Abstract: The production of data-dependent graphs can be a time-consuming and formidable process. Moreover, the display of multiple parameters, multiple graphs per page, or a combination of the two can lead to programming nightmares. The graphical presentation of several parameters involves either repetitive code or hardcoding the TREPLAY statements in PROC GREPLAY for each parameter. To overcome this problem, we have developed a program that employs macros to produce multiple plots per page for any number of parameters without the need to generate repetitive code or hardcoding the TREPLAY statements in PROC GREPLAY. Since macros are used to process the data, produce titles and footnotes in PROC GSGLIDE, axis labels in an ANNOTATE data step and the appropriate TDEF and TREPLAY statements in PROC GREPLAY, minimal input by the user is required. This program was applied to the graphical display of multiple parameters, multiple graphs per page, the display of multiple parameters, as well as treatment group, patient number, investigator, sex, and time point.

The variable name for each clinical laboratory parameter is also the "in-house" standardized 3 or 4 character abbreviation for that parameter. Associated with each of these parameters are the normal range low and high values as well as a code for units. Their variable names on this data set include the abbreviation for the parameter to which they refer. For example, the variable names for hematocrit, hematocrit normal range low, hematocrit normal range high, and hematocrit units are HCT, HCTXL, HCTXH, and HCTXU, respectively.

Formats for all coded information on these data sets are stored in catalogs which are located in the same subdirectory as the data sets.

Parameters: There are 3 macro variables for which values must be supplied through keyword parameters when the macro is called. An additional 10 variables have default values which may be changed at the user's option through keyword parameters. Descriptions of all these variables and options follow.

Required Parameters

SET - the name and fileref of the clinical laboratory data set; this should be supplied in the form [directory]/filename
LABLIST - the list of all the clinical laboratory variable names to be plotted separated by blanks, and
EVAL2 - the follow-up time point to be plotted; there are 4 options:
1. the code (as it appears on the data set) for the follow-up time point to be plotted,
2. L - if the largest value after baseline (or reference time point) is to be plotted,
3. S - if the smallest value after baseline (or reference time point) is to be plotted, or
4. C - if the value which produced the greatest absolute value of the change from baseline (or reference time point) is to be plotted.

Optional Parameters and Defaults

EVAL1 - reference time point; the default is the baseline code (0).
LABELX - X-Axis label, can be up to 50 characters long; the default is the decode for the reference time point,
LABELY - Y-Axis label, can be up to 50 characters long; the default is the decode for the follow-up time point, or if L, S, or C was specified the default label will describe which values were chosen (i.e. for L - Largest Values Post Baseline),
INV - if set to Y (Yes) plots will be generated by investigator; by default plots are not generated by
investigator,

SEX - if set to Y (Yes) plots will be generated by sex; by default plots are not generated by sex.

OUTLOW/OUTHIGH - user defined low/high range; the default is the range value on the clinical laboratory data set; the user defined value should be entered as a percent of this default (i.e. 10 would mean the user defined range is an additional 10% of the default); specifying a value generates a second low/high range line on the plot.

OUTLIER - if set to N (No) no listing of patients with outlying values will be produced next to the plot; the default is to include this list; outliers are considered to be all points outside the ranges - if OUTHIGH and/or OUTLOW are specified these ranges will be used to determine outliers.

PLOTPAG - the number of plots (1-4) desired per page; the default is one, and

FIGURE - by default "FIGURE n" will appear at the top of each page where n is 1 for the first clinical laboratory parameter and is incremented by 1 for each additional parameter; entering zero will suppress the printing of this heading; entering an integer greater than 1 will start numbering figures sequentially beginning with that number.

Logic: When MULTPLOT is invoked, variables are initialized and a driver macro is then called. The driver macro is basically a DO loop which executes one time for each parameter in the clinical laboratory parameter list (LABLIST). With each DO loop execution, several other macros are called subject to the options specified by the user.

During the first execution of the DO loop, information is written to the SAS log. This information restates the options specified by the user. The data set is then checked for the time points specified by EVAL2 (unless L, S, or C was entered) and EVAL1 (if entered). If either time point is not found on the data set a message to that effect is written to the log and all processing is stopped.

