (1967). Exponential survival with covariance. *Journal of the American Statistical Association*, *62*, 561-568.

Gnanadesikan, R., Roy, S., & Srivastava, J. (1971). *Analysis and design of certain quantitative multiresponse experiments*. Oxford: Pergamon Press, Ltd.

Goldberg, D. E. (1989). *Genetic Algorithms.* Reading, MA: Addison Wesley.

Golub, G. and Kahan, W. (1965). Calculating the singular values and pseudo-inverse of a matrix. *SIAM Numerical Analysis, B 2 (2)*, 205-224.

Golub, G. H. and van Load, C. F. (1996) *Matrix Computations*, The Johns Hopkins University Press

Golub, G. H., & Van Loan, C. F. (1983). *Matrix computations.* Baltimore: Johns Hopkins University Press.

Gompertz, B. (1825). On the nature of the function expressive of the law of human mortality. *Philosophical Transactions of the Royal Society of London,* Series A, 115, 513-580.

Goodman, L .A., & Kruskal, W. H. (1972). Measures of association for cross-classifications IV: Simplification of asymptotic variances. *Journal of the American Statistical Association*, *67*, 415-421.

Goodman, L. A. (1954). Kolmogorov-Smirnov tests for psychological research. *Psychological Bulletin*, *51*, 160-168.

Goodman, L. A. (1971). The analysis of multidimensional contingency tables: Stepwise procedures and direct estimation methods for models building for multiple classification. *Technometrics, 13,* 33-61.

Goodman, L. A., & Kruskal, W. H. (1954). Measures of association for cross-classifications. *Journal of the American Statistical Association*, *49*, 732-764.

Goodman, L. A., & Kruskal, W. H. (1959). Measures of association for cross-classifications II: Further discussion and references. *Journal of the American Statistical Association*, *54*, 123-163.

Goodman, L. A., & Kruskal, W. H. (1963). Measures of association for cross-classifications III: Approximate sampling theory. *Journal of the American Statistical Association*, *58*, 310-364.

Goodnight, J. H. (1980). Tests of hypotheses in fixed effects linear models. *Communications in Statistics*, *A9*, 167-180.

Gorman, R.P., & Sejnowski, T.J. (1988). Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks 1 (1)*, 75-89.

Grant, E. L., & Leavenworth, R. S. (1980). *Statistical quality control* (5th ed.). New York: McGraw-Hill.

Green, P. E., & Carmone, F. J. (1970). *Multidimensional scaling and related techniques in marketing analysis*. Boston: Allyn & Bacon.

Green, P. J. & Silverman, B. W. (1994) *Nonparametric regression and generalized linear models: A roughness penalty approach*. New York: Chapman & Hall.

Greenacre, M. J. & Hastie, T. (1987). The geometric interpretation of correspondence analysis. *Journal of the American Statistical Association*, *82*, 437-447.

Greenacre, M. J. (1984). *Theory and applications of correspondence analysis*. New York: Academic Press.

Greenacre, M. J. (1988). Correspondence analysis of multivariate categorical data by weighted least-squares. *Biometrica*, *75*, 457-467.

Greenhouse, S. W., & Geisser, S. (1958). Extension of Box's results on the use of the *F* distribution in multivariate analysis. *Annals of Mathematical Statistics*, *29*, 95-112.

Greenhouse, S. W., & Geisser, S. (1959). On methods in the analysis of profile data. *Psychometrika*, *24*, 95-112.

Grizzle, J. E. (1965). The two-period change-over design and its use in clinical trials. *Biometrics*, *21*, 467-480.

Gross, A. J., & Clark, V. A. (1975). *Survival distributions: Reliability applications in the medical sciences*. New York: Wiley.

Gruska, G. F., Mirkhani, K., & Lamberson, L. R. (1989). *Non-Normal data Analysis.* Garden City, MI: Multiface Publishing.

Gruvaeus, G., & Wainer, H. (1972). Two additions to hierarchical cluster analysis. *The British Journal of Mathematical and Statistical Psychology, 25*, 200-206.

Guttman, L. (1954). A new approach to factor analysis: the radex. In P. F. Lazarsfeld (Ed.), *Mathematical thinking in the social sciences.* New York: Columbia University Press.

Guttman, L. (1968). A general nonmetric technique for finding the smallest coordinate space for a configuration of points. *Pyrometrical*, *33*, 469-506.

Guttman, L. B. (1977). What is not what in statistics. *The Statistician*, *26*, 81107.

Haberman, S. J. (1972). Loglinear fit for contingency tables. *Applied Statistics*, *21*, 218-225.

Haberman, S. J. (1974). *The analysis of frequency data*. Chicago: University of Chicago Press.

Hahn, G. J., & Shapiro, S. S. (1967). *Statistical models in engineering*. New York: Wiley.

Hakstian, A. R., Rogers, W. D., & Cattell, R. B. (1982). The behavior of numbers of factors rules with simulated data. *Multivariate Behavioral Research*, *17*, 193-219.

Hald, A. (1949). Maximum likelihood estimation of the parameters of a normal distribution which is truncated at a known point. *Skandinavisk Aktuarietidskrift*, *1949*, 119-134.

Hald, A. (1952). *Statistical theory with engineering applications.* New York: Wiley.

Han, J., Kamber, M. (2000). *Data mining: Concepts and Techniques*. New York: Morgan-Kaufman.

Han, J., Lakshmanan, L. V. S., & Pei, J. (2001). Scalable frequent-pattern mining methods: An overview. In T. Fawcett (Ed.). *KDD 2001: Tutorial Notes of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York: The Association for Computing Machinery.

Harlow, L. L., Mulaik, S. A., & Steiger, J. H. (Eds.) (1997). *What if there were no significance tests.* Mahwah, NJ: Lawrence Erlbaum Associates.

Harman, H. H. (1967). *Modern factor analysis.* Chicago: University of Chicago Press.

Harris, R. J. (1976). The invalidity of partitioned *U* tests in canonical correlation and multivariate analysis of variance. *Multivariate Behavioral Research*, *11*, 353-365.

Harrison, D. & Rubinfield, D. L. (1978). Hedonic prices and the demand for clean air. *Journal of Environmental Economics and Management*, *5*, 81-102.

Hart, K. M., & Hart, R. F. (1989). *Quantitative methods for quality improvement.* Milwaukee, WI: ASQC Quality Press.

Hartigan, J. A. & Wong, M. A. (1978). Algorithm 136. A *k*-means clustering algorithm. *Applied Statistics*, *28*, 100.

Hartigan, J. A. (1975). *Clustering algorithms.* New York: Wiley.

Hartley, H. O. (1959). Smallest composite designs for quadratic response surfaces. *Biometrics*, *15*, 611-624.

Harville, D. A. (1977). Maximum likelihood approaches to variance component estimation and to related problems. *Journal of the American Statistical Association, 72*, 320-340.

Haskell, A. C. (1922). *Graphic Charts in Business*. New York: Codex.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2001). *The elements of statistical learning : Data mining, inference, and prediction*. New York: Springer.

Haviland, R. P. (1964). *Engineering reliability and long life design*. Princeton, NJ: Van Nostrand.

Hayduk, L. A. (1987). *Structural equation modelling with LISREL: Essentials and advances.* Baltimore: The Johns Hopkins University Press.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation.* New York: Macmillan Publishing.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation.* New York: Macmillan College Publishing.

Hays, W. L. (1981). *Statistics* (3rd ed.). New York: CBS College Publishing.

Hays, W. L. (1988). *Statistics* (4th ed.). New York: CBS College Publishing.

Heiberger, R. M. (1989). *Computation for the analysis of designed experiments*. New York: Wiley.

Hemmerle, W. J., & Hartley, H., O. (1973). Computing maximum likelihood estimates for the mixed A.O.V. model using the W transformation. *Technometrics*, *15,* 819-831.

Henley, E. J., & Kumamoto, H. (1980). *Reliability engineering and risk assessment*. New York: Prentice-Hall.

Hettmansperger, T. P. (1984). *Statistical inference based on ranks*. New York: Wiley.

Hibbs, D. (1974). Problems of statistical estimation and causal inference in dynamic time series models. In H. Costner (Ed.), *Sociological Methodology 1973/1974* (pp. 252-308). San Francisco: Jossey-Bass.

Hill, I. D., Hill, R., & Holder, R. L. (1976). Fitting Johnson curves by moments. *Applied Statistics. 25*, 190-192.

Hilton, T. L. (1969). *Growth study annotated bibliography. Princeton, NJ: Educational Testing Service Progress Report 69-11.*

Hochberg, J., & Krantz, D. H. (1986). Perceptual properties of statistical graphs. *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 29-35.

Hocking, R. R. (1985). *The analysis of linear models*. Monterey, CA: Brooks/Cole.

Hocking, R. R. (1996). *Methods and Applications of Linear Models. Regression and the Analysis of Variance*. New York: Wiley.

Hocking, R. R., & Speed, F. M. (1975). A full rank analysis of some linear model problems. *Journal of the American Statistical Association*, *70*, 707-712.

Hoerl, A. E. (1962). Application of ridge analysis to regression problems. *Chemical Engineering Progress*, *58*, 54-59.

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Applications to nonorthogonal problems. *Technometrics*, *12*, 69-82.

Hoff, J. C. (1983). *A practical guide to Box-Jenkins forecasting*. London: Lifetime Learning Publications.

Hoffman, D. L. & Franke, G. R. (1986). Correspondence analysis: Graphical representation of categorical data in marketing research. *Journal of Marketing Research, 13*, 213-227.

Hogg, R. V., & Craig, A. T. (1970). *Introduction to mathematical statistics*. New York: Macmillan.

Holzinger, K. J., & Swineford, F. (1939). *A study in factor analysis: The stability of a bi-factor solution.* University of Chicago: Supplementary Educational Monographs, No. 48.

Hooke, R., & Jeeves, T. A. (1961). Direct search solution of numerical and statistical problems. *Journal of the Association for Computing Machinery*, *8,* 212-229.

Hosmer, D. W and Lemeshow, S. (1989), *Applied Logistic Regression*, John Wiley & Sons, Inc.

Hotelling, H. (1947). Multivariate quality control. In Eisenhart, Hastay, and Wallis (Eds.), *Techniques of Statistical Analysis*. New York: McGraw-Hill.

