

Comparing Clinical Trials Data Using Population Trees

Lisa McNamara Hlina, Quintiles, Inc., Research Triangle Park, NC

ABSTRACT

Population trees generated using the SAS/GRAPH® GCHART Procedure are a great way to interpret data from two different populations. In this paper, population trees are used to compare the psychological symptoms obtained from clinical trials data for two different treatment groups over three 28 day cycles. This paper will also discuss how the use of color, shading, and annotated labels help to communicate information quickly and easily. The target audience is for both beginner and intermediate audiences and can be adapted to any platform.

INTRODUCTION

The primary use of graphics is to display large quantities of information for quick and easy interpretation. In this paper, PROC GCHART is used to create population trees summarizing clinical trials data of three psychological symptoms for two groups of women over three 28 day cycles. The SAS® program that was written uses several different SAS features to accomplish this task.

Since the goal is to create 9 similar graphs (one for each of the three psychological symptoms for each of the three cycles), a macro was written and called 9 times. The pieces of information that are unique to each graph were passed as macro parameters. These included:

- symptom variable (&VAR)
- cycle number (&VISIT)
- symptom description (&SYMP)
- subsetting where clause (&WHR)
- figure number (&FIG)

PREPARING THE DATA

The preprocessing of the data for these graphs is rather typical of clinical trials data. Actual counts (frequencies) of the total number of

patients in each treatment group that started the given cycle and the number of patients that suffered from the given psychological symptom during that cycle are calculated. Two FREQ procedures are used to generate these counts.

```
** calculate total number of pts in each trt group that **;  
** started a given cycle **;
```

```
proc sort  
  data = dlib.psych  
  out = denom  
  nodupkey;  
  by patid;  
run;
```

```
proc freq  
  data = denom(where=(&whr));
```

```
tables treat /sparse list noprint out=denom;  
run;
```

```
** calculate the total number of pts that had a given **;  
** symptom at a given visit (day) within a given cycle **;  
proc freq  
  data = dlib.psych (where=(&var=1 and visit=&visit));
```

```
tables treat*visit*sdcyc*&var /sparse list noprint  
  out=outdsn;  
run;
```

Since the population tree will contain the percentage of patients that experiences the psychological symptom at each day and cycle, the DENOM and OUTDSN data sets are combined to calculate these percentages. Also, in order to force PROC GCHART to add a zero reference line in the horizontal bar chart, the percentages for one treatment group are changed to negative values.

```
data perc;  
  merge outdsn (drop=percent)  
  denom (rename=(count=denom) drop=percent);  
  by treat;  
  percent = count/denom * 100;  
  if treat = "45 mg" then do;  
    percent = -percent;  
  end;  
run;
```

By applying a format to the percent values, the numbers will be printed left of zero but will appear as positive numbers.

```
proc format;
  picture posval low-high='009';
run;
```

To insure that the data set contained a record for every possible symptom/treatment/cycle/day combination, a zero filled dummy data set is created and merged with the percentages previously calculated in the PERC data set.

```
data dummy;

  do sdcyc = 1 to 28;
    visit = &visit;
    &var = 1;
    count = 0;
    percent = 0;
    treat = "45 mg";
    output;
    treat = "90 mg";
    output;
  end;

run;

data perc;
  set perc
      dummy;
  by treat sdcyc count;

  if last.sdcyc;

run;
```

This data set, PERC, will be the input data set for the PROC GCHART later in the program.

ANNOTATE STATEMENTS VS. ANNOTATE MACROS

The annotation of information can be done in many ways. Originally the annotate data sets created for this program used annotate statements. Later the statements were replaced by the use of the annotate macros provided in the SAS sample library. The use of the annotate macros considerably reduce the amount of code that is needed to annotate the same information. For example, the following 14 lines of code could be rewritten in 3 lines by using the

annotate macros.

```
** ANNOTATE STATEMENTS **;
function = 'label';
xsys = '2';
ysys = '2';
when = 'b';
x = -1;
y = 80;
text = '45 mg';
color = magenta;
angle = 0;
rotate = 0;
size = 0.75;
style = 'triplex';
position = '4';
output;

** ANNOTATE MACROS **;
%system (2,2);
%sequence (before);
%label(-1,80,'45 mg',magenta,0,0,.75,triplex,4);
```

Refer to the SAS/GRAPH® Software: Reference Version 6, Volume 1, pages 487-488 for more information about the annotate macros.

