Introduction to Applied Statistics (Draft)

Jerry Brunner

March 17, 2007

Chapter 1

Introduction

This course is about using statistical methods to draw conclusions from real data. It is deliberately non-mathematical, relying on translations of statistical theory into English. For the most part, formulas are avoided. While this involves some loss of precision, it also makes the course accessible to students from non-statistical disciplines (particularly graduate students and advanced undergraduates on their way to graduate school) who need to use statistics in their research. Even for students with strong training in theoretical statistics, the use of plain English can help reveal the connections between theory and applications, while also suggesting a useful way to communicate with non-statisticians.

We will avoid mathematics, but we will not avoid computers. Learning to apply statistical methods to real data involves actually doing it, and the use of software is not optional. Furthermore, we will *not* employ "user-friendly" menu-driven statistical programs. Why?

- It's just too easy to poke around in the menus trying different things, produce some results that seem reasonable, and then two weeks later be unable to say exactly what one did.

- Real data sets tend to be large and complex, and most statistical analyses involve a sizable number of operations. If you discover a tiny mistake after you produce your results, you don't want to go back and repeat two hours of menu selections and mouse clicks, with one tiny variation.

- If you need to analyze a data set that is similar to one you have analyzed in the past, it's a lot easier to edit a program than to remember a collection of menu selections from last year.

Don't worry! The word "program" does *not* mean we are going to write programs in some true programming language like C or Java. We'll use statistical software in which most of the actual statistical procedures have already been written by experts; usually, all we have to do is invoke them by using high-level commands.

The statistical packages we will use in this course are `SAS` and `R`. These packages are command-oriented rather than menu-oriented, and are very powerful. They are industrial strength tools, and will be illustrated in an industrial strength environment — `unix`. This is mostly for local convenience. There are Windows versions of both `SAS` and `R` that work just as well as the unix versions, except for very big jobs.

Applied Statistics really refers to two related enterprises. The first might be more accurately termed "Applications of Statistics," and consists of the appropriate application of standard general techniques. The second enterprise is the development of specialized techniques that are designed specifically for the data at hand. The difference is like buying your clothes from Walmart versus sewing them yourself (or going to a tailor). In this course, we will do both. We'll maintain the non-mathematical nature of the course in the second half by substituting computing power and random number generation for statistical theory.

1.1 Vocabulary of data analysis

We start with a **data file**. Think of it as a rectangular array of numbers, with the rows representing **cases** (units of analysis, observations, subjects, replicates) and the columns representing **variables** (pieces of information available for each case).

- A physical data file might have several lines of data per case, but you can imagine them listed on a single long line.

- Data that are *not* available for a particular case (for example because a subject fails to answer a question, or because a piece of measuring equipment breaks down) will be represented by missing value codes.

Missing value codes allow observations with missing information to be automatically excluded from a computation.

- Variables can be **quantitative** (representing amount of something) or **categorical**. In the latter case the "numbers" are codes representing category membership. Categories may be **ordered** (small vs. medium vs. large) or **unordered** (green vs. blue vs. yellow). When a quantitative variable reflects measurement on a scale capable of very fine gradation, it is sometimes described as **continuous**. Some statistical texts use the term **qualitative** to mean categorical. When an anthropologist uses the word "qualitative," however, it usually refers to ethnographic or case study research in which data are not explicitly assembled into a data file.

Another very important way to classify variables is

Independent Variable (IV): Predictor = X (actually $X_i, i = 1, \ldots, n$)

Dependent Variable (DV): Predicted = Y (actually $Y_i, i = 1, \ldots, n$)

Example: X = weight of car in kilograms, Y = fuel efficiency in litres per kilometer

Sample Question 1.1.1 *Why isn't it the other way around?*

Answer to Sample Question 1.1.1 *Since weight of a car is a factor that probably influences fuel efficiency, it's more natural to think of predicting fuel efficiency from weight.*

The general principle is that if it's more natural to think of predicting A from B, then A is the dependent variable and B is the independent variable. This will usually be the case when B is thought to cause or influence A. Sometimes it can go either way or it's not clear. But usually it's easy to decide.

Sample Question 1.1.2 *Is it possible for a variable to be both quantitative and categorical? Answer Yes or No, and either give an example or explain why not.*

Answer to Sample Question 1.1.2 *Yes. For example, the number of cars owned by a person or family.*

In some fields, you may hear about **nominal, ordinal, interval** and **ratio** variables, or variables measured using "scales of measurement" with those names. Ratio means the scale of measurement has a true zero point, so that a value of 4 represents twice as much as 2. An interval scale means that the difference (interval) between 3 and 4 means the same thing as the difference between 9 and 10, but zero does not necessarily mean absence of the thing being measured. The usual examples are shoe size and ring size. In ordinal measurement, all you can tell is that 6 is less than 7, not how much more. Measurement on a nominal scale consists of the assignment of unordered categories. For example, citizenship is measured on a nominal scale.

It is usually claimed that one should calculate means (and therefore, for example, do multiple regression) only with interval and ratio data; it's usually acknowledged that people do it all the time with ordinal data, but they really shouldn't. And it is obviously crazy to calculate a mean on numbers representing unordered categories. Or is it?

Sample Question 1.1.3 *Give an example in which it's meaningful to calculate the mean of a variable measured on a nominal scale.*

Answer to Sample Question 1.1.3 *Code males as zero and females as one. The mean is the proportion of females.*

It's not obvious, but actually all this talk about what you should and shouldn't do with data measured on these scales does not have anything to do with *statistical* assumptions. That is, it's not about the mathematical details of any statistical model. Rather, it's a set of guidelines for what statistical model one ought to adopt. Are the guidelines reasonable? It's better to postpone further discussion until after we have seen some details of multiple regression.

1.2 Statistical significance

We will often pretend that our data represent a **random sample** from some **population**. We will carry out formal procedures for making inferences about this (usually fictitious) population, and then use them as a basis for drawing conclusions from the data.

Why do we do all this pretending? As a formal way of filtering out things that happen just by coincidence. The human brain is organized to find *meaning* in what it perceives, and it will find apparent meaning even in a sequence of random numbers. The main purpose of testing for statistical significance is to protect Science against this. Even when the data do not fully satisfy the assumptions of the statistical procedure being used (for example, the data are not really a random sample) significance testing can be a useful as a way of restraining scientists from filling the scientific literature with random garbage. This is such an important goal that we will spend a substantial part of the course on significance testing.

1.2.1 Definitions

Numbers that can be calculated from sample data are called **statistics**. Numbers that could be calculated if we knew the whole population are called **parameters**. Usually parameters are represented by Greek letters such as α, β and γ, while statistics are represented by ordinary letters such as a, b, c. Statistical inference consists of making decisions about parameters based on the values of statistics.

The **distribution** of a variable corresponds roughly to a histogram of the values of the variable. In a large population for a variable taking on many values, such a histogram will be indistinguishable from a smooth curve.

For each value x of the independent variable X, in principle there is a separate distribution of the dependent variable Y. This is called the **conditional distribution** of Y given $X = x$.

We will say that the independent and dependent variables are **unrelated** if the *conditional distribution of the dependent variable is identical for each value of the independent variable*. That is, the histogram of the dependent variable does not depend on the value of the independent variable. If the distribution of the dependent variable does depend on the value of the independent variable, we will describe the two variables as **related**. All this applies to sample as well as population data-sets (a population data set may be entirely hypothetical).

Most research questions involve more than one independent variable. It is also common to have more than one dependent variable. When there is one dependent variable, the analysis is called **univariate**. When more than one dependent variable is being considered simultaneously, the analysis is called **multivariate**.

Sample Question 1.2.1 *Give an example of a study with two categorical independent variables, one quantitative independent variable, and two quantitative dependent variables.*

Answer to Sample Question 1.2.1 *In a study of success in university, the subjects are first-year university students. The categorical independent variables are Sex and Immigration Status (Citizen, Permanent Resident or Visa), and the quantitative independent variable is family income. The dependent variables are cumulative Grade Point Average at the end of first year, and number of credits completed in first year.*

Many problems in data analysis reduce to asking whether one or more variables are related – not in the actual data, but in some hypothetical population from which the data are assumed to have been sampled. The reasoning goes like this. Suppose that the independent and dependent variables are actually unrelated *in the population*. If this **null hypothesis** is true, what is the probability of obtaining a *sample* relationship between the variables that is as strong or stronger than the one we have observed? If the probability is small (say, $p < 0.05$), then we describe the sample relationship as **statistically significant**, and it is socially acceptable to discuss the results. In particular, there is some chance of having the results taken seriously enough to publish in a scientific journal.

The number 0.05 is called the **significance level**. In principle, the exact value of the significance level is arbitrary as long as it is fairly small, but scientific practice has calcified around a suggestion of R. A. Fisher (in whose honour the F-test is named), and the 0.05 level is an absolute rule in many journals in the social and biological sciences.

We will willingly conform to this convention. We conform *willingly* because we understand that scientists can be highly motivated to get their results into print, even if those "results" are just trends that could easily be random noise. To restrain these people from filling the scientific literature with random garbage, we need a clear rule.

For those who like precision, the formal definition of a p-value is this. It is the minimum significance level α at which the null hypothesis (of no relationship between IV and DV in the population) can be rejected.

Here is another useful way to talk about p-values. *The p-value is the probability of getting our results (or better) just by chance.* If p is small enough, then the data are very unlikely to have arisen by chance, assuming there is

really no relationship between the independent variable and the dependent variable in the population. In this case we will conclude there really is a relationship between the independent variable and the dependent variable.

What should we do if $p > .05$? Fisher suggested that we should not conclude anything. In particular, he suggested that we should *not* conclude that the independent and dependent variables are unrelated. Instead, we can say that there is insufficient evidence evidence of a relationship between the independent variable and the dependent variable. A good reference is Fisher's masterpiece, *Statistical methods for research workers* [5], which had its first edition in 1925, and its 14th and last edition in 1970, eight years after Fisher's death.

The trouble with Fisher's formulation is that it never allows us to conclude that the null hypothesis is true. But sometimes, experimental treatments just don't do anything, and it is of scientific and practical importance to be able to say so. For example, medical researchers frequently conclude that drugs don't work. On what basis are they drawing these conclusions? On what basis *should* they draw such conclusions? We will get back to this important issue later. For now, let us agree that if a test is not significant, then we certainly can agree with Fisher that there is not enough evidence to conclude that the independent and dependent variables are related. As for concluding that the variables are *not* related, we don't yet have a formal rule.

1.2.2 Standard elementary significance tests

We will now consider some of the most common elementary statistical methods. For each one, you should be able to answer the following questions.

1. Make up your own original example of a study in which the technique could be used.
2. In your example, what is the independent variable (or variables)?
3. In your example, what is the dependent variable (or variables)?
4. Indicate how the data file would be set up.

Independent observations One assumption shared by most standard methods is that of *"independent observations."* The meaning of the assumption is this. Observations 13 and 14 are independent if and only if the conditional distribution of observation 14 given observation 13 is the same for each possible value observation 13. For example if the observations are temperatures on consecutive days, this would not hold. If the dependent variable is score on a homework assignment and students copy from each other, the observations will not be independent.

When significance testing is carried out under the assumption that observations are independent but really they are not, results that are actually due to chance will often be detected as significant with probability considerably greater than 0.05. This is sometimes called the problem of *inflated* n. In other words, you are pretending you have more separate pieces of information than you really do. When observations cannot safely be assumed independent, this should be taken into account in the statistical analysis. We will return to this point again and again.

Independent (two-sample) t-test

This is a test for whether the means of two independent groups are different. Assumptions are independent observations, normality within groups, equal variances. For large samples normality does not matter. For large samples with nearly equal sample sizes, equal variance assumption does not matter. The assumption of independent observations is always important.

Sample Question 1.2.2 *Make up your own original example of a study in which a two-sample t-test could be used.*

Answer to Sample Question 1.2.2 *An agricultural scientist is interested in comparing two types of fertilizer for potatoes. Fifteen small plots of ground receive fertilizer A and fifteen receive fertilizer B. Crop yield for each plot in pounds of potatoes harvested is recorded.*

Sample Question 1.2.3 *In your example, what is the independent variable (or variables)?*

Answer to Sample Question 1.2.3 *Fertilizer, a binary variable taking the values A and B.*

Sample Question 1.2.4 *In your example, what is the dependent variable (or variables)?*

Answer to Sample Question 1.2.4 *Crop yield in pounds.*

Sample Question 1.2.5 *Indicate how the data file might be set up.*

Answer to Sample Question 1.2.5

A	13.1
A	11.3
⋮	⋮
B	12.2
⋮	⋮

Matched (paired) t-test

Again comparing two means, but from paired observations. Pairs of observations come from the same case (subject, unit of analysis), and presumably are non-independent. Again, the data from a given pair are not really separate pieces of information, and if you pretend they are, then you are pretending to have more accurate estimation of population parameters — and a more sensitive test — than you really do. The probability of getting results that are statistically significant will be greater than 0.05, even if nothing is going on.

In a matched t-test, this problem is taken care of by computing a difference for each pair, reducing the volume of data (and the apparent sample size) by half. This is our first example of a *repeated measures* analysis. Here is a general definition. We will say that there are **repeated measures** on an independent variable if a case (unit of analysis, subject, participant in the study) contributes a value of the dependent variable for each value of the independent variable in question. A variable on which there are repeated measures is sometimes called a **within-subjects** variable. When this language is being spoken, variables on which there are not repeated measures are called **between-subjects**.

The assumptions of the matched t-test are that the differences represent independent observations from a normal population. For large samples, normality does not matter. The assumption that different cases represent independent observations is always important.

Sample Question 1.2.6 *Make up your own original example of a study in which a matched t-test could be used.*

Answer to Sample Question 1.2.6 *Before and after a 6-week treatment, participants in a quit-smoking program were asked "On the average, how many cigarettes do you smoke each day?"*

Sample Question 1.2.7 *In your example, what is the independent variable (or variables)?*

Answer to Sample Question 1.2.7 *Presence versus absence of the program, a binary variable taking the values "Absent" or "Present" (or maybe "Before" and "After"). We can say there are* repeated measures *on this factor, or that it is a* within-subjects *factor.*

Sample Question 1.2.8 *In your example, what is the dependent variable (or variables)?*

Answer to Sample Question 1.2.8 *Reported number of cigarettes smoked per day.*

Sample Question 1.2.9 *Indicate how the data file might be set up.*

Answer to Sample Question 1.2.9 *The first column is "Before," and the second column is "After."*

$$\begin{array}{cc} 22 & 18 \\ 40 & 34 \\ 20 & 10 \\ \vdots & \vdots \end{array}$$

One-way Analysis of Variance

Extension of the independent t-test to two or more groups. Same assumptions, everything. $F = t^2$ for two groups.

Sample Question 1.2.10 *Make up your own original example of a study in which a one-way analysis of variance could be used.*

Answer to Sample Question 1.2.10 *Eighty branches of a large bank were chosen to participate in a study of the effect of music on tellers' work behaviour. Twenty branches were randomly assigned to each of the following 4 conditions. 1=No music, 2=Elevator music, 3=Rap music, 4=Individual choice (headphones). Average customer satisfaction and worker satisfaction were assessed for each bank branch, using a standard questionnaire.*

Sample Question 1.2.11 *In your example, what are the cases?*

Answer to Sample Question 1.2.11 *Branches, not people answering the questionnaire.*

Sample Question 1.2.12 *Why do it that way?*

Answer to Sample Question 1.2.12 *To avoid serious potential problems with independent observations within branches. The group of interacting people within social setting is the natural unit of analysis, like an organism.*

Sample Question 1.2.13 *In your example, what is the independent variable (or variables)?*

Answer to Sample Question 1.2.13 *Type of music, a categorical variable taking on 4 values.*

Sample Question 1.2.14 *In your example, what is the dependent variable (or variables)?*

Answer to Sample Question 1.2.14 *There are 2 dependent variables, average customer satisfaction and average worker satisfaction. If they were analyzed simultaneously the analysis would be multivariate (and not elementary).*

Sample Question 1.2.15 *Indicate how the data file might be set up.*

Answer to Sample Question 1.2.15 *The columns correspond to Branch, Type of Music, Customer Satisfaction and Worker Satisfaction*

$$\begin{array}{cccc} 1 & 2 & 4.75 & 5.31 \\ 2 & 4 & 2.91 & 6.82 \\ \vdots & \vdots & \vdots & \vdots \\ 80 & 2 & 5.12 & 4.06 \end{array}$$

Sample Question 1.2.16 *How could this be made into a repeated measures study?*

Answer to Sample Question 1.2.16 *Let each branch experience each of the 4 music conditions in a random order (or better, use only 72 branches, with 3 branches receiving each of the 24 orders). There would then be 10 pieces of data for each bank: Branch, Order (a number from 1 to 24), and customer satisfaction and worker satisfaction for each of the 4 conditions.*

Including all orders of presentation in each experimental condition is an example of **counterbalancing** — that is, presenting stimuli in such a way that order of presentation is unrelated to experimental condition. That way, the effects of the treatments are not confused with fatigue or practice effects (on the part of the experimenter as well as the subjects). In counterbalancing, it is often not feasible to include *all* possible orders of presentation it each experimental condition, because sometimes there are two many. The point is that order of presentation has to be unrelated to any manipulated independent variable.

Two (and higher) way Analysis of Variance

Extension of One-Way ANOVA to allow assessment of the joint relationship of several categorical independent variables to one quantitative dependent variable that is assumed normal within treatment combinations. Tests for interactions between IVs are possible. An interaction means that the relationship of one independent variable to the dependent variable *depends* on the value of another independent variable. More on this later.

Crosstabs and chi-squared tests

Cross-tabulations (Crosstabs) are joint frequency distribution of two categorical variables. One can be considered an IV, the other a DV if you like. In any case (even when the IV is manipulated in a true experimental study) we will test for significance using the *chi-squared test of independence*. Assumption is independent observations are drawn from a multinomial distribution. Violation of the independence assumption is common and very serious.

Sample Question 1.2.17 *Make up your own original example of a study in which this technique could be used.*

Answer to Sample Question 1.2.17 *For each of the prisoners in a Toronto jail, record the race of the offender and the race of the victim. This is illegal; you could go to jail for publishing the results. It's totally unclear which is the IV and which is the DV, so I'll make up another example.*

For each of the graduating students from a university, record main field of study and and gender of the student (male or female).

Sample Question 1.2.18 *In your example, what is the independent variable (or variables)?*

Answer to Sample Question 1.2.18 *Gender*

Sample Question 1.2.19 *In your example, what is the dependent variable (or variables)?*

Answer to Sample Question 1.2.19 *Main field of study (many numeric codes).*

Sample Question 1.2.20 *Indicate how the data file would be set up.*

Answer to Sample Question 1.2.20 *The first column is Gender (0=Male, 1=F). The second column is Field.*

$$\begin{array}{cc} 1 & 2 \\ 0 & 14 \\ 0 & 9 \\ \vdots & \vdots \end{array}$$

Correlation and Simple Regression

Correlation Start with a **scatterplot** showing the association between two (quantitative, usually continuous) variables. A scatterplot is a set of Cartesian coordinates with a dot or other symbol showing the location of each (x, y) pair. If one of the variables is clearly the independent variable, it's traditional to put it on the x axis. There are n points on the scatterplot, where n is the number of cases in the data file.

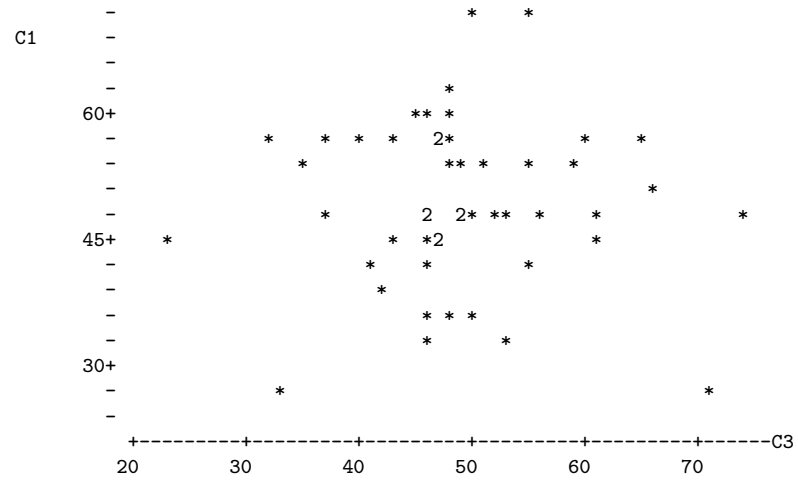
Often, the points in a scatterplot cluster around a straight line. The correlation coefficient (Pearson's r) expresses the extent to which the points cluster tightly around a straight line.

Here are some properties of the correlation coefficient r:

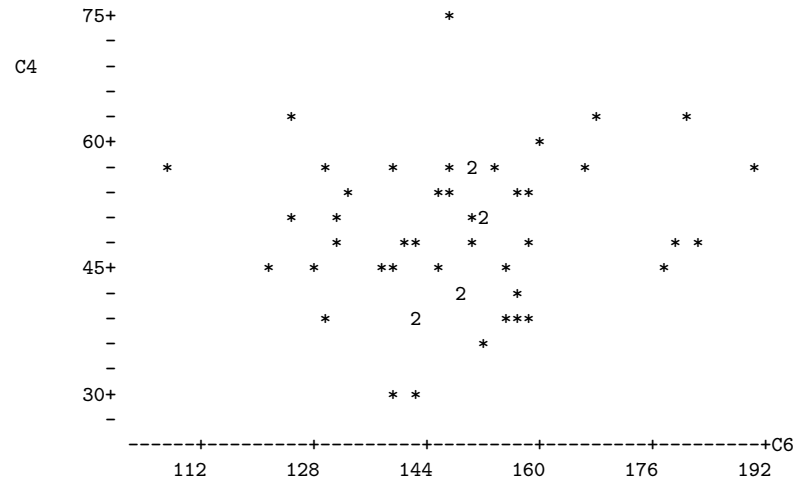
- $-1 \leq r \leq 1$
- $r = +1$ indicates a perfect positive linear relationship. All the points are exactly on a line with a positive slope.
- $r = -1$ indicates a perfect negative linear relationship. All the points are exactly on a line with a negative slope.
- $r = 0$ means no *linear* relationship (curve possible)
- r^2 represents explained variation, reduction in (squared) error of prediction. For example, the correlation between scores on the Scholastic Aptitude Test (SAT) and first-year grade point average (GPA) is around +0.50, so we say that SAT scores explain around 25% of the variation in first-year GPA.

The test of significance for Pearson's r assumes a bivariate normal distribution for the two variables; this means that the only possible relationship between them is linear. As usual, the assumption of independent observations is always important.

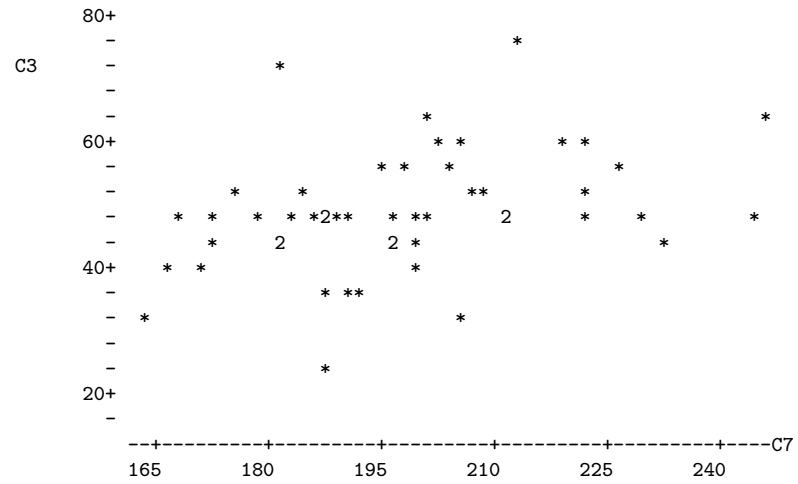
Here are some examples of scatterplots and the associated correlation coefficients. The number 2 on a plot means that two points are on top of each other, or at least too close to be distinguished in this crude line printer graphic.



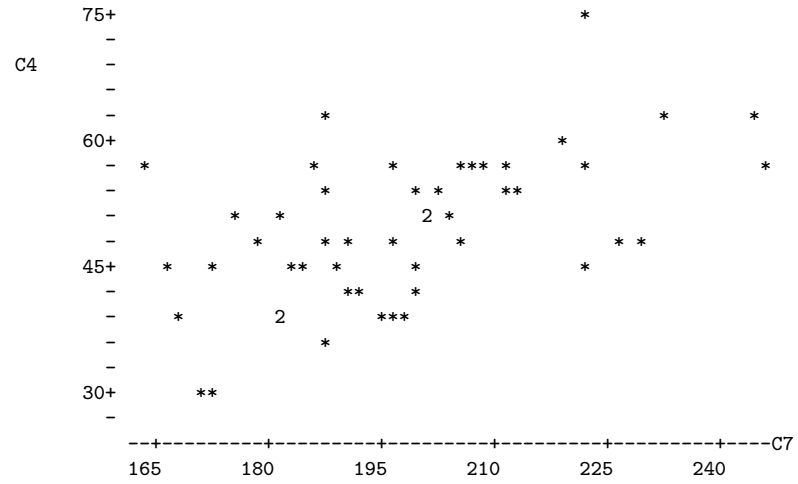
Correlation of C1 and C3 = 0.004



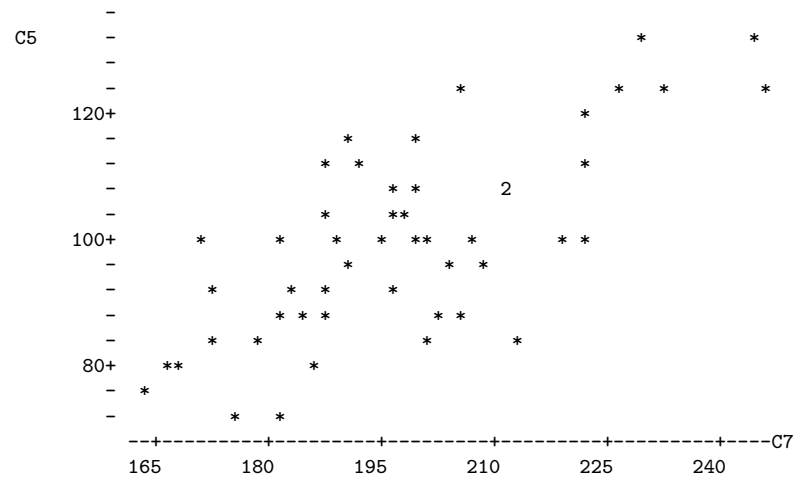
Correlation of C4 and C6 = 0.112



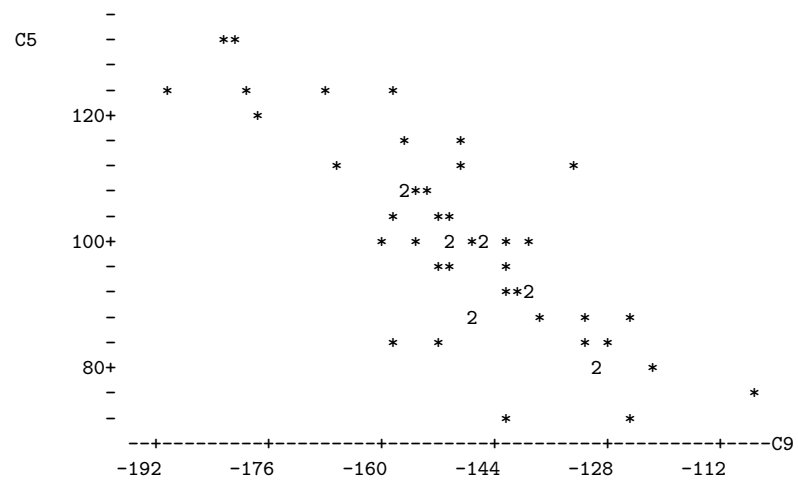
Correlation of C3 and C7 = 0.368



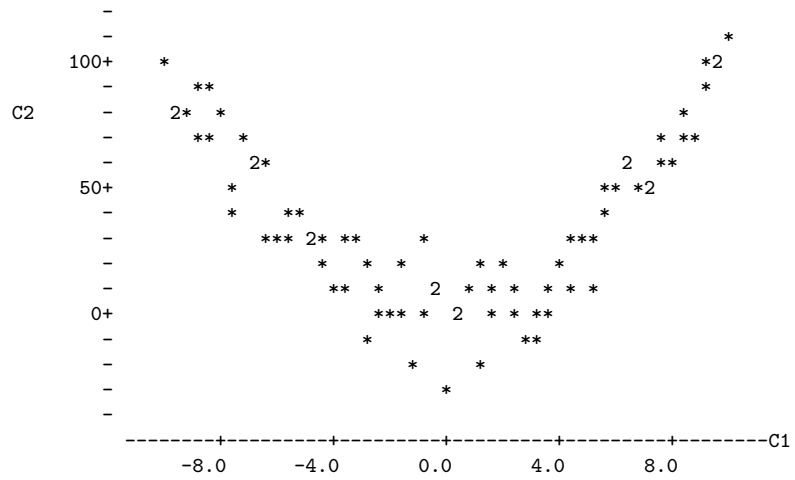
Correlation of C4 and C7 = 0.547



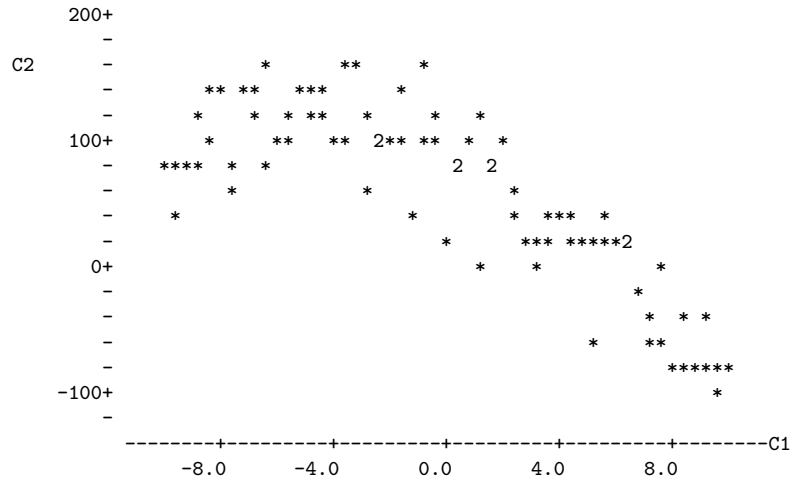
Correlation of C5 and C7 = 0.733



Correlation of C5 and C9 = -0.822



Correlation of C1 and C2 = 0.025



Correlation of C1 and C2 = -0.811

Simple Regression One independent variable, one dependent. In the usual examples both are quantitative (continuous). We fit a **least-squares** line to the cloud of points in a scatterplot. The least-squares line is the unique line that minimizes the sum of squared vertical distances between the line and the points in the scatterplot. That is, it minimizes the total

(squared) error of prediction.

Denoting the slope of the least-squares line by b_1 and the intercept of the least-squares line by b_0,

$$b_1 = r\frac{s_y}{s_x} \text{ and } b_0 = \overline{Y} - b_1\overline{X}.$$

That is, the slope of the least squares has the same sign as the correlation coefficient, and equals zero if and only if the correlation coefficient is zero.

Usually, you want to test whether the slope is zero. This is the same as testing whether the correlation is zero, and mercifully yields the same p-value. Assumptions are independent observations (again) and that within levels of the IV, the DV has a normal distribution with the same variance (variance does not depend on value of the DV). Robustness properties are similar to those of the 2-sample t-test. The assumption of independent observations is always important.

Multiple Regression

Regression with several independent variables at once; we're fitting a (hyper) plane rather than a line. Multiple regression is very flexible; all the other techniques mentioned above (except the chi-squared test) are special cases of multiple regression. More details later.

1.3 Experimental versus observational studies

Why might someone want to predict a dependent variable from an independent variable? There are two main reasons.

- There may be a practical reason for prediction. For example, a company might wish to predict who will buy a product, in order to maximize the productivity of its sales force. Or, an insurance company might wish to predict who will make a claim, or a university computer centre might wish to predict the length of time a type of hard drive will last before failing. In each of these cases, there will be some independent variables that are to be used for prediction, and although the people doing the study may be curious and may have some ideas

about how things might turn out and why, they don't really care why it works, as long as they can predict with some accuracy. Does variation in the IV *cause* variation in the DV? Who cares?

- This may be science (of some variety). The goal may be to understand how the world works — in particular, to understand the dependent variable. In this case, most likely we are implicitly or explicitly thinking of a causal relationship between the IV and DV. Think of attitude similarity and interpersonal attraction ....

Sample Question 1.3.1 *A study finds that high school students who have a computer at home get higher grades on average than students who do not. Does this mean that parents who can afford it should buy a computer to enhance their children's chances of academic success?*

Here is an answer that gets **zero** points. "Yes, with a computer the student can become computer literate, which is a necessity in our competitive and increasingly technological society. Also the student can use the computer to produce nice looking reports (neatness counts!), and obtain valuable information on the World Wide Web." **ZERO**.

The problem with this answer is that while it makes some fairly reasonable points, it is based on personal opinion, and fails to address the real question, which is "**Does this mean** ..." Here is an answer that gets full marks.

Answer to Sample Question 1.3.1 *Not necessarily. While it is possible that some students are doing better academically and therefore getting into university because of their computers, it is also possible that their parents have enough money to buy them a computer, and also have enough money to pay for their education. It may be that an academically able student who is more likely to go to university will want a computer more, and therefore be more likely to get one somehow. Therefore, the study does not provide good evidence that a computer at home will enhance chances of academic success.*

Note that in this answer, the *focus is on whether the study provides good evidence* for the conclusion, not whether the conclusion is reasonable on other grounds. And the answer gives *specific alternative explanations* for the results as a way of criticizing the study. If you think about it, suggesting plausible alternative explanations is a very damaging thing to say about any empirical study, because you are pointing out that the investigators expended

a huge amount of time and energy, but didn't establish anything conclusive. Also, suggesting alternative explanations is extremely valuable, because that is how research designs get improved and knowledge advances.

Now here are the general principles. If X and Y are measured at roughly the same time, X could be causing Y, Y could be causing X, or there might be some third variable (or collection of variables) that is causing both X and Y. Therefore we say that "Correlation does not necessarily imply causation." Here, by correlation we mean association (lack of independence) between variables. It is not limited to situations where you would compute a correlation coefficient.

A **confounding variable** is a variable not included as an independent variable, that might be related to both the independent variable and the dependent variable – and that might therefore create a seeming relationship between them where none actually exists, or might even hide a relationship that is present. Some books also call this a "lurking variable." You are responsible for the vocabulary "confounding variable."

An **experimental study** is one in which cases are randomly assigned to the different values of an independent variable (or variables). An **observational study** is one in which the values of the independent variables are not randomly assigned, but merely observed.

Some studies are purely observational, some are purely experimental, and many are mixed. It's not really standard terminology, but in this course we will describe independent *variables* as experimental (i.e., randomly assigned, manipulated) or observed.

In an experimental study, there is no way the dependent variable could be causing the independent variable, because values of the IV are assigned by the experimenter. Also, it can be shown (using the Law of Large Numbers) that when units of observation are randomly assigned to values of an IV, all potential confounding variables are cancelled out as the sample size increases. This is very wonderful. You don't even have to know what they are!

Sample Question 1.3.2 *Is it possible for a continuous variable to be experimental, that is, randomly assigned?*

Answer to Sample Question 1.3.2 *Sure. In a drug study, let one of the independent variables consist of n equally spaced dosage levels spanning some range of interest, where n is the sample size. Randomly assign one participant to each dosage level.*

Sample Question 1.3.3 *Give an original example of a study with one quantitative observed independent variable and one categorical manipulated independent variable. Make the study multivariate, with one dependent variable consisting of unordered categories and two quantitative dependent variables. categorical*

Answer to Sample Question 1.3.3 *Stroke patients in a drug study are randomly assigned to either a standard blood pressure drug or one of three experimental blood pressure drugs. The categorical dependent variable is whether the patient is alive or not 5 years after the study begins. The quantitative dependent variables are systolic and diastolic blood pressure one week after beginning drug treatment.*

In practice, of course there would be a lot more variables; but it's still a good answer.

Because of possible confounding variables, only an experimental study can provide good evidence that an independent variable *causes* a dependent variable. Words like effect, affect, leads to etc. imply claims of causality and are only justified for experimental studies.

Sample Question 1.3.4 *Design a study that could provide good evidence of a causal relationship between having a computer at home and academic success.*

Answer to Sample Question 1.3.4 *High school students without computers enter a lottery. The winners (50% of the sample) get a computer and modem to use at home. The dependent variable is whether or not the student enters university.*

Sample Question 1.3.5 *Is there a problem with independent observations here? Can you fix it?*

Answer to Sample Question 1.3.5 *Oops. Yes. Students who win may be talking to each other, sharing software, etc.. Actually, the losers will be communicating too. Therefore their behaviour is non-independent and standard significance tests will be invalid. One solution is to hold the lottery in n separate schools, with one winner in each school. If the dependent variable were GPA, we could do a matched t-test comparing the performance of the winner to the average performance of the losers.*

Sample Question 1.3.6 *What if the DV is going to university or not?*

Answer to Sample Question 1.3.6 *We are getting into deep water here. Here is how I would do it. In each school, give a score of "1" to each student who goes to university, and a "0" to each student who does not. Again, compare the scores of the winners to the average scores of the losers in each school using a matched t-test. Note that the mean difference that is to be compared with zero here is the mean difference in probability of going to university, between students who get a computer to use and those who do not. While the differences for each school will not be normally distributed, the central limit theorem tells us that the mean difference will be approximately normal if there are more than about 20 schools, so the t-test is valid. In fact, the t-test is conservative, because the tails of the t distribution are heavier than those of the standard normal. This answer is actually beyond the scope of the present course.*

Artifacts and Compromises

Random assignment to experimental conditions will take care of confounding variables, but only if it is done right. It is amazingly easy for for confounding variables to sneak back into a true experimental study through defects in the procedure. For example, suppose you are interested in studying the roles of men and women in our society, and you have a 50-item questionnaire that (you hope) will measure traditional sex role attitudes on a scale from 0 = Very Non-traditional to 50 = Very Traditional. However, you suspect that the details of how the questionnaire is administered could have a strong influence on the results. In particular, the sex of the person administering the questionnaire and how he or she is dressed could be important.

Your subjects are university students, who must participate in your study in order to fulfill a course requirement in Introductory Psychology. You randomly assign your subjects to one of four experimental conditions: Female research assistant casually dressed, Female research assistant formally dressed, Male research assistant casually dressed, or Male research assistant formally dressed. Subjects in each experimental condition are instructed to report to a classroom at a particular time, and they fill out the questionnaire sitting all together.

This is an appealing procedure from the standpoint of data collection, because it is fast and easy. However, it is so flawed that it may be a complete waste of time to do the study at all. Here's why. Because subjects are

run in four batches, an unknown number of confounding variables may have crept back into the study. To name a few, subjects in different experimental conditions will be run at different times of day or different days of the week. Suppose subjects in the the male formally dressed condition fill out the questionnaire at 8 in the morning. Then *all* the subjects in that condition are exposed to the stress and fatigue of getting up early, as well as the treatment to which they have been randomly assigned.

There's more, of course. Presumably there are just two research assistants, one male and one female. So there can be order effects; at the very least, the lab assistant will be more practiced the second time he or she administers the questionnaire. And, though the research assistants will surely try to administer the questionnaire in a standard way, do you really believe that their body language, facial expressions and tone of voice will be identical both times?

Of course, the research assistants know what condition the subjects are in, they know the hypotheses of the study, and they probably have a strong desire to please the boss — the investigator (professor or whatever) who is directing this turkey, uh, excuse me, I mean this research. Therefore, their behaviour could easily be slanted (perhaps unconsciously so) to produce the hypothesized effects.

This kind phenomenon is well-documented. It's called *experimenter expectancy*. Experimenters find what they expect to find. If they are led to believe that certain mice are very intelligent, then those mice will do better on all kinds of learning tasks, even though in fact the mice were randomly assigned to be labeled as "intelligent." This kind of thing applies all the way down to flatworms. The classic reference is Robert Rosenthal's *Experimenter expectancy in behavioral research* [11]. Naturally, the expectancy phenomenon applies to teachers and students in a classroom setting, where it is called *teacher expectancy*. The reference for this is Rosenthal and Jacobson's *Pygmalion in the classroom* [12].

It is wrong (and complacent) to believe that expectancy effects are confined to psychological research. In medicine, *placebo effects* are well-documented. Patients who are given an inert substance like a sugar pill do better than patients who are not, provided that they (or their doctors) believe that they are getting medicine that works. Is it the patients' expectancies that matter, or the doctors'? Probably both. The standard solution, and the *only* acceptable solution in clinical trials of new drugs, is the so called *double blind*, in which subjects are randomly assigned to receive either the drug or a placebo,

and neither the patient nor the doctor knows which it is. This is the gold standard. Accept no substitutes.

Until now, we have been discussing threats to the *Internal Validity* of research. A study has good internal validity if it's designed to eliminate the influence of confounding variables, so one can be reasonably sure that the observed effects really are being produced by the independent variables of interest. But there's also *External Validity*. External validity refers to how well the phenomena outside the laboratory or data-collection situation are being represented by the study. For example, well-controlled, double-blind taste tests indicated that the Coca-cola company had a recipe that consumers liked better than the traditional one. But attempts to market "New" Coke were an epic disaster. There was just more going on in the real world of soft drink consumption than in the artificial laboratory setting of a taste test. Cook and Campbell's *Quasi-experimentation* [3] contains an excellent discussion of internal versus external validity.

In Industrial-Organizational psychology, we have the *Hawthorne Effect*, which takes its name from the Hawthorne plant of General Electric, where some influential studies of worker productivity were carried out in the 1930's. The basic idea is that when workers know that they are part of a study, almost anything you do will increase productivity. Make the lights brighter? Productivity increases. Make the lights dimmer? Productivity increases. This is how the Hawthorne Effect is usually described. The actual details of the studies and their findings are more complex [10], but the general idea is that when people know they are participating in a study, they tend to feel more valued, and act accordingly. In this respect, the fact that the subjects know that a study is being carried can introduce a serious distortion into the way things work, and make the results unrepresentative of what normally happens.

Medical research on non-human animals is always at least subject to discussion on grounds of external validity, as is almost any laboratory research in Psychology. Do you know why the blood vessels running away from the heart are called "arteries?" It's because they were initially thought to contain air. Why? Because medical researchers were basing their conclusions entirely on dissections of dead bodies. In live bodies, the arteries are full of blood.

Generally speaking, the controlled environments that lead to the best internal validity also produce the greatest threats to external validity. Is a given laboratory setup capturing the essence of the phenomena under con-

sideration, or is it artificial and irrelevant? It's usually hard to tell. The best way to make an informed judgement is to compare laboratory studies and field studies that are trying to answer the same questions. The laboratory studies usually have better internal validity, and the field studies usually have better external validity. When the results are consistent, we feel more comfortable.

Chapter 2

First set of tools: SAS running under unix (including linux)

The SAS language is the same regardless of what hardware you use or what operating system is running on the hardware. SAS programs are simple text files that can be transported from one machine to another with minimal difficulty. In this course, everything will be illustrated with SAS running under the unix operating system, but it's not a problem even if the next place you go only has PCs. The adjustment to SAS-PC should be fast and fairly painless.

2.1 Unix

Unix is a line-oriented operating system. Well, there's X-windows (a graphical shell that runs on top of unix), but we won't bother with it. Basically, you type a command, press Enter, and unix does something for (or to) you. It may help to think of unix as DOS on steroids, if you remember DOS. The table below has all the unix commands you will need for this course. Throughout, *fname* stands for the name of a file.

A Minimal Set of unix Commands

exit Logs you off the system: ALWAYS log off before leaving!

passwd Lets you change your password. Recommended.

man *command name* Online help: explains *command name*, (like `man sort`).

ls Lists names of the files in your directory.

less *fname* Displays fname on screen, one page at a time. Spacebar for next page, `q` to quit.

lpr *fname* Prints hard copy on a laser printer. lpr stands for line printer. These physical devices no longer exist in most installations.

rm *fname* Removes *fname*, erasing it forever.

cp *fname1 fname2* Makes a copy of *fname1*. The new copy is named *fname2*.

mv *fname1 fname2* Moves (renames) *fname1*

emacs *fname* Starts the `emacs` text editor, editing *fname* (can be new file).

R Gets you into the `R` implementation of the S environment.

sas *fname* Executes `SAS` commands in the file *fname*`.sas`, yielding *fname*`.log` and (if no fatal errors) *fname*`.lst`.

ps Shows active processes

kill -9 # Kills process (job) number `#`. Sometimes you must do this when you can't log off because there are stopped jobs. Use `ps` to see the job numbers.

mail yourname@yourisp.com < *fname* Email a file to yourself. Very handy for getting files to your home computer for printing.

curl *URL* > *fname* A *URL* is a Web address. This command is intended to help you get a copy of the source code of Web pages. But when the web page contains just a data file, as it sometimes does in this course, this is a great way to get a copy of the data. Copy the *URL* from your browser. `curl http://fisher.utstat.toronto.edu/~brunner/429s07/code_n_data/drp.dat > drp.dat`

This really is a minimal set of commands. The unix operating system is extremely powerful, and has an enormous number of commands. You can't really see the power from the minimal set above, but you can see the main drawback from the standpoint of the new user. Commands tend to be terse, consisting of just a few keystrokes. They make sense once you are familiar with them (like `ls` for listing the files in a directory, or `rm` for remove), but they are hard to guess. The `man` command (short for manual) gives very accurate information, but you have to know the name of the command before you can use `man` to find out about it.

Just for future reference, here are a few more commands that you may find useful, or otherwise appealing.

A Few More unix Commands

mkdir *dirname* Makes a new sub-directory (like a folder) named *dirname*. You can have sub-directories within sub-directories; it's a good way to organize your work.

cp *fname dirname* Copies the file *fname* into the directory *dirname*.

cd *dirname* Short for Change Directory. Takes you to the sub-directory *dirname*.

cd .. Moves you up a directory level.

cd Moves you to your main directory from wherever you are.

ls > *fname* Sends the output of the `ls` command to the file *fname* instead of to the screen.

cat *fname* Lists the whole file on your screen, not one page at a time. It goes by very fast, but usually you can scroll back up to see the entire file, if it's not too long.

cat *fname1 fname2* > *fname3* Concatenates *fname1* and *fname2* (sticks them together) and re-directs the output to *fname3*

grep ERROR cartoon1.log Searches for the string ERROR in the file cartoon1.log. Echos each line containing the string. Silent if ERROR does not occur. Case sensitive.

alias chk "grep ERROR *.log ; grep WARN *.log" Makes a new command called `chk`. It checks for the string ERROR and the string WARN in any log file.

cal Displays a calendar for this month

cal 1 3002 Displays a calendar for January 3002.

unset noclobber Are you tired of being asked if you really want to remove or overwrite a file?

rm *fname1 fname2* Remove both

rm -f *fname* Remove without asking for confirmation, this time only.

alias rm "rm -f" `rm` now means `rm -f`.

rm -r *dirname* Remove the directory, and everything in it recursively.

R –vanilla < *fname1* > *fname2* Execute the S language commands in *fname1*, sending output to *fname2*. Run in "plain vanilla" mode.

Printing files at home This is a question that always comes up. Almost surely, the printer connected to your printer at home is not directly connected to the university network. If you want to do something like print your SAS output at home, you have to transfer the file on the unix machine to the hard drive of your home computer, and print it from there. One way is to use some kind of `ftp` (file transfer protocol) tool to get the file in question onto your hard drive. For short files, you can also use the `less` or `cat` command to list the file on your screen, select it with your mouse, copy it, paste it to a word processing document, and print it from there. It is a good idea to use a fixed-width font like Courier, and not the Times or Times Roman font. Everything will be lined up better.

Perhaps easiest of all is to email yourself the file. This is illustrated in the first set of unix commands. To repeat, `mail yourname@yourisp.com < fname`.

2.2 Introduction to SAS

SAS stands for "Statistical Analysis System." Even though it runs on PCs as well as on bigger computers, it is truly the last of the great old mainframe

statistical packages. The first beta release was in 1971, and the SAS Institute, Inc. was spun off from North Carolina State University in 1976, the year after Bill Gates dropped out of Harvard. This is a serious pedigree, and it has both advantages and disadvantages.

The advantages are that the number of statistical procedures SAS can do is truly staggering, and the most commonly used ones have been tested so many times by so many people that their correctness and numerical efficiency is beyond any question. For the purposes of this class, there are no bugs. The disadvantages of SAS are all related to the fact that it was *designed* to run in a batch-oriented mainframe environment. So, for example, the SAS Institute has tried hard to make SAS an "interactive" program, but the interface still basically file and text oriented, not graphical.

2.2.1 The Four Main File Types

A typical SAS job will involve four main types of file.

- **The Raw Data File**: A file consisting of rows and columns of numbers; or maybe some of the columns have letters (character data) instead of numbers. The rows represent observations and the columns represent variables, as described at the beginning of Section 1.1. In the first example we will consider below, the raw data file is called `drp.dat`.

- **The Program File**: This is also sometimes called a "command file," because it's usually not much of a program. It consists of commands that the SAS software tries to follow. You create this file with a text editor like `emacs`. The command file contains a reference to the raw data file (in the `infile` statement), so SAS knows where to find the data. In the first example we will consider below, the command file is called `reading.sas`. SAS expects program files to have the extension `.sas`, and you should always follow this convention.

- **The Log File**: This file is produced by every SAS run, whether it is successful of unsuccessful. It contains a listing of the command file, as well any error messages or warnings. The name of the log file is automatically generated by SAS; it combines the first part of the command file's name with the extension `.log`. So for example, when SAS executes the commands in `reading.sas`, it writes a log file named `reading.log`.

- **The List File**: The list file contains the output of the statistical procedures requested by the command file. The list file has the extension `.lst` — so, for example, running SAS on the command file `reading.sas` will produce `reading.lst` as well as `reading.log`. A successful SAS run will almost always produce a list file. The absence of a list file indicates that there was at least one fatal error. The presence of a list file does not mean there were no errors; it just means that SAS was able to do *some* of what you asked it to do. Even if there are errors, the list file will usually not contain any error messages; they will be in the log file.

2.2.2 Running SAS from the Command Line

There are several ways to run SAS. In this text, all the examples will be run from the unix command line (`terminal`). In my view, this way is simplest and also the best way to start. Also, it is by far the easiest way to use SAS from home, assuming that SAS is running on a remote server and not your home computer.

The following illustrates a simple SAS run from the command line (using an application called `terminal` in some unix and linux environments). Initially, there are only two files in the directory — `reading.sas` (the program file) and `drp.dat` (the raw data file). The command `sas reading` produces two additional files — `reading.log` and `reading.lst`. In this and other examples, the unix prompt is `appsrv01.srv` (the name of the unix machine used to produce the examples), followed by a `>` sign.

```
appsrv01.srv> ls
drp.dat         reading.sas
appsrv01.srv> sas reading
appsrv01.srv> ls
drp.dat         reading.log     reading.lst     reading.sas
```

2.2.3 Structure of the Program File

A SAS program file is composed of units called *data steps* and *proc steps*. The typical SAS program has one data step and at least one proc step, though other structures are possible.

- Most SAS commands belong either in data step or in a proc step; they will generate errors if they are used in the wrong kind of step.
- Some statements, like the `title` and `options` commands, exist outside of the data and proc steps, but there are relatively few of these.

The Data Step The data step takes care of data acquisition and modification. It almost always includes a reference to at least one raw data file, telling SAS where to look for the data. It specifies variable names and labels, and provides instructions about how to read the data; for example, the data might be read from fixed column locations. Variables from the raw data file can be modified, and new variables can be created.

Each data step creates a **SAS data set**, a file consisting of the data (after modifications and additions), labels, and so on. Statistical procedures operate on SAS data sets, so you must create a SAS data set before you can start computing any statistics.

A SAS data set is written in a binary format that is very convenient for SAS to process, but is not readable by humans. In the old days, SAS data sets were always written to temporary scratch files on the computer's hard drive; these days, they may be maintained in RAM if they are small enough. In any case, the default is that a SAS data set disappears after the job has run. If the data step is executed again in a later run, the SAS data set is re-created.

Actually, it is possible to save a SAS data set on disk for later use. We won't do this here, but it makes sense when the amount of processing in a data step is large relative to the speed of the computer. As an extreme example, one of my colleagues uses SAS to analyze data from Ontario hospital admissions; the data files have millions of cases. Typically, it takes around 20 hours of CPU time on a very strong unix machine just to read the data and create a SAS data set. The resulting file, hundreds of gigabytes in size, is saved to disk, and then it takes just a few minutes to carry out each analysis. You wouldn't want to try this on a PC.

To repeat, SAS data *steps* and SAS data *sets* sound similar, but they are distinct concepts. A SAS data *step* is part of a SAS program; it generates a SAS data *set*, which is a file – usually a temporary file.

SAS data sets are not always created by SAS data steps. Some statistical procedures can create SAS data sets, too. For example, `proc standard` can take an ordinary SAS data set as input, and produce an output data set that

has all the original variables, and also some of the variables converted to z-scores (by subtracting off the mean and dividing by the standard deviation). `Proc reg` (the main multiple regression procedure) can produce a SAS data set containing residuals for plotting and use in further analysis; there are many other examples.

The `proc` Step "Proc" is short for procedure. Most procedures are statistical procedures; the most noticeable exception is `proc format`, which is used to provide labels for the values of categorical variables. The `proc` step is where you specify a statistical procedure that you want to carry out. A statistical procedures in the `proc` step will take a SAS data set as input, and write the results (summary statistics, values of test statistics, p-values, and so on) to the list file. The typical SAS program includes one data step and several `proc` steps, because it is common to produce a variety of data displays, descriptive statistics and significance tests in a single run.

2.2.4 A First Example: `reading.sas`

Earlier, we ran SAS on the file `reading.sas`, producing `reading.log` and `reading.lst`. Now we will look at `reading.sas` in some detail. This program is very simple; it has just one data step and one proc step. More details will be given later, but it's based on a study in which one group of grade school students received a special reading programme, and a control group did not. After a couple of months, all students were given a reading test. We're just going to do an independent groups t-test, but first take a look at the raw data file. You'd do this with the unix `less` command.

Actually, it's so obvious that you should look at your data that nobody ever says it. But experienced data analysts always do it — or else they assume everything is okay and get a bitter lesson in something they already knew. This is so important that it gets the formal status of a **data analysis hint**.

Data Analysis Hint 1 *Always look at your raw data file. It the data file is big, do it anyway. At least page through it a screen at a time, looking for anything strange. Check the values of all the variables for a few cases. Do they make sense? If you have obtained the data file from somewhere, along with a description of what's in it, never believe that the description you have been given is completely accurate.*

Anyway, here is the file `drp.dat`, with the middle and end cut out to save space.

Treatment	24
Treatment	43
Treatment	58
⋮	⋮
Control	55
Control	28
Control	48
⋮	⋮

Now we can look at `reading.sas`.

```
/****************** reading.sas *********************
*  Simple SAS job to illustrate a two-sample t-test    *
******************************************************/

options linesize=79 noovp formdlim='_';
title 'More & McCabe (1993) textbook t-test Example 7.8';

data reading;
     infile 'drp.dat';
     input group $ score;
     label group = 'Get Directed Reading Programme?'
           score = 'Degree of Reading Power Test Score';
proc ttest;
     class group;
     var score;
```

Here are some detailed comments about `reading.sas`.

- The first three lines are a comment. Anything between a `/*` and `*/` is a comment, and will be listed on the log file but otherwise ignored by SAS. Comments can appear anywhere in a program. You are not required to use comments, but it's a good idea.

  The most common error associated with comments is to forget to end them with `*/`. In the case of `reading.sas`, leaving off the `*/` (or

typing `\*` by mistake) would cause the whole program to be treated as a comment. It would generate no errors, and no output — because as far as SAS would be concerned, you never requested any. A longer program would eventually exceed the default length of a comment (it's some large number of characters) and SAS would end the "comment" for you. At exactly that point (probably in the middle of a command) SAS would begin parsing the program. Almost certainly, the first thing it examined would be a fragment of a legal command, and this would cause an error. The log file would say that the command caused an error, and not much else. It would be *very* confusing, because probably the command would be okay, and there would be no indication that SAS was only looking at part of it.

- The next two lines (the `options` statement and the `title` statement) exist outside the proc step and the data step. This is fairly rare.

- All SAS statements end with a semi-colon (`;`). SAS statements can extend for several physical lines in the program file (for example, see the `label` statement). Spacing, indentation, breaking up s statement into several lines of text – these are all for the convenience of the human reader, and are not part of the SAS syntax.

- The most common error in SAS programming is to forget the semi-colon. When this happens, SAS tries to interpret the following statement as part of the one you tried to end. This often causes not one error, but a cascading sequence of errors. The rule is, *if you have an error and you do not immediately understand what it is, look for a missing semi-colon.* It will probably be *before* the portion of the program that (according to SAS) caused the first error.

- Cascading errors are not caused just by the dreaded missing semi-colon. They are common in SAS; for example, a runaway comment statement can easily cause a chain reaction of errors (if the program is long enough for it to cause any error messages at all). *If you have a lot of errors in your log file, fix the first one and don't waste time trying to figure out the others.* Some or all of them may well disappear.

- `options linesize=79 noovp formdlim='_';`

These options are highly recommended. The `linesize=79` option is so highly recommended it's almost obligatory. It causes SAS to write the output 79 columns across, so it can be read on an ordinary terminal screen that's 80 characters across. You specify an output width of 79 characters rather than 80, because SAS uses one column for printer control characters, like page ejects (form feeds).

If you do not specify `options linesize=79;`, SAS will use its default of 132 characters across, the width of sheet of paper from an obsolete line printer you probably have never seen. Why would the SAS Institute hang on to this default, when changing it to match ordinary letter paper would be so easy? It probably tells you something about the computing environments of some of SAS's large corporate clients.

- The `noovp` option makes the log files more readable if you have errors. When SAS finds an error in your program, it tries to *underline* the word that caused the error. It does this by going back and *overprinting* the offending word with a series of "underscores" (_ characters). On many printers this works, but when you try to look at the log file on a terminal screen (one that is *not* controlled by the SAS Display Manager), what often appears is a mess. The `noovp` option specifies `no overprinting`. It causes the "underlining" to appear on a separate line under the program line with the error. If you're running SAS from the unix command line and looking at your log files with the `less` command or the `cat` command, you will probably find the `noovp` option to be helpful.

- The `formdlim='_'` option specifies a "form delimiter" to replace most form feeds (new physical pages) in the list file. This can save a lot of paper (and page printing charges). You can use any string you want for a form delimiter. The underscore (the one specified here) causes a solid line to be printed instead of going to a new sheet of paper.

- `title` This is optional, but recommended. The material between the single quotes will appear at the top of each page. This can be a lifesaver when you are searching through a stack of old printouts for something you did a year or two ago.

- `data reading;` This begins the data step, specifying that the name of the SAS data set being created is "reading." The names of data sets

are arbitrary, but you should make them informative.

- `infile` Specifies the name of the raw data file. The file name, enclosed in single quotes, can be the full unix path to the file, like `/dos/brunner/public/senic.raw`. If you just give the name of the raw data file, as in this example, SAS assumes that the file is in the same directory as the command file.

- `input` Gives the names of the variables.

  - A character variable (the values of `group` are "Treatment' and "Control") must be followed by a dollar sign.
  - Variable names must be eight characters or less, and should begin with a letter. They will be used to request statistical procedures in the `proc` step. They should be meaningful (related to what the variable *is*), and easy to remember.
  - This is almost the simplest form of the `input` statement. It can be very powerful; for example, you can read data from different locations and in different orders, depending on the value of a variable you've just read, and so on. It can get complicated, but if the data file has a simple structure, the input statement can be simple too.

- `label` Provide descriptive labels for the variables; these will be used to label the output, usually in very nice way. Labels can be quite useful, especially when you're trying to recover what you did a while ago. Notice how this statement extends over two physical lines.

- `proc ttest;` Now the proc step begins. This program has only one data step and one proc step. We are requesting a two-sample t-test.

- `class` Specifies the independent variable.

- `var` Specifies the dependent variable(s). You can give a list of dependent variables. A separate univariate test (actually, as you will see, *collection* of tests is performed for each dependent variable.

reading.log Log files are not very interesting when everything is okay, but here is an example anyway. Notice that in addition to a variety of technical information (where the files are, how long each step took, and so on), it contains a listing of the SAS program — in this case, `reading.sas`. If there were syntax errors in the program, this is where the error messages would appear.

```
appsrv01.srv> cat reading.log
1                                                          The SAS System
                          11:40 Thursday, September 2, 2007

NOTE: Copyright (c) 1999-2001 by SAS Institute Inc., Cary, NC, USA.
NOTE: SAS (r) Proprietary Software Release 8.2 (TS2M0)
      Licensed to UNIVERSITY OF TORONTO/COMPUTING & COMMUNICATIONS, Site 00089870
NOTE: This session is executing on the Linux 2.6.8.1-smp-athlon-bk platform.



This message is contained in the SAS news file, and is presented upon
initialization.  Edit the files "news" in the "misc/base" directory to
display site-specific news and information in the program log.
The command line option "-nonews" will prevent this display.



NOTE: SAS initialization used:
      real time           0.08 seconds
      cpu time            0.01 seconds

1          /****************** reading.sas ********************
2          *  Simple SAS job to illustrate a two-sample t-test    *
3          ******************************************************/
4
5          options linesize=79 noovp formdlim='_';
6          title 'More & McCabe (1993) textbook t-test Example 7.8';
7
8          data reading;
```

```
9               infile 'drp.dat';
10              input group $ score;
11              label group = 'Get Directed Reading Programme?'
12                    score = 'Degree of Reading Power Test Score';

NOTE: The infile 'drp.dat' is:
      File Name=/homes/students/u0/stats/brunner/drp.dat,
      Owner Name=brunner,Group Name=stats,
      Access Permission=rw-r-----,
      File Size (bytes)=660

NOTE: 44 records were read from the infile 'drp.dat'.
      The minimum record length was 14.
      The maximum record length was 14.
NOTE: The data set WORK.READING has 44 observations and 2 variables.
NOTE: DATA statement used:
      real time           0.01 seconds
      cpu time            0.01 seconds


13         proc ttest;
14              class group;
15              var score;
NOTE: There were 44 observations read from the data set WORK.READING.
NOTE: The PROCEDURE TTEST printed page 1.
NOTE: PROCEDURE TTEST used:
      real time           0.08 seconds
      cpu time            0.01 seconds


NOTE: SAS Institute Inc., SAS Campus Drive, Cary, NC USA 27513-2414

2                                The SAS System
                                              11:40 Thursday, September 2, 2007

NOTE: The SAS System used:
      real time           0.24 seconds
      cpu time            0.03 seconds
```

reading.lst Here is the list file. Notice that the title specified in the `title` statement appears at the top, along with the time and date the program was executed. Then we get statistical output — the t-test we want, and also a bunch of other stuff, whether we want it or not. This is typical of SAS, and most other mainstream statistical packages as well. The default output from any given statistical procedures will contain more information than you wanted, and probably some things you don't understand at all. There are usually numerous options that can add *more* information, but almost never options to reduce the default output. So, you just learn what to ignore. It is helpful, but not essential,to have at least a superficial understanding of everything in the default output from procedures you use a lot.

```
appsrv01.srv> cat reading.lst
```

```
---------------------------------------------------------------------------

              More & McCabe (1993) textbook t-test Example 7.8              1
                                          11:40 Thursday, September 2, 2007

                            The TTEST Procedure

                               Statistics

                                  Lower CL          Upper CL  Lower CL
  Variable  group          N       Mean    Mean      Mean   Std Dev  Std Dev

  score     Control       23     34.106  41.522    48.937    13.263   17.149
  score     Treatmen      21     46.466  51.476    56.487    8.4213   11.007
  score     Diff (1-2)           -18.82  -9.954    -1.091    11.998   14.551
```

Statistics

Variable	group	Upper CL Std Dev	Std Err	Minimum	Maximum
score	Control	24.271	3.5758	10	85
score	Treatmen	15.895	2.402	24	71
score	Diff (1-2)	18.495	4.3919		

T-Tests

Variable	Method	Variances	DF	t Value	Pr > |t|
score	Pooled	Equal	42	-2.27	0.0286
score	Satterthwaite	Unequal	37.9	-2.31	0.0264

Equality of Variances

Variable	Method	Num DF	Den DF	F Value	Pr > F
score	Folded F	22	20	2.43	0.0507

Now here are some comments about `reading.lst`.

- The first part of the output is labelled "Statistics," containing confidence intervals and some descriptive statistics. There are three rows to this display: one for the Control group, one for the Treatment group, and one for the difference between groups.
  - `Variable: score` The first column, labelled "Variable," tells you what the dependent variable is – particularly useful if you have more than one.
  - `group` The independent variable. Underneath are the values of the independent variable in the first two rows. The third row is for the difference, computed as Control minus Treatment (1-2).

Well actually, if you look carefully, you see that we do *not* quite get the values of the independent variable under `GROUP`. The values of the (alphanumeric, or character-valued) variable `group` are `Control` and `Treatment`, but the printout says "`Treatmen`." This is not a printing error; it is a subtle error in the reading of the data. The default length of an alphanumeric data value is 8 characters, but "Treatment" has 9 characters. So SAS just read the first eight. No error message was generated and no harm was done in this case, but in other circumstances this error can turn a data file into a giant pile of trash, without warning. Later we will see how to override the default and read longer strings if necessary.

- `N` The third column gives sample sizes; n=23 for the control group, and n=21 for the treatment group.
- The next three columns contain means and their associated 95% confidence intervals. The middle column has the means. For the Control group, the sample mean score on the DRP test was 41.522; for the Treatment group, the sample mean was 51.476. The difference between means is -9.954 = 41.522-51.476 (One minus Two).

