Paper PR06-2009

Stacked Cumulative Percent Plots

Theresa Gilligan, Research Triangle Institute Health Solutions, RTP, NC

ABSTRACT

Longitudinal data present analysts with the unique challenge of summarizing information over time without losing the detail of pattern changes, spikes, or other irregularities. This paper introduces a graphical method for displaying ordinal response trajectories that allows you to compare the percent response of multiple categories. As a result, researchers can not only detect symptom improvements or declines over time, but they can also visualize when the frequency of a category increased or decreased. The analyses and output discussed were created with Base SAS® Version 9.1 and SAS/GRAPH® software and will hopefully be of interest to SAS users of all skill levels.

INTRODUCTION

This paper offers a simple option for those interested in displaying the spread of multiple longitudinal response categories. An example very similar to a real-world project served as the basis for the instructions and illustrations of this technique presented throughout. The entire process, from data derivation to graph creation, is discussed. Additional examples are provided for further application. Lastly, unforeseen data and graph issues are highlighted, and resolutions/fixes are offered.

EXAMPLE BACKGROUND

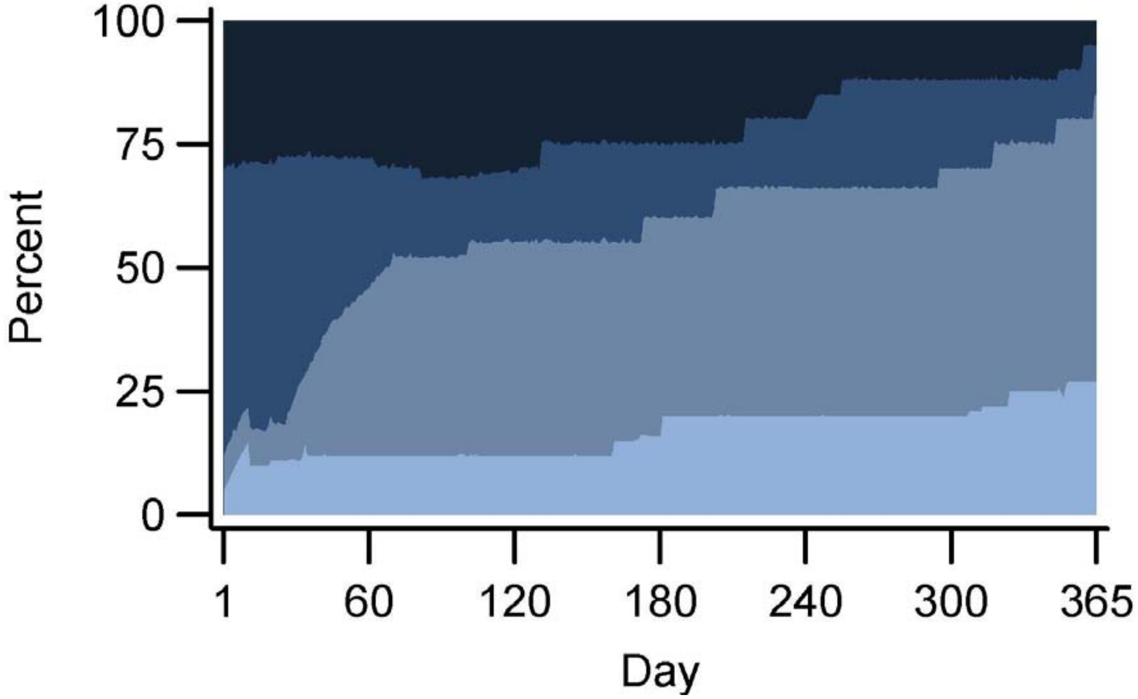
Patients were enrolled in a study to track their responses to various treatments over time. They completed a questionnaire daily that captured the status of their symptoms over 365 days. Each questionnaire item was ordinally scaled so that lower response values (0 and 1) indicated less severe symptoms and higher response values (2 and 3) indicated more severe symptoms. Researchers wished to track the frequency of responses for each ordinal category over time, allowing them to visualize the movement of one response category to another over all 365 days.