Hotelling, H., & Pabst, M. R. (1936). Rank correlation and tests of significance involving no assumption of normality. *Annals of Mathematical Statistics*, *7*, 29-43.

Hoyer, W., & Ellis, W. C. (1996). A graphical exploration of SPC. *Quality Progress, 29,* 65-73.

Hsu, P. L. (1938). Contributions to the theory of Student's *t* test as applied to the problem of two samples. *Statistical Research Memoirs, 2,* 1-24.

Huba, G. J., & Harlow, L. L. (1987). Robust structural equation models: implications for developmental psychology. *Child Development*, *58,* 147-166.

Huberty, C. J. (1975). Discriminant analysis. *Review of Educational Research*, *45*, 543-598.

Hunter, A., Kennedy, L., Henry, J., & Ferguson, R.I. (2000). Application of Neural Networks and Sensitivity Analysis to improved prediction of Trauma Survival. *Computer Methods and Algorithms in Biomedicine* 62, 11-19.

Huynh, H., & Feldt, L. S. (1970). Conditions under which mean square ratios in repeated measures designs have exact *F*-distributions. *Journal of the American Statistical Association*, *65,* 1582-1589.

Ireland, C. T., & Kullback, S. (1968). Contingency tables with given marginals. *Biometrika*, *55*, 179-188.

Jaccard, J., Weber, J., & Lundmark, J. (1975). A multitrait-multimethod factor analysis of four attitude assessment procedures. *Journal of Experimental Social Psychology*, *11*, 149-154.

Jacobs, D. A. H. (Ed.). (1977). *The state of the art in numerical analysis.* London: Academic Press.

Jacobs, R.A. (1988). Increased Rates of Convergence Through Learning Rate Adaptation. *Neural Networks 1 (4)*, 295-307.

Jacoby, S. L. S., Kowalik, J. S., & Pizzo, J. T. (1972). *Iterative methods for nonlinear optimization problems.* Englewood Cliffs, NJ: Prentice-Hall.

James, L. R., Mulaik, S. A., & Brett, J. M. (1982). *Causal analysis. Assumptions, models, and data.* Beverly Hills, CA: Sage Publications.

Jardine, N., & Sibson, R. (1971). *Mathematical taxonomy.* New York: Wiley.

Jastrow, J. (1892). On the judgment of angles and position of lines. *American Journal of Psychology*, *5*, 214-248.

Jenkins, G. M., & Watts, D. G. (1968). *Spectral analysis and its applications.* San Francisco: Holden-Day.

Jennrich, R. I. (1970). An asymptotic test for the equality of two correlation matrices. *Journal of the American Statistical Association*, *65*, 904-912.

Jennrich, R. I. (1977). Stepwise regression. In K. Enslein, A. Ralston, & H.S. Wilf (Eds.), *Statistical methods for digital computers*. New York: Wiley.

Jennrich, R. I., & Moore, R. H. (1975). Maximum likelihood estimation by means of nonlinear least squares. *Proceedings of the Statistical Computing Section*, American Statistical Association, 57-65.

Jennrich, R. I., & Sampson, P. F. (1968). Application of stepwise regression to non-linear estimation. *Technometrics*, *10*, 63-72.

Jennrich, R. I., & Sampson, P. F. (1976). Newton-Raphson and related algorithms for maximum likelihood variance component estimation. *Technometrics*, *18*, 11-17.

Jennrich, R. I., & Schuchter, M. D. (1986). Unbalanced repeated-measures models with structured covariance matrices. *Biometrics*, *42*, 805-820.

Jennrich. R. I. (1977). Stepwise discriminant analysis. In K. Enslein, A. Ralston, & H.S. Wilf (Eds.), *Statistical methods for digital computers*. New York: Wiley.

Johnson, L. W., & Ries, R. D. (1982). *Numerical Analysis* (2nd ed.). Reading, MA: Addison Wesley.

Johnson, N. L. (1961). A simple theoretical approach to cumulative sum control charts. *Journal of the American Statistical Association*, *56*, 83-92.

Johnson, N. L. (1965). Tables to facilitate fitting *SU* frequency curves. *Biometrika*, *52*, 547.

Johnson, N. L., & Kotz, S. (1970). *Continuous univariate distributions, Vol I and II.* New York: Wiley.

Johnson, N. L., Kotz, S., Blakrishnan, N. (1995). Continuous univariate distributions: Volumne II. (2nd Ed). NY: Wiley.

Johnson, N. L., & Leone, F. C. (1962). Cumulative sum control charts - mathematical principles applied to their construction and use. *Industrial Quality Control, 18*, 15-21.

Johnson, N. L., Nixon, E., & Amos, D. E. (1963). Table of percentage points of pearson curves. *Biometrika, 50*, 459.

Johnson, N. L., Nixon, E., Amos, D. E., & Pearson, E. S. (1963). Table of percentage points of Pearson curves for given 1 and 2, expressed in standard measure. *Biometrika*, 50, 459-498.

Johnson, P. (1987). *SPC for short runs: A programmed instruction workbook.* Southfield, MI: Perry Johnson.

Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika*, *32*, 241-254.

Johnston, J. (1972). *Econometric methods*. New York: McGraw-Hill.

Jöreskog, K. G. (1973). A general model for estimating a linear structural equation system. In A. S. Goldberger and O. D. Duncan (Eds.), *Structural Equation Models in the Social Sciences.* New York: Seminar Press.

Jöreskog, K. G. (1974). Analyzing psychological data by structural analysis of covariance matrices. In D. H. Krantz, R. C. Atkinson, R. D. Luce, and P. Suppes (Eds.), *Contemporary Developments in Mathematical Psychology, Vol. II.* New York: W. H. Freeman and Company.

Jöreskog, K. G. (1978). Structural analysis of covariance and correlation matrices. *Psychometrika*, *43*, 443-477.

Jöreskog, K. G., & Lawley, D. N. (1968). New methods in maximum likelihood factor analysis. *British Journal of Mathematical and Statistical Psychology, 21*, 85-96.

Jöreskog, K. G., & Sörbom, D. (1979). *Advances in factor analysis and structural equation models*. Cambridge, MA: Abt Books.

Jöreskog, K. G., & Sörbom, D. (1982). Recent developments in structural equation modeling. *Journal of Marketing Research, 19,* 404-416.

Jöreskog, K. G., & Sörbom, D. (1984). *Lisrel VI. Analysis of linear structural relationships by maximum likelihood, instrumental variables, and least squares methods*. Mooresville, Indiana: Scientific Software.

Jöreskog, K. G., & Sörbom, D. (1989). *Lisrel 7. A guide to the program and applications.* Chicago, Illinois: SPSS Inc.

Judge, G. G., Griffith, W. E., Hill, R. C., Luetkepohl, H., & Lee, T. S. (1985). *The theory and practice of econometrics*. New York: Wiley.

Juran, J. M. (1960). Pareto, Lorenz, Cournot, Bernnouli, Juran and others. *Industrial Quality Control*, *17*, 25.

Juran, J. M. (1962). *Quality control handbook*. New York: McGraw-Hill.

Juran, J. M., & Gryna, F. M. (1970). *Quality planning and analysis*. New York: McGraw-Hill.

Juran, J. M., & Gryna, F. M. (1980). *Quality planning and analysis* (2nd ed.). New York: McGraw-Hill.

Juran, J. M., & Gryna, F. M. (1988). *Juran's quality control handbook* (4th ed.). New York: McGraw-Hill.

Kachigan, S. K. (1986). *Statistical analysis: An interdisciplinary introduction to univariate & multivariate methods*. New York: Redius Press.

Kackar, R. M. (1985). Off-line quality control, parameter design, and the Taguchi method. *Journal of Quality Technology*, *17*, 176-188.

Kackar, R. M. (1986). Taguchi's quality philosophy: Analysis and commentary. *Quality Progress*, *19*, 21-29.

Kahneman, D., Slovic, P., & Tversky, A. (1982). *Judgment under uncertainty: Heuristics and biases*. New York: Cambridge University Press.

Kaiser, H. F. (1958). The varimax criterion for analytic rotation in factor analysis. *Pyrometrical*, *23*, 187-200.

Kaiser, H. F. (1960). The application of electronic computers to factor analysis. *Educational and Psychological Measurement*, *20*, 141-151.

Kalbfleisch, J. D., & Prentice, R. L. (1980). *The statistical analysis of failure time data*. New York: Wiley.

Kane, V. E. (1986). Process capability indices. *Journal of Quality Technology*, *18*, 41-52.

Kaplan, E. L., & Meier, P. (1958). Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, *53*, 457-481.

Karsten, K. G., (1925). *Charts and graphs*. New York: Prentice-Hall.

Kass, G. V. (1980). An exploratory technique for investigating large quantities of categorical data. *Applied Statistics*, *29*, 119-127.

Keats, J. B., & Lawrence, F. P. (1997). Weibull maximum likelihood parameter estimates with censored data. *Journal of Quality Technology*, 29, 105-110.

Keeves, J. P. (1972). *Educational environment and student achievement.* Melbourne: Australian Council for Educational Research.

Kendall, M. G. (1940). Note on the distribution of quantiles for large samples. *Supplement of the Journal of the Royal Statistical Society*, *7*, 83-85.

Kendall, M. G. (1948). *Rank correlation methods*. (1st ed.). London: Griffin.

Kendall, M. G. (1975). *Rank correlation methods* (4th ed.). London: Griffin.

Kendall, M. G. (1984). *Time Series*. New York: Oxford University Press.

Kendall, M., & Ord, J. K. (1990). *Time series* (3rd ed.). London: Griffin.

Kendall, M., & Stuart, A. (1977). *The advanced theory of statistics. (Vol. 1).* New York: MacMillan.

Kendall, M., & Stuart, A. (1979). *The advanced theory of statistics* (Vol. 2). New York: Hafner.

Kennedy, A. D., & Gehan, E. A. (1971). Computerized simple regression methods for survival time studies. *Computer Programs in Biomedicine*, *1*, 235-244.

Kennedy, W. J., & Gentle, J. E. (1980). *Statistical computing.* New York: Marcel Dekker, Inc.

Kenny, D. A. (1979). *Correlation and causality.* New York: Wiley.

Keppel, G. (1973). *Design and analysis: A researcher's handbook.* Engelwood Cliffs, NJ: Prentice-Hall.

