A Macro for Converting Mean Separation Output to Letter Groupings in PROC MIXED

Arnold M. Saxton, University of Tennessee Agricultural Experiment Station
Knoxville, Tennessee

Abstract

This paper describes a SAS® macro, PDMIX612, which takes a set of probability values for testing all pairwise differences among means, and converts this information to letter groups, where means with a common letter are not statistically different at a specified alpha level. The macro is specifically designed to use the output produced by the PDIFF option on the LSMEANS statement in PROC MIXED, but could also be used for PROC GLM output. The converted output from PDMIX612 is more compact and easier to interpret. A further advantage is that the use of letters to indicate differences is widely used, so PDMIX612 output can be copied directly to tables for publication. It requires SAS 6.12, and the SAS/STAT® and SAS/IML® products, but otherwise should run on any platform.

Introduction and Example

Mean separation is a widely used technique for determining where differences among treatment means occur (Zar 1984). When the means all have the same standard error, it is a relatively easy task to rank the means and display groups of means that are statistically similar. When data or the experimental design structure are unbalanced, creating unequal standard errors, ranking the means will not necessarily produce easily identifiable sequences of groups. In such cases the differences among the means are reported by SAS procedures GLM and MIXED as a matrix of pairwise significance probabilities. This paper presents a macro, PDMIX612, which takes this sort of information and converts it to the usual letter grouping. As an example, Table 1 shows a SAS program for a randomized block design with a covariate, analyzed with both GLM and MIXED procedures, and Table 2 gives the output from the PROC GLM MEANS statement.

Table 1. SAS program for an experiment blocked on steer, nested treatments of store and day, and a covariate ired.

```
data one;
  input store $ day $ steer metmb ired;
cards;
   partial    1-4      1      0.150   2.1
   partial    1-4      2      0.185   2.2
   partial    1-4      3      0.200   2.4
   partial    1-4      4      0.237   2.6
   partial    1-4      5      0.208   2.3
   froz5        9      1      0.261   2.6
   froz5        9      2      0.353   2.7
   froz5        9      3      0.199   2.0
   froz5        9      4      0.315   2.7
   froz5        9      5      0.299   2.7
   froz5      180      1      0.207   2.0
   froz5      180      2      0.327   2.8
   froz5      180      3      0.249   2.5
   froz5      180      4      0.276   2.6
   froz5      180      5      0.498   3.0
   froz20      30      1      0.304   2.6
   froz20      30      2      0.370   2.8
   froz20      30      3      0.249   2.6
   froz20      30      4      0.261   2.4
   froz20      30      5      0.481   2.9
   froz20     390      1      0.401   2.8
   froz20     390      2      0.360   2.7
   froz20     390      3      0.401   3.0
   froz20     390      4      0.352   2.6
   froz20     390      5      0.399   2.9
;
proc glm;
  class steer store day;
  model metmb = steer store day(store)
                ired;
  means store  /lsd lines;
  lsmeans store day(store)/stderr pdiff;
run;
proc mixed;
  class steer store day;
  model metmb = store day(store)   ired;
  random  steer;
  lsmeans store day(store)/ pdiff;
run;
```

Table 2. Output from the MEANS statement in PROC GLM.

```
        General Linear Models Procedure

      T tests (LSD) for variable: METMB
NOTE: This test controls the type I com-
parisonwise error rate not the experi-
mentwise error rate.
    Alpha= 0.05  df= 15  MSE= 0.001592
    Critical Value of T= 2.13
    Least Significant Difference= 0.0439
    WARNING: Cell sizes are not equal.
    Harmonic Mean of cell sizes= 7.5

  Means with the same letter are not
  significantly different.

  T Grouping       Mean      N   STORE

     A          0.35780     10  froz20

     B          0.29840     10  froz5

     C          0.19600      5  partial
```

Table 2 shows the most common method of displaying which means differ, namely labeling each mean with letters, and if two means do not have a letter in common, they statistically differ at the chosen significance level (e.g. the default value of 0.05). However, the MEANS statement will not produce such output for any term involving more than one effect, such as day(store). Also, with unbalanced data, the arithmetic averages produced by the MEANS statement are generally not recommended. Therefore, in all but the simplest analyses use of the LSMEANS statement will be necessary.

