

## What HIGHLOW Can Do for You

Kristen Much, Rho<sup>®</sup>, Inc., Chapel Hill, NC  
Kaitlyn Steinmiller, Rho<sup>®</sup>, Inc., Chapel Hill, NC

### ABSTRACT

Longitudinal plots that quickly, creatively, and informatively summarize study data act as powerful tools in understanding a clinical trial. The HIGHLOW statement not only creates a plot with floating bars that represent high and low values but also includes the ability to add markers and text annotations. With a wide variety of options, the HIGHLOW statement can produce a plot that summarizes a complex story with a multitude of different data in a single graphic.

This paper will introduce this relatively new plot statement (in production since SAS 9.3) with an example-based approach, exploring possible applications and plot capabilities. Focus will be placed on how to create HIGHLOW plots using both the PROC SGPLOT procedure and Graph Template Language (GTL). The examples provided will demonstrate the power and flexibility of the HIGHLOW statement.

### INTRODUCTION

Throughout a clinical trial, reports are created to describe the overall study progress, monitor the trial's safety, and disclose study results. While these reports are informative, it is often difficult to grasp the bigger picture without considerable effort. There is generally an abundance of information whereas the crucial material is scattered throughout. One solution to this issue is to create longitudinal plots. These plots can quickly, creatively, and informatively summarize study data over time by integrating several critical aspects of the trial. This allows one to better observe the most important information, including the timing and frequency of these events as well as the patterns within the data. Thus, longitudinal plots can be powerful tools in improving the understanding of the trial as a whole.

There are a number of ways these plots can be implemented within SAS, but the HIGHLOW plot statement in particular offers a direct and easy approach. Since this plot statement has only been available since SAS 9.3, awareness and familiarity are limited. Fortunately, the HIGHLOW statement is easy to learn and requires very little data manipulation. In the upcoming sections, a basic overview of the HIGHLOW plot statement is included along with an exploration of four specific examples that utilize the HIGHLOW plot statement within both the SGPLOT procedure and Graph Template Language (GTL).

### OVERVIEW OF THE HIGHLOW STATEMENT

The HIGHLOW plot statement provides a simple method for creating longitudinal plots. There are three required arguments that must be specified with variables contained in the data set: X or Y, LOW, and HIGH. Specifying X will create vertical floating lines or bars, whereas Y will create horizontal. This variable can be character or numeric; in subsequent examples, this variable is typically equal to the subject identifier. The LOW variable will dictate where the line or bar begins (the lowest value) and the HIGH variable will dictate where the line or bar ends (the highest value). These variables must be numeric. The basic code needed to create a plot using the HIGHLOW plot statement is shown below. The corresponding output can be seen in Figure 1.

#### VERTICAL HIGHLOW PLOT

```
proc sgplot data=plotdata;  
    highlow x=make low=lowprice high=highprice;  
run;
```

#### HORIZONTAL HIGHLOW PLOT

```
proc sgplot data=plotdata;  
    highlow y=animal low=geststart high=gestend;  
run;
```

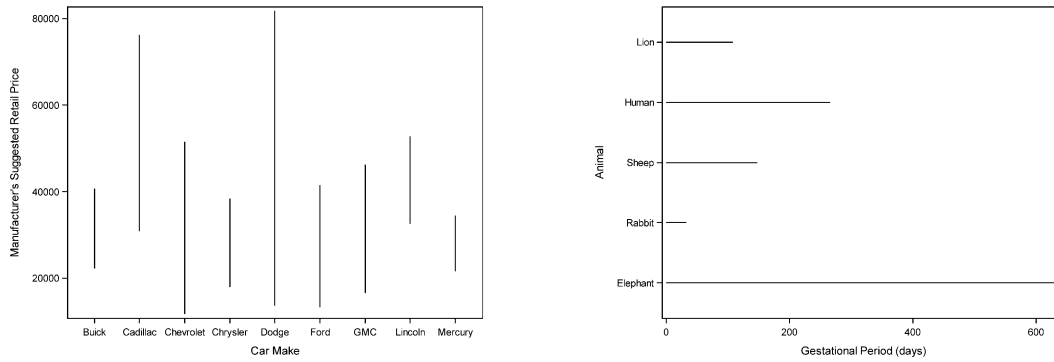


Figure 1. Overview Examples of the HIGHLOW Plot

**EXAMPLE 1: SUBJECT TIMELINE PLOT**

The subject timeline plot displays one line per subject over time. The overall length of the line represents the duration of study participation for each subject. This line changes color over time as each subject enters the various phases of the study. Events of interest are overlaid at the time of the event as indicated with symbols. At a glance, this plot allows for easy examination of the overall trial progress, as well as more specific information such as the frequency and timing of the trial phases and events, subject-specific differences, and data trends. Figure 2 shows a generic version of this figure that can be adapted easily to fit the specifics of any study trial.