Keppel, G. (1982). *Design and analysis: A researcher's handbook* (2nd ed.). Engelwood Cliffs, NJ: Prentice-Hall.

Keselman, H. J., Rogan, J. C., Mendoza, J. L., & Breen, L. L. (1980). Testing the validity conditions for repeated measures *F* tests. *Psychological Bulletin*, *87*, 479-481.

Khuri, A. I., & Cornell, J. A. (1987). *Response surfaces: Designs and analyses*. New York: Marcel Dekker, Inc.

Kiefer, J., & Wolfowitz, J. (1960). The equivalence of two extremum problems. *Canadian Journal of Mathematics, 12*, 363-366.

Kim, J. O., & Mueller, C. W. (1978a). *Factor analysis: Statistical methods and practical issues*. Beverly Hills, CA: Sage Publications.

Kim, J. O., & Mueller, C. W. (1978b). *Introduction to factor analysis: What it is and how to do it.* Beverly Hills, CA: Sage Publications.

Kirk, D. B. (1973). On the numerical approximation of the bivariate normal (tetrachoric) correlation coefficient. *Psychometrika, 38,* 259-268.

Kirk, R. E. (1968). *Experimental design: Procedures for the behavioral sciences*. (1st ed.). Monterey, CA: Brooks/Cole.

Kirk, R. E. (1982). *Experimental design: Procedures for the behavioral sciences*. (2nd ed.). Monterey, CA: Brooks/Cole.

Kirk, R. E. (1995). *Experimental design: Procedures for the behavioral sciences.* Pacific Grove, CA: Brooks-Cole.

Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P. (1983). Optimization by simulated annealing. *Science 220 (4598)*, 671-680.

Kish, L. (1965). *Survey sampling.* New York: Wiley.

Kivenson, G. (1971). *Durability and reliability in engineering design*. New York: Hayden.

Klecka, W. R. (1980). *Discriminant analysis*. Beverly Hills, CA: Sage.

Klein, L. R. (1974). *A textbook of econometrics*. Englewood Cliffs, NJ: Prentice-Hall.

Kleinbaum, D. G. (1996). *Survival analysis: A self-learning text.* New York: Springer-Verlag.

Kline, P. (1979). *Psychometrics and psychology*. London: Academic Press.

Kline, P. (1986). *A handbook of test construction.* New York: Methuen.

Kmenta, J. (1971). *Elements of econometrics*. New York: Macmillan.

Knuth, Donald E. (1981). *Seminumerical algorithms.* 2nd ed., Vol 2 of: *The art of computer programming*. Reading, Mass.: Addison-Wesley.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43, 59-69.

Kohonen, T. (1990). Improved versions of learning vector quantization. *International Joint Conference on Neural Networks 1*, 545-550. San Diego, CA.

Kolata, G. (1984). The proper display of data. *Science*, *226*, 156-157.

Kolmogorov, A. (1941). Confidence limits for an unknown distribution function. *Annals of Mathematical Statistics*, *12*, 461-463.

Korin, B. P. (1969). On testing the equality of *k* covariance matrices. *Biometrika*, *56*, 216-218.

Kramer, M.A. (1991). Nonlinear principal components analysis using autoassociative neural networks. *AIChe Journal 37 (2)*, 233-243.

Kruskal, J. B. (1964). Nonmetric multidimensional scaling: A numerical method. *Pyrometrical*, *29*, 1-27, 115-129.

Kruskal, J. B., & Wish, M. (1978). *Multidimensional scaling*. Beverly Hills, CA: Sage Publications.

Kruskal, W. H. (1952). A nonparametric test for the several sample problem. *Annals of Mathematical Statistics*, *23*, 525-540.

Kruskal, W. H. (1975). Visions of maps and graphs. In J. Kavaliunas (Ed.), *Auto-carto II, proceedings of the international symposium on computer assisted cartography*. Washington, DC: U. S. Bureau of the Census and American Congress on Survey and Mapping.

Kruskal, W. H., & Wallis, W. A. (1952). Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association*, *47*, 583-621.

Ku, H. H., & Kullback, S. (1968). Interaction in multidimensional contingency tables: An information theoretic approach. *J. Res. Nat. Bur. Standards Sect. B*, 72, 159-199.

Ku, H. H., Varner, R. N., & Kullback, S. (1971). Analysis of multidimensional contingency tables. *Journal of the American Statistical Association*, *66*, 55-64.

Kullback, S. (1959). *Information theory and statistics.* New York: Wiley.

Kvålseth, T. O. (1985). Cautionary note about R2. *The American Statistician, 39*, 279-285.

Lagakos, S. W., & Kuhns, M. H. (1978). Maximum likelihood estimation for censored exponential survival data with covariates. *Applied Statistics*, *27*, 190-197.

Lakatos, E., & Lan, K. K. G. (1992). A comparison of sample size methods for the logrank statistic. *Statistics in Medicine*, *11*, 179191.

Lance, G. N., & Williams, W. T. (1966). A general theory of classificatory sorting strategies. *Computer Journal*, *9*, 373.

Lance, G. N., & Williams, W. T. (1966). Computer programs for hierarchical polythetic classification ("symmetry analysis"). *Computer Journal*, *9*, 60.

Larsen, W. A., & McCleary, S. J. (1972). The use of partial residual plots in regression analysis. *Technometrics*, *14*, 781-790.

Lawless, J. F. (1982). *Statistical models and methods for lifetime data*. New York: Wiley.

Lawley, D. N., & Maxwell, A. E. (1971). *Factor analysis as a statistical method.* New York: American Elsevier.

Lawley, D. N., & Maxwell, A. E. (1971). *Factor analysis as a statistical method* (2nd. ed.). London: Butterworth & Company.

Lebart, L., Morineau, A., and Tabard, N. (1977). *Techniques de la description statistique.* Paris: Dunod.

Lebart, L., Morineau, A., and Warwick, K., M. (1984). Multivariate descriptive statistical analysis: Correspondence analysis and related techniques for large matrices. New York: Wiley.

Lee, E. T. (1980). *Statistical methods for survival data analysis.* Belmont, CA: Lifetime Learning.

Lee, E. T., & Desu, M. M. (1972). A computer program for comparing *K* samples with right-censored data. *Computer Programs in Biomedicine, 2*, 315-321.

Lee, E. T., Desu, M. M., & Gehan, E. A. (1975). A Monte-Carlo study of the power of some two-sample tests. *Biometrika*, *62*, 425-532.

Lee, S., & Hershberger, S. (1990). A simple rule for generating equivalent models in covariance structure modeling. *Multivariate Behavioral Research*, *25*, 313-334.

Lee, Y. S. (1972). Tables of upper percentage points of the multiple correlation coefficient. *Biometrika, 59,* 175189.

Legendre, A. M. (1805). *Nouvelles Methodes pour la Determination des Orbites des Cometes*. Paris: F. Didot.

Lehmann, E. L. (1975). *Nonparametrics: Statistical methods based on ranks.* San Francisco: Holden-Day.

Levenberg, K. (1944). A method for the solution of certain non-linear problems in least squares. *Quarterly Journal of Applied Mathematics II (2)*, 164-168.

Lewicki, P., Hill, T., & Czyzewska, M. (1992). Nonconscious acquisition of information. *American Psychologist*, 47, 796-801.

Lieblein, J. (1953). On the exact evaluation of the variances and covariances of order statistics in samples form the extreme-value distribution. *Annals of Mathematical Statistics*, *24*, 282-287.

Lieblein, J. (1955). On moments of order statistics from the Weibull distribution. *Annals of Mathematical Statistics*, *26*, 330-333.

Lilliefors, H. W. (1967). On the Kolmogorov-Smirnov test for normality with mean and variance unknown. *Journal of the American Statistical Association, 64,* 399-402.

Lim, T.-S., Loh, W.-Y., & Shih, Y.-S. (1997). An emprical comparison of decision trees and other classification methods. *Technical Report 979,* Department of Statistics, University of Winconsin, Madison.

Lindeman, R. H., Merenda, P. F., & Gold, R. (1980). *Introduction to bivariate and multivariate analysis*. New York: Scott, Foresman, & Co.

Lindman, H. R. (1974). *Analysis of variance in complex experimental designs*. San Francisco: W. H. Freeman & Co.

Linfoot, E. H. (1957). An informational measure of correlation. *Information and Control*, *1*, 50-55.

Linn, R. L. (1968). A Monte Carlo approach to the number of factors problem. *Psychometrika*, *33*, 37-71.

Lipson, C., & Sheth, N. C. (1973). *Statistical design and analysis of engineering experiments*. New York: McGraw-Hill.

Lloyd, D. K., & Lipow, M. (1977). *Reliability: Management, methods, and mathematics.* New York: McGraw-Hill.

Loehlin, J. C. (1987). *Latent variable models: An introduction to latent, path, and structural analysis.* Hillsdale, NJ: Erlbaum.

Loh, W.-Y, & Shih, Y.-S. (1997). Split selection methods for classification trees. *Statistica Sinica, 7,* 815-840.

Loh, W.-Y., & Vanichestakul, N. (1988). Tree-structured classification via generalized discriminant analysis (with discussion). *Journal of the American Statistical Association, 83,* 715-728.

Long, J. S. (1983a). *Confirmatory factor analysis.* Beverly Hills: Sage.

Long, J. S. (1983b). *Covariance structure models: An introduction to LISREL.* Beverly Hills: Sage.

Longley, J. W. (1967). An appraisal of least squares programs for the electronic computer from the point of view of the user. *Journal of the American Statistical Association, 62,* 819-831.

Longley, J. W. (1984). *Least squares computations using orthogonalization methods.* New York: Marcel Dekker.

Lord, F. M. (1957). A significance test for the hypothesis that two variables measure the same trait except for errors of measurement. *Psychometrika, 22,* 207-220.

Lorenz, M. O. (1904). Methods of measuring the concentration of wealth. *American Statistical Association Publication, 9,* 209-219.

Lowe, D. (1989). Adaptive radial basis function non-linearities, and the problem of generalisation. *First IEEE International Conference on Artificial Neural Networks*, 171-175, London, UK.

Lucas, J. M. (1976). The design and use of cumulative sum quality control schemes. *Journal of Quality Technology, 8,* 45-70.

Lucas, J. M. (1982). Combined Shewhart-CUSUM quality control schemes. *Journal of Quality Technology, 14*, 89-93.

