

**Usage notes: pdmix800.sas macro**  
S. R. Bowley, 2013

Macros are ways of performing a series of common, complex, computations without having to place the statements within a sas editor file. During an analysis, information is passed to the macro, the processes are performed, and the result is then passed back for the remaining steps of the analysis requested by the sas editor statements. Many macros have been created within the sas community; the authors provide the usage notes through comments lines at the beginning of the macro. The description, details, and options for sas macros can be viewed by opening up the sas macro within a sas editor window.

This macro, for use with multiple means comparison from a **Proc Mixed** analysis, was written by Dr. Saxton, University of Tennessee. His research focus is in Animal Science but his contribution is something researchers in all fields have benefited greatly from. The macro converts the matrix of pdiffs values generated from an all-possible pairwise means comparison test into the more digestible *a ab abc...* format for conveying which means differ from which.

The means compared can be either simple main effect means or interaction means. You specify whatever multiple means comparison adjustment you need in the *lsmeans* statement (such as *pdiff adjust=tukey*); the macro takes the SAS pdiff table and compresses the differences into the letter format. There is also a sister macro, called *pdglm800.sas* written for use with a Proc Glm analysis.

One feature of the mixed macro (not available in the glm macro) is you can also include a *slice=* option, which allows you to perform means comparisons of interactions but restrict the comparisons within the levels of one of the factors (ie. Within the rows or within the columns of the interaction means)

Both macros utilize the SAS *output delivery system* (called *ods*). An example in the text illustrates the use of the Proc Mixed macro (see page 136). The statements for *pdglm800* are almost the same but with one important difference. In the two *ods* lines, the variable **diffs** is required for the Proc Mixed version, the variable **diff [no\_s]** is required for the Proc Glm version.

### **Example 1. Comparison of a set of means.**

In Proc Mixed, after the *lsmeans* statement (with the pdiff & adjust options), you add the additional lines required for calling the macro and provide input information for the macro. Note that you can still add all the contrast & estimate statements you need for the Mixed analysis, however, they must be located before the first *run;* statement which is placed before the *% include* statement:

```
ods listing exclude lsmeans diff;
ods output diff=ppp lsmeans=mmm;
*insert other statements such as contrast, estimate, etc;
run;
%include 'e:\sasmacros\pdmix800.sas';
%pdmix800(ppp,mmm,alpha=.05,sort=no);
run;
```

Note that you will need to alter the path listed above in bold to provide the directions to YOUR copy of the pdmix800.sas file. In the example above, I had placed it on a flash drive in drive E in a subdirectory called sasmacros.

The first ods statement (exclude lsmeans diff) turns off the listing of the lsmeans and the table of pairwise differences into the output window. This is not critical to do, but it does reduce the pages of material to wade through.

The second ods statement creates a copy of the (internal) MIXED table of probability of differences as the variable ppp, and the lsmeans as the variable mmm. These two variables are then placed into a new dataset using the output request; we don't bother naming the new dataset. The last two lines take these two variables (probability of differences and the means) and calls the macro and establishes 0.05 as the Type I error to use, and to leave the treatment means in the order that the analysis has them. You can change the Type I error to whatever level you want. A sort=yes results in the treatment means being sorted from high to low.

***Because a new data set is created, if you wish to continue with other analyses after invoking the macro, you need to point to an earlier dataset (eg. Proc plot data=second; ) in subsequent steps.***

## Example 2. Comparison of interaction means

For interaction means, we often want to compare the means within the levels of one of the factors. If one creates a table of interaction means, the rows being the levels of one factor, the columns being the levels of other factor, the comparisons could be within the columns or within the rows (rather than all pairwise combinations).

These comparisons are obtained using the slice= option; the slice can be for either of the interacting factors. The slice= request is required in both the lsmeans statement as well as the call to the pdmix800 macro. In the following example, the interaction means of two factors **a** & **b** are generated and the comparisons are “sliced” into the levels of the second factor

```
lsmeans a*b /pdiff adjust=tukey slice=b;
ods listing exclude lsmeans diff;
ods output diff=ppp lsmeans=mmm;
*insert other statements such as contrast, estimate, etc;
run;
%include 'e:\sasmacros\pdmix800.sas';
%pdmix800(ppp,mmm,alpha=.05,sort=no,slice=b);
run;
```

In addition to subdividing the means comparisons into the slice groups, Proc Mixed will also provide the subdivision of the interaction variance into the levels of the slice=factor. In the case of the above example, the means will be compared within each level of the factor b. It will also subdivide the interaction variance into each level of the factor b and perform a F-test of each subdivision of the variance.

Example 9.5 in the text illustrates the use of the slice option to subdivide a two-factor term (in that example it is was a nested effect) into the levels of the nested factor.