USING SAS SOFTWARE TO PRODUCE SUMMARY TABLES OF GLM-CALCULATED STATISTICS
Margaret A. Grady
Northrop Services, Inc.

Although some of the results (predicted values, residuals) from the GLM procedure are available for further analysis, other statistics such as the MSE, R², and least squares means cannot be accessed directly. In the preparation of reports of statistical analyses of datasets, tables summarizing the analysis of variance results, means, and least squares means can be cumbersome to transcribe. Copying errors can be introduced, particularly if the tables must be retyped for inclusion in a published report.

Using the programming capabilities of the SAS software, the desired statistics are read from the output of an analysis run and printed out in formatted tables. These data can also be introduced, particularly if the tables must be printed for inclusion in a published report.

INTRODUCTION

Production of formatted tables of GLM-calculated statistics requires two SAS software programs. In the first program, the analysis of the data is performed. Any information that will be used in the tables must be included in the variables and processed together using TITLE statements. The output of the analysis must then be stored as an OS dataset. This can be done using WYLBUR or using the SAS procedure PRINTTO. The second SAS software program includes the code to read the analysis program as an input dataset, to print tables of selected data, and, optionally, to output data to disk for further analysis. To run this program, the user must modify the code to include certain information specific to each run. In the attached program, it was necessary to identify the number of effects in the analysis and output the percentage increase can be rather large.

DESCRIPTION OF THE PROGRAM

1. Analysis of the Data

There are several ways to write the analysis program depending on the format desired for the final tables. In this example, the data are divided into groups based on pre-defined criteria, identified using TITLE statements, and analyzed in two programs. In the first program, the output file is output to disk, as the input file created with this procedure will have the attribute RECFM=VBA. Using PRINTTO increases the CPU time required by the job. Additionally, because PRINTTO routes the job output to a permanent file rather than the print file, a separate PRINT statement is required to produce output of the run. As can be seen in Table 1, although the actual CPU times are not substantially different among the methods for producing the analysis and output, the percentage increase can be rather large.

<table>
<thead>
<tr>
<th>Analysis 1</th>
<th>Analysis 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output and File</td>
<td>Method of Producing</td>
</tr>
<tr>
<td>WYLBUR¹</td>
<td>10.26</td>
</tr>
<tr>
<td>PRINTTO²</td>
<td>12.30</td>
</tr>
<tr>
<td>PRINTTO plus PRINT</td>
<td>13.39</td>
</tr>
</tbody>
</table>

¹This does not include the cost of a WYLBUR session to fetch and save the job.
²Using PRINTTO alone will not produce a printout of the analysis.

2. Creating the Formatted Tables Input File from the Analyzed Data

The analysis jobs are run batch under WYLBUR and fetched upon completion. Once it is determined that the run was successful, files are created using the following characteristics RCCL=133 and RECFM=FM which allows for retention of carriage control characters, and the job is printed. Several lines from the file created using this analysis run are listed in Figure 1. If a facility for fetching jobs is not available, the analysis jobs can be run using the SAS procedure PRINTTO. The file created with this procedure will have the attribute RECFM=VBA. Using PRINTTO increases the CPU time required by the job. Additionally, because PRINTTO routes the job output to a permanent file rather than the print file, a separate PRINT statement is required to produce output of the run.

3. Submitting the Formatted Table Run

This run reads in the analysis program which has been output to disk, as the input dataset, produces formatted tables of selected statistics, and can be used to output these statistics to disk if desired. Although there is a core of the program that does not change, there are parts of the code that must be modified depending on the data being analyzed. When the means are ready, the alphanumeric values of the BY variable(s) are assigned numeric values in one (or more) countervariable(s). Reading in the data contained in the analysis of variance is dependent on defining the number of effects in the analysis. In this example, the means and analysis of
variance statistics are stored in different datasets and must be read in separate data steps. The code to read in the means is listed in Figure 4. Three TITLE statements were used. The code to read in the titles (line 90) would need to be changed if more or less titles were used. Similarly, the code which reads in the treatment values, that is the "BY" variables in the MEANS procedure must be specified explicitly for each run (lines 170, 190-210). No other changes are required.