  To the left of the mean, labelled "Lower CL Mean," is the lower confidence limit of the 95% confidence interval for the population mean. Thus, the 95% confidence interval for the Treatment mean is from 46.466 to 56.487, and the 95% confidence interval for the *difference* between means is from -18.82 to -1.091. The fact that this last interval does not contain zero means that the usual two-tailed t-test will be statistically significantly at the 0.05 level. There is a lovely consistency between the classical tests and confidence intervals.
- The next three columns give confidence intervals for the standard deviations. We have the lower confidence limit, the standard deviation, and the upper confidence limit in the continuation below.
- Then we get standard errors (estimated standard deviations of the sample mean or difference between means), and finally the minimum and maximum for each group.

- Then finally, under "T-Tests," we get what we want – a t-test for the difference between the means of the Control and Treatment groups. Two tests are given; as usual there seems to be more output than

we were expecting or wishing for. Probably we were looking for the first one, using the Pooled Method. This is the traditional test, which assumes equal population variances, and therefore is based on a *Pooled* estimate of the common within-groups standard deviation.

- The value of the test statistic is `-2.27`.
- The degrees of freedom $n_1 + n_2 - 2$ is given in the `DF` column.
- The column `Prob>|t|` gives the two-tailed (two-sided) p-value. It is less than the traditional value of 0.05, so the results are statistically significant.

Sample Question 2.2.1 *What do we conclude from this study? Say something about reading, using non-technical language.*

Answer to Sample Question 2.2.1 *Students who received the Directed Reading Program got higher average reading scores than students in the control condition.*

It's worth emphasizing here that the main objective of doing a statistical analysis is to draw conclusions about the data — or to refrain from drawing such conclusions. The question "What do we conclude from this study?" will always be asked. For now, the right answer will always be either "Nothing; the results were not statistically significant," or else it will be something about reading, or fish, or potatoes, or AIDS, or whatever is being studied. Later we will take up the possibility of concluding that the effect we are testing is actually *absent* (or at least trivially small).

Many students, even when they have been warned, respond to the "what do you conclude" question with a barrage of statistical terminology. They go on and on about the null hypothesis and Type I error, and usually say nothing that would tell a reasonable person what actually happened in the study. In the working world, a memo filled with such garbage could get you fired. Here, it will get you a zero for the question, even if the technical details you give are correct.

Remember, the purpose of writing up a statistical analysis is not to sound impressive and technical, but to impart information. To say things in a simple way is a virtue. It shows you understand what is going on. Now back to the printout.

- The Satterthwaite method gives a sort of t-test that does not assume equal variances. Well, it's not really a t-test, because the test statistic does not really have a t distribution, even when the data are exactly normal. But, the (very unpleasant) distribution of the test statistic is well approximated by a t distribution with the right degrees of freedom — not $n_1 + n_2 - 2$, but something messy that depends on the data. See the odd fractional degrees of freedom? See [8], or lots of other elementary texts, for details. In any case, it does not matter much in this case, because the p-value is almost the same as the p-value from the traditional test. They lead to the same conclusions, and there is no problem. What should you do when they disagree? I'd go with the test that makes fewer assumptions.

- Next we see a test for Equality of Variances. This "Folded" F is the traditional test for whether the variances of two groups are equal, and it's *almost* significant. This test is provided so people can test for differences between variances; if it is significantly different they can use the unequal variance t-test, and otherwise they can use the traditional test. This seems reasonable, except for the following.

  Both the two-sample t-test and the F-test for equality of variances assume that the data are normally distributed. However, the normality assumption does not matter much for the t-test when the sample sizes are large, while for the variance test it matters a *lot*, regardless of how much data you have. When the data are non-normal, the test for variances will be significant more than 5% of the time even when the population variances are equal. If you have equal population variances and a large sample of non-normal data, the F-test for variances could easily be significant, leading you to worry unnecessarily about the validity of the t-test.

2.2.5 Background of the First Example

We don't do statistical analysis in a vacuum. Before proceeding with more computing details, let's find out more about the reading data. This first example is from an introductory text. It's Example 7.8 (p. 534) in More and McCabe's excellent *Introduction to the practice of statistics* [8]. We are interested in analyzing *real* data, not in doing textbook exercises. But we will not turn up our noses just yet, because

Data Analysis Hint 2 *When learning how to carry out a procedure using unfamiliar statistical software, always do a textbook example first, and compare the output to the material in the text. Regardless of what the manual might say, never assume you know what the software is doing until you see an example.*

More and McCabe do a great job of explaining the t-test with unequal variances, something SAS produces (along with usual t-test that assumes equal variances) without being asked when you request a t-test. Besides, the data actually come from someone's Ph.D. thesis, so there is an element of realism. Here is Moore and McCabe's description of the study.

> An educator believes that new directed reading activities in the classroom will help elementary school pupils improve some aspects of their reading ability. She arranges for a third grade class of 21 students to take part in these activities. A control classroom of 23 third graders follows the same curriculum without the activities. At the end of 8 weeks, all students are given a Degree of Reading Power (DRP) test, which measures the aspects of reading ability that the program is designed to improve.

Sample Question 2.2.2 *What's wrong with this study?*

Answer to Sample Question 2.2.2 *The independent variable was manipulated by the experimenter, but it is not an experimental study. Even if classrooms were assigned randomly to conditions (it is impossible to tell whether they were, from this brief description), a large number of unobserved variables are potentially confounded with treatment. The teacher in the classroom that received the treatment might be better than the teacher in the control classroom, or possibly there was a particularly aggressive bully in the control classroom, or maybe a mini-epidemic of some childhood disease hit the control classroom .... The list goes on. The point here is that there are many ways in which the classroom experiences of children in the treatment group differ systematically from the experiences of children in the control group.*

Sample Question 2.2.3 *How could the problem be fixed?*

Answer to Sample Question 2.2.3 *Assign classrooms at random to treatments. The unit of analysis should be the classroom, not the individual student.*

2.2.6 SAS Example Two: The statclass data

These data come from a statistics class taught many years ago. Students took eight quizzes, turned in nine computer assignments, and also took a midterm and final exam. The data file also includes gender and ethnic background; these last two variables are just guesses by the professor, and there is no way to tell how accurate they were. The data file looks like this. There are 21 columns and 62 rows of data; columns not aligned. Here are the first few lines.

```
appsrv01.srv>  less statclass1.dat
1  2  9  1  7  8  4  3  5  2  6  10  10  10  5  0  0  0  0  55  43
0  2  10  10  5  9  10  8  6  8  10  10  8  9  9  9  9  10  10  66  79
1  2  10  10  5  10  10  10  9  8  10  10  10  10  10  10  9  10  10  94  67
1  2  10  10  8  9  10  7  10  9  10  10  10  9  10  10  9  10  10  81  65
0  1  10  1  0  0  8  6  5  2  10  9  0  0  10  6  0  5  0  54  .
1  1  10  6  7  9  8  8  5  7  10  9  10  9  5  6  4  8  10  57  52
0  1  0  0  9  9  10  5  2  2  8  7  7  10  10  6  3  7  10  49  .
0  1  10  9  5  8  9  8  5  6  8  7  5  6  10  6  5  9  9  77  64
0  1  10  8  6  8  9  5  3  6  9  9  6  9  10  6  5  7  10  65  42
1  1  10  5  6  7  10  4  6  0  10  9  10  9  10  6  7  8  10  73  .
0  1  9  0  4  6  10  5  3  3  10  8  10  5  10  10  9  9  10  71  37

⋮
```

Notice the periods at the ends of lines 5, 7 and 10. The period is the SAS *missing value code*. These people did not show up for the final exam. They may have taken a makeup exam, but if so their scores did not make it into this data file. When a case has a missing value recorded for a variable, SAS automatically excludes that case from any statistical calculation involving the variable. If a new variable is being created based on the value of a variable with a missing value, the new variable will usually have a missing value for that case too.

Here is the SAS program `statmarks1.sas`. It reads and labels the data, and then does a variety of significance tests. They are all elementary except the last one, which illustrates testing for one set of independent variables controlling for another set in multiple regression.

```
appsrv01.srv>  cat statmarks1.sas

                  /* statmarks1.sas */
options linesize=79 noovp formdlim='_';
title 'Grades from STA3000 at Roosevelt University:  Fall, 1957';
title2 'Illustrate Elementary Tests';

proc format; /* Used to label values of the categorical variables */
     value sexfmt    0 = 'Male'   1 = 'Female';
     value ethfmt    1 = 'Chinese'
                     2 = 'European'
                     3 = 'Other' ;
data grades;
     infile 'statclass1.dat';
     input sex ethnic quiz1-quiz8 comp1-comp9 midterm final;
     /* Drop lowest score for quiz & computer  */
     quizave = ( sum(of quiz1-quiz8) - min(of quiz1-quiz8) ) / 7;
     compave = ( sum(of comp1-comp9) - min(of comp1-comp9) ) / 8;
     label ethnic = 'Apparent ethnic background (ancestry)'
           quizave = 'Quiz Average (drop lowest)'
           compave = 'Computer Average (drop lowest)';
     mark = .3*quizave*10 + .1*compave*10 + .3*midterm + .3*final;
     label mark = 'Final Mark';
     diff = quiz8-quiz1; /* To illustrate matched t-test */
     label diff = 'Quiz 8 minus Quiz 1';
     mark2 = round(mark);
     /* Bump up at grade boundaries */
     if mark2=89 then mark2=90;
     if mark2=79 then mark2=80;
     if mark2=69 then mark2=70;
     if mark2=59 then mark2=60;
     /* Assign letter grade */
     if mark2=. then grade='Incomplete';
        else if mark2 ge 90 then grade = 'A';
        else if 80 le mark2 le 89 then grade='B';
        else if 70 le mark2 le 79 then grade='C';
        else if 60 le mark2 le 69 then grade='D';
        else grade='F';
```

```
     format sex sexfmt.;       /* Associates sex & ethnic    */
     format ethnic ethfmt.;    /* with formats defined above */

/* Now the proc steps */


proc freq;
      title3 'Frequency distributions of the categorical variables';
      tables sex ethnic grade;

proc means n mean std;
     title3 'Means and SDs of quantitative variables';
     var quiz1 -- mark;        /*  single dash only works with numbered
                                  lists, like quiz1-quiz8    */
proc ttest;
     title3 'Independent t-test';
     class sex;
     var mark;
proc means n mean std t;
     title3 'Matched t-test: Quiz 1 versus 8';
     var quiz1 quiz8 diff;
proc glm;
     title3 'One-way anova';
     class ethnic;
     model mark = ethnic;
     means ethnic;
     means ethnic / Tukey Bon Scheffe;
proc freq;
     title3 'Chi-squared Test of Independence';
     tables sex*ethnic sex*grade ethnic*grade / chisq;
proc freq; /*  Added after seeing warning from chisq test above */
     title3 'Chi-squared Test of Independence: Version 2';
     tables sex*ethnic grade*(sex ethnic) / norow nopercent chisq expected;
proc corr;
     title3 'Correlation Matrix';
     var final midterm quizave compave;
proc plot;
     title3 'Scatterplot';
```

```
     plot final*midterm; /* Really should do all combinations */
proc reg;
     title3 'Simple regression';
     model final=midterm;

/* Predict final exam score from midterm, quiz & computer */
proc reg simple;
     title3 'Multiple Regression';
     model final = midterm quizave compave / ss1;
     smalstuf: test quizave = 0, compave = 0;
run;

/* Note that the final run statement is not needed when
   running SAS from the unix command line. */
```

Noteworthy features of this program include

- `options`: Already discussed in connection with `reading.sas`.
- `title2`: Subtitle
- `proc format`: This is a non-statistical procedure – a rarity in the SAS language. It is the way SAS takes care of labelling categorical variables when the categories are coded as numbers. `proc format` defines *printing formats*. For any variable associated with the printing format named `sexfmt`, any time it would print the value "0" (in a table or something) it instead prints the string "Male." The associations between variables and printing formats are accomplished in the `format` statement at the end of the data step. The names of formats have a period at the end to distinguish them from variable names. Of course formats must be defined before they can be associated with variables. This is why `proc format` precedes the data step.
- `quiz1-quiz8`: One may refer to a *range* of variables ending with consecutive numbers using a minus sign. In the `input` statement, a range can be defined (named) this way. It saves typing and is easy to read.
- Creating new variables with assignment statements. The variables `quizave`, `compave` and `mark` are not in the original data file. They are created here, and they are appended to the end of the SAS data

set in oder of creation. Variables like this should never be in the raw data file.

Data Analysis Hint 3 *When variables are exact mathematical functions of other variables, always create them in the data step rather than including them in the raw data file. It saves data entry, and makes the data file smaller and easier to read. If you want to try out a different definition of the variable, it's easy to change a few statements in the data step.*

- `sum(of quiz1-quiz8)`: Without the word "of," the minus sign is ambiguous. In the SAS language, `sum(quiz1-quiz8)` is the sum of a single number, the difference between `quiz1` and `quiz8`.
- `format sex sexfmt.;` Associates the variable `sex` with its printing format. In questionnaire studies where a large number of items have the same potential responses (like a scale from 1 = Strongly Agree to 7=Strongly Disagree), it is common to associate a long list of variables with a single printing format.
- `quiz1 -- mark` in the first `proc means`: A double dash refers to a list of variables *in the order of their creation* in the `data` step. Single dashes are for numerical order, while double dashes are for order of creation; it's very handy.
- Title inside a procedure labels just that procedure.
- `proc means n mean std t` A matched t-test is just a single-variable t-test carried out on differences, testing whether the mean difference is equal to zero.
- `proc glm`
  - `class` Tells SAS that the IV ethnic is categorical.
  - `model` Dependent variable(s) = independent variable(s)
  - `means ethnic`: Mean of `mark` separately for each value of `ethnic`.
  - `means ethnic / Tukey Bon Scheffe`: Post hoc tests (multiple comparisons, probing, follow-ups). Used if the overall F-test is significant, to see which means are different from which other means.

- `chisq` option on `proc freq`: Gives a large collection of chisquare tests. The first one is the familiar Pearson chisquare test of independence (the one comparing observed and expected frequencies).
- `tables sex*ethnic / norow nopercent chisq expected;` In this second version of the crosstab produced `proc freq`, we suppress the row and total percentages, and look at the expected frequencies because SAS warned us that some of them were too small. SAS issues a warning if any expected frequency is below 5; this is the old-fashioned rule of thumb. But it has been known for some time that Type I error rates are affected mostly by expected frequencies smaller than one, not five — so I wanted to take a look.
- `proc corr` After `var`, list the variables you want to see in a correlation matrix.
- `proc plot; plot final*midterm;` Scatterplot: First variable named goes on the y axis.
- `proc reg`: `model` Dependent variable(s) = independent variable(s) again
- `simple` option on `proc reg` gives simple descriptive statistics. This last procedure is an example of multiple regression, and we will return to it later once we have more background.

statmarks1.lst

```
-------------------------------------------------------------------------------

          Grades from STA3000 at Roosevelt University:  Fall, 1957          1
                        Illustrate Elementary Tests
            Frequency distributions of the categorical variables
                                          10:10 Sunday, September 5, 2007

                            The FREQ Procedure

                                          Cumulative    Cumulative
            sex     Frequency     Percent     Frequency      Percent
          -----------------------------------------------------------
          Male            39       62.90            39        62.90
          Female          23       37.10            62       100.00
```

Apparent ethnic background (ancestry)

ethnic	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Chinese	41	66.13	41	66.13
European	15	24.19	56	90.32
Other	6	9.68	62	100.00

grade	Frequency	Percent	Cumulative Frequency	Cumulative Percent
A	3	4.84	3	4.84
B	6	9.68	9	14.52
C	18	29.03	27	43.55
D	21	33.87	48	77.42
F	10	16.13	58	93.55
Incomplete	4	6.45	62	100.00

Grades from STA3000 at Roosevelt University: Fall, 1957 2
Illustrate Elementary Tests
Means and SDs of quantitative variables
10:10 Sunday, September 5, 2007

The MEANS Procedure

Variable	Label	N	Mean	Std Dev
quiz1		62	9.0967742	2.2739413
quiz2		62	5.8870968	3.2294995
quiz3		62	6.0483871	2.3707744
quiz4		62	7.7258065	2.1590022
quiz5		62	9.0645161	1.4471109
quiz6		62	7.1612903	1.9264641
quiz7		62	5.7903226	2.1204477
quiz8		62	6.3064516	2.3787909
comp1		62	9.1451613	1.1430011
comp2		62	8.8225806	1.7604414
comp3		62	8.3387097	2.5020880
comp4		62	7.8548387	3.2180168
comp5		62	9.4354839	1.7237109
comp6		62	7.8548387	2.4350364
comp7		62	6.6451613	2.7526248
comp8		62	8.8225806	1.6745363
comp9		62	8.2419355	3.7050497
midterm		62	70.1935484	13.6235557
final		58	50.3103448	17.2496701
quizave	Quiz Average (drop lowest)	62	7.6751152	1.1266917
compave	Computer Average (drop lowest)	62	8.8346774	1.1204997
mark	Final Mark	58	68.4830049	10.3902874

Illustrate Elementary Tests

Independent t-test

10:10 Sunday, September 5, 2007

The TTEST Procedure

Statistics

Variable	sex	N	Lower CL Mean	Mean	Upper CL Mean	Lower CL Std Dev	Std Dev
mark	Male	36	65.604	68.57	71.535	7.1093	8.7653
mark	Female	22	62.647	68.341	74.036	9.8809	12.843
mark	Diff (1-2)		-5.454	0.2284	5.9108	8.8495	10.482

Statistics

Variable	sex	Upper CL Std Dev	Std Err	Minimum	Maximum
mark	Male	11.434	1.4609	54.057	89.932
mark	Female	18.354	2.7382	48.482	95.457
mark	Diff (1-2)	12.859	2.8366		

T-Tests

Variable	Method	Variances	DF	t Value	Pr > |t|
mark	Pooled	Equal	56	0.08	0.9361
mark	Satterthwaite	Unequal	33.1	0.07	0.9418

Equality of Variances

Variable	Method	Num DF	Den DF	F Value	Pr > F
mark	Folded F	21	35	2.15	0.0443

Illustrate Elementary Tests

Matched t-test: Quiz 1 versus 8

10:10 Sunday, September 5, 2007

The MEANS Procedure

Variable	Label	N	Mean	Std Dev	t Value
quiz1		62	9.0967742	2.2739413	31.50
quiz8		62	6.3064516	2.3787909	20.87
diff	Quiz 8 minus Quiz 1	62	-2.7903226	3.1578011	-6.96

```
---------------------------------------------------------------------------

          Grades from STA3000 at Roosevelt University:  Fall, 1957          5
                        Illustrate Elementary Tests
                              One-way anova  10:10 Sunday, September 5, 2007

                               The GLM Procedure

                            Class Level Information

                 Class         Levels    Values

                 ethnic             3    Chinese European Other


                           Number of observations    62

NOTE: Due to missing values, only 58 observations can be used in this analysis.

---------------------------------------------------------------------------

          Grades from STA3000 at Roosevelt University:  Fall, 1957          6
                        Illustrate Elementary Tests
                              One-way anova  10:10 Sunday, September 5, 2007

                               The GLM Procedure

Dependent Variable: mark   Final Mark

                                     Sum of
Source                       DF        Squares    Mean Square   F Value   Pr > F

Model                         2    1238.960134     619.480067      6.93   0.0021

Error                        55    4914.649951      89.357272

Corrected Total              57    6153.610084


                R-Square     Coeff Var      Root MSE     mark Mean

                0.201339      13.80328      9.452898      68.48300


Source                       DF      Type I SS    Mean Square   F Value   Pr > F

ethnic                        2    1238.960134     619.480067      6.93   0.0021


Source                       DF    Type III SS    Mean Square   F Value   Pr > F

ethnic                        2    1238.960134     619.480067      6.93   0.0021

---------------------------------------------------------------------------

          Grades from STA3000 at Roosevelt University:  Fall, 1957          7
                        Illustrate Elementary Tests
```

One-way anova 10:10 Sunday, September 5, 2007

The GLM Procedure

Level of ethnic	N	mark Mean	mark Std Dev
Chinese	37	65.2688224	7.9262171
European	15	76.0142857	11.2351562
Other	6	69.4755952	13.3097753

Illustrate Elementary Tests
One-way anova 10:10 Sunday, September 5, 2007

The GLM Procedure

Tukey's Studentized Range (HSD) Test for mark

NOTE: This test controls the Type I experimentwise error rate.

Alpha	0.05
Error Degrees of Freedom	55
Error Mean Square	89.35727
Critical Value of Studentized Range	3.40649

Comparisons significant at the 0.05 level are indicated by ***.

ethnic Comparison	Difference Between Means	Simultaneous 95% Confidence Limits		
European - Other	6.539	-4.460	17.538	
European - Chinese	10.745	3.776	17.715	***
Other - European	-6.539	-17.538	4.460	
Other - Chinese	4.207	-5.814	14.228	
Chinese - European	-10.745	-17.715	-3.776	***
Chinese - Other	-4.207	-14.228	5.814	

Illustrate Elementary Tests
One-way anova 10:10 Sunday, September 5, 2007

The GLM Procedure

Bonferroni (Dunn) t Tests for mark

NOTE: This test controls the Type I experimentwise error rate, but it generally has a higher Type II error rate than Tukey's for all pairwise comparisons.

```
                    Alpha                          0.05
                    Error Degrees of Freedom         55
                    Error Mean Square          89.35727
                    Critical Value of t         2.46941


      Comparisons significant at the 0.05 level are indicated by ***.


                                Difference
                                  Between     Simultaneous 95%
          ethnic                    Means    Confidence Limits
        Comparison

        European - Other             6.539      -4.737   17.814
        European - Chinese          10.745       3.600   17.891  ***
        Other    - European         -6.539     -17.814    4.737
        Other    - Chinese           4.207      -6.067   14.480
        Chinese  - European        -10.745     -17.891   -3.600  ***
        Chinese  - Other            -4.207     -14.480    6.067
```

```
                      Illustrate Elementary Tests
                             One-way anova  10:10 Sunday, September 5, 2007

                           The GLM Procedure

                         Scheffe's Test for mark

      NOTE: This test controls the Type I experimentwise error rate, but it
      generally has a higher Type II error rate than Tukey's for all pairwise
                                  comparisons.


                    Alpha                          0.05
                    Error Degrees of Freedom         55
                    Error Mean Square          89.35727
                    Critical Value of F         3.16499


      Comparisons significant at the 0.05 level are indicated by ***.


                                Difference
                                  Between     Simultaneous 95%
          ethnic                    Means    Confidence Limits
        Comparison

        European - Other             6.539      -4.950   18.027
        European - Chinese          10.745       3.466   18.025  ***
        Other    - European         -6.539     -18.027    4.950
        Other    - Chinese           4.207      -6.260   14.674
        Chinese  - European        -10.745     -18.025   -3.466  ***
        Chinese  - Other            -4.207     -14.674    6.260
```

The FREQ Procedure

Table of sex by ethnic

sex ethnic(Apparent ethnic background (ancestry))

Frequency Percent Row Pct Col Pct	Chinese	European	Other	Total
Male	27 43.55 69.23 65.85	7 11.29 17.95 46.67	5 8.06 12.82 83.33	39 62.90
Female	14 22.58 60.87 34.15	8 12.90 34.78 53.33	1 1.61 4.35 16.67	23 37.10
Total	41 66.13	15 24.19	6 9.68	62 100.00

Statistics for Table of sex by ethnic

Statistic	DF	Value	Prob
Chi-Square	2	2.9208	0.2321
Likelihood Ratio Chi-Square	2	2.9956	0.2236
Mantel-Haenszel Chi-Square	1	0.0000	0.9949
Phi Coefficient		0.2170	
Contingency Coefficient		0.2121	
Cramer's V		0.2170	

WARNING: 33% of the cells have expected counts less than 5. Chi-Square may not be a valid test.

Sample Size = 62

The FREQ Procedure

Table of sex by grade

sex	grade						
Frequency| Percent | Row Pct | Col Pct	A	B	C	D	F	Incomplete	Total
Male	1	3	13	14	5	3	39
	1.61	4.84	20.97	22.58	8.06	4.84	62.90
	2.56	7.69	33.33	35.90	12.82	7.69	
	33.33	50.00	72.22	66.67	50.00	75.00	
Female	2	3	5	7	5	1	23
	3.23	4.84	8.06	11.29	8.06	1.61	37.10
	8.70	13.04	21.74	30.43	21.74	4.35	
	66.67	50.00	27.78	33.33	50.00	25.00	
Total	3	6	18	21	10	4	62
	4.84	9.68	29.03	33.87	16.13	6.45	100.00

Statistics for Table of sex by grade

Statistic	DF	Value	Prob
Chi-Square	5	3.3139	0.6517
Likelihood Ratio Chi-Square	5	3.2717	0.6582
Mantel-Haenszel Chi-Square	1	0.2342	0.6284
Phi Coefficient		0.2312	
Contingency Coefficient		0.2253	
Cramer's V		0.2312	

WARNING: 58% of the cells have expected counts less than 5. Chi-Square may not be a valid test.

Sample Size = 62

The FREQ Procedure

Table of ethnic by grade

ethnic(Apparent ethnic background (ancestry)) grade

Frequency / Percent / Row Pct / Col Pct	A	B	C	D	F	Incomplete	Total
Chinese	0	2	11	17	7	4	41
	0.00	3.23	17.74	27.42	11.29	6.45	66.13
	0.00	4.88	26.83	41.46	17.07	9.76	
	0.00	33.33	61.11	80.95	70.00	100.00	
European	2	4	5	3	1	0	15
	3.23	6.45	8.06	4.84	1.61	0.00	24.19
	13.33	26.67	33.33	20.00	6.67	0.00	
	66.67	66.67	27.78	14.29	10.00	0.00	
Other	1	0	2	1	2	0	6
	1.61	0.00	3.23	1.61	3.23	0.00	9.68
	16.67	0.00	33.33	16.67	33.33	0.00	
	33.33	0.00	11.11	4.76	20.00	0.00	
Total	3	6	18	21	10	4	62
	4.84	9.68	29.03	33.87	16.13	6.45	100.00

Statistics for Table of ethnic by grade

Statistic	DF	Value	Prob
Chi-Square	10	18.2676	0.0506
Likelihood Ratio Chi-Square	10	19.6338	0.0329
Mantel-Haenszel Chi-Square	1	5.6222	0.0177
Phi Coefficient		0.5428	
Contingency Coefficient		0.4771	
Cramer's V		0.3838	

WARNING: 78% of the cells have expected counts less than 5. Chi-Square may not be a valid test.

Sample Size = 62

The FREQ Procedure

Table of sex by ethnic

sex ethnic(Apparent ethnic background (ancestry))

Frequency| Expected | Col Pct	Chinese	European	Other	Total
Male	27	7	5	39
	25.79	9.4355	3.7742	
	65.85	46.67	83.33	
Female	14	8	1	23
	15.21	5.5645	2.2258	
	34.15	53.33	16.67	
Total	41	15	6	62

Statistics for Table of sex by ethnic

Statistic	DF	Value	Prob
Chi-Square	2	2.9208	0.2321
Likelihood Ratio Chi-Square	2	2.9956	0.2236
Mantel-Haenszel Chi-Square	1	0.0000	0.9949
Phi Coefficient		0.2170	
Contingency Coefficient		0.2121	
Cramer's V		0.2170	

WARNING: 33% of the cells have expected counts less than 5. Chi-Square may not be a valid test.

Sample Size = 62

The FREQ Procedure

Table of grade by sex

grade sex

Frequency / Expected / Col Pct	Male	Female	Total
A	1 1.8871 2.56	2 1.1129 8.70	3
B	3 3.7742 7.69	3 2.2258 13.04	6
C	13 11.323 33.33	5 6.6774 21.74	18
D	14 13.21 35.90	7 7.7903 30.43	21
F	5 6.2903 12.82	5 3.7097 21.74	10
Incomplete	3 2.5161 7.69	1 1.4839 4.35	4
Total	39	23	62

Statistics for Table of grade by sex

Statistic	DF	Value	Prob
Chi-Square	5	3.3139	0.6517
Likelihood Ratio Chi-Square	5	3.2717	0.6582
Mantel-Haenszel Chi-Square	1	0.2342	0.6284
Phi Coefficient		0.2312	
Contingency Coefficient		0.2253	
Cramer's V		0.2312	

WARNING: 58% of the cells have expected counts less than 5. Chi-Square may not be a valid test.

Sample Size = 62

The FREQ Procedure

Table of grade by ethnic

grade ethnic(Apparent ethnic background (ancestry))

Frequency / Expected / Col Pct	Chinese	European	Other	Total
A	0	2	1	3
	1.9839	0.7258	0.2903	
	0.00	13.33	16.67	
B	2	4	0	6
	3.9677	1.4516	0.5806	
	4.88	26.67	0.00	
C	11	5	2	18
	11.903	4.3548	1.7419	
	26.83	33.33	33.33	
D	17	3	1	21
	13.887	5.0806	2.0323	
	41.46	20.00	16.67	
F	7	1	2	10
	6.6129	2.4194	0.9677	
	17.07	6.67	33.33	
Incomplete	4	0	0	4
	2.6452	0.9677	0.3871	
	9.76	0.00	0.00	
Total	41	15	6	62

Statistics for Table of grade by ethnic

Statistic	DF	Value	Prob
Chi-Square	10	18.2676	0.0506
Likelihood Ratio Chi-Square	10	19.6338	0.0329
Mantel-Haenszel Chi-Square	1	5.6222	0.0177
Phi Coefficient		0.5428	
Contingency Coefficient		0.4771	
Cramer's V		0.3838	

WARNING: 78% of the cells have expected counts less than 5. Chi-Square may not be a valid test.

Sample Size = 62

The CORR Procedure

4 Variables: final midterm quizave compave

Simple Statistics

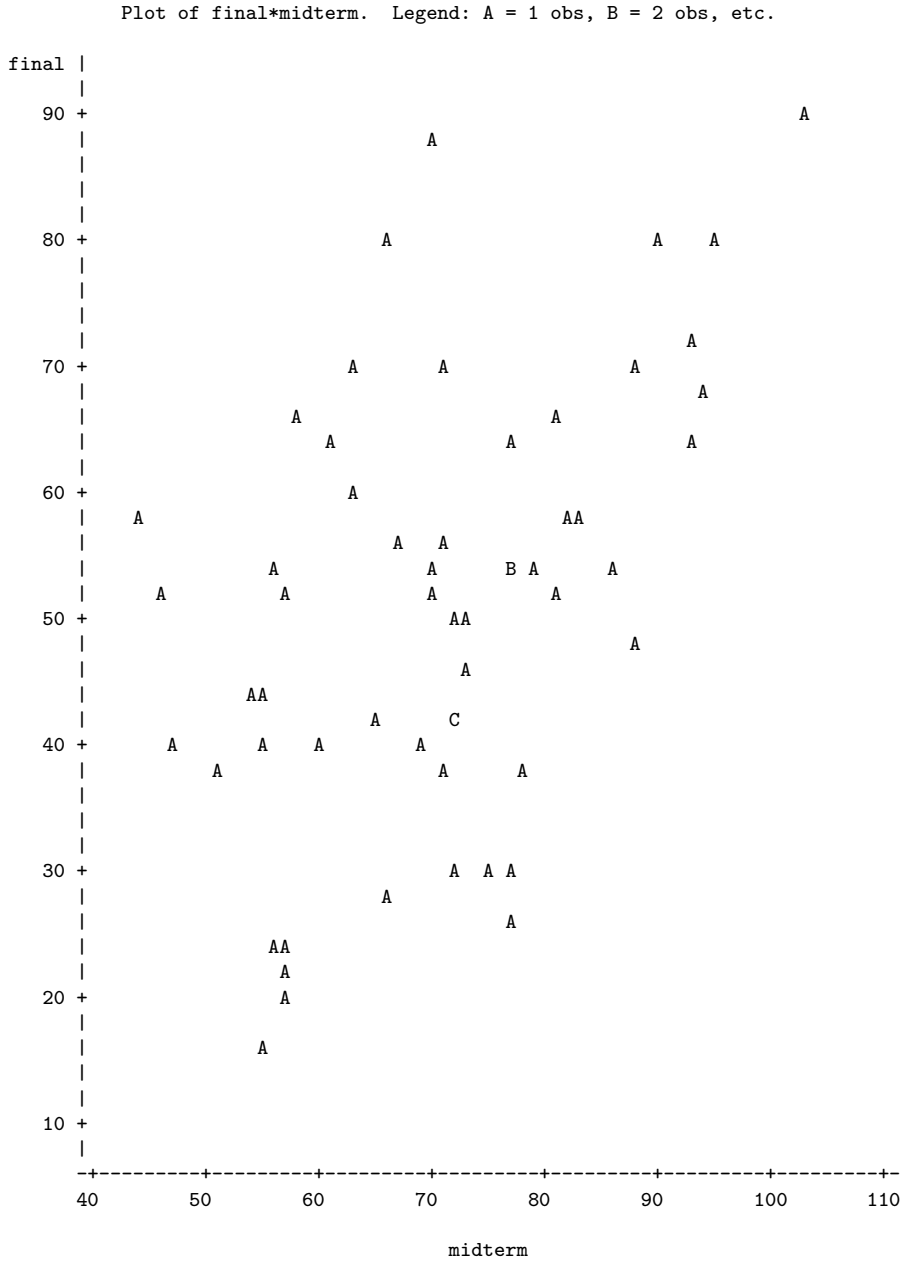
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
final	58	50.31034	17.24967	2918	15.00000	89.00000
midterm	62	70.19355	13.62356	4352	44.00000	103.00000
quizave	62	7.67512	1.12669	475.85714	4.57143	9.71429
compave	62	8.83468	1.12050	547.75000	5.00000	10.00000

Simple Statistics

Variable	Label
final	
midterm	
quizave	Quiz Average (drop lowest)
compave	Computer Average (drop lowest)

Pearson Correlation Coefficients
Prob > |r| under H0: Rho=0
Number of Observations

	final	midterm	quizave	compave
final	1.00000	0.47963	0.41871	0.06060
		0.0001	0.0011	0.6513
	58	58	58	58
midterm	0.47963	1.00000	0.59294	0.41277
	0.0001		<.0001	0.0009
	58	62	62	62
quizave	0.41871	0.59294	1.00000	0.52649
Quiz Average (drop lowest)	0.0011	<.0001		<.0001
	58	62	62	62
compave	0.06060	0.41277	0.52649	1.00000
Computer Average (drop lowest)	0.6513	0.0009	<.0001	
	58	62	62	62



NOTE: 4 obs had missing values.

Simple regression
10:10 Sunday, September 5, 2007

The REG Procedure
Model: MODEL1
Dependent Variable: final

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	3901.64751	3901.64751	16.73	0.0001
Error	56	13059	233.19226		
Corrected Total	57	16960			

Root MSE	15.27063	R-Square	0.2300
Dependent Mean	50.31034	Adj R-Sq	0.2163
Coeff Var	30.35287		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	6.88931	10.80304	0.64	0.5263
midterm	1	0.61605	0.15061	4.09	0.0001

Illustrate Elementary Tests
Multiple Regression
10:10 Sunday, September 5, 2007

The REG Procedure

Descriptive Statistics

Variable	Sum	Mean	Uncorrected SS	Variance	Standard Deviation
Intercept	58.00000	1.00000	58.00000	0	0
midterm	4088.00000	70.48276	298414	180.35935	13.42979
quizave	451.57143	7.78571	3576.51020	1.06498	1.03198
compave	515.50000	8.88793	4641.50000	1.04862	1.02402
final	2918.00000	50.31034	163766	297.55112	17.24967

Descriptive Statistics

Variable	Label
Intercept	Intercept
midterm	
quizave	Quiz Average (drop lowest)

compave Computer Average (drop lowest)
final

The REG Procedure
Model: MODEL1
Dependent Variable: final

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	4995.04770	1665.01590	7.51	0.0003
Error	54	11965	221.58085		
Corrected Total	57	16960			

Root MSE	14.88559	R-Square	0.2945
Dependent Mean	50.31034	Adj R-Sq	0.2553
Coeff Var	29.58754		

Parameter Estimates

Variable	Label	DF	Parameter Estimate	Standard Error
Intercept	Intercept	1	9.01839	19.02591
midterm		1	0.50057	0.18178
quizave	Quiz Average (drop lowest)	1	4.80199	2.46469
compave	Computer Average (drop lowest)	1	-3.53028	2.17562

Parameter Estimates

Variable	Label	DF	t Value	Pr > |t|	Type I SS
Intercept	Intercept	1	0.47	0.6374	146806
midterm		1	2.75	0.0080	3901.64751
quizave	Quiz Average (drop lowest)	1	1.95	0.0566	509.97483
compave	Computer Average (drop lowest)	1	-1.62	0.1105	583.42537

The REG Procedure
Model: MODEL1

```
   Test smalstuf Results for Dependent Variable final

                                  Mean
Source             DF         Square    F Value    Pr > F

Numerator           2      546.70010       2.47    0.0943
Denominator        54      221.58085
```

Data in fixed columns When the data values have at least one space between them, the variables are recorded in the same order for each case, and missing values are indicated by periods, the default version of the `input` statement (list input) does the job perfectly. It is a bonus that the variables need not always be separated by the same number of spaces for each case. Also, there can be more than one line of data for each case, and in fact there need not even be the same number of data lines for all the cases, just as long as there are the same number of variables.

Another common situation is for the data to be lined up in fixed columns, with blanks for missing values. Sometimes, especially when there are many variables, the data are *packed* together, without spaces between values. For example, the Minnesota Multiphasic Personality Inventory (MMPI) consists of over 300 questions, all to be answered True or False. It would be quite natural to code 1=True and 0=False, and pack the data together. There would still be quite a few data lines for each case.

Here is the beginning of the file `statclass2.dat`. It is the same as `statclass1.dat`, except that the data are packed together. Most of the blanks occur because two columns are reserved for the marks on quizzes and computer assignments, because 10 out of 10 is possible. Three columns are reserved for the midterm and final scores, because 100% is possible. For all variables, missing values are represented by blanks. That is, if the field occupied by a variable is completely blank, it's a missing value.

```
appsrv01.srv>  less statclass2.dat
12 9 1 7 8 4 3 5 2 6101010 5 0 0 0 0  55 43
021010 5 910 8 6 81010 8 9 9 9 91010  66 79
121010 5101010 9 8101010101010 91010  94 67
121010 8 910 710 9101010 91010 91010  81 65
0110 1 0 0 8 6 5 210 9 0 010 6 0 5 0  54
1110 6 7 9 8 8 5 710 910 9 5 6 4 810  57 52
01 0 0 9 910 5 2 2 8 7 71010 6 3 710  49
```

```
0110 9 5 8 9 8 5 6 8 7 5 610 6 5 9 9  77 64
0110 8 6 8 9 5 3 6 9 9 6 910 6 5 710  65 42
1110 5 6 710 4 6 010 910 910 6 7 810  73
01 9 0 4 610 5 3 310 810 51010 9 910  71 37

⋮
```

Now we will take a look at `statread.sas`. It contains just the `proc format` and the `data` step; There are no statistical procedures. This file will be read by programs that invoke statistical procedures, as you will see.

```
                              /* statread.sas
Read the statclass data in fixed format, define and label variables. Use
with     %include 'statread.sas';   */

options linesize=79 noovp formdlim='_';
title 'Grades from STA3000 at Roosevelt University:  Fall, 1957';

proc format; /* Used to label values of the categorical variables */
     value sexfmt    0 = 'Male'   1 = 'Female';
     value ethfmt    1 = 'Chinese'
                     2 = 'European'
                     3 = 'Other' ;
data grades;
     infile 'statclass2.dat' missover;
     input (sex ethnic) (1.)
           (quiz1-quiz8 comp1-comp9) (2.)
           (midterm final) (3.);
     /* Drop lowest score for quiz & computer  */
     quizave = ( sum(of quiz1-quiz8) - min(of quiz1-quiz8) ) / 7;
     compave = ( sum(of comp1-comp9) - min(of comp1-comp9) ) / 8;
     label ethnic = 'Apparent ethnic background (ancestry)'
           quizave = 'Quiz Average (drop lowest)'
           compave = 'Computer Average (drop lowest)';
     mark = .3*quizave*10 + .1*compave*10 + .3*midterm + .3*final;
     label mark = 'Final Mark';
     diff = quiz8-quiz1; /* To illustrate matched t-test */
```

```
   label diff = 'Quiz 8 minus Quiz 1';
   mark2 = round(mark);
   /* Bump up at grade boundaries */
   if mark2=89 then mark2=90;
   if mark2=79 then mark2=80;
   if mark2=69 then mark2=70;
   if mark2=59 then mark2=60;
   /* Assign letter grade */
   if mark2=. then grade='Incomplete';
      else if mark2 ge 90 then grade = 'A';
      else if 80 le mark2 le 89 then grade='B';
      else if 70 le mark2 le 79 then grade='C';
      else if 60 le mark2 le 69 then grade='D';
      else grade='F';
   format sex sexfmt.;       /* Associates sex & ethnic    */
   format ethnic ethfmt.;    /* with formats defined above */

/*************************************************************/
```

The data step in `statread.sas` differs from the one in `statmarks1.sas` in only two respects. First, the `missover` option on the infile statement causes blanks to be read as missing values even if they occur at the end of a line and the line just ends rather than being filled in with space characters. That is, such lines are shorter than the others in the file, and when SAS `over`-reads the end of the line, it sets all the variables it would have read to missing. This is what we want, so you should always use the `missover` option when missing values are represented by blanks.

The other difference between this data step and the one in `statmarks1.sas` is in the `input` statement. Here, we are using *formatted* input. `sex` and `ethnic` each occupy 1 column. `quiz1-quiz8` and `comp1-comp9` each occupy 2 columns. `midterm` and `final` each occupy 3 columns. You can supply a list of formats for each list of variables in parentheses, but if the number of formats is less than the number of variables, they are re-used. That's what's happening in the present case.

The program `statread.sas` reads and defines the data, but it requests no statistical output; `statdescribe.sas` pulls in `statread.sas` using a `%include` statement, and produces basic descriptive statistics. Significance tests would be produced by other short programs.

Keeping the data definition in a separate file and using `%include` (the only part of the powerful *SAS macro language* presented here) is often a good strategy, because most data analysis projects involve a substantial number of statistical procedures. It is common to have maybe twenty program files that carry out various analyses. You *could* have the data step at the beginning of each program, but in many cases the data step is long. And, what happens when (inevitably) you want to make a change in the data step and re-run your analyses? You find yourself making the same change in twenty files. Probably you will forget to change some of them, and the result is a big mess. If you keep your data definition in just one place, you only have to edit it once, and a lot of problems are avoided.

```
                                /* statdescribe.sas */
%include 'statread.sas';
title2 'Basic Descriptive Statistics';

proc freq;
      title3 'Frequency distributions of the categorical variables';
      tables sex ethnic grade;

proc means n mean std;
     title3 'Means and SDs of quantitative variables';
     var quiz1 -- mark2;        /*  single dash only works with numbered
                                   lists, like quiz1-quiz8    */

proc univariate normal; /* the normal option gives a test for normality */
     title3 'Detailed look at mark and bumped mark (mark2)';
     var mark mark2;
```

2.2.7 SAS Reference Materials

This course is trying to teach you SAS by example, without full explanation, and certainly without discussion of all the options. If you need more detail, there are several approaches you can take. The most obvious is to consult the SAS manuals. The full set of manuals runs to over a dozen volumes, and most of them look like telephone directories. For a beginner, it is hard to know where to start. And even if you know where to look, the SAS manuals can be hard to read, because they assume you already understand

the statistical procedures fairly thoroughly, and on a mathematical level. They are really written for professional statisticians. The SAS Institute also publishes a variety of manual-like books that are intended to be more instructional, most of them geared to specific topics (like *The SAS system for multiple regression* and *The SAS system for linear models*). These are a bit more readable, though it helps to have a real textbook on the topic to fill in the gaps.

A better place to start is a wonderful book by Cody and Smith [2] entitled *Applied statistics and the SAS programming language*. They do a really good job of presenting and documenting the language of the data step, and and they also cover a set of statistical procedures ranging from elementary to moderately advanced. If you had to own just one SAS book, this would be it.

If you consult *any* SAS book or manual (Cody and Smith's book included), you'll need to translate and filter out some details. Here is the main case. Many of the examples you see in Cody and Smith's book and elsewhere will not have separate files for the raw data and the program. They include the raw data in the program file in the data step, after a `datalines` or `cards` statement. Here is an example from page 3 of [2].

```
data test;
    input subject 1-2 gender $ 4 exam1 6-8 exam2 10-12 hwgrade $ 14;
    datalines;
10 M  80  84 A
 7 M  85  89 A
 4 F  90  86 B
20 M  82  85 B
25 F  94  94 A
14 F  88  84 C
;
proc means data=test;
run;
```

Having the raw data and the SAS code together in one display is so attractive for small data sets that most textbook writers cannot resist it. But think how unpleasant it would be if you had 10,000 lines of data. The way we would do this example is to have the data file (named, say, `example1.dat`) in a separate file. The data file would look like this.

```
10 M  80  84 A
 7 M  85  89 A
 4 F  90  86 B
20 M  82  85 B
25 F  94  94 A
14 F  88  84 C
```

and the program file would look like this.

```
data test;
    infile 'example1.dat'; /* Read data from example1.dat */
    input subject 1-2 gender $ 4 Exam1 6-8 exam2 10-12 hwgrade $ 14;
proc means data=test;
```

Using this as an example, you should be able to translate any textbook example into the program-file data-file format used in this course.

Chapter 3

More Than One Independent Variable at a time

The standard elementary tests typically involve one independent variable and one dependent variable. Now we will see why this can make them very misleading. The lesson you should take away from this discussion is that when important variables are ignored in a statistical analysis — particularly in an observational study — the result can be that we draw incorrect conclusions from the data. Potential confounding variables really need to be included in the analysis.

3.1 The chi-squared test of independence

In order to make sure the central example in this chapter is clear, it may be helpful to give a bit more background on the common Pearson chi-square test of independence. As stated earlier, the chi-square test of independence is for judging whether two categorical variables are related or not. It is based upon a *cross-tabulation*, or *joint frequency distribution* of the two variables. For example, suppose that in the statclass data, we are interested in the relationship between sex and apparent ethnic background. If the ratio of females to males depended upon ethnic background, this could reflect an interesting cultural difference in sex roles with respect to men and women going to university (or at least, taking Statistics classes). In `statmarks1.sas`, we did this test and obtained a chisquare statistic of 2.92 (df=2, $p = 0.2321$), which is not statistically significant. Now we'll do it just a bit differently to

illustrate the details. First, here is the program `ethsex.sas`.

```
/* ethsex.sas */
%include 'statread.sas';
title2 'Sex by Ethnic';
proc freq;
     tables sex*ethnic / chisq norow nocol nopercent expected;
```

And here is the output.

The FREQ Procedure

Table of sex by ethnic

sex ethnic(Apparent ethnic background (ancestry))

Frequency| Expected	Chinese	European	Other	Total
Male	27	7	5	39
	25.79	9.4355	3.7742	
Female	14	8	1	23
	15.21	5.5645	2.2258	
Total	41	15	6	62

Statistics for Table of sex by ethnic

Statistic	DF	Value	Prob
Chi-Square	2	2.9208	0.2321
Likelihood Ratio Chi-Square	2	2.9956	0.2236
Mantel-Haenszel Chi-Square	1	0.0000	0.9949
Phi Coefficient		0.2170	
Contingency Coefficient		0.2121	
Cramer's V		0.2170	

WARNING: 33% of the cells have expected counts less than 5. Chi-Square may not be a valid test.

Sample Size = 62

In each cell of the table, we have an observed frequency and an expected frequency. The expected frequency is the frequency one would expect by chance if the two variables were completely unrelated.[1] If the observed frequencies are different enough from the expected frequencies, one would tend to disbelieve the null hypothesis that the two variables are unrelated. But how should one measure the difference, and what is the meaning of different "enough?"

The Pearson chi-square statistic (named after Karl Pearson, a famous racist, uh, I mean statistician) is defined by

$$\chi^2 = \sum_{\text{cells}} \frac{(f_o - f_e)^2}{f_e}, \tag{3.1}$$

where f_o refers to the observed frequence, f_e refers to expected frequency, and as indicated, the sum is over all the cells in the table.

If the two variables are really independent, then as the total sample size increases, the probability distribution of this statistic approaches a chisquare with degrees of freedom equal to (Number of rows - 1)$\times$(Number of columns - 1). Again, this is an approximate, large-sample result, one that obtains exactly only in the limit as the sample size approaches infinity. A traditional "rule of thumb" is that the approximation is okay if no expected frequency is less than five. This is why SAS gave us a warning.

More recent research suggests that to avoid inflated Type I error (false significance at a rate greater than 0.05), all you need is for no expected frequency to be less than one. You can see from formula (3.1) why an expected frequency less than one would be a problem. Division by a number close to zero can yield a very large quantity even when the observer and expected frequencies are fairly close, and the so-called chisquare value will be seriously inflated.

Anyway, The p-value for the chisquare test is the upper tail area, the area under the chi-square curve beyond the observed value of the test statistic. In the example from the statclass data, the test was not significant and we conclude nothing.

[1]The formula for the expected frequency in a given cell is (row total) $\times$ (column total)/(sample size). This follows from the definition of independent events given in introductory probability: the events A and B are independent if $P(A \cap B) = P(A)P(B)$. But this is too much detail, and we're not going there.

3.2 The Berkeley Graduate Admissions data

Now we're going to look at another example, one that should surprise you. In the 1970's the University of California at Berkeley was accused of discriminating against women in graduate admissions. Data from a large number of applicants are available. The three variables we will consider are sex of the person applying for graduate study, department to which the person applied, and whether or not the person was admitted. First, we will look at the table of sex by admission.

Table of sex by admit

sex	admit		
Frequency|			
Row Pct	No	Yes	Total
Male	1493	1198	2691
	55.48	44.52	
Female	1278	557	1835
	69.65	30.35	
Total	2771	1755	4526

The FREQ Procedure

Statistics for Table of sex by admit

Statistic	DF	Value	Prob
Chi-Square	1	92.2053	<.0001

It certainly looks suspicious. Roughly forty-five percent of the male applicants were admitted, compared to thirty percent of the female applicants. This difference in percentages (equivalent to the relationship between variables here) is highly significant; with $n = 4526$, the p-value is very close to zero.

3.3 Controlling for a variable by subdivision

However, things look different when we take into account the department to which the person applied. Think of a *three-dimensional* table in which the rows are sex, the columns are admission, and the third dimension (call it layers) is department. Such tables are easy to generate with SAS and other statistical packages.

The three-dimensional table is displayed by printing each layer on a separate page, along with test statistics (if requested) for each sub-table. This is equivalent to dividing the cases into sub-samples, and doing the chisquare test separately for each sub-sample. A useful way to talk about this is to say that that we are *controlling* for the third variable; that is, we are looking at the relationship between the other two variables with the third variable held constant. We will have more to say about controlling for collections of independent variables when we get to regression.

Here are the six sub-tables of sex by admit, one for each department, with a brief comment after each table. The SAS output is edited a bit to save paper.

```
      Table 1 of sex by admit
      Controlling for dept=A

sex       admit

Frequency|
Row Pct  |No      |Yes     |  Total
---------+--------+--------+
Male     |    313 |    512 |    825
         |  37.94 |  62.06 |
---------+--------+--------+
Female   |     19 |     89 |    108
         |  17.59 |  82.41 |
---------+--------+--------+
Total         332      601      933


Statistics for Table 1 of sex by admit
       Controlling for dept=A
```

Statistic	DF	Value	Prob
Chi-Square	1	17.2480	<.0001

For department A, 62% of the male applicants were admitted, while 82% of the female applicants were admitted. That is, women were *more* likely to get in than men. This is a *reversal* of the relationship that is observed when the data for all departments are pooled!

Table 2 of sex by admit
Controlling for dept=B

sex admit

Frequency| Row Pct	No	Yes	Total
Male	207	353	560
	36.96	63.04	
Female	8	17	25
	32.00	68.00	
Total	215	370	585

Statistics for Table 2 of sex by admit
Controlling for dept=B

Statistic	DF	Value	Prob
Chi-Square	1	0.2537	0.6145

For department B, women were somewhat more likely to be admitted (another reversal), but it's not statistically significant.

Table 3 of sex by admit
Controlling for dept=C

sex	admit		
Frequency|			
Row Pct	No	Yes	Total
Male	205	120	325
	63.08	36.92	
Female	391	202	593
	65.94	34.06	
Total	596	322	918

Statistics for Table 3 of sex by admit
Controlling for dept=C

Statistic	DF	Value	Prob
Chi-Square	1	0.7535	0.3854

For department C, men were slightly more likely to be admitted, but the 3% difference is much smaller than for the pooled data. Again, it's not statistically significant.

Table 4 of sex by admit
Controlling for dept=D

sex	admit		
Frequency|			
Row Pct	No	Yes	Total
Male	279	138	417
	66.91	33.09	
Female	244	131	375
	65.07	34.93	
Total	523	269	792

Statistics for Table 4 of sex by admit
Controlling for dept=D

Statistic	DF	Value	Prob
Chi-Square	1	0.2980	0.5852

For department D, women were a bit more likely to be admitted (a reversal), but it's far from statistically significant. Now department E:

Table 5 of sex by admit
Controlling for dept=E

sex	admit		
Frequency|			
Row Pct	No	Yes	Total
Male	138	53	191
	72.25	27.75	
Female	299	94	393
	76.08	23.92	
Total	437	147	584

Statistics for Table 5 of sex by admit
Controlling for dept=E

Statistic	DF	Value	Prob
Chi-Square	1	1.0011	0.3171

This time it's a non-significant tendency for men to get in more. Finally, department F:

```
        Table 6 of sex by admit
        Controlling for dept=F

sex       admit

Frequency|
Row Pct  |No      |Yes     |  Total
---------+--------+--------+
Male     |    351 |     22 |    373
         |  94.10 |   5.90 |
---------+--------+--------+
Female   |    317 |     24 |    341
         |  92.96 |   7.04 |
---------+--------+--------+
Total         668       46      714

Statistic                     DF       Value      Prob
------------------------------------------------------
Chi-Square                     1      0.3841    0.5354
```

For department F, women were slightly more likely to get in, but once again it's not significant.

So in summary, the pooled data show that men were more likely to be admitted to graduate study. But when take into account the department to which the student is applying, there is a significant relationship between sex and admission for only one department, and in that department, women are more likely to be accepted.

How could this happen? I generated two-way tables of sex by department and department by admit; both relationships were highly significant. Instead of displaying the SAS output, I have assembled some numbers from these two tables. The same thing could be accomplished with SAS `proc tabulate`, but it's too much trouble, so I did it by hand.

Now it is clear. The two departments with the lowest percentages of female applicants (A and B) also had the highest overall percentage of applicants accepted, while the department with the highest percentage of female applicants (E) also had the second-lowest overall percentage of applicants accepted. That is, the departments most popular with men were easiest to get into, and those most popular with women were more difficult. Clearly,

Table 3.1: Percentage of female applicants and overall percentage of applicants accepted for six departments

Department	Percent applicants female	Percentage applicants accepted
A	11.58%	64.42%
B	4.27	63.25
C	64.60	35.08
D	47.35	33.96
E	67.29	25.17
F	47.76	6.44

this produced the overall tendency for men to be admitted more than women.

By the way, does this mean that the University of California at Berkeley was *not* discriminating against women? By no means. Why does a department admit very few applicants relative to the number who apply? Because they do not have enough professors and other resources to offer more classes. This implies that the departments popular with men were getting more resources, relative to the level of interest measured by number of applicants. Why? Maybe because men were running the show. The "show," by the way definitely includes the U. S. military, which funds a lot of engineering and similar stuff at big American universities.

The Berkeley data, a classic example of *Simpson's paradox*, illustrate the following uncomfortable fact about observational studies. When you include a new variable in an analysis, the results you have could get weaker, they could get stronger, or they could reverse direction — all depending upon the inter-relations of the independent variables. Basically, if an observational study does not include every potential confounding variable you can think of, there is going to be trouble.

Now, the distinguishing feature of the "elementary" tests is that they all involve one independent variable and one dependent variable. Consequently, they can be *extremely* misleading when applied to the data from observational studies, and are best used as tools for preliminary exploration.

Pooling the chi-square tests When using sub-tables to control for a categorical independent variable, it is helpful to have a single test that allows you to answer a question like this: If you control for variable A, is B related

to C? For the chi-square test of independence, it's quite easy. Under the null hypothesis that B is unrelated to C for each value of A, the test statistics for the sub-tables are independent chisquare random variables. Therefore, there sum is also chisquare, with degrees of freedom equal to the sum of degrees of freedom for the sub-tables.

In the Berkeley example, we have a pooled chisquare value of

$$17.2480 + 0.2537 + 0.7535 + 0.2980 + 1.0011 + 0.3841 = 19.9384$$

with 6 degrees of freedom. Using any statistics text (except this one), we can look up the critical value at the 0.05 significance level. It's 12.59; since $19.9 > 12.59$, the pooled test is significant at the 0.05 level. To get a p-value for our pooled chisquare test, we can use SAS. See the program in the next section.

In summary, we need to use statistical methods that incorporate more than one independent variable at the same time; multiple regression is the central example. But even with advanced statistical tools, the most important thing in any study is to collect the right data in the first place. Looking at it the right way is critical too, but no statistical analysis can compensate for having the wrong data.

For more detail on the Berkeley data, see the article in *Science* by Bickel Hammel and O'Connell [1]. For the principle of adding chisquare values and adding degrees of freedom from sub-tables, a good reference is Feinberg's *The analysis of cross-classified categorical data* [4].

3.4 The SAS program

Here is the program `berkeley.sas`. It has several features that you have not seen yet, so a discussion follows the listing of the program.

```
/************************** berkeley.sas ********************************/
options linesize=79 pagesize=35 noovp formdlim='_';
title 'Berkeley Graduate Admissions Data: ';

proc format;
     value sexfmt 1 = 'Female' 0 = 'Male';
     value ynfmt 1 = 'Yes'  0 = 'No';
data berkley;
     input  line sex dept $ admit count;       %$
     format sex sexfmt.; format admit ynfmt.;
     datalines;
   1     0      A      1    512
   2     0      B      1    353
   3     0      C      1    120
   4     0      D      1    138
   5     0      E      1     53
   6     0      F      1     22
   7     1      A      1     89
   8     1      B      1     17
   9     1      C      1    202
  10     1      D      1    131
  11     1      E      1     94
  12     1      F      1     24
  13     0      A      0    313
  14     0      B      0    207
  15     0      C      0    205
  16     0      D      0    279
  17     0      E      0    138
  18     0      F      0    351
  19     1      A      0     19
  20     1      B      0      8
  21     1      C      0    391
  22     1      D      0    244
  23     1      E      0    299
  24     1      F      0    317
;
```

```
proc freq;
     tables sex*admit / nopercent nocol chisq;
     tables dept*sex / nopercent nocol chisq;
     tables dept*admit / nopercent nocol chisq;
     tables dept*sex*admit / nopercent nocol chisq;
     weight count;

/* Get p-value */
proc iml;
     x = 19.9384;
     pval = 1-probchi(x,6);
     print "Chisquare = " x "df=6, p = " pval;
```

The first unusual feature of `berkeley.sas` is in spite of my recommendations to the contrary, the data are in the program itself rather than in a separate file. The data are in the data step, following the `datalines` command and ending with a semicolon. You can always do this, but usually it's a bad idea. Here, it's a good idea. This is why.

I did not have access to a raw data file, only a 2 by 6 by 2 table of sex by department by admission. So I created a data set with just 24 lines, even though there are 4526 cases. Each line of the data set has values for the three variables, and also a variable called `count`, which is simply the observed cell frequency for that combination of sex, department and admission. Then, using the `weight` statement in `proc freq`, I "weighted" each of the 24 cases in the data file by `count`, essentially multiplying the sample size by count for each case.

The advantages are several. First, such a data set is easy to create from published tables, and is much less trouble than a raw data file with thousands of cases. Second, the data file is so short that it makes sense to put it in the data set for portability and ease of reference. Finally, this is the way you can get the data from published tables (which may not include any significance tests at all) into SAS, where you can compute any statistics you want, including sophisticated log-linear modelling analyses.

The last `tables` statement in the `proc freq` gives us the three-dimensional table. For a two-dimensional table, the first variable you mention will correspond to rows and the second will correspond to columns. For higher-dimensional tables, the second-to-last variable mentioned is rows, the last is

columns, and combinations of the variables listed first are the control variables for which sub-tables are produced.

Finally, the `iml` in `proc iml` stands for "Interactive Matrix Language," and you can use it to perform useful calculations in a syntax that is very similar to standard matrix algebra notation; this can be very convenient when formulas you want to compute are in that notation. Here, we're just using it to calculate the area under the curve of the chisquare density with 6 degrees of freedom, beyond the observed test statistic of 19.9384. The `probchi` function is the cumulative distribution function of the chisquare distribution; the second argument (6 in this case) is the degrees of freedom. `probchi(`x`,6)` gives the area under the curve between zero and x, and `1-probchi(`x`,6)` gives the tail area above x – that is, the p-value.

Summary The example of the Berkeley graduate admissions data teaches us that if potential confounding variables are not cancelled out by random assignment to experimental conditions, they need to be explicitly included in a statistical analysis. Otherwise, the results can be very misleading. In the Berkeley example, first we ignored department and there was a relationship between sex and admission that was statistically significant in one direction. Then, when we *controlled* for department — that is, when we took it into account — the relationship was either significant in the opposite direction, or it was not significant (depending on which department).

We also saw how to pool chi-square values and degrees of freedom by adding over sub-tables, obtaining a useful test of whether two categorical variables are related, while controlling for one or more other categorical variables. This is something SAS will not do for you, but it's easy to do with `proc freq` output and a calculator.

Chapter 4

Multiple Regression: Part One

4.1 Three Meanings of Control

In this class, we will use the word **control** to refer to procedures designed to reduce the influence of extraneous variables on our results. The definition of extraneous is "not properly part of a thing," and we will use it to refer to variables we're not really interested in, and which might get in the way of understanding the relationship between the independent variable and the dependent variable.

There are two ways an extraneous variable might get in the way. First, it could be a confounding variable – related to both the independent variable and the dependent variable, and hence capable of creating masking or even reversing relationships that would otherwise be evident. Second, it could be unrelated to the independent variable and hence not a confounding variable, but it could still have a substantial relationship to the dependent variable. If it is ignored, the variation that it could explain will be part of the "background noise," making it harder to see the relationship between IV and DV, or at least causing it to appear relatively weak, and possibly to be non-significant.

The main way to control potential extraneous variables is by holding them constant. In **experimental control**, extraneous variables are literally held constant by the procedure of data collection or sampling of cases. For example, in a study of problem solving conducted at a high school, background noise might be controlled by doing the experiment at the same time of day for each subject (and not when classes are changing). In learning experiments

with rats, males are often employed because their behavior is less variable than that of females.

An alternative to experimental control is **statistical control**, which takes two main forms. One version, **subdivision**, is to subdivide the sample into groups with identical or nearly identical values of the extraneous variable(s), and then to examine the relationship between independent and dependent variable separately in each subgroup – possibly pooling the subgroup analyses in some way. The analysis of the Berkeley graduate admissions data in Chapter 3 is our prime example. As another example where the relationship of interest is between quantitative rather than categorical variables, the correlation of education with income might be studied separately for men and women. The drawback of this subdivision approach is that if extraneous variables have many values or combinations of values, you need a very large sample.

The second form of statistical control, **model-based** control, is to exploit details of the statistical model to accomplish the same thing as the subdivision approach, but without needing a huge sample size. The primary example is multiple linear regression, which is the topic of this chapter.

4.2 Population Parameters

Recall we said two variables are "related" if the distribution of the dependent variable *depends* on the value of the independent variable. Classical regression and analysis of variance are concerned with a particular way in which the independent and dependent variables might be related, one in which the *population mean* of Y depends on the value of X.

Think of a population histogram manufactured out of a thin sheet of metal. The point (along the horizontal axis) where the histogram balances is called the **expected value** or population mean; it is usually denoted by $E[Y]$ or μ (the Greek letter mu). The *conditional* population mean of Y given $X = x$ is just the balance point of the conditional distribution. It will be denoted by $E[Y|X = x]$. The vertical bar | should be read as "given."

Again, for every value of X, there is a separate distribution of Y, and the expected value (population mean) of that distribution depends on the value of X. Furthermore, that dependence takes a very specific and simple form. When there is only one independent variable, the population mean of Y is

$$E[Y|X = x] = \beta_0 + \beta_1 x. \tag{4.1}$$

This is the equation of a straight line. The slope (rise over run) is β_1 and the intercept is β_0. If you want to know the population mean of Y for any given x value, all you need are the two numbers β_0 and β_1.

But in practice, we never know β_0 and β_1. To *estimate* them, we use the slope and intercept of the least-squares line:

$$\widehat{Y} = b_0 + b_1 x. \tag{4.2}$$

If you want to estimate the population mean of Y for any given x value, all you need are the two numbers b_0 and b_1, which are calculated from the sample data.

This has a remarkable implication, one that carries over into multiple regression. Ordinarily, if you want to estimate a population mean, you need a reasonable amount of data. You calculate the sample mean of those data, and that's your estimate of the population mean. If you want to estimate a *conditional* population mean, that is, the population mean of the conditional distribution of Y given a particular $X = x$, you need a healthy amount of data with that value of x. For example, if you want to estimate the average weight of 50 year old women, you need a sample of 50 year old women — unless you are willing to make some assumptions.

What kind of assumptions? Well, the simple structure of (4.1) means that you can use formula (4.2) to estimate the population mean of Y for a given value of $X = x$ *without having any data* at that x value. This is not "cheating," or at any rate, it need not be. If

- the x value in question is comfortably within the range of the data in your sample, and if
- the straight-line model is a reasonable approximation of reality within that range,

then the estimate can be quite good.