MacCallum, R. C., Browne, M. W., & Sugawara, H. M. (1996). Power analysis and determination of sample size for covariance structur modeling. *Psychological Methods, 1,* 130149.

Maddala, G. S. (1977) *Econometrics*. New York: McGraw-Hill.

Maiti, S. S., & Mukherjee, B. N. (1990). A note on the distributional properties of the Jöreskog-Sörbom fit indices. *Psychometrika, 55*, 721-726.

Makridakis, S. G. (1983). Empirical evidence versus personal experience. *Journal of Forecasting, 2,* 295-306.

Makridakis, S. G. (1990). *Forecasting, planning, and strategy for the 21st century.* London: Free Press.

Makridakis, S. G., & Wheelwright, S. C. (1978). *Interactive forecasting: Univariate and multivariate methods* (2nd ed.). San Francisco, CA: Holden-Day.

Makridakis, S. G., & Wheelwright, S. C. (1989). *Forecasting methods for management* (5th ed.). New York: Wiley.

Makridakis, S. G., Wheelwright, S. C., & McGee, V. E. (1983). *Forecasting: Methods and applications* (2nd ed.). New York: Wiley.

Makridakis, S., Andersen, A., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., Newton, J., Parzen, R., & Winkler, R. (1982). The accuracy of extrapolation (time series) methods: Results of a forecasting competition. *Journal of Forecasting, 1,* 11-153.

Malinvaud, E. (1970). *Statistical methods of econometrics*. Amsterdam: North-Holland Publishing Co.

Mandel, B. J. (1969). The regression control chart. *Journal of Quality Technology, 1,* 3-10.

Mann, H. B., & Whitney, D. R. (1947). On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics, 18,* 50-60.

Mann, N. R., Schafer, R. E., & Singpurwalla, N. D. (1974). *Methods for statistical analysis of reliability and life data*. New York: Wiley.

Mann, N. R., Scheuer, R. M, & Fertig, K. W. (1973). A new goodness of fit test for the two-parameter Weibull or exteme value distribution. *Communications in Statistics*, 2, 383-400.

Mantel, N. (1966). Evaluation of survival data and two new rank order statistics arising in its consideration. *Cancer Chemotherapy Reports*, *50*, 163-170.

Mantel, N. (1967). Ranking procedures for arbitrarily restricted observations. *Biometrics, 23*, 65-78.

Mantel, N. (1974). Comment and suggestion on the Yates continuity correction. *Journal of the American Statistical Association, 69,* 378-380.

Mantel, N., & Haenszel, W. (1959). Statistical aspects of the analysis of data from retrospective studies of disease. *Journal of the National Cancer Institute*, *22*, 719-748.

Marascuilo, L. A., & McSweeney, M. (1977). *Nonparametric and distribution free methods for the social sciences.* Monterey, CA: Brooks/Cole.

Marple, S. L., Jr. (1987). *Digital spectral analysis.* Englewood Cliffs, NJ: Prentice-Hall.

Marquardt, D.W. (1963). An algorithm for least-squares estimation of non-linear parameters. *Journal of the Society of Industrial and Applied Mathematics 11 (2),* 431-441.

Marsaglia, G. (1962). Random variables and computers. In J. Kozenik (Ed.), *Information theory, statistical decision functions, random processes: Transactions of the third Prague Conference*. Prague: Czechoslovak Academy of Sciences.

Mason, R. L., Gunst, R. F., & Hess, J. L. (1989). *Statistical design and analysis of experiments with applications to engineering and science.* New York: Wiley.

Massey, F. J., Jr. (1951). The Kolmogorov-Smirnov test for goodness of fit. *Journal of the American Statistical Association, 46,* 68-78.

Masters (1995). *Neural, Novel, and Hybrid Algorithms for Time Series Predictions*. New York: Wiley.

Matsueda, R. L., & Bielby, W. T. (1986). Statistical power in covariance structure models. In N. B. Tuma (Ed.), *Sociological methodology.* Washington, DC: American Sociological Association.

McArdle, J. J. (1978). A structural view of structural models. Paper presented at the *Winter Workshop on Latent Structure Models Applied to Developmental Data, University of Denver, December, 1978.*

McArdle, J. J., & McDonald, R. P. (1984). Some algebraic properties of the Reticular Action Model for moment structures. *British Journal of Mathematical and Statistical Psychology, 37*, 234-251.

McCleary, R., & Hay, R. A. (1980). *Applied time series analysis for the social sciences*. Beverly Hills, CA: Sage Publications.

McCullagh, P. & Nelder, J. A. (1989). *Generalized linear models* (2nd Ed.). New York: Chapman & Hall.

McDonald, R. P. (1980). A simple comprehensive model for the analysis of covariance structures. *British Journal of Mathematical and Statistical Psychology, 31*, 59-72.

McDonald, R. P. (1989). An index of goodness-of-fit based on noncentrality. *Journal of Classification*, *6,* 97-103.

McDonald, R. P., & Hartmann, W. M. (1992). A procedure for obtaining initial value estimates in the RAM model. *Multivariate Behavioral Research*, *27*, 57-76.

McDonald, R. P., & Mulaik, S. A. (1979). Determinacy of common factors: A nontechnical review. *Psychological Bulletin*, *86*, 297-306.

McDowall, D., McCleary, R., Meidinger, E. E., & Hay, R. A. (1980). *Interrupted time series analysis*. Beverly Hills, CA: Sage Publications.

McKenzie, E. (1984). General exponential smoothing and the equivalent ARMA process. *Journal of Forecasting*, *3*, 333-344.

McKenzie, E. (1985). Comments on 'Exponential smoothing: The state of the art' by E. S. Gardner, Jr. *Journal of Forecasting*, *4*, 32-36.

McLachlan, G. J. (1992). *Discriminant analysis and statistical pattern recognition.* New York: Wiley.

McLain, D. H. (1974). Drawing contours from arbitrary data points. *The Computer Journal*, *17*, 318-324.

McLean, R. A., & Anderson, V. L. (1984). *Applied factorial and fractional designs.* New York: Marcel Dekker.

McLeod, A. I., & Sales, P. R. H. (1983). An algorithm for approximate likelihood calculation of ARMA and seasonal ARMA models. *Applied Statistics*, 211-223 (Algorithm AS).

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, *12*, 153-157.

McNemar, Q. (1969). *Psychological statistics* (4th ed.). New York: Wiley.

Melard, G. (1984). A fast algorithm for the exact likelihood of autoregressive-moving average models. *Applied Statistics*, 33, 104-119.

Mels, G. (1989). A general system for path analysis with latent variables. M. S. Thesis: Department of Statistics, University of South Africa.

Mendoza, J. L., Markos, V. H., & Gonter, R. (1978). A new perspective on sequential testing procedures in canonical analysis: A Monte Carlo evaluation. *Multivariate Behavioral Research*, *13*, 371-382.

Meredith, W. (1964). Canonical correlation with fallible data. *Psychometrika*, *29*, 55-65.

Miettinnen, O. S. (1968). The matched pairs design in the case of all-or-none responses. *Biometrics, 24,* 339352.

Miller, R. (1981). *Survival analysis*. New York: Wiley.

Milligan, G. W. (1980). An examination of the effect of six types of error perturbation on fifteen clustering algorithms. *Psychometrika*, *45*, 325-342.

Milliken, G. A., & Johnson, D. E. (1984). *Analysis of messy data: Vol. I. Designed experiments.* New York: Van Nostrand Reinhold, Co.

Milliken, G. A., & Johnson, D. E. (1992). *Analysis of messy data: Vol. I. Designed experiments.* New York: Chapman & Hall.

Minsky, M.L. and Papert, S.A. (1969). *Perceptrons.* Cambridge, MA: MIT Press.

Mitchell, T. J. (1974a). Computer construction of "D-optimal" first-order designs. *Technometrics*, *16*, 211-220.

Mitchell, T. J. (1974b). An algorithm for the construction of "D-optimal" experimental designs. *Technometrics*, *16*, 203-210.

Mittag, H. J. (1993). *Qualitätsregelkarten.* München/Wien: Hanser Verlag.

Mittag, H. J., & Rinne, H. (1993). *Statistical methods of quality assurance.* London/New York: Chapman & Hall.

Monro, D. M. (1975). Complex discrete fast Fourier transform. *Applied Statistics*, *24*, 153-160.

Monro, D. M., & Branch, J. L. (1976). The chirp discrete Fourier transform of general length. *Applied Statistics*, *26*, 351-361.

Montgomery, D. C. (1976). *Design and analysis of experiments*. New York: Wiley.

Montgomery, D. C. (1985). *Statistical quality control*. New York: Wiley.

Montgomery, D. C. (1991) *Design and analysis of experiments* (3rd ed.). New York: Wiley.

Montgomery, D. C. (1996). *Introduction to Statistical Quality Control* (3rd Edition). New York:Wiley.

Montgomery, D. C. (1996). *Statistical quality control* (3rd. Edition). New York: Wiley.

Montgomery, D. C., & Wadsworth, H. M. (1972). Some techniques for multivariate quality control applications. *Technical Conference Transactions*. Washington, DC: American Society for Quality Control.

Montgomery, D. C., Johnson, L. A., & Gardiner, J. S. (1990). *Forecasting and time series analysis* (2nd ed.). New York: McGraw-Hill.

Mood, A. M. (1954). *Introduction to the theory of statistics.* New York: McGraw Hill.

Moody, J. and Darkin, C.J. (1989). Fast learning in networks of locally-tuned processing units. *Neural Computation 1 (2)*, 281-294.

Moré, J. J., (1977). *The Levenberg-Marquardt Algorithm: Implementation and Theory.* In G.A. Watson, (ed.), *Lecture Notes in Mathematics 630*, p. 105-116. Berlin: Springer-Verlag.

Morgan, J. N., & Messenger, R. C. (1973). THAID: A sequential analysis program for the analysis of nominal scale dependent variables. *Technical report,* Institute of Social Research, University of Michigan, Ann Arbor.

Morgan, J. N., & Sonquist, J. A. (1963). Problems in the analysis of survey data, and a proposal. *Journal of the American Statistical Association, 58*, 415-434.

Morris, M., & Thisted, R. A. (1986). Sources of error in graphical perception: A critique and an experiment. *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 43-48.

Morrison, A. S., Black, M. M., Lowe, C. R., MacMahon, B., & Yuasa, S. (1973). Some international differences in histology and survival in breast cancer. *International Journal of Cancer*, *11*, 261-267.

