Lab 9: Two-way ANOVA in SAS
STT 422: Summer, 2004
Vince Melfi

So far we have used the **proc glm** procedure to analyze one-way analysis of variance models. But **proc glm** can fit a wide variety of different analysis of variance models. Today we'll begin using it in the context of two-way analysis of variance.

The data

Three different kinds of food (meat, legumes, vegetables) were cooked in three different types of pots (aluminum, clay, iron), and the iron content in milligrams per 100 grams was measured. The first factor (pot type) has $a = 3$ levels, and the second factor (food) has $b = 3$ levels, so there are a total of $ab = 9$ possible treatment combinations. In addition, there were $r = 4$ replications at each of the 9 levels, yielding a total of $n = 36$ observations. The following SAS program reads in the data, prints it, and computes the means for each of the 9 treatment combinations.

```
data iron;
  infile 'u:\msu\course\stt\422\summer04\iron.dat' firstobs = 2 DLM='09'x;
  input obsnum type $ blah1 food $ blah2  blah3 ironcontent;
  drop obsnum blah1 blah2 blah3;

proc print data = iron;

proc means data = iron;
   var ironcontent;
   by type food notsorted;
   output out = ironmeans mean = meanironcontent;

run;
```

Some comments on the SAS program:

1. The `infile` statement includes `firstobs=2` and `DLM='09'x` to tell SAS to skip the first line of the file, which contains the variable names, and to tell SAS that the file is tab-delimited.

2. There are several variables in the file that aren't of interest here, so they are given uninteresting names and `drop`ped from the data set.

3. In the **proc means** procedure it is necessary to use the `notsorted` option since the data aren't sorted by the two factors. In addition, we'll want to plot the means later to construct an interaction plot, so we save the means in a data set called `ironmeans`, in a variable called `meanironcontent`.

Creating an interaction plot

Next we'll use the saved sample cell means to draw an interaction plot. First we tell SAS the symbols (`l, m, v`) we'd like to use for the three levels of the `food` variable, and also tell SAS that we'd like to `join` the symbols. Then we give the plot statement. Note that in the `plot` statement, we specify a scatter plot of `meanironcontent` versus `type` as usual, but then specify `= food` to tell SAS to draw separate scatter plots for the three levels of `food`.

```
symbol1 v = l i = join c = black;
symbol2 v = m i = join c = black;
symbol3 v = v i = join c = black;

proc gplot data = ironmeans;
   plot meanironcontent*type = food /frame;

run;
```

Fitting the ANOVA model

Now we use **proc glm** to fit a two-way ANOVA model to the data. The only new feature is the `*` syntax for indicating an interaction.

```
proc glm data = iron;
  class food type;
  model ironcontent = type food type*food;

run;
quit;
```

This yields the basic ANOVA output, with F statistics for the usual hypotheses.

It's worth pointing out that we can also construct the test statistics using the general F testing procedure that we learned in the context of multiple regression. For example, we next fit a model that does not include an interaction term.

```
proc glm data = iron;
  class food type;
  model ironcontent = type food;

run;
quit;
```

The difference of the error sums of squares for the two models (one with and one without interaction) is the same as the sum of squares for interaction from the original **proc glm** output. Make sure you verify this.

Computing t percentiles

Our text includes common percentiles of the t and other distributions. But it is sometimes necessary to use other percentiles. For example, if we want a single 95% confidence interval for a mean μ (under the appropriate conditions), we need the 0.975 percentile of a t distribution. (One way to formalize this is to let $1-\alpha$ represent the confidence level, which in our case is 0.95, i.e., $\alpha = 0.05$. Then we need the $1-\alpha/2$ percentile, which in our case is $1-0.05/2 = 1-.025 = 0.975$.)

If we want simultaneous confidence intervals for four population means $\mu_1, \mu_2, \mu_3, \mu_4$ using the Bonferroni procedure, then we need the $1-\alpha/(2(4)) = 1-\alpha/8$ percentile. For the case of simultaneous 95% intervals this leads to the $1-0.05/8 = 0.99375$ percentile. We can compute these in SAS using the `tinv` function. The following program computes the 0.975 percentile and the 0.99375 percentile of a t distribution with 15 degrees of freedom.

```
data tpercentiles;
  t = tinv(.975, 15);
  t1 = tinv(.99375, 15);

run;

proc print data = tpercentiles;

run;
```