The code to read in the analysis of variance statistics is listed in Figure 5. This code requires only that the user identify the number of independent effects in the ANOVA. In the example, these are listed as the character "?" and changed to the correct number using the WYLOUR Editor. Since, in this example, the data are read in from two files, any changes in how the TITLE statements are read in must be made in this section of code as well as in that used in the means.

These are the only modifications necessary to run the job. Once the data are read in, various counter and flag variables are created and the three datasets containing the means, least squares means, and analysis of variance statistics are merged into one dataset. This dataset is then printed using the programming capabilities of the SAS software. The code used to generate the tables in Figure 7 is listed in Figure 6.

4. The Formatted Tables

The means, least squares means, and analysis of variance statistics are printed on separate pages. The formatted tables using the data described above appear as Figure 7.

EXTENSIONS

The data retrieved from the analysis run may be written to disk for further use if needed. Formatting this file may depend on subsequent use, but one method is to output the data as a hierarchical file with several types of records for each variable. Data output using the programs described in this paper were stored with four record types: general information about the variable (name, labels, missing values, flags, and sort flags), means (including standard error and sample size as well as identifying information), ANOVA statistics, and least squares means.

The key to producing formatted tables from a SAS analysis run is understanding how the SAS print file is formatted and either finding or creating flag variables to search for the desired information. Using these concepts, similar programs accessing data from other procedures can easily be written.

### Figure 2
Analysis Run: Analyses of Variance and Least Squares Means

#### Formatted Table Example
**DAT ANOVAS**
**GROUP 4**

**General Linear Models Procedure**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Source</th>
<th>DF</th>
<th>Sum of Squares</th>
<th>Mean Square</th>
<th>F Value</th>
<th>Pr &gt; F</th>
<th>R-Square</th>
<th>C.V.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Model</td>
<td>3</td>
<td>4.3468377</td>
<td>1.44663292</td>
<td>0.75</td>
<td>0.001</td>
<td>0.0001</td>
<td>0.272696</td>
</tr>
<tr>
<td></td>
<td>Error</td>
<td>70</td>
<td>0.16593108</td>
<td>0.16593108</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Corrected Total</td>
<td>73</td>
<td>15.9564459</td>
<td>0.4401656</td>
<td>6.1727027</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>DF</th>
<th>TYPE I SS</th>
<th>F VALUE</th>
<th>Pr &gt; F</th>
<th>DF</th>
<th>TYPE III SS</th>
<th>F VALUE</th>
<th>Pr &gt; F</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BWT</td>
<td>1</td>
<td>1.88707556</td>
<td>23.47</td>
<td>0.0001</td>
<td>1</td>
<td>4.00034617</td>
<td>24.17</td>
<td>0.0001</td>
</tr>
<tr>
<td>TREAT</td>
<td>2</td>
<td>0.49903358</td>
<td>1.39</td>
<td>0.2566</td>
<td>2</td>
<td>0.05701620</td>
<td>1.39</td>
<td>0.2566</td>
</tr>
</tbody>
</table>

#### Formatted Table Example
**DAT ANOVAS**
**GROUP 4**

**General Linear Models Procedure**

**Least Squares Means**

| Treat | Var 1 | Std. Err | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Mean | Std. Levene Mean | Var 1\(\) | Std. Levene Mean | Prob > | Levene Me
Figure 3
Partial Listing of the Sequential Dataset
Created From the Analysis Run Shown
In Figure 2

FORMATTED TABLE EXAMPLE
BUT ANOVAS
** GROUP 4 **

GENERAL LINEAR MODELS PROCEDURE
DEPENDENT VARIABLE: VARI2

<table>
<thead>
<tr>
<th>SOURCE</th>
<th>DF</th>
<th>SUM OF SQUARES</th>
<th>MEAN SQUARE</th>
<th>F VALUE</th>
<th>PR &gt; F</th>
<th>R-SQUARE</th>
<th>C.V</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODEL</td>
<td>1</td>
<td>4.34580877</td>
<td>1.4460292</td>
<td>8.75</td>
<td>0.001</td>
<td>0.27266</td>
<td>6.59</td>
</tr>
<tr>
<td>ERROR</td>
<td>74</td>
<td>11.59567582</td>
<td>0.16586188</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CORRECTED TOTAL</td>
<td>75</td>
<td>15.93646459</td>
<td>0.4069156</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SOURCE</th>
<th>DF</th>
<th>TYPE I SS</th>
<th>F VALUE</th>
<th>PR &gt; F</th>
<th>TYPE III SS</th>
<th>F VALUE</th>
<th>PR &gt; F</th>
</tr>
</thead>
<tbody>
<tr>
<td>BHY</td>
<td>2</td>
<td>3.68579058</td>
<td>23.47</td>
<td>0.0001</td>
<td>4.9202917</td>
<td>24.17</td>
<td>0.000</td>
</tr>
<tr>
<td>TREAT</td>
<td>2</td>
<td>0.45518268</td>
<td>1.39</td>
<td>0.2568</td>
<td>0.4591800</td>
<td>1.39</td>
<td>0.256</td>
</tr>
</tbody>
</table>