Table 3 shows the output from the LSMEANS statement in PROC GLM. It prints a matrix of significance probabilities for all pairwise comparisons between means, with element ij of the matrix being the P-value for comparing mean i with mean j.

Table 3. Output from the PROC GLM LSMEANS statement for the example in Table 1.

```
        General Linear Models Procedure
              Least Squares Means

STORE     METMB       Std Err    Pr > |T|     LSMEAN
          LSMEAN      LSMEAN    H0:LSMEAN=0   Number

froz20    0.32885300  0.01396556     0.0001    1
froz5     0.30225960  0.01264321     0.0001    2
partial   0.24617480  0.02064113     0.0001    3

          Pr > |T| H0: LSMEAN(i)=LSMEAN(j)

               i/j     1       2       3
               1   .       0.1839  0.0085
               2  0.1839   .       0.0328
               3  0.0085  0.0328   .

NOTE: To ensure overall protection level, only
probabilities associated with pre-planned com-
parisons should be used.


DAY STORE     METMB     Std Err    Pr>|T|     LSMEAN
              LSMEAN    LSMEAN    H0:LSMEAN=0   Num

30   froz20   0.31756   0.018128    0.0001       1
390  froz20   0.34014   0.019887    0.0001       2
180  froz5    0.31140   0.017845    0.0001       3
9    froz5    0.29312   0.017916    0.0001       4
1-4  partial  0.24617   0.020641    0.0001       5

        Pr > |T| H0: LSMEAN(i)=LSMEAN(j)

     i/j     1       2       3       4       5
     1   .       0.3960  0.8119  0.3564  0.0249
     2  0.3960   .       0.2990  0.1053  0.0096
     3  0.8119  0.2990   .       0.4808  0.0304
     4  0.3564  0.1053  0.4808   .       0.0993
     5  0.0249  0.0096  0.0304  0.0993   .

NOTE: To ensure overall protection level, only
probabilities associated with pre-planned com-
parisons should be used.
```

PROC MIXED reports the same information in a list format (Table 4). For reporting experimental results the letter group format of Table 2 is advantageous in that it is more compact and makes it easier to identify means that are similar or different. Table 5 displays output from the PDMIX612 macro, which converts Table 4 to the letter group format. Note that this output is essentially identical to what LSMEANS produces, except that the variable LSD(.05) has been added, which contains the letters indicating differences among the means. Details on how to use PDMIX612 are in the Usage section below.

Method

The first question which may arise is whether it is even possible, using letter groupings, to represent all possible patterns of similarities and differences among means. In fact it is, because each comparison can be assigned a letter, as shown in Table 6. Comparing mean 1 to mean 2 uses the letter 'a', mean 1 to mean 3 uses 'b', and so on. The presence or absence of each of these letters would indicate that the two means were similar or different, respectively. This contains all the information that is presented by the letter group format. Thus for n means, n*(n-1)/2 letters are always sufficient to display all similarities and differences among the means.

Table 6. Any pattern of differences can be displayed by letting each pairwise comparison have its own letter.

Mean	
1	**abc**
2	**a de**
3	**b d f**
4	**c ef**

However, if all these letters are used, interpretation is not any easier, so the goal of the algorithm is to find the smallest number of groups that still contain all the original information. If mean 1 equals mean 2, mean 2 equals mean 3, and mean 1 equals mean 3, then these three groups should be collapsed into a single group which shows they are all equal to each other. Attempting to do this by hand with a large number of means will quickly show the difficulty.

The remainder of this section will be of interest mainly for working with the actual PDMIX612 macro code, being a brief description of the algorithm in more detail. Each column of the probability matrix (e.g. Table 3) is examined for elements (i,j) greater than alpha, indicating means i and j must be grouped together. As each such element is found, the means are collected into a group member list. Each new mean added to the member list also must not differ from all means already in the list. Additionally, to insure a minimal set of groups, with each mean i and j found to be equal, an inner loop in the algorithm checks all other means m against i and j to see if it can be added to the member list. As the information in each element is coded into the member list, the element is made negative to flag it as already processed.

