

Patient Profile Graphs Using SAS®

Sanjay Matange, SAS Institute Inc., Cary, NC

ABSTRACT

Patient profiles provide detailed information for a specific subject participating in a study. The report includes relevant data for a subject that can help correlate adverse events, concomitant medications, and other significant events as a narrative or a visual report.

This presentation covers the creation of the graphs that are useful for visual reports from CDISC data. It includes a graph of the adverse events by time and severity, graphs of concomitant medications, vital signs, and labs. All the graphs are plotted on a uniform timeline, so the adverse events can be correlated correctly with the concomitant medications, vital signs and labs. These graphs can be easily incorporated with the demographic and personal data of the individual patient in a report.

INTRODUCTION

Patient profiles come in various forms, describing the sequence of events for a specific patient in a study. This includes adverse events, concomitant medications, vital signs and lab test data for the patient over the duration of the study. It is now widely accepted that graphical representation of this data allows for easier dissemination of the data and might also lead to the detection of any associations between the adverse events, medications, vital signs, and the lab data.

In this paper, we will focus on the process of creating graphs that are effective in disseminating this information. To understand the correlation between the graphs, it is important the graphs have a common uniform time scale. In this presentation, we will create the graphs listed below using the SAS® 9.3 Statistical Graphics (SG) Procedures, which include new plot statements that are especially useful for such graphs. Similar graphs can also be created using the third maintenance release for SAS 9.2 SG Procedures with a bit more coding and creative usage of the plots.

PATIENT PROFILE GRAPHS

In this presentation, we will focus on building the following graphs using SAS data sets in the CDISC standard domain model. We will use the AE, CM, VS, and LB data sets to create the following graphs:

- Adverse Event Timeline showing all the adverse events by date and severity.
- Concomitant medications by date with AE severity events data.
- Vital Statistics by date with the AE severity events data.
- Lab test graphs by date with the AE severity events data.
- A frequency plot of concomitant medication.

We will create the above graphs using the SAS 9.3 SG procedures.

THE STATISTICAL GRAPHICS (SG) PROCEDURES

The SG procedures were first released with SAS 9.2 in the SAS/GRAPH product. These procedures provide an easy-to-use syntax to create analytical graphs. The procedures support a building block approach with a large set of plot statements that can be mixed and matched to create the graph you need.

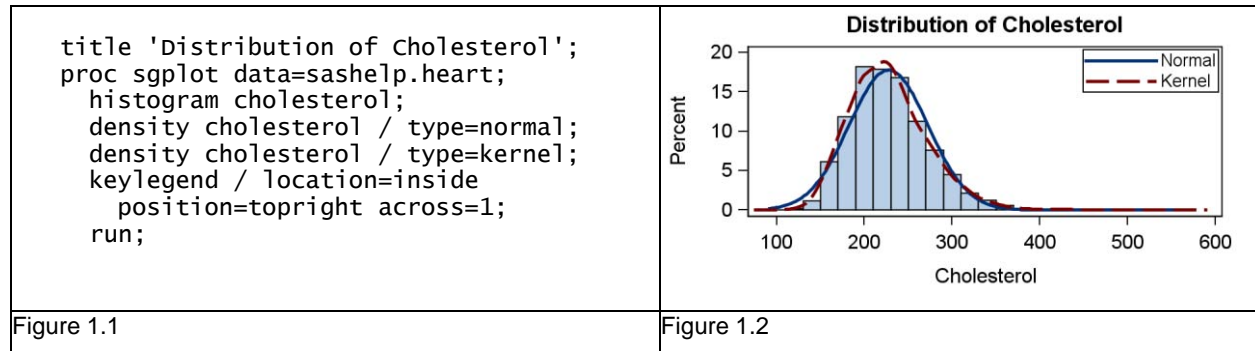
The SG Procedures include the following three procedures:

- SGPLOT procedure for single cell graphs
- SGPANEL procedure for classification panels
- SGSCATTER procedure for scatter plot panels and matrices.