GENERAL LINEAR MODELS PROCEDURE
LEAST SQUARES MEANS

<table>
<thead>
<tr>
<th>TREAT</th>
<th>VARI2</th>
<th>STD ERR</th>
<th>PROB &gt; [T]</th>
</tr>
</thead>
<tbody>
<tr>
<td>WI-1</td>
<td>1.64659927</td>
<td>0.00306997</td>
<td>0.0001</td>
</tr>
<tr>
<td>WI-1.5</td>
<td>1.57268340</td>
<td>0.01452396</td>
<td>0.0001</td>
</tr>
<tr>
<td>WI-2</td>
<td>1.51629055</td>
<td>0.01240093</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TREAT</th>
<th>VARI2</th>
<th>STD ERR</th>
<th>PROB &gt; [T]</th>
</tr>
</thead>
<tbody>
<tr>
<td>WI-1</td>
<td>6.75359624</td>
<td>0.05575094</td>
<td>0.0001</td>
</tr>
<tr>
<td>WI-1.5</td>
<td>6.55049999</td>
<td>0.05264909</td>
<td>0.0001</td>
</tr>
<tr>
<td>WI-2</td>
<td>6.89715316</td>
<td>0.07582752</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

Because the line printer will only print 132 columns, the information in column 133 cannot be shown.

Figure 4
Formatted Tables Run: Inputting the Means

10. /*AMS DD OSH=MAG=DFSH,OUT=DFSH*
20. /*AMS DD OSH=MAG=DFSH,OUT=DFSH*
30. /*AMS DD OSH=MAG=DFSH,OUT=DFSH*
40. DATA NEW; INFILE MNS MISSO'RE;
50. LENGTH TITLE2 $20.
60. RETAIN TITLE1 TITLE3 TREAT TRT.
70. LOOP1: INPUT PAGET EST 101:
80. IF PAGETEST=1 THEN DO.
90. INPUT TITLE $2-110 / TITLE $2-97 / NAME $2-9.
100. IF TITLE3='** ANIMAL DESCRIPTORS **' THEN DELETE;
110. IF NAME='VARIABLE' THEN DO.
120. LOOP2: INPUT PAGETEST 131:
130. IF PAGETEST=1 THEN GOTO LOOP2;
140. IF DEPVAR='** THEN DO; INPUT; END;
150. IF DEPVAR='** THEN GOTO LOOPS;
160. IF DEPVAR='**' THEN DO;
170. INPUT TREAT $68-73;
180. VCNT=0;
190. TRT=1;
200. IF TREAT='WI-0.5' THEN TRT=2; IF TREAT='WI-1.0' THEN TRT=3;
210. IF TREAT='WI-2.0' THEN TRT=4;
220. END;
230. IF DEPVAR NE '** AND DEPVAR NE '---------' THEN DO;
240. INPUT 310 N 2, 204 MEAN 11.6 ARE STDR 10.61;
250. INPUT 310 N 2, 204 MEAN 11.6 ARE STDR 10.61;
260. VCNT=1; IDST=1; TITLE2='MEANS';
270. OUTPUTS;
280. END;
290. END;
300. CARDS;
310. 1
320. 1