The ability to estimate a conditional population mean without a lot of data at any given x value means that we will be able to control for extraneous variables, and remove their influence from a given analysis without having the massive amounts of data required by the subdivision approach to statistical control.

We are getting away with this because we have adopted a *model* for the data that makes reasonably strong assumptions about the way in which the population mean of Y depends on X. If those assumptions are close to the truth, then the conclusions we draw will be reasonable. If the assumptions are badly wrong, we are just playing silly games. There is a general principle here, one that extends far beyond multiple regression.

Data Analysis Hint 4 *There is a direct tradeoff between amount of data and the strength (restrictiveness) of model assumptions. If you have a lot of data, you do not need to assume as much. If you have a small sample size, you will probably have to adopt fairly restrictive assumptions in order to conclude anything from your data.*

Multiple Regression Now consider the more realistic case where there is more than one independent variable. With two independent variables, the model for the population mean of Y is

$$E[Y|\boldsymbol{X} = \boldsymbol{x}] = \beta_0 + \beta_1 x_1 + \beta_2 x_2,$$

which is the equation of a plane in 3 dimensions (x_1, x_2, y). The general case is

$$E[Y|\boldsymbol{X} = \boldsymbol{x}] = \beta_0 + \beta_1 x_1 + \ldots + \beta_{p-1} x_{p-1},$$

which is the equation of a hyperplane in p dimensions.

Comments

- Since there is more than one independent variable, there is a conditional distribution of Y for every *combination* of independent variable values. Matrix notation (boldface) is being used to denote a collection of independent variables.
- There are $p-1$ independent variables. This may seem a little strange, but we're doing this to keep the notation consistent with that of standard regression texts such as [9]. If you want to think of an independent variable $X_0 = 1$, then there are p independent variables.
- What is β_0? It's the height of the population hyperplane when all the independent variables are zero, so it's the *intercept*.

- Most regression models have an intercept term, but some do not ($X_0 = 0$); it depends on what you want to accomplish.
- β_0 is the intercept. We will now see that the other β values are slopes.

Consider

$$E[Y|\boldsymbol{X} = \boldsymbol{x}] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4$$

What is β_3? If you speak calculus, $\frac{\partial}{\partial x_3} E[Y] = \beta_3$, so β_3 is the rate at which the population mean is increasing as a function of x_3, when other independent variables are *held constant* (this is the meaning of a partial derivative).

If you speak high school algebra, β_3 is the change in the population mean of Y when x_3 is increased by one unit and all other independent variables are *held constant*. Look at

$$\begin{array}{rlll} & \beta_0 + \beta_1 x_1 + \beta_2 x_2 & +\beta_3(x_3 + 1) & +\beta_4 x_4 \\ - & (\beta_0 + \beta_1 x_1 + \beta_2 x_2 & +\beta_3 x_3 & +\beta_4 x_4) \end{array}$$

$$\begin{array}{rlll} = & \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 & +\beta_3 & +\beta_4 x_4 \\ - & \beta_0 - \beta_1 x_1 - \beta_2 x_2 - \beta_3 x_3 & & -\beta_4 x_4 \end{array}$$

$$= \beta_3$$

The mathematical device of *holding other variables constant* is very important. This is what is meant by statements like "**Controlling for** parents' education, parents' income and number of siblings, quality of day care is still positively related to academic performance in Grade 1." We have just seen the prime example of model-based statistical control — the third type of control in the "Three meanings of control" section that began this chapter.

We will describe the relationship between X_k and Y as **positive** (controlling for the other independent variables) if $\beta_k > 0$ and **negative** if $\beta_k < 0$.

Here is a useful definition. A quantity (say w) is a **linear combination** of quantities z_1, z_2 and z_3 if $w = a_1 z_1 + a_2 z_2 + a_3 z_3$, where a_1, a_2 and a_3 are constants. Common multiple regression is *linear* regression because the population mean of Y is a linear combination of the β values. It does *not* refer to the shape of the curve relating x to $E[Y|X = x]$. For example,

$E[Y\|X=x]$	$=$	$\beta_0 + \beta_1 x$	Simple linear regression
$E[Y\|X=x]$	$=$	$\beta_0 + \beta_1 x^2$	Also simple linear regression
$E[Y\|X=x]$	$=$	$\beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$	Polynomial regression – still linear
$E[Y\|X=x]$	$=$	$\beta_0 + \beta_1 x + \beta_2 \cos(1/x)$	Still linear in the β values
$E[Y\|X=x]$	$=$	$\beta_0 + \beta_1 \cos(\beta_2 x)$	Truly non-linear

When the relationship between the independent and dependent variables is best represented by a curve, we'll call it **curvilinear**, whether the regression model is linear or not. All the examples just above are curvilinear, except the first one.

Notice that in the polynomial regression example, there is really only one independent variable, x. But in the regression model, x, x^2 and x^3 are considered to be three separate independent variables in a multiple regression. Here, fitting a curve to a cloud of points in two dimensions is accomplished by fitting a hyperplane in four dimensions. The origins of this remarkable trick are lost in the mists of time, but whoever thought of it was having a good day.

4.3 Estimation by least squares

In the last section, the conditional population mean of the dependent variable was modelled as a (population) hyperplane. It is natural to estimate a population hyperplane with a sample hyperplane. This is easiest to imagine in three dimensions. Think of a three-dimensional scatterplot, in a room. The independent variables are X_1 and X_2. The (x_1, x_2) plane is the floor, and the value of Y is height above the floor. Each subject (case) in the sample contributes three coordinates (x_1, x_2, y), which can be represented by a soap bubble floating in the air.

In simple regression, we have a two-dimensional scatterplot, and we seek the best-fitting straight line. In multiple regression, we have a three (or higher) dimensional scatterplot, and we seek the best fitting plane (or hyperplane). Think of lifting and tilting a piece of plywood until it fits the cloud of bubbles as well as possible.

What is the "best-fitting" plane? We'll use the **least-squares plane**, the one that minimizes the sum of squared vertical distances of the bubbles

from the piece of plywood. These vertical distances can be viewed as errors of prediction.

It's hard to visualize in higher dimension, but the algebra is straightforward. Any sample hyperplane may be viewed as an estimate (maybe good, maybe terrible) of the population hyperplane. Following the statistical convention of putting a hat on a population parameter to denote an estimate of it, the equation of a sample hyperplane is

$$\widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \ldots + \widehat{\beta}_{p-1} x_{p-1},$$

and the error of prediction (vertical distance) is the difference between y and the quantity above. So, the least squares plane must minimize

$$Q = \sum_{i=1}^{n} \left(y_i - \widehat{\beta}_0 - \widehat{\beta}_1 x_{i,1} - \ldots - \widehat{\beta}_{p-1} x_{i,p-1} \right)^2$$

over all combinations of $\widehat{\beta}_0, \widehat{\beta}_1, \ldots, \widehat{\beta}_{p-1}$.

Provided that no independent variable (including the peculiar $X_0 = 1$) is a perfect linear combination of the others, the $\widehat{\beta}$ quantities that minimize the sum of squares Q exist and are unique. We will denote them by b_0 (the estimate of β_0, b_1 (the estimate of β_1), and so on.

Again, *a population hyperplane is being estimated by a sample hyperplane.*

$$\begin{aligned} E[Y|\boldsymbol{X} = \boldsymbol{x}] &= \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 \\ \widehat{Y} &= b_0 + b_1 x_1 + b_2 x_2 + b_3 x_3 + b_4 x_4 \end{aligned}$$

- $\widehat{Y}$ means *predicted* Y. It is the height of the best-fitting (least squares) piece of plywood above the floor, at the point represented by the combination of x values. The equation for $\widehat{Y}$ is the equation of the least-squares hyperplane.
- "Fitting the model" means calculating the b values.

4.3.1 Residuals

The **residual**, or error of prediction, is

$$e = Y - \widehat{Y}.$$

The residuals (there are n) represents errors in prediction. A positive residual means over-performance (or under-prediction). A negative residual means

under-performance. Examination of residuals can reveal a lot, since we can't look at 12-dimensional scatterplots.

- Single variable plots (histograms, box plots, stem and leaf diagrams etc.) can identify outliers. (Data errors? Source of new ideas? What might a bimodal distribution of residuals indicate?)
- Plot (scatterplot) of residuals versus potential independent variables not in the model might suggest they be included, or not. How would you plot residuals vs a categorical IV?
- Plot of residuals vs. variables that are in the model may reveal
  - Curvilinear trend (may need transformation of x, or polynomial regression, or even real non-linear regression)
  - Non-constant variance over the range of x, so the DV may depend on the IV not just through the mean. May need transformation of Y, or weighted least squares, or a different model.
- Plot of residuals vs. $\widehat{Y}$ may also reveal unequal variance.

4.3.2 Categorical Independent Variables

Independent variables need not be continuous – or even quantitative. For example, suppose subjects in a drug study are randomly assigned to either an active drug or a placebo. Let Y represent response to the drug, and

$$x = \begin{cases} 1 & \text{if the subject received the active drug, or} \\ 0 & \text{if the subject received the placebo.} \end{cases}$$

The model is $E[Y|X = x] = \beta_0 + \beta_1 x$. For subjects who receive the active drug (so $x = 1$), the population mean is

$$\beta_0 + \beta_1 x = \beta_0 + \beta_1$$

For subjects who receive the placebo (so $x = 0$), the population mean is

$$\beta_0 + \beta_1 x = \beta_0.$$

Therefore, β_0 is the population mean response to the placebo, and β_1 is the difference between response to the active drug and response to the placebo. We are very interested in testing whether β_1 is different from zero, and guess what? We get exactly the same t value as from a two-sample t-test, and exactly the same F value as from a one-way ANOVA for two groups.

Exercise Suppose a study has 3 treatment conditions. For example Group 1 gets Drug 1, Group 2 gets Drug 2, and Group 3 gets a placebo, so that the Independent Variable is Group (taking values 1,2,3) and there is some Dependent Variable Y (maybe response to drug again).

Sample Question 4.3.1 *Why is* $E[Y|X = x] = \beta_0 + \beta_1 x$ *(with* x *= Group) a silly model?*

Answer to Sample Question 4.3.1 *Designation of the Groups as 1, 2 and 3 is completely arbitrary.*

Sample Question 4.3.2 *Suppose* $x_1 = 1$ *if the subject is in Group 1, and zero otherwise, and* $x_2 = 1$ *if the subject is in Group 2, and zero otherwise, and* $E[Y|\boldsymbol{X} = \boldsymbol{x}] = \beta_0 + \beta_1 x_1 + \beta_2 x_2$*. Fill in the table below.*

Group	x_1	x_2	$\beta_0 + \beta_1 x_1 + \beta_2 x_2$
1			$\mu_1 =$
2			$\mu_2 =$
3			$\mu_3 =$

Answer to Sample Question 4.3.2

Group	x_1	x_2	$\beta_0 + \beta_1 x_1 + \beta_2 x_2$
1	1	0	$\mu_1 = \beta_0 + \beta_1$
2	0	1	$\mu_2 = \beta_0 + \beta_2$
3	0	0	$\mu_3 = \beta_0$

Sample Question 4.3.3 *What does each* β *value mean?*

Answer to Sample Question 4.3.3 $\beta_0 = \mu_3$*, the population mean response to the placebo.* β_1 *is the difference between mean response to Drug 1 and mean response to the placebo.* β_2 *is the difference between mean response to Drug 21 and mean response to the placebo.*

Sample Question 4.3.4 *Why would it be nice to simultaneously test whether* β_1 *and* β_2 *are different from zero?*

Answer to Sample Question 4.3.4 *This is the same as testing whether all three population means are equal; this is what a one-way ANOVA does. And we get the same* F *and* p *values (not really part of the sample answer).*

It is worth noting that all the traditional one-way and higher-way models for analysis of variance and covariance emerge as special cases of multiple regression, with dummy variables representing the categorical independent variables.

More about Dummy Variables The exercise above was based on **indicator dummy variables**, which take a value of 1 for observations where a categorical independent variable takes a particular value, and zero otherwise. Notice that x_1 and x_2 contain the same information as the three-category variable Group. If you know Group, you know x_1 and x_2, and if you know x_1 and x_2, you know Group. In models with an intercept term, a categorical independent variable with k categories is always represented by $k-1$ dummy variables. If the dummy variables are indicators, the category that does not get an indicator is actually the most important. The intercept is that category's mean, and it is called the **reference category**, because the remaining regression coefficients represent differences between the reference category and the other category. To compare several treatments to a control, make the control group the reference category by *not* giving it an indicator.

Sample Question 4.3.5 *What would happen if you used k indicator dummy variables instead of $k-1$?*

Answer to Sample Question 4.3.5 *The dummy variables would add up to the intercept; the independent variables would be linearly dependent, and the least-squares estimators would not exist.*

Your software might try to save you by throwing one of the dummy variables out, but which one would it discard?

4.3.3 Explained Variation

Before considering any independent variables, there is a certain amount of variation in the dependent variable. The sample mean is the value around which the sum of squared errors of prediction is at a minimum, so it's a least squares estimate of the population mean of Y when there are no independent variables. We will measure the total variation to be explained by the sum of squared deviations around the mean of the dependent variable.

When we do a regression, variation of the data around the least-squares plane represents errors of prediction. It is variation that is *unexplained* by the regression. But it's always less than the variation around the sample mean (Why? Because the least-squares plane could be horizontal). So, the independent variables in the regression have explained *some* of the variation in the dependent variable. Variation in the residuals is variation that is still *unexplained*.

Variation to explain: **Total Sum of Squares**

$$\text{SSTO} = \sum_{i=1}^{n}(Y_i - \overline{Y})^2$$

Variation that the regression does not explain: **Error Sum of Squares**

$$\text{SSE} = \sum_{i=1}^{n}(e_i - \overline{e})^2 = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2$$

Variation that is explained: **Regression (or Model) Sum of Squares**

$$\text{SSR} = \sum_{i=1}^{n}(Y_i - \overline{Y})^2 - \sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2 = \sum_{i=1}^{n}(\widehat{Y}_i - \overline{Y})^2$$

Regression software (including SAS) displays the sums of squares above in an *analysis of variance summary table*. "Analysis" means to "split up," and that's what we're doing here — splitting up the variation in dependent variable into explained and unexplained parts.

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Prob>F
Model	$p-1$	SSR	$MSR = SSR/(p-1)$	$F = \frac{MSR}{MSE}$	p-value
Error	$n-p$	SSE	$MSE = SSE/(n-p)$		
Total	$n-1$	$SSTO$			

Variance estimates consist of sums of squares divided by degrees of freedom. "DF" stands for Degrees of Freedom. Sums of squares and degrees of freedom each add up to Total. The F-test is for whether $\beta_1 = \beta_2 = \ldots = \beta_{p-1} = 0$ – that is, for whether *any* of the independent variables makes a difference.

The proportion of variation in the dependent variable that is explained by the independent variables (representing *strength of relationship*) is

$$R^2 = \frac{\text{SSR}}{\text{SSTO}}$$

The R^2 from a simple regression is the same as the square of the correlation coefficient: $R^2 = r^2$.

What is a good value of R^2? Well, the weakest relationship I can visually perceive in a scatterplot is around $r = .3$, so I am unimpressed by R^2 values under 0.09. By this criterion, most published results in the social sciences, and many published results in the biological sciences are not strong enough to be scientifically interesting. But this is just my opinion.

4.4 Testing for Statistical Significance in Regression

We are already assuming that there is a separate population defined by each combination of values of the independent variables (the conditional distributions of Y given $\mathbf{X}$), and that the conditional population mean is a linear combination of the β values; the weights of this linear combination are 1 for β_0, and the x values for the other β values. The classical assumptions are that in addition,

- Sample values of Y represent independent observations, conditionally upon the values of the independent variables.
- Each conditional distribution is normal.
- Each conditional distribution has the same population variance.

How important are the assumptions? Well, important for what? The main thing we want to avoid is incorrect p-values, specifically ones that appear smaller than they are. This would increase the chances of concluding that a relationship is present when really it is not. Such "Type I error" is very undesirable, because it tends to load the scientific literature with random garbage.

For large samples, the assumption of normality is not important provided no single observation has too much influence. What is meant by a "large" sample? It depends on how severe the violations are. What is "too much" influence? It's not too much if the influence of the most influential observation tends to zero as the sample size approaches infinity. You're welcome.

The assumption of equal variances can be safely violated provided that the numbers of observations at each combination of IV values are large and

close to equal. This is most likely to be the case with designed experiments having categorical independent variables.

The assumption of independent observations is very important, almost always. Examples where this does not hold is if a student takes a test more than once, members of the same family respond to the same questionnaire about eating habits, litter-mates are used in a study of resistance to cancer in mice, and so on.

When you know in advance which observations form non-independent sets, one option is to average them, and let n be the number of independent sets of observations. There are also ways to incorporate non-independence into the statistical model. We will discuss repeated measures designs, multivariate analysis and other examples later.

4.4.1 The standard F and t-tests

SAS `proc reg` (like other programs) usually starts with an overall F-test, which tests all the independent variables in the equation simultaneously. If this test is significant, we can conclude that one or more of the independent variables is related to the dependent variable.

Again like most programs that do multiple regression, SAS produces t-tests for the individual regression coefficients. If one of these is significant, we conclude that controlling for all other independent variables in the model, the independent variable in question is related to the dependent variable. That is, each variable is tested controlling for all the others.

It is also possible to test subsets of independent variables, controlling for all the others. For example, in an educational assessment where students use 4 different textbooks, the variable "textbook" would be represented by 3 dummy variables. These variables could be tested simultaneously, controlling for several other variables such as parental education and income, child's past academic performance, experience of teacher, and so on.

In general, to test a subset A of independent variables while controlling for another subset B, fit a model with both sets of variables, and simultaneously test the b coefficients of the variables in subset A; there is an F test for this.

This is 100% equivalent to the following. Fit a model with just the variables in subset B, and calculate R^2. Then fit a second model with the A variables as well as the B variables, and calculate R^2 again. Test whether the increase in R^2 is significant. It's the same F test.

Call the regression model with all the independent variables the **Full Model**, and call the model with fewer independent variables (that is, the model without the variables being tested) the **Reduced Model**. Let SSR_F represent the explained sum of squares from the full model, and SSR_R represent the explained sum of squares from the reduced model.

Sample Question 4.4.1 *Why is $SSR_F \geq SSR_R$?*

Answer to Sample Question 4.4.1 *In the full model, if the best-fitting hyperplane had all the b coefficients corresponding to the extra variables equal to zero, it would fit exactly as well as the hyperplane of the reduced model. It could not do any worse.*

Since $R^2 = \frac{SSR}{SSTO}$, it is clear that $SSR_F \geq SSR_R$ implies that adding independent variables to a regression model can only increase R^2. When these additional independent variables are correlated with independent variables already in the model (as they usually are in an observational study),

- Statistical significance can appear when it was not present originally, because the additional variables reduce error variation, and make estimation and testing more precise.
- Statistical significance that was originally present can disappear, because the new variables explain some of the variation previously attributed to the variables that were significant, so when one controls for the new variables, there is not enough explained variation left to be significant. This is especially true of the t-tests, in which each variable is being controlled for all the others.
- Even the signs of the bs can change, reversing the interpretation of how their variables are related to the dependent variable. This is why it's very important not to leave out important independent variables in an observational study.

Suppose the full model contains all the variables in the reduced model, plus s additional variables. These are the ones you are testing. The F-test for the full versus reduced model is based on the test statistic

$$F = \frac{(SSR_F - SSR_R)/s}{MSE_F}, \tag{4.3}$$

where MSE_F is the mean square error for the full model: $MSE_F = \frac{SSE_F}{n-p}$. Equation 4.3 is a very general formula. As we will see, all the standard tests in regression and the usual (fixed effects) Analysis of Variance are special cases of this F-test.

Examples of Full and Reduced Models

At this point, it might help to have some concrete examples. Recall the SENIC data set (catching infection in hospital) that was used to illustrate a collection of elementary significance tests in lecture. For reference, here is the `label` statement again.

```
label id       = 'Hospital identification number'
      stay     = 'Av length of hospital stay, in days'
      age      = 'Average patient age'
      infrisk  = 'Prob of acquiring infection in hospital'
      culratio = '# cultures  / # no hosp acq infect'
      xratio   = '# x-rays / # no signs of pneumonia'
      nbeds    = 'Average # beds during study period'
      medschl  = 'Medical school affiliation'
      region   = 'Region of country (usa)'
      census   = 'Aver # patients in hospital per day'
      nurses   = 'Aver # nurses during study period'
      service  = '% of 35 potential facil. & services' ;
```

The SAS program `senicread.sas` could have defined dummy variables for `region` and `medschl` in the data step as follows:

```
if region = 1 then r1=1; else r1=0;
if region = 2 then r2=1; else r2=0;
if region = 3 then r3=1; else r3=0;
if medschl = 2 then mschool = 0; else mschool = medschl;
/* mschool is an indicator for medical school = yes */
```

The definition of `r1, r2` and `r2` above is correct, but it is risky. It works only because the data file happens to have no missing values for region. If there were missing values for region, the `else` statements would assign them to zero for `r1, r2` and `r3`, because `else` means *anything* else. The definition of `mschool` is a bit more sophisticated; missing values for `medschl` will also be missing for `mschool`.

Here is what I'd recommend for `region`. It's more trouble, but it's worth it.

```
/* Indicator dummy variables for region */
if region = . then r1=.;
   else if region = 1 then r1 = 1;
   else r1 = 0;
if region = . then r2=.;
   else if region = 2 then r2 = 1;
   else r2 = 0;
if region = . then r3=.;
   else if region = 3 then r3 = 1;
   else r3 = 0;
```

When you create dummy variables with `if` statements, always do cross-tabulations of the new dummy variables by the categorical variable they represent, to make sure you did it right. Use the option of `proc freq` to see what happened to the missing values (`missprint` makes "missing" a value of the variables).

```
proc freq;
     tables region * (r1-r3) / missprint nocol norow nopercent ;
```

Sample Question 4.4.2 *Controlling for hospital size as represented by number of beds and number of patients, is average patient age related to infection risk?*

1. *What are the variables in the full model?*
2. *What are the variables in the reduced model?*

Answer to Sample Question 4.4.2

1. `nbeds, census, age`
2. `nbeds, census`

I would never ask for SAS syntax on a test, but for completeness,

```
proc reg;
    model infrisk = nbeds, census, age;
    size: test age=0;
```

Sample Question 4.4.3 *Controlling for average patient age and hospital size as represented by number of beds and number of patients, does infection risk differ by region of the country?*

1. *What are the variables in the full model?*
2. *What are the variables in the reduced model?*

Answer to Sample Question 4.4.3

1. `age, nbeds, census, r1, r2, r3`
2. `age, nbeds, census`

To test the full model versus the reduced model,

```
proc reg;
    model infrisk = age nbeds census r1 r2 r3;
    regn: test r1=r2=r3=0;
```

Sample Question 4.4.4 *Controlling for number of beds, number of patients, average length of stay and region of the country, are number of nurses and medical school affiliation (considered simultaneously) significant predictors of infection risk?*

1. *What are the variables in the full model?*
2. *What are the variables in the reduced model?*

Answer to Sample Question 4.4.4

1. `nbeds, census, stay, r1, r2, r3, nurses, mschool`
2. `nbeds, census, stay, r1, r2, r3`

To test the full model versus the reduced model,

```
proc reg;
    model infrisk = nbeds census stay r1 r2 r3 nurses mschool;
    nursmeds: test nurses=mschool=0;
```

Sample Question 4.4.5 *Controlling for average age of patient, average length of stay and region of the country, is hospital size (as represented by number of beds and number of patients) related to infection risk?*

1. *What are the variables in the full model?*
2. *What are the variables in the reduced model?*

Answer to Sample Question 4.4.5

1. `age, stay, r1, r2, r3, nbeds, census`
2. `age, stay, r1, r2, r3`

To test the full model versus the reduced model,

```
proc reg;
    model infrisk = nbeds census stay r1 r2 r3 nurses mschool;
    size2: test nurses=mschool=0;
```

Sample Question 4.4.6 *Controlling for region of the country and medical school affiliation, are average length of stay and average patient age (considered simultaneously) related to infection risk?*

1. *What are the variables in the full model?*
2. *What are the variables in the reduced model?*

Answer to Sample Question 4.4.6

1. `r1, r2, r3, mschool, stay age`
2. `r1, r2, r3, mschool`

To test the full model versus the reduced model,

```
proc reg;
    model infrisk = nbeds census stay r1 r2 r3 nurses mschool;
    agestay: test age=stay=0;
```

The pattern should be clear. You are "controlling for" the variables in the reduced model. You are testing for the additional variables that appear in the full model but not the reduced model.

Looking at the Formula for F

Formula 4.3 reveals some important properties of the F-test. Bear in mind that the p-value is the area under the F-distribution curve *above* the value of the F statistic. Therefore, anything that makes the F statistic bigger will make the p-value smaller, and if it is small enough, the results will be significant. And significant results are what we want, if in fact the full model is closer to the truth than the reduced model.

- Since there are s more variables in the full model than in the reduced model, the numerator of (4.3) is the *average* improvement in explained sum of squares when we compare the full model to the reduced model. Thus, some of the extra variables might be useless for prediction, but the test could still be significant at least one of them contributes a lot to the explained sum of squares, so that the *average* increase is substantially more than one would expect by chance.

- On the other hand, useless extra independent variables can dilute the contribution of extra independent variables with modest but real explanatory power.

- The denominator is a variance estimate based on how spread out the residuals are. The smaller this denominator is, the larger the F statistic is, and the more likely it is to be significant. Therefore, *control* all the sources of extraneous variation you can.

  - If possible, always collect data on any potential independent variable that is known to have a strong relationship to the dependent variable, and include it in both the full model and the reduced model. This will make the analysis more sensitive, because increasing the explained sum of squares will reduce the unexplained sum of squares. You will be more likely to detect a real result as significant, because it will be more likely to show up against the reduced background noise.

  - On the other hand, the denominator of formula (4.3) for F is $MSE_F = \frac{SSE_F}{n-p}$, where the number of independent variables is $p-1$. Adding useless independent variables to the model will increase the explained sum of squares by at least a little, but the

denominator of MSE_F will go down by one, making MSE_F bigger, and F smaller. The smaller the sample size n, the worse the effect of useless independent variables. You have to be selective.

- The (internal) validity of most experimental research depends on experimental designs and procedures that balance sources of extraneous variation evenly across treatments. But even better are careful experimental procedures that eliminate random noise altogether, or at least hold it to very low levels. Reduce sources of random variation, and the residuals will be smaller. The MSE_F will be smaller, and F will be bigger if something is really going on.
- Most dependent variables are just indirect reflections of what the investigator would really like to study, and in designing their studies, scientists routinely make decisions that are tradeoffs between expense (or convenience) and data quality. When dependent variables represent low-quality measurement, they essentially contain random variation that cannot be explained. This variation will show up in the denominator of (4.3), reducing the chance of detecting real results against the background noise. An example of a dependent variable that might have too much noise would be a questionnaire or subscale of a questionnaire with just a few items.

The comments above sneaked in the topic of **statistical power** by discussing the formula for the F-test. Statistical power is *the probability of getting significant results when something is really going on in the population.* It should be clear that high power is good. We have just seen that statistical power can be increased by including important explanatory variables in the study, by carefully controlled experimental conditions, and by quality measurement. Power can also be increased by increasing the sample size. All this is true in general, and does not depend on the use of the traditional F test.

4.4.2 Connections between Explained Variation and Significance Testing

If you divide numerator and denominator of Equation (4.3) by $SSTO$, the numerator becomes $(R_F^2 - R_R^2)/s$, so we see that the F test is based on change

in R^2 when one moves from the reduced model to the full model. But the F test for the extra variables (controlling for the ones in the reduced model) is based not just on $R^2_F - R^2_R$, but on a quantity I'll denote by $a = \frac{R^2_F - R^2_R}{1 - R^2_R}$. This expresses change in R^2 as a *proportion* of the variation left unexplained by the reduced model. That is, it's the *proportion of remaining variation* that the additional variables explain.

This is actually a more informative quantity than simple change in R^2. For example, suppose you're controlling for a set of variables that explain 80% of the variation in the dependent variable, and you test a variable that accounts for an additional 5%. You have explained 25% of the remaining variation – much more impressive than 5%.

The a notation is non-standard. It's sometimes called a squared multiple partial correlation, but the usual notation for partial correlations is intricate and hard to look at, so we'll just use a.

You may recall that an F test has two degree of freedom values, a numerator degrees of freedom and a denominator degrees of freedom. In the F test for a full versus reduced model, the numerator degrees of freedom is s, the number of extra variables. The denominator degrees of freedom is $n - p$. Recall that the sample size is n, and if the regression model has an intercept, there are $p - 1$ independent variables. Applying a bit of high school algebra to Equation (4.3), we see that the relationship between F and a is

$$F = \left(\frac{n-p}{s}\right)\left(\frac{a}{1-a}\right). \tag{4.4}$$

so that for any given sample size, the bigger a becomes, the bigger F is. Also, for a given value of $a \neq 0$, F increases as a function of n. This means you can get a large F (and if it's large enough it will be significant) from strong results and a small sample, *or* from weak results and a large sample. Again, examining the formula for the F statistic yields a valuable insight.

Expression (4.4) for F can be turned around to express a in terms of F, as follows:

$$a = \frac{sF}{n - p + sF} \tag{4.5}$$

This is a useful formula, because scientific journals often report just F values, degrees of freedom and p-values. It's easy to tell whether the results are significant, but not whether the results are strong in the sense of explained variation. But the equality (4.5) above lets you recover information about

strength of relationship from the F statistic and its degrees of freedom. For example, based on a three-way ANOVA where the dependent variable is rot in potatoes, suppose the authors write "The interaction of bacteria by temperature was just barely significant (F=3.26, df=2,36, p=0.05)." What we want to know is, once one controls for other effects in the model, what proportion of the remaining variation is explained by the temperature-by-bacteria interaction?

We have s=2, $n - p = 36$, and $a = \frac{2 \times 3.26}{36 + (2 \times 3.26)} = 0.153$. So this effect is explaining a respectable 15% of the variation that remains after controlling for all the other main effects and interactions in the model.

4.5 Multiple Regression with SAS

It is always good to start with a textbook example, so that interested students can locate a more technical discussion of what is going on. The following example is based on the "Dwaine Studios" Example from Chapter 6 of Neter et al.'s textbook [9]. The observations correspond to photographic portrait studios in 21 towns. In addition to sales (the dependent variable), the data file contains number of children 16 and younger in the community (in thousands of persons), and per capita disposable income in thousands of dollars. Here is the SAS program.

```
/* appdwaine1.sas */
options linesize=79;
title 'Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al';
title2 'Just the defaults';

data portrait;
     infile 'dwaine.dat';
     input kids income sales;
proc reg;
     model sales = kids income;
/*   model DV(s) = IV(s);           */
```

Here is the list file `appdwaine1.lst`.

```
       Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al      1
                            Just the defaults 15:52 Sunday, January 13, 2007

Model: MODEL1
Dependent Variable: SALES

                           Analysis of Variance

                                Sum of          Mean
      Source            DF      Squares        Square      F Value       Prob>F

      Model              2  24015.28211  12007.64106       99.103       0.0001
      Error             18   2180.92741    121.16263
      C Total           20  26196.20952

          Root MSE      11.00739     R-square       0.9167
          Dep Mean     181.90476     Adj R-sq       0.9075
          C.V.           6.05118

                          Parameter Estimates

                    Parameter      Standard    T for H0:
     Variable  DF      Estimate         Error   Parameter=0    Prob > |T|

     INTERCEP   1    -68.857073   60.01695322        -1.147        0.2663
     KIDS       1      1.454560    0.21178175         6.868        0.0001
     INCOME     1      9.365500    4.06395814         2.305        0.0333
```

Here are some comments on the list file.

- First the ANOVA summary table for the overall F-test, testing all the independent variables simultaneously. In `C Total`, `C` means corrected for the sample mean. The p-value of 0.0001 actually means $p < 0.0001$, in this version of SAS. It's better in later versions.
- `Root MSE` is the square root of MSE.
- `Dep Mean` is the mean of the dependent variable.
- `C.V.` is the coefficient of variation – the standard deviation divided by the mean. Who cares?
- `R-square` is R^2
- `Adj R-sq`: Since R^2 never goes down when you add independent variables, models with more variables always look as if they are doing better. Adjusted R^2 is an attempt to penalize the usual R^2 for the

number of independent variables in the model. It can be useful if you are trying to compare the predictive usefulness of models with different numbers of variables.

- `Parameter Estimates` are the b values. `Standard Error` is the (estimated) standard deviation of the sampling distribution of b. It's the denominator of the t test in the next column.
- The last column is a two-tailed p-value for the t-test.

Here are some sample questions based on the list file.

Sample Question 4.5.1 *Suppose we wish to test simultaneously whether number of kids 16 and under and average family income have any relationship to sales. Give the value of the test statistic, and the associated p-value.*

Answer to Sample Question 4.5.1 $F = 99.103$, $p < 0.0001$

Sample Question 4.5.2 *What can you conclude from just this one test?*

Answer to Sample Question 4.5.2 *Sales is related to either number of kids 16 and under, or average family income, or both. But you'd never do this. You have to look at the rest of the printout to tell what's happening.*

Sample Question 4.5.3 *What percent of the variation in sales is explained by number of kids 16 and under and average family income?*

Answer to Sample Question 4.5.3 *91.67%*

Sample Question 4.5.4 *Controlling for average family income, is number of kids 16 and under related to sales?*

1. *What is the value of the test statistic?*
2. *What is the p-value?*
3. *Are the results significant? Answer Yes or No.*
4. *Is the relationship positive, or negative?*

Answer to Sample Question 4.5.4

1. $t = 6.868$
2. $p < 0.0001$
3. Yes.
4. Positive.

Sample Question 4.5.5 *Controlling for number of kids 16 and under is average family income related to sales?*

1. *What is the value of the test statistic?*
2. *What is the p-value?*
3. *Are the results significant? Answer Yes or No.*
4. *Is the relationship positive, or negative?*

Answer to Sample Question 4.5.5

1. $t = 2.305$
2. $p = 0.0333$
3. Yes.
4. Positive.

Sample Question 4.5.6 *What do you conclude from this entire analysis? Direct your answer to a statistician or researcher.*

Answer to Sample Question 4.5.6 *Number of kids 16 and under and average family income are both related to sales, even when each variable is controlled for the other.*

Sample Question 4.5.7 *What do you conclude from this entire analysis? Direct your answer to a person without statistical training.*

Answer to Sample Question 4.5.7 *Even when you allow for the number of kids 16 and under in a town, the higher the average family income in the town, the higher the average sales. When you allow for the average family income in a town, the higher the number of children under 16, the higher the average sales.*

Sample Question 4.5.8 *A new studio is to be opened in a town with 65,400 children 16 and under, and an average household income of $17,600. What annual sales do you predict?*

Answer to Sample Question 4.5.8 $\widehat{Y} = b_0 + b_1 x_1 + b_2 x_2$ *= -68.857073 + 1.454560*65.4 + 9.365500*17.6 = 191.104, so predicted annual sales = $191,104.*

Sample Question 4.5.9 *For any fixed value of average income, what happens to predicted annual sales when the number of children under 16 increases by one thousand?*

Answer to Sample Question 4.5.9 *Predicted annual sales goes up by $1,454.*

Sample Question 4.5.10 *What do you conclude from the t-test for the intercept?*

Answer to Sample Question 4.5.10 *Nothing. Who cares if annual sales equals zero for towns with no children under 16 and an average household income of zero?*

The final two questions ask for a proportion of remaining variation, the quantity we are denoting by a. If you were doing an analysis yourself and wanted this statistic, you'd likely fit a full and a reduced model (or obtain sequential sums of squares; we'll see how to do this in the next example), and calculate the answer directly. But in the published literature, sometimes all you have are reports of t-tests for regression coefficients.

Sample Question 4.5.11 *Controlling for average household income, what proportion of the remaining variation is explained by number of children under 16?*

Answer to Sample Question 4.5.11 *Using* $F = t^2$ *and plugging into (4.5), we have* $a = \frac{1 \times 6.868^2}{21-3+1 \times 6.868^2}$ *= 0.691944, or around 70% of the remaining variation.*

Sample Question 4.5.12 *Controlling for number of children under 16, what proportion of the remaining variation is explained by average household income?*

Answer to Sample Question 4.5.12 $a = \frac{2.305^2}{18+2.305^2} = 0.2278994$, *or about 23%.*

These a values are large, but the sample size is small; after all, it's a textbook example, not real data. Now here is a program file that illustrates some options, and gives you a hint of what a powerful tool SAS can be.

```
/* appdwaine2.sas */
options linesize=79 pagesize=35;
title 'Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al';
title2 'With bells and whistles';

data portrait;
     infile 'dwaine.dat';
     input kids income sales;

proc reg simple corr;     /* "simple" prints simple descriptive statistics */
     model sales = kids income / ss1;     /* "ss1" prints Sequential SS */
     output out=resdata predicted=presale  residual=resale;
     /* Creates new SAS data set with Y-hat and e as additional variables*/
     /* Now all the default F-tests, in order */
           allivs: test kids = 0, income = 0;
           inter:  test intercept=0;
           child:  test kids=0;
           money:  test income=0;

proc iml; /* Income controlling for kids: Full vs reduced by "hand" */
     fcrit = finv(.95,1,18); print fcrit;
     /* Had to look at printout from an earlier run to get these numbers*/
     f = 643.475809 / 121.16263; /* Using the first F formula */
     pval = 1-probf(f,1,18);
     tsq = 2.305**2; /* t-squared should equal F*/
     a = 643.475809/(26196.20952 - 23372);
     print f tsq pval;
     print "Proportion of remaining variation is " a;

proc glm; /* Use proc glm to get a y-hat more easily */
     model sales=kids income;
```

```
     estimate 'Xh p249'  intercept 1   kids 65.4   income 17.6;

proc print; /* To see the new data set with residuals*/
proc univariate normal plot;
     var resale;
proc plot;
     plot resale * (kids income sales);
```

Here are some comments on `appdwaine2.sas`.

- `simple corr` You could get means and standard deviations from `proc means` and correlations from `proc corr`, but this is convenient.
- `ss1` These are Type I Sums of Squares, produced by default in `proc glm`. In `proc reg`, you must request them is you want to see them. The independent variables in the `model` statement are added to the model in order, so that for each variable, the reduced model has all the variables that come before it, and the full model has all those variables *plus* the current one. The `ss1` option shows the *increase* in explained sum of squares that comes from adding each variable to the model, in the order they appear in the `model` statement.
- `output` creates a new sas data set called `resdata`. It has all the variables in the data set `portrait`, and in addition it has $\widehat{Y}$ (named `presale` for predicted sales) and e (named `resale` for residual of sales).
- Then we have some custom tests, all of them equivalent to what we would get by testing a full versus reduced model. SAS takes the approach of testing whether s linear combinations of β values equal s specified constants (usually zero). Again, this is the same thing as testing a full versus a reduced model. The form of a custom test in `proc reg` is
  1. A name for the test, 8 characters or less, followed by a colon; this name will be used to label the output.
  2. the word `test`.
  3. s linear combinations of independent variable names, each set equal to some constant, separated by commas.
  4. A semi-colon to end, as usual.

If you want to think of the significance test in terms of a collection of linear combinations that specify constraints on the β values (this is what a statistician would appreciate), then we would say that the names of the independent variables (including the weird variable "intercept") are being used to refer to the corresponding βs. But usually, you are testing a subset of independent variables controlling for some other subset. In this case, include all the variables in the `model` statement, and set the variables you are testing equal to zero in the `test` statement. Commas are optional. As an example, for the test `allivs` (all independent variables) we could have written `allivs: test kids = income = 0;`.

- Now suppose you wanted to use the Sequential Sums of Squares to test `income` controlling for `kids`. You could use a calculator and a table of the F distribution from a textbook, but for larger sample sizes the exact denominator degrees of freedom you need are seldom in the table, and you have to interpolate in the table. With `proc iml` (Interactive Matrix Language), which is actually a nice programming environment, you can use SAS as your calculator. Among other things, you can get exact critical values and p-values quite easily. Statistical tables are obsolete.

  In this example, we first get the **critical value** for F; *if the test statistic is bigger than the critical value, the result is significant.* Then we calculate F using formula 4.3 and its p-value. This F should be equal to the square of the t statistic from the printout, so we check. Then we use (4.5) to calculate a, and print the results.

- `proc glm` The `glm` procedure is very useful when you have categorical independent variables, because it makes your dummy variables for you. But it also can do multiple regression. This example calls attention to the `estimate` command, which lets you calculate $\widehat{Y}$ values more easily and with less chance of error than with a calculator or `proc iml`.

- `proc print` prints all the data values, for all the variables. This is a small data set, so it's not producing a telephone book here. You can limit the variables and the number of cases it prints; see the manual or *Applied statistics and the SAS programming language* [2]. By default, all SAS procedures use the most recently created SAS data set; this is `resdata`, which was created by `proc reg` – so the predicted values and residuals will be printed by `proc print`.

- You didn't notice, but `proc glm` also used `resdata` rather than `portrait`. But it was okay, because `resdata` has all the variables in `portrait`, and *also* the predicted Y and the residuals.

- `proc univariate` produces a lot of useful descriptive statistics, along with a fair amount of junk. The `normal` option gives some tests for normality, and textttplot generates some line-printer plots like boxplots and stem-and-leaf displays. These are sometimes informative. It's a good idea to run the residuals (from the full model) through `proc univariate` if you're starting to take an analysis seriously.

- `proc plot` This is how you would plot residuals against variables in the model. It the data file had additional variables you were *thinking* of including in the analysis, you could plot them against the residuals too, and look for a correlation. My personal preference is to start plotting residuals fairly late in the exploratory game, once I am starting to get attached to a regression model.

Here is the list file `appdwaine2.lst`.

```
 Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al      1
                       With bells and whistles
                                    10:58 Saturday, January 19, 2007

                       Descriptive Statistics

Variables                 Sum                Mean      Uncorrected SS

INTERCEP                   21                   1                  21
KIDS                     1302.4         62.019047619             87707.94
INCOME                     360         17.142857143              6190.26
SALES                     3820          181.9047619             721072.4


          Variables            Variance       Std Deviation

          INTERCEP                    0                  0
          KIDS             346.71661905        18.620328113
          INCOME           0.9415714286        0.9703460355
          SALES            1309.8104762        36.191303875


                              Correlation

   CORR                KIDS             INCOME             SALES

   KIDS              1.0000             0.7813            0.9446
   INCOME            0.7813             1.0000            0.8358
   SALES             0.9446             0.8358            1.0000
```

Model: MODEL1
Dependent Variable: SALES

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Prob>F
Model	2	24015.28211	12007.64106	99.103	0.0001
Error	18	2180.92741	121.16263		
C Total	20	26196.20952			

Root MSE	11.00739	R-square	0.9167
Dep Mean	181.90476	Adj R-sq	0.9075
C.V.	6.05118		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	T for H0: Parameter=0	Prob > |T|
INTERCEP	1	-68.857073	60.01695322	-1.147	0.2663
KIDS	1	1.454560	0.21178175	6.868	0.0001
INCOME	1	9.365500	4.06395814	2.305	0.0333

Variable	DF	Type I SS
INTERCEP	1	694876
KIDS	1	23372
INCOME	1	643.475809

Dependent Variable: SALES
Test: ALLIVS Numerator: 12007.6411 DF: 2 F value: 99.1035
Denominator: 121.1626 DF: 18 Prob>F: 0.0001

Dependent Variable: SALES
Test: INTER Numerator: 159.4843 DF: 1 F value: 1.3163
Denominator: 121.1626 DF: 18 Prob>F: 0.2663

Dependent Variable: SALES
Test: CHILD Numerator: 5715.5058 DF: 1 F value: 47.1722
Denominator: 121.1626 DF: 18 Prob>F: 0.0001

Dependent Variable: SALES

```
Test: MONEY    Numerator:    643.4758  DF:    1   F value:   5.3108
               Denominator:  121.1626  DF:   18   Prob>F:    0.0333

       Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al
4
                           With bells and whistles
                                          10:58 Saturday, January 19, 2007

                                    FCRIT
                                4.4138734


                                    F       TSQ      PVAL
                            5.3108439  5.313025 0.0333214


                                                                    A
                 Proportion of remaining variation is  0.2278428

       Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al
5
                           With bells and whistles
                                          10:58 Saturday, January 19, 2007

                        General Linear Models Procedure

                    Number of observations in data set = 21

       Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al
6
                           With bells and whistles
                                          10:58 Saturday, January 19, 2007

                        General Linear Models Procedure

Dependent Variable: SALES
                                      Sum of            Mean
Source                   DF          Squares          Square  F Value    Pr > F

Model                     2     24015.282112    12007.641056    99.10    0.0001

Error                    18      2180.927411      121.162634

Corrected Total          20     26196.209524

                  R-Square             C.V.        Root MSE         SALES Mean

                  0.916746         6.051183       11.007390          181.90476


Source                   DF        Type I SS     Mean Square  F Value    Pr > F

KIDS                      1     23371.806303    23371.806303   192.90    0.0001
INCOME                    1       643.475809      643.475809     5.31    0.0333

Source                   DF      Type III SS     Mean Square  F Value    Pr > F
```

```
KIDS                    1     5715.5058347    5715.5058347    47.17    0.0001
INCOME                  1      643.4758090     643.4758090     5.31    0.0333
        Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al
7
                           With bells and whistles
                                        10:58 Saturday, January 19, 2007

                       General Linear Models Procedure

Dependent Variable: SALES

                                      T for H0:    Pr > |T|   Std Error of
Parameter                  Estimate    Parameter=0                Estimate

Xh p249                  191.103930          69.07     0.0001     2.76679783


                                      T for H0:    Pr > |T|   Std Error of
Parameter                  Estimate    Parameter=0                Estimate

INTERCEPT              -68.85707315          -1.15     0.2663    60.01695322
KIDS                     1.45455958           6.87     0.0001     0.21178175
INCOME                   9.36550038           2.30     0.0333     4.06395814


        Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al      8
                           With bells and whistles
                                        11:32 Tuesday, January 15, 2007

             OBS    KIDS    INCOME    SALES    PRESALE      RESALE

               1    68.5      16.7    174.4    187.184    -12.7841
               2    45.2      16.8    164.4    154.229     10.1706
               3    91.3      18.2    244.2    234.396      9.8037
               4    47.8      16.3    154.6    153.329      1.2715
               5    46.9      17.3    181.6    161.385     20.2151
               6    66.1      18.2    207.5    197.741      9.7586
               7    49.5      15.9    152.8    152.055      0.7449
               8    52.0      17.2    163.2    167.867     -4.6666
               9    48.9      16.6    145.4    157.738    -12.3382
              10    38.4      16.0    137.2    136.846      0.3540
              11    87.9      18.3    241.9    230.387     11.5126
              12    72.8      17.1    191.1    197.185     -6.0849
              13    88.4      17.4    232.0    222.686      9.3143
              14    42.9      15.8    145.3    141.518      3.7816
              15    52.5      17.8    161.1    174.213    -13.1132
              16    85.7      18.4    209.7    228.124    -18.4239
              17    41.3      16.5    146.4    145.747      0.6530
              18    51.7      16.3    144.0    159.001    -15.0013
              19    89.6      18.1    232.6    230.987      1.6130
              20    82.7      19.1    224.1    230.316     -6.2161
              21    52.3      16.0    166.5    157.064      9.4356
        Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al      9
                           With bells and whistles
9
                           With bells and whistles
                                        11:41 Saturday, January 19, 2007
```

Univariate Procedure

Variable=RESALE Residual

Moments

N	21	Sum Wgts	21
Mean	0	Sum	0
Std Dev	10.44253	Variance	109.0464
Skewness	-0.09705	Kurtosis	-0.79427
USS	2180.927	CSS	2180.927
CV	.	Std Mean	2.278746
T:Mean=0	0	Pr>|T|	1.0000
Num ^= 0	21	Num > 0	13
M(Sign)	2.5	Pr>=|M|	0.3833
Sgn Rank	1.5	Pr>=|S|	0.9599
W:Normal	0.955277	Pr<W	0.4190

Quantiles(Def=5)

100% Max	20.21507	99%	20.21507
75% Q3	9.435601	95%	11.51263
50% Med	0.744918	90%	10.17057
25% Q1	-6.21606	10%	-13.1132
0% Min	-18.4239	5%	-15.0013
		1%	-18.4239
Range	38.63896		
Q3-Q1	15.65166		
Mode	-18.4239		

Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al 1
0

With bells and whistles

11:41 Saturday, January 19, 2007

Univariate Procedure

Variable=RESALE Residual

Extremes

Lowest	Obs	Highest	Obs
-18.4239(	16)	9.758578(	6)
-15.0013(	18)	9.803676(	3)
-13.1132(	15)	10.17057(	2)
-12.7841(	1)	11.51263(	11)
-12.3382(	9)	20.21507(	5)

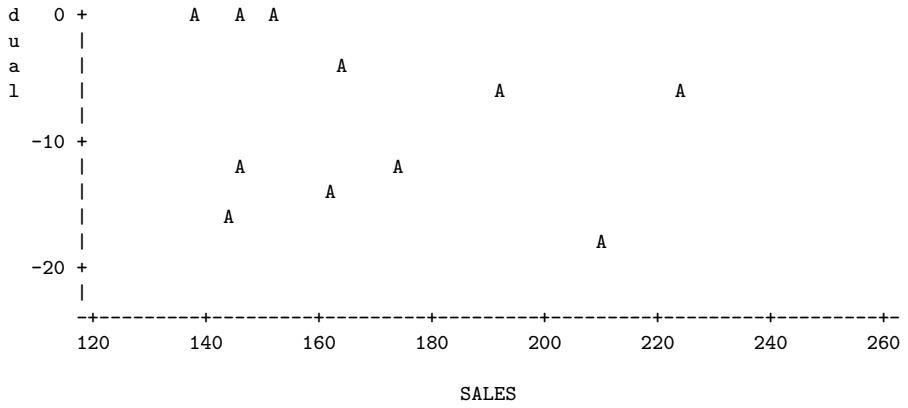
Stem	Leaf	#	Boxplot
2	0	1	|
1			|
1	0002	4	|
0	99	2	+-----+
0	011124	6	*--+--*
-0			| |
-0	665	3	+-----+

```
               -1 332                     3               |
               -1 85                      2               |
                  ----+----+----+----+
              Multiply Stem.Leaf by 10**+1

       Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al     1
1
                           With bells and whistles
                                          11:41 Saturday, January 19, 2007

                            Univariate Procedure

Variable=RESALE        Residual

                               Normal Probability Plot
            22.5+                                                 *****
                |                                            +++++
                |                                       +++++
                |                                  ******
             2.5+                          *******
                |                         ++++
                |                    +++++
                |              +++++ *
            -17.5+       ******
                 +----+----+----+----+----+----+----+----+----+----+----+
                     -2        -1         0        +1        +2
9

                           With bells and whistles
                                          11:32 Tuesday, January 15, 2007

          Plot of RESALE*KIDS.  Legend: A = 1 obs, B = 2 obs, etc.

      |
   20 +                A
      |
      |
      |
      |                                                          A
   10 +              A      A             A                      A  A
R     |
e     |
s     |            A
i     |                A                                          A
d   0 +        A  A        A
u     |
a     |                     A
l     |                                        A         A
      |
  -10 +
      |                  A                  A
      |                      A
      |                     A
      |                                                    A
  -20 +
      |
      -+---------+---------+---------+---------+---------+---------+---------+-
```

```
      30        40        50        60        70        80        90        100

                                    KIDS
```

```
        Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al     12
                           With bells and whistles
                                                 11:32 Tuesday, January 15, 2007

          Plot of RESALE*INCOME.  Legend: A = 1 obs, B = 2 obs, etc.

        |
     20 +                                A
        |
        |
        |
        |                                                    A
     10 +           A             A         A              B
  R     |
  e     |
  s     |       A
  i     |                 A                                A
  d   0 +         A A         A
  u     |
  a     |                                 A
  l     |                                 A                                  A
        |
    -10 +
        |                     AA
        |                                           A
        |                 A
        |                                                    A
    -20 +
        |
        ---+-------+-------+-------+-------+-------+-------+-------+-------+--
         15.5     16.0     16.5     17.0     17.5     18.0     18.5     19.0     19.5

                                    INCOME
```

```
        Dwaine Studios Example from Chapter 6 (Section 6.9) of Neter et al     13
                           With bells and whistles
                                                 11:32 Tuesday, January 15, 2007

          Plot of RESALE*SALES.  Legend: A = 1 obs, B = 2 obs, etc.

        |
     20 +                                A
        |
        |
        |
        |                                                          A
     10 +                      AA                    A            A     A
  R     |
  e     |
  s     |             A
  i     |                 A                                      A
```



Here are some comments.

- `proc reg`
  - In the descriptive statistics produced by the `simple` option, one of the "variables" is `INTERCEP`; it's our friend $X_0 = 1$. The SAS programmers (or the statisticians directing them) are really thinking of this as an independent variable.
  - The Type I (sequential) sum of squares starts with `INTERCEP`, and a really big number for the explained sum of squares. Well, think of a reduced model that does not even have an intercept — that is, one in which there are not only no independent variables, but the population mean is zero. Then add an intercept, so the full model is $E[Y] = \beta_0$. The least squares estimate of β_0 is $\overline{Y}$, so the improvement in explained sum of squares is $\sum_{i=1}^{n}(Y_i - \overline{Y})^2 = SSTO$. That's the first line. It makes sense, in a twisted way.
  - Then we have the custom tests, which reproduce the default tests, in order. See how useful the *names* of the custom tests can be?
- `proc iml`: Everything works as advertised. $F = t^2$ except for rounding error, and a is exactly what we got as the answer to Sample Question 4.5.12.
- `proc glm`

- After an overall test, we get tests labelled `Type I SS` and `Type III SS`. As mentioned earlier, Type One sums of squares are sequential. Each variable is added in turn to the model, in the order specified by the model statement. Each one is tested controlling for the ones that precede it.
- When independent variables are correlated with each other and with the dependent variable, some of the variation in the dependent variable is being explained by the variation *shared* by the correlated independent variables. Which one should get credit? If you use sequential sums of squares, the variable named first *by you* gets all the credit. And your conclusions can change radically as a result of the order in which you name the independent variables. This may be okay, if you have strong reasons for testing A controlling for B and not the other way around.

  In Type Three sums of squares, each variable is controlled for *all* the others. This way, nobody gets credit for the overlap. It's conservative, and valuable. Naturally, the last lines of Type I and Type III summary tables are identical, because in both cases, the last variable named is being controlled for all the others.
- I can never remember what Type II and Type IV sums of squares are.
- The `estimate` statement yielded an `Estimate`, that is, a $|widehatY$ value, of 191.103930, which is what we got with a calculator as the answer to Sample Question 4.5.8. We also get a t-test for whether this particular linear combination differs significantly from zero — insane in this particular case, but useful at other times. The standard error would be very useful if we were constructing confidence intervals or prediction intervals around the estimate, but we are not.
- Then we get a display of the b values and associated t-tests, as in `proc reg`. I believe these are produced by `proc glm` only when none of the independent variables is declared categorical with the `class` statement.

- `proc print` output is self-explanatory. If you are using `proc print` to print a large number of cases, consider specifying a large page size in the `options` statement. Then, the *logical* page length will be very

long, as if you were printing on a long roll of paper, and SAS will not print a new page header with the date and title and so on every 24 line or 35 lines or whatever.

- `proc univariate`: There is so much output to explain, I almost can't stand it. I'll do most of it in class, and just hit a few high points here.
  - `T:Mean=0` A t-test for whether the mean is zero. If the variable consisted of difference scores, this would be a matched t-test. Here, because the mean of residuals from a multiple regression is *always* zero as a by-product of least-squares, t is exactly zero and the p-value is exactly one.
  - `M(Sign)` Sign test, a non-parametric equivalent to the matched t.
  - `Sgn Rank` Wilcoxon's signed rank test, another non-parametric equivalent to the matched t.
  - `W:Normal` A test for normality. As you might infer from `Pr<W`, the associated p-value*lower* tail area of some distribution. If $p < 0.05$, conclude that the data are not normally distributed.

    The assumptions of the hypothesis tests for multiple regression imply that the residuals are normally distributed, though not quite independent. The lack of independence makes the W test a bit too likely to indicate lack of normality. If the test is non-significant, can one conclude that the data *are* normal? This is an example of a more general question: When can one conclude that the null hypothesis is true?

    To answer this question "Never" is just plain stupid, but still I don't want to go there right now. Instead, just two comments:
    - Like most tests, the W test for normality is much more sensitive when the sample size is large. So failure to observe a significant departure from normality does not imply that the data really are normal, for a small sample like this one (n=21).
    - In an observational study, residuals can appear non-normal because important independent variables have been omitted from the full model.
  - `Extremes` are the 5 highest and 5 lowest scores. Very useful for locating outliers. The largest residual in this data set is 20.21507; it's observation 5.

  - `Normal Probability Plot` is supposed to be straight-line if the data are normal. Even though I requested pagesize=35, this plot is pretty squashed. Basically it's useless.

- `proc plot` Does not show much of anything in this case. This is basically good news, though again the data are artificial. The default plotting symbol is `A`; if two points get too close together, they are plotted as `B`, and so on.

Here are a few sample questions.

Sample Question 4.5.13 *What is the mean of the average household incomes of the 21 towns?*

Answer to Sample Question 4.5.13 *$17,143*

Sample Question 4.5.14 *Is this the same as the average income of all the households in the 21 towns?*

Answer to Sample Question 4.5.14 *No way.*

Sample Question 4.5.15 *The custom test labelled `MONEY` is identical to what default test?*

Answer to Sample Question 4.5.15 *The t-test for `INCOME`. $F = t^2$, and the p-value is the same.*

Sample Question 4.5.16 *In the `proc iml` output, what can you learn from comparing `F` to `FCRIT`?*

Answer to Sample Question 4.5.16 $p < 0.05$

Sample Question 4.5.17 *For a town with 68,500 children 16 and under, and an average household income of $16,700, does the full model over-predict or under-predict sales? By how much?*

Answer to Sample Question 4.5.17 *Under-predict by $12,784. This is the first residual produced by `proc print`.*

Chapter 5

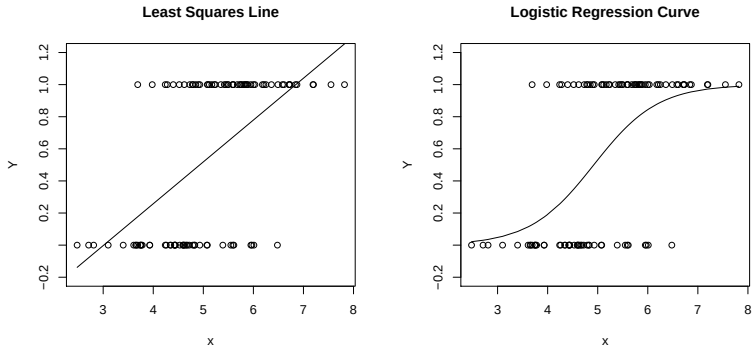
Logistic Regression

In logistic regression, there is s categorical dependent variables, often coded 1=Yes and 0=No. Many important phenomena fit this framework. The patient survives the operation, or does not. The accused is convicted, or is not. The customer makes a purchase, or does not. The marriage lasts at least five years, or does not. The student graduates, or does not.

As usual, we assume that there is a huge population, with a sizable subpopulation at each x value or configuration of x values. And as in ordinary regression, we want a regression surface that consists of the estimated subpopulation mean (conditional expected value) at each x value or configuration of x values. It turns out that for any dependent variable coded zero or one, this conditional mean is exactly the conditional *probability* that $Y = 1$ given that set of x values. Again, for binary data, the population mean is just the probability of getting a one. And since it's a probability, it must lie between zero and one inclusive.

Consider the scatterplot of a single quantitative independent variable and a dependent variable Y equal to zero or one. The left panel of Figure 5.1 shows what happens when we fit a least squares line to such data. It may be reasonable in some sense, but because it is sometimes less than zero and sometimes greater than one, it can't be a probability and it's *not* yielding a sensible estimate of the conditional population mean. However, the logistic regression curve in the right panel stays nicely between zero and one. And like the least-squares line, it indicates a positive relationship for this particular data set.

Figure 5.1: Scatterplots with a binary dependent variable



5.1 A linear model for the log odds

The logistic regression curve arises from an indirect representation of the probability of $Y = 1$ for a given set of x values. Representing the probability of an event by π (it's a probability, not 3.14159...), we define the *odds* of the event as

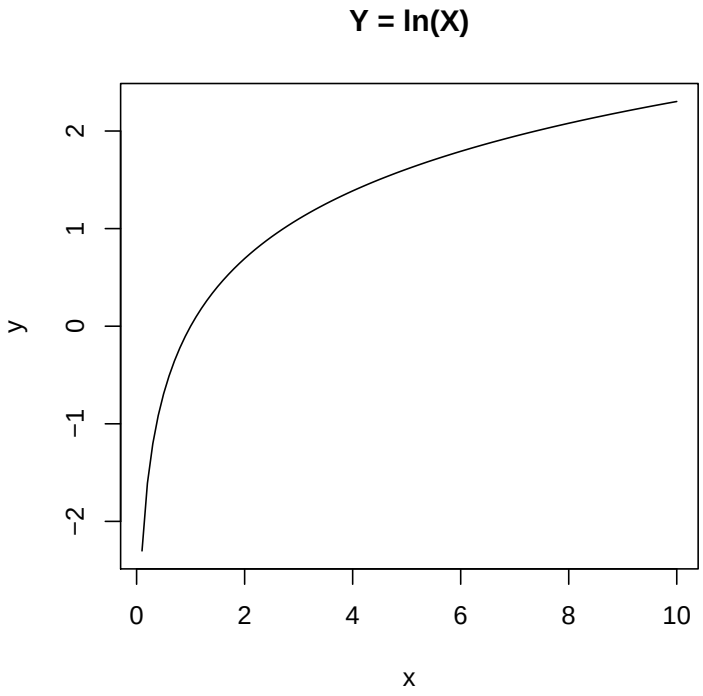
$$\text{Odds} = \frac{\pi}{1-\pi}.$$

Implicitly, we are saying the odds are $\frac{\pi}{1-\pi}$ "to one." That is, if the probability of the event is $\pi = 2/3$, then the odds are $\frac{2/3}{1/3} = 2$, or two to one. Instead of saying the odds are 5 to 2, we'd say 2.5 to one. Instead of saying 1 to four, we'd say 0.25 to one.

The higher the probability, the greater the odds. And as the probability of an event approaches one, the denominator of the odds approaches zero. This means the odds can be anything from zero to an arbitrarily large positive number. Logistic regression adopts a regression-like linear model not for the probability of the event $Y = 1$ nor for the odds, but for the *log odds*. By log we mean the natural or Napierian log, designated by ln on scientific calculators – not the common log base 10. Here are a few necessary facts about the natural log function.

- Figure 5.2 shows that the natural log increases from minus infinity when the odds are zero, to zero when the odds equal one (fifty-fifty),

Figure 5.2: Graph of the natural log function



and then it keeps on increasing as the odds rise, but more and more slowly.

- The fact that the log function is increasing means that if $P(A) > P(B)$, then $\text{Odds}(A) > \text{Odds}(B)$, and therefore $\ln(\text{Odds}(A)) > \ln(\text{Odds}(B))$. That is, the bigger the probability, the bigger the log odds.

- Notice that the natural log is only defined for positive numbers. This is usually fine, because odds are always positive or zero. But if the odds are zero, then the natural log is either minus infinity or undefined – so the methods we are developing here will not work for events of probability exactly zero or exactly one. What's wrong with a probability of one? You'd be dividing by zero when you calculated the odds.

- The natural log is the inverse of exponentiation, meaning that $\ln(e^x) = e^{\ln(x)} = x$, where e is the magic non-repeating decimal number 2.71828.... The number e really is magical, appearing in such seemingly diverse places as the mathematical theory of epidemics, the theory of compound interest, and the normal distribution.

- The log of a product is the sum of logs: $\ln(ab) = \ln(a) + \ln(b)$, and $\ln(\frac{a}{b}) = \ln(a) - \ln(b)$. This means the log of an odds *ratio* is the difference between the two log odds quantities.

To get back to the main point, we adopt a linear regression model for the log odds of the event $Y = 1$. Keeping the notation consistent with ordinary regression, we have $p-1$ independent variables, and the population model is

$$\ln\left(\frac{P(Y=1)}{P(Y=0)}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_{p-1} x_{p-1}.$$

The interpretation of regression coefficients is similar to what we have seen in ordinary regression. β_0 is the intercept in the log odds world. It's the log odds of $Y = 1$ when all independent variables equal zero. And β_k is the increase in log odds of $Y = 1$ when x_k is increased by one unit, and all other independent variables are held constant.

This is on the scale of log odds. But frequently, people choose to think in terms of plain old odds rather than log odds. Skipping a bit of algebra that is not too difficult but really not the point of this course, we have this conclusion: *When x_k is increased by one unit, and all other independent variables are held constant, the odds of $Y = 1$ are multiplied by e^{β_k}.* That is, e^{β_k} is an *odds ratio* — the ratio of the odds of $Y = 1$ when x_k is increased by one unit, to the odds of $Y = 1$ when everything is left alone. As in ordinary regression, we speak of "controlling" for the other variables.

As in ordinary regression, categorical independent variables may be represented by collections of dummy variables, and interactions by product terms. But be a little cautious; parallel slopes on the log odds scale translates to *proportional* odds – like the odds of $Y = 1$ for Group 1 are always 1.3 times the odds of $Y = 1$ for Group, regardless of the value of x. And product terms measure departure from proportional odds. As for what is happening on the scale of raw probability, that is difficult to put into words. All in all, the meaning of interactions between quantitative and categorical independent variables is a little strange in logistic regression.