Morrison, D. (1967). *Multivariate statistical methods.* New York: McGraw-Hill.

Morrison, D. F. (1990). *Multivariate statistical methods.* (3rd Ed.). New York: McGraw-Hill.

Moses, L. E. (1952). Non-parametric statistics for psychological research. *Psychological Bulletin*, *49*, 122-143.

Mulaik, S. A. (1972). *The foundations of factor analysis.* New York: McGraw Hill.

Murphy, K. R., & Myors, B. (1998). *Statistical power analysis: A simple general model for traditional and modern hypothesis tests.* Mahwah, NJ: Lawrence Erlbaum Associates.

Muth, J. F. (1960). Optimal properties of exponentially weighted forecasts. *Journal of the American Statistical Association*, *55*, 299-306.

Nachtsheim, C. J. (1979). *Contributions to optimal experimental design.* Ph.D. thesis, Department of Applied Statistics, University of Minnesota.

Nachtsheim, C. J. (1987). Tools for computer-aided design of experiments. *Journal of Quality Technology, 19*, 132-160.

Nelder, J. A., & Mead, R. (1965). A Simplex method for function minimization. *Computer Journal*, *7*, 308-313.

Nelson, L. (1984). The Shewhart control chart - tests for special causes. *Journal of Quality Technology, 15*, 237-239.

Nelson, L. (1985). Interpreting Shewhart X-bar control charts. *Journal of Quality Technology, 17*, 114-116.

Nelson, W. (1982). *Applied life data analysis*. New York: Wiley.

Nelson, W. (1990). *Accelerated testing: Statistical models, test plans, and data analysis*. New York: Wiley.

Neter, J., Wasserman, W., & Kutner, M. H. (1985). *Applied linear statistical models: Regression, analysis of variance, and experimental designs.* Homewood, IL: Irwin.

Neter, J., Wasserman, W., & Kutner, M. H. (1989). *Applied linear regression models* (2nd ed.). Homewood, IL: Irwin.

Newcombe, Robert G. (1998). Two-sided confidence intervals for the single proportion: comparison of seven methods. *Statistics in Medicine, 17,* 857872.

Neyman, J., & Pearson, E. S. (1931). On the problem of *k* samples. *Bulletin de l'Academie Polonaise des Sciences et Lettres,* Ser. A, 460-481.

Neyman, J., & Pearson, E. S. (1933). On the problem of the most efficient tests of statistical hypothesis. *Philosophical Transactions of the Royal Society of London,* Ser. A, *231,* 289-337.

Nisbett, R. E., Fong, G. F., Lehman, D. R., & Cheng, P. W. (1987). Teaching reasoning. *Science*, *238*, 625-631.

Noori, H. (1989). The Taguchi methods: Achieving design and output quality. *The Academy of Management Executive, 3*, 322-326.

Nunally, J. C. (1970). *Introduction to psychological measurement*. New York: McGraw-Hill.

Nunnally, J. C. (1978). *Psychometric theory*. New York: McGraw-Hill.

Nussbaumer, H. J. (1982). *Fast Fourier transforms and convolution algorithms* (2nd ed.). New York: Springer-Verlag.

O'Brien, R. G., & Kaiser, M. K. (1985). MANOVA method for analyzing repeated measures designs: An extensive primer. *Psychological Bulletin, 97*, 316-333.

Okunade, A. A., Chang, C. F., & Evans, R. D. (1993). Comparative analysis of regression output summary statistics in common statistical packages. *The American Statistician, 47*, 298-303.

Olds, E. G. (1949). The 5% significance levels for sums of squares of rank differences and a correction. *Annals of Mathematical Statistics, 20,* 117-118.

Olejnik, S. F., & Algina, J. (1987). Type I error rates and power estimates of selected parametric and nonparametric tests of scale. *Journal of Educational Statistics, 12,* 45-61.

Olson, C. L. (1976). On choosing a test statistic in multivariate analysis of variance. *Psychological Bulletin, 83*, 579-586.

O'Neill, R. (1971). Function minimization using a Simplex procedure. *Applied Statistics, 3*, 79-88.

Ostle, B., & Malone, L. C. (1988). *Statistics in research: Basic concepts and techniques for research workers* (4th ed.). Ames, IA: Iowa State Press.

Ostrom, C. W. (1978). *Time series analysis: Regression techniques*. Beverly Hills, CA: Sage Publications.

Overall, J. E., & Speigel, D. K. (1969). Concerning least squares analysis of experimental data. *Psychological Bulletin, 83*, 579-586.

Page, E. S. (1954). Continuous inspection schemes. *Biometrics, 41*, 100-114.

Page, E. S. (1961). Cumulative sum charts. *Technometrics, 3,* 1-9.

Palumbo, F. A., & Strugala, E. S. (1945). Fraction defective of battery adapter used in handie-talkie. *Industrial Quality Control, November*, 68.

Pankratz, A. (1983). *Forecasting with univariate Box-Jenkins models: Concepts and cases*. New York: Wiley.

Parker, D.B. (1985). *Learning logic. Technical Report TR-47,* Cambridge, MA: MIT Center for Research in Computational Economics and Management Science.

Parzen, E. (1961). Mathematical considerations in the estimation of spectra: Comments on the discussion of Messers, Tukey, and Goodman. *Technometrics, 3,* 167-190; 232-234.

Parzen, E. (1962). On estimation of a probability density function and mode. *Annals of Mathematical Statistics 33*, 1065-1076.

Patil, K. D. (1975). Cochran's Q test: Exact distribution. *Journal of the American Statistical Association*, *70*, 186-189.

Patterson, D. (1996). *Artificial Neural Networks.* Singapore: Prentice Hall.

Peace, G. S. (1993). *Taguchi methods: A hands-on approach*. Milwaukee, Wisconsin: ASQC.

Pearson, E. S., and Hartley, H. O. (1972). *Biometrika tables for statisticians, Vol II.* Cambridge: Cambridge University Press.

Pearson, K. (1894). Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London*, Ser. A, *185*, 71-110.

Pearson, K. (1895). Skew variation in homogeneous material. *Philosophical Transactions of the Royal Society of London*, Ser. A, *186*, 343-414.

Pearson, K. (1896). Regression, heredity, and panmixia. *Philosophical Transactions of the Royal Society of London*, Ser. A, *187*, 253-318.

Pearson, K. (1900). On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine*, 5th Ser., *50*, 157-175.

Pearson, K. (1904). On the theory of contingency and its relation to association and normal correlation. *Drapers' Company Research Memoirs*, Biometric Ser. I.

Pearson, K. (1905). Das Fehlergesetz und seine Verallgemeinerungen durch Fechner und Pearson. A Rejoinder. *Biometrika*, *4*, 169-212.

Pearson, K. (1908). On the generalized probable error in multiple normal correlation. *Biometrika, 6*, 59-68.

Pearson, K., (Ed.). (1968). *Tables of incomplete beta functions* (2nd ed.). Cambridge, MA: Cambridge University Press.

Pedhazur, E. J. (1973). *Multiple regression in behavioral research*. New York: Holt, Rinehart, & Winston.

Pedhazur, E. J. (1982). *Multiple regression in behavioral research* (2nd ed.). New York: Holt, Rinehart, & Winston.

Peressini, A. L., Sullivan, F. E., & Uhl, J. J., Jr. (1988). *The mathematics of nonlinear programming*. New York: Springer.

Peto, R., & Peto, J. (1972). Asymptotically efficient rank invariant procedures. *Journal of the Royal Statistical Society*, *135*, 185-207.

Phadke, M. S. (1989). *Quality engineering using robust design.* Englewood Cliffs, NJ: Prentice-Hall.

Phatak, A., Reilly, P. M., and Penlidis, A. (1993) An Approach to Interval Estimation in Partial Least Squares Regression, *Analytica Chimica Acta*, 277, 495-501

Piatetsky-Shapiro, G. (Ed.) (1993). *Proceedings of AAAI-93 Workshop on Knowledge Discovery in Databases*. Menlo Park, CA: AAAI Press.

Piepel, G. F. (1988). Programs for generating extreme vertices and centroids of linearly constrained experimental regions. *Journal of Quality Technology, 20*, 125-139.

Piepel, G. F., & Cornell, J. A. (1994). Mixture experiment approaches: Examples, discussion, and recommendations. *Journal of Quality Technology*, *26*, 177-196.

Pigou, A. C. (1920). *Economics of Welfare*. London: Macmillan.

Pike, M. C. (1966). A method of analysis of certain class of experiments in carcinogenesis. *Biometrics, 22*, 142-161.

Pillai, K. C. S. (1965). On the distribution of the largest characteristic root of a matrix in multivariate analysis. *Biometrika*, *52*, 405-414.

Plackett, R. L., & Burman, J. P. (1946). The design of optimum multifactorial experiments. *Biometrika*, *34*, 255-272.

Polya, G. (1920). Uber den zentralen Grenzwertsatz der Wahrscheinlichkeitsrechnung und das Momentenproblem. *Mathematische Zeitschrift*, *8*, 171-181.

Porebski, O. R. (1966). Discriminatory and canonical analysis of technical college data. *British Journal of Mathematical and Statistical Psychology*, *19*, 215-236.

Powell, M. J. D. (1964). An efficient method for finding the minimum of a function of several variables without calculating derivatives. *Computer Journal*, *7*, 155-162.

Pregibon, D. (1997). Data Mining. *Statistical Computing and Graphics*, *7*, 8.

Prentice, R. (1973). Exponential survivals with censoring and explanatory variables. *Biometrika, 60*, 279-288.

Press, W. H., Flannery, B. P., Teukolsky, S. A., Vetterling, W. T. (1992). *Numerical recipies (2nd Edition).* New York: Cambridge University Press.

Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (1992). *Numerical Recipes in C: The Art of Scientific Computing (Second ed.).* Cambridge University Press.

Press, William, H., Flannery, B. P., Teukolsky, S. A., Vetterling, W. T. (1986). *Numerical recipies*. New York: Cambridge University Press.

Priestley, M. B. (1981). *Spectral analysis and time series.* New York: Academic Press.

Pyzdek, T. (1989). *What every engineer should know about quality control.* New York: Marcel Dekker.