670
Figure 5

Formatted Tables Run: Inputting the ANOVA Statistics

18. DATA _NULL_; SET ANOVA; BY IDS TOHT VCHT TRT;
23. FILE PRINT NAMES TITLES3;
32. ARRAY SOURCE (1) $ 20 SOURCE-TITLE$1; ARRAY NAME1 $2-20 NAME2 $2-20 NAME3 $2-20 NAME4 $2-20;
35. IF IDS=1 THEN DO;
40. IF IDS=1 THEN DO;
45. IF IDS=1 THEN DO;
50. IF IDS=1 THEN DO;
55. IF IDS=1 THEN DO;
60. IF IDS=1 THEN DO;
65. IF IDS=1 THEN DO;
70. IF IDS=1 THEN DO;
75. IF IDS=1 THEN DO;
80. IF IDS=1 THEN DO;
85. IF IDS=1 THEN DO;
90. IF IDS=1 THEN DO;
95. IF IDS=1 THEN DO;
100. IF IDS=1 THEN DO;
105. IF IDS=1 THEN DO;
110. IF IDS=1 THEN DO;
115. IF IDS=1 THEN DO;
120. IF IDS=1 THEN DO;
125. IF IDS=1 THEN DO;
130. IF IDS=1 THEN DO;
135. IF IDS=1 THEN DO;
140. IF IDS=1 THEN DO;
145. IF IDS=1 THEN DO;
150. IF IDS=1 THEN DO;
155. IF IDS=1 THEN DO;
160. IF IDS=1 THEN DO;
165. IF IDS=1 THEN DO;
170. IF IDS=1 THEN DO;
175. IF IDS=1 THEN DO;
180. IF IDS=1 THEN DO;
185. IF IDS=1 THEN DO;
190. IF IDS=1 THEN DO;
195. IF IDS=1 THEN DO;
200. IF IDS=1 THEN DO;
205. IF IDS=1 THEN DO;
210. IF IDS=1 THEN DO;
215. IF IDS=1 THEN DO;
220. IF IDS=1 THEN DO;
225. IF IDS=1 THEN DO;
230. IF IDS=1 THEN DO;
235. IF IDS=1 THEN DO;
240. IF IDS=1 THEN DO;
245. IF IDS=1 THEN DO;
250. IF IDS=1 THEN DO;
255. IF IDS=1 THEN DO;
260. IF IDS=1 THEN DO;
265. IF IDS=1 THEN DO;
270. IF IDS=1 THEN DO;
275. IF IDS=1 THEN DO;
280. IF IDS=1 THEN DO;

Figure 6

Formatted Tables Run: Outputting the Tables

19. DATA _NULL_; SET ANOVA; BY IDS TOHT VCHT TRT;
24. FILE PRINT NAMES TITLES3;
33. ARRAY SOURCE (1) $ 20 SOURCE-TITLE$1; ARRAY NAME1 $2-20 NAME2 $2-20 NAME3 $2-20 NAME4 $2-20;
38. IF IDS=1 THEN DO;
43. IF IDS=1 THEN DO;
48. IF IDS=1 THEN DO;
53. IF IDS=1 THEN DO;
58. IF IDS=1 THEN DO;
63. IF IDS=1 THEN DO;
68. IF IDS=1 THEN DO;
73. IF IDS=1 THEN DO;
78. IF IDS=1 THEN DO;
83. IF IDS=1 THEN DO;
88. IF IDS=1 THEN DO;
93. IF IDS=1 THEN DO;
98. IF IDS=1 THEN DO;
103. IF IDS=1 THEN DO;
108. IF IDS=1 THEN DO;
113. IF IDS=1 THEN DO;
118. IF IDS=1 THEN DO;
123. IF IDS=1 THEN DO;
128. IF IDS=1 THEN DO;
133. IF IDS=1 THEN DO;
138. IF IDS=1 THEN DO;
143. IF IDS=1 THEN DO;
148. IF IDS=1 THEN DO;
153. IF IDS=1 THEN DO;
158. IF IDS=1 THEN DO;
163. IF IDS=1 THEN DO;
168. IF IDS=1 THEN DO;
173. IF IDS=1 THEN DO;
178. IF IDS=1 THEN DO;
183. IF IDS=1 THEN DO;
188. IF IDS=1 THEN DO;
193. IF IDS=1 THEN DO;
198. IF IDS=1 THEN DO;
203. IF IDS=1 THEN DO;
208. IF IDS=1 THEN DO;
213. IF IDS=1 THEN DO;
218. IF IDS=1 THEN DO;
223. IF IDS=1 THEN DO;
228. IF IDS=1 THEN DO;
233. IF IDS=1 THEN DO;
238. IF IDS=1 THEN DO;
243. IF IDS=1 THEN DO;
248. IF IDS=1 THEN DO;
253. IF IDS=1 THEN DO;
258. IF IDS=1 THEN DO;
263. IF IDS=1 THEN DO;
268. IF IDS=1 THEN DO;
273. IF IDS=1 THEN DO;
278. IF IDS=1 THEN DO;
283. IF IDS=1 THEN DO;
288. IF IDS=1 THEN DO;
293. IF IDS=1 THEN DO;
298. IF IDS=1 THEN DO;
303. IF IDS=1 THEN DO;
308. IF IDS=1 THEN DO;
313. IF IDS=1 THEN DO;
318. IF IDS=1 THEN DO;
323. IF IDS=1 THEN DO;
328. IF IDS=1 THEN DO;
333. IF IDS=1 THEN DO;
338. IF IDS=1 THEN DO;
343. IF IDS=1 THEN DO;
348. IF IDS=1 THEN DO;
353. IF IDS=1 THEN DO;
358. IF IDS=1 THEN DO;
363. IF IDS=1 THEN DO;
368. IF IDS=1 THEN DO;
373. IF IDS=1 THEN DO;
378. IF IDS=1 THEN DO;
383. IF IDS=1 THEN DO;
388. IF IDS=1 THEN DO;
393. IF IDS=1 THEN DO;
398. IF IDS=1 THEN DO;
403. IF IDS=1 THEN DO;
408. IF IDS=1 THEN DO;
### FORMATTED TABLE EXAMPLE