The linear model for the log odds of $Y = 1$ is equivalent to a distinctly *non-linear* model for the probability of $Y = 1$. Actually, it's a conditional probability, given the values of the independent variables.

$$P(Y = 1|x_1, \ldots, x_{p-1}) = \frac{e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}}}{1 + e^{\beta_0+\beta_1 x_1+\ldots+\beta_{p-1}x_{p-1}}} \tag{5.1}$$

Formula (5.1) can be quite handy for producing a *predicted* probability of $Y = 1$ based on sample data. One simply replaces the population regression coefficients by their estimates. This brings us to the topic of estimation.

5.2 Parameter Estimation by Maximum likelihood

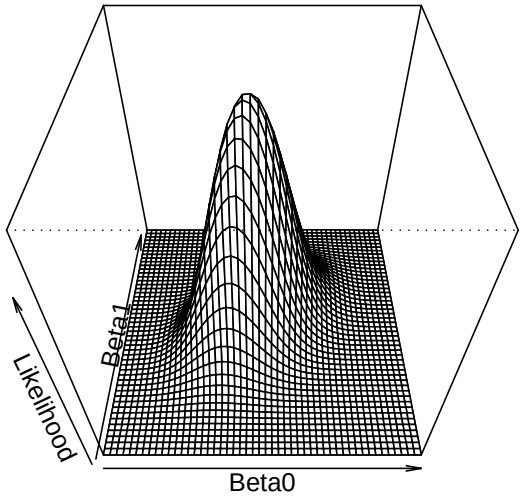
Using formula 5.1, it is possible to calculate the probability of observing the data we did observe, given any set of β values. One of R. A. Fisher's many good suggestions was to take as our estimates of β_0, β_1 and so forth, those values that made the probability of getting the data we actually observed as large as possible. Viewed as a function of the parameter values, the probability that we will get the data we actually did get is called the *likelihood*. The parameter values that make this thing as big as possible are called *maximum likelihood estimates*.

Figure 5.3 is a picture of this for one independent variable. The β_0, β_1 values located right under the peak is our set of maximum likelihood estimates. Of course it's hard to visualize in higher dimension, but the idea is the same.

In regular regression, maximum likelihood estimates are identical to least squares estimates, but not here (though they are close for large samples). Also, the $\widehat{\beta}$ quantities can be calculated by an explicit formula for regular regression, while for logistic regression they need to be found numerically. That is, a program like SAS must calculate the likelihood function for a bunch of sets of β values, and somehow find the top of the mountain. Numerical routines for maximum likelihood estimation essentially march uphill until they find a place where it is downhill in every direction. Then they stop.

For some statistical methods, the place you find this way could be a so-called "local maximum," something like the top of a foothill. You don't know you're not at the top of the highest peak, because you're searching

Figure 5.3: Graph of the Likelihood Function for Simple Logistic Regression



blindfolded, just walking uphill and hoping for the best. Fortunately, this cannot happen with logistic regression. There is only one peak, and no valleys. Start anywhere, walk uphill, and when it levels off you're at the top. This is true regardless of the particular data values and the number of independent variables.

5.3 Chi-square tests

As in regular regression, you can test hypotheses by comparing a full, or unrestricted model to a reduced, or restricted model. Typically the reduced model is the same as the full, except that's it's missing one or more independent variables. But the reduced model may be restricted in other ways, for

example by setting a collection of regression coefficients equal to one another, but not necessarily equal to zero.

There are many ways to test hypotheses in logistic regression; most are large-sample chi-square tests. Two popular ones are likelihood ratio tests and Wald tests.

5.3.1 Likelihood ratio tests

Likelihood ratio tests are based on a direct comparison of the likelihood of the observed data assuming the full model to the likelihood of the data assuming the reduced model. Let $\mathcal{L}_F$ stand for the maximum probability (likelihood) of the observed data under the full model, and $\mathcal{L}_R$ stand for the maximum probability of the observed data under the reduced model. Dividing the latter quantity by the former yields a *likelihood ratio*: $\frac{\mathcal{L}_R}{\mathcal{L}_F}$. It is the maximum probability of obtaining the sample data under the reduced model (null hypothesis), *relative* to the maximum probability of obtaining the sample data under the null hypothesis under the full, or unrestricted model.

As with regular regression, the model cannot fit the data better when it is more restricted, so the likelihood of the reduced model is always less than the likelihood of the full model. If it's a *lot* less – that is, if the observed data are a lot less likely assuming the reduced model than assuming the full model – then this is evidence against the null hypothesis, and perhaps the null hypothesis should be rejected.

Well, if the likelihood ratio is small, then the natural log of the likelihood ratio is a big negative number, and minus the natural log of the likelihood ratio is a big positive number. So is twice minus the natural log of the likelihood ratio. It turns out that if the null hypothesis is true and the sample size is large, then the quantity

$$G = -2 \ln \left(\frac{\mathcal{L}_R}{\mathcal{L}_F} \right)$$

has an approximate chi-square distribution, with degrees of freedom equal to the number of non-redundant restrictions that the null hypothesis places on the set of β parameters. For example, if three egression coefficients are set to zero under the null hypotheses, the degrees of freedom equal three.

5.3.2 Wald tests

You may recall that the Central Limit Theorem says that even when data come from a non-normal distribution, the sampling distribution of the sample mean is approximately normal for large samples. The Wald tests are based on a kind of Central Limit Theorem for maximum likelihood estimates. Under very general conditions that include logistic regression, a collection of maximum likelihood estimates has an approximate multivariate normal distribution, with means approximately equal to the parameters, and variance covariance matrix that has a complicated form, but can be calculated (or approximated as a by-product of the most common types of numerical maximum likelihood).

This was discovered and proved by Abraham Wald, and is the basis of the Wald tests. It is pretty remarkable that he was able to prove this even for maximum likelihood estimates with no explicit formula. Wald was quite a guy. Anyway, if the null hypothesis is true, then a certain sum of squares of the maximum likelihood estimates has a large sample chi-square distribution. The degrees of freedom are the same as for the likelihood ratio tests, and for large enough sample sizes, the numerical values of the two tests statistics get closer and closer.

SAS makes it convenient to do Wald tests and inconvenient to do most likelihood ratio tests, so we'll stick to the Wald tests in this course.

Chapter 6

Multiple Comparisons (Follow-up Tests)

6.1 A One-way Example

The following is a textbook example taken from Neter et al.'s (1996) *Applied linear statistical models* [9]. The Kenton Food Company is interested in testing the effect of different package designs on sales. Five grocery stores were randomly assigned to each of four package designs. The package designs used either three or five colours, and either had cartoons or did not. Because of a fire in one of the stores, there were only four stores in the 5-colour cartoon condition.

The dependent variable is sales, defined as number of cases sold. Actually, there are two independent variables: number of colours and presence versus absence of cartoons. But we will initially consider package design as a single categorical independent variable with four values.

Sample Question 6.1.1 *If there is a statistically significant relationship between package design and sales, would we be justified in concluding that differences in package design* caused *differences in sales?*

Answer to Sample Question 6.1.1 *Yes, if the study is carried out properly. It's an experimental study.*

Sample Question 6.1.2 *Is there a problem with external validity here?*

Answer to Sample Question 6.1.2 *It's impossible to tell for sure, but there easily could be. The behaviour of the sales force would have to be controlled somehow. A double blind would be ideal.*

The SAS program `kenton.sas` does a lot of things, starting with a one-way ANOVA using `proc glm`. The strategy will be to first present the entire program, and then go through it piece by piece and explain what is going on – with a few major digressions to explain the statistics.

```
/********************** kenton.sas *************************/
options linesize=79 pagesize=100 noovp formdlim=' ';
title 'Kenton Oneway Example From Neter et al.';

proc format;
     value pakfmt 1 = '3Colour Cartoon'   2 = '3Col No Cartoon'
                  3 = '5Colour Cartoon'   4 = '5Col No Cartoon';

data food;
     infile 'kenton.dat';
     input package sales;
     label package = 'Package Design'
           sales   = 'Number of Cases Sold';
     format package pakfmt.;

     /* Define ncolours and cartoon */
     if package = 1 or package = 2 then ncolours = 3;
        else if package = 3 or package = 4 then ncolours = 5;
     if package = 1 or package = 3 then cartoon = 'No ';
        else if package = 2 or package = 4 then cartoon = 'Yes';

     /* Indicator Coding for package: Use 3 at a time */
     if package = . then p1 = .; else if package = 1 then p1 = 1;
        else p1 = 0;
     if package = . then p2 = .; else if package = 2 then p2 = 1;
        else p2 = 0;
     if package = . then p3 = .; else if package = 3 then p3 = 1;
        else p3 = 0;
     if package = . then p4 = .; else if package = 4 then p4 = 1;
```

```
        else p4 = 0;

/* Basic one-way ANOVA -- well, not very basic */

proc glm;
     class package;
     model sales = package;
     means package;
     means package / bon tukey scheffe;
     /* Test some custom contrasts */
     contrast '3Colourvs5Colour' package 1  1 -1 -1;
     contrast 'Cartoon'          package 1 -1  1 -1;
     contrast 'CartoonDepends'   package 1 -1 -1  1;
     /* Test a collection of contrasts */
     contrast 'Overall F'        package 1 -1  0  0,
                                 package 0  1 -1  0,
                                 package 0  0  1 -1;
     /* Test effects of Colour and Cartoons simultaneously, allowing for
        a possible interaction */
     contrast 'ColorCartoon'     package 1  1 -1 -1,
                                 package 1 -1  1 -1;
     /* Get estimated value of a contrast along with a test (F=t-squared) */
     estimate '3Colourvs5Colour' package 1  1 -1 -1 / divisor = 2;

proc iml;
     title2 'Table of critical values for all possible Scheffe tests';
     numdf = 3;   /* Numerator degrees of freedom for initial test */
     dendf = 15;  /* Denominator degrees of freedom for initial test */
     alpha = 0.05;
     critval = finv(1-alpha,numdf,dendf);
     zero = {0 0}; S_table = repeat(zero,numdf,1);  /* Make empty matrix */
     /* Label the columns */
     namz = {"Number of Contrasts in followup test"
             "    Scheffe Critical Value"}; mattrib S_table colname=namz;
     do i = 1 to numdf;
        s_table(|i,1|) = i;
        s_table(|i,2|) = numdf/i *  critval;
     end;
```

```
     reset noname; /* Makes output look nicer in this case */
     print "Initial test has"  numdf " and " dendf "degrees of freedom."
           "Using significance level alpha = " alpha;
     print s_table;

proc reg;
     title2 'Using proc reg and dummy variables';
     model sales = p1 p2 p3;
     ncolour: test p1+p2 = p3; /* 3 vs 5 colours */

proc glm;
     title2 "Actually it's a two-way ANOVA";
     class ncolours cartoon;
     model sales = ncolours|cartoon;

/*   The model statement could have been
     model sales = ncolours cartoon ncolours*cartoon; */
```

The `proc format` statement provides labels for the package designs. After reading the data in a routine way, `if` statements are used to construct the categorical independent variables `ncolours` and `cartoon`. Notice the extra space in the 'No ' value of the alphanumeric variable `cartoon`. At first I didn't have a space, and `Yes` was truncated to `Ye`.

Now we'll look at what the first `proc glm` does. The complete `proc glm` statement is given above. Here, we will look at it a piece at a time, examining the output as we go. First, we have

```
proc glm;
     class package;
     model sales = package;
```

The class statement declares package to be categorical. Without it, `proc glm` would do a regression with package as a quantitative independent variable. The main F-test for equality of the four means is

```
                        General Linear Models Procedure

Dependent Variable: SALES   Number of Cases Sold
                                      Sum of            Mean
Source                    DF          Squares          Square  F Value    Pr > F

Model                      3     588.22105263     196.07368421    18.59    0.0001
Error                     15     158.20000000      10.54666667
Corrected Total           18     746.42105263

                   R-Square             C.V.        Root MSE         SALES Mean
                   0.788055         17.43042       3.2475632          18.631579
```

We conclude that package design (or, if the study was poorly controlled, some variable confounded with package design) caused a difference in sales. The statement `means package;` produces mean sales for each value of the variable package.

```
Level of                    ------------SALES------------
PACKAGE            N            Mean                SD

3Col No Cartoon    5      13.4000000        3.64691651
3Colour Cartoon    5      14.6000000        2.30217289
5Col No Cartoon    5      27.2000000        3.96232255
5Colour Cartoon    4      19.5000000        2.64575131
```

Such a display is essential for seeing what is going on, but it still does not tell you which means are different from which other means. But before we lose control and start doing all possible t-tests, consider the following.

6.2 The Curse of a Thousand t-tests

Significance tests are supposed to help screen out random garbage, and help us ignore "trends" that could easily be due to chance. But all the common significance tests are designed in isolation, as if each one were the only test

you would ever be doing. The chance of getting significant results when nothing is going on may be about 0.05 (more or less, depending on how well the assumptions are met), but if you do a *lot* of tests on a data set that is purely noise (no true relationships between any independent variable and any dependent variable), the chances of false significance mount up. It's like looking for your birthday in tables of stock market prices. If you look long enough, you will find it.

This problem definitely applies when you have a significant difference among more than two treatment means, and you want to know which ones are different from each other. For example, in an experiment with 10 treatment conditions (this is not an unusually large number, for real experiments), there are 45 pairwise differences among means.

You have to pity the poor scientist who learns about this and is honest enough to take this problem seriously (let's use the term "scientist" generously to apply to anyone trying to use significance test to learn something about a data set). On one hand, good scientific practice and common sense dictate that if you have gone to the trouble to collect data, you should explore thoroughly and try to learn something from the data. But at the same time, it appears that some stern statistical entity is scolding you, and saying that you're naughty if you peek.

There are two main ways to resolve the dilemma. One is to basically ignore the problem, while perhaps acknowledging that it is there. According to this point of view, well, you're crazy if you don't explore the data. Maybe the true significance level for the entire process is greater than 0.05, but still the use of significance tests is a useful way to decide which results might be real. Nothing's perfect; let's carry on.

The other reaction is to look for ways that significance tests can be modified to allow for the fact that we're doing a lot of them. What we want are methods for holding the chances of false significance to a single low level for a *set* of tests, simultaneously. The general term for such methods is **multiple comparison** procedures. Often, when a significance test (like a one-way ANOVA) tests several things simultaneously and turns out to be significant, multiple comparison procedures are used as a second step, to investigate where the effect came from. In cases like this, the multiple comparisons are called **follow-up** tests, or **post hoc** tests, or sometimes **probing**.

It is generally acknowledged that multiple comparison methods are often helpful (even necessary) for following up significant F-tests in order to see where an effect comes from. There is less agreement on how far the principle

should be extended. Personally, I like the idea of limiting the chance of false significance to 0.05 for an entire study – say, for all the tests reported in a scientific paper, and all the ones that were not reported, too. This is a fairly radical view, shared by almost no one. But it can work in practice if you have enough data. More on this later. For now, let's concentrate on following up a significant F test in a one-way analysis of variance.

In the Kenton package design data, there are 4 treatment conditions, and 6 potential pairwise comparisons. The next line in the SAS program,

```
means package / bon tukey scheffe;
```

requests three kinds of multiple comparison tests for all pairwise differences among means.

6.2.1 Bonferroni

The Bonferroni method is very general, and extends far beyond pairwise comparisons of means. It is a simple correction that can be applied when you are performing multiple tests, and you want to hold the chances of false significance to a single low level for all the tests simultaneously. *It applies when you are testing multiple sets of independent variables, multiple dependent variables, or both.*

The Bonferroni correction consists of simply dividing the desired significance level (that's α, the maximum probability of getting significant results when actually nothing is happening, usually $\alpha = 0.05$) by the number of tests. In a way, you're splitting the alpha equally among the tests you do.

For example, if you want to perform 5 tests at joint significance level 0.05, just do everything as usual, but only declare the results significant at the *joint* 0.05 level if one of the tests gives you $p < 0.01$ (0.01=0.05/5). If you want to perform 20 tests at joint significance level 0.05, do the individual tests and calculate individual p-values as usual, but only believe the results of tests that give $p < 0.0025$ (0.0025=0.05/20). Say something like "Protecting the 20 tests at joint significance level 0.05 by means of a Bonferroni correction, the difference in reported liking between worms and spinach soufflé was the only significant food category effect."

The Bonferroni correction is conservative. That is, if you perform 20 tests, the probability of getting significance at least once just by chance is less than or equal to 0.0025 – almost always less. The big advantages of the

Bonferroni approach are simplicity and flexibility. It is the only way I know to analyze quantitative and categorical dependent variables simultaneously.

The main disadvantages of the Bonferroni approach are

1. *You have to know how many tests you want to perform in advance, and you have to know what they are.* In a typical data analysis situation, not all the significance tests are planned in advance. The results of one test will give rise to ideas for other tests. If you do this and then apply a Bonferroni correction to all the tests that you happened to do, it no longer protects all the tests simultaneously. On the other hand, you could randomly split your data into an exploratory sample and a replication sample. Test to your heart's content on the first sample. Then, when you think you know what your results are, perform only those tests on the replication sample, and protect them simultaneously with a Bonferroni correction. This could be called "Bonferroni-protected cross-validation." It sounds good, eh?

2. *The Bonferroni correction can be too conservative,* especially when the number of tests becomes large. For example, to simultaneously test all 780 correlations in a 40 by 40 correlation matrix at joint $\alpha = 0.05$, you'd only believe correlations with $p < 0.0000641 = 0.05/780$.

   Is this "too" conservative? Well, with $n = 200$ in that 40 by 40 example, you'd need $r = 0.27$ for significance (compared to $r = .14$ with no correction). With $n = 100$ you'd need $r = .385$, or about 14.8% of one variable explained by another *single* variable. Is this too much to ask? You decide.

6.2.2 Tukey

This is Tukey's Honestly Significant Difference (HSD) method. It is not his Least Significant Different (LSD) method, which has a better name but does not really get the job done. Tukey tests apply only to pairwise differences among means in ANOVA. It is based on a deep study of the probability distribution of the difference between the largest sample mean and the smallest sample mean, assuming the population means are in fact all equal.

- If you are interested in all pairwise differences among means and nothing else, and if the sample sizes are equal, Tukey is the best (most powerful) test, period.

- If the sample sizes are unequal, the Tukey tests still get the job of simultaneous protection done, but they are a bit conservative. When sample sizes are unequal, Bonferroni or Scheff can sometimes be more powerful.

6.2.3 Scheffé

Suppose there are p treatments (groups, values of the categorical independent variable, whatever you want to call them). A **contrast** is a special kind of linear combination of means in which the weights add up to zero. A population contrast has the form

$$\ell = a_1\mu_1 + a_2\mu_2 + \cdots + a_p\mu_p$$

where $a_1 + a_2 + \cdots + a_p = 0$. The case where all of the a values are zero is uninteresting, and is excluded. A population contrast is estimated by a sample contrast:

$$L = a_1\overline{Y}_1 + a_2\overline{Y}_2 + \cdots + a_p\overline{Y}_p.$$

By setting $a_1 = 1$, $a_2 = -1$, and the rest of the a values to zero we get $L = \overline{Y}_1 - \overline{Y}_2$, so it's easy to see that any pairwise difference is a contrast. Also, the average of one set of means minus the average of another set is a contrast.

The initial F test for equality of p means can be viewed as a simultaneous test of $p - 1$ contrasts. For example, suppose there are four treatments, and the null hypothesis of the initial test is $H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4$. The table gives the a_1, a_2, a_3, a_4 values for three contrasts; if all three contrasts equal zero then the four population means are equal, and *vice versa*.

a_1	a_2	a_3	a_4
1	-1	0	0
0	1	-1	0
0	0	1	-1

The way you read this table is

$$\begin{array}{rcrcrcl} \mu_1 & - & \mu_2 & & & = & 0 \\ & & \mu_2 & - & \mu_3 & = & 0 \\ & & & & \mu_3 - \mu_4 & = & 0 \end{array}$$

Clearly, if $\mu_1 = \mu_2$ and $\mu_2 = \mu_3$ and $\mu_3 = \mu_4$, then $\mu_1 = \mu_2 = \mu_3 = \mu_4$, and if $\mu_1 = \mu_2 = \mu_3 = \mu_4$, then $\mu_1 = \mu_2$ and $\mu_2 = \mu_3$ and $\mu_3 = \mu_4$. The simultaneous F test for the three contrasts is 100% equivalent to a one-way ANOVA; it yields the same F statistic, the same degrees of freedom, and the same p-value.

There is always more than one way to set up the contrasts to test a given hypothesis. Staying with the example of testing differences among four means, we could have specified

a_1	a_2	a_3	a_4
1	0	0	-1
0	1	0	-1
0	0	1	-1

so that all the means are equal to the last one. These contrasts (differences between means) are actually *equal* to the regression coefficients in a multiple regression with indicator dummy variables, in which the last category is the reference category. But no matter how you set up collection of contrasts, if you do it correctly you always get the same answer.

The Scheffé tests allow testing whether *any* contrast (or set of contrasts) of treatment means differs significantly from zero, with the tests for all possible contrasts simultaneously protected at the same significance level, usually 0.05.

When asked for Scheffé follow-ups to a one-way ANOVA, SAS tests all pairwise differences between means, but *there are infinitely many more contrasts in the same family that it does not do* — and they are all jointly protected against false significance at the 0.05 level.

It's a miracle. You can do infinitely many tests, all simultaneously protected. You do not have to know what they are in advance. It's an license for unlimited data fishing, at least within the class of contrasts of treatment means. And you can test up to $p - 1$ contrasts simultaneously if you wish. They are all part of the same family.

Two more miracles:

- If the initial one-way ANOVA is not significant, it's *impossible* for any of the Scheffé follow-ups to be significant. This is not quite true of Bonferroni or Tukey.

- If the initial one-way ANOVA *is* significant, there *must* be a single contrast that is significantly different from zero. It may not be a pairwise

difference, you may not think of it, and if you do find one it may not be easy to interpret, but there is at least one out there. Well, actually, there are infinitely many, but they may all be extremely similar to one another. Incidentally, if you test any *collection* of contrasts that includes a contrast that is significantly different from zero by a Scheffé test, then the Scheffé test for the collection will be significant too.

Given all this, clearly it is helpful to be able to test any set of contrast you wish. As you will see below, the `contrast` statement of `proc glm` lets you do it easily. For now, let's assume that you have done an initial F test for differences among p treatment means, it's statistically significant, and also you can get F tests for any contrast of collection of contrasts you specify.

As usual, the F tests for contrasts (which are sometimes optimistically called "planned comparisons") are designed in a vacuum, as if each one were the only test you would ever do on your data. But you can convert them into Scheffé follow-ups to the initial test by using a different critical value (Recall that if a test statistic is greater than the critical value, it's significant).

Suppose that the follow-up test you want to do involves s contrasts; for a test of a single difference between means or some other single contrast, $s = 1$. Compute the usual F statistic for testing the contrast, and compare it to a modified critical value that we will call F_{S-crit}; the S is for Scheffé. The formula for F_{S-crit} is

$$F_{S-crit} = \frac{p-1}{s} F_{crit}, \tag{6.1}$$

where F_{crit} is the critical value for the *initial* test — the one you are following up. You reject the null hypothesis and declare your Scheffé test significant if $F > F_{S-crit}$.

You can do as many of these tests as you want easily, using SAS and a small table of F_{S-crit} critical values. You can make the table you need with `proc iml`. This is illustrated in the example below; the code can easily be modified to suit any problem. Or, you can use a textbook table of the F distribution and a calculator.

Please take another look at Formula (6.1). Notice that multiplying by the number of means (minus one) is a kind of penalty for the richness of the infinite family of tests you could do, while dividing by the number of contrasts you're testing reduces the penalty because you're looking for something bigger. As soon as Mr. Scheffé discovered these tests, people started

complaining that the penalty was very severe, and it was too hard to get significance. In my opinion, what's remarkable is not that a license for unlimited fishing is expensive, but that it's for sale at all. You can pay for it by increasing the sample size.

When sample sizes are unequal, SAS presents follow-up tests for pairwise differences between means in the form of confidence intervals. If the 95% confidence interval does not include zero, the test (Bonferroni, Tukey or Scheffé) is significant at 0.05. Since all three types of follow-up test point to exactly the same conclusions for these data, only the Scheffé will be reproduced here.

```
                General Linear Models Procedure

                  Scheffe's test for variable: SALES

      NOTE: This test controls the type I experimentwise error rate but
            generally has a higher type II error rate than Tukey's for all
            pairwise comparisons.

         Alpha= 0.05  Confidence= 0.95  df= 15  MSE= 10.54667
                     Critical Value of F= 3.28738

   Comparisons significant at the 0.05 level are indicated by '***'.

                                  Simultaneous            Simultaneous
                                      Lower    Difference     Upper
          PACKAGE                  Confidence    Between   Confidence
         Comparison                   Limit       Means       Limit

5Col No Cartoon - 5Colour Cartoon        7.700       0.859   14.541  ***
5Col No Cartoon - 3Colour Cartoon       12.600       6.150   19.050  ***
5Col No Cartoon - 3Col No Cartoon       13.800       7.350   20.250  ***
5Colour Cartoon - 5Col No Cartoon       -7.700     -14.541   -0.859  ***
5Colour Cartoon - 3Colour Cartoon         4.900      -1.941   11.741
5Colour Cartoon - 3Col No Cartoon         6.100      -0.741   12.941
3Colour Cartoon - 5Col No Cartoon       -12.600     -19.050   -6.150  ***
3Colour Cartoon - 5Colour Cartoon        -4.900     -11.741    1.941
3Colour Cartoon - 3Col No Cartoon         1.200      -5.250    7.650
3Col No Cartoon - 5Col No Cartoon       -13.800     -20.250   -7.350  ***
3Col No Cartoon - 5Colour Cartoon        -6.100     -12.941    0.741
3Col No Cartoon - 3Colour Cartoon        -1.200      -7.650    5.250
```

Notice that the critical value for the initial test (F_{crit}, not F_{S-crit}) for performing more tests is conveniently provided.

This pairwise confidence interval format is not so easy to look at, even if the significant differences are indicated by "***." For one thing, each comparison is given twice, once in each direction. For another, the actual means are not printed, just the differences between means. It helps to re-arrange

the means from lowest to highest. This next display is not part of the SAS output; it's SAS output edited with a word processor.

Level of PACKAGE	N	SALES Mean	SALES SD
5Col No Cartoon	5	27.2000000	3.96232255
5Colour Cartoon	4	19.5000000	2.64575131
3Colour Cartoon	5	14.6000000	2.30217289
3Col No Cartoon	5	13.4000000	3.64691651

Now we see that the 5-colour No Cartoon treatment is significantly different from each of the others, which are not significantly different from each other. That's the kind of package design they should use; from a marketing standpoint, we're done. But let's look at some more follow-up tests anyway.

Testing Contrasts The `proc glm` in `kenton.sas` continues

```
/* Test some custom contrasts */
contrast '3Colourvs5Colour' package 1  1 -1 -1;
contrast 'Cartoon'          package 1 -1  1 -1;
contrast 'CartoonDepends'   package 1 -1 -1  1;
/* Test a collection of contrasts */
contrast 'Overall F'        package 1 -1  0  0,
                            package 0  1 -1  0,
                            package 0  0  1 -1;
/* Test effects of Colour and Cartoons simultaneously, allowing for
   a possible interaction */
contrast 'ColorCartoon'     package 1  1 -1 -1,
                            package 1 -1  1 -1;
```

The syntax for specifying a contrast goes: The word `contrast`, a label for the test in single or double quotes (this will appear in the output), the name of the independent variable, the coefficients of the contrast (the a values), and a semicolon to end the statement. If you are testing more than one contrast simultaneously, put a comma after the first one, repeat the independent variable name, and give another set of coefficients. The last contrast ends with a semi-colon instead of a comma. As the example shows, you can do as many tests as you like.

6.2.4 Proper Follow-ups

We will describe a set of tests as *proper follow-ups* to to an initial test if

1. The null hypothesis of the initial test logically implies the null hypotheses of all the tests in the follow-up set.

2. All the tests are jointly protected against Type I error (false significance) at a known significance level, usually $\alpha = 0.05$.

The first property requires explanation. First, consider that the Tukey tests, which are limited to pairwise differences between means, automatically satisfy this, because if all the population means are equal, then each pair is equal to each other. But it's possible to make mistakes with Bonferroni and Scheffé if you're not careful.

Here's why the first property is important. Suppose the null hypothesis of a follow-up test *does* follow logically from the null hypothesis of the initial test. Then, if the null hypothesis of the follow-up is false (there's really something going on), then the null hypothesis of the initial test must be incorrect too, and this is one way in which the initial null hypothesis is false. Thus if we correctly reject the follow-up null hypothesis, we have uncovered one of the ways in which the initial null hypothesis is false. In other words, we have (partly, perhaps) identified where the initial effect comes from.

On the other hand, if the null hypothesis of a potential follow-up test is *not* implied by the null hypothesis of the initial test, then the truth or untruth of the follow-up null hypothesis does not tell us *anything* about the null hypothesis of the initial test. They are in different domains. For example, suppose we conclude $2\mu_1$ is different from $3\mu_2$. Great, but if we want to know how the statement $\mu_1 = \mu_2 = \mu_3$ might be wrong, it's irrelevant.

If you stick to testing contrasts as a follow-up to a one-way ANOVA, you're fine. This is because if a set of population means are all equal, then any contrast of those means is equal to zero. That is, the null hypothesis of the initial test automatically implies the null hypotheses of any potential follow-up test, and everything is okay. Furthermore, if you try to specify a linear combination that is not a contrast with the `contrast` statement of `proc glm`, SAS will just say something like `NOTE: CONTRAST SOandSO is not estimable` in the log file. There is no other error message or warning; the test just does not appear in your list file.

If you really want a linear combination that is not a contrast, use the `estimate` statement. It will give the sample value of any linear combination

of treatment means, along with a t-test for whether the linear combination is significantly different from zero. Here's output from the `estimate` statement in `kenton.sas`:

Parameter	Estimate	Standard Error	t Value	Pr > |t|
3Colourvs5Colour	-9.35000000	1.49705266	-6.25	<.0001

Note $t^2 = F$ immediately below.

6.2.5 Converting Tests for Contrasts into Scheffé tests

Here is the output from the `contrast` statements.

Contrast	DF	Contrast SS	Mean Square	F Value	Pr > F
3Colourvs5Colour	1	411.4000000	411.4000000	39.01	<.0001
Cartoon	1	49.7058824	49.7058824	4.71	0.0464
CartoonDepends	1	93.1882353	93.1882353	8.84	0.0095
Overall F	3	588.2210526	196.0736842	18.59	<.0001
ColorCartoon	2	479.5888889	239.7944444	22.74	<.0001

By ordinary one-at-a-time F tests, all the tests are significant. But let's treat them as Scheffé tests. To do this, we need the F_{S-crit} critical values for $s = 1$, 2 and 3. Actually we don't need one for $s = 3$, because by (6.1), it's the same as the critical value of the initial test. And in fact, any test of $p - 1$ non-redundant contrasts is equivalent to the initial one-way ANOVA, always.

It's easy to get the F_{S-crit} values from `proc iml`. The following code is written carefully so that you can use it for any problem by just modifying the vales of `numdf` and `dendf` (and maybe `alpha` if you don't want to use 0.05).

```
proc iml;
     title2 'Table of critical values for all possible Scheffe tests';
     numdf = 3;   /* Numerator degrees of freedom for initial test (p-1) */
     dendf = 15;  /* Denominator degrees of freedom for initial test (n-p) */
     alpha = 0.05;
     critval = finv(1-alpha,numdf,dendf);
     zero = {0 0}; S_table = repeat(zero,numdf,1);  /* Make empty matrix */
     /* Label the columns */
     namz = {"Number of Contrasts in followup test"
             "    Scheffe Critical Value"};
     mattrib S_table colname=namz;
     do i = 1 to numdf;
        s_table(|i,1|) = i;
        s_table(|i,2|) = numdf/i *  critval;
     end;
     reset noname; /* Makes output look nicer in this case */
     print "Initial test has"  numdf " and " dendf "degrees of freedom."
           "Using significance level alpha = " alpha;
     print s_table;
```

Here is the output.

```
                  Kenton Oneway Example From Neter et al.                  8
          Table of critical values for all possible Scheffe tests
                                              13:35 Friday, March 16, 2007

     Initial test has         3  and         15 degrees of freedom.
               Using significance level alpha =       0.05


     Number of Contrasts in followup test     Scheffe Critical Value

                                        1                  9.8621463
                                        2                  4.9310732
                                        3                  3.2873821
```

For the one-degree-of-freedom tests (single contrasts) we need $F > 9.86$ for significance. This means `3Colourvs5Colour` is significant, but `Cartoon`

and `CartoonDepends` are not, even though `CartoonDepends` has a p-value of 0.0095 by the one-at-a-time test. `ColourCartoon` is also significant, because $22.74 > 4.93$. And of course `Overall F` is significant; it's the initial test.

6.2.6 Extensions

This section provides a brief but very powerful extension of the Scheffé tests to multiple regression, and Scheffé-like tests for logistic regression.

Multiple Regression

Suppose the initial hypothesis is that d regression coefficients all are equal to zero. We will follow up the initial test by testing whether s linear combinations of these regression coefficients are different from zero; $s \leq d$. Notice that now we are testing *linear combinations*, not just contrasts. If a set of coefficients are all zero, then any linear combination (weighted sum) of the coefficients is also zero. Thus the null hypotheses of the follow-up tests are implied by the null hypotheses of the initial test. As in the case of Scheffé tests for contrasts in one-way ANOVA, using an adjusted critical value guarantees simultaneous protection for all the follow-up tests at the same significance level as the initial test. This means we have proper follow-ups.

The formula for the modified critical value is

$$F_{S-crit} = \frac{d}{s} F_{crit}, \tag{6.2}$$

where again, the null hypothesis of the initial test is that d regression coefficients are all zero, and the null hypothesis of the follow-up test is that s linear combinations of those coefficients are equal to zero.

For convenience, here is the `proc iml` code to produce a table of adjusted critical values.

```
proc iml;
     title2 'Scheffe tests for Regression: Critical values';
     numdf = 3;   /* Numerator degrees of freedom for initial test (d) */
     dendf = 15;  /* Denominator degrees of freedom for initial test (n-d-1) */
     alpha = 0.05;
     critval = finv(1-alpha,numdf,dendf);
     zero = {0 0}; S_table = repeat(zero,numdf,1);  /* Make empty matrix */
     /* Label the columns */
     namz = {"Number of linear combos in followup test"
             "    Scheffe Critical Value"};
     mattrib S_table colname=namz;
     do i = 1 to numdf;
        s_table(|i,1|) = i;
        s_table(|i,2|) = numdf/i *  critval;
     end;
     reset noname; /* Makes output look nicer in this case */
     print "Initial test has "  numdf " and " dendf "degrees of freedom."
           "Using significance level alpha = " alpha;
     print s_table;
```

The Scheffé tests for contrasts in a one-way ANOVA are special cases of this, because anything you can do with factorial analysis of variance, you can do with dummy variable regression. It's very convenient with `test` statements in `proc reg`.

Logistic Regression

For logistic regression, there are Scheffé-like followups called *union-intersection tests*. The true Scheffé tests are a special kind of union-intersection method that applies to the (multivariate) normal linear model. Scheffé tests have one property that is not true of union-intersection follow-ups in general: the guaranteed existence of a significant one-degree-of-freedom test. This is tied to geometric properties of the multivariate normal distribution.

Just as in normal regression, the suppose that the initial null hypothesis is that d coefficients in the logistic regression model are all equal to zero. Suppose the initial hypothesis is that d regression coefficients all are equal to zero. We will follow up by testing whether s linear combinations of these regression coefficients are different from zero; $s \leq d$. There is no adjustment.

The critical value for the follow-up tests is exactly that of the initial test: a chi-square with d degrees of freedom. This principle applies to both likelihood ratio and Wald tests. In fact, it is true of likelihood ratio and Wald tests in general, not just in logistic regression.

Bibliographic citation

If you want to report the use of union-intersection tests or Scheffé tests for regression, or even Scheffé tests for more than one contrast in a one-way design, you will have difficulty finding it in any published Statistics text. Like Scheffé's original 1953 article [13], they almost universally stick to single contrasts. And it's usually not too helpful to cite unpublished material like this document.

Hochberg and Tamhane's (1987) monograph *Multiple comparison procedures* [7] is a good source for the tests of multiple linear combinations in regression, of which the tests of contrasts presented here are a special case. It's not very readable to non-statisticians, though. The same can be said of Gabriel's (1969) article [6], which is the primary source for the union-intersection follow-ups. But you can just trust me and cite them anyway.

Bibliography

[1] Bickel, P. J., Hammel, E. A., and O'Connell, J. W. (1975). Sex bias in graduate admissions: Data from Berkeley. *Science*, **187**, 398-403.

[2] Cody, R. P. and Smith, J. K. (1991). *Applied statistics and the SAS programming language.* (4th Edition) Upper Saddle River, New Jersey: Prentice-Hall.

[3] Cook, T. D. and Campbell, D. T. (1979). *Quasi-experimentation: design and analysis issues for field settings.* New York: Rand McNally.

[4] Feinberg, S. (1977) *The analysis of cross-classified categorical data.* Cambridge, Massachusetts: MIT Press.

[5] Fisher, R. A. (1925) *Statistical methods for research workers.* London: Oliver and Boyd.

[6] Gabriel, K. R. (1969). "Simultaneous test procedures — some theory of multiple comparisons." *Ann. Math. Statist.*, 40, 224–250.

[7] Hochberg, Y., and Tamhane, A. C. (1987). *Multiple comparison procedures.* New York: Wiley.

[8] Moore, D. S. and McCabe, G. P. (1993). *Introduction to the practice of statistics.* New York: W. H. Freeman.

[9] Neter, J., Kutner, M. H., Nachhtscheim, C. J. and Wasserman, W. (1996) *Applied linear statistical models.* (4th Edition) Toronto: Irwin.

[10] Roethlisberger, F. J. (1941). *Management and morale*. Cambridge, Mass.: Harvard University Press.

[11] Rosenthal, R. (1966). *Experimenter effects in behavioral research*. New York: Appleton-Century-Croft.

[12] Rosenthal, R. and Jacobson, L. (1968). *Pygmalion in the classroom: teacher expectation and pupils' intellectual development*. New York: Holt, Rinehart and Winston.

[13] Scheffé, H. (1953). "A method for judging all contrasts in the analysis of variance." *Biometrika*, 40, 87–104.

Chapter Seven: Factorial ANOVA with Multiple Regression

The Kenton Example with dummy variables

First consider indicator dummy variable coding with an intercept. Here is the part of the data step that defines the dummy variables. Because we have an intercept, we'll represent the four categories of package design with three dummy variables. The table below shows the model for population mean sales.

Design	p1	p2	p3	$E[Y] = \beta_0 + \beta_1 p1 + \beta_2 p2 + \beta_3 p3$
3Colour Cartoon	1	0	0	$\beta_0 + \beta_1 = \mu_1$
3Col No Cartoon	0	1	0	$\beta_0 + \beta_2 = \mu_2$
5Colour Cartoon	0	0	1	$\beta_0 + \beta_3 = \mu_3$
5Col No Cartoon	0	0	0	$\beta_0 = \mu_4$

To clarify the parallel between population parameters and sample statistics, the corresponding table of estimated sales figures is

Design	p1	p2	p3	$\hat{Y} = b_0 + b_1 p1 + b_2 p2 + b_3 p3$
3Colour Cartoon	1	0	0	$b_0 + b_1 = \overline{Y}_1$
3Col No Cartoon	0	1	0	$b_0 + b_2 = \overline{Y}_2$
5Colour Cartoon	0	0	1	$b_0 + b_3 = \overline{Y}_3$
5Col No Cartoon	0	0	0	$b_0 = \overline{Y}_4$

One thing these tables show is something that is true of *any* valid dummy variable coding scheme (when there are only categorical independent variables): $\hat{Y} = \overline{Y}$ for each category or combination of categories.

It is also easy to see that to test for differences among means, we want to simultaneously test whether β_1, β_2 and β_3 are different from zero -- or equivalently, whether b_1, b_2 and b_3 are *significantly* different from zero. That is,

we want to simultaneously test the dummy variables p1, p2 and p3. The overall F test of proc reg does the job.

```
proc reg;
     model sales = p1 p2 p3;
```

Yielding:

```
Model: MODEL1
Dependent Variable: SALES      Number of Cases Sold

                          Analysis of Variance

                             Sum of          Mean
       Source          DF    Squares        Square      F Value       Prob>F

       Model            3   588.22105    196.07368       18.591       0.0001
       Error           15   158.20000     10.54667
       C Total         18   746.42105
```

We got this same same F value for differences among the four means from `proc glm`. The next line does the 3 versus 5 color comparison.

```
     ncolour: test p1+p2 = p3; /* 3 vs 5 colours */
```

It works because we want to test whether $\frac{1}{2}(\mu_1 + \mu_2) = \frac{1}{2}(\mu_3 + \mu_4)$, and

$$\frac{1}{2}(\beta_0 + \beta_1 + \beta_0 + \beta_2) = \frac{1}{2}(\beta_0 + \beta_3 + \beta_0)$$

is algebraically equivalent to

$$\beta_1 + \beta_2 = \beta_3.$$

The estimate statement from `proc glm` yielded t = -6.25. Calculate $F = t^2 = 39.0625$, and compare the output of the `test` statement above:

```
Test: NCOLOUR  Numerator:    411.4000  DF:     1   F value:  39.0076
               Denominator:  10.54667  DF:    15   Prob>F:    0.0001
```

The difference is rounding error. It's the same test. But we'd rather avoid having to do algebra whenever we want to test a contrast. In **cell means coding**, we use an indicator dummy variable for each category (four, in this case), and omit the intercept. The tables that follow indicate why it's called cell mean coding.

Cell Means Coding for Package Design

Design	p1	p2	p3	p4	$E[Y] = \beta_1 p1 + \beta_2 p2 + \beta_3 p3 + \beta_4 p4$
3Colour Cartoon	1	0	0	0	$\beta_1 = \mu_1$
3Col No Cartoon	0	1	0	0	$\beta_2 = \mu_2$
5Colour Cartoon	0	0	1	0	$\beta_3 = \mu_3$
5Col No Cartoon	0	0	0	1	$\beta_4 = \mu_4$

Design	p1	p2	p3	p4	$\hat{Y} = b_1 p1 + b_2 p2 + b_3 p3 + b_4 p4$
3Colour Cartoon	1	0	0	0	$b_1 = \bar{Y}_1$
3Col No Cartoon	0	1	0	0	$b_2 = \bar{Y}_2$
5Colour Cartoon	0	0	1	0	$b_3 = \bar{Y}_3$
5Col No Cartoon	0	0	0	1	$b_4 = \bar{Y}_4$

Here is the `proc reg.`

```
proc reg;
     model sales = p1 p2 p3 p4 / noint;
     alleq:    test p1=p2=p3=p4;
     numcol:   test p1+p2 = p3+p4;
     cartoon:  test p1+p3 = p2+p4;
     inter1:   test p1-p2 = p3-p4; /* Effect of cartoon depends on ncolours */
     inter2:   test p1-p3 = p2-p4; /* Effect of ncolours depends on cartoon */
     Y3_N3:    test p1=p2;  /* All pairwise tests */
```

```
Y3_Y5:    test p1=p3;
Y3_N5:    test p1=p4;
N3_Y5:    test p2=p3;
N3_N5:    test p2=p4;
Y5_N5:    test p3=p4;
```

And the output. First, the overall F test, which is very different from what we had before.

```
Model: MODEL1
NOTE: No intercept in model. R-square is redefined.
Dependent Variable: SALES      Number of Cases Sold

                          Analysis of Variance

                             Sum of          Mean
Source          DF          Squares        Square      F Value       Prob>F

Model            4       7183.80000   1795.95000      170.286       0.0001
Error           15        158.20000     10.54667
U Total         19       7342.00000

    Root MSE       3.24756     R-square       0.9785
    Dep Mean      18.63158     Adj R-sq       0.9727
    C.V.          17.43042
```

With no intercept,

- Total sum of squares is now $\sum_{i=1}^{n} Y_i^2$. It's no longer corrected for the mean; U means uncorrected. R^2 is radically affected.

- The overall F-test is for whether ALL the betas are zero - usually uninteresting

Notice now the parameter estimates are exactly the cell means.

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	T for H0: Parameter=0	Prob > |T|
P1	1	14.600000	1.45235441	10.053	0.0001
P2	1	13.400000	1.45235441	9.226	0.0001
P3	1	19.500000	1.62378159	12.009	0.0001
P4	1	27.200000	1.45235441	18.728	0.0001

Now the custom tests. I will repeat the test statement for each one, and provide some discussion.

The Statement

```
alleq:    test p1=p2=p3=p4;
```

yields this output:

```
Dependent Variable: SALES
Test: ALLEQ    Numerator:    196.0737  DF:    3   F value:  18.5911
               Denominator:  10.54667  DF:   15   Prob>F:    0.0001
```

This really is the overall test for whether all four means are equal -- again. The F value is the same as we got earlier at least two times. But look at the test statement. As usual, it specifies restrictions on the betas that give us the reduced model. But this time, those restrictions are not of the simple form we saw before, setting a subset of the betas equal to zero. Now we're setting them all to be equal. This shows you two things:

- The `test` statement in `proc reg` is a little more general than it seemed at first. It lets you test simultaneously whether several linear combinations of betas equal zero. Here, we're testing three linear combinations: $\beta_1-\beta_2=0$, $\beta_2-\beta_3=0$, $\beta_3-\beta_4=0$. The test statement could have read: `alleq: test p1-p2=0, p2-p3=0, p3-p4=p4;`

- The full versus reduced model business is also more general than you might think. In ordinary regression, "all" we can do is test collections linear restrictions on the parameters. But

in the most general hypothesis testing framework, all one *ever* does is to compare the fit of a full model to the fit of a reduced model in which some restriction has been placed on the values of the parameters. Those restrictions are called the "null hypothesis." You didn't really need to know this.

To really understand the next several test statements, we need to recognize that the 4-category variable Package Design actually represents the combination of two independent variables: Number of Colours and Presence versus absence of cartoons. That is, we have a two-factor design. Consider the following table:

Population Cell Means and Marginal Means for the Kenton Example

	Cartoon	No Cartoon	
3 Colours	μ_1	μ_2	$\frac{\mu_1 + \mu_2}{2}$
5 Colours	μ_3	μ_4	$\frac{\mu_3 + \mu_4}{2}$
	$\frac{\mu_1 + \mu_3}{2}$	$\frac{\mu_2 + \mu_4}{2}$	

In addition to population mean sales for each package design (denoted by μ_1 through μ_4), the table above shows **marginal means** -- quantities like $\frac{\mu_2 + \mu_4}{2}$, which are obtained by averaging over rows or columns.

If there are differences among marginal means for a categorical independent variable in a two-way (or higher) layout like this, we say there is a **main effect** for that variable. Tests for main effects are of great interest; they can indicate whether, averaging over the values of the other categorical independent variables in the design, whether the independent variable in question is related to the dependent variable. Note that averaging over the values of other independent variables is not the same thing as controlling for them, but it can still be a valuable thing to do.

The population means in the preceding table are estimated by corresponding sample quantities. The numbers in the table below come from the `means` output of the first `proc glm`.

Sample Cell and Marginal Means for the Kenton Example

	Cartoon	No Cartoon	
3 Colours	14.6	13.4	14
5 Colours	19.5	27.2	23.35
	17.05	20.3	

(14.6+13.4)/2 = 14, and so on.

The next custom test is for the main effect of number of colours (3 vs. 5). It tests whether $\frac{\mu_1 + \mu_2}{2} = \frac{\mu_3 + \mu_4}{2}$. It's the same thing as asking whether the marginal mean for 2 Colours (14) is *significantly* different from the marginal mean for 5 colours (23.35).

The test command, obtained directly by multiplying both sides =f $\frac{\mu_1 + \mu_2}{2} = \frac{\mu_3 + \mu_4}{2}$ by 2 (this has no effect on the test), is

```
numcol:   test p1+p2 = p3+p4;
```

yielding this output:

```
Dependent Variable: SALES
Test: NUMCOL   Numerator:    411.4000  DF:    1   F value:  39.0076
               Denominator:  10.54667  DF:   15   Prob>F:    0.0001
```

So the answer is Yes. There is a significant main effect for number of colours, with 5-colour packages generating more sales when you average across Cartoon and No-cartoon designs. And notice how much more convenient the cell means coding makes this test. Recall

```
ncolour: test p1+p2 = p3; /* 3 vs 5 colours */
```

from Page 13.

Similarly, the main effect for presence versus absence of cartoons on the package is tested by asking whether $\frac{\mu_1 + \mu_3}{2} = \frac{\mu_2 + \mu_4}{2}$.

```
cartoon:  test p1+p3 = p2+p4;
```

```
Dependent Variable: SALES
Test: CARTOON  Numerator:     49.7059  DF:    1   F value:   4.7129
               Denominator:  10.54667  DF:   15   Prob>F:    0.0464
```

So the main effect for Cartoon is barely significant, with Non-cartoon designs doing better. With a Scheffee test, though, it's not significant. Fsch = `4.7129/3 = 1.57, which is less than the critical value of 3.29.`

The two-way design we have been looking at is called a factorial design. In a factorial design, there are two or more categorical independent variables (called factors, in this context) typically with data with for combinations of the factors being collected. Factorial designs are often found in experimental studies, but not always.

When Sir Ronald Fisher (in whose honour the F-test is named) dreamed up factorial designs, he pointed out that they enable the scientist to investigate the effects of several independent variables at much less expense than if a separate experiment had to be conducted to test each one. In addition, they allow one to ask systematically whether the effect of one independent variable *depends* on the value of another independent variable. If the effect of one independent variable depends on another, we will say there is an **interaction** between those variables. We talk about an A "by" B or A x B interaction. An interaction means "it depends."

Let's look at the table of population means again.

	Cartoon	No Cartoon	
3 Colours	μ_1	μ_2	$\frac{\mu_1 + \mu_2}{2}$
5 Colours	μ_3	μ_4	$\frac{\mu_3 + \mu_4}{2}$
	$\frac{\mu_1 + \mu_3}{2}$	$\frac{\mu_2 + \mu_4}{2}$	

The effect of Cartoons when the package has three colours is represented by μ_1-μ_2. The effect of Cartoons when the package has five colours is represented by μ_3-μ_4. Therefore, the interaction of

Cartoon by number of colours is a *difference between differences*, and we want to test whether $\mu_1-\mu_2=\mu_3-\mu_4$. That's what we're doing below:

```
     inter1:   test p1-p2 = p3-p4; /* Effect of cartoon depends on ncolours */

Dependent Variable: SALES
Test: INTER1   Numerator:     93.1882  DF:    1   F value:   8.8358
               Denominator:  10.54667  DF:   15   Prob>F:    0.0095
```

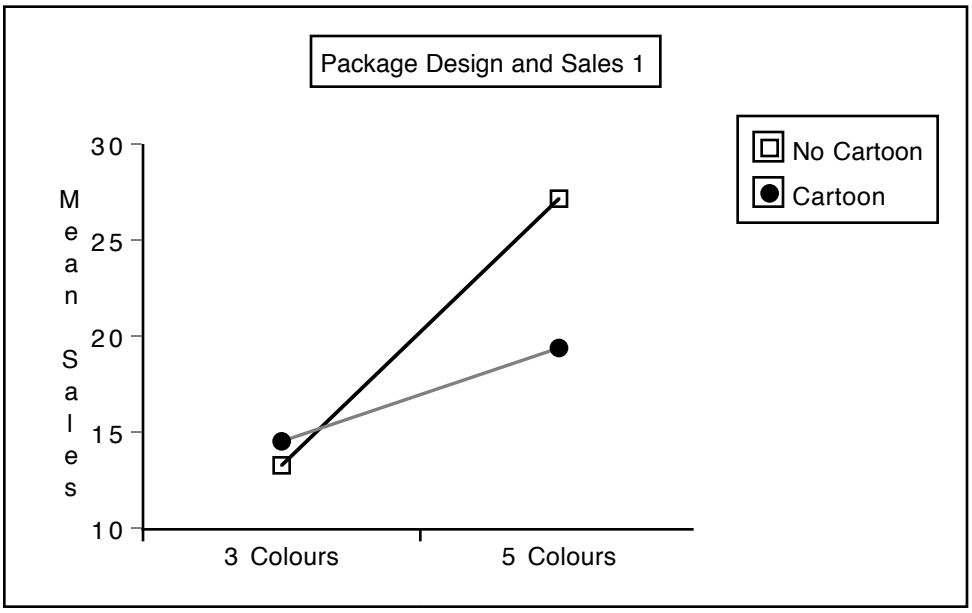
Another way to think about the interaction is to ask whether the effect of number of colours depends on presence versus absence of cartoon pictures. We are asking whether $\mu_1-\mu_3=\mu_2-\mu_4$. Here's the `test` statement and the output.

```
     inter2:   test p1-p3 = p2-p4; /* Effect of ncolours depends on cartoon */

Dependent Variable: SALES
Test: INTER2   Numerator:     93.1882  DF:    1   F value:   8.8358
               Denominator:  10.54667  DF:   15   Prob>F:    0.0095
```

Notice that this F test is identical to the last one? It happens because $\mu_1-\mu_2=\mu_3-\mu_4$ is algebraically equivalent to $\mu_1-\mu_3=\mu_2-\mu_4$. So the two ways of talking about the interaction are the same thing, mathematically. Fortunately, this *always* happens, no matter how big the design. If you express an interaction correctly as a collection of differences between differences, it is algebraically equivalent to all other correct ways of expressing the interaction. Choose the one that is easiest to think about.

Incidentally, p = `0.0095 seems impressive, but the test is not significant if it is considered as a Scheffe follow-up:  Fsch = 8.8358/3 = 2.945267 < 3.29.`

If an interaction is significant, you should graph it to figure out what it means. Here is one example:

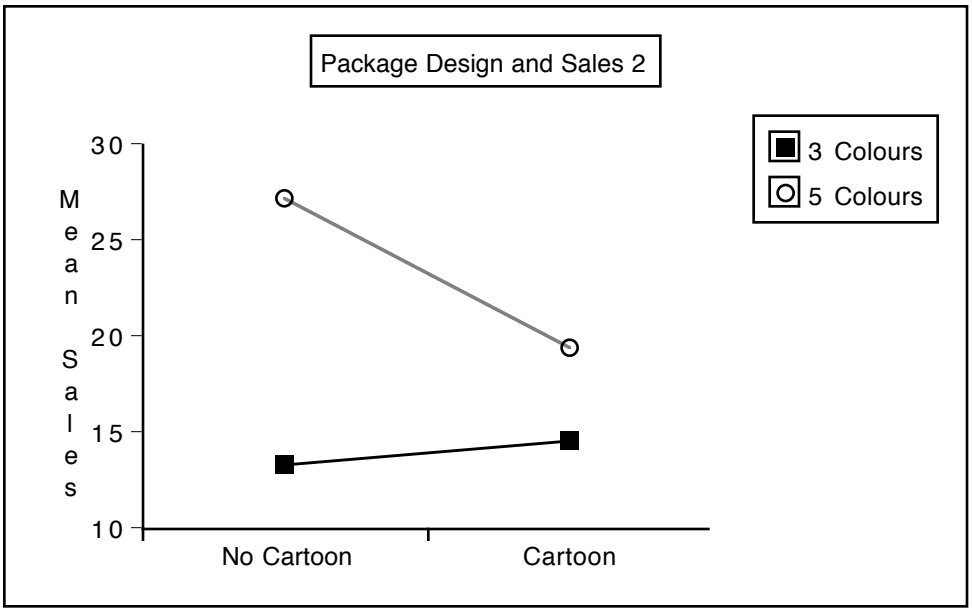


Whenever you have an interaction, such graphs will display non-parallel lines. Well actually, when you plot an interaction with real data, the lines will always be at least a little non-parallel. The question is whether they depart *significantly* from being parallel. Here, the advantage of 5 colours over 3 is significantly greater for designs without cartoons (unless you are a member of the Scheffé cult, as I am), and we can see it in the graph.

The post-hoc tests tell us that there is a significantly more sales with 5-colour designs, for both the cartoon and non-cartoon conditions. The interaction tells us that this effect is significantly greater when there are no cartoons.

Remember the significant main effect for cartoon? It was just barely significant: $p = 0.0464$. The graph above shows quite clearly that this effect is entirely due to the advantage of no-cartoon designs in the 5-colour condition. So here, we have a main effect that's significant, but we really should not interpret it, because of the interaction.

Some texts claim that if you have an interaction, you should *never* interpret the main effects. But look at the next figure, which graphs the same interaction in the other direction (there are only two ways to do it, because it is a two-factor interaction).



The picture that emerges here is that 5-colour designs are better overall, and the advantage is greater in the No-cartoon condition. Here, we can see that it makes sense to interpret both the main effect for number of colours *and* the interaction. This example shows why I disagree with the advice to never interpret main effects when there is an interaction.

The last six tests are the pairwise differences between means. Their value is that we can convert them easily to post-hoc Bonferroni or Scheffé tests. Personally, I like the idea of letting the tests for main effects, interactions and all pairwise differences as follow-ups to the initial oneway ANOVA. I prefer Scheffé, because I don't need to know in advance how many tests I'm going to do. I also love the Scheffé tests because of their 100% consistency with the initial tests. If the initial test is non-significant, no Scheffé follow-up can be significant, as a mathematical certainty. And if the initial test is significant, then there *must* be a significant Scheffé follow-up.

```
Dependent Variable: SALES
Test: Y3_N3     Numerator:      3.6000  DF:     1   F value:    0.3413
                Denominator:  10.54667  DF:    15   Prob>F:     0.5677


Dependent Variable: SALES
Test: Y3_Y5     Numerator:     53.3556  DF:     1   F value:    5.0590
                Denominator:  10.54667  DF:    15   Prob>F:     0.0399
```

```
Dependent Variable: SALES
Test: Y3_N5    Numerator:    396.9000  DF:    1   F value:  37.6327
               Denominator:  10.54667  DF:   15   Prob>F:    0.0001


Dependent Variable: SALES
Test: N3_Y5    Numerator:     82.6889  DF:    1   F value:   7.8403
               Denominator:  10.54667  DF:   15   Prob>F:    0.0135


Dependent Variable: SALES
Test: N3_N5    Numerator:    476.1000  DF:    1   F value:  45.1422
               Denominator:  10.54667  DF:   15   Prob>F:    0.0001


Dependent Variable: SALES
Test: Y5_N5    Numerator:    131.7556  DF:    1   F value:  12.4926
               Denominator:  10.54667  DF:   15   Prob>F:    0.0030
```

Sample Question: What p-value is required for significance if these 9 tests are to be protected with a Bonferroni correction at the 0.05 level? **Answer**: 0.05/9 = 0.0056

Effect	F	p	F_{sch} =F/3*	Significant with Bonferroni?	Significant with Scheffé?
Main Effect for Ncolours	39.0076	0.0001	13.0025	Yes	Yes
Main effect for Cartoon	4.7129	0.0464	1.57097	No	No
Interaction	8.8358	0.0095	2.9453	No	No
Cartoon3 vs NoCartoon3	0.3413	0.5677	0.1138	No	No
Cartoon3 vs Cartoon5	5.0590	0.0399	1.6863	No	No
Cartoon3 vs NoCartoon5	37.6327	0.0001	12.5442	Yes	Yes
NoCartoon3 vs Cartoon5	7.8403	0.0135	2.6134	No	No
NoCart3 vs Nocart5	45.1422	0.0001	15.0474	Yes	Yes
Cartoon5 vs NoCartoon5	12.4926	0.0030	4.1642	Yes	Yes

* Compare with critical value of F= 3.28738

The main thing to note here is that when you treat the test for interaction as a follow-up test instead of a one-at-a-time test, it's no longer significant. You are left with a simpler story. Five-colour designs work better than three-colour designs, and designs without cartoons work better in the 5-colour condition.

In general, if you go the multiple comparison route, it's going to make you more conservative. You will draw fewer conclusions. On the other hand, in terms of this particular example, the implications for *action* (marketing action) are the same whether or not you use multiple comparisons. The Kenton company should use a 5-colour design without cartoons.

We've seen how to do the tests above with dummy variables and `proc reg`. If you are only interested in testing single contrasts, the `estimate` command of `proc glm` is a bit more convenient, because proc glm sets up the dummy variables for you. All you have to do is give the coefficients of the contrast you want.

```
/* Single contrasts are just as convenient with the ESTIMATE
   statement of proc glm. Illustrate all pairwise.
   Note F = t-squared  */

proc glm;
     class package;
     model sales=package;
     estimate 'Y3_N3' package 1 -1  0  0;
     estimate 'Y3_Y5' package 1  0 -1  0;
     estimate 'Y3_N5' package 1  0  0 -1;
     estimate 'N3_Y5' package 0  1 -1  0;
     estimate 'N3_N5' package 0  1  0 -1;
     estimate 'Y5_N5' package 0  0  1 -1;
```

It's nice to have this degree of control, but not always necessary. In factorial analysis of variance, we commonly wish to test all main effects and interactions. Proc glm will compose the contrasts for you, as well as setting up the dummy variables:

```
proc glm;
     class ncolours cartoon;
     model sales = ncolours|cartoon;
/*   The model statement could have been
     model sales = ncolours cartoon ncolours*cartoon; */
```

In `proc glm`, if you separate a collection of classification variables with vertical bars, it means include all the main effects and interactions among the variables.

Here is the output:

```
                    General Linear Models Procedure
```

```
Dependent Variable: SALES   Number of Cases Sold
                                      Sum of            Mean
Source                     DF         Squares          Square  F Value    Pr > F

Model                       3     588.22105263    196.07368421    18.59    0.0001

Error                      15     158.20000000     10.54666667

Corrected Total            18     746.42105263

                   R-Square             C.V.        Root MSE         SALES Mean

                   0.788055         17.43042       3.2475632          18.631579


Source                     DF        Type I SS     Mean Square  F Value    Pr > F

NCOLOURS                    1     452.86549708    452.86549708    42.94    0.0001
CARTOON                     1      42.16732026     42.16732026     4.00    0.0640
NCOLOURS*CARTOON            1      93.18823529     93.18823529     8.84    0.0095

Source                     DF      Type III SS     Mean Square  F Value    Pr > F

NCOLOURS                    1     411.40000000    411.40000000    39.01    0.0001
CARTOON                     1      49.70588235     49.70588235     4.71    0.0464
NCOLOURS*CARTOON            1      93.18823529     93.18823529     8.84    0.0095
```

The output starts with an overall test that is 100% identical to the initial oneway ANOVA. It has the same R^2, the same F, the same p-value --- everything. This always happens. No matter how many independent variables you have or how many values each one has, simultaneously testing all the main effects and interactions is the same as defining a new independent variable whose values are the *combinations* of the variable values from the factorial ANOVA --- and then doing a one-way analysis of variance using that variable.

By default, SAS `proc glm` produces two sets of tests for the main effects and interaction(s). In the tests based on Type I Sums of Squares, each effect is controlled only for those before it in the table. In Type III Sums of Squares, each effect is controlled for all the others. That's why the last test is always identical for these two methods. When sample sizes are all equal or proportional, the independent variables are completely unrelated, and tests based on Type I and Type III sums of squares are all the same -- not just the last one.

The F and p values we get from Type III sums of squares match what we've done using `proc reg`. Most of the time, the tests from the Type III sums of squares are what we want.

Beyond the two-by-two Case

Methods for factorial ANOVA and testing interactions can easily be extended in several ways.

- More independent variables
- More than two values for an independent variable
- Interactions between continuous independent variables
- Interactions between categorical independent variables and continuous independent variables.

Extension to more than two factors is straightforward. Suppose we had grocery stores of three different sizes (small, medium and large), and within each size, the four package designs were randomly allocated to stores. We would have three factors -- store size, number of colours, and presence versus absence of cartoons.

- For each independent variable, averaging over the other two variables would give marginal means -- the basis for estimating and testing for main effects.
- Averaging over each of the independent variables in turn, we would have a two-way marginal table of means for the other two variables, and the pattern of means in that table could show a two-way interaction.

The full three-dimensional table of means would provide a basis for looking at a three-way, or three-factor interaction. The interpretation of a three-way interaction is that the nature of the two-way interaction depends on the value of the third variable. This principle extends to any number of factors, so we would interpret a six-way interaction to mean that the nature of the 5-way interaction depends on the value of the sixth variable.

- Fortunately, the order in which one considers the variables does not matter. For example, we can say that the A by B interaction depends on the value of C, or that the A by C interaction depends on B, or that the B by C interaction depends on the value of A. The translations of these statements into algebra are all equivalent to one another, always. This principle extends to any number of factors.

- As you might imagine, as the number of factors becomes large, *interpreting* higher-way interactions -- that is, figuring out what they mean -- becomes more and more difficult. For this reason, sometimes the higher-order interactions are deliberately omitted from the full model in big experimental designs; they are never tested. Is this reasonable? Most of my answers are just elaborate ways to say I don't know.

More than two values for an independent variable

Regardless of how many factors we have, or how many levels there are in each factor, we could always form a combination variable -- that is, a single categorical independent variable whose values represent all the combinations of independent variable values in the factorial design. We have seen that in a two-by-two design, the tests for both main effects and the interaction resolve themselves into tests for single contrasts -- contrasts of the means of the combination variable. When independent variables have more than two values, the same thing is true, except that tests for main effects and interactions appear as test for *collections* of contrasts on the combination variable.

It is useful to pursue this principle in detail, for three reasons.

- First, it thinking of an interaction as a collection of contrasts can really help you understand what an interaction is.
- Second, once you have seen the tests for main effects and interactions as collections of contrasts, you can easily compose a test for any collection of contrasts that is of interest.
- Third, seeing main effects and interactions in terms of contrasts makes it easy to see how they can be modified to become Bonferroni or Scheffe follow-ups to initial significant one-way ANOVA on the combination variable --- if you choose to follow this conservative data analytic strategy.

We'll start with an example.

The seeds of the canola plant yield a high-quality cooking oil. Canola is one of Canada's biggest cash crops. But each year, millions of dollars are lost because of a fungus that kills canola plants. Or is it just one fungus? All this stuff looks the same. It's a nasty black rot that grows fastest under moist, warm conditions. It looks quite a bit like the fungus that grows in between shower tiles.

