

**Paper SAS4321-2016**  
**Clinical Graphs Using SAS®**  
Sanjay Matange, SAS Institute Inc.

## **ABSTRACT**

Graphs are essential for many clinical and health care domains, including analysis of clinical trials safety data and analysis of the efficacy of the treatment, such as change in tumor size. Creating such graphs is a breeze with procedures from SAS® 9.4 ODS Graphics. This paper shows how to create many industry-standard graphs such as Lipid Profile, Swimmer Plot, Survival Plot, Forest Plot with Subgroups, Waterfall Plot, and Patient Profile using Study Data Tabulation Model (SDTM) data.

## **INTRODUCTION**

The SAS ODS Graphics system was first released eight years ago in 2008 with SAS 9.2 and it included the Statistical Graphics (SG) procedures and the Graph Template Language (GTL). This opened up a new way to create graphs in SAS, and the feature set of these tools has been growing steadily, making it easier for you to create graphs with every release. (The Statistical Graphics procedures has since been renamed ODS Graphics procedures. However, this document uses SG procedures.)

The ODS Graphics system supports the need of different audiences as follows.

- The analyst can get automatic graphs from procedures. No knowledge of graph syntax is required.
- The analyst can create graphs using ODS Graphics Designer. No knowledge of graph syntax is required.
- The graphics programmer can use the SG procedures to create graphs.
- The advanced graphics programmer can use GTL to create complex graphs.

Starting with SAS 9.3, new features were added for making clinical graphs. These features included cluster grouping, box plots on linear axes, data set based attribute maps for controlling the assignment of group attributes, and annotation for SG procedures for detailed customization of the graphs.

With SAS 9.4, creating clinical graphs became even easier with the advent of new features including the axis tables.

The goal of this paper is to show you how to create some of the commonly requested graphs through step-by-step examples. These include the Lipid Profile graph, Swimmer Plot, Survival Plot, Forest Plot, Waterfall Plot, and Patient Profile graph using the SAS 9.4 SGPLOT procedure. We will also see how such graphs can be created using SAS 9.3.

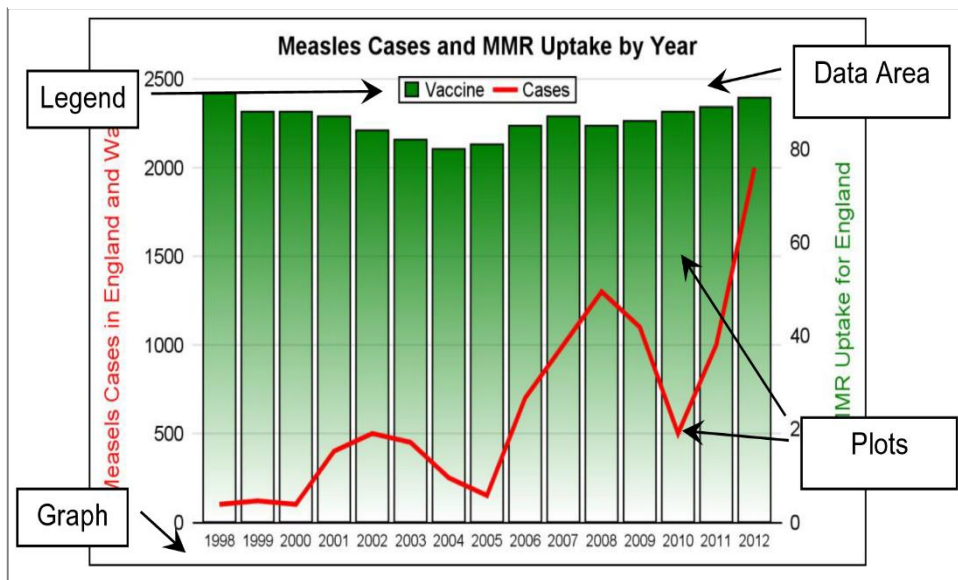
## **CREATING CLINICAL GRAPHS USING THE SGPLOT PROCEDURE**

Many commonly used clinical graphs are single-cell graphs that can be created using the SGPLOT procedure. A typical single-cell graph has the following components:

- Zero or more titles at the top of the graph.
- Zero or more footnotes at the bottom of the graph.
- One region in the middle often referred to as a "Cell" that displays the data.
- One or more plots used to display the data.
- A set of axes shared by the plots in the cell. A cell can have up to two horizontal and two vertical axes.
- Zero or more legends inside or outside the data region.
- Zero or more insets for display of relevant statistics inside or outside the data region.

We refer to the output from the graphical procedure as the "Graph". Every statement responsible to draw the data in the cell, whether it is a scatter plot or bar chart, is referred to as a "plot". Figure 1 shows a graph of the Measles and MMR Uptake by year created using the SGPLOT procedure.

Figure 1. Single-Cell Graph Using SGPLOT Procedure



The SGPLOT code to create the graph in Figure 1 is shown below.

```

title 'Measles Cases and MMR Uptake by Year';
proc sgplot data=Measles noborder;
  vbar year / response=vaccine nostatlabel y2axis
    fillattrs=(color=green) filltype=gradient
    baselineattrs=(thickness=0) baseline=0;
  vline year / response=cases nostatlabel
    lineattrs=(color=red thickness=3);
  keylegend / location=inside position=top linelength=15;
  yaxis offsetmin=0 display=(noline noticks) thresholdmax=0 max=2500 grid
    label='Measels Cases in England and Wales' labelattrs=(color=red);
  y2axis offsetmin=0 min=0 max=95 display=(noline noticks) thresholdmax=0
    label='MMR Uptake for England' labelattrs=(color=green);
  xaxis display=(nolabel noticks) valueattrs=(size=7);
run;

```

## A BRIEF REVIEW OF THE SGPLOT PROCEDURE

The structure of SGPLOT procedure syntax is as follows:

```

PROC SGPLOT <DATA=data-set> <SGANNO=data-set> <options>;
  plot-statement(s) required-parameters / <options>;
  <styleattrs-statement(s)>;
  <refline-statement(s)>;
  <dropline-statement(s)>;
  <inset-statement(s)>;
  <axis-statement(s)>;
  <keylegend-statement(s)>;
  <gradlegend-statement(s)>;
RUN;

```

The procedure statement supports multiple options. We will not attempt to describe each feature of the procedures. Instead, these features will become clear from the examples shown in this paper.

One or more plot statements can be used to represent the data. Each plot statement has its own set of required data-roles and options. These options will become evident as we create multiple clinical graphs. Many plot statements are supported, and can be grouped as shown below.

- Basic Plots: Such as scatter, series, and so on.
- Fit and Confidence Plots: Such as regression and loess plots.
- Distribution plots: Such as histograms and box plots.
- Categorical Plots: Such as bar charts and dot plots.

Supporting statements can be used to customize the graph.

- Style-attrs, symbol-char, and symbol-image statement.
- Reference lines and drop lines.
- Insets.
- Axes.
- Legends.

## Required Roles

A role name allows the assigned variable to be used in a specific way for the plot. Some common role names are 'X', 'Y', 'GROUP', 'CATEGORY', 'RESPONSE' and so on. Each plot statement has required roles and options needed to render the plot. Data set variables must be assigned to the required roles to produce a graph. Some required roles can take scalar values. Here are some examples:

```
SCATTER X=<var-name> Y=<var-name>;  
SERIES X=<var-name> Y=<var-name>;
```

Sometimes there is no role name per se, but a variable name still needs to be provided as shown below.

```
HISTOGRAM <var-name>;  
VBOX <var-name>;
```

## Optional Data Roles

Optional data roles can be provided for each statement that go after the "/". Data options are assigned variable names from the data set for rendering features that are data dependent, such as group classification, or color by response.

```
SCATTER X=<var-name> Y=<var-name> / GROUP=<var-name>;  
VBAR <var-name> / RESPONSE=<var-name> COLORRESPONSE=<num-var-name>;
```

## Plot Options

Plot options can be used to change the behavior of the plot, or to assign attributes for different parts of the plot. Each plot can have custom options that control the plot behavior and have names that are specific to the plot, such as MARKERCHAR for a scatter plot or MU and SIGMA for a density plot.