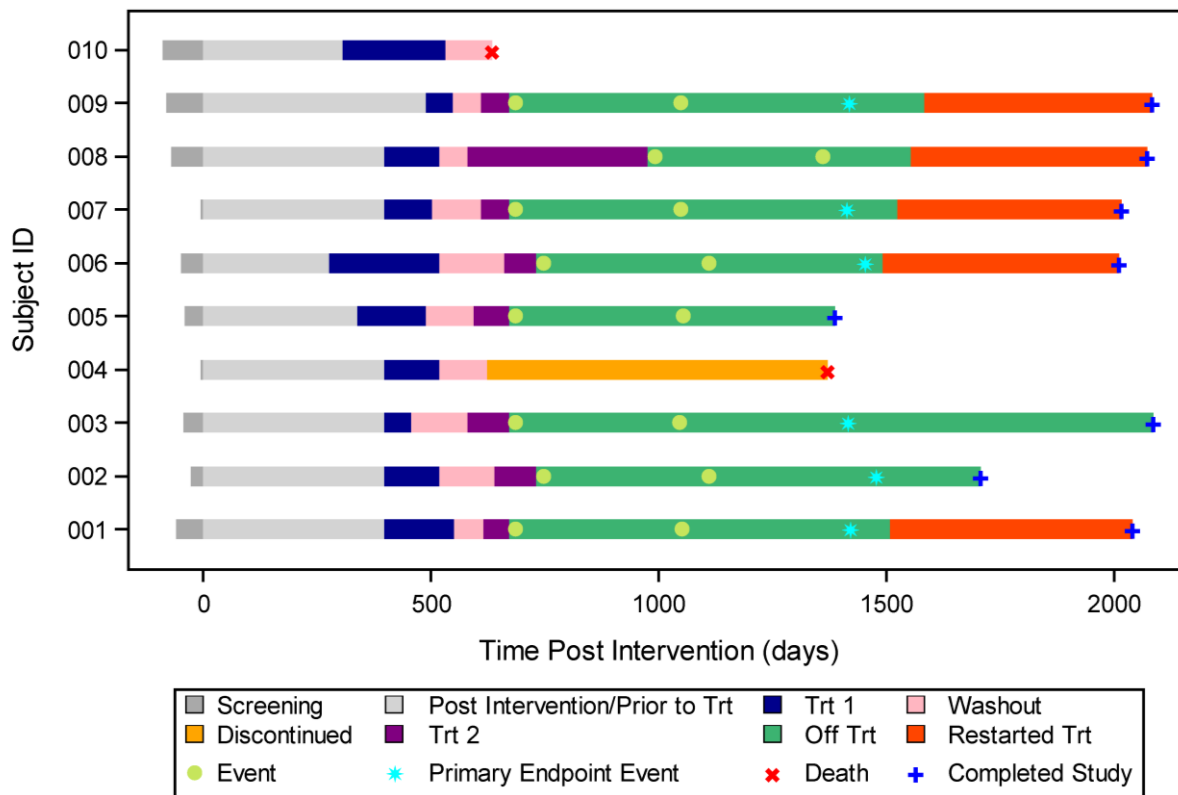


Figure 2. Subject Timeline Plot

The SAS code to produce Figure 2 can be found below.

## FIGURE 2 CODE

```
proc sgplot data=plotdata;
  highlow y=id low=screen high=intervent / <HIGHLOW OPTIONS>;
  highlow y=id low=intervent high=predrug / <HIGHLOW OPTIONS>;
  highlow y=id low=predrug high=drug_1 / <HIGHLOW OPTIONS>;
  highlow y=id low=drug_1 high=washout / <HIGHLOW OPTIONS>;
  highlow y=id low=washout high=drug_2 / <HIGHLOW OPTIONS>;
  highlow y=id low=washout high=disc / <HIGHLOW OPTIONS>;
  highlow y=id low=drug_2 high=off / <HIGHLOW OPTIONS>;
  highlow y=id low=off high=restart / <HIGHLOW OPTIONS>;

  symbolchar name=sym1 char='271A'x;
  symbolchar name=sym2 char='2716'x;
  symbolchar name=sym3 char='2738'x;

  scatter y=id x=event_1 / <SCATTER OPTIONS>;
  scatter y=id x=event_2 / <SCATTER OPTIONS>;
  scatter y=id x=event_3 / <SCATTER OPTIONS>;
  scatter y=id x=death / <SCATTER OPTIONS>;
  scatter y=id x=comp / <SCATTER OPTIONS>;

  keylegend 'l1' 'l2' 'l3' 'l4' 'l5' 'l6' 'l7' 'l8' 's1' 's2' 's3' 's4' / across=4;
run;
```

### HIGHLOW OPTIONS

The code for <HIGHLOW OPTIONS> is specified below. The only changes between each HIGHLOW statement are the color specified for the bar, the legend label name, and the short name used within the KEYLEGEND statement.

```
type=bar barwidth=.4 nooutline fillattrs=(color=purple) legendlabel='Trt 2' name='l5'
```

### SCATTER OPTIONS

Similarly, the only changes between each SCATTER statement are the color specified for the symbol, the symbol shape, the legend label name, and the short name used within the KEYLEGEND statement. The code for the <SCATTER OPTIONS> is displayed below.

```
markerattrs=(color=blue symbol=sym1 size=8pt) legendlabel='Completed Study' name='s4'
```

## FIGURE 2 KEY FEATURES

### MULTICOLOR BAR

In order to create a bar that changes color over time, a separate HIGHLOW statement must be used for each unique segment. The start time for each segment is set to LOW and the end time is set to HIGH. To display a thick bar as opposed to a thin line, TYPE is specified as BAR. The width of the bar can be adjusted with the BARWIDTH statement, which can be a value from 0.0 (narrowest) to 1.0 (widest).