A team of botanists recognized that although the fungus may look the same, there are actually several different kinds that are genetically distinct. There are also quite a few strains of canola plant, so the questions arose

- Are some strains of fungus more aggressive than others? That is, do they grow faster and overwhelm the plant's defenses faster?
- Are some strains of canola plant more vulnerable to infection than others?
- Are some strains of fungus more dangerous to certain strains of plant and less dangerous to others?

These questions can be answered directly by looking at main effects and the interaction, so a factorial experiment was designed in which canola plants of three different varieties were randomly selected to be infected with one of six genetically different types of fungus. The way they did it was to scrape a little patch at the base of the plant, and wrap the wound with a moist band-aid that had some fungus on it. Then the plant was placed in a very moist dark environment for three days. After three days the bandage was removed and the plant was put in a commercial greenhouse. On each of 14 consecutive days, various measurements were made on the plant. Here, we will be concerned with lesion length, the length of the fungus patch on the plant, measured in millimeters.

The dependent variable will be mean lesion length; the mean is over the 14 daily lesion length measurements for each plant. The independent variables are Cultivar (type of canola plant) and MCG (type of fungus). Type of plant is called cultivar because the fungus grows (is "cultivated") on the plant. MCG stands for "Mycelial Compatibility Group." This strange name comes from the way that the botanists decided whether two types of fungus were genetically distinct. The would grow two samples on the same dish in a nutrient solution, and if the two fungus patches stayed separate, they were genetically different. If they grew together into a single patch of fungus (that is, they were compatible), then they were genetically identical. Apparently, this phenomenon is well established.

Here is the SAS program `appgreen1.sas`. As usual, the entire program is listed first. Then pieces of the program are repeated, together with pieces of output and discussion.

```
/* appgreen1.sas */
%include 'gh91read.sas';
options pagesize=100;
proc freq;
     tables plant*mcg /norow nocol nopercent;
proc glm;
     class plant mcg;
     model meanlng = plant|mcg;
     means plant|mcg;
proc tabulate;
     class mcg plant;
     var meanlng ;
     table (mcg all),(plant all) * (mean*meanlng);

/* Replicate tests for main effects and interactions, using contrasts on a
   combination variable. This is the hard way to do it, but if you can do
   this, you understand interactions and you can test any collection of
   contrasts. The definition of the variable combo could have been in
   gh91read.sas   */

data slime;
     set mould; /* mould was created by ghread91.sas */
     if         plant=1 and mcg=1 then combo =  1;
        else if plant=1 and mcg=2 then combo =  2;
        else if plant=1 and mcg=3 then combo =  3;
        else if plant=1 and mcg=7 then combo =  4;
        else if plant=1 and mcg=8 then combo =  5;
        else if plant=1 and mcg=9 then combo =  6;
        else if plant=2 and mcg=1 then combo =  7;
        else if plant=2 and mcg=2 then combo =  8;
        else if plant=2 and mcg=3 then combo =  9;
        else if plant=2 and mcg=7 then combo = 10;
        else if plant=2 and mcg=8 then combo = 11;
        else if plant=2 and mcg=9 then combo = 12;
        else if plant=3 and mcg=1 then combo = 13;
        else if plant=3 and mcg=2 then combo = 14;
        else if plant=3 and mcg=3 then combo = 15;
        else if plant=3 and mcg=7 then combo = 16;
        else if plant=3 and mcg=8 then combo = 17;
        else if plant=3 and mcg=9 then combo = 18;
     label combo = 'Plant-MCG Combo';
```

```
/* Getting main effects and the interaction with CONTRAST statements */
proc glm;
     class combo;
     model meanlng = combo;
     contrast 'Plant Main Effect'
         combo 1  1  1  1  1  1  -1 -1 -1 -1 -1 -1   0  0  0  0  0  0,
         combo 0  0  0  0  0  0   1  1  1  1  1  1  -1 -1 -1 -1 -1 -1;
     contrast 'MCG Main Effect'
         combo 1 -1  0  0  0  0   1 -1  0  0  0  0   1 -1  0  0  0  0,
         combo 0  1 -1  0  0  0   0  1 -1  0  0  0   0  1 -1  0  0  0,
         combo 0  0  1 -1  0  0   0  0  1 -1  0  0   0  0  1 -1  0  0,
         combo 0  0  0  1 -1  0   0  0  0  1 -1  0   0  0  0  1 -1  0,
         combo 0  0  0  0  1 -1   0  0  0  0  1 -1   0  0  0  0  1 -1;
     contrast 'Plant by MCG Interaction'
         combo -1  1  0  0  0  0   1 -1  0  0  0  0   0  0  0  0  0  0,
         combo  0  0  0  0  0  0  -1  1  0  0  0  0   1 -1  0  0  0  0,
         combo  0 -1  1  0  0  0   0  1 -1  0  0  0   0  0  0  0  0  0,
         combo  0  0  0  0  0  0   0 -1  1  0  0  0   0  1 -1  0  0  0,
         combo  0  0 -1  1  0  0   0  0  1 -1  0  0   0  0  0  0  0  0,
         combo  0  0  0  0  0  0   0  0 -1  1  0  0   0  0  1 -1  0  0,
         combo  0  0  0 -1  1  0   0  0  0  1 -1  0   0  0  0  0  0  0,
         combo  0  0  0  0  0  0   0  0  0 -1  1  0   0  0  0  1 -1  0,
         combo  0  0  0  0 -1  1   0  0  0  0  1 -1   0  0  0  0  0  0,
         combo  0  0  0  0  0  0   0  0  0  0 -1  1   0  0  0  0  1 -1;

/* proc reg's test statement may be easier, but first we need to
   make 16 dummy variables for cell means coding. This will illustrate
   arrays and loops, too */

data yucky;
     set slime;
     array mu{18} mu1-mu18;
     do i=1 to 18;
          if combo=. then mu{i}=.;
          else if combo=i then mu{i}=1;
          else mu{i}=0;
     end;

proc reg;
     model meanlng = mu1-mu18 / noint;
     alleq:   test mu1=mu2=mu3=mu4=mu5=mu6=mu7=mu8=mu9=mu10=mu11=mu12
                   = mu13=mu14=mu15=mu16=mu17=mu18;

     plant:   test mu1+mu2+mu3+mu4+mu5+mu6    = mu7+mu8+mu9+mu10+mu11+mu12,
                   mu7+mu8+mu9+mu10+mu11+mu12 = mu13+mu14+mu15+mu16+mu17+mu18;

     fungus:  test mu1+mu7+mu13 = mu2+mu8+mu14 = mu3+mu9+mu15
                   = mu4+mu10+mu16 = mu5+mu11+mu17 = mu6+mu12+mu18;

     p_by_f:  test mu2-mu1=mu8-mu7=mu14-mu13,
                   mu3-mu2=mu9-mu8=mu15-mu14,
```

```
                    mu4-mu3=mu10-mu9=mu16-mu15,
                    mu5-mu4=mu11-mu10=mu17-mu16,
                    mu6-mu5=mu12-mu11=mu18-mu17;

/* Now illustrate effect coding, with the interaction represented by a
   collection of product terms.  */

data nasty;
     set yucky;
     /* Two dummy variables for plant */
        if plant=. then p1=.;
        else if plant=1 then p1=1;
        else if plant=3 then p1=-1;
        else p1=0;
     if plant=. then p2=.;
        else if plant=2 then p2=1;
        else if plant=3 then p2=-1;
        else p2=0;
     /* Five dummy variables for mcg */
     if mcg=. then f1=.;
        else if mcg=1 then f1=1;
        else if mcg=9 then f1=-1;
        else f1=0;
     if mcg=. then f2=.;
        else if mcg=2 then f2=1;
        else if mcg=9 then f2=-1;
        else f2=0;
     if mcg=. then f3=.;
        else if mcg=3 then f3=1;
        else if mcg=9 then f3=-1;
        else f3=0;
     if mcg=. then f4=.;
        else if mcg=7 then f4=1;
        else if mcg=9 then f4=-1;
        else f4=0;
     if mcg=. then f5=.;
        else if mcg=8 then f5=1;
        else if mcg=9 then f5=-1;
        else f5=0;
     /* Product terms for interactions */
        p1f1 = p1*f1; p1f2=p1*f2 ; p1f3=p1*f3 ; p1f4=p1*f4; p1f5=p1*f5;
        p2f1 = p2*f1; p2f2=p2*f2 ; p2f3=p2*f3 ; p2f4=p2*f4; p2f5=p2*f5;

proc reg;
     model meanlng = p1 -- p2f5;
     plant:  test p1=p2=0;
     mcg:    test f1=f2=f3=f4=f5=0;
     p_by_f: test p1f1=p1f2=p1f3=p1f4=p1f5=p2f1=p2f2=p2f3=p2f4=p2f5 = 0;
```

The SAS program starts with a `%include` statement that reads `ghread91.sas`. The file `ghread91.sas` consists of a single big data step. We'll skip it, because all we really need are the two independent variables `plant` and `mcg`, and the dependent variable `meanlng`.

Just to see what we've got, we do a `proc freq` to show the sample sizes.

```
proc freq;
     tables plant*mcg /norow nocol nopercent;
```

and we get

TABLE OF PLANT BY MCG

PLANT(Type of Plant) MCG(Mycelial Compatibility Group)

Frequency	1	2	3	7	8	9	Total
GP159	6	6	6	6	6	6	36
HANNA	6	6	6	6	6	6	36
WESTAR	6	6	6	6	6	6	36
Total	18	18	18	18	18	18	108

So it's a nice 3 by 6 factorial design, with 6 plants in each treatment combination. The `proc glm` for analyzing this is straightforward. Again, we get all main effects and interactions for the factor names separated by vertical bars.

```
proc glm;
     class plant mcg;
     model meanlng = plant|mcg;
     means plant|mcg;
```

And the output is

General Linear Models Procedure
Class Level Information

Class	Levels	Values
PLANT	3	GP159 HANNA WESTAR
MCG	6	1 2 3 7 8 9

Number of observations in data set = 108

General Linear Models Procedure

Dependent Variable: MEANLNG Average Lesion length

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	17	328016.87350	19295.11021	19.83	0.0001
Error	90	87585.62589	973.17362		
Corrected Total	107	415602.49939			

R-Square	C.V.	Root MSE	MEANLNG Mean
0.789256	48.31044	31.195731	64.573479

Source	DF	Type I SS	Mean Square	F Value	Pr > F
PLANT	2	221695.12747	110847.56373	113.90	0.0001
MCG	5	58740.26456	11748.05291	12.07	0.0001
PLANT*MCG	10	47581.48147	4758.14815	4.89	0.0001

Source	DF	Type III SS	Mean Square	F Value	Pr > F
PLANT	2	221695.12747	110847.56373	113.90	0.0001
MCG	5	58740.26456	11748.05291	12.07	0.0001
PLANT*MCG	10	47581.48147	4758.14815	4.89	0.0001

Notice that the Type I and Type III tests are the same. This always happens when the sample sizes are equal.

```
                1991 Greenhouse Study                              4
                                        10:42 Tuesday, February 19, 2002

          General Linear Models Procedure

      Level of          -----------MEANLNG-----------
      PLANT         N         Mean                SD

      GP159        36       14.055159        12.1640757
      HANNA        36       55.700198        30.0137912
      WESTAR       36      123.965079        67.0180440

      Level of          -----------MEANLNG-----------
      MCG           N         Mean                SD

      1            18      41.4500000        33.6183462
      2            18      92.1333333        78.3509451
      3            18      87.5857143        61.7086751
      7            18      81.7603175        82.6711755
      8            18      50.8579365        39.3417859
      9            18      33.6535714        39.1480830

Level of    Level of         -----------MEANLNG-----------
PLANT       MCG          N         Mean                SD

GP159       1            6        12.863095        12.8830306
GP159       2            6        21.623810        17.3001296
GP159       3            6        14.460714         7.2165396
GP159       7            6        17.686905        16.4258441
GP159       8            6         8.911905         7.3162618
GP159       9            6         8.784524         6.5970501
HANNA       1            6        45.578571        26.1430472
HANNA       2            6        67.296429        30.2424997
HANNA       3            6        94.192857        20.2877876
HANNA       7            6        53.621429        24.8563497
HANNA       8            6        47.838095        12.6419109
HANNA       9            6        25.673810        17.1723150
WESTAR      1            6        65.908333        35.6968616
WESTAR      2            6       187.479762        45.1992178
WESTAR      3            6       154.103571        26.5469183
WESTAR      7            6       173.972619        79.1793105
WESTAR      8            6        95.823810        22.3712022
WESTAR      9            6        66.502381        52.5253101
```

The main effects are fairly easy to look at, and we definitely can construct a plot from the 18 cell means (or copy them into a nicer-looking table. But the following `proc tabulate` prints a table that is much easier to look at.

```
proc tabulate;
     class mcg plant;
     var meanlng ;
     table (mcg all),(plant all) * (mean*meanlng);
```

The syntax of `proc tabulate` is fairly elaborate, and at times it's worth the effort. Any reader who has seen the type of stub-and-banner tables favoured by professional market researchers will be impressed to hear that `proc tabulate` can come close to that. I figured out how to make the table below by looking in the manual. I then promptly forgot the overall principles, because it's not a tool I use a lot -- and the syntax is rather arcane. However, this example is easy to follow if you want to produce good-looking two-way tables of means. Here's the output.

	Type of Plant			
	GP159	HANNA	WESTAR	ALL
	MEAN	MEAN	MEAN	MEAN
	Average Lesion length	Average Lesion length	Average Lesion length	Average Lesion length
Mycelial Compatibility Group				
1	12.86	45.58	65.91	41.45
2	21.62	67.30	187.48	92.13
3	14.46	94.19	154.10	87.59
7	17.69	53.62	173.97	81.76
8	8.91	47.84	95.82	50.86
9	8.78	25.67	66.50	33.65
ALL	14.06	55.70	123.97	64.57

The proc tabulate output makes it easy to graph the means. But before we do so, let's look at the main effects and interactions as collections of contrasts. This will actually make it easier to figure out what the results mean, once we see what they are.

We have a three by six factorial design that looks like this. Population means are shown in the cells. The single-subscript notation encourages us to think of the combination of MCG and cultivar as a single categorical independent variable with 18 categories.

	MCG (Type of Fungus)					
Cultivar (Type of Plant)	1	2	3	7	8	9
GP159	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6
Hanna	μ_7	μ_8	μ_9	μ_{10}	μ_{11}	μ_{12}
Westar	μ_{13}	μ_{14}	μ_{15}	μ_{16}	μ_{17}	μ_{18}

Next is the part of the SAS program that creates the combination variable. Notice that it involves a data step that comes *after* the `proc glm`. This usually doesn't happen. I did it by creating a new data set called `slime` that starts by being identical to `mould`, which was created in the file `gh91read.sas`. The `set` command is used to read in the data set `mould`, and then we start from there. This is done just for teaching purposes. Ordinarily, I would not create multiple data sets that are mostly copies of each other. I'd put the whole thing in one data step. Here's the code.

```
data slime;
     set mould; /* mould was created by ghread91.sas */
     if         plant=1 and mcg=1 then combo =  1;
        else if plant=1 and mcg=2 then combo =  2;
        else if plant=1 and mcg=3 then combo =  3;
        else if plant=1 and mcg=7 then combo =  4;
        else if plant=1 and mcg=8 then combo =  5;
        else if plant=1 and mcg=9 then combo =  6;
        else if plant=2 and mcg=1 then combo =  7;
        else if plant=2 and mcg=2 then combo =  8;
        else if plant=2 and mcg=3 then combo =  9;
        else if plant=2 and mcg=7 then combo = 10;
        else if plant=2 and mcg=8 then combo = 11;
        else if plant=2 and mcg=9 then combo = 12;
        else if plant=3 and mcg=1 then combo = 13;
        else if plant=3 and mcg=2 then combo = 14;
        else if plant=3 and mcg=3 then combo = 15;
        else if plant=3 and mcg=7 then combo = 16;
        else if plant=3 and mcg=8 then combo = 17;
        else if plant=3 and mcg=9 then combo = 18;
     label combo = 'Plant-MCG Combo';
```

	MCG (Type of Fungus)					
Cultivar (Type of Plant)	1	2	3	7	8	9
GP159	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6
Hanna	μ_7	μ_8	μ_9	μ_{10}	μ_{11}	μ_{12}
Westar	μ_{13}	μ_{14}	μ_{15}	μ_{16}	μ_{17}	μ_{18}

It is clear that the absence of a main effect for Cultivar is the same as

$$\mu_1+\mu_2+\mu_3+\mu_4+\mu_5+\mu_6 = \mu_7+\mu_8+\mu_9+\mu_{10}+\mu_{11}+\mu_{12} = \mu_{13}+\mu_{14}+\mu_{15}+\mu_{16}.$$

There are two equalities here, and they are saying that two contrasts of the eighteen cell means are equal to zero. To see why this is true, consider the first equality

$$\mu_1+\mu_2+\mu_3+\mu_4+\mu_5+\mu_6 = \mu_7+\mu_8+\mu_9+\mu_{10}+\mu_{11}+\mu_{12}$$

Subtracting the quantity on the right-hand side from both sider of the equation, we get

$$\mu_1+\mu_2+\mu_3+\mu_4+\mu_5+\mu_6 - (\mu_7+\mu_8+\mu_9+\mu_{10}+\mu_{11}+\mu_{12}) = 0,$$

and then distributing the minus sign to get rid of the parentheses yields

$$\mu_1+\mu_2+\mu_3+\mu_4+\mu_5+\mu_6-\mu_7-\mu_8-\mu_9-\mu_{10}-\mu_{11}-\mu_{12} = 0. \qquad (4.2)$$

Recall that here, a contrast is a linear combination of the form

$$L = a_1\mu_1 + a_2\mu_2 + ... + a_{18}\mu_{18}.,$$

where the *a* weights add up to zero. Expression (4.2) fits this description, with the first 6 weights equal to one, the next six weights equal to minus one (so they add to zero), and the last 6 weights equal to zero.

The table below gives the weights of the contrasts defining the test for the main effect of plant, one set of weights in each row.

a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}
1	1	1	1	1	1	-1	-1	-1	-1	-1	-1	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	1	-1	-1	-1	-1	-1	-1

This is the basis of the first contrast statement in `proc glm`. Notice how the contrasts are separated by commas. Also notice that the variable on which we're doing contrasts (`combo`) has to be repeated.

```
/* Getting main effects and the interaction with CONTRAST statements */
proc glm;
     class combo;
     model meanlng = combo;
     contrast 'Plant Main Effect'
         combo 1  1  1  1  1  1  -1 -1 -1 -1 -1 -1   0  0  0  0  0  0,
         combo 0  0  0  0  0  0   1  1  1  1  1  1  -1 -1 -1 -1 -1 -1;
```

If there is no main effect for MCG, we are saying

$$\mu_1+\mu_7+\mu_{13} = \mu_2+\mu_8+\mu_{14} = \mu_3+\mu_9+\mu_{15} = \mu_4+\mu_{10}+\mu_{16} = \mu_5+\mu_{11}+\mu_{17} = \mu_6+\mu_{12}+\mu_{18}.$$

There are 5 contrasts here, one for each equals sign; there is always an equals sign for each contrast. Here is the table showing the contrasts.

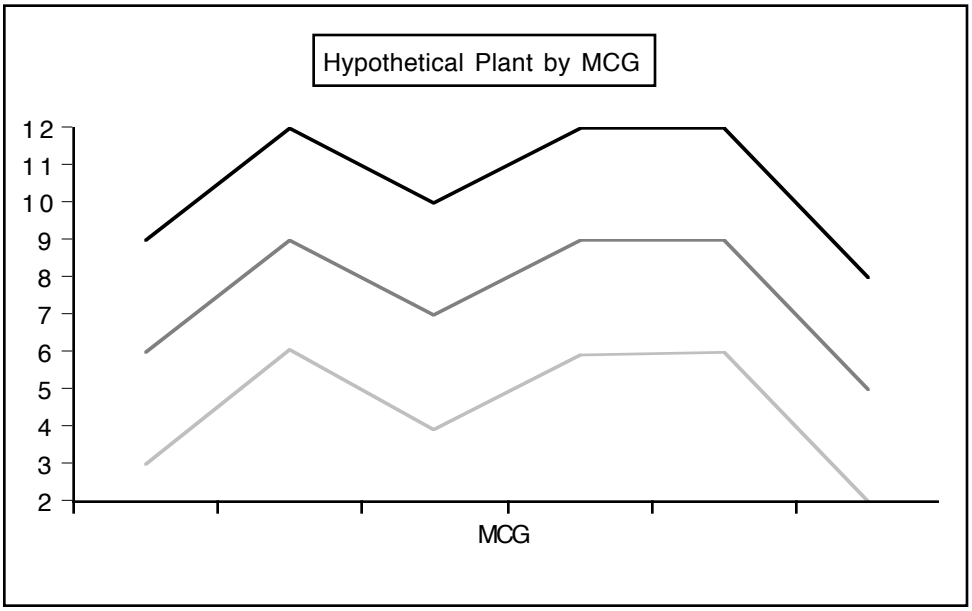
a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}
1	-1	0	0	0	0	1	-1	0	0	0	0	1	-1	0	0	0	0
0	1	-1	0	0	0	0	1	-1	0	0	0	0	1	-1	0	0	0
0	0	1	-1	0	0	0	0	1	-1	0	0	0	0	1	-1	0	0
0	0	0	1	-1	0	0	0	0	1	-1	0	0	0	0	1	-1	0
0	0	0	0	1	-1	0	0	0	0	1	-1	0	0	0	0	1	-1

And here is the corresponding `test` statement in `proc glm`.

```
contrast 'MCG Main Effect'
    combo 1 -1  0  0  0  0   1 -1  0  0  0  0   1 -1  0  0  0  0,
    combo 0  1 -1  0  0  0   0  1 -1  0  0  0   0  1 -1  0  0  0,
    combo 0  0  1 -1  0  0   0  0  1 -1  0  0   0  0  1 -1  0  0,
    combo 0  0  0  1 -1  0   0  0  0  1 -1  0   0  0  0  1 -1  0,
    combo 0  0  0  0  1 -1   0  0  0  0  1 -1   0  0  0  0  1 -1;
```

	MCG (Type of Fungus)					
Cultivar (Type of Plant)	1	2	3	7	8	9
GP159	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6
Hanna	μ_7	μ_8	μ_9	μ_{10}	μ_{11}	μ_{12}
Westar	μ_{13}	μ_{14}	μ_{15}	μ_{16}	μ_{17}	μ_{18}

To compose the Plant by MCG interaction, consider the following hypothetical graph. You can think of the "effect" of MCG as a *profile*, representing a pattern of differences among means. If the three profiles are the same shape for each type of plant -- that is, if they are parallel, the effect of MCG does not depend on the type of plant, and there is no interaction.



For the profiles to be parallel, each set of corresponding line segments must be parallel. To start with the three line segments on the left, the rise represented by $\mu_2-\mu_1$ must equal the rise $\mu_8-\mu_7$, and $\mu_8-\mu_7$ must equal $\mu_{14}-\mu_{13}$. This is two contrasts that equal zero:

$$\mu_2 - \mu_1 - \mu_8 + \mu_7 = 0 \text{ and } \mu_8-\mu_7 -\mu_{14}+\mu_{13} = 0.$$

There are two contrasts for each of the four remaining sets of three line segments, for a total of ten contrasts. They appear directly in the `contrast` statement of `proc glm`. Notice how each row adds to zero; these are *contrasts*, not just linear combinations.

```
contrast 'Plant by MCG Interaction'
      combo -1  1  0  0  0  0   1 -1  0  0  0  0   0  0  0  0  0  0,
      combo  0  0  0  0  0  0  -1  1  0  0  0  0   1 -1  0  0  0  0,
      combo  0 -1  1  0  0  0   0  1 -1  0  0  0   0  0  0  0  0  0,
      combo  0  0  0  0  0  0   0 -1  1  0  0  0   0  1 -1  0  0  0,
      combo  0  0 -1  1  0  0   0  0  1 -1  0  0   0  0  0  0  0  0,
      combo  0  0  0  0  0  0   0  0 -1  1  0  0   0  0  1 -1  0  0,
      combo  0  0  0 -1  1  0   0  0  0  1 -1  0   0  0  0  0  0  0,
      combo  0  0  0  0  0  0   0  0  0 -1  1  0   0  0  0  1 -1  0,
      combo  0  0  0  0 -1  1   0  0  0  0  1 -1   0  0  0  0  0  0,
      combo  0  0  0  0  0  0   0  0  0  0 -1  1   0  0  0  0  1 -1;
```

Now we can compare the tests we get from these contrast statements with what we got from a two-way ANOVA. For easy reference, here is part of the two-way output.

Source	DF	Type III SS	Mean Square	F Value	Pr > F
PLANT	2	221695.12747	110847.56373	113.90	0.0001
MCG	5	58740.26456	11748.05291	12.07	0.0001
PLANT*MCG	10	47581.48147	4758.14815	4.89	0.0001

And here is the output from the `contrast` statements.

Contrast	DF	Contrast SS	Mean Square	F Value	Pr > F
Plant Main Effect	2	221695.12747	110847.56373	113.90	0.0001
MCG Main Effect	5	58740.26456	11748.05291	12.07	0.0001
Plant by MCG Interac	10	47581.48147	4758.14815	4.89	0.0001

So it worked. Here are some comments.

- Of course this is *not* the way you'd want to test for main effects and interactions. On the contrary, it makes you appreciate all the work that `glm` does for you when you say `model meanlng = plant|mcg;`

- These contrasts are supposed to be an aid to understanding --- understanding what main effects and interactions really are, and understanding how you can test nearly any hypothesis you can think of in a multi-factor design. Almost without exception, what you want to do is test whether some collection of contrasts are equal to zero. Now you can do it, whether the collection you're interested in happens to be standard, or not.

- On the other hand, this was brutal. Even though I am comfortable with high school algebra, the size of the design made specifying those contrasts an unpleasant experience. There is an easier way.

An Easier Way to test Sets of Contrasts in Factorial ANOVA

Because the `test` statement of `proc reg` has a more flexible syntax than the `contrast` statement of `proc glm`, it's a lot easier if you use cell means dummy variable coding, fit a model with no intercept in `proc reg`, and use `test` statements. In the following example, the indicator dummy variables are named `mu1` to `mu18`. This choice makes it possible to directly transcribe statements about the population cell means into test statements. I highly recommend it. Of course if you really hate Greek letters, you could always name them `m1` to `m18` or something.

First, we need to define 18 dummy variables. In general, it's a bit more tedious to define dummy variables than to make a combination variable. Here, I use the combination variable `combo` (which has already been created) to make the task a bit easier -- and also to illustrate the use of arrays and loops in the data step.

```
/* proc reg's test statement may be easier, but first we need to
   make 16 dummy variables for cell means coding. This will illustrate
   arrays and loops, too */

data yucky;
     set slime;
     array mu{18} mu1-mu18;
     do i=1 to 18;
          if combo=. then mu{i}=.;
          else if combo=i then mu{i}=1;
          else mu{i}=0;
     end;

proc reg;
     model meanlng = mu1-mu18 / noint;
     alleq:   test mu1=mu2=mu3=mu4=mu5=mu6=mu7=mu8=mu9=mu10=mu11=mu12
                   = mu13=mu14=mu15=mu16=mu17=mu18;

     plant:   test mu1+mu2+mu3+mu4+mu5+mu6    = mu7+mu8+mu9+mu10+mu11+mu12,
                   mu7+mu8+mu9+mu10+mu11+mu12 = mu13+mu14+mu15+mu16+mu17+mu18;

     fungus:  test mu1+mu7+mu13 = mu2+mu8+mu14 = mu3+mu9+mu15
                   = mu4+mu10+mu16 = mu5+mu11+mu17 = mu6+mu12+mu18;

     p_by_f:  test mu2-mu1=mu8-mu7=mu14-mu13,
                   mu3-mu2=mu9-mu8=mu15-mu14,
                   mu4-mu3=mu10-mu9=mu16-mu15,
                   mu5-mu4=mu11-mu10=mu17-mu16,
                   mu6-mu5=mu12-mu11=mu18-mu17;
```

Looking again at the table of means, it's easy to see how natural the syntax is.

	MCG (Type of Fungus)					
Cultivar (Type of Plant)	1	2	3	7	8	9
GP159	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6
Hanna	μ_7	μ_8	μ_9	μ_{10}	μ_{11}	μ_{12}
Westar	μ_{13}	μ_{14}	μ_{15}	μ_{16}	μ_{17}	μ_{18}

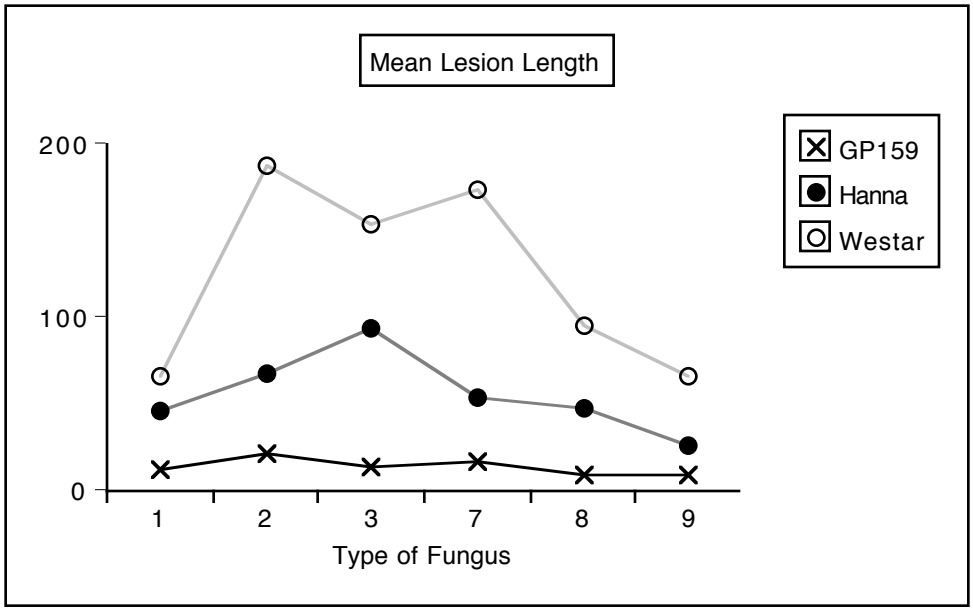
And again, the tests are correct. First, repeat the output from the `contrast` statements of `proc glm` (which matched the `proc glm` two-way ANOVA output).

```
Contrast                DF      Contrast SS     Mean Square  F Value    Pr > F

Plant Main Effect        2     221695.12747    110847.56373   113.90    0.0001
MCG Main Effect          5      58740.26456     11748.05291    12.07    0.0001
Plant by MCG Interac    10      47581.48147      4758.14815     4.89    0.0001
```

Then, compare output from the test statements of proc reg.

```
Dependent Variable: MEANLNG
Test: ALLEQ    Numerator:  19295.1102  DF:   17   F value:  19.8270
               Denominator:  973.1736  DF:   90   Prob>F:    0.0001


Dependent Variable: MEANLNG
Test: PLANT    Numerator: 110847.5637  DF:    2   F value: 113.9032
               Denominator:  973.1736  DF:   90   Prob>F:    0.0001


Dependent Variable: MEANLNG
Test: FUNGUS   Numerator:  11748.0529  DF:    5   F value:  12.0719
               Denominator:  973.1736  DF:   90   Prob>F:    0.0001


Dependent Variable: MEANLNG
Test: P_BY_F   Numerator:   4758.1481  DF:   10   F value:   4.8893
               Denominator:  973.1736  DF:   90   Prob>F:    0.0001
```

Okay, now we know how to do anything. Finally, it is time to graph the interaction, and find out what these results mean!



First, we see a sizable and clear main effect for Plant. In fact, going back to the analysis of variance summary tables and dividing the Sum of Squares explained by Plant by the Total Sum of Squares, we observe that Plant explains around 53% of the variation in mean lesion length. That's huge. We will definitely want to look at pairwise comparisons of marginal means, too; we'll get back to this later.

Looking at the pattern of means, it's clear that while the main effect of fungus type is statistically significant, this is not something that should be interpreted, because which one is best (worst) depends on the type of plant. That is, we need to look at the interaction.

The profiles really look different. In particular, GP159 not only has a smaller average lesion length, but it seems to exhibit less responsiveness to different strains of fungus. A test for the equality of μ_1 through μ_6 would be valuable. Pairwise comparisons of the 6 means for Hanna and the 6 means for Westar look promising, too.

A Brief Consideration of Multiple Comparisons

The mention of pairwise comparisons brings up the issue of formal multiple comparison follow-up tests for this problem. The way people often do follow-up tests for factorial designs is to make a combination variable and then do all pairwise comparisons. It seems like they do this because they think it's the only thing the software will let them do. Certainly it's better than nothing. Some comments:

With SAS, pairwise comparisons of cell means are *not* the only thing you can do. `Proc glm` will do all pairwise comparisons of *marginal* means quite easily. This means it's easy to follow up a significant and meaningful main effect.

For the present problem, there are 120 possible pairwise comparisons of the 16 cell means. If we do all these as one-at-a-time tests, the chances of false significance are certainly mounting. There is a strong case here for doing multiple comparisons.

Since the sample sizes are equal, Tukey tests are most powerful for all pairwise comparisons. But it's not so simple. Pairwise comparisons within plants (for example, comparing the 6 means for Westar) are interesting, and pairwise comparisons within fungus types (for example, comparison of Hanna, Westar and GP159 for fungus Type 1) are interesting, but the remaining 57 pairwise comparisons are a lot less so.

Also, pairwise comparisons of cell means are not all we want to do. We've already mentioned the need for pairwise comparisons of the marginal means for plants, and we'll soon see that other, less standard comparisons are of interest.

Everything we need to do will involve testing collections of contrasts. The approach we'll take is to do everything as a one-at-a-time custom test initially, and then figure out how we should correct for the fact that we've done a lot of tests.

It's good to be guided by the data. Here we go. The analyses will be done in the SAS program `appgreen2.sas`. As usual, the entire program is given first. But you should be aware that the program was written one piece at a time and executed many times, with later analyses being suggested by the earlier ones.

The program starts by reading in the file `gh91bread.sas`, which is just `gh91read.sas` with the additional variables defined (especially `combo` and `mu1` through `mu18`) that were defined in `appgreen1.sas`.

```
/* appgreen2.sas:  */
%include 'gh91bread.sas';
options pagesize=100;


proc glm;
     title 'Repeating initial Plant by MCG ANOVA, full design';
     class plant mcg;
     model meanlng = plant|mcg;
     means plant|mcg;

/*  A.  Pairwise comparisons of marginal means for plant, full design
    B.  Test all GP159 means equal, full design
    C.  Test profiles for Hanna & Westar parallel, full design         */

proc reg;
     model meanlng = mu1-mu18 / noint;
     A_GvsH:   test  mu1+mu2+mu3+mu4+mu5+mu6 = mu7+mu8+mu9+mu10+mu11+mu12;
     A_GvsW:   test  mu1+mu2+mu3+mu4+mu5+mu6 = mu13+mu14+mu15+mu16+mu17+mu18;
     A_HvsW:   test  mu7+mu8+mu9+mu10+mu11+mu12 = mu13+mu14+mu15+mu16+mu17+mu18;
     B_G159eq: test  mu1=mu2=mu3=mu4=mu5=mu6;
     C_HWpar:  test  mu8-mu7=mu14-mu13,  mu9-mu8=mu15-mu14,
                     mu10-mu9=mu16-mu15, mu11-mu10=mu17-mu16,
                     mu12-mu11=mu18-mu17;


/*  D.  Oneway on mcg, GP158 subset  */

data just159;  /* This data set will have just GP159 */
     set mould;
     if plant=1;

proc glm data=just159;
     title 'D.  Oneway on mcg, GP158 subset';
     class mcg;
     model meanlng = mcg;

/*  E.  Plant by MCG, Hanna-Westar subset   */

data hanstar;  /* This data set will have just Hanna and Westar */
     set mould;
     if plant ne 1;

proc glm data=hanstar;
     title 'E.  Plant by MCG, Hanna-Westar subset';
     class plant mcg;
     model meanlng = plant|mcg;
```

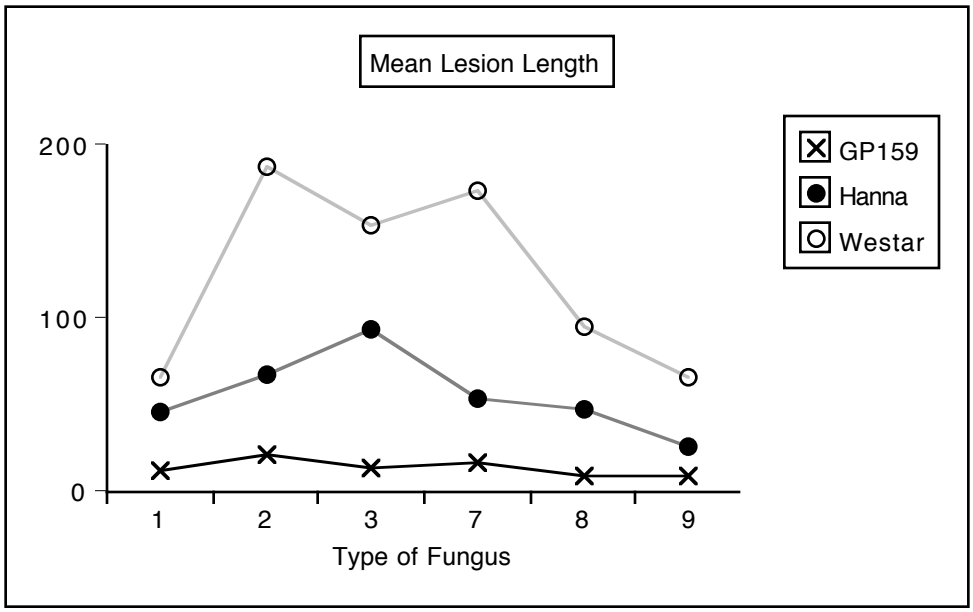
```
/*  F.  Plant by MCG followup, Hanna-Westar subset
                    Interaction:  Follow with all pairwise differences of
                    Westar minus Hanna differences
   G.   Differences within Hanna?
   H.   Differences within Westar?  */

proc reg;
     model meanlng = mu7-mu18 / noint;
     F_inter: test   mu13-mu7=mu14-mu8=mu15-mu9
                   = mu16-mu10=mu17-mu11=mu18-mu12;
     F_1vs2:   test   mu13-mu7=mu14-mu8;
     F_1vs3:   test   mu13-mu7=mu15-mu9;
     F_1vs7:   test   mu13-mu7=mu16-mu10;
     F_1vs8:   test   mu13-mu7=mu17-mu11;
     F_1vs9:   test   mu13-mu7=mu18-mu12;
     F_2vs3:   test   mu14-mu8=mu15-mu9;
     F_2vs7:   test   mu14-mu8=mu16-mu10;
     F_2vs8:   test   mu14-mu8=mu17-mu11;
     F_2vs9:   test   mu14-mu8=mu18-mu12;
     F_3vs7:   test   mu15-mu9=mu16-mu10;
     F_3vs8:   test   mu15-mu9=mu17-mu11;
     F_3vs9:   test   mu15-mu9=mu18-mu12;
     F_7vs8:   test   mu16-mu10=mu17-mu11;
     F_7vs9:   test   mu16-mu10=mu18-mu12;
     F_8vs9:   test   mu17-mu11=mu18-mu12;
     G_Hanaeq: test   mu7=mu8=mu9=mu10=mu11=mu12;
     H_Westeq: test   mu13=mu14=mu15=mu16=mu17=mu18;


proc iml; /* Critical values for Scheffe tests */
     interac = finv(.95,5,60) ; print interac;
     oneway = finv(.95,11,60); print oneway;
```

After reading and defining the data with a `%include` statement, the program repeats the initial three by six ANOVA from `appgreen1.sas`. This is just for completeness.

A. It then uses `proc reg` to fit a cell means model, and then tests for all three pairwise differences among Plant means. They are all significantly different from each other, confirming what appears visually in the interaction plot.

```
proc reg;
     model meanlng = mu1-mu18 / noint;
     A_GvsH: test  mu1+mu2+mu3+mu4+mu5+mu6 = mu7+mu8+mu9+mu10+mu11+mu12;
     A_GvsW: test  mu1+mu2+mu3+mu4+mu5+mu6 = mu13+mu14+mu15+mu16+mu17+mu18;
     A_HvsW: test  mu7+mu8+mu9+mu10+mu11+mu12 = mu13+mu14+mu15+mu16+mu17+mu18;
```

```
Dependent Variable: MEANLNG
Test: A_GVSH   Numerator:  31217.5679  DF:    1   F value:  32.0781
               Denominator:  973.1736  DF:   90   Prob>F:    0.0001

Dependent Variable: MEANLNG
Test: A_GVSW   Numerator: 217443.4318  DF:    1   F value: 223.4374
               Denominator:  973.1736  DF:   90   Prob>F:    0.0001

Dependent Variable: MEANLNG
Test: A_HVSW   Numerator:  83881.6915  DF:    1   F value:  86.1940
               Denominator:  973.1736  DF:   90   Prob>F:    0.0001
```



As mentioned earlier, GP159 not only has a smaller average lesion length, but it seems to exhibit less variation in its vulnerability to different strains of fungus. Part of the significant interaction must come from this, and part from differences in the profiles of Hanna and Westar. Two questions arise:

1. Are μ_1 through μ_6 (the means for GP159) actually different from each other?
2. Are the profiles for Hanna and Westar different?

There are two natural ways to address these questions. The naive way is to subset the data --- that is, do a one-way ANOVA to compare the 6 means for GP159, and a two-way (2 by 6) on the Hanna-Westar subset. In the latter analysis, the interaction of Plant by MCG would indicate whether the two profiles were different.

A more sophisticated approach is not to subset the data, but to recognize that both questions can be answered by testing collections of contrasts of the entire set of 18 means; it's easy to do with the `test` statement of `proc reg`., or with `contrast` statements in `proc glm` -- see Chapter 6.

The advantage of the sophisticated approach is this. Remember that the model specifies a conditional normal distribution of the dependent variable for each combination of independent variable values (in this case there are 18 combinations of independent variable values), and that each conditional distribution has the *same variance*. The test for, say, the equality of μ_1 through μ_6 would use only $\bar{Y}_1$ through $\bar{Y}_6$ (that is, just GP159 data) to estimate the 5 contrasts involved, but it would use *all* the data to estimate the common error variance. From both a commonsense viewpoint and the deepest possible theoretical viewpoint, it's better not to throw information away. This is why the sophisticated approach should be better.

However, this argument is convincing only if it's really true that the dependent variable has the same variance for every combination of independent variable values. Repeating some output from the `means` command of the very first `proc glm`,

Level of PLANT	Level of MCG	N	MEANLNG Mean	MEANLNG SD
GP159	1	6	12.863095	12.8830306
GP159	2	6	21.623810	17.3001296
GP159	3	6	14.460714	7.2165396
GP159	7	6	17.686905	16.4258441
GP159	8	6	8.911905	7.3162618
GP159	9	6	8.784524	6.5970501
HANNA	1	6	45.578571	26.1430472
HANNA	2	6	67.296429	30.2424997
HANNA	3	6	94.192857	20.2877876
HANNA	7	6	53.621429	24.8563497
HANNA	8	6	47.838095	12.6419109
HANNA	9	6	25.673810	17.1723150
WESTAR	1	6	65.908333	35.6968616
WESTAR	2	6	187.479762	45.1992178
WESTAR	3	6	154.103571	26.5469183
WESTAR	7	6	173.972619	79.1793105
WESTAR	8	6	95.823810	22.3712022
WESTAR	9	6	66.502381	52.5253101

we see that the sample standard deviations for GP159 look quite a bit smaller on average. Without bothering to do a formal test, we have some reason to doubt the equal variances assumption.

It's easy to see *why* GP159 would have less plant-to-plant variation in lesion length. It's so resistant to the fungus that there's just not that much fungal growth, period. So there's less *opportunity* for variation.

Note that the equal variances assumption is essentially just a mathematical convenience. Here, it's clearly unrealistic. But what's the consequence of violating it? It's well known that the equal variance assumption can be safely violated if the cell sample sizes are equal and large. Well, here they're equal, but n=6 is not large. So this is not reassuring.

In general, it's not easy to say HOW the tests will be affected when the equal variance assumption is violated, but for the two particular cases we're interested in here (are the GP159 means equal and are the Hanna and Westar profiles parallel), we can figure it out. Recall Formula (3.3) for the F-test.

$$F = \frac{(SSR_F - SSR_R) / s}{MSE_F} .$$

The denominator --- Mean Squared Error from the full model --- is the estimated population error variance. That's the variance that's supposed to be the same for each conditional distribution. Since

$$MSE_F = \frac{\sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}{n - p} ,$$

and the predicted value $\hat{Y}_i$ is always the cell mean, we can draw the following conclusions.

1. When we test for equality of the GP159 means, using the Hanna-Westar data to help compute MSE will make the denominator of F bigger than it should be -- so F is made smaller, and the test is too conservative.

2. When we test whether the Hanna and Westar profiles are parallel, use of the GP159 data to help compute MSE will make the denominator of F *smaller* than it should be -- so F is made bigger, and the test is not conservative enough. That is, the chance of significance if the effect is absent will be greater than 0.05.

This makes me inclined to favour the "naive" subsetting approach. Because the GP159 means LOOK so equal, and I want them to be equal, I'd like to give the test for difference among them the best possible chance. And because it looks like the profiles for Hanna and Westar are not parallel (and I want them to be non-parallel, because it's more interesting for the effect of Fungus type to depend on type of Plant), I want a more conservative test.

Another argument in favour of subsetting is based on botany rather than statistics. Hanna and Westar are commercial canola crop varieties, but while GP159 is definitely in the canola family, it is more like a hardy weed than a food plant. It's just a different kind of entity, and so analyzing its data separately makes a lot of sense.

You may wonder, if it's so different, why was it included in the design in the first place? Well, taxonomically it's quite similar to Hanna and Westar; really no one knew it would be such a vigorous monster in terms of resisting fungus. That's why people do research -- to find out things they didn't already know.

Anyway, we'll do the analysis both ways -- both the seemingly naive way which is probably better once you think about it, and the sophisticated way that uses the complete set of data for all analyses.

Parts B and C represent the "sophisticated" approach that does not subset the data.

B. Test all GP159 means equal, full design

C. Test profiles for Hanna & Westar parallel, full design

```
proc reg;
     model meanlng = mu1-mu18 / noint;
     A_GvsH:   test  mu1+mu2+mu3+mu4+mu5+mu6 = mu7+mu8+mu9+mu10+mu11+mu12;
     A_GvsW:   test  mu1+mu2+mu3+mu4+mu5+mu6 = mu13+mu14+mu15+mu16+mu17+mu18;
     A_HvsW:   test  mu7+mu8+mu9+mu10+mu11+mu12 = mu13+mu14+mu15+mu16+mu17+mu18;
     B_G159eq: test  mu1=mu2=mu3=mu4=mu5=mu6;
     C_HWpar:  test  mu8-mu7=mu14-mu13,  mu9-mu8=mu15-mu14,
                     mu10-mu9=mu16-mu15, mu11-mu10=mu17-mu16,
                     mu12-mu11=mu18-mu17;


Dependent Variable: MEANLNG
Test: B_G159EQ Numerator:    151.5506  DF:    5   F value:   0.1557
               Denominator:  973.1736  DF:   90   Prob>F:    0.9778

Dependent Variable: MEANLNG
Test: C_HWPAR  Numerator:   5364.0437  DF:    5   F value:   5.5119
               Denominator:  973.1736  DF:   90   Prob>F:    0.0002
```

This confirms the visual impression of no differences among means for GP159, and non-parallel profiles for Hanna and Westar. Now compare the subsetting approach. Notice the creation of SAS data sets with subsets of the data.

D. Oneway on mcg, GP158 subset

E. Plant by MCG, Hanna-Westar subset

```
data just159; /* This data set will have just GP159 */
     set mould;
     if plant=1;

proc glm data=just159;
     title 'D.  Oneway on mcg, GP158 subset';
     class mcg;
     model meanlng = mcg;
```

```
                        D.  Oneway on mcg, GP158 subset                        2
                                                  10:52 Friday, February 22, 2002

                        General Linear Models Procedure

Dependent Variable: MEANLNG   Average Lesion length
                                       Sum of            Mean
Source                     DF          Squares          Square  F Value    Pr > F

Model                       5     757.75319161    151.55063832     1.03    0.4189

Error                      30    4421.01258503    147.36708617

Corrected Total            35    5178.76577664

                   R-Square             C.V.        Root MSE       MEANLNG Mean

                   0.146319         86.37031       12.139485          14.055159


Source                     DF        Type I SS     Mean Square  F Value    Pr > F

MCG                         5     757.75319161    151.55063832     1.03    0.4189

Source                     DF      Type III SS     Mean Square  F Value    Pr > F

MCG                         5     757.75319161    151.55063832     1.03    0.4189
```

This analysis is consistent with what we got without subsetting the data. That is, it does not provide evidence that the means for GP159 are different. But when we didn't subset the data, we had p = 0.9778. This happened exactly because including Hanna and Westar data made MSE larger, F smaller, and hence p bigger.

```
data hanstar; /* This data set will have just Hanna and Westar */
     set mould;
     if plant ne 1;

proc glm data=hanstar;
     title 'E.  Plant by MCG, Hanna-Westar subset';
     class plant mcg;
     model meanlng = plant|mcg;
```

E. Plant by MCG, Hanna-Westar subset 3
10:52 Friday, February 22, 2002

General Linear Models Procedure
Class Level Information

Class	Levels	Values
PLANT	2	HANNA WESTAR
MCG	6	1 2 3 7 8 9

Number of observations in data set = 72

E. Plant by MCG, Hanna-Westar subset 4
10:52 Friday, February 22, 2002

General Linear Models Procedure

Dependent Variable: MEANLNG Average Lesion length

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	11	189445.68433	17222.33494	12.43	0.0001
Error	60	83164.61331	1386.07689		
Corrected Total	71	272610.29764			

R-Square	C.V.	Root MSE	MEANLNG Mean
0.694932	41.44379	37.230054	89.832639

```
Source                    DF        Type I SS     Mean Square  F Value    Pr > F

PLANT                      1     83881.691486    83881.691486    60.52    0.0001
MCG                        5     78743.774570    15748.754914    11.36    0.0001
PLANT*MCG                  5     26820.218272     5364.043654     3.87    0.0042

Source                    DF      Type III SS     Mean Square  F Value    Pr > F

PLANT                      1     83881.691486    83881.691486    60.52    0.0001
MCG                        5     78743.774570    15748.754914    11.36    0.0001
PLANT*MCG                  5     26820.218272     5364.043654     3.87    0.0042
```

=========

The significant interaction indicates that the profiles for Hanna and Westar are non-parallel, confirming the visual impression we got from the interaction plot. But the p-value is larger this time. When all the data were used to calculate the error term, we had p = 0.0002. This is definitely due to the low variation in GP159.

Further analyses will be limited to the Hanna-Westar subset.

Now think of the interaction in a different way. Overall, Hanna is more vulnerable than Westar, but the interaction says that the degree of that greater vulnerability depends on the type of fungus. Look at all pairwise comparisons of the DIFFERENCE between Hanna and Westar. First, verify that the interaction can be expressed this way. Of course it can.

F. Plant by MCG followup, Hanna-Westar subset

All pairwise differences of Westar minus Hanna differences

```
proc reg;
     model meanlng = mu7-mu18 / noint;
     F_inter: test   mu13-mu7=mu14-mu8=mu15-mu9
                   = mu16-mu10=mu17-mu11=mu18-mu12;
     F_1vs2:  test   mu13-mu7=mu14-mu8;
     F_1vs3:  test   mu13-mu7=mu15-mu9;
     F_1vs7:  test   mu13-mu7=mu16-mu10;
     F_1vs8:  test   mu13-mu7=mu17-mu11;
     F_1vs9:  test   mu13-mu7=mu18-mu12;
     F_2vs3:  test   mu14-mu8=mu15-mu9;
     F_2vs7:  test   mu14-mu8=mu16-mu10;
     F_2vs8:  test   mu14-mu8=mu17-mu11;
     F_2vs9:  test   mu14-mu8=mu18-mu12;
     F_3vs7:  test   mu15-mu9=mu16-mu10;
     F_3vs8:  test   mu15-mu9=mu17-mu11;
```

```
     F_3vs9:  test   mu15-mu9=mu18-mu12;
     F_7vs8:  test   mu16-mu10=mu17-mu11;
     F_7vs9:  test   mu16-mu10=mu18-mu12;
     F_8vs9:  test   mu17-mu11=mu18-mu12;


Dependent Variable: MEANLNG
Test: F_INTER  Numerator:   5364.0437  DF:    5   F value:   3.8699
               Denominator:  1386.077  DF:   60   Prob>F:    0.0042


Dependent Variable: MEANLNG
Test: F_1VS2   Numerator:  14956.1036  DF:    1   F value:  10.7902
               Denominator:  1386.077  DF:   60   Prob>F:    0.0017


Dependent Variable: MEANLNG
Test: F_1VS3   Numerator:   2349.9777  DF:    1   F value:   1.6954
               Denominator:  1386.077  DF:   60   Prob>F:    0.1979


Dependent Variable: MEANLNG
Test: F_1VS7   Numerator:  15006.4293  DF:    1   F value:  10.8265
               Denominator:  1386.077  DF:   60   Prob>F:    0.0017


Dependent Variable: MEANLNG
Test: F_1VS8   Numerator:   1147.2776  DF:    1   F value:   0.8277
               Denominator:  1386.077  DF:   60   Prob>F:    0.3666


Dependent Variable: MEANLNG
Test: F_1VS9   Numerator:    630.3018  DF:    1   F value:   0.4547
               Denominator:  1386.077  DF:   60   Prob>F:    0.5027


Dependent Variable: MEANLNG
Test: F_2VS3   Numerator:   5449.1829  DF:    1   F value:   3.9314
               Denominator:  1386.077  DF:   60   Prob>F:    0.0520


Dependent Variable: MEANLNG
Test: F_2VS7   Numerator:      0.0423  DF:    1   F value:   0.0000
               Denominator:  1386.077  DF:   60   Prob>F:    0.9956


Dependent Variable: MEANLNG
Test: F_2VS8   Numerator:   7818.7443  DF:    1   F value:   5.6409
               Denominator:  1386.077  DF:   60   Prob>F:    0.0208


Dependent Variable: MEANLNG
Test: F_2VS9   Numerator:   9445.7674  DF:    1   F value:   6.8147
```

```
                Denominator:  1386.077  DF:    60   Prob>F:    0.0114


Dependent Variable: MEANLNG
Test: F_3VS7    Numerator:   5479.5767  DF:     1   F value:   3.9533
                Denominator:  1386.077  DF:    60   Prob>F:    0.0513


Dependent Variable: MEANLNG
Test: F_3VS8    Numerator:    213.3084  DF:     1   F value:   0.1539
                Denominator:  1386.077  DF:    60   Prob>F:    0.6962


Dependent Variable: MEANLNG
Test: F_3VS9    Numerator:    546.1923  DF:     1   F value:   0.3941
                Denominator:  1386.077  DF:    60   Prob>F:    0.5326


Dependent Variable: MEANLNG
Test: F_7VS8    Numerator:   7855.1432  DF:     1   F value:   5.6672
                Denominator:  1386.077  DF:    60   Prob>F:    0.0205



Dependent Variable: MEANLNG
Test: F_7VS9    Numerator:   9485.7704  DF:     1   F value:   6.8436
                Denominator:  1386.077  DF:    60   Prob>F:    0.0112


Dependent Variable: MEANLNG
Test: F_8VS9    Numerator:     76.8370  DF:     1   F value:   0.0554
                Denominator:  1386.077  DF:    60   Prob>F:    0.8147
```

These analyses are summarized in the table below. Westar-Hanna differences marked with the same letter are not significantly different.

MCG	Westar-Hanna Difference		
7	120.35	A	
2	120.18	A	
3	59.91	A	B
8	47.98		B
9	40.83		B
1	20.33		B

The last two tests investigate whether there are significant differences in response to type of fungus, separately within Hanna and within Westar. We see that they are statistically significant for Westar, and almost reach significance for Hanna.

```
     G_Hanaeq: test   mu7=mu8=mu9=mu10=mu11=mu12;
     H_Westeq: test   mu13=mu14=mu15=mu16=mu17=mu18;


Dependent Variable: MEANLNG
Test: G_HANAEQ Numerator:   3223.5872  DF:    5   F value:   2.3257
               Denominator:  1386.077  DF:   60   Prob>F:    0.0536


Dependent Variable: MEANLNG
Test: H_WESTEQ Numerator:  17889.2114  DF:    5   F value:  12.9064
               Denominator:  1386.077  DF:   60   Prob>F:    0.0001
```

It makes sense to follow up with pairwise comparisons of the means with Westar, but first let's review what we've done so far, limiting the discussion to just the Hanna-Westar subset of the data. We've tested

- Overall difference among the 12 means
- Main effect for PLANT
- Main effect for MCG
- PLANT*MCG interaction
- 15 pairwise comparisons of the Hanna-Westar difference, following up the interaction
- One comparison of the 6 means for Hanna
- One comparison of the 6 means for Westar

That's 21 tests in all, and we really should do at least 15 more, testing for pairwise differences among the Westar means. Somehow, we should make this into a set of proper post-hoc tests, and correct for the fact that we've done a lot of them. But how?

Tukey tests are only good for pairwise comparisons, and a Bonferroni correction is very ill-advised, since these tests were not all planned before seeing the data. This pretty much leaves us with Scheffé or nothing. The earlier discussion of Scheffé tests was limited to testing single contrasts. Here, some of our involve testing collections of

contrasts, so we need a little more generality.

General Scheffé Tests Assume a multifactor design. Create a combination independent variable whose values are all combinations of factor levels. All the tests we do will be tests for collections consisting of one or more contrasts of the cell means.

Start with an *initial* test, an F-test for *s* contrasts. A Scheffé follow-up test will be a test for *d* contrasts, *not* necessarily a subset of the contrasts of the initial test. The follow-up test must obey these rules:

- $d < s$
- If all *s* contrasts of the initial test are zero in the population, then all *d* contrasts of the follow-up test must be zero in the population. In other words, the null hypothesis of the follow-up test must be implied by the null hypothesis of the initial test.

Next, compute the ordinary one-at-a-time F statistic for the follow-up test (it will have *d* and *n-p* degrees of freedom). Then, use a calculator to compute

$$F_{sch} = \frac{d}{s} F, \qquad (4.2)$$

and if F_{sch} is bigger than the critical value of F for the initial test, the Scheffé follow-up is significant.

Actually, Formula (4.2) is more general. It applies to testing linear combinations of regression coefficients in a multiple regression setting. The initial test is a test of *s* linear constraints on the regression coefficients, and the follow-up test is a test of *d* linear constraints, where $d < s$ and the linear constraints of the initial test imply the linear constraints of the follow-up test. This is very nice because it allows, for example, Scheffé follow-ups to a significant analysis of covariance.

Before applying Scheffé follow-ups to the greenhouse data, a few comments are in order.

- The term "linear constraints" sounds imposing, but a linear constraint is just a statement that some linear combination equals a constant. Almost always, the constant is zero. So for example, saying that a contrast of cell means is equal to zero is the same as specifying a linear constraint on the betas of a multiple regression model (with cell means coding).

° If you're testing 6 independent variables controlling for some other set of independent variables, the null hypothesis says that 6 regression coefficients are equal to zero. That's six linear constraints on the regression coefficients.

° In the initial one-way ANOVA setting where we were testing single contrasts of p cell means, the Scheffe F statistic was defined by $F_{sch} = F/(p-1)$. This was a special case of formula (4.2). The initial test for equality of p means involved $p-1$ contrasts, so $s = p-1$. The followup tests were all for single contrasts, so $d=1$.

° As in the case of testing single contrasts in a one-way design, it is impossible for a followup to be significant if the initial test is not. And if the initial test is significant, there is always something to find in the family of Scheffé follow-ups.

° Suppose we have a follow-up test for d linear constraints, and it's not significant. Then *every* follow-up test whose null hypothesis is implied by those constraints will also be non-significant. To use the metaphor of data fishing, once you've looked for fish in a particular region of the lake and determined that there's nothing there, further detailed exploration in that region is a waste of time.

Formula (4.2) is very simple to apply. There are only two potential complications, and they are related to one another.

° First, you have to know what significance test you are following up. For example, if your initial test is the test for equality of *all* cell means, then the test for a given main effect could be carried out as a Scheffé followup, and a pairwise comparison of marginal means would be another followup to the same initial test. Or, you could start with the test for the main effect. Then, the pairwise comparison of marginal means would be a follow-up to the one-at-a-time test for the main effect. You could do it either way, and the conclusions might differ. Where you start is a matter of data-analytic philosophy. But starting with the standard tests for main effects and interactions is more traditional.

° The second potential complication is that you really have to be sure that the null hypothesis of the initial test implies the null hypothesis of the follow-up test. In terms of `proc reg` syntax, it means that the `test` statement of the initial test implies the `test` statements of all the follow-up tests. Sometimes this is easy to check, and sometimes it is tricky. To a large extent, how easy it is to check depends on what the initial test is.

a. If the initial test is a test for all cell means being equal (a one-way ANOVA on the combination variable), then it's easy, because if all the cell means are equal, then any possible contrast of the cell means equals zero. The proof is one line of High School algebra.

b. Similarly, suppose we are using a regression model with an intercept, and the initial test is for all the regression coefficients except β_0 simultaneously. This means that the null hypothesis of the initial test is $\beta_1 = ... = \beta_{p-1} = 0$, and therefore any linear combination of those quantities is zero. This means that you can test any subset of independent variables controlling for all the others as a proper Scheffé follow-up to the first test SAS prints.

c. If you're following up tests for main effects, then the standard test for any contrast of marginal means is a proper follow-up to the test for the main effect.

Beyond these principles, the logical connection between initial and follow-up tests really needs to be checked on a case-by-case basis. Often, the initial test can be expressed more than one way in the `test` statement of `proc reg`, and one of those statements will make things clear enough so you don't need to do any algebra. This is what I did with the significant Plant by Fungus interaction for the Hanna-Westar subset. When the interaction was written as

```
F_inter: test   mu13-mu7=mu14-mu8=mu15-mu9
              = mu16-mu10=mu17-mu11=mu18-mu12;
```

it was clear that all the pairwise comparisons of Westar-Hanna differences were implied.

```
F_1vs2:  test   mu13-mu7=mu14-mu8;
F_1vs3:  test   mu13-mu7=mu15-mu9;
F_1vs7:  test   mu13-mu7=mu16-mu10;
F_1vs8:  test   mu13-mu7=mu17-mu11;
F_1vs9:  test   mu13-mu7=mu18-mu12;
F_2vs3:  test   mu14-mu8=mu15-mu9;
F_2vs7:  test   mu14-mu8=mu16-mu10;
F_2vs8:  test   mu14-mu8=mu17-mu11;
F_2vs9:  test   mu14-mu8=mu18-mu12;
F_3vs7:  test   mu15-mu9=mu16-mu10;
F_3vs8:  test   mu15-mu9=mu17-mu11;
F_3vs9:  test   mu15-mu9=mu18-mu12;
F_7vs8:  test   mu16-mu10=mu17-mu11;
F_7vs9:  test   mu16-mu10=mu18-mu12;
F_8vs9:  test   mu17-mu11=mu18-mu12;
```

Sometimes it is easy to get this wrong. Just note that SAS will do all pairwise comparisons of marginal means (in the `means` statement of `proc glm`) as Scheffé follow-ups, but don't trust it unless the sample sizes are equal. Do it yourself. This warning applies up to SAS version 6.10. Is it a real error, or was it done deliberately to minimize calls to technical support? It's impossible to tell.

Now let's proceed, limiting the analysis to the Hanna-Westar subset. Just for fun, we'll start in two places. Our initial test will be either the test for equality of all 12 cell means, or the test for the Plant by Fungus interaction. Thus, we need two critical values.

```
proc iml; /* Critical values for Scheffe tests */
     interac = finv(.95,5,60) ; print interac;
     oneway = finv(.95,11,60); print oneway;

                                        INTERAC
                                     2.3682702

                                        ONEWAY
                                     1.9522119
```

Initial Test is for Difference Among 12 Cell Means

Let's start by treating the tests for main effects and the interaction as follow-ups to the significant ANOVA on the combination variable ($F = 12.43$; $df=11,71$; $p < .0001$). The table below is based on numbers displayed earlier.

Effect	One-at-a-time F	$F_{sch} = \frac{d}{s} F$	d	Significant with Scheffé?
PLANT	60.52	5.50	1	Yes
MCG	11.36	5.16	5	Yes
PLANT*MCG	3.87	1.75	5	No
All Hanna Equal?	2.33	1.06	5	No
All Westar Equal?	12.91	5.87	5	Yes

The main effect for Plant is still significant; it means that Westar is more vulnerable than Hanna. The main effect for Fungus (MCG) is significant, but as mentioned earlier, it should not be interpreted.

The interesting Plant by MCG interaction is no longer significant as a Scheffe test. This means that all the pairwise comparisons among Westar-Hanna differences will also be non-significant, as Scheffe follow-ups to the oneway ANOVA on the combination variable. There are no fish in that part of the lake. Just to check, the biggest Westar-Hanna difference was 120.35 for MCG 7, and the smallest was 20.33 for MCG 1. Comparing these two

differences yielded a one-at-a-time F of 10.83. But d=1 here and s=11, so that F_{sch}=.98. This falls short of the 1.95 required for significance, and as expected, none of the proper follow-ups to a non-significant follow-up are significant.