Plot options are used to customize the behavior or appearance of the plot, such as placement of the group values or to set the color of the line or shape of the marker symbol.

```
VBAR <var-name> / RESPONSE=<var-name> GROUP=<var-name> GROUPDISPLAY=CLUSTER;  
SCATTER X=<var-name> Y=<var-name> / MARKERATTRS=(SYMBOL=plus);
```

## Plot Layering

An important feature of the SGPLOT procedure is the ability to layer compatible plot statements to create more complex and intricate graphs. The SGPLOT procedure supports over thirty plot statements that are grouped in four groups as mentioned above. These are "Basic plots", "Fit and Confidence plots", "Distribution plots" and "Categorical plots". In general, plots can be layered as follows:

- "Basic Plots" can be combined with each other or with statements in the "Fit and Confidence plots".
- Plots in other groups can be combined with other plots in the same group.
- All plots can be combined with the "Supporting statements" like REFLINE and DROPLINE.
- Starting with SAS 9.4, box plots can be combined with Basic plots.

## CREATING CLINICAL GRAPHS USING SAS 9.4

In the section below, you will find examples of commonly requested clinical graphs created using the SAS 9.4 SGPLOT procedure. These graphs are easy to create using SAS 9.4 because some of the new statements and features have been specifically added to address the needs of such graphs. These include the XAXISTABLE and YAXISTABLE statements.

### MEDIAN OF LIPID PROFILE BY VISIT AND TREATMENT

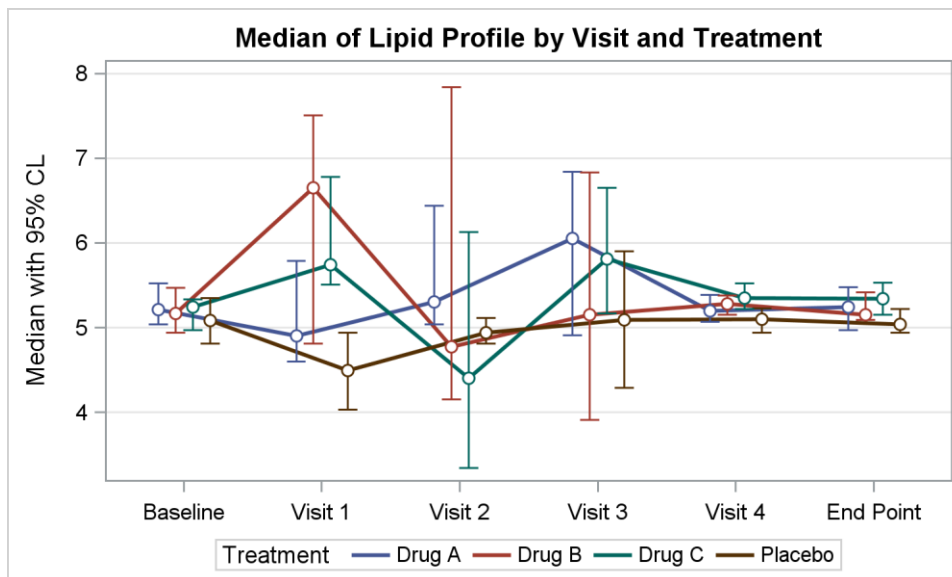
#### Lipid Profile on a Discrete Axis.

This graph displays the median of the lipid values by visit and treatment on a discrete x-axis. In this graph, the visits are at regular intervals and represented as discrete data. The values for each treatment are displayed along with the 95% confidence limits as adjacent groups using GROUPDISPLAY of "Cluster" and CLUSTERWIDTH=0.5.

The individual group values are connected across visits, which is useful to guide the eye across the graph. The slope of the line is not significant since the x-axis is discrete.

The style used for the graph is HTMLBlue, which is a "Color" priority style. This means for cycling of group attributes, first only the color is changed for each new group, holding the first marker symbol and the line style constant. After all 12 colors are used, the next marker symbol and line style are used.

**Figure 2. Median of Lipid Profile by Visit and Treatment on a Discrete Axis**



```

title 'Median of Lipid Profile by Visit and Treatment';
proc sgplot data=lipid_grp;
  series x=day y=median / lineattrs=(pattern=solid) group=trt name='s'
    groupdisplay=cluster clusterwidth=0.5 lineattrs=(thickness=2);
  scatter x=day y=median / yerrorlower=lcl yerrorupper=ucl group=trt
    groupdisplay=cluster clusterwidth=0.5
    errorbarattrs=(thickness=1) filledoutlinedmarkers
    markerattrs=(symbol=circlefilled)
    markerfillattrs=(color=white);
  keylegend 's' / title='Treatment' linelength=20;

```

```

yaxis label='Median with 95% CL' grid;
xaxis display=(nolabel);
run;

```

This graph displays the median of the lipid data by visit and treatment. The visits are at regular intervals and represented as discrete data. The median values for each treatment are displayed along with the 95% confidence limits as adjacent groups using `GROUPDISPLAY=Cluster` and `CLUSTERWIDTH=0.5`. The values across visits are joined using a series plot which also uses cluster groups with the same cluster width.

Two new SAS 9.4 options are worth noting.

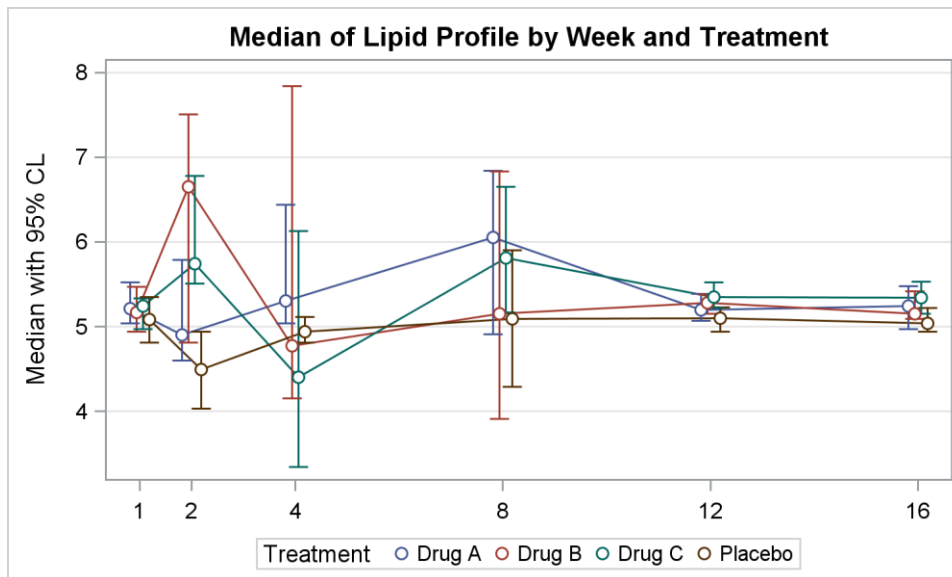
- **FILLEDOUTLINEDMARKERS option in the SCATTER statement:** This option allows filled markers such as `CircleFilled` to be drawn using a fill color for the interior and the contrast color for the outline. In this case, the interior is colored in white.
- **LINELength option in the KEYLEGEND statement.** When the patterns for the lines are all solid, as in this case, it is not necessary to have long line segments in the legend. The length of the line segment to be used in the legend can be set using this option.

### Lipid Profile on a Linear Axis.

When the intervals along the x-axis are numeric, it is often useful to display the data using a scaled linear axis. In this graph, the visits are at unequal time intervals. The values for each treatment are displayed along with the 95% confidence limits as adjacent groups using `GROUPDISPLAY` of "Cluster" and `CLUSTERWIDTH=0.5` using a Linear x-axis.

Now, each cluster of values is displayed at the time value starting with week 1. The first visit is at week 2, and the other visits are at week 4, 8, 12 and 16. The median data is shown at the correctly scaled linear distance from the origin. With a numeric x-axis, connecting the data along the x-axis provides more information about the rate of change represented by the slopes of the lines.