### USE OF SPECIAL SYMBOLS

The SYMBOLCHAR statement allows symbols to be defined using any Unicode character, which means symbols are not limited to the 30 marker options available within SAS. In the code above, a thick plus ('271A'x), a thick X ('2716'x), and a multipoint star ('2738'x) have been selected. These are only three of the hundreds (if not thousands) of Unicode symbols available. The full list of Unicode symbols can be found here: <http://www.unicode.org/charts/charindex.html>. To define a symbol, simply specify a NAME for the symbol within the SYMBOLCHAR statement as well as the Unicode specification (4 digit alphanumeric code within single quotes followed by an 'x') to CHAR. That NAME is then specified as the SYMBOL within the MARKERATTRS.

### LEGEND CUSTOMIZATION

The KEYLEGEND statement creates the legend for the plot. For any plot statement that needs to be included in the legend, a NAME must be specified and then referenced within the KEYLEGEND statement. LEGENDLABEL is used to assign the label within the legend. If LEGENDLABEL is not assigned, the variable's label will be used instead. Within the legend, the ACROSS option represents the number of elements that will be displayed within one row before a new row is displayed.

## EXAMPLE 2: ADVERSE EVENT AND CONCOMITANT MEDICATION PLOT

The ability to link concomitant medication and adverse event information is highly desired among individuals who monitor safety in clinical trials. The HIGHLOW plot in Figure 3 visually connects this data by presenting one line for each concomitant medication and adverse event for a particular subject. The bars are sorted by time of occurrence and the length of the lines represent the duration of the adverse event or concomitant medication. Arrows are used to indicate if an adverse event or concomitant medication is still ongoing.

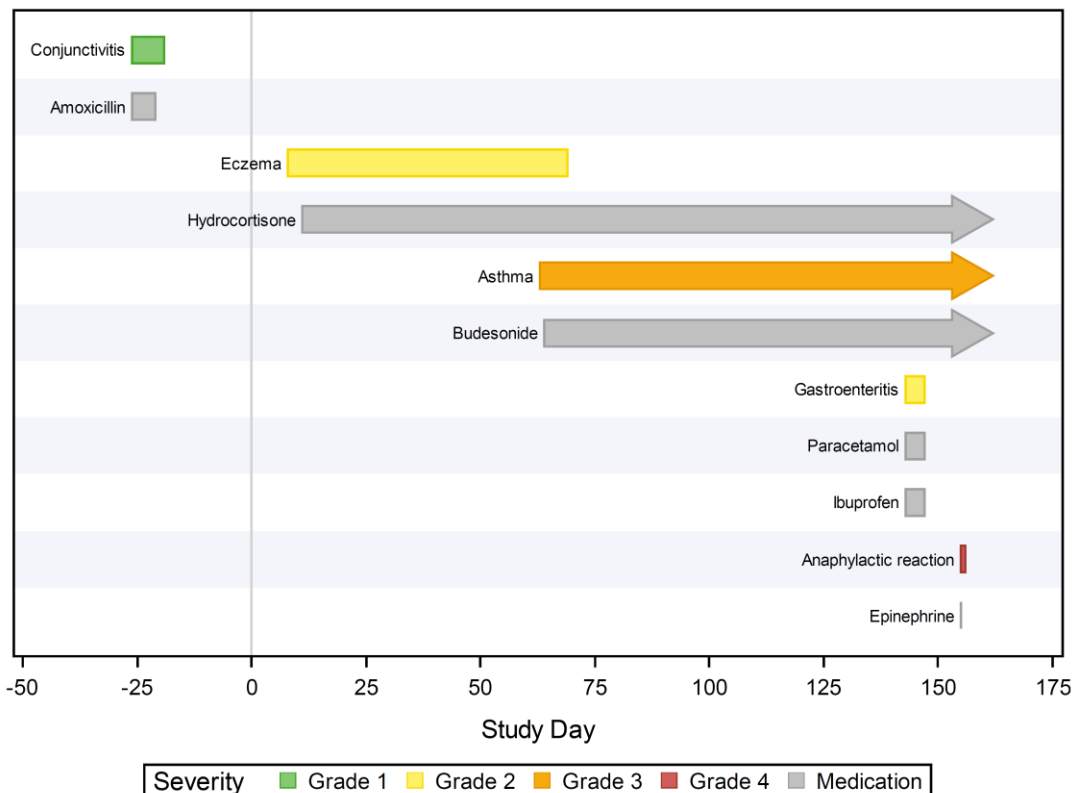


Figure 3. Adverse Event and Concomitant Medication Plot

The SAS code to produce Figure 3 can be found below.