CREATING LABELS, REFERENCE AREAS, AND FOOTNOTES

Two annotate data sets were created to add color labels, shaded reference areas and footnotes to the graphs. In addition to creating a set of directions for the SAS software to use to place labels and footnotes on the graph, the first annotate data set, ANNO1, uses macro code to determine if a shaded reference area should be placed on the graph. In this clinical trial, the patients did not receive any study medication during the first cycle, but received medication from Day 15 through Day 26 for Cycles 2 and 3. In addition to the gray bar on the graphs for Cycles 2 and 3, a footnote is annotated explaining the bar.

```
%if &visit ne 1 %then %do;
  %system (2,2);
  %sequence (before);
  %bar(-100,26,100,15,gray,0,solid);
  %system(2,3);
  %sequence(after);
  %label(-100,3,'Note: The shaded gray box denotes the
days the patients were on study drug.',black,0,0,.75,
triplex,6);
%end;
```

When annotating information, sometimes the order in which things are annotated becomes important. In this example, it is important that the gray box is annotated before the procedure output is generated. If the box was generated last, it would be overlaid on top of the population tree.

The second annotate data set, ANNO2, is used to label the horizontal percentage bars with the frequency counts of the number of patients experiencing the psychological symptom on each of the 28 days of the cycle. Values from the previously calculated data set, OUTDSN, are used to determine the position of the labels. In order to add some white space before and after the counts, the hexadecimal value for a null character, which takes up 1 space in SAS/GRAPH, is concatenated before and after the value. This small addition makes the graph much easier to read.

```
text = '00'x || trim(left(put(count,posval.))) || '00'x;
```

The positioning of the frequency counts depends on the treatment group. The 45 mg group is to the left of the 0 reference line, therefore the position is right aligned. The 90 mg group is right of the reference line and therefore left aligned.

```
if treat = '45 mg' then
  position = '4';      ** centered right aligned **;
else
  position = '6';      ** centered left aligned **;
```

GENERATING THE GRAPH

Before the actual PROC GCHART is called, the axis statements and pattern statements are created. Color is used in the pattern statements to help easily differentiate between the two treatment groups. When this graph was originally done, color was not available so gray shading was used instead.

```
axis1 label=(h=.85 "Cycle &visit" justify=right 'Day');
axis2 label=(justify=center h=.85 "Percentage of Patients
Who Experienced &symp")
order=(-100 to 100 by 10) minor=none;
```

```
** use color=gray20 if not using a color printer **;
pattern1 value=solid color='magenta';
```

```
** use color=graya0 if not using a color printer **;
pattern2 value=solid color='blue';
```

The HBAR statement in PROC GCHART is used to generate the population tree. A format statement is used to suppress the negative sign on the values in the 45 mg treatment group so they appear as positive numbers. The SUBGROUP option is used in the HBAR statement to determine the different groups. In this case, there are two groups (treatment groups 45 mg and 90 mg). The patterns are assigned to these subgroups. The NOSTAT option suppresses the descriptive statistics generated by the SUMVAR option from being printed on the graph. The NOLEGEND option suppresses the default legend generated by the SUBGROUP option from being printed on the graph.

```
proc gchart
  data = perc
  annotate=anno1;
```

```
format percent posval.;
```

```
hbar sdcyc /sumvar=percent discrete nostat
subgroup=treat maxis=axis1 raxis=axis2 nolegend
annotate=anno2;
```

```
title1 h=.75 j=1 " " move=(+5,+0) "Your Favorite
Company" j=r "Page 1 of 1";
title2 h=.75 j=1 " " move=(+5,+0) "Your Favorite
Protocol";
```

```

title3 h=1.5 " ";
title4 h=.85 j=c "FIGURE &fig";
title5 h=.85 " ";
title6 h=.85 j=c "SUMMARY OF PSYCHOLOGICAL
SYMPTOM: " "%upcase(&symp)";
title7 h=.85 j=c "INTENT-TO-TREAT
POPULATION";
title8 h=.85 j=c "CYCLE = &visit";
title9 h=1.70 " ";
footnote1 " ";
footnote2 " ";
run;

```

The macro variable for the psychological symptom description (&SYMP) is used in annotate macros, axis labels and in the titles. In all three instances, the case needs to be different. Instead of creating three macro variables, one in uppercase, one in lowercase and one in mixed case. One variable was created in mixed case and the %UPCASE and %LOWCASE functions were used, where appropriate.

RESIZING THE GRAPHIC OUTPUT

An in-house macro is used to adjust the margins of the graph. Refer to Kenny, Susan J. (1996), "Useful SAS/GRAPH Tools for a Production Environment", Proceedings of the Twenty-First Annual SAS Users Group International Conference, 21, 836-843 for more details on this very useful macro.

GENERATING NINE GRAPHS

The macro to generate this graph is called 9 times to generate a graph for each of the three psychological symptoms at each of the three cycles.

```

%genplot(var=sddep, visit=1, symp=Depression,
whr=itt eval=1, fig=1.0);
%genplot(var=sddep, visit=2, symp=Depression,
whr=str_cyc2=1, fig=2.0);
%genplot(var=sddep, visit=3, symp=Depression,
whr=str_cyc3=1, fig=3.0);

```

CONCLUSIONS

The code and explanations presented here are only one of many ways these graphs could have been produced. Personal style, hardware, and

time constraints all factor in to how a program is written. Fortunately, SAS/GRAPH® software offers the you many creative ways to meet your graphics needs.