Figure 3. Median of Lipid Profile by Week and Treatment on a Linear Axis



```

title 'Median of Lipid Profile by Visit and Treatment';
proc sgplot data=lipid_grp;
  series x=day y=median / lineattrs=(pattern=solid) group=trt
  groupdisplay=cluster clusterwidth=0.5
  lineattrs=(thickness=2) name='s';
  scatter x=day y=median / yerrorlower=lcl yerrorupper=ucl group=trt
  groupdisplay=cluster errorbarattrs=(thickness=1)

```

```

filledoutlinedmarkers markerattrs=(symbol=circlefilled)
markerfillattrs=(color=white) clusterwidth=0.5;
keylegend 's' / title='Treatment' linelength=20;
yaxis label='Median with 95% CL' grid;
xaxis display=(nolabel);
run;

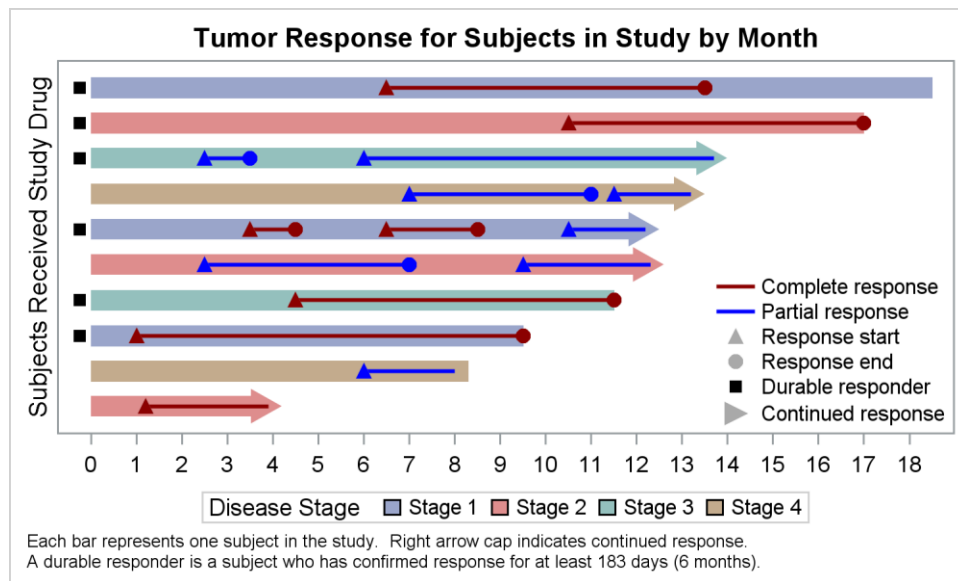
```

## SWIMMER PLOT

In her paper "Swimmer Plot: Tell a Graphical Story of Your Time to Response Data Using PROC SGPLOT", Stacey Phillips describes how investigators in oncology studies are frequently interested in the effects of a study drug on patients' tumor size and composition. Investigators want to know whether an individual subject has a response, and the timing of the response in relation to the study drug.

The Swimmer plot shown in Figure 4 is a graphical way of showing multiple pieces of a subject's response "story" in one graph.

**Figure 4. Swimmer Plot for Tumor Response**



```

title 'Tumor Response for Subjects in Study by Month';
proc sgplot data= swimmer datrmap=attrmap nocycleattrs;
  highlow y=item low=low high=high / highcap=highcap type=bar group=stage
    lineattrs=(color=black) name='stage' barwidth=1 nomissinggroup
    transparency=0.3 fill nooutline;
  highlow y=item low=startline high=endline / group=status
    lineattrs=(thickness=2 pattern=solid)
    name='status' nomissinggroup attrid=statusC;
  scatter y=item x=start / name='s' legendlabel='Response start'
    markerattrs=(symbol=trianglefilled size=8 color=darkgray);
  scatter y=item x=end / name='e' legendlabel='Response end'
    markerattrs=(symbol=circlefilled size=8 color=darkgray);
  scatter y=item x=xmin / name='x' legendlabel='Continued response '
    markerattrs=(symbol=trianglerightfilled size=12 color=darkgray);
  scatter y=item x=durable / name='d' legendlabel='Durable responder'
    markerattrs=(symbol=squarefilled size=6 color=black);
  scatter y=item x=start / group=status attrid=statusC
    markerattrs=(symbol=trianglefilled size=8);
  scatter y=item x=end / group=status attrid=statusC

```

```

markerattrs=(symbol=circlefilled size=8);
axis display=(nolabel) label='Months' values=(0 to 20 by 1) valueshint;
yaxis reverse display=(noticks novalues noline)
label='Subjects Received Study Drug';
keylegend 'stage' / title='Disease Stage';
keylegend 'status' 's' 'e' 'd' 'x' / noborder location=inside
position=bottomright across=1 linelength=20;
run;

```

**Figure 5. Data for the Swimmer Plot**

Obs	Item	Stage	Low	High	Highcap	Status	Start	End	Durable	Startline	Endline	Xmin
1	1	Stage 1	0	18.5		Complete response	6.5	13.5	-0.25	6.5	13.5	.
2	2	Stage 2	0	17.0		Complete response	10.5	17.0	-0.25	10.5	17.0	.
3	3	Stage 3	0	14.0	FilledArrow	Partial response	2.5	3.5	-0.25	2.5	3.5	.
4	3		0	14.0	FilledArrow	Partial response	6.0	.	.	6.0	13.7	.
5	4	Stage 4	0	13.5	FilledArrow	Partial response	7.0	11.0	.	7.0	11.0	.
6	4		0	13.5	FilledArrow	Partial response	11.5	.	.	11.5	13.2	.

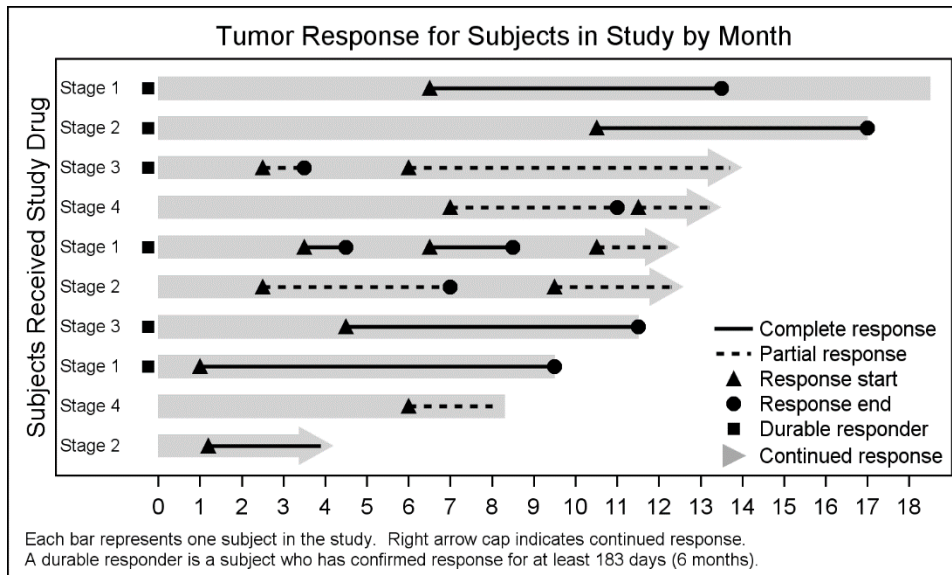
Note the following features of the program shown above.

- We use a HIGHLOW plot of TYPE=Bar of Low and High by Item and Stage to draw the main bars.
- Response duration is shown using a HIGHLOW plot of Start and End by Item and Status.
- We use overlaid SCATTER with TriangleFilled markers for Start by Item to populate the legend
- We use overlaid SCATTER with CircleFilled markers for the End by Item to populate the legend.
- We use overlaid SCATTER with TriangleRightFilled markers for the XMin by Item. This is used only to display a right triangle in the legend that represents the arrow head of the HIGHLOW plot.
- We use overlaid SCATTER with SquareFilled markers for Durable by Item.
- We use overlaid SCATTER with TriangleFilled markers for Start by Item by Status.
- We use overlaid SCATTER with CircleFilled markers for End by Item by Status.

## SWIMMER PLOT IN GRAY-SCALE

Often it is necessary to create a graph for inclusion in a report or journal where the graph needs to be rendered in gray-scale, as shown in Figure 6.