### FIGURE 3 CODE

```

data getattrs;
  retain id 'Color';
  input value $ 1-10 fillcolor $ 12-20 linecolor $ 21-29;
  datalines;
  Grade 1    cx87c873 cx47a82a
  Grade 2    cxfff166 cxf9da04
  Grade 3    cxf7aa0f cxe19600
  Grade 4    cxd05b5b cxa23a2e
  Medication cxc0c0c0 cx9f9f9f
  ;
run;

proc sgplot data=prepdatt dattrmap=getattrs;
  refline 0 / axis=x lineattrs=(color=lightgrey);
  highlow y=label low=startdy high=stopdy /<HIGHLOW OPTIONS>;
  xaxis label='Study Day' values=(-50 to 175 by 25);
  yaxis <YAXIS OPTIONS>;
  label bartype='Severity';
run;

```

### HIGHLOW OPTIONS

The code below for <HIGHLOW OPTIONS> controls the appearance of the floating bars. The GROUP option specifies the severity variable used to group the data and the LOWLABEL option specifies the adverse event or medication label printed at the low end of the bar. There is a similar HIGHLABEL option to specify a label to be printed at the high end of the bar.

```
type=bar group=bartype grouporder=ascending lowlabel=label labelattrs=(color=black)
highcap=cap attrid=Color
```

### Y AXIS OPTIONS

The code below for <YAXIS OPTIONS> removes the ticks, values, and labels from printing as specified in the DISPLAY option. These options are not necessary due to the LOWLABEL used above. Since the data are sorted by the start day, the REVERSE option prints the first adverse event or concomitant medication that occurred at the top of the plot.

```
reverse display=(noticks novalues nolabel) colorbands=even colorbandsattrs=(transparency=0.7)
```

### FIGURE 3 KEY FEATURES

#### DATTRMAP

This option maps discrete values to figure properties by referencing a separate attribute map data set. Typically this data set is created within the program in a DATA step and must contain the following variables:

- ID – identifies the attribute map you want to use
- VALUE – associates the grouping variables from your data to specific figure attributes
- One or more reserved attribute variables such as FILLCOLOR, LINECOLOR, or MARKERSYMBOL

An extensive list of the reserved variables can be found in the Discrete Attribute Map Data Sets section of the SAS® 9.4 ODS Graphics: Procedure Guide.

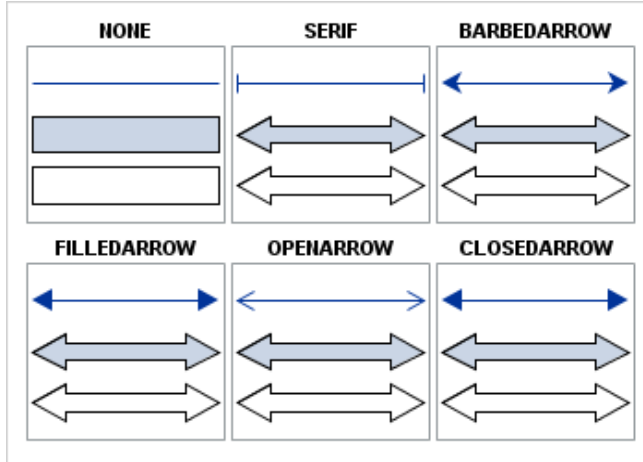
The DATA step in the Figure 3 code creates the GETATTRS data set shown below in Display 1. This data set maps the severity category values to the specified fill and line colors. On the SGPLOT statement, the DATTRMAP option associates the GETATTRS data set with the figure. The ATTRID option on the HIGHLOW statement associates the ID value of “Color” to the HIGHLOW statement, which maps the 5 categories to the fill and line colors. If desired, multiple attribute maps can be created and used on different group variables within a single figure.

	id	value	fillcolor	linecolor
1	Color	Grade 1	cx87c873	cx47a82a
2	Color	Grade 2	cxfff166	cx9da04
3	Color	Grade 3	cx7aa0f	cxe19600
4	Color	Grade 4	cx05b5b	cx23a2e
5	Color	Medication	cx0c0c0	cx9f9f9f

Display 1. GETATTRS Data Set

#### HIGHCAP

The HIGHCAP and corresponding LOWCAP options enables users to specify the type of cap used at the high or low end of a bar or line. This option allows you to specify a keyword or a character variable that contains a keyword. The following figure from the SAS® 9.4 HIGHLOW documentation illustrates the available cap types for lines, filled bars, and unfilled bars along with the associated keywords.



**Display 2. HIGHCAP and LOWCAP Keywords**

Since the HIGHCAP in Figure 3 is conditional on whether an adverse event or medication is ongoing at a specific time, the character variable, CAP, is specified. CAP is programmed to equal 'FilledArrow' if the adverse event or medication does not have a stop date.