Once the time points are determined to be valid, a check is made for the clinical laboratory variable name. If this variable is found on the data set, the variable and its corresponding units are decoded and a message is written to the log. If the variable name is not on the data set, a message is written to the log, and processing continues with the next LABLIST variable name.

If the variable names and time points entered are valid, a working data set which includes only the necessary variables is created. The appropriate macro, as determined by the value for EVAL2, is called to subset this data set to include only the required time points. For each BY variable (or combination of BY variables) the highest and lowest observed values are found as well as the normal range and user specified high and low values.

A new data set is created for plotting. Each observation now contains both X and the Y values. Values for both sets of ranges are merged onto this data set and out of range values are flagged. The actual construction of the graphs is performed by additional macros. An ANNOTATE data step is used to generate a box on the graph that represents the normal range values, create appropriate labels for any BY variables, and, if the OUTLIER option is enabled, display the outlier values next to the graph.

Before the graphs are created, titles and footnotes are generated by PROC GSLIDE. If GPLOT were used to create the titles, then the display of more than one graph per page would result in each graph having the same title above it. With GSLIDE, only one set of titles/footnotes is displayed on a page.

After the graphs are generated by PROC GPLOT, GREPLAY is used to replay the graphs for output. The graphics catalog WORK.PLOT is used to store the output from GSLIDE and GPLOT. Various templates have been hardcoded into the program; the correct template is chosen based on the value of PLOTPAG. The output from GSLIDE is always replayed in the first template, while the GPLOT output is replayed in the remaining templates. To correctly display the titles/footnotes and multiple graphs per page, the TREPLAY statement was written in the macro language.

The coding of the TREPLAY statement is dependent on the number of graphs per page, the number of parameters to be graphed and the total number of graphs produced for each parameter. Since the program does not terminate until the parameter list is exhausted, all titles and graphs are stored in the temporary graphics catalog. For example, the titles and graphs generated for 2 parameters (A, B), with 3 groups per parameter (A1, A2, A3, B1, B2, B3) and 2 graphs per page would be stored by GREPLAY as follows:

Catalog Entry Param
Entry # Type .meter Group
1 Title A
2 Graph A A1
3 Graph A A2
4 Graph A A3
5 Title B
6 Graph B B1
7 Graph B B2
8 Graph B B3

If two graphs per page were desired the appropriate TREPLAY statements would be as follows (General form: TREPLAY template#entry#):

TREPLAY 1:1 2:2 3:3;
TREPLAY 1:1 2:4;
TREPLAY 1:1 2:2 3:3;
TREPLAY 1:5 2:6 3:7;
TREPLAY 1:5 2:8;

In the first pass through the program, the first and second TREPLAY statements would be generated for parameter A by the macro. In the second pass, the third and fourth TREPLAY statements would be generated for parameter B. The graphs for each parameter would then be located in separate output files. The output files from GREPLAY are labeled param.PLOT where param is the variable name for the clinical laboratory parameter. The X statement is used to delete the output files created by GSLIDE and GPLOT.

Output and Examples: With each new parameter in LABLIST, a new plotting page is started. A separate graph is produced for each treatment group and combination of BY variables. Some examples of the graphs are provided. While more than one parameter may be entered for LABLIST, only the first page which was produced appears here. The macro calls which generated the examples follow.

Figure 1: %MULTPLOT (SET=)

Figure 2: %MULTPLOT (SET=)

Figure 3: %MULTPLOT (SET=)
Conclusion: Although the generation of graphs with SAS/GRAPH® is a straightforward process, the task of programming complicated displays can quickly become cumbersome. In this paper, we have described a program that simplifies the generation of multiple graphs by using macros. Although this program was applied to the presentation of clinical laboratory transition data, with some modification this program could be employed in many other situations where multiple output of data-dependent graphical displays are required.

References:


SAS and SAS/GRAPH are registered trademarks of SAS Institute, Inc.

VMS is a trademark of Digital Equipment Corporation.
FIGURE 1
Scatter Plot of Transitions for Hematocrit (%)
FIGURE 2
Scatter Plot of Transitions for Hematocrit (%)

KEY:

--- Normal Ranges
--- User Defined Range for Outliers
FIGURE 3
Scatter Plot of Transitions for Hematocrit (%)

KEY:
- Normal Ranges
- User Defined Range for Outliers