**Figure 6. Swimmer Plot in Gray-scale.**



In this case, the bar for each subject is displayed in gray, so it is not possible to use color to encode the Stage. In this graph, we have displayed the stage for each subject explicitly on the left side using the YAXISTABLE statement.

```
yaxistable stage / location=inside position=left nolabel;
```

A Discrete Attribute Map is used to set the attributes for the duration line and the start / end markers by Status using the "ATTRID" of StatusC for the color graph and StatusJ for the gray-scale graph. For the color graph in Figure 4, these are set to the red and blue colors. For the gray-scale graph in Figure 6, these use the solid or dashed line.

## PRODUCT-LIMIT SURVIVAL ESTIMATES PLOT

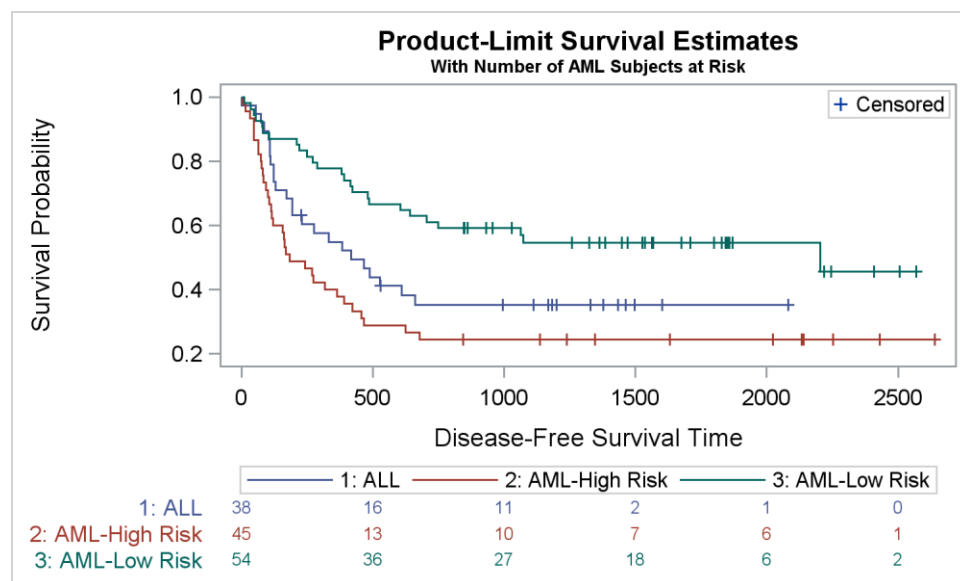
The survival plot is one of the most popular graphs that is customized to individual needs. For this example, I have run the LIFETEST procedure to generate the data for this graph. The output is saved into the "SurvivalPlotData" data set. The procedure itself creates this graph automatically. However, here the intention is to show how you can get this data and customize the graph to your specifications.

Here is the LIFETEST procedure code I have used that generates the data set needed to render the graph. The ODS OUTPUT statement is used to save the data in the "SurvivalPlotData" data set.

```
ods graphics on;
ods output Survivalplot=SurvivalPlotData;
proc lifetest data=sashelp.BMT plots=survival(atrisk=0 to 2500 by 500);
  time T * Status(0);
  strata Group / test=logrank adjust=sidak;
run;
```

Now we can use the SGPLOT procedure to create the Survival Plot using the "SurvivalPlotData" data set.

**Figure 7. Survival Plot**



```
title 'Product-Limit Survival Estimates';
title2 h=0.8 'With Number of AML Subjects at Risk';
proc sgplot data=SurvivalPlotData;
  step x=time y=survival / group=stratum name='s';
  scatter x=time y=censored / markerattrs=(symbol=plus) name='c';
  scatter x=time y=censored / markerattrs=(symbol=plus) GROUP=stratum;
  yaxistable atrisk / x=tatrisk class=stratum colorgroup=stratum;
```



```

keylegend 'c' / location=inside position=topright;
keylegend 's';
run;

```

**Figure 8. Data for the Survival Plot**

Obs	Time	Survival	AtRisk	Event	Censored	tAtRisk	Stratum	StratumNum
1	0	1.00000	38	0	.	.	1: ALL	1
2	0	.	38	.	.	0	1: ALL	1
3	1	0.97368	38	1	.	.	1: ALL	1
4	55	0.94737	37	1	.	.	1: ALL	1
5	74	0.92105	36	1	.	.	1: ALL	1

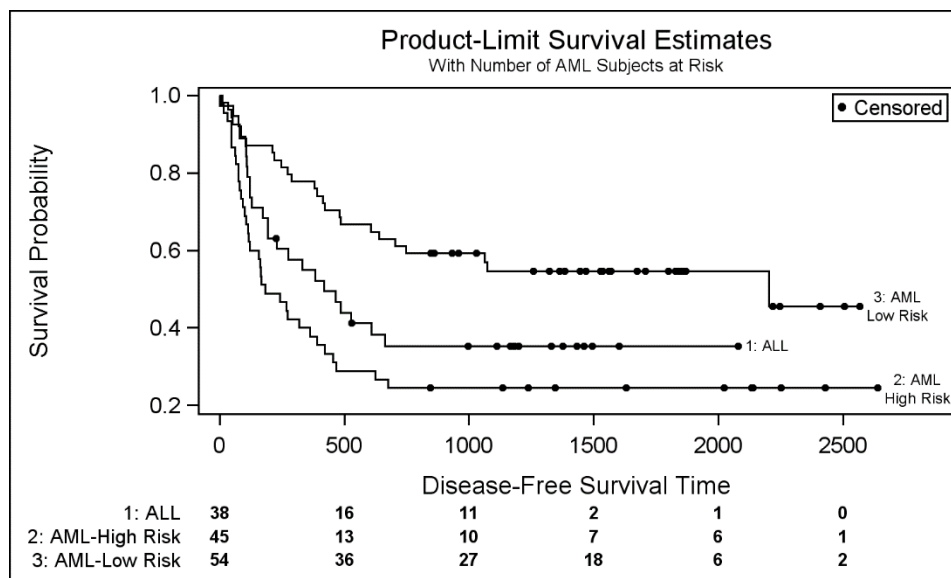
A few observations from the data set used for the graph are shown in Figure 8. The graph displays the survival probability using a STEP plot of Survival by Time and Stratum. The data has three distinct values for the Stratum column. Some key elements of the program are as follows:

- The survival curves are displayed using the STEP plot of Survival by Time and Stratum. The Stratum levels are displayed in the legend at the bottom of the graph.
- The censored observations are first displayed using a SCATTER by Time where all markers are set to "Plus". This displays all markers in black, which are included in the inner legend.
- The censored markers are over-plotted using a SCATTER by Time and Stratum with all markers set to "Plus". This displays the markers colored by Stratum, hiding the black markers.
- A XAXISTABLE is used to display the AtRisk values by TAtRisk. TAtRisk values are nonmissing only at increments of 500 on the x-axis. Hence, the table displays the risk values only at these locations.

### Survival Plot In Gray-scale

The graph shown in Figure 9 is the same Survival Plot rendered in gray-scale for inclusion in journals. We have used the JOURNAL style to render this graph.

**Figure 9. Survival Plot in Gray-scale**



Normally, the Journal style will use different line styles to represent the different group levels in the graph. In this case, the three values for "ALL", "AML Low-Risk" and "AML High-Risk" would be represented by

three different line patterns. However, use of line patterns is not optimal for step plots. So, in this example, I have set all the curves to have a solid pattern, which works well for step plots.

To identify each group level for the Stratum variable, I have used the CURVELABEL option. This labels each curve with its group value at the end of the curve. This turns out to be a good solution in this case, especially as SAS 9.4 supports splitting curve label values on "white space".