GETTING STARTED

THE GOAL

The final goal is to create a plot that retains categorical responses and tracks the movement of these responses over time. As with most longitudinal plots, the x-axis will represent time. The y-axis will display the stacked percent of each coded response. In other words, the objective is to create something much like a stacked frequency plot, but using cumulative percentages instead of frequencies. Percent is chosen for the y-axis so that the proportion of responses can be compared.

Figure 1 displays an example of the final goal.



In this example, the x-axis represents time in days, while the y-axis represents the stacked percent of each category response. Over time, the lighter-blue areas increase, indicating an increase in lower-intensity symptoms. The darker-blue areas decrease, indicating a decrease in the frequency of higher-intensity symptoms compared to the frequency of other responses.

THE DATA

Figure 1 crosses cumulative response percentages by diary day. Therefore, the raw analysis data set should include at least the following information:

1. Patient ID variable
2. Time-indicator variable
3. Response variable(s)

To achieve this final data set, you should begin with a preliminary data set, as presented in Table 1.

Table 1. Example Raw Data Set

PATIENTID	TIME	VARIABLEA
1	1	1
1	2	2
1	3	2
1	4	0
1	5	3
1	↓	↓
1	365	2
2	1	3
2	2	3
2	3	.
2	4	1
2	5	1
2	↓	↓
2 ↓ **100**	365 ↓ **365**	2 ↓ **1**

Note that more than one record per person (i.e., N = 100 with 365 records per person) is presented. The variable “TIME” represents each day and will be shown on the x-axis. If your diary lasts 2 weeks, you would need 14 rows per person, representing 14 days. VariableA represents daily item responses and will be used to derive variables that are shown on the y-axis.

Next, using the FREQ procedure, compute item response frequencies separately for each day (or other preferred time interval) for the response categories intended for plotting. The following is sample code for time period 1:

```
ODS OUTPUT OneWayFreqs = OutputFreq1;
 PROC FREQ DATA = Change;
   WHERE Time = 1 and VariableA ne .;
   TABLES VariableA;
 RUN;
```

It is suggested that you create a macro to run this frequency for each value of the time interval variable. Given this example, you would need to run the frequency 365 times. If your diary lasted 3 weeks, you would run the frequency 21 times. You will only need the response values and cumulative percent information from the output data set above. Table 2 is an example of the output data set from the frequency above for 1 day. The response values and cumulative percent variables are highlighted in grey.

Table 2. Example Output Data Set

TABLE	F_VARIABLEA	VARIABLEA	FREQUENCY	PERCENT	CUMFREQUENCY	CUMPERCENT
Table VariableA	0	0	163	42.45	163	42.45
Table VariableA	1	1	164	42.71	327	85.16
Table VariableA	2	2	55	14.32	382	99.48
Table VariableA	3	3	2	0.52	384	100

Finally, set the PROC FREQ to output (for all days captured) into one data set and transpose so that each ordinal response category becomes a column. You will need to create an indicator variable for day on each data set, as in Table 3 (TIME) so that you can differentiate which cumulative percent values are associated with which days. The following is sample code for transposing the large data set with a cumulative frequency for each response by day:

```
PROC TRANSPOSE DATA = OutputFreq1 OUT = TransOutputFreq1 (DROP = _LABEL_)
      Prefix = Cat;
      VAR CumPercent_1-CumPercent_365;
RUN;
```

The data in each column represents the cumulative frequency percent. Include one row per time interval. The final data set looks like the following example in Table 3. The response values in VARIABLEA in Table 2 become individual columns in Table 3 (0 becomes CATEGORY1, 1 becomes CATEGORY2 and so forth).

Table 3. Example Derived Data Set

CATEGORY1	CATEGORY2	CATEGORY3	CATEGORY4	TIME
0	30	55	100	1
0	30	55	100	2
2	32	57	100	3
4	34	59	100	4
7	37	62	100	5
↓	↓	↓	↓	↓
3	40	70	100	365

GENERAL CODE

Using the final data set above, you can now create stacked cumulative percentage plots. To create these plots, use the following basic GPLOT procedure code:

```
PROC GPLOT DATA = Fig1 GOUT = work.gseg;
     PLOT (Category1 Category2 Category3 Category4) * day /OVERLAY AREAS = 4
       NOLEGEND NOFRAME HAXIS = axis1 VAXIS = axis2;
RUN;
QUIT;
```

Each category variable is plotted against the day variable as a separate horizontal line in the graph by including all four variables within parentheses on the plot line. Then, categories 1 through 4 are overlaid on the same plot (option OVERLAY on the plot line). The lines will never intersect because they represent cumulative frequencies. The AREAS = *4* option fills the area below each line with a color. To specify colors, text styles, and other formatting options in the plots, select additional options. Table 4 provides descriptions of some helpful options.