#### MEANS

<table>
<thead>
<tr>
<th></th>
<th>W1-1.5</th>
<th>W1-1.1</th>
<th>W1-2.1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GROUP 2</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR22</td>
<td>1.6253 + 0.0090 (17)</td>
<td>1.7037 + 0.0366 (37)</td>
<td>1.7907 + 0.0513 (20)</td>
</tr>
<tr>
<td>VAR20</td>
<td>-0.2279 + 1.3631 (17)</td>
<td>-0.15 + 0.7646 (37)</td>
<td>-0.25 + 0.9717 (20)</td>
</tr>
<tr>
<td>VAR21</td>
<td>-2.0444 + 1.3112 (17)</td>
<td>-2.475 + 0.6964 (37)</td>
<td>-0.48 + 0.2954 (20)</td>
</tr>
<tr>
<td><strong>GROUP 3</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR33</td>
<td>1.5717 + 0.0623 (17)</td>
<td>1.6639 + 0.0721 (37)</td>
<td>1.5199 + 0.0810 (20)</td>
</tr>
<tr>
<td>VAR35</td>
<td>6.4246 + 0.0570 (17)</td>
<td>6.7569 + 0.0579 (37)</td>
<td>6.8611 + 0.0872 (20)</td>
</tr>
<tr>
<td>VAR32</td>
<td>7.2541 + 0.1121 (17)</td>
<td>7.4146 + 0.0669 (37)</td>
<td>7.6546 + 0.0996 (20)</td>
</tr>
<tr>
<td><strong>GROUP 4</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR13</td>
<td>1.5717 + 0.0623 (17)</td>
<td>1.6639 + 0.0721 (37)</td>
<td>1.5199 + 0.0810 (20)</td>
</tr>
<tr>
<td>VAR19</td>
<td>6.6616 + 0.1153 (17)</td>
<td>6.7721 + 0.0711 (37)</td>
<td>6.8534 + 0.1143 (20)</td>
</tr>
<tr>
<td>VAR11</td>
<td>7.2853 + 0.1299 (17)</td>
<td>7.4595 + 0.0706 (37)</td>
<td>7.4888 + 0.1285 (20)</td>
</tr>
<tr>
<td>VAR12</td>
<td>6.0522 + 0.0973 (17)</td>
<td>6.1887 + 0.0706 (37)</td>
<td>6.2328 + 0.1290 (20)</td>
</tr>
</tbody>
</table>

