Lab 7: Proc GLM and one-way ANOVA
STT 422: Summer, 2004
Vince Melfi

SAS has several procedures for analysis of variance models, including **proc anova**, **proc glm**, **proc varcomp**, and **proc mixed**. We mainly will use **proc glm** and **proc mixed**, which the SAS manual terms the "flagship" procedures for analysis of variance. In this lab we'll learn about **proc glm**, and see learn how to use it to fit one-way analysis of variance models.

Introduction to proc glm

The "glm" in **proc glm** stands for "general linear models." Included in this category are multiple linear regression models and many analysis of variance models. In fact, we'll start by using **proc glm** to fit an ordinary multiple regression model. Here is a description of the data we'll use, which is taken from the SAS manual:

```
 *-------------------Data on Physical Fitness-------------------*
  | These measurements were made on men involved in a physical   |
  | fitness course at N.C.State Univ. The variables are Age      |
  | (years), Weight (kg), Oxygen intake rate (ml per kg body     |
  | weight per minute), time to run 1.5 miles (minutes), heart   |
  | rate while resting, heart rate while running (same time      |
  | Oxygen rate measured), and maximum heart rate recorded while |
  | running.                                                     |
  | ***Certain values of MaxPulse have been changed.             |
  *--------------------------------------------------------------*;
```

The following SAS program reads in the data, fits a regression model using **proc reg** with `Oxygen` as the response and `RunTime` and `Weight` as predictors, and then fits the same model using **proc glm**.[1] Look at the output of both.

```
data oxygen;
  infile 'u:\msu\course\stt\422\summer04\oxy.dat';
  input Age Weight Oxygen RunTime RestPulse RunPulse MaxPulse;

proc reg data = oxygen;
  model Oxygen = RunTime Weight;

proc glm data = oxygen;
  model Oxygen = RunTime Weight;

run;
```

[1] You may be wondering why we didn't use **proc glm** to fit regression models throughout. The **proc reg** procedure is more convenient and more powerful for the usual multiple regression model.

Recall that it is possible to cast the analysis of variance model in the context of multiple regression, by using indicator variables as the predictors. This is the approach taken by **proc glm**. We won't need to be overly concerned with the details of how **proc glm** fits analysis of variance models, but rather with understanding how to specify models and to interpret the output. Later in this lab, though, we'll learn a bit about the regression approach to analysis of variance.

One-way analysis of variance using proc glm

We'll investigate one-way analysis of variance using Example 12.6 from the text. The data give the scores of students on a reading comprehension test. Students were taught using one of three teaching methods, called "basal," "DRTA," and "Strat." The goal is to investigate whether there are differences in scores among the three groups and, if so, what the differences are. First read in the data and look at it.

```
data reading;
  infile 'u\msu\course\stt\422\summer04\reading.dat';
  input group $ score;

proc print data = reading;

run;
```

Next, look at some plots and basic descriptive statistics to investigate the data. The plots are particularly uninformative in this example, in part because the `score` variable has so few possible values. None of the output suggests a big difference between the scores for the three treatments.

```
proc boxplot data = reading;
  plot score * group;

proc gplot data = reading;
  plot score * group;

proc means data = reading;
  var score;
  by group;

run;
```

Next fit a one-way analysis of variance model using **proc glm**. First we must tell SAS which variable is the classification variable, i.e., the variable that indicates the teaching method. Then the model is specified, similar to a regression model. The `means` statement asks SAS to print the sample sizes, means and standard deviations separately for each of the three groups.

```
proc glm data = reading;
   class group;
   model score = group;
   means group;

run;
```

The F test with p-value of 0.3288 confirms the suspicion, based on the plots and descriptive statistics, that there isn't any difference in the mean score for the three treatments.

The regression approach to ANOVA; constraints

There are many ways to write the one-way analysis of variance model as a multiple regression model. For many purposes it doesn't matter which of these is chosen, but it's worth thinking a bit about the various options, since it has an effect on how we interpret the parameters of the model. For the sake of concreteness, consider the reading data set from above, for which the teaching method variable has three levels. Since in our data there are three levels and 22 observations per level, we can write the one-way analysis of variance model as

$$y_{ij} = \mu + \tau_i + \epsilon_{ij}, \quad i = 1, 2, 3; \quad j = 1, \ldots, 22.$$

This model is equivalent to a multiple regression model with predictors x_1, x_2, x_3 which are dummy variables for the treatments. For example, x_1 would be 1 when the data come from treatment 1, and 0 otherwise. Since the model has four parameters, $\mu, \tau_1, \tau_2, \tau_3$ in the anova terminology, $\beta_0, \beta_1, \beta_2, \beta_3$ in the regression notation, but there are only three populations of interest, the model is said to be "overspecified." One can just live with this fact (this is what SAS does when we fit an analysis of variance model using **proc glm**), or one can introduce constraints on the model to reduce the number of parameters.