Table 4. Selected Options Within the GOPTIONS Procedure and the GPLOT Procedure

OPTION	**DESCRIPTION**
GOPTIONS	
RESET = ALL	Clears any previous options from other graphs you may have created
HSIZE and VSIZE	Determines the size of the horizontal and vertical graphics areas, respectively
FTEXT	Specifies default font text (in this case, Helvetica)
DEVICE = sasprtc and TARGET = sasprtc	Specifies the device driver for graphics (SASPRTC is the universal printer)
COLORS	Specifies the colors used to fill in the areas under each line
DISPLAY	Displays graph[a]
HTEXT	Specifies the default size of the text
Axis	
COLOR	Specifies the color of the axis lines
LABEL	Specifies the text label for the axis
MAJOR	Specifies the height and width of the major tick marks
MINOR	Specifies the height and width of the major tick marks (in this case, no tick marks)
ORDER	Lists the values associated with the major tick marks (in this case, specified the exact values)
VALUE	Specifies the major tick mark values
WIDTH	Specifies the width of the axis lines
Symbol	
INTERPOL	Specifies connection criteria for plot points. (The JOIN option specifies that the data points will be joined with a line.)
WIDTH	Specifies the width of the line(s)
Title	
COLOR	Title color
JUSTIFY	Specifies the justification of the title text
HEIGHT	Specifies the height of the title text
GPLOT	
OVERLAY	Over lays all the categories listed in the plot statement on one plot
AREAS	Instructs SAS to fill in colors below each line created
NOLEGEND	Instructs SAS not to create a legend
NOFRAME	Instructs SAS not to create a frame around the plotting areas
HAXIS	Indicates which axis number is associated with the horizontal axis
VAXIS	Indicates which axis number is associated with the vertical axis

[a] This option is the default. The converse option is NODISPLAY, which does not display the graph.

The following options were used to create Figure 1:

```
ODS LISTING;

GOPTIONS RESET = all NOPCLIP NOPOLYGONCLIP NOPOLYGONFILL
HSIZE=8.4in VSIZE=6.3in
FTEXT="Helvetica"
DEVICE=sasprtc TARGET=sasprtc
COLORS=(VLIGB LIGB STGB DEGB BLACK) DISPLAY HTEXT = 2;

AXIS1
COLOR = black
LABEL = ("Diary Day")
MAJOR = (HEIGHT = 1.5 WIDTH = 10)
MINOR = none
ORDER = (1 60 120 180 240 300 365)
VALUE = (height = 2)
WIDTH = 10;
```

```
AXIS2
COLOR = black
LABEL = (angle = 90 "Percent")
MAJOR = (HEIGHT = 1.5 WIDTH = 10)
MINOR = none
ORDER = (0 to 100 by 25)
VALUE = (height = 2)
WIDTH = 10;

SYMBOL1 INTERPOL = join WIDTH = 1;

TITLE1 COLOR = black JUSTIFY = center HEIGHT = 6;

ODS PDF file = "C:\Figure1.PDF";

PROC GPLOT DATA = Fig1 gout=work.gseg;
  PLOT (Category1 Category2 Category3 Category4) * DAY /OVERLAY AREAS = 4
  NOLEGEND NOFRAME  HAXIS = axis1 VAXIS = axis2;
RUN;
QUIT;

ODS PDF CLOSE;
```

Make sure to list the category variables from least to greatest in the plot statement because the areas are filled given the order of the variables in this statement.

The key pieces of this code are the OVERLAY (to put the data for each category on one plot), INTERPOL (to connect the points for each category from day to day), and AREAS (to fill the area below the lines) options. These options make the stacked cumulative percent graphs possible.