REFERENCES

SAS Institute Inc., SAS/GRAPH® Software: Reference Version 6, First Edition, Cary NC: SAS Institute Inc., 1990.

SAS Institute Inc., SAS/GRAPH® Software: Examples Version 6, First Edition, Cary NC: SAS Institute Inc., 1990.

Kenny, Susan J. (1996), "Useful SAS/GRAPH Tools for a Production Environment", Proceedings of the Twenty-First Annual SAS Users Group International Conference, 21, 836-843.

Author Contact

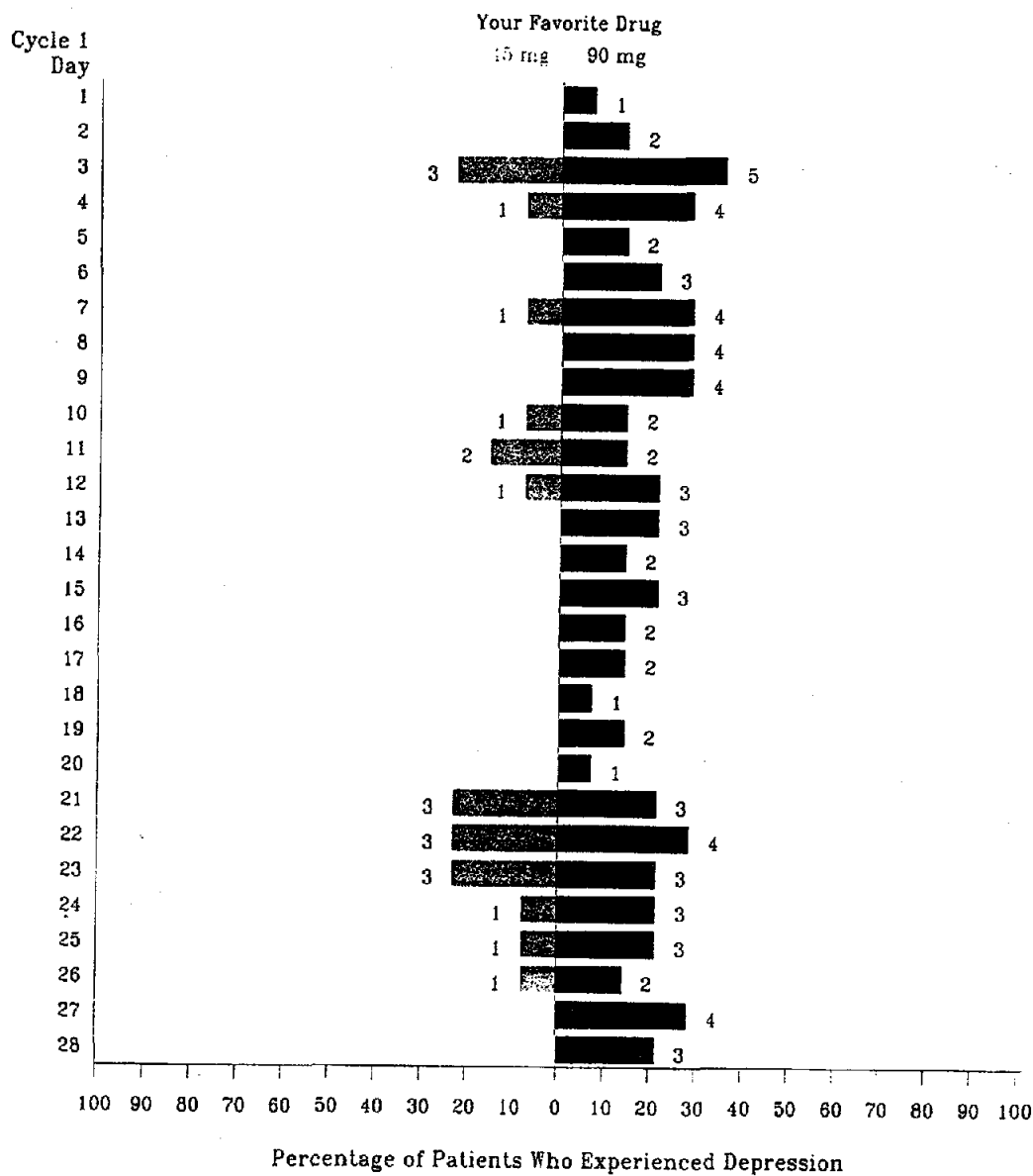
Lisa McNamara Hlina
Statistical Programming Systems
Quintiles, Inc.
P.O. Box 13979
Research Triangle Park, NC 27709
(919) 941-7327
lhlina@quintiles.com