**SHADING**

Since Y is a discrete axis, the COLORBANDS option specifies alternating color bands that correspond to the bins on the Y axis. EVEN or ODD can be used to specify how the shaded and unshaded bands alternate. COLORBANDATTRS specifies the transparency and can also be used to change the fill color.

### EXAMPLE 3: SUBJECT TIMELINE PLOT WITH AXIS-ALIGNED TABLE

Similar to Figure 2, this plot summarizes the disposition of subjects over time, with key events highlighted by specific markers. To make this information more interpretable, the figure links this information with key baseline characteristics in an axis aligned table.

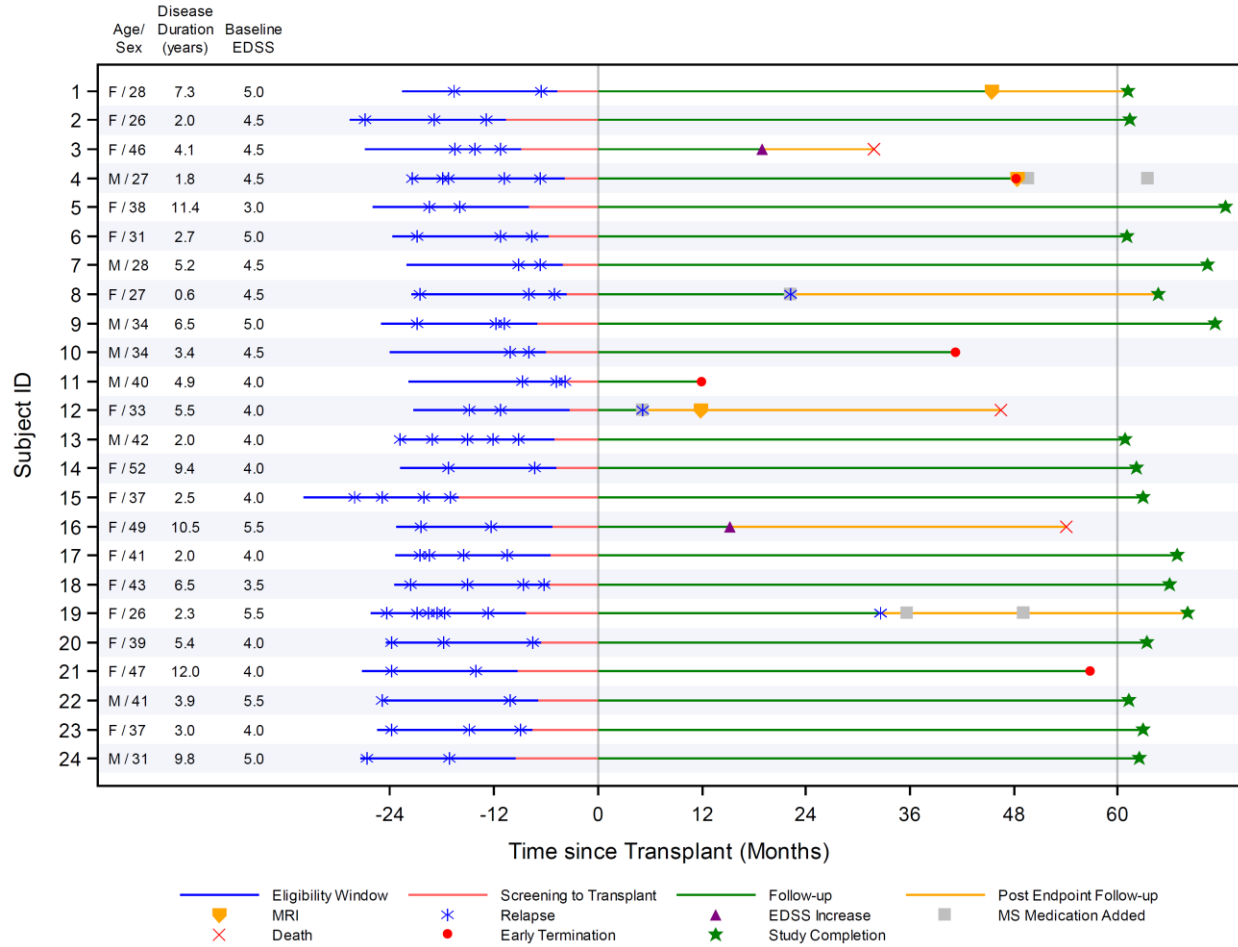


Figure 4. Subject Timeline Plot with Axis-Aligned Table

The SAS code to produce Figure 4 can be found below.