Pairwise comparisons of the Westar means are of interest, and the easiest way to get them is to ask `proc glm` for all pairwise comparisons of cell means.

```
proc glm data=hanstar;
     class combo;
     model meanlng = combo;
     means combo / scheffe;
```

```
                Scheffe's test for variable: MEANLNG

        NOTE: This test controls the type I experimentwise error rate but
              generally has a higher type II error rate than REGWF for all
              pairwise comparisons

                    Alpha= 0.05  df= 60  MSE= 1386.077
                       Critical Value of F= 1.95221
                   Minimum Significant Difference= 99.608

        Means with the same letter are not significantly different.

             Scheffe Grouping              Mean      N  COMBO

                        A            187.48      6  14
                        A
                        A            173.97      6  16
                        A
                     B  A            154.10      6  15
                     B  A
                     B  A   C         95.82      6  17
                     B  A   C
                     B  A   C         94.19      6  9
                     B      C
                     B      C         67.30      6  8
                     B      C
                     B      C         66.50      6  18
                     B      C
                     B      C         65.91      6  13
                            C
                            C         53.62      6  10
                            C
                            C         47.84      6  11
                            C
                            C         45.58      6  7
                            C
                            C         25.67      6  12
```

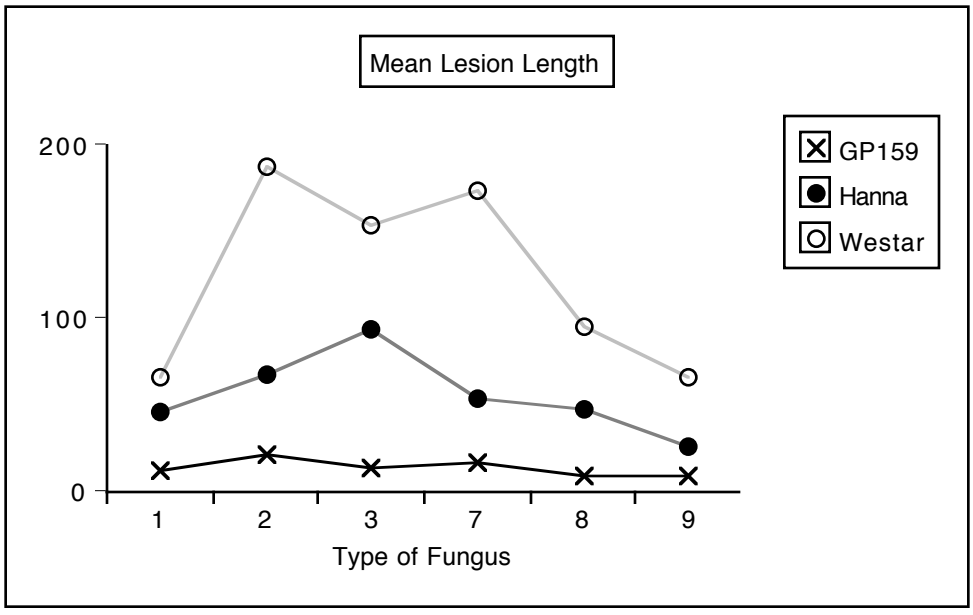
On Westar, Fungus types 2, 3 and 7 grow significantly faster than types 1 and 9, while type 8 is not significantly different from either group. As expected, there are no significant differences among Fungus types for Hanna.

Starting with the Interaction

Logically, a test for interaction can be a follow-up test, but almost no one ever does this in practice. It's much more traditional to start with a one-at a time test for interaction and then, if you're very sophisticated, do Scheffe follow-ups to that initial test. Now s = 5 and the critical value is `2.3682702`.

Again, the biggest Westar-Hanna difference was 120.35 for MCG 7, and the smallest was 20.33 for MCG 1. Comparing these two differences yielded a one-at-a-time F of 10.83. This yields $F_{sch} = \frac{d}{s} F = \frac{1}{5} * 10.83 = 2.16$. But this falls short of the critical value of 2.37, so none of the pairwise comparisons of Westar-Hanna differences reaches significance as a Scheffe follow-up -- even though they look very promising.

As a mathematical certainty, there *is* a single-contrast Scheffe follow-up to the interaction that is significant, but I am still looking for it. The next place I will look is: pairwise comparisons of the differences of line-segment slopes from the interaction plot.



Interactions as Products of Independent Variables

Categorical by Quantitative

An interaction between a quantitative variable and a categorical variable means that differences in E[Y] between categories depend on the value of the quantitative variable, or (equivalently) that the slope of the lines relating x to E[Y] are different, depending on category membership. Such an interaction is represented by **products** of the quantitative variable and the dummy variables for the categorical variable.

For example, consider the metric cars data (mcars.dat). It has length, weight, origin and fuel efficiency in kilometers per litre, for a sample of cars. The three origins are US, Japanese and Other. Presumably these refer to the location of the head office, not to where the car was manufactured.

Let's use indicator dummy variable coding for origin, with an intercept. In an Analysis of Covariance (ANCOVA), we'd test country of origin controlling, say, for weight. Letting x represent weight and c1 and c2 the dummy variables for country of origin, the model would be

$$E[Y] = b_0 + b_1x + b_2c_1 + b_3c_2.$$

This model assumes no interaction between country and weight. The following model includes product terms for the interaction, and would allow you to test it.

$$E[Y] = \beta_0 + \beta_1x + \beta_2c_1 + \beta_3c_2 + \beta_4c_1x + \beta_5c_2x$$

Country	c1	c2	Expected KPL (let x = weight)
U. S.	1	0	$(\beta_0 + \beta_2) + (\beta_1+\beta_4)x$
Japan	0	0	$\beta_0 + \beta_1 x$
European	0	1	$(\beta_0 + \beta_3) + (\beta_1+\beta_5)x$

It's clear that the slopes are parallel if and only if $\beta_4=\beta_5=0$, and that in this case the relationship of fuel efficiency to country would not depend on weight of the car.

As the program below shows, interaction terms are created by literally multiplying independent variables, and using products as additional independent variables in the regression equation.

```
/********************* mcars.sas *************************/
options linesize=79 pagesize=100 noovp formdlim='-';
title 'Metric Cars Data: Dummy Vars and Interactions';

proc format; /* Used to label values of the categorical variables */
     value carfmt    1 = 'US'
                     2 = 'Japanese'
                     3 = 'European' ;
data auto;
     infile 'mcars.dat';
     input id country kpl weight length;
/* Indicator dummy vars: Ref category is Japanese */
     if country = 1 then c1=1;  else c1=0;
     if country = 3 then c2=1;  else c2=0;
     /* Interaction Terms */
     cw1 = c1*weight; cw2 = c2*weight;
     label country = 'Country of Origin'
           kpl = 'Kilometers per Litre';
     format country carfmt.;

proc means;
     class country;
     var weight kpl;

proc glm;
     title 'One-way ANOVA';
     class country;
     model kpl = country;
     means country / tukey;

proc reg;
     title 'ANCOVA';
     model kpl = weight c1 c2;
     country: test c1 = c2 = 0;

proc reg;
     title 'Test parallel slopes (Interaction)';
     model kpl = weight c1 c2 cw1 cw2;
     interac: test cw1 = cw2 = 0;
     useuro:  test cw1=cw2;
     country: test c1 = c2 = 0;
     eqreg:   test c1=c2=cw1=cw2=0;

proc iml; /* Critical value for Scheffe tests */
     critval = finv(.95,4,94) ; print critval;
```

```
/* Could do most of it with proc glm: ANCOVA, then test interaction */

proc glm;
     class country;
     model kpl = weight country;
     lsmeans country;

proc glm;
     class country;
     model kpl = weight country weight*country;
```

Let's take a look at the output. First, proc means indicates that the US cars get lower gas mileage, and that weight is a potential confounding variable.

```
 COUNTRY   N Obs  Variable  Label                        N           Mean
-------------------------------------------------------------------------
US            73  WEIGHT                              73        1540.23
                  KPL       Kilometers per Litre      73      8.1583562

Japanese      13  WEIGHT                              13        1060.27
                  KPL       Kilometers per Litre      13      9.8215385

European      14  WEIGHT                              14        1080.32
                  KPL       Kilometers per Litre      14     11.1600000
-------------------------------------------------------------------------

 COUNTRY   N Obs  Variable  Label                          Std Dev       Minimum
-----------------------------------------------------------------------------
US            73  WEIGHT                               327.7785402   949.5000000
                  KPL       Kilometers per Litre         1.9760813     5.0400000

Japanese      13  WEIGHT                               104.8370989   891.0000000
                  KPL       Kilometers per Litre         2.3976719     7.5600000

European      14  WEIGHT                               240.9106607   823.5000000
                  KPL       Kilometers per Litre         4.2440764     5.8800000
-----------------------------------------------------------------------------

 COUNTRY   N Obs  Variable  Label                          Maximum
-------------------------------------------------------------------
US            73  WEIGHT                                  2178.00
                  KPL       Kilometers per Litre        12.6000000

Japanese      13  WEIGHT                                  1237.50
                  KPL       Kilometers per Litre        14.7000000

European      14  WEIGHT                                  1539.00
                  KPL       Kilometers per Litre        17.2200000
-------------------------------------------------------------------
```

The one-way ANOVA indicates that fuel efficiency is significantly related to country of origin; country explains 17% of the variation in fuel efficiency.

```
                          General Linear Models Procedure

   Dependent Variable: KPL   Kilometers per Litre
                                       Sum of            Mean
   Source                    DF          Squares          Square  F Value    Pr > F

   Model                      2     121.59232403     60.79616201    10.09    0.0001
   Error                     97     584.29697197      6.02368012
   Corrected Total           99     705.88929600

                     R-Square             C.V.        Root MSE           KPL Mean
                     0.172254         27.90648       2.4543187          8.7948000
```

The Tukey follow-ups are not shown, but they indicate that only the US-European difference is significant. Maybe the US cars are less efficient because they are big and heavy. So let's do the same test, controlling for weight of car. Here's the SAS code. Note this is a standard Analysis of Covariance, and we're *assuming* no interaction.

```
proc reg;
     title 'ANCOVA';
     model kpl = weight c1 c2;
     country: test c1 = c2 = 0;
```

```
   Dependent Variable: KPL        Kilometers per Litre

                              Analysis of Variance

                                 Sum of          Mean
         Source           DF     Squares        Square      F Value       Prob>F

         Model             3    436.21151    145.40384       51.761       0.0001
         Error            96    269.67779      2.80914
         C Total          99    705.88930

             Root MSE       1.67605     R-square       0.6180
             Dep Mean       8.79480     Adj R-sq       0.6060
             C.V.          19.05728
```

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	T for H0: Parameter=0	Prob > |T|
INTERCEP	1	16.226336	0.76312281	21.263	0.0001
WEIGHT	1	-0.006041	0.00057080	-10.583	0.0001
C1	1	1.236147	0.57412989	2.153	0.0338
C2	1	1.459591	0.64565633	2.261	0.0260

Dependent Variable: KPL
Test: COUNTRY Numerator: 8.6168 DF: 2 F value: 3.0674
Denominator: 2.809144 DF: 96 Prob>F: 0.0511

First notice that by including weight, we're now explaining 61% of the variation, while before we explained just 17%. Also, while the effect for country was comfortably significant before we controlled for weight, now it narrowly fails to reach the traditional criterion (p = 0.0511). But to really appreciate these results, we need to make a table.

Country	c1	c2	$E[Y] = \beta_0 + \beta_1 x + \beta_2 c_1 + \beta_3 c_2$
U. S.	1	0	$(\beta_0 + \beta_2) + \beta_1 x$
Japan	0	0	$\beta_0 + \beta_1 x$
European	0	1	$(\beta_0 + \beta_3) + \beta_1 x$

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	T for H0: Parameter=0	Prob > |T|
INTERCEP	1	16.226336	0.76312281	21.263	0.0001
WEIGHT	1	-0.006041	0.00057080	-10.583	0.0001
C1	1	1.236147	0.57412989	2.153	0.0338
C2	1	1.459591	0.64565633	2.261	0.0260

Observe that both b_2 and b_3 are positive -- and significant. Before we controlled for weight, Japanese gas mileage was a little better than US, though not significantly so. Now, because b_2 estimates β_2, and β_2 is the population difference between U.S. and Japanese mileage (for any fixed weight), a positive value of b_2 means that once you control for weight, the U.S. cars are getting better gas mileage than the Japanese -- significantly better, too, if you believe the t-test and not the F-test.

The *direction* of the results has changed because we controlled for weight. This can happen.

Also, may seem strange that the tests for β_2 and β_3 are each significant individually, but the simultaneous test for both of them is not. But this the simultaneous test implicitly includes a comparison between U.S. and European cars, and they are very close, once you control for weight.

The best way to summarize these results would be to calculate Y-hat for each country of origin, with weight set equal to its mean value in the sample. Instead of doing that, though, let's first test the interaction, which this analysis is *assuming* to be absent.

```
proc reg;
     title 'Test parallel slopes (Interaction)';
     model kpl = weight c1 c2 cw1 cw2;
        interac: test cw1 = cw2 = 0;
        useuro:  test cw1=cw2;
        country: test c1 = c2 = 0;
        eqreg:   test c1=c2=cw1=cw2=0;
```

```
   Dependent Variable: KPL        Kilometers per Litre

                              Sum of          Mean
         Source           DF      Squares       Square      F Value       Prob>F

         Model             5    489.27223     97.85445       42.463       0.0001
         Error            94    216.61706      2.30444
         C Total          99    705.88930

             Root MSE       1.51804     R-square       0.6931
             Dep Mean       8.79480     Adj R-sq       0.6768
             C.V.          17.26062

                              Parameter Estimates

                       Parameter      Standard    T for H0:
        Variable  DF      Estimate         Error   Parameter=0    Prob > |T|

        INTERCEP   1     29.194817    4.45188417         6.558        0.0001
        WEIGHT     1     -0.018272    0.00418000        -4.371        0.0001
        C1         1    -12.973668    4.53404398        -2.861        0.0052
        C2         1     -4.891978    4.85268101        -1.008        0.3160
        CW1        1      0.013037    0.00421549         3.093        0.0026
        CW2        1      0.006106    0.00453064         1.348        0.1810
```

```
---------------------------------------------------------------------------

Dependent Variable: KPL
Test: INTERAC  Numerator:     26.5304  DF:    2   F value:  11.5127
               Denominator:  2.304437  DF:   94   Prob>F:    0.0001

Dependent Variable: KPL
Test: USEURO   Numerator:     33.0228  DF:    1   F value:  14.3301
               Denominator:  2.304437  DF:   94   Prob>F:    0.0003

Dependent Variable: KPL
Test: COUNTRY  Numerator:     24.4819  DF:    2   F value:  10.6238
               Denominator:  2.304437  DF:   94   Prob>F:    0.0001

Dependent Variable: KPL
Test: EQREG    Numerator:     17.5736  DF:    4   F value:   7.6260
               Denominator:  2.304437  DF:   94   Prob>F:    0.0001
```

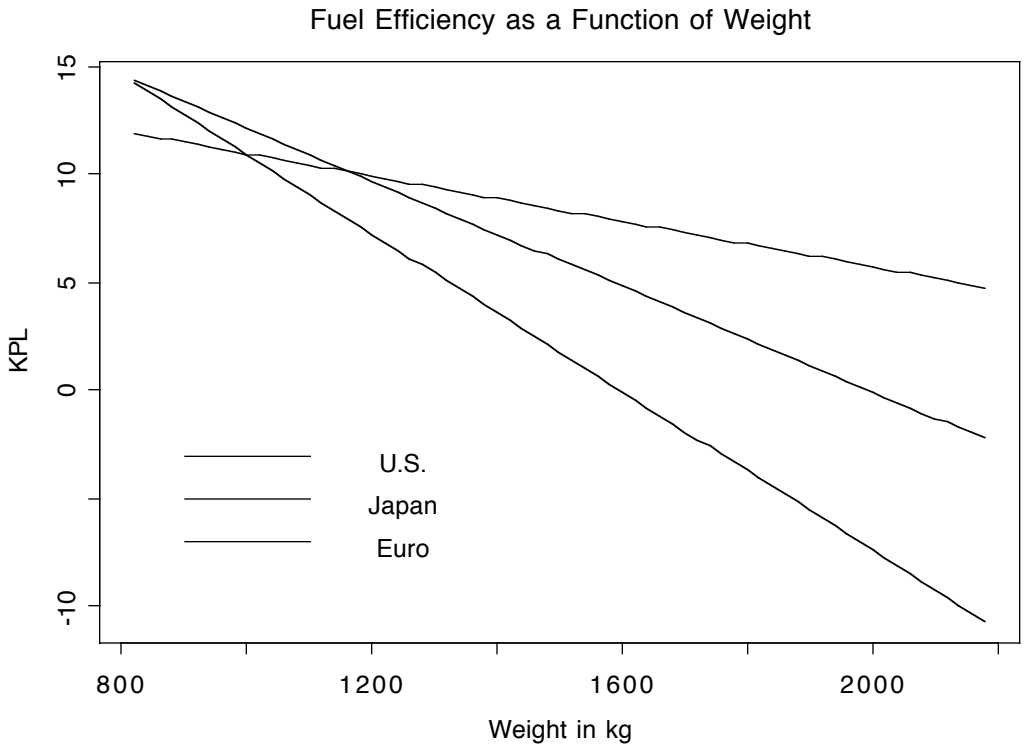
Now the coefficients for the dummy variables are both negative, and the coefficients for the interaction terms are positive. To see what's going on, we need a table *and* a picture -- of $\hat{Y}$.

$$\hat{Y} = b_0 + b_1x + b_2c_1 + b_3c_2 + b_4c_1x + b_5c_2x$$

$$= 29.194817 - 0.018272x - 12.973668c_1 - 4.891978c_2 + 0.013037c_1x + 0.006106c_2x$$

Country	c1	c2	Predicted KPL (let x = weight)
U. S.	1	0	$(b_0 + b_2) + (b_1+b_4)x$ = 16.22 - 0.005235 x
Japan	0	0	$b_0 + b_1 x$ = 29.19 - 0.018272 x
European	0	1	$(b_0 + b_3) + (b_1+b_5)x$ = 24.30 - 0.012166 x

From the proc means output, we find that the lightest car was 823.5kg, while the heaviest was 2178kg. So we will let the graph range from 820 to 2180.



When there were no interaction terms, b2 and b3 represented a main effect for country. What do they represent now?

From the picture, it is clear that the most interesting thing is that the slope of the line relating weight to fuel efficiency is least steep for the U.S. Is it significant? $0.05/3 = 0.0167$.

Repeating earlier material, ...

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	T for H0: Parameter=0	Prob > |T|
INTERCEP	1	29.194817	4.45188417	6.558	0.0001
WEIGHT	1	-0.018272	0.00418000	-4.371	0.0001
C1	1	-12.973668	4.53404398	-2.861	0.0052
C2	1	-4.891978	4.85268101	-1.008	0.3160
CW1	1	0.013037	0.00421549	3.093	0.0026
CW2	1	0.006106	0.00453064	1.348	0.1810

```
useuro:  test cw1=cw2;
```

```
Dependent Variable: KPL
Test: USEURO   Numerator:     33.0228  DF:    1   F value:  14.3301
               Denominator:  2.304437  DF:   94   Prob>F:    0.0003
```

The conclusion is that with a Bonferroni correction, the slope is less (less steep) for US than for either Japanese or European, but Japanese and European are not significantly different from each other.

Another interesting follow-up would be to use Scheffé tests to compare the heights of the regression lines at many values of weight; infinitely many comparisons would be protected simultaneously. This is not a proper follow-up to the interaction. What is the initial test?

Quantitative by Quantitative

An interaction of two quantitative variables is literally represented by their product. For example, consider the model

$$E[Y] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2$$

Hold x_2 fixed at some particular value, and re-arrange the terms. This yields

$$E[Y] = (\beta_0 + \beta_2 x_2) + (\beta_1 + \beta_3 x_2) x_1.$$

so that there is a linear relationship between x_1 and E[Y], with both the slope and the intercept depending on the value of x_2. Similarly, for a fixed value of x_1,

$$E[Y] = (\beta_0 + \beta_1 x_1) + (\beta_2 + \beta_3 x_1) x_2,$$

and the (linear) relationship of x_2 to E[Y] depends on the value of x_1. We always have this kind of symmetry.

Three-way interactions are represented by 3-way products, etc. Its interpretation would be "the 2-way interaction depends ..."

Product terms represent interactions ONLY when all the variables involved and all lower order interactions involving those variables are also included in the model!

Categorical by Categorical

It is no surprise that interactions between categorical independent variables are represented by products. If A and B are categorical variables, IVs representing the A by B interaction are obtained by multiplying each dummy variable for A by each dummy variable for B. If there is a third IV cleverly named C and you want the 3-way interaction, multiply each of the dummy variables for C by each of the products representing the A by B interaction. This rule extends to interactions of any order.

Up till now, we have represented categorical independent variables with indicator dummy variables, coded 0 or 1. If interactions between categorical IVs are to be represented, it is much better to use "effect coding," so that the regression coefficients for the dummy variables correspond to main effects. (In a 2-way design, products of indicator dummy variables still correspond to interaction terms, but if an interaction is present, the interpretation of the coefficients for the indicator dummy variables is not what you might guess.)

Effect coding. There is an intercept. As usual, a categorical independent variable with k categories is represented by k-1 dummy variables. The rule is

Dummy var 1: First value of the IV gets a 1, last gets a minus 1, all others get zero.
Dummy var 2: Second value of the IV gets a 1, last gets a minus 1, all others get zero.

. . .

Dummy var k-1: k-1st value of the IV gets a 1, last gets a minus 1, all others get zero.

Here is a table showing effect coding for Plant from the Greenhouse data.

Country	p1	p2	$E[Y] = \beta_0 + \beta_1 p_1 + \beta_2 p_2$
GP159	1	0	$\mu_1 = \beta_0 + \beta_1$
Hanna	0	1	$\mu_2 = \beta_0 + \beta_2$
Westar	-1	-1	$\mu_3 = \beta_0 - \beta_1 - \beta_2$

It is clear that $\mu_1 = \mu_2 = \mu_3$ if and only if $\beta_1=\beta_2=0$, so it's a valid dummy variable coding scheme even though it looks strange.

Country	p1	p2	E[Y] = $\beta_0 + \beta_1 p_1 + \beta_2 p_2$
GP159	1	0	$\mu_1 = \beta_0 + \beta_1$
Hanna	0	1	$\mu_2 = \beta_0 + \beta_2$
Westar	-1	-1	$\mu_3 = \beta_0 - \beta_1 - \beta_2$

Effect coding has these properties, which extend to any number of categories.

- $\mu_1 = \mu_2 = \mu_3$ if and only if $\beta_1=\beta_2=0$.
- The average population mean (grand mean) is $(\mu_1+\mu_2+\mu_3)/3 = \beta_0$.
- β_1, β_2 and $-(\beta_1+\beta_2)$ are deviations from the grand mean.

The real advantage of effect coding is that the dummy variables behave nicely when multiplied together, so that main effects correspond to collections of dummy variables, and interactions correspond to their products -- in a simple way. This is illustrated for Plant by MCG analysis, using the full greenhouse data set).

```
data nasty;
     set yucky;
     /* Two dummy variables for plant */
        if plant=. then p1=.;
        else if plant=1 then p1=1;
        else if plant=3 then p1=-1;
        else p1=0;
     if plant=. then p2=.;
        else if plant=2 then p2=1;
        else if plant=3 then p2=-1;
```

```
      else p2=0;
   /* Five dummy variables for mcg */
   if mcg=. then f1=.;
      else if mcg=1 then f1=1;
      else if mcg=9 then f1=-1;
      else f1=0;
   if mcg=. then f2=.;
      else if mcg=2 then f2=1;
      else if mcg=9 then f2=-1;
      else f2=0;
   if mcg=. then f3=.;
      else if mcg=3 then f3=1;
      else if mcg=9 then f3=-1;
      else f3=0;
   if mcg=. then f4=.;
      else if mcg=7 then f4=1;
      else if mcg=9 then f4=-1;
      else f4=0;
   if mcg=. then f5=.;
      else if mcg=8 then f5=1;
      else if mcg=9 then f5=-1;
      else f5=0;
   /* Product terms for the interaction */
      p1f1 = p1*f1; p1f2=p1*f2 ; p1f3=p1*f3 ; p1f4=p1*f4; p1f5=p1*f5;
      p2f1 = p2*f1; p2f2=p2*f2 ; p2f3=p2*f3 ; p2f4=p2*f4; p2f5=p2*f5;

proc reg;
   model meanlng = p1 -- p2f5;
   plant:  test p1=p2=0;
   mcg:    test f1=f2=f3=f4=f5=0;
   p_by_f: test p1f1=p1f2=p1f3=p1f4=p1f5=p2f1=p2f2=p2f3=p2f4=p2f5 = 0;
```

Here is the output from the test statement. For comparison, it is followed by `proc glm` output from `model meanlng = plant|mcg`.

```
Dependent Variable: MEANLNG
Test: PLANT    Numerator: 110847.5637  DF:    2   F value: 113.9032
               Denominator:  973.1736  DF:   90   Prob>F:    0.0001


Dependent Variable: MEANLNG
Test: MCG      Numerator:  11748.0529  DF:    5   F value:  12.0719
               Denominator:  973.1736  DF:   90   Prob>F:    0.0001


Dependent Variable: MEANLNG
Test: P_BY_F   Numerator:   4758.1481  DF:   10   F value:   4.8893
               Denominator:  973.1736  DF:   90   Prob>F:    0.0001
```

Source	DF	Type III SS	Mean Square	F Value	Pr > F
PLANT	2	221695.12747	110847.56373	113.90	0.0001
MCG	5	58740.26456	11748.05291	12.07	0.0001
PLANT*MCG	10	47581.48147	4758.14815	4.89	0.0001

It worked.

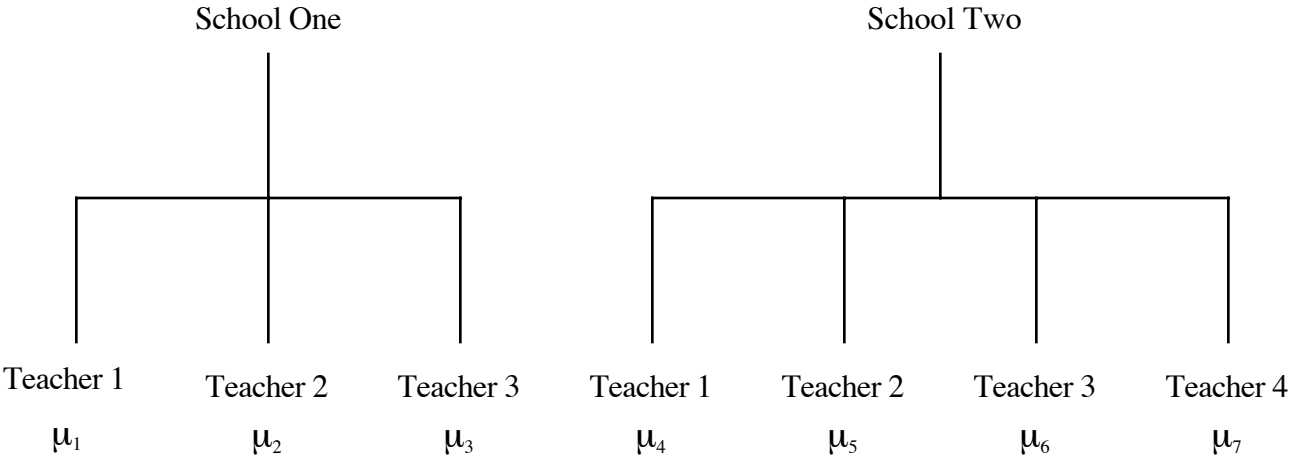
Effect coding works as expected in conjunction with quantitative independent variables. In particular, products of quantitative and indicator variables still represent interactions. In fact, the big advantage of effect coding is that you can use it to test categorical independent variables, and interactions between categorical independent variables -- in a bigger multiple regression context.

Nested and Random Effect models

Nested Designs

Suppose a chain of commercial business colleges is teaching a software certification course. After 6 weeks of instruction, students take a certification exam and receive a score ranging from zero to 100. The owners of the business school chain want to see whether performance is related to which school students attend, or which instructor they have -- or both. They compare two schools; one of the schools has three instructors teaching the course, and the other school has 4 instructors teaching the course. A teacher only works in one school.

There are two independent variables, school and teacher. But it's not a factorial design, because ``Teacher 1" does not mean the same thing in School 1 and School 2; it's a different person. This is called a **nested** design. By the way, it's also **unbalanced**, because there are different numbers of teachers withing each school. We say that *teacher is nested within school*. The diagram below shows what is going on, and give a clue about how to conduct the analysis.



To compare schools, we want to test $\frac{1}{3}(\mu_1+\mu_2+\mu_3) = \frac{1}{4}(\mu_4+\mu_5+\mu_6+\mu_7)$.

To compare instructors within schools, we want to test $\mu_1=\mu_2=\mu_3$ and $\mu_4=\mu_5=\mu_6=\mu_7$ simultaneously.

The first test involves one contrast of μ_1 through μ_7; the second test involves five contrasts. There really is nothing to it.

You can do it with `proc reg` and cell means coding, or you can take advantage of `proc glm`'s syntax for nested models.

```
proc glm;
     class school teacher;
     model score = school teacher(school);
```

The notation `teacher(school)` should be read ``teacher within school."

- It's easy to extend this to more than one level of nesting. You could have climate zones, lakes within climate zones, fishing boats within lakes, ...

- There is no problem with combining nested and factorial structures. You just have to keep track of what's nested within what. Factors that are not nested are sometimes called ``crossed."

Random Effect Models The preceding discussion (and indeed, the entire course to this point) has been limited to ``fixed effects" models. In a **random effects** model, *the values of the categorical independent variables represent a random sample from some population of values*. For example, suppose the business school had 200 branches, and just selected 2 of them at random for the investigation. Also, maybe each school has a lot of teachers, and we randomly sampled teachers within schools. Then, teachers within schools would be a random effects factor too.

It's quite possible to have random effect factors and fixed effect factors in the same design; such designs are called ``mixed." SAS `proc mixed` is built around this, but it does a lot of other things too.

Nested models are often viewed as random effects models, but there is no necessary connection between the two concepts. It depends on how the study was conducted. Were the two schools randomly selected from some population of schools, or did someone just pick those two (maybe because there are just two schools)?

Of course lots of the time, nothing is randomly selected -- but people use random effects models anyway. Why pretend? Well, sometimes they are thinking that in a better world, lakes *would* have been randomly selected. Or sometimes, the scientists are thinking that they really would like to generalize to the entire population of lakes, and therefore should use statistical tools that support such generalization -- even if there was no random sampling. (By the way, no statistical method can compensate for a biased sample.) Or sometimes it's just a tradition in certain sub-areas of research, and everybody expects to see random effects models.

In the traditional analysis of models with random or mixed effects and a normal assumption, F-tests are often possible, but they don't always use Mean Squared Error in the denominator of the F statistic. Often, it's the Mean Square for some interaction term or other. The choice of what error term to use is relatively mechanical for balanced models with equal sample sizes, but even then, sometimes (especially when it's a mixed model) a valid F-test for an effect of interest just doesn't exist.

When the design is unbalanced or has unequal sample sizes, a valid F-test rarely exists. It's a real pain. Sometimes, you can find an error term that produces a valid F-test *assuming* that some interaction (or maybe more than one interaction) is absent. Usually, you can't test for that interaction either. But people do it anyway and hope for the best.

SAS `proc mixed` goes a long way toward solving these problems. It's a great piece of software, based on recent, state-of the-art research as well as more venerable stuff. But we're running out of time. Goodbye, `proc mixed`. Goodbye, random effects.

Choosing Sample Size

The purpose of this section is to describe three related methods for choosing sample size before data are collected -- the classical power method, the sample variation method and the population variation method. The classical power method applies to almost any statistical test. After presenting general principles, the discussion zooms in on the important special case of factorial analysis of variance with no covariates. The sample variation method and the population variation methods are limited to multiple linear regression, including the analysis of variance and covariance. Throughout, it will be assumed that the person designing the study is a scientist who will only be allowed to discuss results if a null hypothesis is rejected at some conventional significance level such as $\alpha = 0.05$ or $\alpha = 0.01$. Thus, it is vitally important that the study be designed so that scientifically interesting effects are likely to be be detected as statistically significant.

The classical power method. The term "null hypothesis" has mostly been avoided until now, but it's much easier to talk about the classical power method if we're allowed to use it. Most statistical tests are based on comparing a full model to a reduced model. Under the reduced model, the values of population parameters are constrained in some way. For example, in a one-way ANOVA comparing three treatments, the parameters are μ_1, μ_2, μ_3 and σ^2. The reduced model says that $\mu_1=\mu_2=\mu_3$. This is a *constraint* on the parameter values. The **null hypothesis** (symbolized H_0) is a statement of how the parameters are constrained under the reduced model. When a test of a null hypothesis yields a small p-value, it means that the data are quite unlikely if the null hypothesis is true. We then reject the null hypothesis -- that is, we conclude it's not true, and therefore that some effect of interest is present in the population.

The following definition applies to hypothesis tests in general, not just those associated with common multiple regression. Assume that data are drawn from some population with parameter θ -- that's the Greek letter theta. Theta is typically a vector; for example, in simple linear regression with normal errors, $\theta = (\beta_0, \beta_1, \sigma^2)$.

The **power** of a statistical test is the probability of obtaining signific
the true parameter values. That is, it is a function of θ.

The **power** of a statistical test is the probability of obtaining significant results. Power is a function of the true parameter values. That is, it is a function of θ.

a) The common statistical tests have infinitely many power values.
b) If the null hypothesis is true, power cannot exceed å; in fact, this is the technical definition of α. Usually, $\alpha = 0.05$.
c) If the null hypothesis is false, more power is good.
d) For a good test, power $\rightarrow$ 1 (for fixed n) as the true parameter values get farther from those specified by the null hypothesis.
e) For a good test, power $\rightarrow$ 1 as n $\rightarrow \infty$ for any combination of fixed parameter values, provided the null hypothesis is false.

Classical power analysis is used to select a sample size n as follows. Choose an effect -- a particular combination of parameter values that makes the null hypothesis false. If possible, select the weakest effect that would still be scientifically important if it were present in the population. If the null hypothesis is false in this way, we would like to have a high probability of rejecting it and obtaining significance. Choose a sample size n, and calculate the probability of significance (that is, calculate power) for that sample size and that set of parameter values. Increase (or decrease) n, calculating power each time. Stop when the power is what you want. A common target value for power is 0.80. My guess is that it would be higher, except that, for common tests and effect sizes, the sample would have to be prohibitively large.

There are only two difficulties with carrying out a classical power analysis in practice; one is conceptual, the other technical. The conceptual problem is that scientists often have difficulty choosing a configuration of parameter values corresponding to an effect that is scientifically interesting. Maybe that's not too surprising, because scientists usually think in terms of data rather than in terms of statistical models. It could be different if the statistical models were serious scientific models of what the scientists are studying, but usually they're quite generic.

The technical problem is that sometimes -- especially for statistical methods other than those based on common multiple regression -- it can be difficult to calculate the probability of significance when the null hypothesis is false. This problem is not really serious; it can always be overcome with some effort and

the right software. Once you move beyond multiple regression, SAS is not the right software.

Power for Factorial ANOVA. Considering this special case will provide a concrete example of the classical power method. It is also the most common example of power analysis.

The distributions commonly used for practical hypothesis testing (mainly the chi-square, t and F) are ones that hold when the null hypothesis is true. When the null hypothesis is false, these are no longer the distributions of the common test statistics; instead, they have probability distributions that migrate more into the rejection region (tail area, above the critical value) of the statistical test. The F distribution used for testing hypotheses in multiple regression is the central F distribution. If the null hypothesis is *false*, the F statistic has a non-central F distribution with parameters s, n-p and ϕ. The quantity ϕ is a kind of squared distance between the reduced model and the true model. It is called the **non–centrality parameter** of the non-central F distribution; $\phi \geq 0$, and $\phi = 0$ gives the usual central F distribution. The larger the non-centrality parameter, the greater the chance of significance -- that is, the greater the power.

The general formula for ϕ is best written in the notation of matrix algebra; it will not be given here. But the general idea, and some of its essential properties, are shown by the special case where we are comparing two treatment means (as in a two-sample t-test, or a simple regression with a binary independent variable). In this situation, the general formula for the non-centrality parameter of the non-central F distribution reduces to

$$\phi = \frac{(\mu_1 - \mu_2)^2}{\sigma^2(\frac{1}{n_1} + \frac{1}{n_2})} = \frac{\delta^2}{(\frac{1}{n_1} + \frac{1}{n_2})}, \qquad (4.3)$$

where $\delta = \frac{|\mu_1 - \mu_2|}{\sigma}$. Right away, it is possible to make some useful comments.

$$\phi = \frac{(\mu_1 - \mu_2)^2}{\sigma^2(\frac{1}{n_1} + \frac{1}{n_2})} = \frac{\delta^2}{(\frac{1}{n_1} + \frac{1}{n_2})}, \qquad (4.3)$$

where $\delta \equiv \frac{|\mu_1 - \mu_2|}{\sigma}$.

- The quantity δ is called **effect size**. It specifies how wrong the statement $\mu_1=\mu_2$ is, by expressing the absolute difference between μ_1 and μ_2 in units of the common within-cell standard deviation σ.

- For any statistical test, power is a function of the parameter values. Here, the non-centrality parameter (and hence, power) depends on the three parameters μ_1, μ_2 and σ^2 *only* through the effect size. This is quite wonderful; it does not always happen, even in the analysis of variance.

- The larger the effect size (that is, the more wrong the reduced model is -- in this metric), the larger the non-centrality parameter ϕ, and therefore the larger the probability of significance.

- If $\mu_1=\mu_2$, then $\delta=0$, $\phi=0$,the non-central F distribution becomes the usual central F distribution, and the probability of significance becomes exactly $\alpha=0.05$.

- The size of the non-centrality parameter depends on another quantity involving *both* n_1 and n_2, not just the total sample size $n = n_1+n_2$.

This last point can be illuminated by a bit of algebra. Let

- $\delta = \frac{|\mu_1 - \mu_2|}{\sigma}$
- $n = n_1 + n_2$
- $q = \frac{n_1}{n}$, the proportion of the sample allocated to Group One.

Then expression (4.3) can be re-written

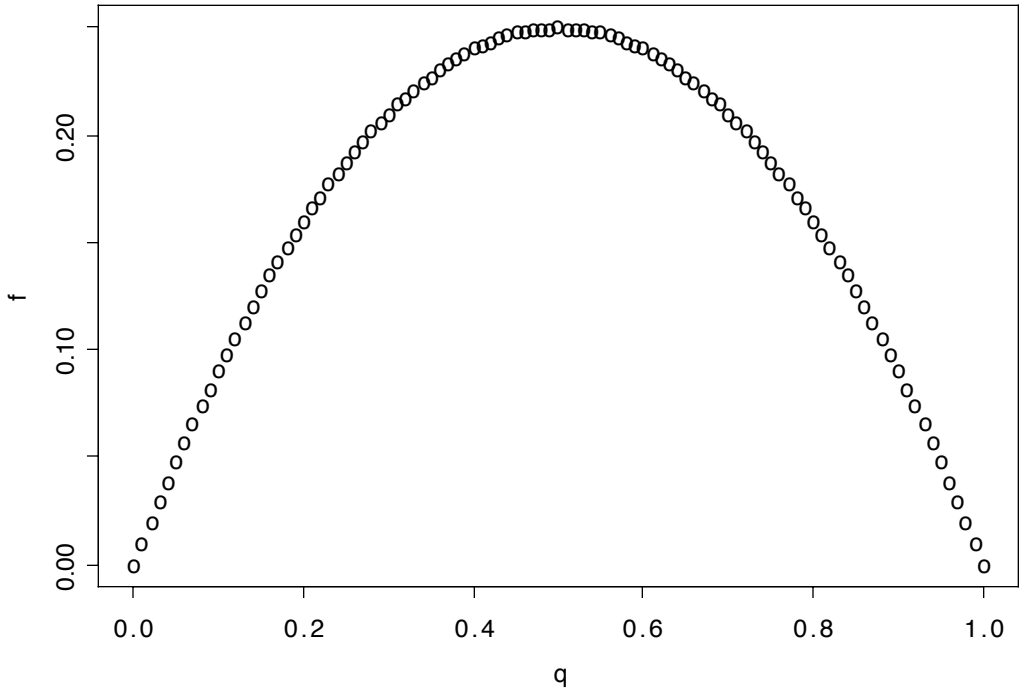
$$\phi = n\ q(1-q)\ \delta^2. \qquad (4.4)$$

Now it's clear.

- For any non-zero effect size and any (?) allocation of sample size to the two treatments, the greater the total sample size, the greater the power.
- For any sample size and any (?) allocation of sample size to the two treatments, the greater the effect size, the greater the power.
- Power depends not just on sample size and effect size, but on an aspect of *design* -- the allocation of sample size to the two treatments. This is a general feature of power in the analysis of variance and other statistical methods. It is important, but usually not mentioned.

Let's continue to pursue this interesting special case. For any given sample size and any non-zero effect size, we can maximize power by choosing q (the proportion of cases allocated to Group One) so that the function f(q) = q(1-q) is as large as possible. What's the best value of q?

This is a simple calculus exercise, but the following plot gives the answer by brute force. I just computed f(q) = q(1–q) for 100 equally spaced values of q ranging from zero to one.



So the best value of q is 1/2. That is, for comparing two means using the classical normal model, power is highest when the sample sizes are equal -- and this holds regardless of the total sample size or the magnitude of the effect.

This is a clear, simple example of something that holds for *any* classical ANOVA. The non-centrality parameter, and hence the power, depends on the total sample size, the effect, *and* the allocation of the sample to treatment combinations.

Equal sample sizes do not always yield the highest power. In general, the optimal allocation depends on the hypothesis being tested *and* the nature of the true effect. For example, suppose you have a design with 18 treatment combinations, and the test in question is to compare μ_1 with the average of μ_2 and μ_3. Further, suppose that $\mu_2 = \mu_3 \neq \mu_1$ (σ^2 can be anything); this is the effect. The optimal allocation is to give half the sample to Treatment One, split the other half any way at all between Treatments 2 and 3, and let n=0 for the other 15 treatments. This is why observations are not usually allocated to treatments based on a power analysis; it often advises you to put all your eggs in one basket.

In the analysis of variance, power analysis is used to select a sample size n as follows.

1. Choose an allocation of observations to treatments; usually, this is done without formal analysis, equal sample sizes being the most common choice.

2. Choose an effect. Your null hypothesis says that some collection of contrasts (of the treatment combination means) are all zero in the population. The "effect" you need to specify is that one or more of those contrasts is *not* zero. You must provide exact non-zero values, in units of the common within-treatment population standard deviation σ -- like, the difference between μ_1 and the average of μ_2 and μ_3 is minus 0.75σ. You don't need to know the numerical value of σ (thank goodness!), but you do need to be able to express differences between population means in units of σ. If possible, select the weakest effect that is still scientifically important.

3. Choose a desired power; again, a common choice is 0.80, but it's up to you.

4. Start with a modest but realistic value for the total sample size n. Increase it, each time determining the critical value of F, calculating the non-centrality parameter ϕ (you have enough information), and using ϕ to compute the probability that F will exceed the critical value. When that power becomes high enough, stop.

This is a rational strategy for choosing sample size. In practice, the hard part is selecting an effect. Scientists often can say what's a scientifically meaningful difference between means, but they usually have no clue about σ. Statisticians respond with the suggestion that σ^2 be estimated by MSE_F from similar studies. Scientists respond that there are no "similar" studies; the investigation being planned is new -- that's why we're doing it. In the end, the whole thing is based on so much guesswork that everyone feels uncomfortable. In my experience, this is what happens most of the time when people try to do a classical power analysis. Of course, there are exceptions; sometimes, everyone is happy.

The Sample Variation Method (Note STA442f05 has better sas programs. Fix this up!)

There are at least two main meanings of ``significance." One is statistical significance, and another is *explanatory* significance in the sense of explained variation. Formula (4.4) from Chapter 4 is relevant. It is reproduced here.

$$F = \left(\frac{n-p}{s}\right)\frac{a}{1-a}, \qquad (4.4)$$

where, after controlling for the effects in a reduced model, a is the proportion of the *remaining* variation that is explained by the full model.

Formula (4.4) tells us that the two meanings of ``significance" need not coincide, since statistical significance can come from either strong results or from a large sample. The sample variation method can be viewed as a way of bringing the two types of significance into agreement. It's not really a power analysis, but it is a rational way to decide on sample size.

In equation (4.4), F is an increasing function of both n and a, so its p-value (the tail area beyond F) is a decreasing function of both n and a. The sample variation method is to choose a value of a that is just large enough to be interesting, and increase n, calculating F and its p-value each time until $p < 0.05$; then stop. The final value of n is the smallest sample size for which an effect explaining that much of the remaining variation will be significant. With that sample size, the effect will be significant if and only if it explains a or more of the remaining variation.

That's all there is to it. You tell me a proportion of remaining variation that you want to be significant, and I'll tell you a sample size. In exchange, you agree not to moan and complain and go fishing for more covariates if your results are almost significant, because they were too weak to be interesting anyway.

There are two questions you might want to ask.

- For a given proportion of the remaining variation, what sample size do I need for statistical significance?

- For a given sample size, what proportion of the remaining variation do I need for statistical significance?

To make things more definite, let us suppose we are contemplating a 2x3x4 analysis of covariance, with two covariates and factors cleverly named A, B and C. We are setting it up as a regression model, with one dummy variable for A, 2 dummy variables for B, and 3 for C. Interactions are represented by product terms, and there are 2 products for the AxB interaction, 3 for AxC, 6 for BxC, and 1*2*3 = 6 for AxBxC. The regression coefficients for these plus two for the covariates and one for the intercept give us p = 26. The null hypothesis is that of no BxC interaction, so s = 6. The "other effects in the model" for which we are "controlling" are represented by 2 covariates and 17 dummy variables and products of dummy variables.

First, let's find out what sample size we need for the interaction to be significant provided it explains at least 10% of the remaining variation after controlling for other effects in the model. This is accomplished by the program `sampvar1.sas`. It is a little unusual in that it uses the SAS `put` statement to write results to the *log* file. It never produces a list file, because there is no `proc` step.

```
/************************  sampvar1.sas ************************/
/*      Finds n needed for significance, for a given proportion of */
/*      remaining variation                                        */
/*****************************************************************/

options linesize = 79 pagesize = 200;
data explvar;     /* Can replace alpha, s, p, and a below.   */
   alpha = 0.05;  /* Significance level.                     */
   s = 6;         /* Numerator df = # IVs being tested.      */
   p = 26;        /* There are p beta parameters.            */
   a = .10  ;     /* Proportion of remaining variation after */
                  /* controlling for all other variables.    */

   /* Initializing ... */  pval = 1; n = p+1;
   do until (pval <= alpha);
      F = (n-p)/s * a/(1-a);
      df2 = n-p;
      pval = 1-probf(F,s,df2);
      n = n+1 ;
   end;
   /* When finished, n is one too many */
   n = n-1; F = (n-p)/s * a/(1-a); df2 = n-p;
   pval = 1-probf(F,s,df2);

   put ' ******************************************************';
   put ' ';
   put '  For a multiple regression model with ' p 'betas, ';
   put '  testing ' s ' variables controlling for the others,';
   put '  a sample size of ' n 'is needed for significance at the';
   put '  alpha = ' alpha 'level, when the effect explains a = ' a ;
   put '  of the remaining variation after allowing for all other ';
   put '  variables in the model. ';
   put '  F = ' F ',df = (' s ',' df2 '), p = ' pval;
   put ' ';
   put ' ******************************************************';
```

Here is the part of the log file produced by the put statements.

```
 ******************************************************

  For a multiple regression model with 26 betas,
  testing 6  variables controlling for the others,
  a sample size of 144 is needed for significance at the
  alpha = 0.05 level, when the effect explains a = 0.1
  of the remaining variation after allowing for all other
  variables in the model.
  F = 2.1851851852 ,df = (6 ,118 ), p = 0.0491182815

 ******************************************************
```

Suppose you were considering n=120, and you wanted to know what proportion of the remaining variation the interaction must explain in order to be significant. This is accomplished by `sampvar2.sas`.

```
/*************************  sampvar2.sas ***************************/
/*  Finds proportion of remaining variation needed for significance, */
/*  given sample size n                                              */
/*******************************************************************/

options linesize = 79 pagesize = 200;
data explvar;     /* Replace alpha, s, p, and a below.  */
   alpha = 0.05;  /* Significance level.                */
   s = 6;         /* Numerator df = # IVs being tested. */
   p = 26;        /* There are p beta parameters.       */
   n = 120  ;     /* Sample size                        */

   /* Initializing ... */  pval = 1; a = 0; df2 = n-p;
   do until (pval <= alpha);
      F = (n-p)/s * a/(1-a);
      pval = 1-probf(F,s,df2);
      a = a + .001 ;
     end;
  /* When finished, a is .001 too much */
   a = a-.001; F = (n-p)/s * a/(1-a); pval = 1-probf(F,s,df2);

   put ' ******************************************************';
   put ' ';
   put '  For a multiple regression model with ' p 'betas, ';
   put '  testing ' s ' variables at significance level ';
   put '  alpha = ' alpha ' controlling for the other variables,';
   put '  and a sample size of ' n', the variables need to explain';
   put '  a = ' a ' of the remaining variation to be significant.';
   put '  F = ' F ', df = (' s ',' df2 '), p = ' pval;
   put '   ';
   put ' ******************************************************';
```

And here is the output.

```
*****************************************************

 For a multiple regression model with 26 betas,
 testing 6  variables at significance level
 alpha = 0.05  controlling for the other variables,
 and a sample size of 120 , the variables need to explain
 a = 0.123  of the remaining variation to be significant.
 F = 2.1972633979 , df = (6 ,94 ), p = 0.0499350803

*****************************************************
```

It's worth mentioning that the Sample Variation method is so simple that lots of people must know about it -- but I have never seen it described in print.

The Population Variation Method

This is a method of sample size selection for multiple regression due to Cohen (1988). It combines elements of classical power analysis and the sample variation method. Cohen does not call it the ``Population Variation Method;" he calls it ``Statistical Power Analysis." For most research psychologists, the population variation method *is* statistical power analysis, period.

The basic idea is this. Looking closely at the formula for the non-centrality parameter ϕ, Cohen decides that it is based on something he interprets as a *population* version of the quantity we are denoting by *a*. That is, one thinks of it as the proportion of remaining variation (Cohen uses the term variance instead of variation) that is explained by the effect in question -- in the population. He calls it ``effect size."

Just a comment: Of course the problem of comparing two means is a special case of multiple regression, but ``effect size" in the population variation method does not reduce to the traditional definition of effect size for the two-sample t-test with equal variances. In fact, effect size in the population variation method mixes the effect together with the design in such a way that they cannot be separated (by the way, this is true of the sample variation method too).

Still, from a so-called ``effect size" and a sample size, it's easy to calculate a non-centrality parameter, and then you can compute power, and increase the sample size until the power is as high as you wish. For most people, most of the time, it's a lot easier to think about proportions of explained variation than to think about collections of non-zero contrasts in units of σ. Plus, it applies to regression models in general, not just factorial ANOVA. To do a classical power analysis with observational data, you need the joint probability distribution of all the observed independent variables (which are presumably independent of any manipulated independent variables). Cohen's method is a lot easier. Here's a program that does it.

```
/**********************  popvar.sas ****************************/
options linesize = 79 pagesize = 200;
data fpower;        /* Replace alpha, s, p, and wantpow below    */
     alpha = 0.05;  /* Significance level                        */
     s = 6;         /* Numerator df = # IVs being tested         */
     p = 26;        /* There are p beta parameters               */
     a = .10  ;     /* Effect size                               */
     wantpow = .80; /* Find n to yield this power.               */
     power = 0; n = p+1; oneminus = 1-alpha; /* Initializing ... */
     do until (power >= wantpow);
        ncp = (n-p)*a/(1-a);
        df2 = n-p;
        power = 1-probf(finv(oneminus,s,df2),s,df2,ncp);
        n = n+1 ;
     end;
     n = n-1;
     put ' ********************************************************';
     put '   ';
     put '   For a multiple regression model with ' p 'betas, ';
     put '   testing ' s 'independent variables using alpha = ' alpha ',';
     put '   a sample size of ' n 'is needed';
     put '   in order to have probability ' wantpow 'of rejecting H0';
     put '   for an effect of size a = ' a ;
     put '   ';
     put ' ********************************************************';

 *******************************************************

   For a multiple regression model with 26 betas,
   testing 6 independent variables using alpha = 0.05 ,
   a sample size of 155 is needed
   in order to have probability 0.8 of rejecting H0
   for an effect of size a = 0.1

 *******************************************************
```

For comparison, when we specified a *sample* proportion of remaining variation equal to 10%, a sample size of 144 was required for significance. Getting into the spirit of the population variation method, we can talk about it like this. If the *population* effect size is 0.10 and n=155, then with 80% probability we'll get a *sample* effect size large enough for significance. How big does the sample effect size have to be? Running `sampvar2.sas`, it turns out that with n=155, you need a sample a=0.092 for significance. So if a=0.10 in the population and n=155, the probability that the sample a exceeds 0.092 is equal to 0.80.

Chapter 8: Multivariate Analysis and Repeated Measures

Multivariate -- More than one dependent variable at once. Why do it? Primarily because if you do parallel analyses on lots of outcome measures, the probability of getting significant results just by chance will definitely exceed the apparent å = 0.05 level. It is also possible in principle to detect results from a multivariate analysis that are not significant at the univariate level.

The simplest way to do multivariate analysis is to do a univariate analysis on each dependent variable separately, and apply a Bonferroni correction. The disadvantage is that testing this way is less powerful than doing it with real multivariate tests.

Another advantage of a true multivariate analysis is that it can "notice" things missed by several Bonferroni-corrected univariate analyses, because ...

Under the surface, a classical multivariate analysis involves the construction of the unique linear combination of the dependent variables that shows the strongest relationship (in the sense explaining the remaining variation) with the independent variables.

The linear combination in question is called the first **canonical variate** or **canonical variable**.

> The number of canonical variables equals the number of dependent variables (or IVs, whichever is fewer).
>
> The canonical variables are all uncorrelated with each other. The second one is constructed so that it has as strong a relationship as possible to the independent variables -- subject to the constraint that it have zero correlation with the first one, and so on.

This why it is not optimal to do a principal components analysis (or factor analysis) on a set of dependent variables, and then treat the components (or factor scores) as dependent variables. Ordinary multivariate analysis is already doing this, and doing it much better.

Assumptions

As in the case of univariate analysis, the statistical assumptions of multivariate analysis concern *conditional distributions* -- conditional upon various configurations of independent variable **X** values. Here we are talking about the conditional *joint* distribution of several dependent variables observed for each case, say Y_1, ..., Y_k. These are often described as a "vector" of observations. It may help to think of the collection of DV values for a case as a point in k-dimensional space, and to imagine an arrow pointing from the origin (0, ...,0) to the point (Y_1, ..., Y_k); the arrow is literally a vector. As I say, this may help. Or it may not.

The classical assumptions of multivariate analysis depend on t
covariance. The population covariance between Y_2 and Y_4 is denot
defined by $\sigma_{2,4} = \rho \sigma_2 \sigma_4$, where

σ_2 is the population standard deviation of Y_2,
σ_4 is the population standard deviation of Y_4, and
ρ is the population correlation between Y_2 and Y_4
(that's the Greek letter rho).

The population covariance can be estimated by the sample covariance, defined in a parallel way by $s_{2,4} = r\, s_2\, s_4$, where s_2 and s_4 are the sample standard deviations and r is the Pearson correlation coefficient.

Whether we are talking about population parameters or sample statistics, it is clear that zero covariance means zero correlation and vice versa.

We will use Σ (the capital Greek letter sigma) to stand for the population **variance-covariance matrix**. This is a k by k rectangular array of numbers with variances on the main diagonal, and covariances on the off-diagonals. For 4 dependent variables it would look like this:

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{1,2} & \sigma_{1,3} & \sigma_{1,4} \\ \sigma_{1,2} & \sigma_2^2 & \sigma_{2,3} & \sigma_{2,4} \\ \sigma_{1,3} & \sigma_{2,3} & \sigma_3^2 & \sigma_{3,4} \\ \sigma_{1,4} & \sigma_{2,4} & \sigma_{3,4} & \sigma_4^2 \end{bmatrix}$$

With this background, the assumptions of classical multivariate analysis are that (conditional on the **X** values)

Sample vectors $\mathbf{Y} = (Y_1, ..., Y_k)$ represent **independent observations** for different cases.

Each conditional distribution is **multivariate normal.**

Each conditional distribution has the **same population variance-covariance matrix** Í.

These assumptions are directly parallel to those of classical univariate regression. Also parallel to univariate analysis is a linear model for each population mean (now we have k of them).

$$\mathbf{E[Y|x]} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_k \end{bmatrix} = \begin{bmatrix} E[Y_1|\mathbf{x}] \\ E[Y_2|\mathbf{x}] \\ \vdots \\ E[Y_k|\mathbf{x}] \end{bmatrix} = \begin{bmatrix} \beta_{0,1} + \beta_{1,1}x_1 + \cdots + \beta_{p-1,1}x_{p-1} \\ \beta_{0,2} + \beta_{1,2}x_1 + \cdots + \beta_{p-1,2}x_{p-1} \\ \vdots \\ \beta_{0,k} + \beta_{1,k}x_1 + \cdots + \beta_{p-1,k}x_{p-1} \end{bmatrix}$$

There are **k different sets of regression coefficients** -- one for each dependent variable.

There is only **one set of independent variables** -- the same for each DV. Dummy variables, interactions etc. are exactly as in univariate regression.

Estimation: The least squares estimates of those doubly-subscripted betas are **exactly what one would get from k separate univariate analyses**. Since the estimated regression coefficients are the same, so are the Yô values and so are the residuals. All methods for univariate residual analysis apply.

Only the tests and confidence intervals (probability statements) are different for univariate and multivariate analysis.

Testing: In univariate analysis, different standard methods for deriving tests (these are hidden from you) all point to Fisher's F test. In multivariate analysis there are four major test statistics, **Wilks' Lambda**, **Pillai's Trace**, **the Hotelling-Lawley Trace**, and **Roy's Greatest Root**.

When there is only one dependent variable, these are all equivalent to F. When there is more than one DV they are all about equally "good" (in any reasonable sense), and conclusions from them generally agree -- but not always. Sometimes one will designate a finding as significant and another will not. In this case you

have borderline results and there is no conventional way out of the dilemma.

The four multivariate test statistics all have F approximations that are used by SAS and other stat packages to compute p-values. Tables are available in textbooks on multivariate analysis. For the first three tests (Wilks' Lambda, Pillai's Trace and the Hotelling-Lawley Trace), the F approximations are very good. For Roy's greatest root the F approximation is lousy. This is a problem with the cheap method for getting p-values, not with the test itself. One can always use tables.

When a multivariate test is significant, many people then follow up with ordinary univariate tests to see "which dependent variable the results came from." This is a reasonable exploratory strategy. More conservative is to follow up with Bonferroni-corrected univariate tests. When you do this, however, there is no guarantee that any of the Bonferroni-corrected tests will be significant.

It is also possible, and in some ways very appealing, to follow up a significant multivariate test with Scheffée tests. For example, Scheffe follow-ups to a significant one-way multivariate ANOVA would include adjusted versions of all the corresponding univariate one-way ANOVAs, all multivariate pairwise comparisons, all univariate pairwise comparisons, and lots of other possibilities — all simultaneously protected at the 0.05 level.

You can also try interpret a significant multivariate effect by looking at the canonical variates, but there is no guarantee they will make sense.

In the following example, cases are hospitals in 4 different regions of the U.S..
The hospitals either have a medical school affiliation or no
variables are average length of time a patient stays at the
risk -- the estimated probability that a patent will acquire
to what he or she caame in with. We will analyze these data
multivariate analysis of variance.

```
/******************** senicmv96a.sas ************************/
options linesize=79;
title 'Senic data: SAS glm & reg multivariate intro';

%include 'senicdef.sas';  /* senicdef.sas reads data, etc.
                             Includes reg1-reg3, ms1 & mr1-mr3 */


/* First a nice two-factor MANOVA on infrisk & stay */

proc glm;
     class region medschl;
     model infrisk stay = region|medschl;
     manova h = _all_;
```

The glm output starts with full univariate output for each I
effect tested) some multivariate output you ignore,

General Linear Models Procedure
Multivariate Analysis of Variance

Characteristic Roots and Vectors of: E Inverse * H, where
H = Type III SS&CP Matrix for REGION E = Error SS&CP Matrix

Characteristic Root	Percent	Characteristic Vector V'EV=1	
		INFRISK	STAY
0.14830859	95.46	-0.00263408	0.06067199
0.00705986	4.54	0.08806967	-0.03251114

Followed by the interesting part.

Manova Test Criteria and F Approximations for
the Hypothesis of no Overall REGION Effect
H = Type III SS&CP Matrix for REGION E = Error SS&CP Matrix

S=2 M=0 N=51

Statistic	Value	F	Num DF	Den DF	Pr > F
Wilks' Lambda	0.86474110	2.6127	6	208	0.0183
Pillai's Trace	0.13616432	2.5570	6	210	0.0207
Hotelling-Lawley Trace	0.15536845	2.6672	6	206	0.0163
Roy's Greatest Root	0.14830859	5.1908	3	105	0.0022

NOTE: F Statistic for Roy's Greatest Root is an upper bound.
NOTE: F Statistic for Wilks' Lambda is exact.

...

Manova Test Criteria and Exact F Statistics for
the Hypothesis of no Overall MEDSCHL Effect
H = Type III SS&CP Matrix for MEDSCHL E = Error SS&CP Matrix

S=1 M=0 N=51

Statistic	Value	F	Num DF	Den DF	Pr > F
Wilks' Lambda	0.92228611	4.3816	2	104	0.0149
Pillai's Trace	0.07771389	4.3816	2	104	0.0149
Hotelling-Lawley Trace	0.08426224	4.3816	2	104	0.0149
Roy's Greatest Root	0.08426224	4.3816	2	104	0.0149

NOTE: F Statistic for Roy's Greatest Root is an upper bound.

...

Manova Test Criteria and F Approximations for
the Hypothesis of no Overall REGION*MEDSCHL Effect
H = Type III SS&CP Matrix for REGION*MEDSCHL E = Error SS&CP Matrix

S=2 M=0 N=51

Statistic	Value	F	Num DF	Den DF	Pr > F
Wilks' Lambda	0.95784589	0.7546	6	208	0.6064
Pillai's Trace	0.04228179	0.7559	6	210	0.6054
Hotelling-Lawley Trace	0.04387599	0.7532	6	206	0.6075
Roy's Greatest Root	0.04059215	1.4207	3	105	0.2409

NOTE: F Statistic for Roy's Greatest Root is an upper bound.
NOTE: F Statistic for Wilks' Lambda is exact.

Remember the output started with the univariate analyses. W
here (out of order) -- just Type III SS, because that's para
tests. We are tracking down the significant multivariate ef
Medical School Affiliation. Using Bonferroni correction mea
$p^{\circ} < 0.025$.

```
Dependent Variable: INFRISK   prob of acquiring infection in hospital

Source                   DF      Type III SS     Mean Square  F Value    Pr > F

REGION                    3       6.61078342      2.20359447     1.35    0.2623
MEDSCHL                   1       6.64999500      6.64999500     4.07    0.0461
REGION*MEDSCHL            3       5.32149160      1.77383053     1.09    0.3581

Dependent Variable: STAY   av length of hospital stay, in days

Source                   DF      Type III SS     Mean Square  F Value    Pr > F

REGION                    3      41.61422755     13.87140918     5.19    0.0022
MEDSCHL                   1      22.49593643     22.49593643     8.41    0.0045
REGION*MEDSCHL            3       0.92295998      0.30765333     0.12    0.9511
```

We conclude that the multivariate effect comes from a univar between the IVs and stay. Question: If this is what we wer end, why do a multivariate analysis at all? Why not just tw with a Bonferroni correction?

The command file senicmv96a.sas continues as follows;

```
/* Now do it with proc reg. Syntax is the same, except list more
   than one dependent variable, and say "mtest" instead of "test." */

proc reg;
     model infrisk stay = reg1-reg3 ms1 mr1-mr3;
     regtest:  mtest reg1=reg2=reg3=0;
     mstest:   mtest ms1=0;
     m_by_r:   mtest mr1=mr2=mr3=0;
```

This gives us exactly the same results we got from proc glm. multivariate analysis of variance is just a special case of you can do it either way. Proc reg can give more control o the cost of setting up your own dummy variables.

Repeated measures

In certain kinds of experimental research, it is common to c
measurements of a variable from the same individual at several different points in time. Usually it is unrealistic to assume tha observations are uncorrelated, and it is very desirable to l correlations into the statistical model.

Sometimes, an individual (in some combination of experimenta measured under essentially the same conditions at several di In that case we will say that time is a **within-subjects factor**or, subject contributes data at more than one value of the IV "t experiences only one value of an IV, it is called a **between s**

.Sometimes, an individual (in some combination of other expe conditions) experiences more than one experimental treatment judging the same stimuli under different background noise le order of presentation of different noise levels would be *cou* time and noise level are unrelated (not confounded). Here n within-subjects factor. The same study can definitely have within-subjects factor and more than one between subjects fa

The meaning of main effects and interactions, as well as the presentation, is the same for within and between subjects fa

We will discuss three methods for analyzing repeated measure that is convenient but not historically chronological they a

1. The multivariate approach.
2. The classical univariate approach.
3. The covariance structure approach.

The multivariate approach to repeated measures

First, note that any of the 3 methods can be multivariate, i dependent variables can be measured at more than one time po with the simpler case in which a single dependent variable i subject on several different occasions.

The basis of the multivariate approach to repeated measures is that **the di measurements conducted on each individual should be considered as multiple dependent variables.**

If there are k dependent variables, regular multivariate ana analysis of up to k linear combinations of those DVs, *instea* dependent variables.

The multivariate approach to repeated measures sets up those to be *meaningful* in terms of representing the repeated measu data.