Setting one parameter to 0

Conceptually, the simplest constraint is to set one of the τ parameters to 0. For simplicity, we'll set $\tau_3 = 0$. In the regression setting this leaves the predictors x_1 and x_2. We'll fit this model in SAS. First we create the dummy variables and view them. Make sure you look at the output of the print statement to see what the variables are. Note that we're calling the new dataset `reading2` to distinguish it from the original data set.

```
data reading2; set reading;
   x1 = (group = 'Basal');
   x2 = (group = 'DRTA');

proc print data = reading2;

run;
```

Now we fit the regression model with x_1 and x_2 as predictors. We're also computing the means of the scores for the three teaching methods for reasons that will be apparent soon.

```
proc reg data = reading2;
   model score = x1 x2;

proc means data = reading2;
   var score;
   by group;

run;
```

Look at the output of **proc reg** carefully. Note that the sums of squares and the F test are exactly the same as we got from **proc glm** above. What about the parameter estimates? Compare them with the separate group means. You'll find that the intercept parameter estimate is equal to the mean of the scores for the third teaching method; the parameter estimate for the slope of x_1 is the difference of the mean of the scores for the first teaching method and the scores for the third teaching method; and the parameter estimate for the slope of x_2 is the difference of the mean of the scores for the second teaching method and the scores for the third teaching method. Symbolically,

$$\begin{aligned} b_0 &= \overline{y}_{3.} \\ b_1 &= \overline{y}_{1.} - \overline{y}_{3.} \\ b_2 &= \overline{y}_{2.} - \overline{y}_{3.}. \end{aligned}$$

The method we've just investigated is equivalent to the method that SAS uses in **proc glm**, although it's hard to discover this from the SAS documentation. The `solution` option in the `model` statement in **proc glm** asks for parameter estimates, as in the following program. Compare the parameter estimates from **proc glm** to the parameter estimates from **proc reg** above.

```
proc glm data = reading;
   class group;
   model score = group / solution;
```

Setting the sum of the parameters to 0

A constraint that many people find more appealing conceptually is to set the sum of the τ parameters to 0: $\sum_{i=1}^{I} \tau_i = 0$. In terms of the analysis of variance model, this makes μ the mean of all the individual groups' means, and makes τ_i the deviation of the ith group's mean from the overall mean.

Let's again consider this in the context of the reading data. If $\sum_{i=1}^{3} \tau_i = 0$, then $\tau_3 = -\tau_1 - \tau_2$. So data from the third group can be represented as

$$y_{3j} = \mu - \tau_1 - \tau_2 + \epsilon_{3j}.$$

In terms of dummy variables, we'll want to set

$$x_1 = \begin{cases} 1 & \text{if the data come from group 1;} \\ 0 & \text{if the data come from group 2;} \\ -1 & \text{if the data come from group 3;} \end{cases}$$

and

$$x_2 = \begin{cases} 0 & \text{if the data come from group 1;} \\ 1 & \text{if the data come from group 2;} \\ -1 & \text{if the data come from group 3;} \end{cases}$$

The SAS program below creates the dummy variables in the data set `reading3`, prints the data, and then fits the regression of y on x_1 and x_2.

```
data reading3; set reading;
   x1 = (group = 'Basal');
   x2 = (group = 'DRTA');
   x3 = (group = 'Strat');
   x1 = x1 - x3;
   x2 = x2 - x3;

proc print data = reading3;

proc reg data = reading3;
   model score = x1 x2;

run;
```

Again, note that the sums of squares and the F test are the same as those from **proc glm**. In this case the parameter estimates are

$$\begin{aligned} b_0 &= (1/3)(\overline{y}_{1.} + \overline{y}_{2.} + \overline{y}_{3.}) \\ b_1 &= \overline{y}_{1.} - \overline{y}_{..} \\ b_2 &= \overline{y}_{2.} - \overline{y}_{..}. \end{aligned}$$