Quinlan. (1992). *C4.5: Programs for Machine Learning*, Morgan Kaufmann

Quinlan, J.R., & Cameron-Jones, R.M. (1995). Oversearching and layered search in empirical learning. *Proceedings of the 14th International Joint Conference on Artificial Intelligence, Montreal* (Vol. 2). Morgan Kaufman, 1019-10244.

Ralston, A., & Wilf, H.S. (Eds.). (1960). *Mathematical methods for digital computers.* New York: Wiley.

Ralston, A., & Wilf, H.S. (Eds.). (1967). *Mathematical methods for digital computers* (Vol. II). New York: Wiley.

Randles, R. H., & Wolfe, D. A. (1979). *Introduction to the theory of nonparametric statistics.* New York: Wiley.

Rannar, S., Lindgren, F., Geladi, P, and Wold, S. (1994) A PLS Kernel Algorithm for Data Sets with Many Variables and Fewer Objects. Part 1: Theory and Algorithm, *Journal of Chemometrics*, 8, 111-125.

Rao, C. R. (1951). An asymptotic expansion of the distribution of Wilks' criterion. *Bulletin of the International Statistical Institute, 33,* 177-181.

Rao, C. R. (1952). *Advanced statistical methods in biometric research*. New York: Wiley.

Rao, C. R. (1965). *Linear statistical inference and its applications.* New York: Wiley.

Rhoades, H. M., & Overall, J. E. (1982). A sample size correction for Pearson chi-square in 2 x 2 contingency tables. *Psychological Bulletin, 91,* 418-423.

Rinne, H., & Mittag, H. J. (1995). *Statistische Methoden der Qualitätssicherung (3rd. edition).* München/Wien: Hanser Verlag.

Ripley, B. D. (1981). *Spacial statistics.* New York: Wiley.

Ripley, B. D. (1996). *Pattern recognition and neural networks*. Cambridge: Cambridge University Press.

Ripley, B. D., (1996) *Pattern Recognition and Neural Networks*, Cambridge University Press

Rodriguez, R. N. (1992). Recent developments in process capability analysis. *Journal of Quality Technology, 24*, 176-187.

Rogan, J. C., Keselman, J. J., & Mendoza, J. L. (1979). Analysis of repeated measurements. *British Journal of Mathematical and Statistical Psychology, 32,* 269-286.

Rosenberg, S. (1977). New approaches to the analysis of personal constructs in person perception. In A. Landfield (Ed.), *Nebraska symposium on motivation* (Vol. 24). Lincoln, NE: University of Nebraska Press.

Rosenberg, S., & Sedlak, A. (1972). Structural representations of implicit personality theory. In L. Berkowitz (Ed.). *Advances in experimental social psychology* (Vol. 6). New York: Academic Press.

Rosenblatt, F. (1958). The Perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review 65*, 386-408.

Roskam, E. E., & Lingoes, J. C. (1970). MINISSA-I: A Fortran IV program for the smallest space analysis of square symmetric matrices. *Behavioral Science, 15,* 204-205.

Ross, P. J. (1988). *Taguchi techniques for quality engineering: Loss function, orthogonal experiments, parameter, and tolerance design*. Milwaukee, Wisconsin: ASQC.

Roy, J. (1958). Step-down procedure in multivariate analysis. *Annals of Mathematical Statistics*, *29*, 1177-1187.

Roy, J. (1967). *Some aspects of multivariate analysis.* New York: Wiley.

Roy, R. (1990). *A primer on the Taguchi method*. Milwaukee, Wisconsin: ASQC.

Royston, J. P. (1982). An extension of Shapiro and Wilks' W test for normality to large samples. *Applied Statistics, 31*, 115-124.

Rozeboom, W. W. (1979). Ridge regression: Bonanza or beguilement? *Psychological Bulletin, 86*, 242-249.

Rozeboom, W. W. (1988). Factor indeterminacy: the saga continues. *British Journal of Mathematical and Statistical Psychology, 41,* 209-226.

Rubinstein, L.V., Gail, M. H., & Santner, T. J. (1981). Planning the duration of a comparative clinical trial with loss to follow-up and a period of continued observation. *Journal of Chronic Diseases, 34,* 469479.

Rud, O., P. (2001). *Data mining cookbook: Modeling data for marketing, risk, and customer relationship management.* NY: Wiley.

Rumelhart, D.E. and McClelland, J. (eds.) (1986). *Parallel Distributed Processing, Vol 1.* Cambridge, MA: MIT Press.

Rumelhart, D.E., Hinton, G.E. and Williams, R.J. (1986). Learning internal representations by error propagation. In D.E. Rumelhart, J.L. McClelland (Eds.), *Parallel Distributed Processing, Vol 1.* Cambridge, MA: MIT Press.

Runyon, R. P., & Haber, A. (1976). *Fundamentals of behavioral statistics*. Reading, MA: Addison-Wesley.

Ryan, T. P. (1989). *Statistical methods for quality improvement.* New York: Wiley.

Ryan, T. P. (1997). *Modern Regression Methods*. New York: Wiley.

Sandler, G. H. (1963). *System reliability engineering*. Englewood Cliffs, NJ: Prentice-Hall.

SAS Institute, Inc. (1982). *SAS user's guide: Statistics, 1982 Edition*. Cary, NC: SAS Institute, Inc.

Satorra, A., & Saris, W. E. (1985). Power of the likelihood ratio test in covariance structure analysis. *Psychometrika, 50,* 83-90.

Saxena, K. M. L., & Alam, K. (1982). Estimation of the noncentrality parameter of a chi squared distribution. *Annals of Statistics, 10,* 1012-1016.

Scheffé, H. (1953). A method for judging all possible contrasts in the analysis of variance. *Biometrica, 40,* 87-104.

Scheffé, H. (1959). *The analysis of variance*. New York: Wiley.

Scheffé, H. (1963). The simplex-centroid design for experiments with mixtures. *Journal of the Royal Statistical Society, B25,* 235-263.

Scheffé, H., & Tukey, J. W. (1944). A formula for sample sizes for population tolerance limits. *Annals of Mathematical Statistics, 15,* 217.

Scheines, R. (1994). Causation, indistinguishability, and regression. In F. Faulbaum, (Ed.), *SoftStat '93. Advances in statistical software 4*. Stuttgart: Gustav Fischer Verlag.

Schiffman, S. S., Reynolds, M. L., & Young, F. W. (1981). *Introduction to multidimensional scaling: Theory, methods, and applications*. New York: Academic Press.

Schmidt, F. L., & Hunter, J. E. (1997). Eight common but false objections to the discontinuation of significance testing in the analysis of research data. In Harlow, L. L., Mulaik, S. A., & Steiger, J. H. (Eds.), *What if there were no significance tests.* Mahwah, NJ: Lawrence Erlbaum Associates.

Schmidt, P., & Muller, E. N. (1978). The problem of multicollinearity in a multistage causal alienation model: A comparison of ordinary least squares, maximum-likelihood and ridge estimators. *Quality and Quantity*, *12*, 267-297.

Schmidt, P., & Sickles, R. (1975). On the efficiency of the Almon lag technique. *International Economic Review*, *16*, 792-795.

Schmidt, P., & Waud, R. N. (1973). The Almon lag technique and the monetary versus fiscal policy debate. *Journal of the American Statistical Association*, *68*, 11-19.

Schnabel, R. B., Koontz, J. E., and Weiss, B. E. (1985). A modular system of algorithms for unconstrained minimization. *ACM Transactions on Mathematical Software*, *11*, 419-440.

Schneider, H. (1986). *Truncated and censored samples from normal distributions*. New York: Marcel Dekker.

Schneider, H., & Barker, G.P. (1973). *Matrices and linear algebra* (2nd ed.). New York: Dover Publications.

Schönemann, P. H., & Steiger, J. H. (1976). Regression component analysis. *British Journal of Mathematical and Statistical Psychology*, *29*, 175-189.

Schrock, E. M. (1957). *Quality control and statistical methods*. New York: Reinhold Publishing.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, *6*, 461-464.

Scott, D. W. (1979). On optimal and data-based histograms. *Biometrika*, *66*, 605-610.

Searle, S. R. (1987). *Linear models for unbalanced data*. New York: Wiley.

Searle, S. R., Casella, G., & McCullock, C. E. (1992). *Variance components*. New York: Wiley.

Searle, S., R., Speed., F., M., & Milliken, G. A. (1980). The population marginal means in the linear model: An alternative to least squares means. *The American Statistician*, *34,* 216-221.

Seber, G. A. F., & Wild, C. J. (1989). *Nonlinear regression.* New York: Wiley.

Sebestyen, G. S. (1962). *Decision making processes in pattern recognition.* New York: Macmillan.

Sen, P. K., & Puri, M. L. (1968). On a class of multivariate multisample rank order tests, II: Test for homogeneity of dispersion matrices. *Sankhya*, *30*, 1-22.

Serlin, R. A., & Lapsley, D. K. (1993). Rational appraisal of psychological research and the good-enough principle. In G. Keren & C. Lewis (Eds.), *A handbook for data analysis in the behavioral sciences: Methodological issues* (pp. 199-228). Hillsdale, NJ: Lawrence Erlbaum Associates.

Serlin. R. A., & Lapsley, D. K. (1985). Rationality in psychological research: The good-enough principle. *American Psychologist,* 40, 7383.

Shapiro, A., & Browne, M. W. (1983). On the investigation of local identifiability: A counter example. *Psychometrika*, *48*, 303-304.

Shapiro, S. S., Wilk, M. B., & Chen, H. J. (1968). A comparative study of various tests of normality. *Journal of the American Statistical Association*, *63*, 1343-1372.

Shepherd, A. J. (1997). *Second-Order Methods for Neural Networks.* New York: Springer.

Shewhart, W. A. (1931). *Economic control of quality of manufactured product*. New York: D. Van Nostrand.

Shewhart, W. A. (1939). *Statistical method from the viewpoint of quality*. Washington, DC: The Graduate School Department of Agriculture.

Shirland, L. E. (1993). *Statistical quality control with microcomputer applications.* New York: Wiley.

Shiskin, J., Young, A. H., & Musgrave, J. C. (1967). *The X-11 variant of the census method II seasonal adjustment program.* (Technical paper no. 15). Bureau of the Census.

Shumway, R. H. (1988). *Applied statistical time series analysis.* Englewood Cliffs, NJ: Prentice Hall.