You can output these plots in more than one form. The example above illustrates one way to create PDF versions of the graphs. Another popular output type is RTF.

APPLICATIONS

VARIATIONS

Variations of this code can be used to create graphs that display the percent of change from Time A to Time B. Some medical treatments tend to yield symptom changes more quickly than others. For treatments that control symptoms, change rarely occurs. However, with treatments such as antibiotics, relatively quick changes occur in responses over time. If you use a graph to show the change from Time A to Time B, take into account the reaction time: graphs displaying antibiotic responses might show change from morning to evening over a period of a few days or weeks while treatments that gradually control symptoms may only require a weekly or monthly change time point.

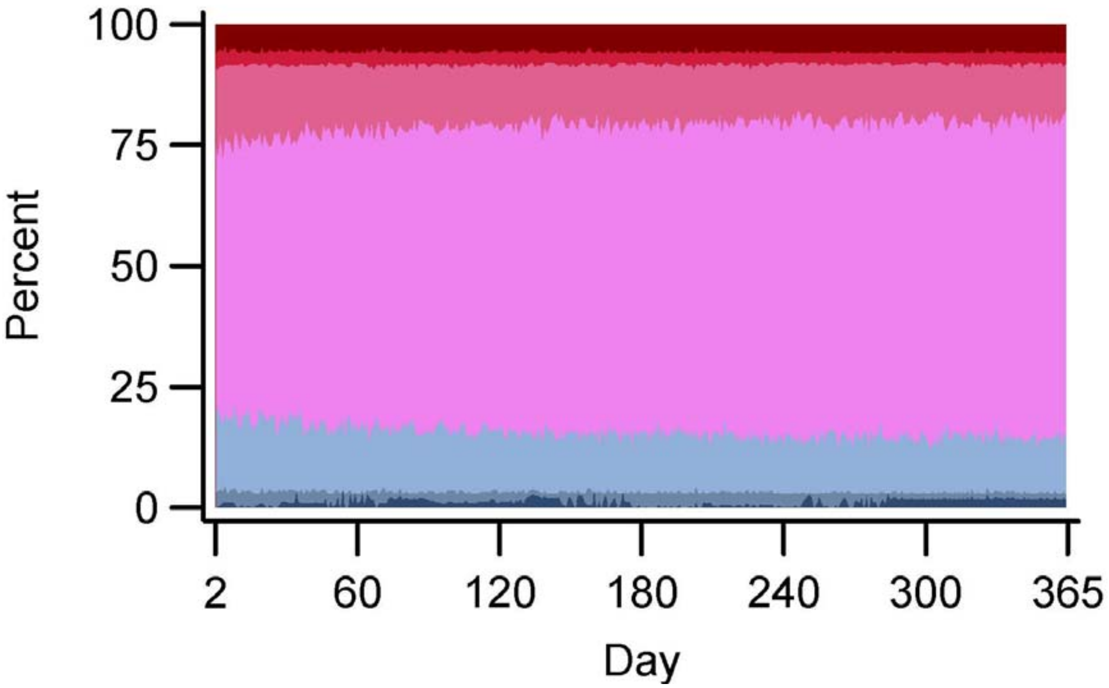
To create these figures, it is important to include an additional DATA step to capture the difference from day to day. If missing data occurs, the difference will also be considered a blank. The first row will always be blank for difference variables because there is no way to find the difference from Day 0 to Day 1. Table 5 was created from the original data in Table 1. DIFFVARIABLEA was calculated by subtracting Day 1 from Day 2 and so forth.

Table 5. Example Raw Data Set With Difference Variable

PATIENTID	TIME	VARIABLEA	DIFFVARIABLEA
1	1	1	.
1	2	2	1
1	3	2	0
1	4	0	–2
1	5	3	3
1	↓	↓	↓
1	364	2	–1
2	1	3	1
2	2	3	0
2	3	.	.
2	4	1	.
2	5	1	0
2	↓	↓	↓
2	364	2	0
↓	↓	↓	↓
100	**365**	**1**	.