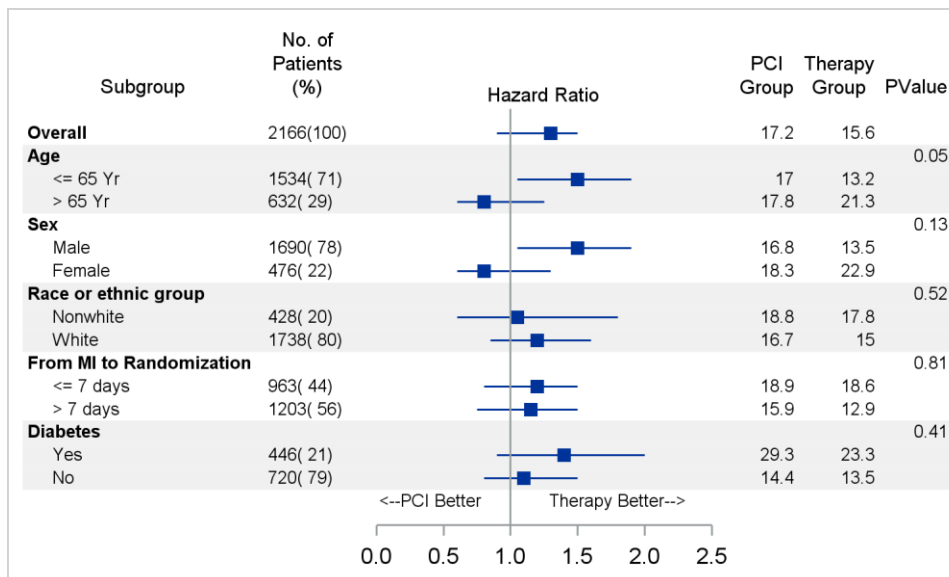
The same Stratum values are also displayed as labels on the left for the risk table. A CircleFilled marker is used for the Censored observations.

## FOREST PLOT WITH SUBGROUPS

A forest plot is a graphical representation of a meta-analysis of the results of randomized controlled trials. Normally, the graph consists of the Odds Ratio of the outcome by study along with display of study names, and relevant statistics for each study.

More recently, there has been an interest in such a graph where the information is displayed by subgroups, along with the relevant information as shown in Figure 10.

**Figure 10. Subgrouped Forest Plot**



The SGPLOT procedure code for this graph is shown below. The graph contains a hazard ratio plot in the middle created using high-low and scatter plots. The study values and statistics are displayed using axis tables.

```
proc sgplot data=forest_subgroup_2 nowall noborder nocycleattrs
  dattrmap=attrmap noautolegend;
  styleattrs axisextent=data;
  refline ref / lineattrs=(thickness=13 color=cxf0f0f0);
  highlow y=obsid low=low high=high;
  scatter y=obsid x=mean / markerattrs=(symbol=squarefilled);
  scatter y=obsid x=mean / markerattrs=(size=0) x2axis;
  refline 1 / axis=x;
  text x=x1 y=obsid text=text / position=center contributeoffsets=none;
  yaxistable subgroup / location=inside position=left textgroup=id
    labelattrs=(size=8) textgroupid=text indentweight=indentWt;
  yaxistable countpct / location=inside position=left
    labelattrs=(size=8) valueattrs=(size=7);
  yaxistable PCIGroup group pvalue / location=inside position=right
    labelattrs=(size=8) valueattrs=(size=7);
```

```

yaxis reverse display=none colorbands=odd
      colorbandsattrs=(transparency=1) offsetmin=0.0;
xaxis display=(nolabel) values=(0.0 0.5 1.0 1.5 2.0 2.5);
x2axis label='Hazard Ratio' labelattrs=(size=8)
      display=(noline noticks novalues);
run;

```

The data for this graph is shown in Figure 11 and the Discrete Attribute Map is shown in Figure 12.

**Figure 11. Data for Forest Plot with Subgroups**

Obs	ObsId	Id	Subgroup	indentWt	Count	Percent	Mean	Low	High	PCIGroup	Group	PValue	ref
1	1	1	Overall	0	2166	100	1.3	0.90	1.50	17.2	15.6		
2	2	1	Age	0								0.05	2
3	3	2	<= 65 Yr	1	1534	71	1.5	1.05	1.90	17.0	13.2		3
4	4	2	> 65 Yr	1	632	29	0.8	0.60	1.25	17.8	21.3		4
5	5	1	Sex	0								0.13	
6	6	2	Male	1	1690	78	1.5	1.05	1.90	16.8	13.5		
7	7	2	Female	1	476	22	0.8	0.60	1.30	18.3	22.9		

**Figure 12. Discrete Attribute Map for Column Attributes.**

Obs	id	value	textcolor	textsize	textweight
1	text	1	Black	7	bold
2	text	2	Black	5	normal

Here is a step-by-step description of how we built this graph using the SGPLOT procedure. While this does not look like a one-cell graph, it still has only one region displaying the data in a graphical format. The SAS 9.4 SGPLOT has new features that allow us to build this graph as a one-cell graph, since the procedure takes care of creating the multiple cells for us behind the scenes.

1. Note the graph has a clean table like appearance using the options NOBORDER and NOWALL.
2. The confidence range of the Hazard Ratio plot is displayed using a high-low plot of Low and High by ObsId.
3. The mean value of the Hazard Ratio plot is displayed using a scatter plot.
4. A reference line is drawn at x=1.
5. The annotation of "PCI Better" and "Therapy Better" are drawn using the text plot.
6. The subgroup and values are displayed on the left using a XAXISTABLE. Text attributes are controlled by the TEXTGROUP=ID option. The values for the text size and weight come from the Discrete Attribute Map shown in Figure 12. Regular values are displayed using 5 pt. normal font while the subgroups labels are displayed using 7 pt. bold font. The values are indented using the IndentWt column. The subgroup labels are not indented.
7. Count and percent values are displayed by another column on the left.
8. The statistics on the right are displayed by an XAXISTABLE of three columns. The labels are shown above each column.
9. The title "Hazard Ratio" is really the X2Axis label, which was enabled using the 2<sup>nd</sup> scatter plot with zero size markers.
10. Thick reference lines are used for every alternating 3 observations to help the eye across the graph.

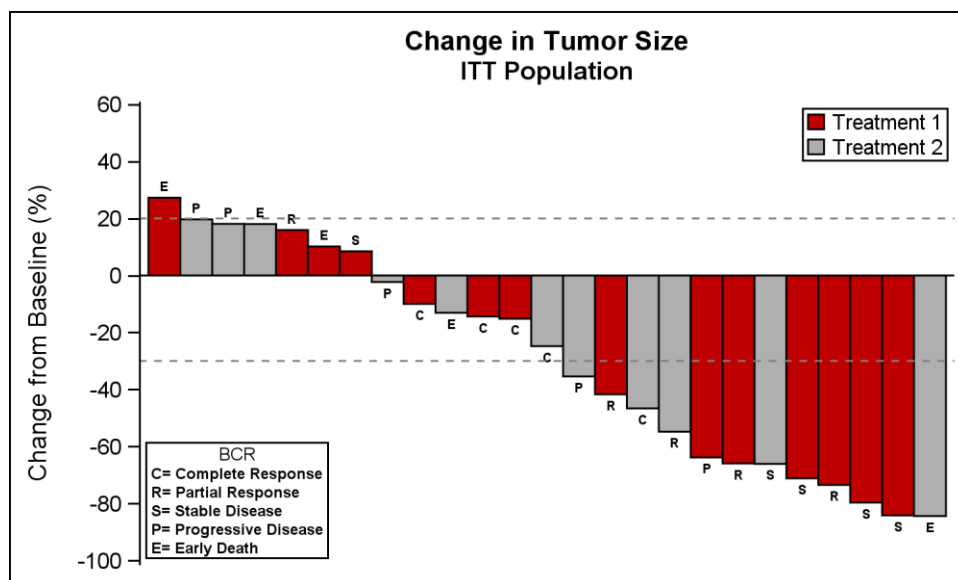
The new XAXISTABLE statement provides many flexible options to display tabular data on the left and right side of the graph. The statement creates the appropriate multi-cell LATTICE structure in the generated GTL code to place the tables. The width of each table is computed automatically based on the text attributes.

## WATERFALL CHART FOR CHANGE IN TUMOR SIZE

A waterfall chart is commonly used in the Oncology domain to track the change in tumor size for subjects in a study by treatment. The graph displays the change in tumor size for each subject in the study by descending percent change from baseline. A bar is displayed for each subject and the bar for the subject with maximum decrease is displayed on the right.