#### LEAST SQUARES MEANS

<table>
<thead>
<tr>
<th></th>
<th>W1-1.5</th>
<th>W1-1.1</th>
<th>W1-2.1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GROUP 2</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR22</td>
<td>1.6213 + 0.0469</td>
<td>1.7043 + 0.0318</td>
<td>1.7971 + 0.0494</td>
</tr>
<tr>
<td>VAR20</td>
<td>-0.2279 + 1.046</td>
<td>-0.15 + 0.0554</td>
<td>-0.24 + 1.1150</td>
</tr>
<tr>
<td>VAR21</td>
<td>-2.0444 + 1.816</td>
<td>-1.379 + 0.902</td>
<td>-2.390 + 0.9405</td>
</tr>
<tr>
<td><strong>GROUP 3</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR33</td>
<td>1.5727 + 0.0618</td>
<td>1.6446 + 0.0284</td>
<td>1.5193 + 0.0537</td>
</tr>
<tr>
<td>VAR35</td>
<td>6.4246 + 0.0724</td>
<td>6.7569 + 0.0541</td>
<td>6.8534 + 0.0969</td>
</tr>
<tr>
<td>VAR32</td>
<td>7.2541 + 0.0937</td>
<td>7.4146 + 0.0365</td>
<td>7.6546 + 0.0966</td>
</tr>
<tr>
<td><strong>GROUP 4</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR13</td>
<td>1.5727 + 0.0918</td>
<td>1.6446 + 0.0284</td>
<td>1.5193 + 0.0537</td>
</tr>
<tr>
<td>VAR19</td>
<td>6.6616 + 0.0822</td>
<td>6.7519 + 0.0557</td>
<td>6.8572 + 0.0750</td>
</tr>
<tr>
<td>VAR11</td>
<td>7.2853 + 0.1038</td>
<td>7.4595 + 0.0706</td>
<td>7.4888 + 0.0995</td>
</tr>
<tr>
<td>VAR12</td>
<td>6.0522 + 0.0987</td>
<td>6.1887 + 0.0467</td>
<td>6.2328 + 0.0912</td>
</tr>
</tbody>
</table>

#### FORMATTED TABLE EXAMPLE

##### BWT ANOVA

<table>
<thead>
<tr>
<th></th>
<th>F</th>
<th>P</th>
<th>R2</th>
<th>MSE</th>
<th>DF</th>
<th>F</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GROUP 2</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR22</td>
<td>3.76</td>
<td>0.0156</td>
<td>0.16</td>
<td>0.0374</td>
<td>0.06</td>
<td>4.13</td>
<td>0.05</td>
</tr>
<tr>
<td>VAR20</td>
<td>2.66</td>
<td>0.0537</td>
<td>0.13</td>
<td>26.9956</td>
<td>0.07</td>
<td>4.59</td>
<td>0.04</td>
</tr>
<tr>
<td>VAR21</td>
<td>0.02</td>
<td>0.9915</td>
<td>0.00</td>
<td>17.5976</td>
<td>0.01</td>
<td>0.01</td>
<td>0.91</td>
</tr>
<tr>
<td><strong>GROUP 3</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR33</td>
<td>3.06</td>
<td>0.0332</td>
<td>0.12</td>
<td>0.0279</td>
<td>0.06</td>
<td>1.71</td>
<td>0.19</td>
</tr>
<tr>
<td>VAR35</td>
<td>14.76</td>
<td>0.0001</td>
<td>0.39</td>
<td>0.0072</td>
<td>0.00</td>
<td>39.25</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>VAR32</td>
<td>7.01</td>
<td>0.0001</td>
<td>0.25</td>
<td>0.1493</td>
<td>0.01</td>
<td>10.09</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td><strong>GROUP 4</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR13</td>
<td>3.06</td>
<td>0.0332</td>
<td>0.12</td>
<td>0.0279</td>
<td>0.06</td>
<td>1.71</td>
<td>0.19</td>
</tr>
<tr>
<td>VAR19</td>
<td>22.20</td>
<td>0.0001</td>
<td>0.44</td>
<td>0.1437</td>
<td>0.01</td>
<td>63.67</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>VAR11</td>
<td>12.11</td>
<td>0.0001</td>
<td>0.34</td>
<td>0.1894</td>
<td>0.00</td>
<td>34.74</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>VAR12</td>
<td>8.75</td>
<td>0.0001</td>
<td>0.27</td>
<td>0.1658</td>
<td>0.00</td>
<td>20.17</td>
<td>&lt;0.01</td>
</tr>
</tbody>
</table>