After creating Table 5, you can use PROC FREQ, which you used to create Table 2, to create a similar table with seven categories (–3 to +3) instead of four (0 to 3). Given this example, seven categories will capture all of the possible change values from –3 to +3. The number of categories will vary depending upon the number of options given to the respondent.

Figure 2 displays an example of what a graph may look like when the treatment gradually controls symptoms over time.



The majority of Figure 2 is lavender, indicating that most people had no change from Time A to Time B. In fact, the lavender area gradually widens over time, which corresponds to a gradual stabilization in symptoms from day to day for the majority of the patients. There were some +/–3 responses; however, no clear patterns are seen that indicate a precipitous response.

Figure 3 displays an example of what a fast-acting antibiotic treatment may look like.

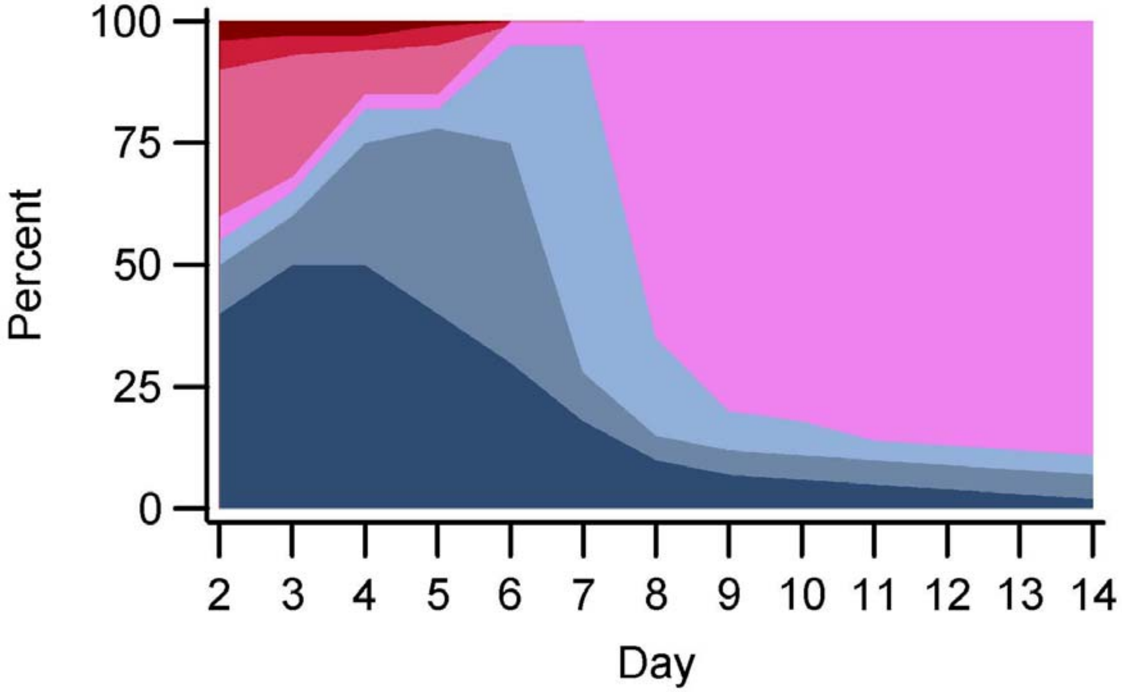
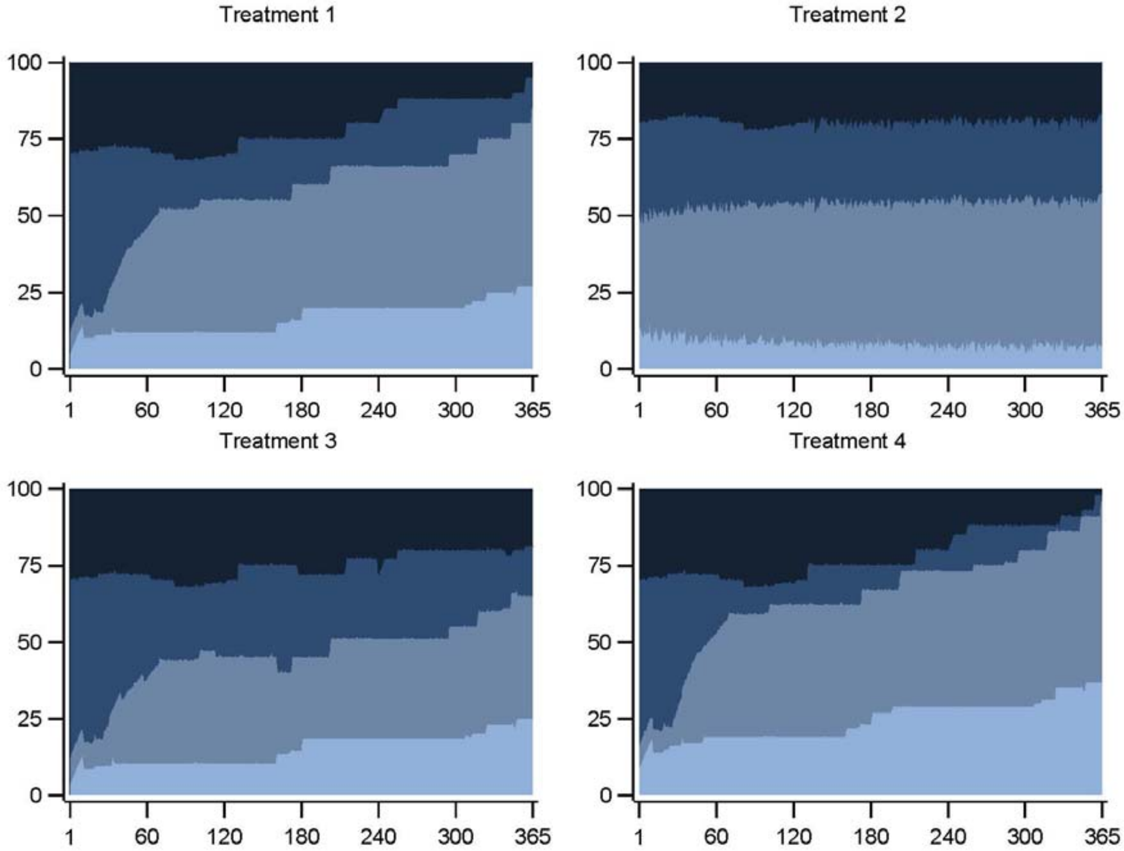