Each bar is classified by the treatment. The response category is displayed at the end of the bar. Reference lines are drawn at RECIST threshold of -30% and at 20%.

**Figure 13. Waterfall Chart for Change in Tumor Size**



The SGPLOT procedure code for this graph is shown below. The graph contains a hazard ratio plot in the middle created using high-low and scatter plots. The study values and statistics are displayed using axis tables.

```

title 'Change in Tumor Size';
title2 'ITT Population';
proc sgplot data=TumorSize nowall noborder;
  styleattrs datacolors=(cxbf0000 cx4f4f4f) datacontrastcolors=(black);
  vbar cid / response=change group=group categoryorder=respdesc
    datalabel=label datalabelattrs=(size=5 weight=bold)
    groupdisplay=cluster clusterwidth=1;
  refline 20 -30 / lineattrs=(pattern=shortdash);
  xaxis display=none;
  yaxis values=(60 to -100 by -20);
  inset "C= Complete Response" "R= Partial Response" "S= Stable Disease"
    "P= Progressive Disease" "E= Early Death" / title='BCR'
    position=bottomleft border textattrs=(size=6 weight=bold);
  keylegend / title='' location=inside position=topright across=1 border;
run;

```

**Figure 14. Data for Waterfall Chart**

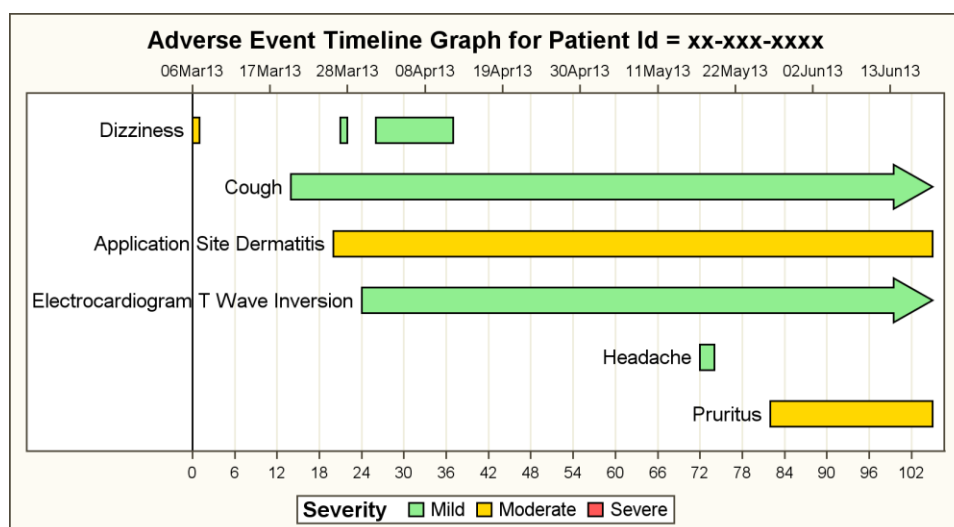
Obs	Cid	Change	Id	Group	Label
1	1	-14.3910	1	Treatment 1	C
2	2	-65.9578	2	Treatment 1	R
3	3	-71.1846	3	Treatment 1	S
4	4	19.6850	4	Treatment 2	P
5	5	18.0494	5	Treatment 2	N

The program uses a VBAR statement to draw the bars with CATEGORYORDER=RESPDESC. The data is plotted in descending order of the response.

### ADVERSE EVENT TIMELINE

The Adverse Event Timeline graph displays the adverse events for a specific subject by the adverse event and severity over time.

**Figure 15. Adverse Event Timeline**



The data for the graph is shown in Figure 16. The columns aeecode, aesev, stdate and enddate come from the SDTM AE domain.

The data might need some cleaning. If the enddate is missing, the highest value from the data is substituted, and the high-cap value is set to "FilledArrow". If an aeecode is repeated, the multiple events are displayed in one row. The aeecode is displayed as the low-label only once.

**Figure 16. Data for Adverse Event Timeline**

Obs	aeseq	aedecod	aesev	sev	stdate	enddate	stday	enday	highcap	lowlabel
1	1	Dizziness	Moderate	2	06MAR2013	06MAR2013	0	1		Dizziness
2	2	Cough	Mild	1	20MAR2013	18JUN2013	14	105	FilledArrow	Cough
3	6	Application Site Dermatitis	Moderate	2	26MAR2013	18JUN2013	20	105		Application Site Dermatitis
4	3	Dizziness	Mild	1	27MAR2013	27MAR2013	21	22		
5	4	Electrocardiogram T Wave Inversion	Mild	1	30MAR2013	18JUN2013	24	105	FilledArrow	Electrocardiogram T Wave Inversion
6	5	Dizziness	Mild	1	01APR2013	11APR2013	26	37		
7	7	Headache	Mild	1	17MAY2013	18MAY2013	72	74		Headache
8	8	Pruritus	Moderate	2	27MAY2013	18JUN2013	82	105		Pruritus

A discrete attribute map is used to ensure that the severity values are displayed using the colors defined in the map as shown in Figure 17. Green, Gold, and Red colors are used for severity of Mild, Moderate, and Severe.

Also note the column "Show" with values of "Attrmap". This causes all values from the map to be displayed in the legend even though some values might not be present in the data. So, even though the severity value of "Severe" is not present, it is displayed in the legend.

**Figure 17. Discrete Attribute Map**

Obs	Id	Show	Value	Fillcolor	Linecolor
1	Severity	Attrmap	Mild	lightgreen	darkgreen
2	Severity	Attrmap	Moderate	gold	cx9f7f00
3	Severity	Attrmap	Severe	lightred	darkred

## CREATING CLINICAL GRAPHS USING SAS 9.3

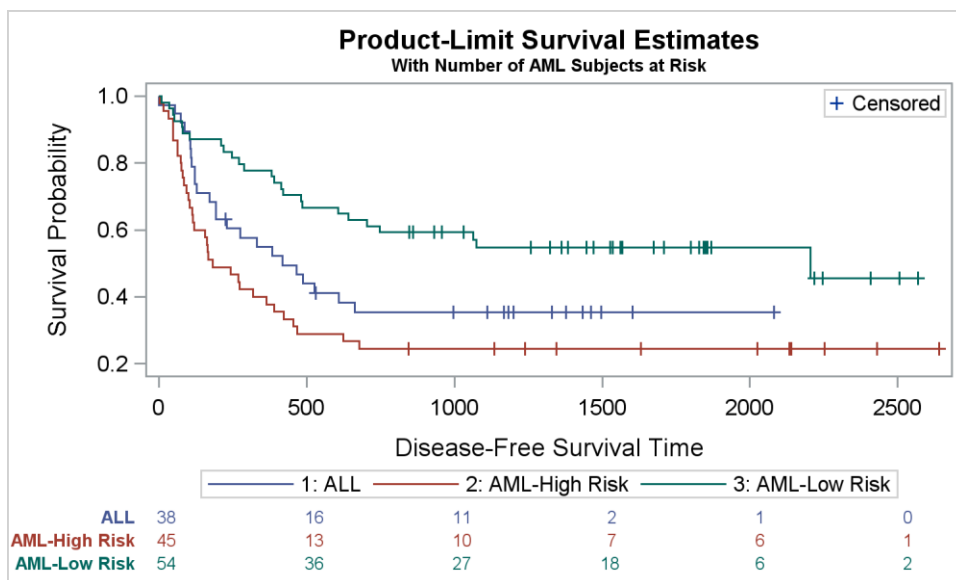
In the section above, we presented some examples of creating commonly requested clinical graphs using the SAS 9.4 SGPLOT procedure. These graphs are relatively easy to create using the new features released with SAS 9.4

However, many of you might have the SAS 9.3 release where you do not have access to these new features. Let us now examine how you can create some of these popular graphs using SAS 9.3 with the SG annotation feature. We will go through the process of creating the Survival Plot and the Subgrouped Forest plot using.

### PRODUCT-LIMIT SURVIVAL ESTIMATES PLOT – SAS 9.3

In Figure 7, I described the process to create a survival plot by creating the data using the LIFETEST procedure and then using the data to create the graph using the SAS 9.4 SGPLOT procedure. For this example, I will use the "SurvivalPlotData" data set created earlier to make the graph shown in Figure 18.