The minimal groups found by this algorithm are stored in an integer matrix (GROUP) with a row for each mean, and the contents of the row being a list of group identifiers that the mean belongs to. The last portion of code then converts these numbers to letters.

Table 4. Output from the PROC MIXED LSMEANS statement.

Least Squares Means

Effect	STORE	DAY	LSMEAN	Std Error	DF	t	Pr > |t|
STORE	froz20		0.32484471	0.01537295	15	21.13	0.0001
STORE	froz5		0.30279404	0.01437132	15	21.07	0.0001
STORE	partial		0.25312250	0.02143094	15	11.81	0.0001
DAY(STORE)	froz20	30	0.31542385	0.01941088	15	16.25	0.0001
DAY(STORE)	froz20	390	0.33426558	0.02081831	15	16.06	0.0001
DAY(STORE)	froz5	180	0.31140000	0.01918736	15	16.23	0.0001
DAY(STORE)	froz5	9	0.29418808	0.01924348	15	15.29	0.0001
DAY(STORE)	partial	1-4	0.25312250	0.02143094	15	11.81	0.0001

Differences of Least Squares Means

Effect	STORE	DAY	_STORE	_DAY	Difference	Std Error	DF	t	Pr > |t|
STORE	froz20		froz5		0.02205067	0.01905984	15	1.16	0.2654
STORE	froz20		partial		0.07172221	0.02670374	15	2.69	0.0169
STORE	froz5		partial		0.04967154	0.02375135	15	2.09	0.0539
DAY(STORE)	froz20	30	froz20	390	-0.01884173	0.02598189	15	-0.73	0.4795
DAY(STORE)	froz20	30	froz5	180	0.00402385	0.02563715	15	0.16	0.8774
DAY(STORE)	froz20	30	froz5	9	0.02123577	0.02584663	15	0.82	0.4242
DAY(STORE)	froz20	30	partial	1-4	0.06230135	0.02836326	15	2.20	0.0442
DAY(STORE)	froz20	390	froz5	180	0.02286558	0.02671859	15	0.86	0.4056
DAY(STORE)	froz20	390	froz5	9	0.04007750	0.02719865	15	1.47	0.1613
DAY(STORE)	froz20	390	partial	1-4	0.08114308	0.03097151	15	2.62	0.0193
DAY(STORE)	froz5	180	froz5	9	0.01721192	0.02551064	15	0.67	0.5101
DAY(STORE)	froz5	180	partial	1-4	0.05827750	0.02719865	15	2.14	0.0490
DAY(STORE)	froz5	9	partial	1-4	0.04106558	0.02671859	15	1.54	0.1451

Table 5. Converted output from PDMIX612 for the data in Table 4. A variable containing letter groups is added to the PROC MIXED default output. The variable name is based on the ADJUST= option, the default being LSD mean separation. The alpha value is included in the label.

BYGROUP=1 Effect=STORE

OBS	STORE	DAY	LSMEAN	Std Error	DF	t	Pr > |t|	LSD(0.05)
1	froz20		0.32484471	0.01537295	15	21.13	0.0001	A
2	froz5		0.30279404	0.01437132	15	21.07	0.0001	AB
3	partial		0.25312250	0.02143094	15	11.81	0.0001	B

BYGROUP=2 Effect=DAY(STORE)

OBS	STORE	DAY	LSMEAN	Std Error	DF	t	Pr > |t|	LSD(0.05)
4	froz20	30	0.31542385	0.01941088	15	16.25	0.0001	A
5	froz20	390	0.33426558	0.02081831	15	16.06	0.0001	A
6	froz5	180	0.31140000	0.01918736	15	16.23	0.0001	A
7	froz5	9	0.29418808	0.01924348	15	15.29	0.0001	AB
8	partial	1-4	0.25312250	0.02143094	15	11.81	0.0001	B

Usage

The algorithm has been coded in the IML procedure's matrix language, and for the convenience of MIXED users, incorporated into a macro which is easily called by the following supplemental code.

```
proc mixed;
        class ….;
        model ….;
        lsmeans …./pdiff;
        make 'lsmeans' out=MMM noprint;
        make 'diffs' out=PPP noprint;
run;
%include 'A:PDMIX612.SAS';
%pdmix612(PPP,MMM);
```

