TABULATING THE RESULTS: AN APPLICATION OF SAS® SOFTWARE TO PERIODIC REPORTING IN A LONGITUDINAL EPIDEMIOLOGIC STUDY

William C. Murphy, Howard M. Proskin, and Dennis H. Leverett
Eastman Dental Center

Abstract

Reports consisting of dozens of similar tables are required for the periodic monitoring of an ongoing epidemiologic survey. Data gathered in the field are regularly assembled into the SAS system for study and analysis. PROC TABULATE, with its various options, is ideal for producing the tables needed for the periodic reporting of study results. The use of the extended ASCII character set on a PC allows the alteration of the FORMCHAR option in order to generate publication-quality tables on a laser printer. The use of formatting, and both LABEL and KEYLABEL statements, enables the further customization of tables. Furthermore, the use of the SAS macro language facilitates the automation of table production. Macros routines are also useful for generating indexes, and for providing date-stamps on the tables.

Introduction

In a large-scale multifaceted longitudinal epidemiologic study, a database may be accumulated from a variety of input sources over a span of several years' time. The periodic interim reporting of results to date will often include an enumeration of what data has been incorporated thus far, as well as the preparation of numerous summary tables.

As an example of such a study, the authors are participating in a longitudinal investigation of the oral health of senior adults, (initially) aged 54 through 77. Study subjects are examined periodically to obtain dental, periodontal, and oral soft-tissue health assessments. A denture examination is also conducted, when necessary. Furthermore, a questionnaire on life style and general health is also administered. Each aspect of these exams is entered using separate case-report forms, and is entered into separate respective databases.

Tables

The SAS system can readily be used to make tables from our datasets using PROC TABULATE. One of the first tables we wish to make involves checking what exam each subject received. To do this we write

```sas
proc tabulate;
class idno;
var t1 t2 t3 t4 t5 count;
table idno, (t1 t2 t3 t4 t5 count)*sum;
```

where the variables t1 t2 t3 t4 t5 are either 1 or 0. These variables were formed by merging the caries, denture, periodontal, soft tissue, and questionnaire exam datasets and noting the presence of the subject with an 'in' option for each dataset. Count is the sum of the 't' variables. Since there is only one entry for each identification number (idno), the sum function in PROC TABULATE gives the value of the 't' variables. The table resulting from this code is given in figure 1.

![Figure 1. Table without enhancements.](image)

Although this table contains all of the information that we need, it is obscure for those who are unfamiliar with the names and values of our variables, and the dashed lines of the frame are not aesthetically pleasing. Furthermore, the label 'SUM' is unnecessary and a total number in each exam group would be useful. Fortunately,
SAS software supplies us with the tools to correct these problems.

**Solid Frames**

The SAS system using PROC TABULATE draws tables using the character specified in the FORMCHAR option. The default option however uses plus signs, minus signs, and other standard characters to construct the tables. If the tables are constructed on an IBM compatible personal computer, the ASCII character set has been extended to include a wide variety of graphic characters. To access these extended characters, one holds down the 'ALT' key while pressing the character code on the side keypad. To construct tables with solid borders, use the following option statement before the PROC TABULATE:

```plaintext
options formchar=' - || | - +--'|;
```

where ALT 179 ( | ), ALT 196 ( – ), ALT 218 ( _ ), ALT 191 ( _ ), ALT 195 ( – ), ALT 197 ( _ ), ALT 199 ( – ), ALT 193 ( – – ), ALT 217 ( – – ) and the remaining characters are retained at their default values. If you so desire, the capability also exist for making double line sides, either overall or on one set of sides. It should be noted that these characters can only be printed out on a graphics printer, such as a Hewlett Packard LaserJet. Also since we always use this set of character for our tables, we changed the system default. (Call up the Options Window from the Command Line of the SAS Editor and change the formchar as above; then set PROFILE to 'on' so the changes will be saved when you exit SAS.)

**Formats and Labels**

Since we desire a table that is a check-off list, the 1 and 0 entries can be improved by introducing the following format:

```plaintext
proc format;
  value 0=""
    1=' .1 .f'
    2=' .2 .f'
    (sum=')/rts=7 box='ID';
```

where the blocks are ALT 254. We chose these characters since a blank in the total block might be taken to suggest that we forget something. Furthermore, it serves as an error check, since the accidental duplication of idno's would be highlighted by this format.