Figure 3 does not follow the 365-day timeline because most patients experience a decrease in disease symptoms within a few days. Therefore, only 14 diary days were captured and displayed. The blue shades indicate negative changes or decreased symptoms (improvement). Red shades indicate positive changes or increased symptoms (worsening). A researcher would expect to see more negative changes (blue) than positive changes (red) over time for an effective antibiotic. In Figure 3, the darkest blue peaks first because most patients will improve after the first few days of treatment. The percentage of red shades decreases over time, indicating that most patients will not experience more severe symptoms as time goes on. The lavender color is the most prominent and fills most of the chart area after 7 days. As time progresses, most people will experience no change in symptoms because the antibiotic has already controlled most of the disease.

To create the seven-category table, the only option changes required to the Table 2 PROC GPLOT are with the variable names, colors (i.e., seven colors instead of four), and the number of areas to be filled in the *AREA option* (i.e., seven areas instead of four).

GREPLAY PROCEDURE

Stacked cumulative frequency plots are also useful for comparison purposes. The figure below displays four graphs, comparing four treatments over the same time period. This example is an extension of Figure 1 in that the figures retain categorical responses and track the movement of these responses over time. The graphs are set in a 2x2 matrix so that they are easily compared.



All four treatments display the same time interval and scale. As in Figure 1, less severe symptoms are characterized by the lighter blue tones and the more severe symptoms are characterized by the darker blue colors. Better performing treatments are obvious at first glance. Patients in treatments 1, 3, and 4 experienced an increase in less-severe symptoms. Treatment 4 yielded the most noticeable increase in less severe symptoms. The graph displaying the reaction of treatment 2 is flat. This type of presentation indicates that change rarely occurred for patients over time. Roughly the same percent of patients is found within each symptom response.