Four statements are added to the basic PROC MIXED program which produces the LSMEANS output. There are two MAKE statements to save the means and pairwise differences among them into SAS data sets MMM and PPP respectively (these names can be changed). The noprint option turns off printing of LSMEANS output by PROC MIXED, and should generally be used since PDMIX612 will print out the same information. The other two statements are macro statements used to run the macro on the data sets PPP and MMM. The %INCLUDE makes the macro available to the program, with syntax being simply the complete path and file name of the macro in quotes. This statement is not required if you use the AUTOCALL facility. The second statement actually executes the macro. Complete syntax is

```
%pdmix612(DIFFSDATASET, LSMEANSDATASET,
            alpha=.05, sort=NO, worksize=1)
```

where the first two arguments are the data sets from the MAKE statements, are order dependent and required. The remaining arguments are optional (default values are shown), not order dependent but require the lowercase keyword to be specified. The alpha value is the significance level for deciding if means differ. The sort option is either YES, in which case the means are printed in descending order of least square mean value, or any other value, in which case the default sort order in PROC MIXED is used. Note that regardless of sort order, the letters are assigned in order, with the numerically largest means assigned the letter A, and the last letter used assigned to the smallest valued means. The worksize option changes the memory made available to PROC IML. For extremely large numbers of means this option may be useful, and you simply specify the number of kilobytes PROC IML should have access to.

Discussion

A disadvantage of the letter grouping format is loss of information on the level of significance, as only "equal" or "not equal" information at some alpha level can be displayed. However, its compact format is very useful for large experiments. Consider an agricultural experiment testing 100 varieties, a not uncommon event. There are 100*(100-1)/2 or 4950 probability values, which at 50 per page would take 99 pages. Output from PDMIX612 would take 2 pages, 50 means with letters per page.

When there are a large number of means being compared, it is possible for the number of groups to exceed 26 (letters in the alphabet). In this case PDMIX612 creates sections of letters, allowing letters to be reused. For example, output like Table 7 might result, and shows three means that all differ, because they have no letters in common. The letter A for mean 1 is a different letter from the A in section (2) for mean 2, and both differ from mean 3, with letters in section (3).

Table 7. Example of "sections" of letters used for more than 26 groups. The notation (#) starts a new section.

Mean	Letter Group
1	A(2)B(3)C
2	(2)A
3	(3)AB

For PROC GLM users, since there is no way to selectively put just the PDIFF option output into a file during program execution, PROC GLM would need to be run, and then the matrix of P-values copied by hand into the IML code, or a program written to read the output and create the required files. I discourage the use of PROC GLM, because of differences in means and std. errors as compared to PROC MIXED (compare Tables 3 and 5).

The variable names in the MAKE data sets that PROC MIXED creates changed from Version 6.11 to 6.12, so this macro is version specific. A 6.11 version is available, but does not have the same capabilities. The macro code pdmix612.sas can be obtained by anonymous FTP from the server SCS1.AG.UTK.EDU.

An effort has been made to insure that PDMIX612 handles all situations. For example, BY processing has been programmed, and if the ADJUST= option on the LSMEANS statement is used, the adjusted probabilities are used. ADJUST=Dunnett is not supported because it does not do all pairwise comparisons. I would be happy to correct any errors brought to my attention, and welcome suggestions for improvement. A final caution, capabilities provided by PDMIX612 do not imply that mean separations should be done under all circumstances.

References

SAS Institute. 1989. SAS/IML Software: Usage and Reference, Version 6. Cary, NC.

SAS Institute. 1997. SAS/STAT Software: Changes and Enhancements through Release 6.12. Cary, NC.

Zar, J. H. 1984. Biostatistical Analysis, 2nd Ed. Englewood Cliffs: Prentice Hall.

Arnold M. Saxton
225 Morgan Hall
University of Tennessee
Knoxville, TN 37996-4500
Voice: 423-974-7189
Fax: 423-974-7335
Email: asaxton@utk.edu