For example, suppose that men and women in 3 different age g their ability to detect a signal under 5 different levels of There are 10 women and 10 men in each age group for a total presentation of noise levels is randomized for each subject, themselves are tested in random order. This is a three-fact sex are between subjects factors, and noise level is a withi

Let Y_1, Y_2, Y_3, Y_4 and Y_5 be the "Detection Scores" under the levels. Their population means are μ_1, μ_2, μ_3, μ_4 and μ_5 re

Now construct 5 linear combinations of the Y's, as follows.

$W_1 = (Y_1+Y_2+Y_3+Y_4+Y_5\ ) / 5$ $E(W_1) = (\mu_1+\mu_2+\mu_3+\mu_4+\mu_5\ ) / 5$

$W_2 = Y_1 - Y_2$ $E(W_2) = \mu_1 - \mu_2$

$W_3 = Y_2 - Y_3$ $E(W_3) = \mu_2 - \mu_3$

$W_4 = Y_3 - Y_4$ $E(W_4) = \mu_3 - \mu_4$

$W_5 = Y_4 - Y_5$ $E(W_5) = \mu_4 - \mu_5$

All the population means are of course *conditional* on the va independent variables. We will adopt a linear model for ea multivariate setup. In this case the independent variables linear combinations of "'s) are dummy variables for the cate variables sex & age, and the product terms for their intera

Between-subjects effects: The main effects for age and sex, and sex interaction, are just analyses conducted as usual on a s of the DVs, that is, on W_1. This is what we want; we are ju within-subject values.

Within-subject effects: Suppose that (for each configuration of

$$E(W_2) = E(W_3) = E(W_4) = E(W_5) = 0$$

This means $\mu_1 = \mu_2,\ \mu_2 = \mu_3,\ \mu_3 = \mu_4,\ \mu_4 = \mu_5.$

That is, there is no difference among noise level means, i.e the within-subjects factor.

Interactions of between and within-subjects factors are between-subjects effects tested simultaneously on the dependent variables representing differences among within-subject values -- W_2 through

W_5 in this case. For example, a significant sex difference
means that the pattern of differences in mean discrimination
different for males and females. Conceptually, this is exac
interaction.

Similarly, a sex by age interaction on W_2 through W_5 simulta
the pattern of differences in mean discrimination among nois
special combinations of age and sex -- a three-way (age by s
interaction.

Note: There is nothing in this discussion that limits us to
categorical independent variables. Thus, multiple regressio
measures is completely reasonable and presents no special di

Here is noise.dat. Order of vars is
ident, interest, sex, age, noise level, time noise level pre

```
esc> less noise.dat
   1  2.5  1  2  1  4  50.7
   1  2.5  1  2  2  1  27.4
   1  2.5  1  2  3  3  39.1
   1  2.5  1  2  4  2  37.5
   1  2.5  1  2  5  5  35.4
   2  1.9  1  2  1  3  40.3
   2  1.9  1  2  2  1  30.1
   2  1.9  1  2  3  5  38.9
   2  1.9  1  2  4  2  31.9
   2  1.9  1  2  5  4  31.6
   3  1.8  1  3  1  2  39.0
   3  1.8  1  3  2  5  39.1
   3  1.8  1  3  3  4  35.3
   3  1.8  1  3  4  3  34.8
   3  1.8  1  3  5  1  15.4
   4  2.2  0  1  1  2  41.5
   4  2.2  0  1  2  4  42.5
```

```
/*************** noise96a.sas ***********************/
options linesize=79 pagesize=250;
title 'Repeated measures on Noise data:  Multivariate approach';
proc format;      value sexfmt    0 = 'Male'  1 = 'Female' ;
```

```
data loud;
      infile 'noise.dat';  /* Multivariate data read */
     input ident  interest  sex  age  noise1 time1 discrim1
           ident2 inter2    sex2 age2 noise2 time2 discrim2
           ident3 inter3    sex3 age3 noise3 time3 discrim3
           ident4 inter4    sex4 age4 noise4 time4 discrim4
           ident5 inter5    sex5 age5 noise5 time5 discrim5 ;
     format sex sex2-sex5 sexfmt.;
     /* noise1 = 1, ... noise5 = 5. time1 = time noise 1 presented etc.
        ident, interest, sex & age are identical on each line */
     label interest = 'Interest in topic (politics)';

proc glm;
     class age sex;
     model discrim1-discrim5 = age|sex;
     repeated noise profile/ short summary;
```

First we get univariate analyses of discrim1-discrim5 -- not vars yet. Then,

General Linear Models Procedure
Repeated Measures Analysis of Variance
Repeated Measures Level Information

Dependent Variable	DISCRIM1	DISCRIM2	DISCRIM3	DISCRIM4	DISCRIM5
Level of NOISE	1	2	3	4	5

Manova Test Criteria and Exact F Statistics for
the Hypothesis of no NOISE Effect
H = Type III SS&CP Matrix for NOISE E = Error SS&CP Matrix

S=1 M=1 N=24.5

Statistic	Value	F	Num DF	Den DF	Pr > F
Wilks' Lambda	0.45363698	15.3562	4	51	0.0001
Pillai's Trace	0.54636302	15.3562	4	51	0.0001
Hotelling-Lawley Trace	1.20440581	15.3562	4	51	0.0001
Roy's Greatest Root	1.20440581	15.3562	4	51	0.0001

Manova Test Criteria and F Approximations for
the Hypothesis of no NOISE*AGE Effect
H = Type III SS&CP Matrix for NOISE*AGE E = Error SS&CP Matrix

S=2 M=0.5 N=24.5

Statistic	Value	F	Num DF	Den DF	Pr > F
Wilks' Lambda	0.84653930	1.1076	8	102	0.3645
Pillai's Trace	0.15589959	1.0990	8	104	0.3700
Hotelling-Lawley Trace	0.17839904	1.1150	8	100	0.3597
Roy's Greatest Root	0.16044230	2.0857	4	52	0.0960

NOTE: F Statistic for Roy's Greatest Root is an upper bound.
NOTE: F Statistic for Wilks' Lambda is exact.

Manova Test Criteria and Exact F Statistics for
the Hypothesis of no NOISE*SEX Effect
H = Type III SS&CP Matrix for NOISE*SEX E = Error SS&CP Matrix

S=1 M=1 N=24.5

Statistic	Value	F	Num DF	Den DF	Pr > F
Wilks' Lambda	0.93816131	0.8404	4	51	0.5060
Pillai's Trace	0.06183869	0.8404	4	51	0.5060
Hotelling-Lawley Trace	0.06591477	0.8404	4	51	0.5060
Roy's Greatest Root	0.06591477	0.8404	4	51	0.5060

Manova Test Criteria and F Approximations for
the Hypothesis of no NOISE*AGE*SEX Effect
H = Type III SS&CP Matrix for NOISE*AGE*SEX E = Error SS&CP Matrix

S=2 M=0.5 N=24.5

Statistic	Value	F	Num DF	Den DF	Pr > F
Wilks' Lambda	0.84817732	1.0942	8	102	0.3735
Pillai's Trace	0.15679252	1.1058	8	104	0.3654
Hotelling-Lawley Trace	0.17313932	1.0821	8	100	0.3819
Roy's Greatest Root	0.12700316	1.6510	4	52	0.1755

NOTE: F Statistic for Roy's Greatest Root is an upper bound.
NOTE: F Statistic for Wilks' Lambda is exact.

General Linear Models Procedure
Repeated Measures Analysis of Variance
Tests of Hypotheses for Between Subjects Effects

Source	DF	Type III SS	Mean Square	F Value	Pr > F
AGE	2	1751.814067	875.907033	5.35	0.0076
SEX	1	77.419200	77.419200	0.47	0.4946
AGE*SEX	2	121.790600	60.895300	0.37	0.6911
Error	54	8839.288800	163.690533		

Then we are given "Univariate Tests of Hypotheses for Within
We will discuss these later. After that in the lst file, ..

Repeated measures on Noise data: Multivariate approach

General Linear Models Procedure
Repeated Measures Analysis of Variance
Analysis of Variance of Contrast Variables

NOISE.N represents the nth successive difference in NOISE

Contrast Variable: NOISE.1

Source	DF	Type III SS	Mean Square	F Value	Pr > F
MEAN	1	537.00416667	537.00416667	5.40	0.0239
AGE	2	10.92133333	5.46066667	0.05	0.9466
SEX	1	45.93750000	45.93750000	0.46	0.4996
AGE*SEX	2	83.67600000	41.83800000	0.42	0.6587
Error	54	5370.09100000	99.44612963		

Contrast Variable: NOISE.2

Source	DF	Type III SS	Mean Square	F Value	Pr > F
MEAN	1	140.14816667	140.14816667	1.36	0.2489
AGE	2	106.89233333	53.44616667	0.52	0.5985
SEX	1	33.90016667	33.90016667	0.33	0.5688
AGE*SEX	2	159.32233333	79.66116667	0.77	0.4670
Error	54	5569.94700000	103.14716667		

Contrast Variable: NOISE.3

Source	DF	Type III SS	Mean Square	F Value	Pr > F
MEAN	1	50.41666667	50.41666667	0.72	0.4012
AGE	2	56.40633333	28.20316667	0.40	0.6720
SEX	1	195.84266667	195.84266667	2.78	0.1012
AGE*SEX	2	152.63633333	76.31816667	1.08	0.3456
Error	54	3802.61800000	70.41885185		

Contrast Variable: NOISE.4

Source	DF	Type III SS	Mean Square	F Value	Pr > F
MEAN	1	518.61600000	518.61600000	7.77	0.0073
AGE	2	449.45100000	224.72550000	3.37	0.0418
SEX	1	69.55266667	69.55266667	1.04	0.3118
AGE*SEX	2	190.97433333	95.48716667	1.43	0.2479
Error	54	3602.36600000	66.71048148		

The classical univariate approach to repeated measures

The univariate approach to repeated measures is chronologica
can be derived in a clever way from the multivariate tests i
subjects factors. It's what you get at the end of the defau
the analysis of transformed variables, which you have to req

General Linear Models Procedure
Repeated Measures Analysis of Variance
Univariate Tests of Hypotheses for Within Subject Effects

Source: NOISE

DF	Type III SS	Mean Square	F Value	Pr > F	Adj Pr > F G - G	Adj Pr > F H - F
4	2289.31400000	572.32850000	14.12	0.0001	0.0001	0.0001

Source: NOISE*AGE

DF	Type III SS	Mean Square	F Value	Pr > F	Adj Pr > F G - G	Adj Pr > F H - F
8	334.42960000	41.80370000	1.03	0.4134	0.4121	0.4134

(The adj. G - G business will be explained later)

```
Source: NOISE*SEX
                                                                 Adj  Pr > F
     DF      Type III SS      Mean Square   F Value   Pr > F    G - G    H - F
      4     142.42280000      35.60570000      0.88   0.4777   0.4722   0.4777

Source: NOISE*AGE*SEX
                                                                 Adj  Pr > F
     DF      Type III SS      Mean Square   F Value   Pr > F    G - G    H - F
      8     345.66440000      43.20805000      1.07   0.3882   0.3877   0.3882

Source: Error(NOISE)

     DF      Type III SS      Mean Square
    216    8755.83320000      40.53626481

                    Greenhouse-Geisser Epsilon = 0.9356
                           Huynh-Feldt Epsilon = 1.1070
```

To explain the classical univariate approach to repeated mea
about random effects and nested designs.

Nested effects. Suppose a company runs computer training schoo
different cities. One of the cities has 2 schools, the seco
the third city also has 3 schools. In each school, 4 instru
evaluation (students' knowledge is measured somehow).

There are three factors in this study, city, school and inst
is of course only in one city, and let's also say that an in
one school. We say that school is *nested* within city, and i
within school. There is a good dummy variable coding scheme
designs, but we'll skip it. Proc glm uses the syntax

```
model learn = city school(city) instr(school);
```

Designs can have some factors that are nested, and others th
called "crossed"). The patterns can be complex, and the des
useful, very relevant to certain types of research.

Random effects. The models we have been dealing with until now included only fixed effects. In a random effects model, **the independent variable represent a random sample from some population of values.** In the computer school example, if instructor designated for inclusion in the study, instructor would be a comparing Chris to Pat). If they were randomly sampled from instructors (this is a big company), instructor would be a r model that contains both fixed and random effects is called

Significance tests in random and mixed models use F statistic denominator is not always MSE, as it is for purely fixed eff Sometimes it is an interaction term. Choosing the right er models can be complicated job, guided by expected values of (SS/df) terms; these are called *expected mean squares*. Some right error term and certain hypotheses are untestable with Fortunately the whole process can be automated, and SAS does **When the design is unbalanced, usually none of the error terms is useful, and the expected mean squares approach breaks down.**

Random effects, like fixed effects, can either be nested or logic of the design. An interesting case of nested and pure provided by **sub-sampling.** For example, we take a random sampl from each town we select a random sample of households, and household we select a random sample of individuals to test, question.

In such cases the population variance of the DV can truly be pieces -- the variance due to towns, the variance due to hou and the variance due to individuals within households. These variance can be estimated, and they are, by a program called specialized tool for just exactly this design. All effects nested within the preceding one.

Another example: Suppose we are studying waste water treatm
the porosity of "flocks," nasty little pieces of something f
randomly select a sample of flocks, and then cut each one up
We then randomly select a sample of slices (called "sections
look at it under a microscope, and assign a number represent
(how much empty space there is in a designated region of the
independent variables are flock and section. The research q
section is explaining a significant amount of the variance i
if not, we can use just one section per flock, and save cons
expense.

The SAS syntax for this would be

```
proc sort; by flock section; /* Data must be sorted */
proc nested;
     class flock section;
     var por;
```

The F tests on the output are easy to locate. The last colu
of total") is estimated percent of total variance due to the
to R^2, but not the same. To include a covariate (say "windo
var window por; instead of var por;. You'll get an analysis
window as the covariate (which is what you want) and an anal
por as the covariate (which you should ignore).

Anyway, **the classical univariate approach to repeated measures is to treat "subjects" as a random effect that is nested within the between-subjects factors, and which does not interact with any other factors.** Interactions between subjects and various factors may be for
actually these are error terms; they are not tested.

In the noise level example, we could do

```
/*************** noise96b.sas ***********************/
options linesize=79 pagesize=250;
title 'Repeated measures on Noise data:  Univariate approach';
proc format;      value sexfmt    0 = 'Male'  1 = 'Female' ;

data loud;
     infile 'noise.dat';  /* Univariate data read */
     input ident interest sex age noise time discrim ;
     format sex sexfmt.;
     label interest = 'Interest in topic (politics)'
           time     = 'Order of presenting noise level';

proc glm;
     class age sex noise ident;
     model discrim = ident(age*sex) age|sex|noise;
     random ident(age*sex) / test;
```

Notice the univariate data read! We are assuming n = number of observations, not number of cases.

The results are identical to the univariate output produced multivariate approach to repeated measures -- if you know wh

The overall test, and tests associated with Type I & Type II

There are expected mean squares, which you should probably i

There are also repeated warnings that "This test assumes one or other fixed effects are zero." SAS is buying testability of the by assuming that you're only interested in an effect if all interactions involving the effect are absent.

Why do it this way at all? Time-varying covariates.

The univariate approach to repeated measures has some real v

Because n = the number of observations rather than the numbe
possible to have more parameters in a model than cases, or e
measurements than cases. In this situation the multivariate
up.

(Statistical methods should not be a Procrustean bed.)

The univariate approach may assume n is the number of observ
not assume those observations are independent. In fact, the
come from the same subject are assumed to be correlated, as

The "random effect" for subjects is a little piece of rando
of an individual. We think of it as random because the indiv
sampled from a population. If, theoretically, the only reas
measurements from a case are correlated is that each one is
little piece of under-performance or over-performance, the
represents a very good model.

The "random effect for a subject" idea implies a variance-cc
the DVs (say Y_1, ..., Y_4) with a **compound symmetry** structure.

$$\Sigma = \begin{bmatrix} \sigma^2 + \sigma_1 & \sigma_1 & \sigma_1 & \sigma_1 \\ \sigma_1 & \sigma^2 + \sigma_1 & \sigma_1 & \sigma_1 \\ \sigma_1 & \sigma_1 & \sigma^2 + \sigma_1 & \sigma_1 \\ \sigma_1 & \sigma_1 & \sigma_1 & \sigma^2 + \sigma_1 \end{bmatrix}$$

Actually, compound symmetry is sufficient but not necessary
repeated F tests to be valid. All that's necessary is "**sphe**
the covariances of all differences among Y's within a case a

Another virtue of the univariate approach is that it allows
covariates. Standard multivariate analysis has the same X v
dependent variable.

Now some **weak points** of the classical univariate approach:

The model is good if the *only* reason for correlation among t
measures is that one little piece of individuality added to
subject. However, if there are other sources of covariation
measures (like learning, or fatigue, or memory of past perf
much chance rejection of the null hypothesis. In this case
approach, with its unknown variance-covariance matrix, is mo

Even more conservative (overly so, if the assumptions of the
approach are met) is the Greenhouse-Geisser correction, whic
the problem by reducing the error degrees of freedom.

If the design is unbalanced (non-proportional n's), the "F-t
univariate approach do not have an F distribution (even if a
assumptions are satisfied), and it is unclear what they mean

Like the multivariate approach, the univariate approach to r
analysis throws out a case if any of the observations are mi
say "mean substitution?" Oh no!)

The univariate approach has real trouble with unequally spac
with very natural and high quality data sets where (probably
of observations are collected for each individual.

The covariance structure approach to repeated measures.

In the covariance structure approach, the data are set up tc
manner, and one of the variables is a case identification, w
determine which observations of a variable come from the sam
data lines from the same case should be adjacent in the file

Instead of assuming independence or inducing compound symmet
subjects by random effects assumptions, **one directly specifies the**
of the covariance matrix of the observations that come from the same
subject.

The following present no problem at all:

Time-varying covariates (categorical, too)
Unbalanced designs
Unequally spaced observations*
Missing or unequal numbers of observations within su
More variables than subjects (but not more parameter

It's implemented with SAS proc mixed. Only SAS seems to hav

- Lots of different covariance structures are possible
  compound symmetry and unknown.
- A good number of powerful features will not be discu
- Everything's still assumed multivariate normal.

---

* Provided this is unrelated to the variable being repeatedl
the DV is how sick a person is, and the data might be missin
is too sick to be tested, there is a serious problem.

```
/*************** noise96c.sas ***********************/
options linesize=79 pagesize=250;
title 'Repeated measures on Noise data:  Cov Struct Approach';
proc format;      value sexfmt    0 = 'Male'  1 = 'Female' ;

data loud;
     infile 'noise.dat';  /* Univariate data read */
     input ident  interest  sex  age  noise  time  discrim ;
     format sex sexfmt.;
     label interest = 'Interest in topic (politics)'
           time     = 'Order of presenting noise level';

proc mixed method = ml;
     class age sex noise;
     model discrim = age|sex|noise;
     repeated / type = un subject = ident r;
     lsmeans age noise;

proc mixed method = ml;
     class age sex noise;
     model discrim = age|sex|noise;
     repeated / type = cs subject = ident r;
```

Now part of noise95c.lst

The MIXED Procedure

Class Level Information

Class	Levels	Values
AGE	3	1 2 3
SEX	2	Female Male
NOISE	5	1 2 3 4 5

ML Estimation Iteration History

Iteration	Evaluations	Objective	Criterion
0	1	1521.4783527	
1	1	1453.7299937	0.00000000

Convergence criteria met.

R Matrix for Subject 1

Row	COL1	COL2	COL3	COL4	COL5
1	54.07988333	17.08300000	21.38658333	17.91785000	24.27668333
2	17.08300000	69.58763333	15.56748333	29.98861667	21.71448333
3	21.38658333	15.56748333	54.37978333	25.15906667	21.00126667
4	17.91785000	29.98861667	25.15906667	59.31531667	27.58265000
5	24.27668333	21.71448333	21.00126667	27.58265000	55.88941667

Covariance Parameter Estimates (MLE)

Cov Parm		Estimate	Std Error	Z	Pr > |Z|
DIAG	UN(1,1)	54.07988333	9.87359067	5.48	0.0001
	UN(2,1)	17.08300000	8.22102992	2.08	0.0377
	UN(2,2)	69.58763333	12.70490550	5.48	0.0001
	UN(3,1)	21.38658333	7.52577602	2.84	0.0045
	UN(3,2)	15.56748333	8.19197469	1.90	0.0574
	UN(3,3)	54.37978333	9.92834467	5.48	0.0001
	UN(4,1)	17.91785000	7.66900119	2.34	0.0195
	UN(4,2)	29.98861667	9.15325956	3.28	0.0011
	UN(4,3)	25.15906667	8.01928166	3.14	0.0017
	UN(4,4)	59.31531667	10.82944565	5.48	0.0001
	UN(5,1)	24.27668333	7.75870531	3.13	0.0018
	UN(5,2)	21.71448333	8.52518917	2.55	0.0109
	UN(5,3)	21.00126667	7.61610965	2.76	0.0058
	UN(5,4)	27.58265000	8.24206793	3.35	0.0008
	UN(5,5)	55.88941667	10.20396474	5.48	0.0001
Residual		1.00000000	.	.	.

Model Fitting Information for DISCRIM

Description	Value
Observations	300.0000
Variance Estimate	1.0000
Standard Deviation Estimate	1.0000
Log Likelihood	-1002.55
Akaike's Information Criterion	-1017.55
Schwarz's Bayesian Criterion	-1045.32
-2 Log Likelihood	2005.093
Null Model LRT Chi-Square	67.7484
Null Model LRT DF	14.0000
Null Model LRT P-Value	0.0000

Tests of Fixed Effects

Source	NDF	DDF	Type III F	Pr > F
AGE	2	54	5.95	0.0046
SEX	1	54	0.53	0.4716
AGE*SEX	2	54	0.41	0.6635
NOISE	4	216	18.07	0.0001
AGE*NOISE	8	216	1.34	0.2260
SEX*NOISE	4	216	0.99	0.4146
AGE*SEX*NOISE	8	216	1.30	0.2455

From the multivariate approach we had F = 5.35, p < .001 fo
15.36 for noise.

Least Squares Means

Level	LSMEAN	Std Error	DDF	T	Pr > |T|
AGE 1	38.66100000	1.21376060	54	31.85	0.0001
AGE 2	35.24200000	1.21376060	54	29.04	0.0001
AGE 3	32.76700000	1.21376060	54	27.00	0.0001
NOISE 1	39.82166667	0.94938474	216	41.94	0.0001
NOISE 2	36.83000000	1.07693727	216	34.20	0.0001
NOISE 3	35.30166667	0.95201351	216	37.08	0.0001
NOISE 4	34.38500000	0.99427793	216	34.58	0.0001
NOISE 5	31.44500000	0.96513744	216	32.58	0.0001

Now for the second mixed run we get the same kind of beginni
compound symmetry structure,

Tests of Fixed Effects

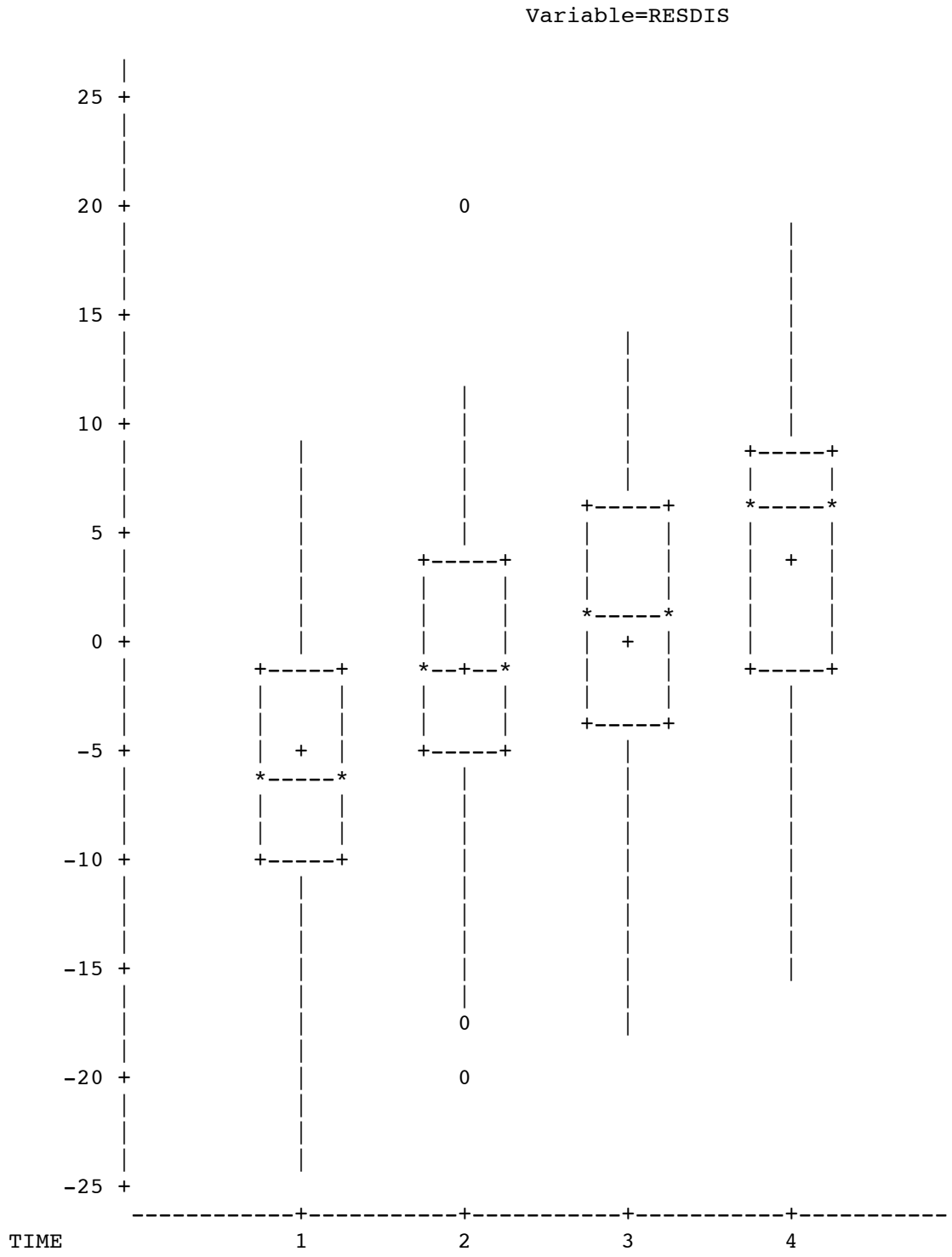
Source	NDF	DDF	Type III F	Pr > F
AGE	2	54	5.95	0.0046
SEX	1	54	0.53	0.4716
AGE*SEX	2	54	0.41	0.6635
NOISE	4	216	15.69	0.0001
AGE*NOISE	8	216	1.15	0.3338
SEX*NOISE	4	216	0.98	0.4215
AGE*SEX*NOISE	8	216	1.18	0.3096

From the univariate approach we had F = 14.12 for noise.

Now proc glm will allow easy examination of residuals no mat
you take to repeated measures, provided the data are read ir

```
/*************** noise96d.sas **********************/
options linesize=79 pagesize=60;
title 'Repeated measures on Noise data: Residuals etc.';
proc format;      value sexfmt    0 = 'Male'  1 = 'Female' ;

data loud;
     infile 'noise.dat';  /* Univariate data read */
     input ident  interest  sex  age  noise  time  discrim ;
     format sex sexfmt.;
     label interest = 'Interest in topic (politics)'
           time     = 'Order of presenting noise level';

proc glm;
     class age sex noise;
     model discrim = age|sex|noise;
     output out=resdata predicted=predis  residual=resdis;

/* Look at some residuals */
proc sort; by time;
proc univariate plot;
    var resdis; by time;
proc plot;
     plot resdis * (ident interest);

/* Include time */
proc mixed method = ml;
     class age sex noise time;
     model discrim = time age|sex|noise;
     repeated / type = un subject = ident r;
     lsmeans time age noise;
```

(Then I generated residuals from this new model using glm, a
Nothing.)



Unfortunately time = 5 wound up on a separate page. .

When time is included the results get stronger but conclusio

Tests of Fixed Effects

Source	NDF	DDF	Type III F	Pr > F
TIME	4	266	17.67	0.0001
AGE	2	266	18.45	0.0001
SEX	1	266	1.63	0.2027
AGE*SEX	2	266	1.28	0.2789
NOISE	4	266	10.95	0.0001
AGE*NOISE	8	266	0.51	0.8488
SEX*NOISE	4	266	0.44	0.7784
AGE*SEX*NOISE	8	266	0.74	0.6573

Least Squares Means

Level	LSMEAN	Std Error	DDF	T	Pr > |T|
TIME 1	29.54468242	0.91811749	266	32.18	0.0001
TIME 2	34.61557451	0.91794760	266	37.71	0.0001
TIME 3	36.18863723	0.92819179	266	38.99	0.0001
TIME 4	39.72344496	0.91838886	266	43.25	0.0001
TIME 5	37.71099421	0.93376736	266	40.39	0.0001
AGE 1	38.66100000	0.68895774	266	56.12	0.0001
AGE 2	35.24200000	0.68895774	266	51.15	0.0001
AGE 3	32.76700000	0.68895774	266	47.56	0.0001
NOISE 1	39.69226830	0.89132757	266	44.53	0.0001
NOISE 2	36.80608879	0.89274775	266	41.23	0.0001
NOISE 3	35.35302821	0.89130480	266	39.66	0.0001
NOISE 4	34.12899017	0.89502919	266	38.13	0.0001
NOISE 5	31.80295787	0.89180628	266	35.66	0.0001

Some good covariance structures are available in proc mixed.

Variance Components: type = vc $\Sigma = \begin{bmatrix} \sigma_1^2 & 0 & 0 & 0 \\ 0 & \sigma_2^2 & 0 & 0 \\ 0 & 0 & \sigma_3^2 & 0 \\ 0 & 0 & 0 & \sigma_4^2 \end{bmatrix}$

Compound Symmetry: type = cs $\Sigma = \begin{bmatrix} \sigma^2+\sigma_1 & \sigma_1 & \sigma_1 & \sigma_1 \\ \sigma_1 & \sigma^2+\sigma_1 & \sigma_1 & \sigma_1 \\ \sigma_1 & \sigma_1 & \sigma^2+\sigma_1 & \sigma_1 \\ \sigma_1 & \sigma_1 & \sigma_1 & \sigma^2+\sigma_1 \end{bmatrix}$

Unknown: type = un $\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{1,2} & \sigma_{1,3} & \sigma_{1,4} \\ \sigma_{1,2} & \sigma_2^2 & \sigma_{2,3} & \sigma_{2,4} \\ \sigma_{1,3} & \sigma_{2,3} & \sigma_3^2 & \sigma_{3,4} \\ \sigma_{1,4} & \sigma_{2,4} & \sigma_{3,4} & \sigma_4^2 \end{bmatrix}$

Banded: type = $\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_5 & 0 & 0 \\ \sigma_5 & \sigma_2^2 & \sigma_6 & 0 \\ 0 & \sigma_6 & \sigma_3^2 & \sigma_7 \\ 0 & 0 & \sigma_7 & \sigma_4^2 \end{bmatrix}$

First order autoregressive: type = sar(1) $\Sigma = \sigma^2 \begin{bmatrix} 1 & \rho & \rho^2 & \rho^3 \\ \rho & 1 & \rho & \rho^2 \\ \rho^2 & \rho & 1 & \rho \\ \rho^3 & \rho^2 & \rho & 1 \end{bmatrix}$

There are more, including Toeplitz, banded Toeplitz & spatia function of Euclidian distance).

Chapter 9

Introduction to S

9.1 History and Terminology

Most major statistical packages are computer programs that have their own control language. The syntax goes with one computer program and just one. The SAS language controls the SAS software, and that's it. Minitab syntax controls Minitab, and that's it. S is a little different. Originally, S was both a program and a language; they were developed together at the former AT&T Bell Labs starting in the late 1970's. Like the unix operating system (also developed around the same time at Bell Labs, among other places), S was open-source and in the public domain. "Open-source" means that the actual program code (initially in Fortran, later in C) was public. It was free to anyone with the expertise to compile and install it.

Later, S was spun off into a private company that is now called Insightful Corporation. They incorporated both the S syntax and the core of the S software into a commercial product called S-Plus. S-Plus is *not* open-source. The "Plus" part of S-Plus is definitely proprietary. S-Plus uses the S language, but the S language is not owned by Insightful Corporation. It's in the public domain.

R also uses the S language. This is a unix joke. You know, like how the unix `less` command is an improved version of `more`. Get it? R is produced by a team of volunteer programmers and statisticians, under the auspices of the Free Software Foundation. It is an official GNU project. What is GNU? GNU stands for "GNU's Not Unix." The recursive nature of this answer is a unix joke. Get it?

The GNU project was started by a group of programmers (led by the great Richard Stallman, author of `emacs`) who believed that software should be open-source and free for anyone to use, copy or modify. They were irritated by the fact that corporations could take unix, enhance it in a few minor (or major) ways, and copyright the result. Solaris, the version of unix used on many Sun workstations, is an example. An even more extreme example is Macintosh OS X, which is just a very elaborate graphical shell running on top of Berkeley Standard Distribution unix.

The GNU operating system was to look and act like unix, but to be rewritten from the ground up, and legally protected in such a way that it could not be incorporated into any piece of software that was proprietary. Anybody would be able to modify it and even sell the modified version – or the original. But any modified version, like the original, would have to be open-source, with no restrictions on copying or use of the software. The main GNU project has been successful; the result is called linux.

R is another successful GNU project. The R development team rewrote the S software from scratch without using any of the original code. It runs under the unix, linux, MS Windows and Macintosh operating systems. It is free, and easy to install. Go to `http://www.R-project.org` to obtain a copy of the software or more information. There are also links on the course home page.

While they were redoing S, the R development team quietly fixed an extremely serious problem. While the S *language* provides a beautiful environment for simulation and customized computer-intensive statistical methods, the S *software* did the job in a terribly inefficient way. The result was that big simulations ran very slowly, and long-running jobs often aborted or crashed the system unless special and very unpleasant measures were taken. S-Plus, because it is based on the original S code, inherits these problems. R is immune to them.

Anyway, S is a language, and R is a piece of software that is controlled by the S language. The discussion that follows will usually refer to S, but all the examples will use the R implementation of S, running in a unix environment. However, R is supposed to be almost entirely the same regardless of hardware and operating system. Mostly, what we do here will also work in S-Plus. Why would you ever want to use S-Plus? Well, it does have some capabilities that R does not have (yet), particularly in the areas of survival analysis and spatial statistics.

9.2 S as a Calculator

To start R, type "R" and return at the unix prompt. Like this:

```
/res/jbrunner/442/S > R

R : Copyright 2001, The R Development Core Team
Version 1.4.0  (2001-12-19)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for a HTML browser interface to help.
Type 'q()' to quit R.

>
```

S is built around *functions*. As you can see above, even asking for help and quitting are functions (with no arguments).

The primary mode of operation of S is line oriented and interactive. It is quite unix-like, with all the good and evil that implies. S gives you a prompt that looks like a "greater than" sign. You type a command, press Return (Enter), and the program does what you say. Its default behaviour is to return the value of what you type, often a numerical value. In the first example, we receive the ">" prompt, type "1+1" and then press the Enter key. S tells us that the answer is 2. Then we obtain $2^3 = 8$.

```
> 1+1
[1] 2
> 2^3 # Two to the power 3
[1] 8
```

What is this `[1]` business? It's clarified when we ask for the numbers from 1 to 30.

```
> 1:30
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
[26] 26 27 28 29 30
```

S will give you an array of numbers in compact form, using a number in brackets to indicate the ordinal position of the first item on each line. When it answered "`1+1`" with `[1] 2`, it was telling us that the first item in this array (of one item) was 2.

S has an amazing variety of mathematical and statistical functions. For example, the gamma function is defined by $\Gamma(a) = \int_0^\infty e^{-t}t^{a-1}\,dt$, and with enough effort you can prove that $\Gamma(\frac{1}{2}) = \sqrt{\pi}$. Note that everything to the left of a `#` is a comment.

```
> gamma(.5)^2       # Gamma(1/2) = Sqrt(Pi)
[1] 3.141593
```

Assignment of values is carried out by a "less than" sign followed *immediately* by a minus sign; it looks like an arrow pointing to the left. The command `x <- 1` would be read "x gets 1."

```
> x <- 1            # Assigns the value 1 to x
> y <- 2
> x+y
[1] 3
> z <- x+y
> z
[1] 3
> x <- c(1,2,3,4,5,6)    #  Collect these numbers; x is now a vector
```

Originally, x was a single number. Now it's a vector (array) of 6 numbers. S operates naturally on vectors.

```
> y <- 1 + 2*x
> cbind(x,y)
     x  y
[1,] 1  3
[2,] 2  5
[3,] 3  7
[4,] 4  9
[5,] 5 11
[6,] 6 13
```

The `cbind` command `bind`s the vectors x and y into `c`olumns. The result is a matrix whose value is returned (displayed on the screen), since it is not assigned to anything.

The bracket (subscript) notation for selecting elements of an array is very powerful. The following is just a simple example.

```
> z <- y[x>4]           # z gets y such that x > 4
> z
[1] 11 13
```

If you put an array of integers inside the brackets, you get those elements, in the order indicated.

```
> y[c(6,5,4,3,2,1)] # y in opposite order
[1] 13 11  9  7  5  3
> y[c(2,2,2,3,4)] # Repeats are okay
[1] 5 5 5 7 9
> y[7] # There is no seventh element.  NA is the missing value code
[1] NA
```

Most operations on arrays are performed element by element. If you take a function of an array, S applies the function to each element of the array and returns an array of function values.

```
> z <- x/y       # Most operations are performed element by element
> cbind(x,y,z)
     x  y         z
[1,] 1  3 0.3333333
[2,] 2  5 0.4000000
[3,] 3  7 0.4285714
[4,] 4  9 0.4444444
[5,] 5 11 0.4545455
[6,] 6 13 0.4615385
> x <- seq(from=0,to=3,by=.1)   # A sequence of numbers
> y <- sqrt(x)
```

S is a great environment for producing high-quality graphics, though we won't use it much for that. Here's just one example. We activate the `pdf` graphics device, so that all subsequent graphics in the session are written to a file that can be viewed with Adobe's *Acrobat Reader*. We then make a line plot of the function $y = \sqrt{x}$, and quit.

```
> pdf("testor.pdf")
> plot(x,y,type='l')      # That's a lower case L
> q()
```

Actually, graphics are a good reason to download and install R on your desktop or laptop computer. By default, you'll see nice graphics output on your screen. Under unix, it's a bit of a pain unless you're in an X-window environment (and we're assuming that you are not). You have to transfer that `pdf` file somewhere and view it with *Acrobat* or *Acrobat Reader*.

Continuing the session, a couple of interesting things happen when we quit. First, we are asked if we want to save the "workspace image." The responses are Yes, No and Cancel (don't quit yet). If you say Yes, R will write a file containing all the objects (x, y and z in the present case) that have been created in the session. Next time you start R, your work will be "restored" to where it was when you quit.

```
Save workspace image? [y/n/c]: y
credit.erin > ls
testor.pdf
```

Notice that when we type `ls`, to list the files, we see only `testor.pdf`, the pdf file containing the plot of $y = \sqrt{x}$. Where is the workspace image? It's an invisible file; type `ls -a` to see *all* the files.

```
credit.erin > ls -a
./              ../             .RData          testor.pdf
```

There it is: `.RData`. Files beginning with a period don't show up in output to the `ls` command unless you use the `-a` option. R puts `.RData` *in the (sub)directory from which R was invoked*. This means that if if you have a separate subdirectory for each project or assignment (not a bad way to organize your work), R will save the workspace from each job in a separate place, so that you can have variables with names like x in more than one place, containing different numbers. When we return to R,

```
credit.erin > R

R : Copyright 2001, The R Development Core Team
Version 1.4.0  (2001-12-19)
```

```
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for a HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

> ls()
[1] "x" "y" "z"

> max(x)
[1] 3
```

All the examples so far (and many of the examples to follow) are interactive, but for serious work, it's better to work with a command file. Put your commands in a file and execute them all at once. Suppose your commands are in a file called `commands.R`. At the S prompt, you'd execute them with `source("commands.R")`. From the unix prompt, you'd do it like this. The `--vanilla` option invokes a "plain vanilla" mode of operation suitable for this situation.

```
credit.erin > R --vanilla < commands.R > homework.out
```

For really big simulations, you may want to run the job in the background at a lower priority. The `&` suffix means run it in the background. `nohup` means don't hang up on me when I log out. `nice` means be nice to other users, and run it at a lower priority.

```
credit.erin > nohup nice R --vanilla < bvnorm.R > bvnorm.out &
```

9.3 S as a Stats Package

Here, we illustrate traditional multiple regression with S, testing the parallel slopes assumption for the metric cars data. Compare `mcars.sas` and `mcars.lst`. There are lots of comment statements that help explain what is going on. More detail will be given in lecture. In addition, the course home page has a link to a nice 100-page manual. If you plan to use R seriously, you should download this manual and read it. But if you come to lecture, you probably don't need to look at it for the purposes of this class.

Here is the "program" named `lesson2.R`.

```
######################################################################
# lesson2.R: execute with    R --vanilla < lesson2.R > lesson2.out #
######################################################################

datalist <-  scan("mcars.dat",list(id=0,country=0,kpl=0,weight=0,length=0))
# datalist is a linked list.
datalist
# There are other ways to read raw data. See help(read.table).
weight <- datalist$weight ; length <- datalist$length ; kpl <- datalist$kpl
country <- datalist$country
cor(cbind(weight,length,kpl))
# The table command gives a bare-bones frequency distribution
table(country)
# That was a matrix. The numbers 1 2 3 are labels.
# You can save it, and you can get at its contents
countrytable <- table(country)
countrytable[2]
# There is an "if" function that you could use to make dummy variables,
# but it's easier to use factor.
countryfac <- factor(country,levels=c(1,2,3),
                     label=c("US","Japanese","European"))
# This makes a FACTOR corresponding to country, like declaring it
# to be categorical. How are dummy variables being set up?
contrasts(countryfac)
# The first level specified is the reference category. You can get a
# different reference category by specifying the levels in a different order.
cntryfac <- factor(country,levels=c(2,1,3),
                     label=c("Japanese","US","European"))
contrasts(cntryfac)
# Test interaction. For comparison, with SAS we got F = 11.5127, p < .0001
# First fit (and save!) the reduced model. lm stands for linear model.
redmod <- lm(kpl ~ weight+cntryfac)
# The object redmod is a linked list, including lots of stuff like all the
# residuals. You don't want to look at the whole thing, at least not now.
summary(redmod)

# Full model is same stuff plus interaction. You COULD specify the whole thing.
fullmod <- update(redmod,. ~ . + weight*cntryfac)
anova(redmod,fullmod)
# The ANOVA summary table is a matrix. You can get at its (i,j)th element.
aovtab <- anova(redmod,fullmod)
aovtab[2,5] # The F statistic
```

```
aovtab[2,6] < .05       #    p < .05 -- True or false?
1>6 # Another example of an expression taking the logical value true or false.
```

Here is the output file `lesson2.out`. Note that it shows the commands. This would not happen if you used `source("lesson2.R")` from within R. I have added some blank lines to the output file to make it more readable.

```
> ######################################################################
> # lesson2.R: execute with    R --vanilla < lesson2.R > lesson2.out #
> ######################################################################
>
> datalist <-  scan("mcars.dat",list(id=0,country=0,kpl=0,weight=0,length=0))
Read 100 records
> # datalist is a linked list.
> datalist
$id
  [1]   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
 [19]  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36
 [37]  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54
 [55]  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72
 [73]  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90
 [91]  91  92  93  94  95  96  97  98  99 100

$country
  [1] 1 2 1 1 1 1 3 1 3 1 2 1 1 3 2 1 1 1 3 2 1 1 1 3 2 1 1 1 1 1 2 1 2 1 1 1 3
 [38] 3 1 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1 1 1 3 1 1 1 2 1 1 1 1 2 2 1 1 1 2 1 1
 [75] 1 2 2 3 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 3 3 3 1 1 1

$kpl
  [1]  5.04 10.08  9.24  7.98  7.98  7.98  9.66  7.56  5.88 10.92 12.60  8.40
 [13]  8.82 10.92  7.56 12.18  5.04  5.88  7.14 13.02  5.88 10.92  6.72 10.50
 [25]  8.82  5.88  6.72 11.76  9.24  7.56  7.56 11.76 10.50  5.88  9.24  7.98
 [37]  7.14 17.22  6.72  7.98  7.14  6.30  5.88  8.82  9.24  9.24  5.88  8.40
 [49] 10.50  9.24  7.56  7.56 12.60 12.60  7.98  7.56  8.40  9.66  7.56  6.30
 [61]  5.88  7.56 10.08  5.04  8.82 11.76 14.70 10.08  9.24 10.92 10.50  7.56
 [73]  8.82  7.56  7.14  7.56 10.08  8.82  5.88  8.82  8.82 10.08 17.22  6.72
 [85]  9.24  5.88  7.56 11.76  7.98  8.82  5.88  5.88  7.14  5.04 17.22 17.22
 [97]  7.14 10.50  6.72  7.56

$weight
  [1] 2178.0 1026.0 1188.0 1444.5 1485.0 1485.0  972.0 1665.0 1539.0 1003.5
 [11]  891.0 1273.5 1930.5  823.5 1084.5  949.5 2178.0 1755.0 1426.5  990.0
 [21] 1827.0 1134.0 1813.5 1192.5 1237.5 1858.5 1813.5 1062.0 1431.0 1651.5
 [31] 1201.5 1062.0 1008.0 1858.5 1318.5 1440.0 1273.5  918.0 1813.5 1530.0
 [41] 1683.0 1836.0 1723.5 1827.0 1449.0 1318.5 1858.5 1273.5  868.5 1318.5
 [51] 1665.0 1620.0  954.0  954.0 1516.5 1665.0 1462.5  972.0 1665.0 1674.0
 [61] 1755.0 1201.5 1237.5 2178.0 1930.5 1062.0  922.5 1026.0 1449.0 1134.0
 [71]  990.0 1084.5 1930.5 1516.5 1507.5 1084.5 1026.0  958.5 1858.5 1930.5
 [81] 1192.5 1237.5  918.0 1813.5 1449.0 1755.0 1561.5 1062.0 1489.5 1192.5
 [91] 1827.0 1755.0 1683.0 2178.0  918.0  918.0 1426.5  990.0 1660.5 1498.5

$length
  [1] 591.82 431.80 426.72 510.54 502.92 502.92 436.88 543.56 487.68 431.80
 [11] 391.16 495.30 518.16 360.68 441.96 414.02 591.82 518.16 490.22 419.10
 [21] 561.34 462.28 523.24 449.58 467.36 551.18 523.24 431.80 490.22 553.72
```

```
 [31] 444.50 431.80 436.88 551.18 472.44 505.46 480.06 393.70 523.24 508.00
 [41] 558.80 563.88 510.54 558.80 508.00 472.44 551.18 495.30 393.70 472.44
 [51] 543.56 523.24 414.02 414.02 508.00 543.56 497.84 436.88 543.56 538.48
 [61] 518.16 444.50 454.66 591.82 518.16 431.80 416.56 431.80 508.00 462.28
 [71] 419.10 441.96 518.16 502.92 439.42 441.96 431.80 408.94 551.18 518.16
 [81] 454.66 454.66 393.70 523.24 508.00 518.16 502.92 431.80 502.92 454.66
 [91] 561.34 518.16 558.80 591.82 393.70 393.70 490.22 419.10 538.48 510.54

> # There are other ways to read raw data. See help(read.table).

> weight <- datalist$weight ; length <- datalist$length ; kpl <- datalist$kpl
> country <- datalist$country
> cor(cbind(weight,length,kpl))
           weight     length        kpl
weight  1.0000000  0.9462018 -0.7704194
length  0.9462018  1.0000000 -0.7899859
kpl    -0.7704194 -0.7899859  1.0000000

> # The table command gives a bare-bones frequency distribution
> table(country)
country
 1  2  3
73 13 14

> # That was a matrix. The numbers 1 2 3 are labels.
> # You can save it, and you can get at its contents
> countrytable <- table(country)
> countrytable[2]
 2
13

> # There is an "if" function that you could use to make dummy variables,
> # but it's easier to use factor.
> countryfac <- factor(country,levels=c(1,2,3),
+                      label=c("US","Japanese","European"))
> # This makes a FACTOR corresponding to country, like declaring it
> # to be categorical. How are dummy variables being set up?

> contrasts(countryfac)
         Japanese European
US              0        0
Japanese        1        0
European        0        1

> # The first level specified is the reference category. You can get a
> # different reference category by specifying the levels in a different order.
> cntryfac <- factor(country,levels=c(2,1,3),
+                      label=c("Japanese","US","European"))

> contrasts(cntryfac)
         US European
Japanese  0        0
US        1        0
European  0        1
```

```
> # Test interaction. For comparison, with SAS we got F = 11.5127, p < .0001

> # First fit (and save!) the reduced model. lm stands for linear model.
> redmod <- lm(kpl ~ weight+cntryfac)

> # The object redmod is a linked list, including lots of stuff like all the
> # residuals. You don't want to look at the whole thing, at least not now.

> summary(redmod)

Call:
lm(formula = kpl ~ weight + cntryfac)

Residuals:
    Min      1Q  Median      3Q     Max
-3.0759 -0.9810 -0.1919  0.4725  5.0795

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      16.2263357  0.7631228  21.263   <2e-16 ***
weight           -0.0060407  0.0005708 -10.583   <2e-16 ***
cntryfacUS        1.2361472  0.5741299   2.153   0.0338 *
cntryfacEuropean  1.4595914  0.6456563   2.261   0.0260 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.676 on 96 degrees of freedom
Multiple R-Squared: 0.618,      Adjusted R-squared: 0.606
F-statistic: 51.76 on 3 and 96 DF,  p-value:     0

>
> # Full model is same stuff plus interaction. You COULD specify the whole thing.

> fullmod <- update(redmod,. ~ . + weight*cntryfac)

> anova(redmod,fullmod)

Analysis of Variance Table

Model 1: kpl ~ weight + cntryfac
Model 2: kpl ~ weight + cntryfac + weight:cntryfac
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1     96 269.678
2     94 216.617  2    53.061 11.513 3.372e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> # The ANOVA summary table is a matrix. You can get at its (i,j)th element.
> aovtab <- anova(redmod,fullmod)
> aovtab[2,5] # The F statistic
[1] 11.51273

> aovtab[2,6] < .05       #    p < .05 -- True or false?
[1] TRUE

> 1>6 # Another example of an expression taking the logical value true or false.
[1] FALSE
```

```
$
```

9.4 Random Numbers and Simulation

S is a superb environment for simulation and customized computer-intensive statistical methods. That's really why it is being discussed. Simulation is an extremely general and powerful method for calculating probabilities that are difficult to figure out by other means. Well, technically it's a way of *estimating* those probabilities, based a sample of random numbers. Before proceeding, we need a couple of definitions.

We will use the term **statistical experiment** to refer to any procedure whose outcome is not known in advance with certainty. The most standard, and the most boring example of a statistical experiment is to toss a coin and observe whether it comes up heads or tails. We *model* statistical experiments by pretending that they obey the laws of probability.

When we carry out a statistical experiment, the things that can happen (the things we pay attention to) are called **outcomes**. Sets of outcomes are called **events**. For example, if you roll a die, the outcomes are the numbers 1 through 6, and "even" is an event consisting of the outcomes $\{2, 4, 6\}$.

The main principle we will use is called the **Law of Large Numbers**. There are quite a few versions of this law. Here's a verbal statement of the one we will use. *If a statistical experiment is carried out independently a very large number of times (trials) under identical conditions, the proportion of times an event occurs approaches the probability of the event, as the number of trials increases.* In elementary texts, this is sometimes used as the definition of probability. But in more sophisticated treatments, it's a theorem.

For example, suppose you are planning to test differences between means for an experimental versus a control group, and you have strong reason to believe that your data will have a chi-square distribution within groups. You are going to log-transform the data to take care of the positive skewness of the chi-square, and then use a common t-test.

Suppose data in the experimental group is chi-square with one degree of freedom (so the population mean is one and the variance is two), and the data in the control group is chi-square with two degree of freedom (so the population mean is two and the variance is four). What is the power of the t-test on the transformed data with $n = 20$ in each group?

Nobody can figure this out mathematically, but it's pretty easy with simulation. Here's how to do it.

1. Using the random number generator in some software package, generate 20 independent chi-square values with one degree of freedom, and 20 independent chi-square values with two degrees of freedom.

2. Log transform all the values.

3. Compute the t-test.

4. Check to see if $p < 0.05$.

Do this a large number of times. The proportion of times $p < 0.05$ is the power — or more precisely, a Monte Carlo estimate of the power.

The number of times a statistical experiment is repeated is called the **Monte Carlo sample size**. How big should the Monte Carlo sample size be? It depends on how much precision you need. We will produce confidence intervals for all our Monte Carlo estimates, to get a handle on the probable margin of error of the statements we make. Sometimes, Monte Carlo sample size can be chosen by a power analysis. More details will be given later.

The example below shows several simulations of taking a random sample of size 20 from a standard normal population ($\mu = 0$, $\sigma^2 = 1$). Now actually, computer-generated random numbers are not *really* random; they are completely determined by the execution of the computer program that generates them. The most common (and best) random number generators actually produce a *stream* of pseudo-random numbers that will eventually repeat. In the good ones (and R uses a good one), "eventually" means after the end of the universe. So the pseudo-random numbers that R produces really *act* random, even though they are not. It's safe to say that they come closer to satisfying the assumptions of significance tests than any real data.

If you don't instruct it otherwise, R will use the system clock to decide on *where* in the random number stream it should begin. But sometimes you want to be able to reproduce the results of a simulation exactly, say if you're debugging your program, or you have already spent a lot of time making a graph based on it. In this case you can control the starting place in the random number stream, by setting the "seed" of the random number generator. The seed is a big integer; I used 12345 just as an example.

```
> rnorm(20) # 20 standard normals
 [1]  0.24570675 -0.38857202  0.47642336  0.75657595  0.71355871 -0.74630629
 [7] -0.02485569  1.93346357  0.15663167  1.16734485  0.57486449  1.32309413
[13]  0.63712982  2.00473940  0.04221730  0.70896768  0.42128470 -0.12115292
[19]  1.42043470 -1.04957255

> set.seed(12345) # Be able to reproduce the stream of pseudo-random numbers.
> rnorm(20)
 [1]  0.77795979 -0.89072813  0.05552657  0.67813726  0.80453336 -0.35613672
 [7] -1.24182991 -1.05995791 -2.67914037 -0.01247257 -1.22422266  0.88672878
[13] -1.32824804 -2.73543539  0.40487757  0.41793236 -1.47520817  1.15351981
[19] -1.24888614  1.11605686
> rnorm(20)
 [1]  0.866507371  2.369884323  0.393094088 -0.970983967 -0.292948278
 [6]  0.867358962  0.495983546  0.331635970  0.702292771  2.514734599
[11]  0.522917841 -0.194668990 -0.089222053 -0.491125596 -0.452112445
[16] -0.515548826 -0.244409517 -0.008373764 -1.459415684 -1.433710170

> set.seed(12345)
> rnorm(20)
 [1]  0.77795979 -0.89072813  0.05552657  0.67813726  0.80453336 -0.35613672
 [7] -1.24182991 -1.05995791 -2.67914037 -0.01247257 -1.22422266  0.88672878
[13] -1.32824804 -2.73543539  0.40487757  0.41793236 -1.47520817  1.15351981
[19] -1.24888614  1.11605686
```

The `rnorm` function is probably the most important random number generator, because it is used so often to investigate the properties of statistical tests that assume a normal distribution. Here is some more detail about `rnorm`.

```
> help(rnorm)
Normal                 package:base                 R Documentation

The Normal Distribution

Description:

     Density, distribution function, quantile function and random
     generation for the normal distribution with mean equal to 'mean'
     and standard deviation equal to 'sd'.

Usage:

     dnorm(x, mean=0, sd=1, log = FALSE)
     pnorm(q, mean=0, sd=1, lower.tail = TRUE, log.p = FALSE)
     qnorm(p, mean=0, sd=1, lower.tail = TRUE, log.p = FALSE)
     rnorm(n, mean=0, sd=1)
```

```
Arguments:

     x,q: vector of quantiles.

       p: vector of probabilities.

       n: number of observations. If 'length(n) > 1', the length is
          taken to be the number required.

    mean: vector of means.

      sd: vector of standard deviations.

log, log.p: logical; if TRUE, probabilities p are given as log(p).

lower.tail: logical; if TRUE (default), probabilities are P[X <= x],
          otherwise, P[X > x].

Details:

     If 'mean' or 'sd' are not specified they assume the default values
     of '0' and '1', respectively.

     The normal distribution has density

       f(x) = 1/(sqrt(2 pi) sigma) e^-((x - mu)^2/(2 sigma^2))

     where mu is the mean of the distribution and sigma the standard
     deviation.

     'qnorm' is based on Wichura's algorithm AS 241 which provides
     precise results up to about 16 digits.

Value:

     'dnorm' gives the density, 'pnorm' gives the distribution
     function, 'qnorm' gives the quantile function, and 'rnorm'
     generates random deviates.

References:

     Wichura, M. J. (1988) Algorithm AS 241: The Percentage Points of
     the Normal Distribution. Applied Statistics, 37, 477-484.

See Also:

     'runif' and '.Random.seed' about random number generation, and
     'dlnorm' for the Lognormal distribution.

Examples:

     dnorm(0) == 1/ sqrt(2*pi)
     dnorm(1) == exp(-1/2)/ sqrt(2*pi)
     dnorm(1) == 1/ sqrt(2*pi*exp(1))

     ## Using "log = TRUE" for an extended range :
     par(mfrow=c(2,1))
```

```
plot(function(x)dnorm(x, log=TRUE), -60, 50, main = "log { Normal density }")
curve(log(dnorm(x)), add=TRUE, col="red",lwd=2)
mtext("dnorm(x, log=TRUE)", adj=0); mtext("log(dnorm(x))", col="red", adj=1)

plot(function(x)pnorm(x, log=TRUE), -50, 10, main = "log { Normal Cumulative }")
curve(log(pnorm(x)), add=TRUE, col="red",lwd=2)
mtext("pnorm(x, log=TRUE)", adj=0); mtext("log(pnorm(x))", col="red", adj=1)
```

After generating normal random numbers, the next most likely thing you might want to do is randomly scramble some existing data values. The `sample` function will select the elements of some array, either with replacement or without replacement. If you select all the numbers in a set without replacement, you've rearranged them in a random order. This is the basis of randomization tests. Sampling *with* replacement is the basis of the bootstrap.

```
> help(sample)
sample                 package:base                 R Documentation

Random Samples and Permutations

Description:

     'sample' takes a sample of the specified size from the elements of
     'x' using either with or without replacement.

Usage:

     sample(x, size, replace = FALSE, prob = NULL)

Arguments:

       x: Either a (numeric, complex, character or logical) vector of
          more than one element from which to choose, or a positive
          integer.

    size: A positive integer giving the number of items to choose.

 replace: Should sampling be with replacement?

    prob: A vector of probability weights for obtaining the elements of
          the vector being sampled.

Details:

     If 'x' has length 1, sampling takes place from '1:x'.

     By default 'size' is equal to 'length(x)' so that 'sample(x)'
     generates a random permutation of the elements of 'x' (or '1:x').

     The optional 'prob' argument can be used to give a vector of
     weights for obtaining the elements of the vector being sampled.
     They need not sum to one, but they should be nonnegative and not
     all zero.  If 'replace' is false, these probabilities are applied
```

```
     sequentially, that is the probability of choosing the next item is
     proportional to the probabilities amongst the remaining items. The
     number of nonzero weights must be at least 'size' in this case.

Examples:

     x <- 1:12
     # a random permutation
     sample(x)
     # bootstrap sampling
     sample(x,replace=TRUE)

     # 100 Bernoulli trials
     sample(c(0,1), 100, replace = TRUE)
```

9.4.1 Illustrating the Regression Artifact by Simulation

In the ordinary use of the English language, to "regress" means to go backward. In Psychiatry and Abnormal Psychology, the term "regression" is used when a person's behaviour changes to become more typical of an earlier stage of development — like when an older child starts wetting the bed, or an adult under extreme stress sucks his thumb. Isn't this a strange word to use for the fitting of hyperplanes by least-squares?

The term "regression" (as it is used in Statistics) was coined by Sir Francis Galton (1822-1911). For reasons that now seem to have a lot to do with class privilege and White racism, he was very interested in heredity. Galton was investigating the relationship between the heights of fathers and the heights of sons. What about the mothers? Apparently they had no height.

Anyway, Galton noticed that very tall fathers tended to have sons that were a bit shorter than they were, though still taller than average. On the other hand, very short fathers tended to have sons that were taller than they were, though still shorter than average. Galton was quite alarmed by this "regression toward mediocrity" or "regression toward the mean," particularly when he found it in a variety of species, for a variety of physical characteristics. See Galton's "Regression towards mediocrity in hereditary stature", *Journal of the Anthropological Institute* **15** (1886), 246-263. It even happens when you give a standardized test twice to the same people. The people who did the very best the first time tend to do a little worse the second time, and the people who did the very worst the first time tend to do a little better the second time.

Galton thought he had discovered a Law of Nature, though in fact the

whole thing follows from the algebra of least squares. Here's a verbal alternative. Height is influenced by a variety of chance factors, many of which are *not* entirely shared by fathers and sons. These include the mother's height, environment and diet, and the vagaries of genetic recombination. You could say that the tallest fathers included some who "got lucky," if you think it's good to be tall (Galton did, of course). The sons of the tall fathers had some a genetic predisposition to be tall, but on average, they didn't get as lucky as their fathers in every respect. A similar argument applies to the short fathers and their sons.

This is the basis for the so-called **regression artifact**. Pre-post designs with extreme groups are doomed to be misleading. Programs for the disadvantaged "work" and programs for the gifted "hurt." This is a very serious methodological trap that has doomed quite a few evaluations of social programs, drug treatments – you name it.

Is this convincing? Well, the argument above may be enough for some people. But perhaps if it's illustrated by simulation, you'll be even more convinced. Let's find out.

Suppose an IQ test is administered to the same 10,000 students on two occasions. Call the scores `pre` and `post`. After the first test, the 100 individuals who did worst are selected for a special remedial program, but it does *nothing*. And, the 100 individuals who did best on the pre-test get a special program for the gifted, but it does *nothing*. We do a matched t-test on the students who got the remedial program, and a matched $t - test$ on the students who got the gifted program.

What should happen? If you followed the stuff about regression artifacts, you'd expect significant improvement from the students who got the remedial program, and significant deterioration from the students who got the gifted program – even though in fact, both programs are completely ineffective (and harmless). How will we simulate this?

According to classical psychometric theory, a test score is the sum of two independent pieces, the *True Score* and *measurement error*. If you measure an individual twice, she has the same True Score, but the measurement error component is different.

True Score and measurement error have population variances. Because they are independent, the variance of the observed score is the sum of the true score variance and the error variance. The proportion of the observed score variance that is True Score variance is called the test's *reliability*. Most "intelligence" tests have a mean of 100, a standard deviation of 15, and a

reliability around 0.80.

So here's what we do. Making everything normally distributed and selecting parameter values so the means, standard deviations and reliability come out right, we

- Simulate 10,000 true scores.
- Simulate 10,000 measurement errors for the pre-test and an independent 10,000 measurement errors for the post-test.
- Calculate 10,000 pre-test scores by `pre = True + error1`.
- Calculate 10,000 post-test scores by `pre = True + error2`.
- Do matched t-tests on the individuals with the 100 worst and the 100 best pre-test scores.

This procedure is carried out *once* by the program `regart.R`. In addition, `regart.R` carries out a matched t-test on the entire set of 10,000 pairs, just to verify that there is no systematic change in "IQ" scores.

```
# regart.R    Demonstrate Regression Artifact
###################### Setup ######################
N <- 10000 ; n <- 100
truevar <- 180 ; errvar <- 45
truesd <- sqrt(truevar) ; errsd <- sqrt(errvar)
# set.seed(44444)
# Now define the function ttest, which does a matched t-test

ttest <- function(d) # Matched t-test. It operates on differences.
    {
    ttest <- numeric(4)
    names(ttest) <- c("Mean Difference"," t "," df "," p-value ")
    ave <- mean(d) ; nn <- length(d) ; sd <- sqrt(var(d)) ; df <- nn-1
    tstat <- ave*sqrt(nn)/sd
    pval <- 2*(1-pt(abs(tstat),df))
    ttest[1] <- ave ; ttest[2] <- tstat; ttest[3] <- df; ttest[4] <- pval
    ttest # Return the value of the function
    }
```

```
#####################################################


error1 <- rnorm(N,0,errsd) ; error2 <- rnorm(N,0,errsd)
truescor <- rnorm(N,100,truesd)
pre <- truescor+error1 ; rankpre <- rank(pre)

# Based on their rank on the pre-test, we take the n worst students and
# place them in a special remedial program, but it does NOTHING.

# Based on their rank on the pre-test, we take the n best students and
# place them in a special program for the gifted, but it does NOTHING.

post <- truescor+error2
diff <- post-pre # Diff represents "improvement."
                 # But of course diff = error2-error1 = noise

cat("\n") # Skip a line
cat("---------------------------------- \n")
dtest <- ttest(diff)
cat("Test on diff (all scores) \n") ; print(dtest) ; cat("\n")

remedial <- diff[rankpre<=n] ; rtest <- ttest(remedial)
cat("Test on Remedial \n") ; print(rtest) ; cat("\n")

gifted <- diff[rankpre>=(N-n+1)] ; gtest <- ttest(gifted)
cat("Test on Gifted \n") ; print(gtest) ; cat("\n")
cat("---------------------------------- \n")
```

The `ttest` function is a little unusual because it takes a whole vector of numbers (length unspecified) as input, and returns an array of 4 values. Often, functions take one or more numbers as input, and return a single value. We will see some more examples shortly. At the R prompt,

```
> source("regart.R")

-----------------------------------
Test on diff (all scores)
Mean Difference              t              df         p-value
   1.872566e-02    1.974640e-01    9.999000e+03    8.434685e-01

Test on Remedial
Mean Difference              t              df         p-value
   7.192531e+00    8.102121e+00    9.900000e+01    1.449729e-12

Test on Gifted
Mean Difference              t              df         p-value
  -8.311569e+00   -9.259885e+00    9.900000e+01    4.440892e-15

-----------------------------------
> source("regart.R")

-----------------------------------
Test on diff (all scores)
Mean Difference              t              df         p-value
   2.523976e-02    2.659898e-01    9.999000e+03    7.902525e-01

Test on Remedial
Mean Difference              t              df         p-value
   5.510484e+00    5.891802e+00    9.900000e+01    5.280147e-08

Test on Gifted
Mean Difference              t              df         p-value
      -8.972938      -10.783356       99.000000        0.000000

-----------------------------------
> source("regart.R")

-----------------------------------
Test on diff (all scores)
Mean Difference              t              df         p-value
```

```
      0.0669827       0.7057641    9999.0000000       0.4803513

Test on Remedial
Mean Difference              t              df         p-value
   8.434609e+00    9.036847e+00    9.900000e+01    1.376677e-14

Test on Gifted
Mean Difference              t              df         p-value
      -8.371483      -10.215295       99.000000       0.000000

-----------------------------------
```

The preceding simulation was unusual in that the phenomenon it illustrates happens virtually every time. In the next example, we need to use the Law of Large Numbers.