**FIGURE 4 CODE**

```

proc template;
  define statgraph highlow;
    begingraph;

      layout overlay / <LAYOUT OVERLAY OPTIONS>;
        referenceline x=0 / clip=true lineattrs=(color=CX808080 pattern=1) datatransparency=0.5;
        referenceline x=60 / clip=true lineattrs=(color=CX808080 pattern=1) datatransparency=0.5;

        highlowplot y=plotid high=line1h low=line1l / <HIGHLOW OPTIONS>;
        highlowplot y=plotid high=line2h low=line2l / <HIGHLOW OPTIONS>;
        highlowplot y=plotid high=line3h low=line3l / <HIGHLOW OPTIONS>;
        highlowplot y=plotid high=line4h low=line4l / <HIGHLOW OPTIONS>;

        scatterplot x=edss_mo y=plotid / <SCATTERPLOT OPTIONS>;
        scatterplot x=med_mo y=plotid / <SCATTERPLOT OPTIONS>;
        scatterplot x=mri_mo y=plotid / <SCATTERPLOT OPTIONS>;
        scatterplot x=dth_mo y=plotid / <SCATTERPLOT OPTIONS>;
        scatterplot x=rel2_mo y=plotid / <SCATTERPLOT OPTIONS>;
        scatterplot x=rel_mo y=plotid / <SCATTERPLOT OPTIONS>;
        scatterplot x=trans_mo y=plotid / <SCATTERPLOT OPTIONS>;
        scatterplot x=term_mo y=plotid / <SCATTERPLOT OPTIONS>;
        scatterplot x=comp_mo y=plotid / <SCATTERPLOT OPTIONS>;

        axistable y=plotid value=agesex / <AXISTABLE OPTIONS>;
        axistable y=plotid value=dur / <AXISTABLE OPTIONS>;
        axistable y=plotid value=edss / <AXISTABLE OPTIONS>;

        discretelegend 'line1' 'line2' 'line3' 'line4' 'MRI' 'REL' 'EDSS' 'DMT'
          'DTH' 'TERM' 'COMP' / <DISCRETELEGEND OPTIONS>;
    endlayout;

  endgraph;
end;
run;

```

**LAYOUT OVERLAY OPTIONS**

The <LAYOUT OVERLAY OPTIONS> control the attributes of the X and Y axis and apply the shading to every other line.

```

xaxisopts=(offsetmin=0.18 label='Time since Transplant (Months)' labelfitpolicy=Split
type=linear linearopts=(tickvaluelist=(-24 -12 0 12 24 36 48 60)))
yaxisopts=(Label='Subject ID' type=discrete reverse=1 discreteopts=(colorbands=even
colorbandsattrs=(transparency=0.7)))

```

**HIGHLOW OPTIONS**

The <HIGHLOWPLOT OPTIONS> control the color and pattern of each segment of the line. The only differences between the three option statements are the specified color, legend label and associated legend name.

```

lineattrs=(color=CX0000FF Pattern=1) legendlabel='Eligibility Window' name='line1'

```

**SCATTERPLOT OPTIONS**

The <SCATTERPLOT OPTIONS> control the color and symbol type of each key event. The only differences between nine option statements are the specified color, symbol, legend label, and associated legend name.

```

markerattrs=(color=CX800080 symbol=trianglefilled size=8) legendlabel='EDSS Increase' name='EDSS'

```

**AXISTABLE OPTIONS**

The <AXISTABLE OPTIONS> control the position of each of the data columns. The only differences between the three option statements are the specified position and data label.

```

display=(label) labelattrs=(size=7) label='Baseline EDSS' position = 0.13 valuejustify=center

```



### DISCRETELEGEND OPTIONS

The <DISCRETELEGEND OPTIONS> control the location and size of the legend. The DISPLAYCLIPPED option specifies that the legend will be displayed when any portion of the legend is not entirely rendered. This option is particularly useful when working with legends with numerous entries or long labels. By default, if there is not enough space to write all legend entries without overlapping the text, the entire legend will be omitted. The DISPLAYCLIPPED option forces the legend to be printed, which allows you to problem shoot the issue. The ACROSS option specifies the number of legend entries to be placed horizontally before moving to the next row.

```
displayclipped=true location=outside valueattrs=(size=7pt) border=false across=4
```

### FIGURE 4 KEY FEATURES

#### AXIS-ALIGNED TABLE

AXISTABLE creates an axis-aligned inset of the key baseline data. The OFFSETMIN option is used on the X axis to reserve space on the left hand side of the figure for the baseline data. The POSITION option on each AXISTABLE statement then specifies the position that the data are printed.

#### SHADING

Similar to the SGPLOT example in Figure 3, the COLORBANDS and COLORBANDSATTRS can be used in GTL within the DISCRETEOPS list to shade every other line.

### EXAMPLE 4: SUBJECT DOSING STATUS PLOT

The subject dosing status plot shows dosing levels over time for each subject. Each line segment represents a dosing level and the status of that segment is identified by its color. Additionally, the exact dosage is annotated above each segment. This type of plot is helpful when following subjects through complex dosing schedules and can be adapted for dose tapering, minimization, and escalation studies.