Finally we write the replacement for the table statement in the above PROC TABULATE with

```plaintext
table idno=" all='Total',
  (t1=t=chkfmt. t2=t=chkfmt. t3=t=chkfmt. t4=t=chkfmt. t5=t=chkfmt. count=t=chkfmt.)*
  (sum=')/rts=7 box='ID';
```

where, by labeling the variable idno and the keyword sum as blank, we eliminate the 2nd line from our table. The column of data for the variable idno is then given a heading on the first line by including the box keyword. The keyword 'all' is labeled and the title row size is adjusted. Titles are also used to label the table with the study name and the table name and a footnote statement is used to date the table. This gives us the result in Figure 2.

```
\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
ID & Caries & Denture & Perio & Tissue & Quest & COUNT \\
\hline
727 & & & & & & 4 \\
728 & & & & & & 2 \\
730 & & & & & & 3 \\
731 & & & & & & 4 \\
732 & & & & & & 3 \\
733 & & & & & & 2 \\
734 & & & & & & 3 \\
735 & & & & & & 3 \\
\hline
Total & 64 & 9 & 55 & 22 & 71 & \\
\hline
\end{tabular}
\caption{Table with enhancements.}
\end{table}
```
Many Tables

The check-off list generated above is only one of many tables that must be reproduced. The techniques used in the tables are essentially identical with those given above. However, the summaries that we wish to prepare are stratified by sex and age of the subjects. Since most of the tables are of the same form we can write one table statement and include it in a macro loop.

The subtitles will now change, table numbers will be introduced, and the date given in the footnote must reflect the time the table was created, since tables will be prepared periodically. We can create an ASCII file with the 4 lines of information for each table (blank lines if we don't use all 4) along with a blank line separator, and then read these strings into SAS macros:

```sas
data _null_;
infile 'c:\work\titles' missover eof=last;
input tno $11 #2 ta $65. #3 tb $65. #4 sasvar $8'/;
call symputC'ttln''llieftLnj,trim(tno»; call symput(''ttla''llieftLnj,trim(ta»; call symputC'ttlb''llleftLnj,trim(tb»; call symput(''var''llieftLnj,sasvar); return;
last: call symput("tally",_ n _-1); return;
```

After running this, our first table number will be stored in &ttln1; the first line of the subtitle for the first table in &ttlal; the second line of the subtitle for the first table in &ttlb1; and the first variable to be include in the table will be contained in &var1. The first two lines of the title are globally used and defined in the calling program. The total number of tables will be contained in the macro variable &tally. The information for the second table will be in the macros variables &ttln2, &ttla2, &ttlb2, &var2, and so on. To include these in our tables, we write the header macro routine:

```sas
%macro head(n);
titles "&ttln"n; title4 ;
title5 "&ttla"n; title6 "&ttlb"n; %mend head;
```

To accurately date our tables we can use the macro variable &sysdate formatted into the global macro variable &date (SAS Guide to Macro Processing, 6th ed, p. 204). This is used in a footnote statement in our main program:

```sas
footnote "Database Current as of &date";
```

Finally we can write our macro routine for creating tables:

```sas
%macro tab;
%do i=1 %to &tally;
  proc tabulate
    data=one(keep=sex age &var&i);
    %head;
    class sex age;
    var &var&i;
    table age=" All = 'All Ages', (sex=" all='Both Sexes')*
    (&var&i=")*
    (n*='4. mean='Mean' f=6.2 std f=5.2)
    /box=' Age' rts=10;
  run;
%end;
%mend tab;
```

To save time and memory, we have restricted our input data to contain only the variables needed. (We also should note that another macro routine is used in our reporting in which 'pctn' replaces the keywords 'mean' and 'std' for use with categorical variables.) A sample output from this routine is given in Figure 3.

```
<table>
<thead>
<tr>
<th>Age</th>
<th>Male</th>
<th>Female</th>
<th>Both Sexes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>STD</td>
<td>N</td>
</tr>
<tr>
<td>54-64</td>
<td>6</td>
<td>21.35</td>
<td>20</td>
</tr>
<tr>
<td>65-74</td>
<td>17</td>
<td>21.65</td>
<td>20</td>
</tr>
<tr>
<td>75-84</td>
<td>22</td>
<td>21.27</td>
<td>52</td>
</tr>
<tr>
<td>85-94</td>
<td>35</td>
<td>22.17</td>
<td>54</td>
</tr>
<tr>
<td>95-104</td>
<td>46</td>
<td>20.65</td>
<td>82</td>
</tr>
<tr>
<td>105-114</td>
<td>68</td>
<td>19.76</td>
<td>70</td>
</tr>
<tr>
<td>115-124</td>
<td>44</td>
<td>17.78</td>
<td>48</td>
</tr>
<tr>
<td>125-134</td>
<td>57</td>
<td>17.51</td>
<td>64</td>
</tr>
<tr>
<td>All Ages</td>
<td>209</td>
<td>19.73</td>
<td>531</td>
</tr>
</tbody>
</table>
```

Figure 3. Table generated with macro routine.

Conclusions

We see that the SAS system using PROC TABULATE can be readily used to create publication quality tables on a graphics printer. Using options, labels, and formats, we have a great deal of control over the final product. Using macro programming, we are able to readily automate the preparation of a large number of similar tables for periodic reports.

For Additional Information Contact

William C. Murphy
Eastman Dental Center
625 Elmwood Ave.
Rochester, NY 14620
(716)-275-8296

Database Current as of March 19, 1982
Acknowledgements

Support provided by NIH NIDR grant R01 DE09109; D. H. Leverett, Principal Investigator.

SAS is a registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

IBM is a registered trademark or trademark of International Business Corporation.

Other brand and product names are registered trademarks or trademarks of their respective companies.