Your Favorite Company
Your Favorite Protocol

Page 1 of 1

FIGURE 1.0

SUMMARY OF PSYCHOLOGICAL SYMPTOM: DEPRESSION
INTENT-TO-TREAT POPULATION
CYCLE = 1



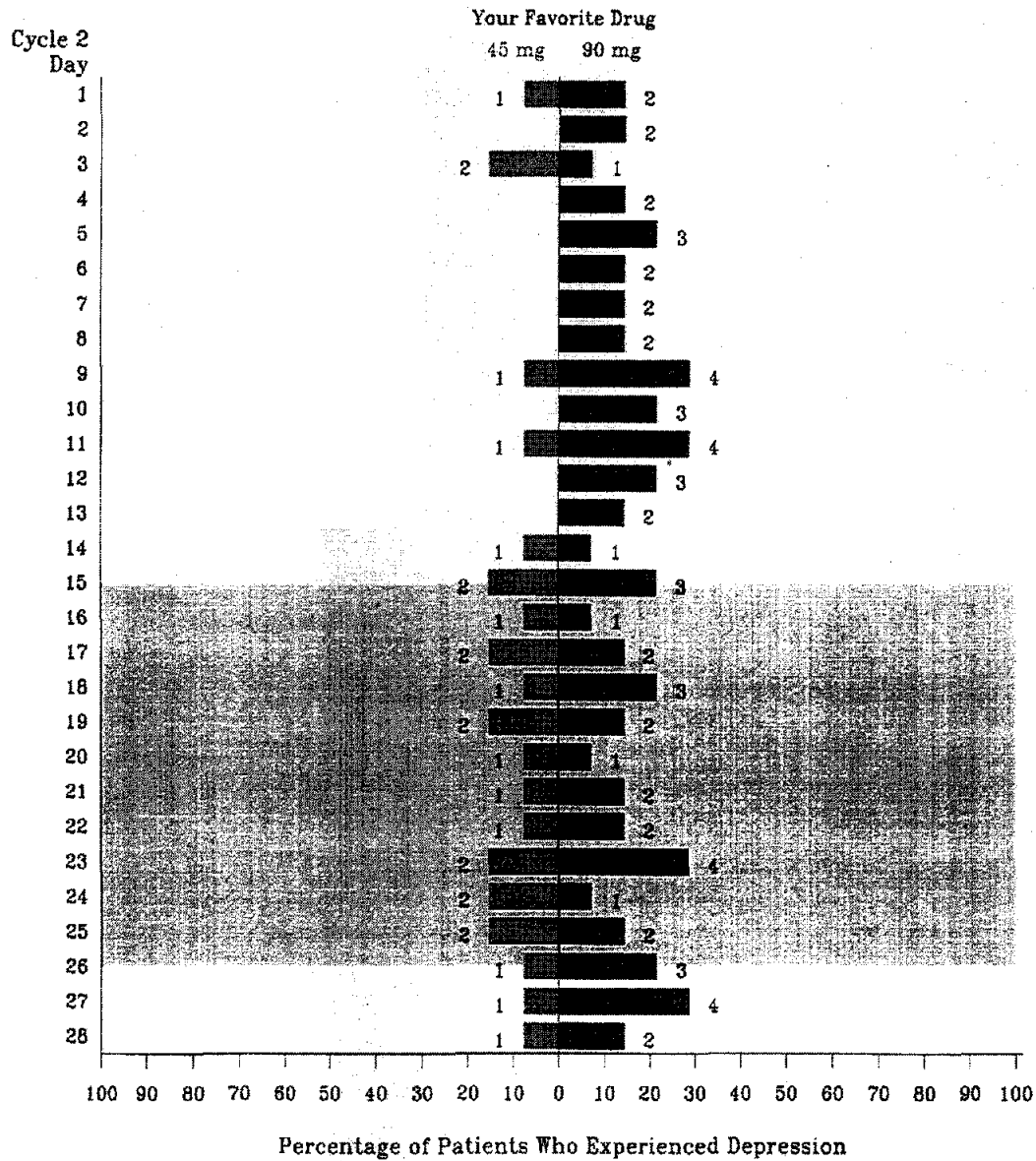
Note: The numbers printed with each bar are the number of patients experiencing depression.
Reference: Table 19.0

Your Favorite Company
Your Favorite Protocol

Page 1 of 1

FIGURE 1.1

SUMMARY OF PSYCHOLOGICAL SYMPTOM: DEPRESSION
INTENT-TO-TREAT POPULATION
CYCLE = 2



Note: The shaded gray box denotes the days the patients were on study drug.

Note: The numbers printed with each bar are the number of patients experiencing depression.

Reference: Table 19.0