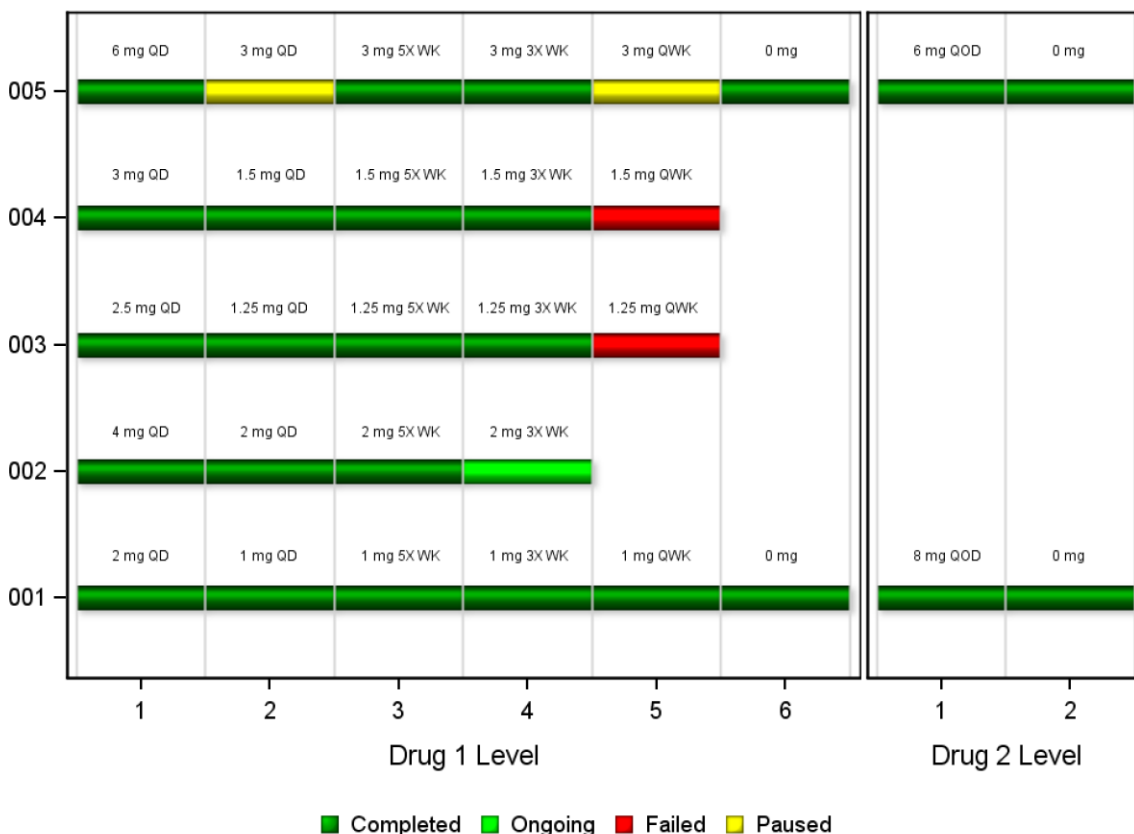


Figure 5. Subject Dosing Status Plot

The SAS code to produce Figure 5 can be found below.

### FIGURE 5 CODE

```
proc template;
  define statgraph dosinggraph;
    beginnograph / drawspace=datavalue;

    discreteattrmap name='colors' / ignorecase=true;
      value "Ongoing" / fillattrs=(color=lime);
      value "Completed" / fillattrs=(color=green);
      value "Paused" / fillattrs=(color=yellow);
      value "Failed" / fillattrs=(color=red);
    enddiscreteattrmap;

    discreteattrvar attrvar=flagcolors var=flag attrmap='colors';

    layout lattice / rows=2 columns=1 rowweights=(.9 .1);
      layout lattice / rows=1 columns=2 columnweights=(.75 .25);
        layout overlay / <LAYOUT OVERLAY OPTIONS>;
          highlowplot y=id low=taplvl_l high=taplvl_h / <HIGHLOWPLOT OPTIONS>;
          referenceline x=eval(coln(0,1,2,3,4,5,6)) / lineattrs=(color=lightgrey);
          drawtext textattrs=(size=6pt) "&freq001" / <DRAWTEXT OPTIONS>;
          drawtext textattrs=(size=6pt) "&freq002" / <DRAWTEXT OPTIONS>;
          drawtext textattrs=(size=6pt) "&freq003" / <DRAWTEXT OPTIONS>;
          drawtext textattrs=(size=6pt) "&freq004" / <DRAWTEXT OPTIONS>;
          drawtext textattrs=(size=6pt) "&freq005" / <DRAWTEXT OPTIONS>;
        endlayout;

        layout overlay / <LAYOUT OVERLAY OPTIONS>;
          highlowplot y=id low=taplvl_l high=taplvl_h / <HIGHLOWPLOT OPTIONS>;
          referenceline x=eval(coln(8,9,10)) / lineattrs=(color=lightgrey);
          drawtext textattrs=(size=6pt) "&freq001d2" / <DRAWTEXT OPTIONS>;
          drawtext textattrs=(size=6pt) "&freq005d2" / <DRAWTEXT OPTIONS>;
        endlayout;

        layout overlay;
          discretelegend "11"/ across=4 border=false valueattrs=(size=8pt);
        endlayout;
      endlayout;

    endgraph;
  end;
run;

proc sgrender data=plotdata template=dosinggraph;
run;
```