With the GREPLAY procedure, templates can be created to compare multiple treatment responses simultaneously. SAS creates some templates. You must create other templates, such as the 2x2. The code below creates the 2x2 template.

```
LIBNAME EXAMPLE "C:\SAS Project";
PROC GREPLAY tc = EXAMPLE.tempcat nofs;
tdef four des='Four panels (2x2)'
      /* define panel 1 */
      1/llx=0 lly=50 ulx=0 uly=100 urx=50 ury=100 lrx=50 lry=50
      /* define panel 2 */
      2/llx=50 lly=50 ulx=50 uly=100 urx=100 ury=100 lrx=100 lry=50
      /* define panel 3 */
      3/llx=0 lly=0 ulx=0 uly=50 urx=50 ury=50 lrx=50 lry=0
      /* define panel 4 */
      4/llx=50 lly=0 ulx=50 uly=50 urx=100 ury=50 lrx=100 lry=0;
RUN;
QUIT;
```

Use the template in the following PROC GREPLAY code.

```
ODS LISTING;

GOPTIONS RESET = all NOCLIP NOPOLYGONCLIP NOPOLYGONFILL HSIZE=8.4in
VSIZE=6.3in FTEXT="Helvetica" DEVICE=sasprtc TARGET=sasprtc DISPLAY;
```

```
ods PDF file = "C:\Fig4.PDF";
PROC GREPLAY igout=work.gseg tc=PRO.tempcat template=four nofs nobyline;
      TREPLAY 1:Gplot 2:Gplot1 3:Gplot2 4:Gplot3;
RUN;
QUIT;

ODS PDF CLOSE;
```

The number of plots in the TREPLAY procedure statement varies depending upon the matrix you decide to use. Note the use of the DISPLAY option in this section of code. This procedure is required for the final 2x2 plots and DISPLAY is necessary for printing the final output.

LESSONS LEARNED

COLOR FILLS

If the lines created from the cumulative frequencies are relatively close together, SAS has trouble filling colors between lines. If this problem occurs, use the GOPTIONS procedure to change the way SAS fills the colors.

GOPTIONS NOCLIP NOPOLYGONCLIP NOPOLYGONFILL;

Be sure to choose colors that are easily differentiated, especially if the data yields similar results for two separate lines. Otherwise, plot areas may be difficult to distinguish.

MISSING VALUES

Missing values can also cause problems in the creation of each plot. Missing values occur when all examinees do not report a response category on one time interval (in this case, day). The INTERPOL = *join* option skips missing values and connects the last non-missing value with the most recent non-missing value. If a data set begins with missing values on the first day or includes many missing values in the lowest or highest categories, fill problems may occur. To avoid "hiccups," manually fill missing values with the last non-missing value.

```
DATA derived.OutputFreqs;
      set OutputFreqs;
      if category1 = . then category1 = 0;
      if category4 = . then category4 = 100;
      if category3 = . then category3 = category2;
      if category2 = . then category2 = category1;
RUN;
```

DISPLAY OPTIONS

If you choose to compare plots with PROC GREPLAY, you will have to create multiple plots separately, and then integrate them into this procedure. To save time, use a NODISPLAY option to suppress the printing of graphs prior to PROC GREPLAY.

```
GOPTIONS NODISPLAY;
```

CONCLUSION

Stacked cumulative percent plots offer an alternative way to present longitudinal information that is both intuitive and easy to create. Enjoy!

RECOMMENDED READING

Massengill, A.D. “Tips and Tricks: Using SAS/GRAPH Effectively” *Proceedings of the Thirtieth Annual SAS Users Group International Conference,* SUGI 30 April 2005. <http://www2.sas.com/proceedings/sugi30/090-30.pdf> (October 2008).

ACKNOWLEDGMENTS

The author would like to thank Drs. Lori McLeod and Cheryl Hill for their extraordinary support. Additionally, SAS Customer Support (www.support.sas.com) provided extensive and detailed guidance.

CONTACT INFORMATION

Contact the author at:
Theresa Gilligan
200 Park Drive
PO Box 12194
Research Triangle Park, NC 27709, USA
Phone: (919) 316-3843
Fax: (919) 541-7222
E-mail: gilligan@rti.org
Web: www.rtihs.org