**Figure 18. Survival Plot Using Annotation**



```

title 'Product-Limit Survival Estimates';
title2 h=0.8 'With Number of AML Subjects at Risk';
proc sgplot data=SurvivalPlotData sganno=anno_out
    pad=(bottom=15pct left=6pct);
    step x=time y=survival / group=stratum name='s';
    scatter x=time y=censored / markerattrs=(symbol=plus) name='c';
    scatter x=time y=censored / markerattrs=(symbol=plus) GROUP=stratum;
    keylegend 'c' / location=inside position=topright;
    keylegend 's';
run;

```

The portion of the program for creating the survival curves along with the inside legend and the legend at the bottom are the same as used for Figure 9. The main difference is the display of the "Subjects At-Risk" table at the bottom of the graph. In Figure 18, this is done using SG Annotation.

Note the use of the procedure option SGANNO and PAD options. The SGANNO option provides the "Anno\_Out" data set that contains all the annotation functions needed to draw the table at the bottom. The PAD=(BOTTOM=15pct LEFT=6pct) option instructs the graph to reserve 15% of the height of the graph at the bottom and 6% of the width of the graph at the left. This space is left empty by the procedure, and the graph created by the plot statements is restricted to the rest of the graph area.

Figure 19 displays the first 3 observations from the "Anno\_out" data set used to display the values of the subjects at-risk by time and stratum. The code for creating this data set is shown later.

**Figure 19. Part of the Annotation Data Set for Display of the Values by Stratum.**

Obs	Function	X1Space	Y1Space	X1	Y1	Label	TextColor	TextSize	TextWeight	Anchor
1	text	datavalue	graphpercent	0	12	38	GraphData1:contrastcolor	7	Normal	center
2	text	datavalue	graphpercent	500	12	16	GraphData1:contrastcolor	7	Normal	center
3	text	datavalue	graphpercent	1000	12	11	GraphData1:contrastcolor	7	Normal	center

The instructions in each observation of the data set are interpreted as follows:

- The Function "Text" is used to draw the text in the "Label" column.

- The X location for the text uses draw space of "DataValue". The text is drawn in the "Data" context using "Value" units, ensuring the labels are correctly aligned with the X1 value on the x-axis.
- The Y location for the text uses draw space of "GraphPercent". The text is drawn in the "Graph" context, using "Percent" units. So, the values for the first Stratum are displayed at 12% above the bottom of the graph, using GraphData1.contrastcolor as the attribute. Values for other Stratum are drawn at 8% and 4% above the bottom of the graph.

Figure 20 displays the last 3 observations from the "Anno\_out" data set. These are used to display the Stratum value labels for each of the rows of the risk data.

**Figure 20. Part of the Annotation Data Set for Display of the Stratum Labels**

Obs	Function	X1Space	Y1Space	X1	Y1	Label	TextColor	TextSize	TextWeight	Anchor
19	text	wallpercent	graphpercent	-1	12	ALL	GraphData1:contrastcolor	7	Bold	Right
20	text	wallpercent	graphpercent	-1	8	AML-High Risk	GraphData2:contrastcolor	7	Bold	Right
21	text	wallpercent	graphpercent	-1	4	AML-Low Risk	GraphData3:contrastcolor	7	Bold	Right

The instructions in each observation of the data set are interpreted as described above. Note some additional features of how these values are interpreted:

- The function "Text" draws each text string from the column "Label".
- The X coordinate for each label is "-1" in "WallPercent" context with Anchor of "Right". This means that the text string is drawn to the left of the wall, with the text anchored on the right, so the string extends to the left of the wall, as much as is needed.
- Each label is drawn at 12%, 8%, and 4% above the bottom to line up vertically with the values, using the appropriate color.

```
data anno_out;
retain Function 'text' Y1Space 'graphpercent' Anchor 'Center';
retain TextSize 7;
length TextColor $25 Label $100 X1Space $12 TextWeight $6;
set SurvivalPlotData(keep=tAtRisk atRisk Stratum Stratumnum) end=last;

Width=10; Anchor='center'; X1Space='datavalue'; TextWeight='Normal';
if tAtRisk ne . then do;
  Label=put(atRisk, 5.0); X1=tatrisk;
  if stratumnum=1 then do;
    Y1=12; TextColor='GraphData1:contrastcolor'; end;
  else if stratumnum=2 then do;
    Y1=8; TextColor='GraphData2:contrastcolor'; end;
  else do;
    Y1=4; TextColor='GraphData3:contrastcolor'; end;
  output;
end;

if last then do;
  Width=20; TextSize=7; TextWeight='Bold';
  X1Space='wallpercent'; X1=-1; Anchor='Right';
  Y1=12; TextColor='GraphData1:contrastcolor'; Label='ALL';
  output;
  Y1=8; TextColor='GraphData2:contrastcolor'; Label='AML-High Risk';
  output;
  Y1=4; TextColor='GraphData3:contrastcolor'; Label='AML-Low Risk';
end;
```



```

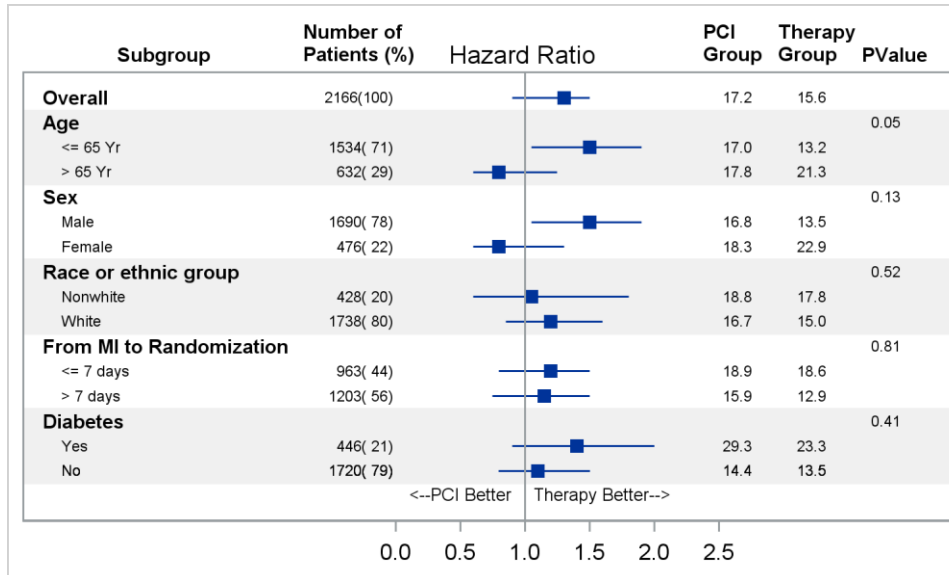
output;
end;
run;

```

### SUBGROUPED FOREST PLOT – SAS 9.3

In Figure 10, I described the process to create the subgrouped forest plot using the SAS 9.4 SGPLOT procedure. The key feature used in that graph was the new Axis Table, which is designed to handle the features needed to display the data columns with varying text attributes and indentation. Here we will build the same graph using SAS 9.3 SGPLOT code with annotation as shown in Figure 21.

**Figure 21. Subgrouped Forest Plot Using Annotation**



The SGPLOT procedure code to create the graph from the data set shown in Figure 22 is shown below. The code for the display of the hazard plot is pretty much same as for Figure 10. The high-low plot is used to display the confidence interval and the scatter plot to display the mean value.