### LAYOUT OVERLAY OPTIONS

The <LAYOUT OVERLAY OPTIONS> control the appearance of the x and y axes, and the code for each LAYOUT OVERLAY OPTION is shown below (in order of appearance in the code above).

```
xaxisopts=(linearopts=(viewmin=0 viewmax=6 tickvaluelist=(0.5 1.5 2.5 3.5 4.5 5.5)
tickvalueformat=level.) display=(line ticks tickvalues label) label='Drug 1 Level')
yaxisopts=(display=(line ticks tickvalues))

xaxisopts=(linearopts=(tickvaluelist=(8.5 9.5) viewmin=8 viewmax=10
tickvalueformat=level.) display=(line ticks tickvalues label) label='Drug 2 Level')
yaxisopts=(display=(line))
```

## HIGHLOWPLOT OPTIONS

The <HIGHLOWPLOT OPTIONS> control the appearance of each segment of the line. This includes the segment colors and sheen, as well as the outline color and thickness. This code does not change between each HIGHLOWPLOT statement.

```
group=flagcolors type=bar display=(outline fill) outlineattrs=(color=black thickness=2)
barwidth=0.2 name="l1" dataskin=sheen
```

## DRAWTEXT OPTIONS

The <DRAWTEXT OPTIONS> control the appearance of the text and its location. The only changes between each DRAWTEXT statement are the values of X and Y to account for the varying annotations.

```
yspace=graphpercent y=35 xspace=datavalue x=0.25 widthunit=data justify=left anchor=bottomleft
```

## FIGURE 5 KEY FEATURES

### COMPLEX LAYOUT

The complex layout is implemented using GTL. The plot area has been divided into two rows by the outermost LAYOUT LATTICE statement. Using ROWWEIGHTS equal to .9 and .1, the first row is assigned 90% of the plot area, the second row 10%. Within the first row, the plot has been divided into two columns by using a LAYOUT LATTICE statement but this time setting COLUMNWEIGHTS equal to .75 and .25. For each column plot, a LAYOUT OVERLAY statement is assigned with a corresponding HIGHLOWPLOT. The second row merely defines the legend, which required its own row in order to center the legend underneath the two plots in the first row.

### ATTRIBUTE MAPPING

The DISCRETEATTRMAP assigns graphical attributes to specified data values. In this case, a bar color has been assigned to each of the possible values of the variable FLAG. To achieve this, the DISCRETEATTRMAP must be assigned a NAME and the values must be mapped to their graphical attributes. That NAME is then set to ATTRMAP in a DISCRETEATTRVAR statement. VAR is assigned to the variable in the data set that contains the mapped values, and a new ATTRVAR name is assigned. This ATTRVAR is then used within the plot statement.

### TEXT ANNOTATION

Using a series of DRAWTEXT statements, dosing information was annotated onto the figure for each subject. Text can be written anywhere on the graph with just a few options—there's no need to define an annotate data set. Since the text could change as the data change, the text of the annotation was made possible with predefined macro variables. In order to place the text in the correct location, YSPACE was chosen as GRAPHPERCENT and XSPACE was chosen as DATAVALUE. This means that a value of Y is equal to a percentage of the vertical graph space, running from 0 at the bottom to 100 at the top. Conversely, a value of X is equal to an actual value from the range specified on the x-axis.

## CONCLUSION

The HIGHLOW plot is a simple yet powerful tool for summarizing complex, longitudinal clinical trial data into a single figure. The examples explored in this paper illustrate the simplicity as well as the numerous possibilities of this procedure. While the HIGHPLOT is offered within both SGPLOT and GTL, there is significantly more customization available within the later. Now go forth and find out what HIGHLOW can do for you!

## REFERENCES

- SAS® 9.4 ODS Graphics Procedures Guide, Fifth Edition

## RECOMMENDED READING

- Graphically Speaking Blog (<http://blogs.sas.com/content/graphicallyspeaking/>)
- McConville, Kaitlyn, and Kristen Much. 2015. "Creating Sophisticated Graphics with GTL." *Proceedings of the PharmaSUG 2015 Conference*. Available at <http://www.pharmasug.org/proceedings/2015/DV/PharmaSUG-2015-DV02.pdf>.

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:

Name: Kristen Much  
Enterprise: Rho, Inc.  
Address: 6330 Quadrangle Dr.  
City, State ZIP: Chapel Hill, NC 27517  
Work Phone: 919-408-8000  
Fax: 919-408-0999  
E-mail: kristen\_much@rhoworld.com

Name: Kaitlyn Steinmiller  
Enterprise: Rho, Inc.  
Address: 6330 Quadrangle Dr.  
City, State ZIP: Chapel Hill, NC 27514  
Work Phone: 919-408-8000  
Fax: 919-408-0999  
E-mail: kaitlyn\_steinmiller@rhoworld.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

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