9.4.2 An Example of Power Analysis by Simulation

Suppose we want to test the effect of some experimental treatment on mean response, comparing an experimental group to a control. We are willing to assume normality, but *not* equal variances. We're ready to use an unequal-variances t-test, and we want to do a power analysis.

Unfortunately it's safe to say that nobody knows the exact non-central distribution of this monster. In fact, even the central distribution isn't exact; it's just a very good approximation. So, we have to resort to first principles. There are four parameters: $\theta = (\mu_1, \mu_2, \sigma_1^2, \sigma_2^2)$. For a given set of parameter values, we will simulate samples of size n_1 and n_2 from normal distributions, do the significance test, and see if it's significant. We'll do it over and over. By the Law of Large Numbers, the proportion of times the test is significant will approach the power as the Monte Carlo sample size (the number of data sets we simulate) increases.

The number we get, of course, will just be an *estimate* of the power. How accurate is the estimate? As promised earlier, we'll accompany every Monte Carlo estimate of a probability with a confidence interval. Here's the formula. For the record, it's based on the normal approximation to the binomial, not bothering with a continuity correction.

$$\widehat{P} \pm z_{1-\frac{\alpha}{2}} \sqrt{\frac{\widehat{P}(1-\widehat{P})}{m}} \tag{9.1}$$

This formula will be implemented in the S function `merror` for "margin of error."

```
merror <- function(phat,m,alpha) # (1-alpha)*100% merror for a proportion
    {
    z <- qnorm(1-alpha/2)
    merror <- z * sqrt(phat*(1-phat)/m)  # m is (Monte Carlo) sample size
    merror
    }
```

The Monte Carlo estimate of the probability is denoted by $\widehat{P}$, the quantity m is the Monte Carlo sample size, and $z_{1-\alpha/2}$ is the value with area $1-\frac{\alpha}{2}$ to the left of it, under the standard normal curve. Typically, we will choose $\alpha = 0.01$ to get a 99% confidence interval, so $z_{1-\alpha/2} = 2.575829$.

How should we choose m? In other words, how many data sets should we simulate? It depends on how much accuracy we want.Since our policy is to accompany Monte Carlo estimates with confidence intervals, we will choose the Monte Carlo sample size to control the width of the confidence interval.

According to Equation (9.1), the confidence interval is an estimated probability, plus or minus a margin of error. The margin of error is $z_{1-\frac{\alpha}{2}}\sqrt{\frac{\widehat{P}(1-\widehat{P})}{m}}$, which may be viewed as an *estimate* of $z_{1-\frac{\alpha}{2}}\sqrt{\frac{P(1-P)}{m}}$. So, for any given probability we are trying to estimate, we can set the desired margin of error to some small value, and solve for m. Denoting the *criterion* margin of error by c, the general solution is

$$m = \frac{z^2_{1-\frac{\alpha}{2}}}{c^2} P(1-P), \tag{9.2}$$

which is implemented in the S function `mmargin`.

```
mmargin <- function(p,cc,alpha)
           # Choose m to get (1-alpha)*100% margin of error equal to cc
           {
           mmargin <- p*(1-p)*qnorm(1-alpha/2)^2/cc^2
           mmargin <- trunc(mmargin+1) # Round up to next integer
           mmargin
           } # End definition of function mmargin
```

Suppose we want a 99% confidence interval around a power of 0.80 to be accurate to plus or minus 0.01.

```
> mmargin(.8,.01,.01)
[1] 10616
```

The table below shows Monte Carlo sample sizes for estimating power with a 99% confidence interval.

Table 6.1: Monte Carlo Sample Size Required to Estimate Power with a Specified 99% Margin of Error

	Power Being Estimated					
Margin of Error	0.70	0.75	0.80	0.85	0.90	0.99
0.10	140	125	107	85	60	7
0.05	558	498	425	339	239	27
0.01	13,934	12,441	10,616	8,460	5,972	657
0.005	55,734	49,762	42,464	33,838	23,886	2,628
0.001	1,393,329	1,244,044	1,061,584	845,950	59,7141	65,686

It's somewhat informative to see how the rows of the table were obtained.

```
> wpow <- c(.7,.75,.8,.85,.9,.99)
> mmargin(wpow,.1,.01)
[1] 140 125 107  85  60   7
> mmargin(wpow,.05,.01)
[1] 558 498 425 339 239  27
> mmargin(wpow,.01,.01)
[1] 13934 12441 10616  8460  5972   657
> mmargin(wpow,.005,.01)
[1] 55734 49762 42464 33838 23886  2628
> mmargin(wpow,.001,.01)
[1] 1393329 1244044 1061584  845950  597141   65686
```

Equations (9.1) and (9.2) are general; they apply to the Monte Carlo estimation of *any* probability, and Table 6.1 applies to any Monte Carlo estimation of power. Let's return to the specific example at hand. Suppose we the population standard deviation of the Control Group is 2 and the standard deviation of the Experimental Group is 6. We'll let the population means be $\mu_1 = 1$ and $\mu_2 = 3$, so that the difference between population means is half the *average* within-group population standard deviation.

To select a good starting value of n, let's pretend that the standard deviations are equal to the average value, and we are planning an ordinary

two-sample t-test. Referring to formula (4.4) for the non-centrality parameter of the non-central F-distribution, we'll let $q = \frac{1}{2}$; this is optimal when the variances are equal. Since $\delta = \frac{1}{2}$, we have $\phi = n\,q(1-q)\,\delta^2 = \frac{n}{16}$. Here's some S. It's short — and sweet. Well, maybe it's an acquired taste. It's also true that I know this problem pretty well, so I knew a good range of n values to try.

```
> n <- 125:135
> pow <- 1-pf(qf(.95,1,(n-2)),1,(n-2),(n/16))
> cbind(n,pow)
        n       pow
 [1,] 125 0.7919594
 [2,] 126 0.7951683
 [3,] 127 0.7983349
 [4,] 128 0.8014596
 [5,] 129 0.8045426
 [6,] 130 0.8075844
 [7,] 131 0.8105855
 [8,] 132 0.8135460
 [9,] 133 0.8164666
[10,] 134 0.8193475
[11,] 135 0.8221892
```

We will start the unequal variance search at $n = 128$. And, though we are interested in more accuracy, it makes sense to start with a target margin of error of 0.05. The idea is to start out with rough estimation, and get more accurate only once we think we are close to the right n.

```
> n1 <- 64 ; mu1 <- 1 ; sd1 <- 2 # Control Group
> n2 <- 64 ; mu2 <- 3 ; sd2 <- 6 # Experimental Group
>
> con <- rnorm(n1,mu1,sd1) ; exp <- rnorm(n2,mu2,sd2)

> help(t.test)
```

The output of help is omitted, but we learn that the default is a test assuming unequal variances – just what we want.

```
> t.test(con,exp)

Welch Two Sample t-test

data:  con and exp
t = -2.4462, df = 78.609, p-value = 0.01667
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -3.4632952 -0.3556207
sample estimates:
mean of x mean of y
 1.117435  3.026893

> t.test(con,exp)[1]
$statistic
        t
-2.446186

> t.test(con,exp)[3]
$p.value
[1] 0.01667109
>
> m <- 500 # Monte Carlo sample size (Number of simulations)
> numsig <- 0 # Initializing
> for(i in 1:m)
+     {
+     con <- rnorm(n1,mu1,sd1) ; exp <- rnorm(n2,mu2,sd2)
+     numsig <- numsig+(t.test(con,exp)[3]<.05)
+     }
> pow <- numsig/m
> cat ("Monte Carlo Power = ",pow,"\n") ; cat ("\n")
Monte Carlo Power =  0.708

> m <- 500 # Monte Carlo sample size (Number of simulations)
> numsig <- 0 # Initializing
> for(i in 1:m)
+     {
+     con <- rnorm(n1,mu1,sd1) ; exp <- rnorm(n2,mu2,sd2)
```

```
+     numsig <- numsig+(t.test(con,exp)[3]<.05)
+     }
> pow <- numsig/m
> cat ("Monte Carlo Power = ",pow,"\n") ; cat ("\n")
Monte Carlo Power =  0.698
```

Try it again.

```
>
> m <- 500 # Monte Carlo sample size (Number of simulations)
> numsig <- 0 # Initializing
> for(i in 1:m)
+     {
+     con <- rnorm(n1,mu1,sd1) ; exp <- rnorm(n2,mu2,sd2)
+     numsig <- numsig+(t.test(con,exp)[3]<.05)
+     }
> pow <- numsig/m
> cat ("Monte Carlo Power = ",pow,"\n") ; cat ("\n")
Monte Carlo Power =  0.702
```

Try a larger sample size.

```
> n1 <- 80 ; mu1 <- 1 ; sd1 <- 2 # Control Group
> n2 <- 80 ; mu2 <- 3 ; sd2 <- 6 # Experimental Group
> m <- 500 # Monte Carlo sample size (Number of simulations)
> numsig <- 0 # Initializing
> for(i in 1:m)
+     {
+     con <- rnorm(n1,mu1,sd1) ; exp <- rnorm(n2,mu2,sd2)
+     numsig <- numsig+(t.test(con,exp)[3]<.05)
+     }
> pow <- numsig/m
> cat ("Monte Carlo Power = ",pow,"\n") ; cat ("\n")
Monte Carlo Power =  0.812
```

Try it again.

```
> n1 <- 80 ; mu1 <- 1 ; sd1 <- 2 # Control Group
> n2 <- 80 ; mu2 <- 3 ; sd2 <- 6 # Experimental Group
```

```
> m <- 500 # Monte Carlo sample size (Number of simulations)
> numsig <- 0 # Initializing
> for(i in 1:m)
+     {
+     con <- rnorm(n1,mu1,sd1) ; exp <- rnorm(n2,mu2,sd2)
+     numsig <- numsig+(t.test(con,exp)[3]<.05)
+     }
> pow <- numsig/m
> cat ("Monte Carlo Power = ",pow,"\n") ; cat ("\n")
Monte Carlo Power =  0.792
```

It seems that was a remarkably lucky guess. Now seek margin of error around 0.01.

```
>
> m <- 10000 # Monte Carlo sample size (Number of simulations)
> numsig <- 0 # Initializing
> for(i in 1:m)
+     {
+     con <- rnorm(n1,mu1,sd1) ; exp <- rnorm(n2,mu2,sd2)
+     numsig <- numsig+(t.test(con,exp)[3]<.05)
+     }
> pow <- numsig/m
> cat ("Monte Carlo Power = ",pow,"\n") ; cat ("\n")
Monte Carlo Power =  0.8001

> merror <- function(phat,m,alpha) # (1-alpha)*100% merror for a proportion
+     {
+     z <- qnorm(1-alpha/2)
+     merror <- z * sqrt(phat*(1-phat)/m)  # m is (Monte Carlo) sample size
+     merror
+     }
> margin <- merror(.8001,10000,.01) ; margin
[1] 0.01030138
> cat("99% CI from ",(pow-margin)," to ",(pow+margin),"\n")
99% CI from  0.7897986  to  0.810
```

This is very nice, except that I can't believe equal sample sizes are optimal when the variances are unequal. Let's try sample sizes proportional to the

standard deviations, so $n_1 = 40$ and $n_2 = 120$. The idea is that perhaps the two population means should be estimated with roughly the same precision, and we need a bigger sample size in the experimental condition to compensate for the larger variance. Well, actually I chose the relative sample sizes to minimize the standard deviation of the sampling distribution of the difference between means — the quantity that is estimated by the denominator of the t statistic.

```
> n1 <- 40 ; mu1 <- 1 ; sd1 <- 2 # Control Group
> n2 <- 120 ; mu2 <- 3 ; sd2 <- 6 # Experimental Group
> m <- 500 # Monte Carlo sample size (Number of simulations)
> numsig <- 0 # Initializing
> for(i in 1:m)
+     {
+     con <- rnorm(n1,mu1,sd1) ; exp <- rnorm(n2,mu2,sd2)
+     numsig <- numsig+(t.test(con,exp)[3]<.05)
+     }
> pow <- numsig/m
> cat ("Monte Carlo Power = ",pow,"\n") ; cat ("\n")
Monte Carlo Power =  0.89

> margin <- merror(pow,m,.01)
> cat("99% CI from ",(pow-margin)," to ",(pow+margin),"\n")
99% CI from  0.8539568  to  0.9260432
>
> # This is promising. Get some precision.
>
> n1 <- 40 ; mu1 <- 1 ; sd1 <- 2 # Control Group
> n2 <- 120 ; mu2 <- 3 ; sd2 <- 6 # Experimental Group
> m <- 10000 # Monte Carlo sample size (Number of simulations)
> numsig <- 0 # Initializing
> for(i in 1:m)
+     {
+     con <- rnorm(n1,mu1,sd1) ; exp <- rnorm(n2,mu2,sd2)
+     numsig <- numsig+(t.test(con,exp)[3]<.05)
+     }
> pow <- numsig/m
> cat ("Monte Carlo Power = ",pow,"\n") ; cat ("\n")
```

```
Monte Carlo Power =  0.8803

> margin <- merror(pow,m,.01)
> cat("99% CI from ",(pow-margin)," to ",(pow+margin),"\n")
99% CI from  0.8719386  to  0.8886614
```

So again we see that power depends on *design* as well as on effect size and sample size. It will be left as an exercise to find out how much sample size we could save (over the $n_1 = n_2 = 80$ solution) by taking this into account in the present case.

Finally, it should be clear that R has a t-test function, and the custom function `ttest` was unnecessary. What other classical tests are available?

```
> library(help=ctest)
ctest           Classical Tests

Description:

Package: ctest
Version: 1.4.0
Priority: base
Title: Classical Tests
Author: Kurt Hornik <Kurt.Hornik@ci.tuwien.ac.at>, with major
  contributions by Peter Dalgaard <p.dalgaard@kubism.ku.dk> and
  Torsten Hothorn <Torsten.Hothorn@rzmail.uni-erlangen.de>.
Maintainer: R Core Team <R-core@r-project.org>
Description: A collection of classical tests, including the
  Ansari-Bradley, Bartlett, chi-squared, Fisher, Kruskal-Wallis,
  Kolmogorov-Smirnov, t, and Wilcoxon tests.
License: GPL

Index:

ansari.test             Ansari-Bradley Test
bartlett.test           Bartlett Test for Homogeneity of Variances
binom.test              Exact Binomial Test
chisq.test              Pearson's Chi-squared Test for Count Data
```

cor.test	Test for Zero Correlation
fisher.test	Fisher's Exact Test for Count Data
fligner.test	Fligner-Killeen Test for Homogeneity of Variances
friedman.test	Friedman Rank Sum Test
kruskal.test	Kruskal-Wallis Rank Sum Test
ks.test	Kolmogorov-Smirnov Tests
mantelhaen.test	Cochran-Mantel-Haenszel Chi-Squared Test for Count Data
mcnemar.test	McNemar's Chi-squared Test for Count Data
mood.test	Mood Two-Sample Test of Scale
oneway.test	Test for Equal Means in a One-Way Layout
pairwise.prop.test	Pairwise comparisons of proportions
pairwise.t.test	Pairwise t tests
pairwise.table	Tabulate p values for pairwise comparisons
pairwise.wilcox.test	Pairwise Wilcoxon rank sum tests
power.prop.test	Power calculations two sample test for of proportions
power.t.test	Power calculations for one and two sample t tests
print.pairwise.htest	Print method for pairwise tests
print.power.htest	Print method for power calculation object
prop.test	Test for Equal or Given Proportions
prop.trend.test	Test for trend in proportions
quade.test	Quade Test
shapiro.test	Shapiro-Wilk Normality Test
t.test	Student's t-Test
var.test	F Test to Compare Two Variances
wilcox.test	Wilcoxon Rank Sum and Signed Rank Tests

Chapter 11

Computer-intensive Tests

This chapter covers two methods of statistical inference in which computing power and random number generation largely substitute for statistical theory: randomization tests and tests based on the bootstrap. These methods allow the creation of customized non-parametric tests without having to produce a new statistical theory each time.

11.1 Permutation Tests and Randomization Tests

11.1.1 Permutation Tests

Randomization tests use the Law of Large Numbers to approximate permutation tests, so we will begin with permutation tests. A **permutation** is an arrangement of a set of objects in some order; so for example, we say there are $5! = 5 \times 4 \times 3 \times 2 \times 1$ permutations of 5 objects. That is, 5 objects may be arranged in 120 different orders.

Permutation tests are most natural in the setting of a true experimental study with random assignment of subjects to treatments, so that all possible assignments are equally likely. The reasoning goes like this. If the treatment is completely ineffective, then the data are what they are, and the only reason that some test statistic might differ between treatments is by chance, because of the random assignment. This is the null hypothesis.

The set of all possible permutations of the data yields the set of all possible assignments to experimental conditions. Under the null hypothesis, these

are equally likely. This does *not* mean that all values of the test statistic are equally likely; not at all! Depending on the particular values of the data, there might be quite a few ties, and the distribution of the test statistic might have an arbitrarily peculiar shape. However, if we had enough time, we could calculate exactly what it is, as follows.

Generate all possible permutations of the data. For each permutation, compute the value of the test statistic. The histogram of the test statistic's values (to be precise, the relative frequency histogram of those values) is called the **permutation distribution** of the test statistic.

If the null hypothesis holds, the test statistic has the permutation distribution. If not, it has some other distribution. Suppose the observed value of the test statistic (that is, the one that we computed from the *unscrambled* data) is far out on the tail of the permutation distribution. Then the data may be deemed unlikely given the hull hypothesis — possibly unlikely enough so that the null hypothesis may be rejected, and we may conclude that the treatment has some effect.

In particular, the proportion of the permutation distribution at or beyond the observed test statistic will be called the **permutation p-value**. As usual,if $p < 0.05$, we'll claim statistical significance.

Don't you think this is more reasonable than doing an experiment with random assignment, and then proceeding to assume a normal distribution in some hypothetical "population" of subjects who *might* have received the various experimental treatments? Fisher (who came up with permutation tests as well as the F-test) thought so. In his classic *Statistical Methods for Research Workers* (1936) he wrote, after describing how to do a permutation test,

> Actually, the statistician does not carry out this very tedious process but his conclusions have no justification beyond the fact they could have been arrived at by this very elementary method.

To summarize, a permutation test is conducted by following these three steps.

1. Compute some test statistic using the set of original observations
2. Re-arrange the observations in all possible orders, computing the test statistic each time.

3. Calculate the permutation test p-value, which is the proportion of test statistic values from the re-arranged data that equal or exceed the value of the test statistic from the original data.

Several comments about permutation tests are in order.

- Please notice that no distribution at all is being assumed for the data. They are what they are, period. In fact, for observational data as well as experimental data, *permutation tests are distribution-free under the null hypothesis*. In this sense, permutation tests are non-parametric.

- For observational studies too, the null hypothesis is that the independent variable(s) and dependent variable(s) are independent.

- It's even better than that. Bell and Doksum (1967) proved that *any* valid distribution test of independence *must* be a permutation test (maybe a permutation test in disguise).

- Some non-parametric methods depend on large sample sizes for their validity. Permutation tests do not. Even for tiny samples, the chance of false significance cannot exceed 0.05.

- It doesn't matter if data are categorical or quantitative. By scrambling the data, any possible relationship between IV and DV is destroyed.

- If either IV or DV is multivariate, scramble *vectors* of data.

- The explanation of permutation tests referred to "the" test statistic, without indicating what that test statistic might be. In fact, the test statistic is up to you. No matter what you choose, the chance of false significance is limited to 0.05.

  What choice is best? It depends on the exact way in which the independent and dependent variables are related. A test statistic that captures the nature of the dependence will yield a more powerful, and hence a better test. So one option is to use your intuition, and make something up. Another option is to look in a book like Good's *Permutation Tests*. There, you'll find good suggestions for a lot of common hypothesis-testing problems. These suggestions are not just based on hunches. They are based on research, in which the statistical researcher has tried to derive a test statistic with maximum power for some class

of alternative hypotheses. If you think the null hypothesis might be false in the specified way, such a test statistic will likely perform better than anything you happen to come up with.

Many scientists who use permutation tests just compute something traditional like an F statistic, but compare it to a permutation distribution rather than the F distribution. You usually can't go too far wrong with this approach. It's optimal when the traditional assumptions hold, quite good when they almost hold, and the resulting tests tend to become very powerful for a broad range of alternative hypotheses as the sample size increases.

Another advantage of using traditional test statistics is that everyone has heard of them, and they do not arouse suspicion. If you make up something strange, people may think that you tried more traditional quantities first, and then eventually found a statistic that made the test significant. There's no doubt about it; you *can* fraudulently obtain significance with a permutation test by fishing for a test statistic until you find one that exploits a chance pattern in the data.

- Even with some combinatoric simplification (you can often get away without listing *all* the permutations) and a lot of computing power, permutation tests are not easy to do in practice. Fisher himself considered permutation tests to be entirely hypothetical, but that was before computers.

- One way around the computational problem is to convert the data to ranks, and then do it. Then, permutation distributions can be figured out in advance, by a combination of cleverness and brute force. All the common non-parametric rank tests are permutation tests carried out on ranks. Fisher's exact test is a permutation test for categorical data.

  Often, you'll see Z or chi-square statistics for the rank tests. Since the normal and chi-square distributions are continuous, while permutation distributions are always discrete, you know these have to be large-sample approximations based somehow on the Central Limit Theorem. But aren't permutation tests valid for small samples? Yes! The way it works is that good nonparametric books have tables that give exact critical values for small samples; the Z and chi-square approximations are used once the sample size becomes big enough for the approximations

to be valid – and big enough so that the exact permutation distribution (even of the ranks) is hard to compute. But statistical *software* often gives you p-values based on the large-sample approximation, regardless of what the sample size is. This throws away the small-sample virtues of the tests. If you use rank tests with small samples, it's up to you to find the appropriate table and learn how to use it.

- The modern way around the computational problem is to approximate (that is, estimate) the p-value of a permutation test using the Law of Large Numbers. That's called a randomization test, and it's the topic of the next section.

11.1.2 Randomization Tests

The permutation test p-value is the area under the curve (relative frequency histogram) of the permutation distribution, at or beyond the observed value of the test statistic. When we approximate the p-value of a permutation test by simulation, it's called a **randomization test**. Here's how to do it.

- Place the values of the dependent variable in a random order.
- Compute the test statistic for the randomly shuffled data.

In this way, we have randomly sampled a value of the test statistic from its permutation distribution. Carry out this procedure a large number of times. By the Law of Large Numbers, the the permutation p-value is approximated by the proportion of randomly generated values that exceed or equal the observed value of the test statistic. This proportion is the p-value of the randomization test.

The approximation gets better as the Monte Carlo sample size increases. We'll denote the Monte Carlo sample size by m, the permutation test p-value by p, and the randomization test p-value by $\widehat{p}$.

How big should the Monte Carlo sample size be? Here's one approach. As usual, it's based on a normal approximation to the binomial distribution.

```
############################################################
# Choose Monte Carlo sample size for a randomization     #
# test. Estimate p (p-value of permutation test) with    #
# p-hat. For a given true p (default = 0.04) and         #
# a given alpha (default = 0.05), returns the MC sample  #
# size needed to get p-hat < alpha with probability cc   #
# (default = .99).                                       #
############################################################
randm <- function(p=.04,alpha=0.05,cc=.99)
    {
    randm <- qnorm(cc)^2 * p*(1-p) / (alpha-p)^2
    randm <- trunc(randm+1) # Round up to next integer
    randm
    } # End of function randm

> probs <- c(.01,.02,.03,.04,.045,.049)
> cbind(probs,randm(p=probs)) # Use default values of alpha and cc
       [,1]   [,2]
[1,]  0.010     34
[2,]  0.020    118
[3,]  0.030    394
[4,]  0.040   2079
[5,]  0.045   9304
[6,]  0.049 252189
```

Student's Sleep Data

This example is simple as well as classical, but its simplicity allows the examination of basic issues. The data are from a paper by William Gossett, who published anonymously under the name "Student," and after whom the *Student's t* distribution is named. The data show the effect of two soporific drugs (increase in hours of sleep) on groups consisting of 10 patients each. The independent variable is `group`, and the dependent variable is `extra` (for extra hours of sleep). The source is Student (1908) The probable error of the mean. *Biometrika*, **6**, 20.

```
credit.erin > cat sleep.dat
```

```
   extra group
1    0.7     1
2   -1.6     1
3   -0.2     1
4   -1.2     1
5   -0.1     1
6    3.4     1
7    3.7     1
8    0.8     1
9    0.0     1
10   2.0     1
11   1.9     2
12   0.8     2
13   1.1     2
14   0.1     2
15  -0.1     2
16   4.4     2
17   5.5     2
18   1.6     2
19   4.6     2
20   3.4     2

credit.erin > R --vanilla < randex1.R > randex1.out
credit.erin > cat randex1.out

R : Copyright 2001, The R Development Core Team
Version 1.4.0  (2001-12-19)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type ‘license()’ or ‘licence()’ for distribution details.

R is a collaborative project with many contributors.
Type ‘contributors()’ for more information.

Type ‘demo()’ for some demos, ‘help()’ for on-line help, or
‘help.start()’ for a HTML browser interface to help.
Type ‘q()’ to quit R.
```

```
> # randex1.R :  First randomization test example, with Student's Sleep Data
> # Monte Carlo sample size m may be set interactively
> set.seed(4444) # Set seed for random number generation
>
> # Define margin of error functions
> merror <- function(phat,M,alpha) # (1-alpha)*100% merror for a proportion
+     {
+     z <- qnorm(1-alpha/2)
+     merror <- z * sqrt(phat*(1-phat)/M)  # M is (Monte Carlo) sample size
+     merror
+     }
> mmargin <- function(p,cc,alpha)
+            # Choose m to get (1-alpha)*100% margin of error equal to cc
+            {
+            mmargin <- p*(1-p)*qnorm(1-alpha/2)^2/cc^2
+            mmargin <- trunc(mmargin+1) # Round up to next integer
+            mmargin
+            } # End definition of function mmargin
> ###############
> sleepy <- read.table("sleep.dat")
> t.test(extra ~ group, var.equal=TRUE, data = sleepy)

        Two Sample t-test

data:  extra by group
t = -1.8608, df = 18, p-value = 0.07919
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -3.3638740  0.2038740
sample estimates:
mean in group 1 mean in group 2
           0.75            2.33

> t.test(extra ~ group, var.equal=TRUE, data = sleepy)[1]
$statistic
        t
-1.860813
```

```
> # It's a list element, not a number
> ObsT <- t.test(extra ~ group, var.equal=TRUE, data = sleepy)[[1]]
> ObsT
        t
-1.860813
>
> # If M is not assigned, it's 1210
> if(length(objects(pattern="M"))==0) M <- 1210
> cat("Monte Carlo Sample size M = ",M,"\n")
Monte Carlo Sample size M =  1210
> dv <- sleepy$extra ; iv <- sleepy$group
> trand <- numeric(M)
> for(i in 1:M)
+     { trand[i] <- t.test(sample(dv) ~ iv, var.equal=TRUE)[[1]] }
> randp <- length(trand[abs(trand)>=abs(ObsT)])/M
> margin <- merror(randp,M,.01)
>
> cat ("\n")

> cat ("Randomization p-value = ",randp,"\n")
Randomization p-value =  0.08429752
> cat("99% CI from ",(randp-margin)," to ",(randp+margin),"\n")
99% CI from  0.06372398  to  0.1048711
> cat ("\n")

>
> # Now try difference between medians
> cat("\n")

> cat("Median extra sleep for Group = 1: ",median(dv[iv==1]),"\n")
Median extra sleep for Group = 1:  0.35
> cat("Median extra sleep for Group = 2: ",median(dv[iv==2]),"\n")
Median extra sleep for Group = 2:  1.75
> ObsMedDif <- abs(median(dv[iv==1])-median(dv[iv==2]))
> cat("Absolute difference is ",ObsMedDif,"\n")
Absolute difference is  1.4
> cat("\n")
```

```
> trand2 <- numeric(M)
> for(i in 1:M)
+     {
+     rdv <- sample(dv)
+     trand2[i] <- abs(median(rdv[iv==1])-median(rdv[iv==2]))
+     }
> randp2 <- length(trand2[abs(trand2)>=abs(ObsMedDif)])/M
> margin <- merror(randp2,M,.01)
>
> cat ("\n")

> cat ("Randomization p-value for diff bet medians  = ",randp2,"\n")
Randomization p-value for diff bet medians  =  0.2090909
> cat("99% CI from ",(randp2-margin)," to ",(randp2+margin),"\n")
99% CI from  0.1789778  to  0.239204
> cat ("\n")
```

The main conclusion here is that the difference between group means is *not* significant. The traditional t-test (in fact, the first published t-test!) and the randomization test both have p-values around 0.08. This is not too surprising. We randomized the t statistic, and the traditional t-test is going to be appropriate for these data.

Then we try another test statistic — the difference between medians. This time we get a p-value near 0.21. This probably reflects lower power of the randomization test when we test medians rather than means on data that are actually normal.

Another thing to notice is that the 99% confidence interval for p does not include 0.05. This means that $\widehat{p}$ is not just less than 0.05, it's *significantly* less than 0.05 (at the 0.01 level). This is good. In fact, maybe it should be obligatory.

If it's really obligatory, then we need some kind of power analysis for choosing m. Letting p denote the true p-value from the permutation test, and letting α denote the significance level (for us, $\alpha = 0.05$ unless we're applying a Bonferroni correction), the traditional statistic for testing whether

p is different from α would be

$$Z^* = \frac{\widehat{P} - \alpha}{\sqrt{\frac{\alpha(1-\alpha)}{m}}},$$

which has a standard normal distribution under the null hypothesis. Some medium-grade calculations show that the probability that $\widehat{P}$ will be *significantly* different from α at level L (i.e., the power) with a true p-value of p is approximately

$$1 - Pr\left\{ \frac{\sqrt{m}(\alpha - p)}{\sqrt{p(1-p)}} - z_{1-L/2}\sqrt{\frac{\alpha(1-\alpha)}{p(1-p)}} < Z < \frac{\sqrt{m}(\alpha - p)}{\sqrt{p(1-p)}} + z_{1-L/2}\sqrt{\frac{\alpha(1-\alpha)}{p(1-p)}} \right\}$$

where Z has a standard normal distribution, and the approximation is excellent for m larger than a few hundred.

The preceding formula is just for the record, and to provide another opportunity to illustrate how a formula can be transcribed more or less directly into an S function.

```
# Power for detecting p-hat significantly different from alpha at
# significance level L, given true p and MC sample size M.
randmpow <- function(M,alpha=0.05,p=0.04,L=0.01)
    {
    z <- qnorm(1-L/2)
    left <- sqrt(M)*(alpha-p)/sqrt(p*(1-p))
    right <- sqrt( alpha/p * (1-alpha)/(1-p) )
    randmpow <- 1 - pnorm(left+z*right) + pnorm(left-z*right)
    randmpow
    } # End function randmpow
```

The function `findm` uses `randmpow` to search for the Monte Carlo sample size needed for a specified power. Again, the *power* we're talking about here is the power of a test for whether the randomization test p-value $\widehat{P}$ is different from 0.05.

```
findm <- function(wantpow=.8,mstart=1,aa=0.05,pp=0.04,LL=0.01)
    {
    pow <- 0
    mm <- mstart
    while(pow < wantpow)
        {
        mm <- mm+1
        pow <- randmpow(mm,aa,pp,LL)
        } # End while
    findm <- mm
    findm
    } # End function findm
```

Table 11.1.2 shows the result of applying the function `findm` to a selected set of true p values and desired power values.

Table 11.1: Monte Carlo sample size required to have specified probability that $\widehat{P}$ will be significantly different from 0.05 at the 0.01 level, when the true p-value is P

	Probability of Significance				
P	0.70	0.75	0.80	0.85	0.90
0.0001	129	130	131	132	133
0.0010	140	142	144	148	151
0.0050	177	184	191	199	210
0.0100	236	247	261	276	297
0.0200	448	478	513	555	610
0.0300	1,059	1,144	1,243	1,363	1,522
0.0400	4,411	4,811	5,276	5,845	6,602
0.0450	17,962	19,669	21,660	24,103	27,362
0.0550	18,548	20,459	22,697	25,452	29,143
0.0600	4,705	5,207	5,796	6,522	7,496
0.0700	1,209	1,345	1,506	1,705	1,974
0.0800	551	616	693	789	919
0.0900	317	356	403	461	539
0.1000	207	234	265	305	358
0.3000	11	13	15	18	22
0.5000	4	4	5	6	8

The Greenhouse Data Again

With permutation and randomization tests, it's a tricky business to carry out a test for a set of independent variables while controlling for another set. It's easy to preserve the relationships among multiple independent variables or multiple dependent variables by keeping them together, but it's hard to preserve the relationship of the dependent variable to one set of independent variables while destroying its relationship to another set by randomization.

There's one very important case where this is *not* a problem. In factorial designs with equal or proportional sample sizes, the independent variables are completely unrelated to each other, so we can just randomize the dependent variable (or collection of dependent variables). Here's an example from the greenhouse data.

```
credit.erin > head green.dat
                              PLANT    MCG    MEANLNG

                          1      1       7      50.714
                          2      1       9      10.793
                          3      3       8     106.514
                          4      3       7     102.243
                          5      3       9      73.214
                          6      1       3      10.471
                          7      2       2      13.536
credit.erin > R

> green <- read.table("green.dat")
> plant <- factor(green$PLANT) ; mcg <- factor(green$MCG)
> meanlng <- green$MEANLNG #$
> obs <- anova(lm(meanlng ~ plant*mcg))
> obs
Analysis of Variance Table

Response: meanlng
          Df Sum Sq Mean Sq  F value    Pr(>F)
plant      2 221695  110848 113.9032 < 2.2e-16 ***
mcg        5  58740   11748  12.0719 5.894e-09 ***
plant:mcg 10  47581    4758   4.8893 1.273e-05 ***
Residuals 90  87586     973
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> # This agrees with what we got from SAS

> obsF <- obs[1:3,4]
> obsF
         1          2          3
113.903170  12.071871   4.889303
>
> set.seed(4444)
> M <- 500 ; simf <- NULL
> for(i in 1:M)
+    {
```

```
+    simf <- rbind(simf,anova(lm(sample(meanlng)~plant*mcg))[1:3,4])
+    } # Next i (next simulation)

>
> plantp <- length(simf[,1][simf[,1]>=obsF[1]])/M ; plantp
[1] 0
> max(simf[,1])
[1] 7.460185
> min(simf[,1])
[1] 0.0003066219
> mcgp <- length(simf[,2][simf[,2]>=obsF[2]])/M ; mcgp
[1] 0
> intp <- length(simf[,3][simf[,3]>=obsF[3]])/M ; intp
[1] 0
> max(simf[,2])
[1] 4.54209
> max(simf[,3])
[1] 3.209669
```

The randomization p-value is approximately zero. We can't compute a meaningful confidence interval (why not?) but we can conclude that the permutation p-value is less than 0.05, because

```
> .05*sqrt(500)/sqrt(.05*.95)
[1] 5.129892
```

The Twins Data

Sherlock Holmes and the hat.

Long ago, there was more space in journals, and a journal called *Human Biology* used to publish raw data. The `twin data` contains educational test scores and physical measurements for a sample of high school age identical and fraternal twin pairs. Members of each twin pair were of the same sex. Except for a few cases where the parents were not sure, Twin One was born first and Twin Two was born second. The variables are:

1. SEX: 0=Male, 1=Female
2. IDENT: 0=Fraternal 1=Identical

3. PROGMAT1: Progressive matrices (puzzle) score for twin 1
4. REASON1: Reasoning score for twin 1
5. VERBAL1: Verbal (reading and vocabulary) score for twin 1
6. PROGMAT2: Progressive matrices (puzzle) score for twin 2
7. REASON2: Reasoning score for twin 2
8. VERBAL2: Verbal (reading and vocabulary) score for twin 2
9. HEADLNG1: Head Length of Twin 1
10. HEADBRD1: Head Breadth of Twin 1
11. HEADCIR1: Head Circumference of Twin 1
12. HEADLNG2: Head Length of Twin 2
13. HEADBRD2: Head Breadth of Twin 2
14. HEADCIR2: Head Circumference of Twin 2

This is a subset of the original data. Some variables like height and weight are not included. The reference is Clark, P. J., Vandenberg, S. G., and Proctor, C. H. (1961), "On the relationship of scores on certain psychological tests with a number of anthropometric characters and birth order in twins," *Human Biology*, **33**, 163-180.

We want to see if performance on the educational tests is related to head size.

```
/res/jbrunner/www/442/S > head smalltwin.dat
sex ident progmat1 reason1 verbal1 progmat2 reason2 verbal2 headlng1 headbrd1
headcir1 headlng2 headbrd2 headcir2
    1     1    1  48  53  66  35  42  61   183 140 522 188 138 535
    2     1    1  47  69  88  53  74  84   189 137 542 186 140 543
    3     1    1  35  68  92  42  61  86   185 145 549 186 140 550
    4     1    1  34  42  73  26  38  68   183 151 544 185 147 545
    5     1    1  49  71  95  38  72  97   174 145 534 186 143 543
    6     1    1  50  90 122  46  82 101   191 143 551 191 141 552
    7     1    1  25  30  42  28  37  43   184 143 511 186 143 535
```

```
    8    1   1  25  74  64  41  78  65   180 146 532 179 144 527
    9    1   1  23  19  52  23  36  59   193 146 560 191 145 551

/res/jbrunner/www/442/S > R

> twinframe <-  read.table("smalltwin.dat")
> sex <- twinframe$sex ; ident <- twinframe$ident
> sexfac <- factor(twinframe$sex,levels=c(0,1),label=c("Male","Female"))
> identfac <- factor(twinframe$ident,levels=c(0,1),
+                 label=c("Fraternal","Identical"))
> table(sexfac,identfac)
        identfac
sexfac   Fraternal Identical
  Male          13        21
  Female        20        20
> mental <- twinframe[,3:8]  # All rows, cols 3 to 8
> phys   <- twinframe[,9:14] # All rows, cols 9 to 14
> cor(mental,phys)
          headlng1   headbrd1  headcir1  headlng2   headbrd2  headcir2
progmat1 0.1945786 0.02669260 0.2046808 0.2070390 0.09577333 0.2204541
reason1  0.1232977 0.03186775 0.2052615 0.0978289 0.04733736 0.1955942
verbal1  0.2259473 0.05372263 0.2452086 0.2132409 0.07487114 0.2333709
progmat2 0.2863199 0.19917360 0.3128950 0.3446627 0.22308623 0.3739253
reason2  0.2127977 0.06950846 0.2767257 0.1226885 0.11543427 0.2521013
verbal2  0.2933130 0.16693928 0.3242051 0.2537764 0.22801336 0.3350497

>
> # But that's IGNORING sex and ident-frat. Want to CONTROL for them.
> n <- length(sex)
> mf <- (1:n)[sex==0&ident==0] # mf are indices of male fraternal pairs
> mi <- (1:n)[sex==0&ident==1] # mi are indices of male identical pairs
> ff <- (1:n)[sex==1&ident==0] # ff are indices of female fraternal pairs
> fi <- (1:n)[sex==1&ident==1] # fi are indices of female identical pairs

> mf
 [1] 62 63 64 65 66 67 68 69 70 71 72 73 74

> # Sub-sample sizes
```

```
> nmf <- length(mf)  ; nmi <- length(mi)
> nff  <- length(ff) ; nfi <- length(fi)
> nmf ; nmi ; nff ; nfi
[1] 13
[1] 21
[1] 20
[1] 20
> table(sexfac,identfac)
        identfac
sexfac   Fraternal Identical
  Male          13        21
  Female        20        20

> # mentalmf are mental scores of male fraternal pairs, etc.
> mentalmf <- mental[mf,] ; physmf <- phys[mf,]
> mentalmi <- mental[mi,] ; physmi <- phys[mi,]
> mentalff <- mental[ff,] ; physff <- phys[ff,]
> mentalfi <- mental[fi,] ; physfi <- phys[fi,]

> mentalmf
   progmat1 reason1 verbal1 progmat2 reason2 verbal2
62       58      91     128       54      73     129
63       44      46      79       42      34      42
64       44      43      70       43      36      58
65       36      40      63       42      39      63
66       34      21      53       45      31      70
67       50      70      93       45      67     109
68       50      81     101       41      47      96
69       31      76     122       43      70      75
70       23      29      62       26      29      42
71       52      66     114       42      69     120
72       48      51      62       30      35      49
73       23      48      78       38      62      87
74       28      38      62       55      70     105

> # First three rows
> mentalmf[1:3,]
   progmat1 reason1 verbal1 progmat2 reason2 verbal2
```

```
62       58      91     128       54      73     129
63       44      46      79       42      34      42
64       44      43      70       43      36      58
> # Last 3 columns
> mentalmf[,4:6]
   progmat2 reason2 verbal2
62       54      73     129
63       42      34      42
64       43      36      58
65       42      39      63
66       45      31      70
67       45      67     109
68       41      47      96
69       43      70      75
70       26      29      42
71       42      69     120
72       30      35      49
73       38      62      87
74       55      70     105
> # Rows in random order
> mentalmf[sample(1:13),]
   progmat1 reason1 verbal1 progmat2 reason2 verbal2
71       52      66     114       42      69     120
73       23      48      78       38      62      87
66       34      21      53       45      31      70
69       31      76     122       43      70      75
65       36      40      63       42      39      63
68       50      81     101       41      47      96
64       44      43      70       43      36      58
70       23      29      62       26      29      42
62       58      91     128       54      73     129
63       44      46      79       42      34      42
74       28      38      62       55      70     105
67       50      70      93       45      67     109
72       48      51      62       30      35      49
>
```

That's how we'll randomize. Back to CONTROLLING for sex, ident.

```
> # mentalmf are mental scores of male fraternal pairs, etc.
> mentalmf <- mental[mf,] ; physmf <- phys[mf,]
> mentalmi <- mental[mi,] ; physmi <- phys[mi,]
> mentalff <- mental[ff,] ; physff <- phys[ff,]
> mentalfi <- mental[fi,] ; physfi <- phys[fi,]
>
> cor(mentalmf,physmf)
          headlng1    headbrd1   headcir1   headlng2   headbrd2   headcir2
progmat1 0.3534186 -0.53715165 0.05247501 -0.1486551 -0.3335911 -0.2541279
reason1  0.4784903 -0.04435345 0.40868525  0.2009069 -0.1853897  0.1574282
verbal1  0.3333061  0.02578888 0.36744645  0.1507982 -0.1958353  0.1267843
progmat2 0.5712273 -0.16389337 0.37080025  0.5622139 -0.1996214  0.4073323
reason2  0.4886337  0.38731941 0.63957418  0.4271557  0.2587126  0.6682264
verbal2  0.5278153  0.25599312 0.62836834  0.3403694  0.1966882  0.6113976
>
> # Don't want to correlate mental twin 1 with phys twin 2
>
> cor(mentalmf[,1:3],physmf[,1:3])
          headlng1    headbrd1   headcir1
progmat1 0.3534186 -0.53715165 0.05247501
reason1  0.4784903 -0.04435345 0.40868525
verbal1  0.3333061  0.02578888 0.36744645
> max(abs(cor(mentalmf[,1:3],physmf[,1:3])))
[1] 0.5371517
>


> cor(mentalmf[,4:6],physmf[,4:6])
          headlng2   headbrd2  headcir2
progmat2 0.5622139 -0.1996214 0.4073323
reason2  0.4271557  0.2587126 0.6682264
verbal2  0.3403694  0.1966882 0.6113976
> max(abs(cor(mentalmf[,4:6],physmf[,4:6])))
[1] 0.6682264
>
>
> cor(mentalmi[,1:3],physmi[,1:3])
```

```
          headlng1   headbrd1  headcir1
progmat1 0.2334577 0.26536909 0.3193472
reason1  0.2622690 0.37549903 0.3534622
verbal1  0.4436284 0.06643773 0.3480645
> max(abs(cor(mentalmi[,1:3],physmi[,1:3])))
[1] 0.4436284
> cor(mentalmi[,4:6],physmi[,4:6])
          headlng2  headbrd2  headcir2
progmat2 0.3645763 0.2537397 0.3699872
reason2  0.1682737 0.4212712 0.3873012
verbal2  0.1814358 0.1590209 0.2112241
> max(abs(cor(mentalmi[,4:6],physmi[,4:6])))
[1] 0.4212712
>
> cor(mentalff[,1:3],physff[,1:3])
            headlng1  headbrd1  headcir1
progmat1 -0.09894825 0.1031112 0.1024857
reason1   0.10353527 0.1974691 0.2299249
verbal1   0.04068947 0.1458637 0.0710240
> max(abs(cor(mentalff[,1:3],physff[,1:3])))
[1] 0.2299249
> cor(mentalff[,4:6],physff[,4:6])
            headlng2  headbrd2  headcir2
progmat2 -0.05058245 0.3809976 0.1205803
reason2   0.19569669 0.3570053 0.2617820
verbal2   0.24212501 0.3964967 0.2463883
> max(abs(cor(mentalff[,4:6],physff[,4:6])))
[1] 0.3964967
>
> cor(mentalfi[,1:3],physfi[,1:3])
            headlng1    headbrd1     headcir1
progmat1 -0.01443227 -0.34580801 -0.004887716
reason1   0.15174745  0.04052029  0.304039946
verbal1   0.22504203 -0.01581501  0.341174647
> max(abs(cor(mentalfi[,1:3],physfi[,1:3])))
[1] 0.345808
> cor(mentalfi[,4:6],physfi[,4:6])
          headlng2    headbrd2  headcir2
```

```
progmat2 0.4030654 -0.02036423 0.4244152
reason2  0.3233766  0.05661767 0.4178053
verbal2  0.2702130  0.15930201 0.4025376
> max(abs(cor(mentalfi[,4:6],physfi[,4:6])))
[1] 0.4244152
>
> # test sta will be absobs = 0.6682264
> obsmax <- max( c(
+                 cor(mentalmf[,1:3],physmf[,1:3]),
+                 cor(mentalmf[,4:6],physmf[,4:6]),
+                 cor(mentalmi[,1:3],physmi[,1:3]),
+                 cor(mentalmi[,4:6],physmi[,4:6]),
+                 cor(mentalff[,1:3],physff[,1:3]),
+                 cor(mentalff[,4:6],physff[,4:6]),
+                 cor(mentalfi[,1:3],physfi[,1:3]),
+                 cor(mentalfi[,4:6],physfi[,4:6])    )   )
>
> obsmax
[1] 0.6682264
>
> obsmin <- min( c(
+                 cor(mentalmf[,1:3],physmf[,1:3]),
+                 cor(mentalmf[,4:6],physmf[,4:6]),
+                 cor(mentalmi[,1:3],physmi[,1:3]),
+                 cor(mentalmi[,4:6],physmi[,4:6]),
+                 cor(mentalff[,1:3],physff[,1:3]),
+                 cor(mentalff[,4:6],physff[,4:6]),
+                 cor(mentalfi[,1:3],physfi[,1:3]),
+                 cor(mentalfi[,4:6],physfi[,4:6])    )   )
> obsmin
[1] -0.5371517
>
> absobs <- max(abs(obsmax),abs(obsmin)) # Test Statistic
> absobs
[1] 0.6682264
>
> ####
> # Here's how we'll sample. Recall mentalmf <- mental[mf,]
```

```
> ####
> mf
 [1] 62 63 64 65 66 67 68 69 70 71 72 73 74
> mentalmf
   progmat1 reason1 verbal1 progmat2 reason2 verbal2
62       58      91     128       54      73     129
63       44      46      79       42      34      42
64       44      43      70       43      36      58
65       36      40      63       42      39      63
66       34      21      53       45      31      70
67       50      70      93       45      67     109
68       50      81     101       41      47      96
69       31      76     122       43      70      75
70       23      29      62       26      29      42
71       52      66     114       42      69     120
72       48      51      62       30      35      49
73       23      48      78       38      62      87
74       28      38      62       55      70     105
> mental[sample(mf),]
   progmat1 reason1 verbal1 progmat2 reason2 verbal2
72       48      51      62       30      35      49
66       34      21      53       45      31      70
62       58      91     128       54      73     129
69       31      76     122       43      70      75
70       23      29      62       26      29      42
71       52      66     114       42      69     120
67       50      70      93       45      67     109
74       28      38      62       55      70     105
63       44      46      79       42      34      42
68       50      81     101       41      47      96
73       23      48      78       38      62      87
65       36      40      63       42      39      63
64       44      43      70       43      36      58
>
> rmentalmf <- mental[sample(mf),]
> rmentalmi <- mental[sample(mi),]
> rmentalff <- mental[sample(ff),]
> rmentalfi <- mental[sample(fi),]
```

```
>
> rcorrs <- c(
+             cor(rmentalmf[,1:3],physmf[,1:3]),
+             cor(rmentalmf[,4:6],physmf[,4:6]),
+             cor(rmentalmi[,1:3],physmi[,1:3]),
+             cor(rmentalmi[,4:6],physmi[,4:6]),
+             cor(rmentalff[,1:3],physff[,1:3]),
+             cor(rmentalff[,4:6],physff[,4:6]),
+             cor(rmentalfi[,1:3],physff[,1:3]),
+             cor(rmentalfi[,4:6],physff[,4:6])  )
>
> min(rcorrs) ; max(rcorrs)
[1] -0.5673855
[1] 0.5166834
> rmin <- NULL ; rmax <- NULL ; rabs <- NULL
>
> # Now simulate
> M <- 200 ; set.seed(4444)
> for(i in 1:M)
+     {
+     rmentalmf <- mental[sample(mf),]
+     rmentalmi <- mental[sample(mi),]
+     rmentalff <- mental[sample(ff),]
+     rmentalfi <- mental[sample(fi),]
+     rcorrs <- c(
+             cor(rmentalmf[,1:3],physmf[,1:3]),
+             cor(rmentalmf[,4:6],physmf[,4:6]),
+             cor(rmentalmi[,1:3],physmi[,1:3]),
+             cor(rmentalmi[,4:6],physmi[,4:6]),
+             cor(rmentalff[,1:3],physff[,1:3]),
+             cor(rmentalff[,4:6],physff[,4:6]),
+             cor(rmentalfi[,1:3],physff[,1:3]),
+             cor(rmentalfi[,4:6],physff[,4:6])  )
+     rmin <- c(rmin,min(rcorrs))
+     rmax <- c(rmax,max(rcorrs))
+     rabs <- c(rabs,max(abs(min(rcorrs)),abs(max(rcorrs))))
+     }
> cbind(rmin,rmax,rabs)[1:20,] # First 20 rows
```

```
            rmin      rmax      rabs
 [1,] -0.6521097 0.6024060 0.6521097
 [2,] -0.4410713 0.6091124 0.6091124
 [3,] -0.5635999 0.3953340 0.5635999
 [4,] -0.6655059 0.6937127 0.6937127
 [5,] -0.5110777 0.3692450 0.5110777
 [6,] -0.4513148 0.7600707 0.7600707
 [7,] -0.3180858 0.5724620 0.5724620
 [8,] -0.6258317 0.4013421 0.6258317
 [9,] -0.4061387 0.5174977 0.5174977
[10,] -0.5004209 0.4688702 0.5004209
[11,] -0.6437074 0.3458846 0.6437074
[12,] -0.4065318 0.2945435 0.4065318
[13,] -0.6115288 0.5631299 0.6115288
[14,] -0.4709578 0.5452405 0.5452405
[15,] -0.6060098 0.6110585 0.6110585
[16,] -0.4220454 0.3177893 0.4220454
[17,] -0.3407132 0.5021933 0.5021933
[18,] -0.5861414 0.3645763 0.5861414
[19,] -0.6137978 0.4693924 0.6137978
[20,] -0.4509271 0.4157352 0.4509271
>
> length(rabs[rabs>=absobs])/M # Two sided
[1] 0.135
> length(rmin[rmin<=obsmin])/M # Lower tailed
[1] 0.395
> length(rmax[rmax>=obsmax])/M # Upper tailed
[1] 0.07
```

Now let's put the wole thing together. Make a file that just does the analysis and prints the results. How many simulations should we use? I'd like to make sure that $\widehat{P}$ is significantly different from 0.07, so I run

```
> findm
function(wantpow=.8,mstart=1,aa=0.05,pp=0.04,LL=0.01)
    {
    pow <- 0
    mm <- mstart
```

```
    while(pow < wantpow)
        {
        mm <- mm+1
        pow <- randmpow(mm,aa,pp,LL)
        } # End while
    findm <- mm
    findm
    } # End function findm
>
> findm(pp=.07)
[1] 1506
```

and choose $m = 1600$. First I'll show you the output, then a listing of the program `twins.R`.

```
> source("twins.R")
Male Fraternal
   Twin 1
          headlng1    headbrd1   headcir1
progmat1 0.3534186 -0.53715165 0.05247501
reason1  0.4784903 -0.04435345 0.40868525
verbal1  0.3333061  0.02578888 0.36744645

   Twin 2
          headlng2   headbrd2  headcir2
progmat2 0.5622139 -0.1996214 0.4073323
reason2  0.4271557  0.2587126 0.6682264
verbal2  0.3403694  0.1966882 0.6113976


Male Identical
   Twin 1
          headlng1   headbrd1  headcir1
progmat1 0.2334577 0.26536909 0.3193472
reason1  0.2622690 0.37549903 0.3534622
verbal1  0.4436284 0.06643773 0.3480645
   Twin 2
          headlng2  headbrd2  headcir2
```

```
progmat2 0.3645763 0.2537397 0.3699872
reason2  0.1682737 0.4212712 0.3873012
verbal2  0.1814358 0.1590209 0.2112241

Female Fraternal
   Twin 1
           headlng1  headbrd1  headcir1
progmat1 -0.09894825 0.1031112 0.1024857
reason1   0.10353527 0.1974691 0.2299249
verbal1   0.04068947 0.1458637 0.0710240
   Twin 2
           headlng2  headbrd2  headcir2
progmat2 -0.05058245 0.3809976 0.1205803
reason2   0.19569669 0.3570053 0.2617820
verbal2   0.24212501 0.3964967 0.2463883

Female Identical
   Twin 1
           headlng1    headbrd1     headcir1
progmat1 -0.01443227 -0.34580801 -0.004887716
reason1   0.15174745  0.04052029  0.304039946
verbal1   0.22504203 -0.01581501  0.341174647
   Twin 2
          headlng2    headbrd2  headcir2
progmat2 0.4030654 -0.02036423 0.4244152
reason2  0.3233766  0.05661767 0.4178053
verbal2  0.2702130  0.15930201 0.4025376


Correlations Between Mental and Physical


     Minimum Observed Correlation:  -0.5371517
     Randomization p-value (one-sided): p-hat =  0.416875
     Plus or minus 99% Margin of error =  0.03174979

     Maximum Observed Correlation:  0.6682264
     Randomization p-value (one-sided): p-hat =  0.10625
```

```
     Plus or minus 99% Margin of error =  0.01984402

     Maximum Observed Absolute Correlation:  0.6682264
     Randomization p-value (two-sided): p-hat =  0.199375
     Plus or minus 99% Margin of error =  0.02572806
```

And here is a listing of the program.

```
# twins.R
# Just do the analysis - no examples or explanation with source("twins.R")
twinframe <-  read.table("smalltwin.dat")
sex <- twinframe$sex ; ident <- twinframe$ident
mental <- twinframe[,3:8]  # All rows, cols 3 to 8
phys   <- twinframe[,9:14] # All rows, cols 9 to 14
n <- length(sex)
mf <- (1:n)[sex==0&ident==0] # mf are indices of male fraternal pairs
mi <- (1:n)[sex==0&ident==1] # mi are indices of male identical pairs
ff <- (1:n)[sex==1&ident==0] # ff are indices of female fraternal pairs
fi <- (1:n)[sex==1&ident==1] # fi are indices of female identical pairs
# Sub-sample sizes
nmf <- length(mf)  ; nmi <- length(mi)
nff  <- length(ff) ; nfi <- length(fi)
# mentalmf are mental scores of male fraternal pairs, etc.
mentalmf <- mental[mf,] ; physmf <- phys[mf,]
mentalmi <- mental[mi,] ; physmi <- phys[mi,]
mentalff <- mental[ff,] ; physff <- phys[ff,]
mentalfi <- mental[fi,] ; physfi <- phys[fi,]

cat("Male Fraternal \n")
cat("   Twin 1   \n")
print(cor(mentalmf[,1:3],physmf[,1:3]))
cat("   Twin 2   \n")
print(cor(mentalmf[,4:6],physmf[,4:6]))
cat(" \n")

cat("Male Identical \n")
cat("   Twin 1   \n")
print(cor(mentalmi[,1:3],physmi[,1:3]))
cat("   Twin 2   \n")
```

```
print(cor(mentalmi[,4:6],physmi[,4:6]))
cat(" \n")

cat("Female Fraternal \n")
cat("   Twin 1   \n")
print(cor(mentalff[,1:3],physff[,1:3]))
cat("   Twin 2   \n")
print(cor(mentalff[,4:6],physff[,4:6]))
cat(" \n")

cat("Female Identical \n")
cat("   Twin 1   \n")
print(cor(mentalfi[,1:3],physfi[,1:3]))
cat("   Twin 2   \n")
print(cor(mentalfi[,4:6],physfi[,4:6]))
cat(" \n")

# test sta will be absobs = 0.6682264
# Keep track of minimum (neg corr: obsmin =  -0.5371517) and max too.


obsmax <- max( c(
                cor(mentalmf[,1:3],physmf[,1:3]),
                cor(mentalmf[,4:6],physmf[,4:6]),
                cor(mentalmi[,1:3],physmi[,1:3]),
                cor(mentalmi[,4:6],physmi[,4:6]),
                cor(mentalff[,1:3],physff[,1:3]),
                cor(mentalff[,4:6],physff[,4:6]),
                cor(mentalfi[,1:3],physfi[,1:3]),
                cor(mentalfi[,4:6],physfi[,4:6])    )   )
obsmin <- min( c(
                cor(mentalmf[,1:3],physmf[,1:3]),
                cor(mentalmf[,4:6],physmf[,4:6]),
                cor(mentalmi[,1:3],physmi[,1:3]),
                cor(mentalmi[,4:6],physmi[,4:6]),
                cor(mentalff[,1:3],physff[,1:3]),
                cor(mentalff[,4:6],physff[,4:6]),
                cor(mentalfi[,1:3],physfi[,1:3]),
```

```
                cor(mentalfi[,4:6],physfi[,4:6])    )   )

absobs <- max(abs(obsmax),abs(obsmin)) # Test Statistic

rmin <- NULL ; rmax <- NULL ; rabs <- NULL
# Now simulate. Want p-hat sig diff from 0.07. Use findm(pp=.07), get
# 1506, so use m=1600

M <- 1600 ; set.seed(4444)
for(i in 1:M)
    {
    rmentalmf <- mental[sample(mf),]
    rmentalmi <- mental[sample(mi),]
    rmentalff <- mental[sample(ff),]
    rmentalfi <- mental[sample(fi),]
    rcorrs <- c(
            cor(rmentalmf[,1:3],physmf[,1:3]),
            cor(rmentalmf[,4:6],physmf[,4:6]),
            cor(rmentalmi[,1:3],physmi[,1:3]),
            cor(rmentalmi[,4:6],physmi[,4:6]),
            cor(rmentalff[,1:3],physff[,1:3]),
            cor(rmentalff[,4:6],physff[,4:6]),
            cor(rmentalfi[,1:3],physff[,1:3]),
            cor(rmentalfi[,4:6],physff[,4:6])  )
    rmin <- c(rmin,min(rcorrs))
    rmax <- c(rmax,max(rcorrs))
    rabs <- c(rabs,max(abs(min(rcorrs)),abs(max(rcorrs))))
    }

twot <- length(rabs[rabs>=absobs])/M # Two sided
lowt <- length(rmin[rmin<=obsmin])/M # Lower tailed
upt <- length(rmax[rmax>=obsmax])/M # Upper tailed

merror <- function(phat,M,alpha) # (1-alpha)*100% merror for a proportion
     {
     z <- qnorm(1-alpha/2)
     merror <- z * sqrt(phat*(1-phat)/M)  # M is (Monte Carlo) sample size
     merror
```

```
     } # End function merror

cat("Correlations Between Mental and Physical \n")
cat(" \n") ; cat(" \n")
cat("     Minimum Observed Correlation: ",obsmin,"\n")
cat("     Randomization p-value (one-sided): p-hat = ",lowt," \n")
cat("     Plus or minus 99% Margin of error = ",merror(lowt,M,0.01),"\n")
cat(" \n")
cat("     Maximum Observed Correlation: ",obsmax,"\n")
cat("     Randomization p-value (one-sided): p-hat = ",upt," \n")
cat("     Plus or minus 99% Margin of error = ",merror(upt,M,0.01),"\n")
cat(" \n")
cat("     Maximum Observed Absolute Correlation: ",absobs,"\n")
cat("     Randomization p-value (two-sided): p-hat = ",twot," \n")
cat("     Plus or minus 99% Margin of error = ",merror(twot,M,0.01),"\n")
cat(" \n")
```

11.2 Bootstrap

To appreciate the bootstrap, recall the idea of a *sampling distribution*.

If the sample size is large enough, the histogram of the sample data is a lot like the histogram of the entire population. Thus, sampling from the sample *with replacement* is a lot like sampling from the population. Sampling from the sample is called **resampling**. One can approximate the sampling distribution of a statistic as follows.

- Select a random sample of size n from the sample data, *with replacement*.
- Compute the statistic from the resampled data.
- Do this over and over again, accumulating the values of the statistic.
- A histogram of the values you have accumulated will resemble the sampling distribution of the statistic.

```
> # boot1.R    Working on the bootstrap
> # Run with    R --vanilla < boot1.R > boot1.out &
> # grades.dat has 4 columns: ID, Verbal SAT, Math SAT and 1st year GPA
>
> marks <- read.table("grades.dat")
> n <- length(marks$verbal) #$
> n
[1] 200
> marks[1:10,]
   verbal math gpa
1     623  509 2.6
2     454  471 2.3
3     643  700 2.4
4     585  719 3.0
5     719  710 3.1
6     693  643 2.9
7     571  665 3.1
8     646  719 3.3
9     613  693 2.3
10    655  701 3.3
```

```
> obscorr <- cor(marks)
> obscorr
          verbal      math       gpa
verbal 1.0000000 0.2746341 0.3224477
math   0.2746341 1.0000000 0.1942431
gpa    0.3224477 0.1942431 1.0000000
> # Question: Is the correlation between Verbal SAT and GPA the same as
> # the correlation between math SAT and GPA?
> # What is the sampling distribution of the difference between correlation
> # coefficients?
> #
> obsdiff <- obscorr[3,1]-obscorr[3,2] # Verbal minus math
> obsdiff
[1] 0.1282046
> # The strategy will be to obtain a 95% bootstrap confidence interval for
> # the difference between verbal correlation and math correlation. This
> # confidence interval will be approximately centered around the observed
> # difference obsdiff = .128.  If the confidence interval does not include
> # zero, we will conclude that the observed difference is significantly
> # different from zero.
>
> BOOT <- 1000  ;  bsdiff <- NULL ; set.seed(9999)
> # Accumulate bootstrap values in bsdiff
> # For clarity, do operations in several separate steps inside the loop
> for(i in 1:BOOT)
+     {
+     bootmarks <- marks[sample(1:n,replace=TRUE),] # sample rows with
+                                                   # replacement
+     bcorr <- cor(bootmarks)  # Correlation matrix of bootstrap sample
+     bdiffer <- bcorr[3,1]-bcorr[3,2] # Differencce between correlation
+                                      # coefficients
+     bsdiff <- c(bsdiff,bdiffer) # Add bdiffer to the end of bsdiff
+     } # Next bootstrap sample
> bsdiff <- sort(bsdiff)
> # Now get lower and upper limits of 95% CI
> low <- bsdiff[.025*BOOT] ; up <- bsdiff[.975*BOOT + 1]
```

```
> low ; up
[1] -0.03643594
[1] 0.3032818
> (low+up)/2
[1] 0.1334230
> obsdiff
[1] 0.1282046
> write(bsdiff,"bsdiff.dat") # Maybe for later analysis
> pdf("bsdiff.pdf") # Send graphics output to pdf file
> hist(bsdiff)
```

Bootstrap regression tests Fit the reduced model. Assemble resampled data sets by sampling withy replacement from the residuals, and forming $\widehat{Y}$ plus the residual. Test full vs reduced model each time. Proportion of simulated F statistics at or above observed F is the bootstrap p-value.