**Figure 22. Data Set for Subgrouped Forest Plot**

Obs	countpct	Id	Subgroup	Count	Percent	Mean	Low	High	PCIGroup	Group	PValue	indentWt	ObsId	val	ref	xl	yl	text
11		1	From MI to Randomization	.	.	.	.	.	.	.	0.81	0	11	4	.	.	.	
12	963( 44)	2	<= 7 days	963	44	1.20	0.80	1.5	18.9	18.6	.	1	12	5	.	.	.	
13	1203( 56)	2	> 7 days	1203	56	1.15	0.75	1.5	15.9	12.9	.	1	13	0	.	.	.	
14		1	Diabetes	.	.	.	.	.	.	.	0.41	0	14	1	14	.	.	
15	446( 21)	2	Yes	446	21	1.40	0.90	2.0	29.3	23.3	.	1	15	2	15	.	.	
16	1720( 79)	2	No	1720	79	1.10	0.80	1.5	14.4	13.5	.	1	16	3	16	.	.	
17		.		.	.	.	.	.	.	.	.	1	16	.	.	1.6	17	Therapy Better-->
18		.		.	.	.	.	.	.	.	.	1	16	.	.	0.5	17	<--PCI Better

```

proc sgplot data=Forest2 nocycleattrs noautolegend
    sganno=anno pad=(top=6pct);
    reflate ref / lineattrs=(thickness=15 color=cxf0f0f0);
    highlow y=obsid low=low high=high;
    scatter y=obsid x=mean / markerattrs=(symbol=squarefilled);

```

```

scatter y=obsid x=mean / markerattrs=(size=0) x2axis;
refline 1 / axis=x;
refline &Rows / noclip lineattrs=(thickness=0);
scatter y=y1 x=x1 / markerchar=text;
yaxis reverse offsetmax=0 offsetmin=0 display=none;
xaxis display=(noline nolabel) offsetmin=0.4 offsetmax=0.25
      values=(0.0 0.5 1.0 1.5 2.0 2.5);
x2axis display=(noline noticks novalues) offsetmin=0.4 offsetmax=0.25
      label='          Hazard Ratio' ;
run;

```

In the code above, we have used an additional reference line at Y value of &ROWS (17), which is set to (number of rows +1) to add a blank line. In this space, we have drawn the annotation for "PCI Better" using the scatter plot with MARKERCHAR.

The five columns of data for Subgroup, Number of Patients, PCI Group, Therapy Group, and PValue are displayed using the annotation data set "Anno". The PAD option is used to reserve space for the column headings.

The annotation data set is created from the original data set as shown in Figure 23. Note, all the observations are not shown, as each column will have 16 observations to draw the text strings. Instead, I have created a "reduced" data set that shows 3 observations for each column in the graph, identified by the "Anno" and "AnnoType" columns. These columns are for description only.

**Figure 23. Annotation Data Set.**

Obs	Anno	AnnoType	Function	x1space	y1Space	label	x1	y1	Anchor	TextWeight	textsize	width
1	1	Subgroups	Text	WallPercent	datavalue	Overall	2	1.0	Left	Bold	8	50
2	1	Subgroups	Text	WallPercent	datavalue	Age	2	2.0	Left	Bold	8	50
3	1	Subgroups	Text	WallPercent	datavalue	<= 65 Yr	4	3.0	Left	Normal	6	50
4	2	CountPct	Text	WallPercent	datavalue	2166(100)	40	1.0	Right	Normal	6	50
5	2	CountPct	Text	WallPercent	datavalue		40	2.0	Right	Normal	6	50
6	2	CountPct	Text	WallPercent	datavalue	1534( 71)	40	3.0	Right	Normal	6	50
7	3	PCIGroup	Text	WallPercent	datavalue	17.2	75	1.0	Left	Normal	6	50
8	3	PCIGroup	Text	WallPercent	datavalue		75	2.0	Left	Normal	6	50
9	3	PCIGroup	Text	WallPercent	datavalue	17.0	75	3.0	Left	Normal	6	50
10	4	Therapy G	Text	WallPercent	datavalue	15.6	83	1.0	Left	Normal	6	50
11	4	Therapy G	Text	WallPercent	datavalue		83	2.0	Left	Normal	6	50
12	4	Therapy G	Text	WallPercent	datavalue	13.2	83	3.0	Left	Normal	6	50
13	5	PValue	Text	WallPercent	datavalue		91	1.0	Left	Normal	6	50
14	5	PValue	Text	WallPercent	datavalue	0.05	91	2.0	Left	Normal	6	50
15	5	PValue	Text	WallPercent	datavalue		91	3.0	Left	Normal	6	50
16	6	Headers	Text	WallPercent	WallPercent	Subgroup	10	100.8	BottomLeft	Bold	8	14
17	6	Headers	Text	WallPercent	WallPercent	Number of Patients (%)	30	100.8	BottomLeft	Bold	8	14
18	6	Headers	Text	WallPercent	WallPercent	PCI Group	73	100.8	BottomLeft	Bold	8	10

Part of the code for generating the annotation data set is shown below.

```

data anno;
  set forest2(keep= subgroup obsid id countpct PCIGroup group pvalue)
    end=last;
  length Anchor $10 y1Space $12;
  retain Function 'Text' x1space 'WallPercent' width 50;

```

```

retain TextWeight 'Normal';

y1=obsid; y1space='datavalue';

/*--Subgroups--*/
Anno=1; AnnoType='Subgroups';
label=subgroup; Anchor='Left';
x1=2; textweight='Bold'; textsize=8;
if id = 2 then do;
    x1=4; textweight='Normal'; textsize=6;
end;
output;

<Similar Code for other columns>

if last then do;
Anno=6; AnnoType='Headers';
y1space='WallPercent'; textweight='Bold'; textsize=8; width=14;
anchor='BottomLeft';
y1=100.8;
label='Subgroup'; x1=10; output;
label='Number of Patients (%)'; x1=30; output;
label='PCI Group'; x1=73; width=10; output;
label='Therapy Group'; x1=81; output;
label='PValue'; x1=90; output;
end;
run;

```

## CONCLUSION

This paper describes how you can create many commonly requested clinical graphs using the SAS 9.4 SGPLOT procedure. The SAS 9.4 version includes the XAXISTABLE and YAXISTABLE statements that are specifically designed to add axis aligned statistics to a graph.

The XAXISTABLE can be used to add one or more rows of textual data aligned with the x-axis as for a "Subjects At-Risk" table for a survival plot. The YAXISTABLE can be used to add one or more columns of textual data to a graph aligned with the y-axis, such as the statistics table for a forest plot.

All of the graphs discussed in this paper are easy to create using SAS 9.4. They can also be created with SAS 9.3 SGPLOT procedure with just a little extra effort using annotation.

## REFERENCES

- Matange, Sanjay. 2016. *Clinical Graphs Using SAS*. SAS Institute. Available at: [https://www.sas.com/store/prodBK\\_68179\\_en.html](https://www.sas.com/store/prodBK_68179_en.html)
- Matange, Sanjay and Heath, Dan. 2011. *Statistical Graphics Procedures by Example: Effective Graphs Using SAS*. SAS Institute. Available at: [https://www.sas.com/store/prodBK\\_63855\\_en.html](https://www.sas.com/store/prodBK_63855_en.html)
- Heath, Dan. 2011. "Now You Can Annotate Your Statistical Graphics Procedure Graphs" SAS Global Forum, Las Vegas. Available at <http://support.sas.com/resources/papers/proceedings11/277-2011.pdf>
- Heath, Dan. 2016. "Annotating the ODS Graphics Way." SAS Global Forum, Las Vegas. Available at <http://support.sas.com/resources/papers>
- Pandya, Niraj. 2012. "Waterfall Charts in Oncology Trials – Ride the Wave." PharmaSUG. Available at <http://www.pharmasug.org/proceedings/2012/DG/PharmaSUG-2012-DG13.pdf>

Phillips, Stacey. 2014. "Swimmer Plot: Tell a Graphical Story of Your Time to Response Data Using PROC SGPLOT." PharmaSUG, San Diego. Available at <http://www.pharmasug.org/proceedings/2014/DG/PharmaSUG-2014-DG07.pdf>

Matange, Sanjay. "Graphically Speaking." Available at <http://blogs.sas.com/content/graphicallyspeaking/>

## RECOMMENDED READING

- *Base SAS® Procedures Guide*
- *Statistical Graphics Procedures by Example: Effective Graphics using SAS®*

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author:

Sanjay Matange  
SAS Institute, Inc.  
100 SAS Campus Drive  
Cary, NC 27513  
Sanjay.Matange@sas.com  
<http://blogs.sas.com/content/graphicallyspeaking/>.  
<http://www.sas.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.