Siegel, A. E. (1956). Film-mediated fantasy aggression and strength of aggressive drive. *Child Development*, *27*, 365-378.

Siegel, S. (1956). *Nonparametric statistics for the behavioral sciences.* New York: McGraw-Hill.

Siegel, S., & Castellan, N. J. (1988). *Nonparametric statistics for the behavioral sciences* (2nd ed.) New York: McGraw-Hill.

Simkin, D., & Hastie, R. (1986). Towards an information processing view of graph perception. *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 11-20.

Sinha, S. K., & Kale, B. K. (1980). *Life testing and reliability estimation*. New York: Halstead.

Smirnov, N. V. (1948). Table for estimating the goodness of fit of empirical distributions. *Annals of Mathematical Statistics, 19,* 279-281.

Smith, D. J. (1972). *Reliability engineering*. New York: Barnes & Noble.

Smith, K. (1953). Distribution-free statistical methods and the concept of power efficiency. In L. Festinger and D. Katz (Eds.), *Research methods in the behavioral sciences* (pp. 536-577). New York: Dryden.

Sneath, P. H. A., & Sokal, R. R. (1973). *Numerical taxonomy.* San Francisco: W. H. Freeman & Co.

Snee, R. D. (1975). Experimental designs for quadratic models in constrained mixture spaces. *Technometrics*, *17*, 149-159.

Snee, R. D. (1979). Experimental designs for mixture systems with multi-component constraints. *Communications in Statistics - Theory and Methods*, *A8(4)*, 303-326.

Snee, R. D. (1985). Computer-aided design of experiments - some practical experiences. *Journal of Quality Technology*, *17*, 222-236.

Snee, R. D. (1986). An alternative approach to fitting models when re-expression of the response is useful. *Journal of Quality Technology, 18*, 211-225.

Sokal, R. R., & Mitchener, C. D. (1958). A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, *38*, 1409.

Sokal, R. R., & Sneath, P. H. A. (1963). *Principles of numerical taxonomy*. San Francisco: W. H. Freeman & Co.

Soper, H. E. (1914). Tables of Poisson's exponential binomial limit. *Biometrika*, *10,* 25-35.

Spearman, C. (1904). "General intelligence," objectively determined and measured. *American Journal of Psychology*, *15,* 201-293.

Speckt, D.F. (1990). Probabilistic Neural Networks. *Neural Networks 3 (1)*, 109-118.

Speckt, D.F. (1991). A Generalized Regression Neural Network. *IEEE Transactions on Neural Networks 2 (6)*, 568-576.

Spirtes, P., Glymour, C., & Scheines, R. (1993). *Causation, prediction, and search.* Lecture Notes in Statistics, V. 81. New York: Springer-Verlag.

Spjotvoll, E., & Stoline, M. R. (1973). An extension of the *T*-method of multiple comparison to include the cases with unequal sample sizes. *Journal of the American Statistical Association*, *68,* 976-978.

Springer, M. D. (1979). *The algebra of random variables*. New York: Wiley.

Spruill, M. C. (1986). Computation of the maximum likelihood estimate of a noncentrality parameter. *Journal of Multivariate Analysis*, *18*, 216-224.

Steiger, J. H. (1979). Factor indeterminacy in the 1930's and in the 1970's; some interesting parallels. *Psychometrika*, *44*, 157-167.

Steiger, J. H. (1980a). Tests for comparing elements of a correlation matrix. *Psychological Bulletin*, *87*, 245-251.

Steiger, J. H. (1980b). Testing pattern hypotheses on correlation matrices: Alternative statistics and some empirical results. *Multivariate Behavioral Research*, *15*, 335-352.

Steiger, J. H. (1988). Aspects of person-machine communication in structural modeling of correlations and covariances. *Multivariate Behavioral Research*, *23,* 281-290.

Steiger, J. H. (1989). *EzPATH: A supplementary module for SYSTAT and SYGRAPH.* Evanston, IL: SYSTAT, Inc.

Steiger, J. H. (1990). Some additional thoughts on components and factors. *Multivariate Behavioral Research*, *25*, 41-45.

Steiger, J. H., & Browne, M. W. (1984). The comparison of interdependent correlations between optimal linear composites. *Psychometrika*, *49*, 11-24.

Steiger, J. H., & Fouladi, R. T. (1992). *R2:* A computer program for interval estimation, power calculation, and hypothesis testing for the squared multiple correlation. *Behavior Research Methods, Instruments, and Computers, 4,* 581582.

Steiger, J. H., & Fouladi, R. T. (1997). Noncentrality interval estimation and the evaluation of statistical models. In Harlow, L. L., Mulaik, S. A., & Steiger, J. H. (Eds.), *What if there were no significance tests.* Mahwah, NJ: Lawrence Erlbaum Associates.

Steiger, J. H., & Hakstian, A. R. (1982). The asymptotic distribution of elements of a correlation matrix: Theory and application. *British Journal of Mathematical and Statistical Psychology, 35*, 208-215.

Steiger, J. H., & Lind, J. C. (1980). Statistically-based tests for the number of common factors. Paper presented at the annual Spring Meeting of the Psychometric Society in Iowa City. May 30, 1980.

Steiger, J. H., & Schönemann, P. H. (1978). A history of factor indeterminacy. In S. Shye, (Ed.), *Theory Construction and Data Analysis in the Social Sciences.* San Francisco: Jossey-Bass.

Steiger, J. H., Shapiro, A., & Browne, M. W. (1985). On the multivariate asymptotic distribution of sequential chi-square statistics. *Psychometrika, 50*, 253-264.

Stelzl, I. (1986). Changing causal relationships without changing the fit: Some rules for generating equivalent LISREL models. *Multivariate Behavioral Research, 21,* 309-331.

Stenger, F. (1973). Integration formula based on the trapezoid formula. *Journal of the Institute of Mathematics and Applications, 12,* 103-114.

Stevens, J. (1986). *Applied multivariate statistics for the social sciences.* Hillsdale, NJ: Erlbaum.

Stevens, W. L. (1939). Distribution of groups in a sequence of alternatives. *Annals of Eugenics, 9*, 10-17.

Stewart, D. K., & Love, W. A. (1968). A general canonical correlation index. *Psychological Bulletin, 70*, 160-163.

Steyer, R. (1992). *Theorie causale regressionsmodelle* [Theory of causal regression models]. Stuttgart: Gustav Fischer Verlag.

Steyer, R. (1994). Principles of causal modeling: a summary of its mathematical foundations and practical steps. In F. Faulbaum, (Ed.), *SoftStat '93. Advances in statistical software 4.* Stuttgart: Gustav Fischer Verlag.

Stone, M. and Brooks, R. J. (1990) Continuum Regression: Cross-validated Sequentially Constructed Prediction Embracing Ordinary Least Squares, Partial Least Squares, and Principal Components Regression, *Journal of Royal Statistical Society*, 52, No. 2, 237-269.

Student (1908). The probable error of a mean. *Biometrika, 6*, 1-25.

Swallow, W. H., & Monahan, J. F. (1984). Monte Carlo comparison of ANOVA, MIVQUE, REML, and ML estimators of variance components. *Technometrics, 26*, 47-57.

Taguchi, G. (1987). *Jikken keikakuho* (3rd ed., Vol I & II). Tokyo: Maruzen. English translation edited by D. Clausing. *System of experimental design*. New York: UNIPUB/Kraus International

Taguchi, G., & Jugulum, R. (2002). *The Mahalanobis-Taguchi strategy*. New York, NY: Wiley.

Tanaka, J. S., & Huba, G. J. (1985). A fit index for covariance structure models under arbitrary GLS estimation. *British Journal of Mathematical and Statistical Psychology, 38*, 197-201.

Tanaka, J. S., & Huba, G. J. (1989). A general coefficient of determination for covariance structure models under arbitrary GLS estimation. *British Journal of Mathematical and Statistical Psychology*, *42*, 233-239.

Tatsuoka, M. M. (1970). *Discriminant analysis*. Champaign, IL: Institute for Personality and Ability Testing.

Tatsuoka, M. M. (1971). *Multivariate analysis*. New York: Wiley.

Tatsuoka, M. M. (1976). Discriminant analysis. In P. M. Bentler, D. J. Lettieri, and G. A. Austin (Eds.), *Data analysis strategies and designs for substance abuse research.* Washington, DC: U.S. Government Printing Office.

Taylor, D. J., & Muller, K. E. (1995). Computing confidence bounds for power and sample size of the general linear univariate model. *The American Statistician, 49,* 4347.

Thorndike, R. L., & Hagen, E. P. (1977). *Measurement and evaluation in psychology and education*. New York: Wiley.

Thurstone, L. L. (1931). Multiple factor analysis. *Psychological Review, 38,* 406-427.

Thurstone, L. L. (1947). *Multiple factor analysis.* Chicago: University of Chicago Press.

Timm, N. H. (1975). *Multivariate analysis with applications in education and psychology*. Monterey, CA: Brooks/Cole.

Timm, N. H., & Carlson, J. (1973). *Multivariate analysis of non-orthogonal experimental designs using a multivariate full rank model*. Paper presented at the American Statistical Association Meeting, New York.

Timm, N. H., & Carlson, J. (1975). Analysis of variance through full rank models. *Multivariate behavioral research monographs,* No. *75-1*.

Tracey, N. D., Young, J., C., & Mason, R. L. (1992). Multivariate control charts for individual observations. *Journal of Quality Technology, 2,* 88-95.

Tribus, M., & Sconyi, G. (1989). An alternative view of the Taguchi approach. *Quality Progress*, *22*, 46-48.

Trivedi, P. K., & Pagan, A. R. (1979). Polynomial distributed lags: A unified treatment. *Economic Studies Quarterly, 30,* 37-49.

Tryon, R. C. (1939). *Cluster Analysis*. Ann Arbor, MI: Edwards Brothers.

Tucker, L. R., Koopman, R. F., & Linn, R. L. (1969). Evaluation of factor analytic research procedures by means of simulated correlation matrices. *Psychometrika, 34,* 421-459.

Tufte, E. R. (1983). *The visual display of quantitative information*. Cheshire, CT: Graphics Press.

Tufte, E. R. (1990). *Envisioning information*. Cheshire, CT: Graphics Press.

Tukey, J. W. (1953). *The problem of multiple comparisons*. Unpublished manuscript, Princeton University.

Tukey, J. W. (1962). The future of data analysis. *Annals of Mathematical Statistics, 33,* 1-67.

Tukey, J. W. (1967). An introduction to the calculations of numerical spectrum analysis. In B. Harris (Ed.), *Spectral analysis of time series*. New York: Wiley.

Tukey, J. W. (1972). Some graphic and semigraphic displays. In *Statistical Papers in Honor of George W. Snedecor*; ed. T. A. Bancroft, Arnes, IA: Iowa State University Press, 293-316.

Tukey, J. W. (1977). *Exploratory data analysis*. Reading, MA: Addison-Wesley.

Tukey, J. W. (1984). *The collected works of John W. Tukey*. Monterey, CA: Wadsworth.

Tukey, P. A. (1986). A data analyst's view of statistical plots. *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 21-28.

Tukey, P. A., & Tukey, J. W. (1981). Graphical display of data sets in 3 or more dimensions. In V. Barnett (Ed.), *Interpreting multivariate data*. Chichester, U.K.: Wiley.

Upsensky, J. V. (1937). *Introduction to Mathematical Probability*. New York: McGraw-Hill.

Vale, C. D., & Maurelli, V. A. (1983). Simulating multivariate non-normal distributions. *Psychometrika*, *48*, 465-471.

Vandaele, W. (1983). *Applied time series and Box-Jenkins models*. New York: Academic Press.

Vaughn, R. C. (1974). *Quality control*. Ames, IA: Iowa State Press.

Velicer, W. F., & Jackson, D. N. (1990). Component analysis vs. factor analysis: some issues in selecting an appropriate procedure. *Multivariate Behavioral Research*, *25*, 1-28.

Velleman, P. F., & Hoaglin, D. C. (1981). *Applications, basics, and computing of exploratory data analysis*. Belmont, CA: Duxbury Press.

Von Mises, R. (1941). Grundlagen der Wahrscheinlichkeitsrechnung. *Mathematische Zeitschrift*, *5*, 52-99.

Wainer, H. (1995). Visual revelations. *Chance*, *8*, 48-54.

Wald, A. (1939). Contributions to the theory of statistical estimation and testing hypotheses. *Annals of Mathematical Statistics*, *10*, 299-326.

Wald, A. (1945). Sequential tests of statistical hypotheses. *Annals of Mathematical Statistics*, *16*, 117-186.

Wald, A. (1947). *Sequential analysis*. New York: Wiley.

Walker, J. S. (1991). *Fast Fourier transforms*. Boca Raton, FL: CRC Press.

Wallis, K. F. (1974). Seasonal adjustment and relations between variables. *Journal of the American Statistical Association*, *69*, 18-31.

Wang, C. M., & Gugel, H. W. (1986). High-performance graphics for exploring multivariate data. *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 60-65.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, *58*, 236.

Warner B. & Misra, M. (1996). Understanding Neural Networks as Statistical Tools. *The American Statistician*, *50*, 284-293.

Weatherburn, C. E. (1946). *A First Course in Mathematical Statistics*. Cambridge: Cambridge University Press.

Wei, W. W. (1989). *Time series analysis: Univariate and multivariate methods.* New York: Addison-Wesley.

Weibull, W. (1951). A statistical distribution function of wide applicability. *Journal of Applied Mechanics*, *September*.

Weibull, W., (1939). A statistical theory of the strength of materials. *Ing. Velenskaps Akad. Handl.*, *151*, 1-45.

Weigend, A.S., Rumelhart, D.E. and Huberman, B.A. (1991). Generalization by weight-elimination with application to forecasting. In R.P. Lippmann, J.E. Moody and D.S. Touretzky (Eds.) *Advances in Neural Information Processing Systems 3*, 875-882. San Mateo, CA: Morgan Kaufmann.

Weiss, S. M., & Indurkhya, N. (1997). *Predictive data mining: A practical guide*. New York: Morgan-Kaufman.

Welch, B. L. (1938). The significance of the differences between two means when the population variances are unequal. *Biometrika*, *29*, 350-362.

Welstead, S. T. (1994). *Neural network and fuzzy logic applications in C/C++*. New York: Wiley.

Werbos, P.J. (1974). *Beyond regression: new tools for prediction and analysis in the behavioural sciences.* Ph.D. thesis, Harvard University, Boston, MA.

Wescott, M. E. (1947). Attribute charts in quality control. *Conference Papers, First Annual Convention of the American Society for Quality Control.* Chicago: John S. Swift Co.

Westphal, C., Blaxton, T. (1998). *Data mining solutions*. New York: Wiley.

Wheaton, B., Múthen, B., Alwin, D., & Summers G. (1977). Assessing reliability and stability in panel models. In D. R. Heise (Ed.), *Sociological Methodology.* New York: Wiley.

Wheeler, D. J., & Chambers, D.S. (1986). *Understanding statistical process control*. Knoxville, TN: Statistical Process Controls, Inc.

Wherry, R. J. (1984). *Contributions to correlational analysis*. New York: Academic Press.

Whitney, D. R. (1948). *A comparison of the power of non-parametric tests and tests based on the normal distribution under non-normal alternatives.* Unpublished doctoral dissertation, Ohio State University.

Whitney, D. R. (1951). A bivariate extension of the *U* statistic. *Annals of Mathematical Statistics*, *22*, 274-282.

Widrow, B., and Hoff Jr., M.E. (1960). Adaptive switching circuits. *IRE WESCON Convention Record*, 96-104.

Wiggins, J. S., Steiger, J. H., and Gaelick, L. (1981). Evaluating circumplexity in models of personality. *Multivariate Behavioral Research*, *16*, 263-289.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrica Bulletin*, *1*, 80-83.

Wilcoxon, F. (1947). Probability tables for individual comparisons by ranking methods. *Biometrics*, *3*, 119-122.

Wilcoxon, F. (1949). *Some rapid approximate statistical procedures.* Stamford, CT: American Cyanamid Co.

Wilde, D. J., & Beightler, C. S. (1967). *Foundations of optimization.* Englewood Cliffs, NJ: Prentice-Hall.

Wilks, S. S. (1943). *Mathematical Statistics*. Princeton, NJ: Princeton University Press.

Wilks, S. S. (1946). *Mathematical statistics.* Princeton, NJ: Princeton University Press.

Williams, W. T., Lance, G. N., Dale, M. B., & Clifford, H. T. (1971). Controversy concerning the criteria for taxonometric strategies. *Computer Journal*, *14*, 162.

Wilson, G. A., & Martin, S. A. (1983). An empirical comparison of two methods of testing the significance of a correlation matrix. *Educational and Psychological Measurement*, *43*, 11-14.

Winer, B. J. (1962). *Statistical principles in experimental design*. New York: McGraw-Hill.

Winer, B. J. (1971). *Statistical principles in experimental design* (2nd ed.). New York: McGraw Hill.

Winer, B. J., Brown, D. R., Michels, K. M. (1991). *Statistical principals in experimental design. (3rd ed.).* New York: McGraw-Hill.

Witten, I., H., & Frank, E. (2000). *Data Mining: Practical Machine Learning Tools and Techniques*. New York: Morgan Kaufmann.

Wolfowitz, J. (1942). Additive partition functions and a class of statistical hypotheses. *Annals of Mathematical Statistics, 13*, 247-279.

Wolynetz, M. S. (1979a). Maximum likelihood estimation from confined and censored normal data. *Applied Statistics, 28*, 185-195.

Wolynetz, M. S. (1979b). Maximum likelihood estimation in a linear model from confined and censored normal data. *Applied Statistics, 28*, 195-206.

Wonnacott, R. J., & Wonnacot, T. H. (1970). *Econometrics*. New York: Wiley.

Woodward, J. A., & Overall, J. E. (1975). Multivariate analysis of variance by multiple regression methods. *Psychological Bulletin, 82*, 21-32.

Woodward, J. A., & Overall, J. E. (1976). Calculation of power of the *F* test. *Educational and Psychological Measurement*, *36*, 165-168.

Woodward, J. A., Bonett, D. G., & Brecht, M. L. (1990). *Introduction to linear models and experimental design*. New York: Harcourt, Brace, Jovanovich.

Woodward, J. A., Douglas, G. B., & Brecht, M. L. (1990). *Introduction to linear models and experimental design*. New York: Academic Press.

Yates, F. (1933). The principles of orthogonality and confounding in replicated experiments. *Journal of Agricultural Science*, *23*, 108-145.

Yates, F. (1937). *The Design and Analysis of Factorial Experiments*. Imperial Bureau of Soil Science, Technical Communication No. 35, Harpenden.

Yokoyama, Y., & Taguchi, G. (1975). *Business data analysis: Experimental regression analysis.* Tokyo: Maruzen.

Youden, W. J., & Zimmerman, P. W. (1936). Field trials with fiber pots. *Contributions from Boyce Thompson Institute, 8*, 317-331.

Young, F. W, & Hamer, R. M. (1987). *Multidimensional scaling: History, theory, and applications.* Hillsdale, NJ: Erlbaum

Young, F. W., Kent, D. P., & Kuhfeld, W. F. (1986). Visuals: Software for dynamic hyper-dimensional graphics. *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 69-74.

Younger, M. S. (1985). *A first course in linear regression* (2nd ed.). Boston: Duxbury Press.

Yuen, C. K., & Fraser, D. (1979). *Digital spectral analysis*. Melbourne: CSIRO/Pitman.

Yule, G. U. (1897). On the theory of correlation. *Journal of the Royal Statistical Society, 60*, 812-854.

Yule, G. U. (1907). On the theory of correlation for any number of variables treated by a new system of notation. *Proceedings of the Royal Society*, Ser. A, *79*, 182-193.

Yule, G. U. (1911). *An Introduction to the Theory of Statistics*. London: Griffin.

Zippin, C., & Armitage, P. (1966). Use of concomitant variables and incomplete survival information in the estimation of an exponential survival parameter. *Biometrics*, *22*, 665-672.

Zupan, J. (1982). *Clustering of large data sets*. New York: Research Studies Press.

Zweig, M.H., & Campbell, G. (1993). Receiver-Operating Characteristic (ROC) Plots: A Fundamental Evaluation Tool in Clinical Medicine. *Clin. Chem 39 (4)*, pp. 561-577.

Zwick, W. R., & Velicer, W. F. (1986). Comparison of five rules for determining the number of components to retain. *Psychological Bulletin, 99*, 432-442.