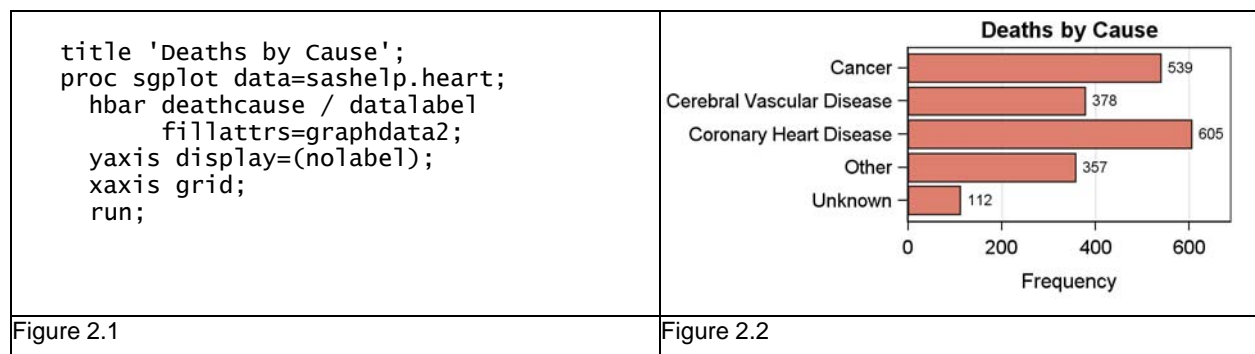
The SAS 9.3 versions of these procedures support many new features that make it far easier to create these graphs. In this paper, I will show the way to create these graphs using the SAS 9.3 SG Procedures. Similar graphs can be created using SAS 9.2 with some creative coding. Links to web pages showing the code needed to create similar graphs using the third maintenance release for SAS 9.2 are provided in the references section.

A QUICK REVIEW OF THE SG PROCEDURES

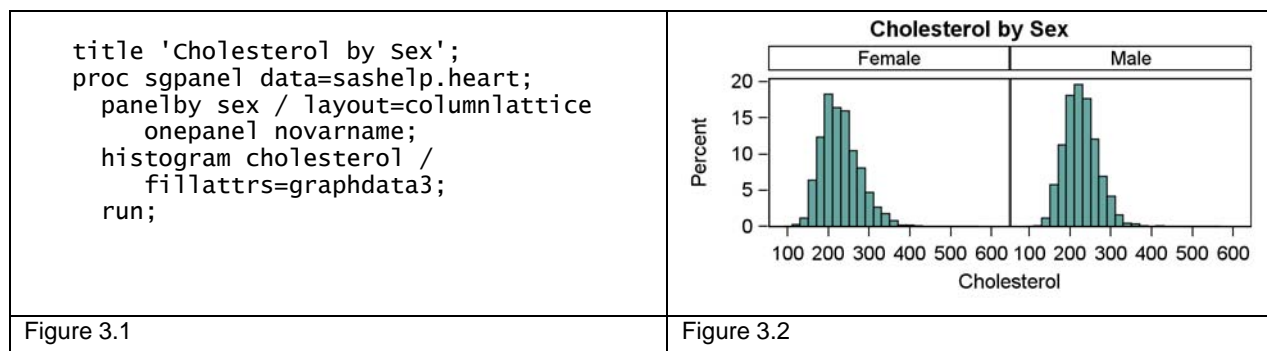
The **SGPLOT** procedure is ideally suited to create a single-cell graph with concise syntax. A single-cell graph, like the ones shown in Figures 1.2 and 2.2, contains a single region for displaying the data, with a set of x and y axes. In addition, the graph can contain titles, footnotes, and multiple legends. Here are some common examples of graphs created using the SGPLOT procedure along with the code:



In the graph shown in Figure 1.2, we have created a histogram of cholesterol, and then overlaid it with two density plots. The program needed to create this graph is shown in Figure 1.1. An example of a horizontal bar chart is shown below in Figure 2.2 with the code in Figure 2.1



The **SGPANEL** procedure creates a classification panel of plots using the information provided in the PANELBY statement as shown in Figure 3.1. In this case, the class variable is SEX. Because the data contains two distinct values for SEX, two cells are created, and populated using the plots specified after the PANELBY statement, in this case, a histogram.



Now that we know the basics of these procedures, let us use them to create the graphs needed for a Patient Profile.

THE ADVERSE EVENTS TIMELINE

For the Adverse Events timeline graph, we will use the AE data set. This data set contains many columns of data for all the subjects in a study. We will extract the relevant information for one patient (ID = 01-709-1217). The data needed to draw the graph is shown in Figure 4.

Obs	AESEQ	AESTDY	AEENDY	aesev	aedecod	highcap	aestdate
1	.	.	107.0	MILD			.
2	.	.	107.0	MODERATE			.
3	.	.	107.0	SEVERE			.
4	1	3	3.5	MODERATE	DIZZINESS		2013-03-06
5	2	17	107.0	MILD	COUGH	FilledArrow	2013-03-20
6	3	23	107.0	MILD	APPLICATION SITE DERMATITIS		2013-03-26
7	4	24	24.5	MILD	DIZZINESS		2013-03-27
8	5	27	107.0	MILD	ELECTROCARDIOGRAM T WAVE INVERSION	FilledArrow	2013-03-30
9	6	29	39.0	MILD	DIZZINESS		2013-04-01
10	7	29	39.0	MILD	DIZZINESS		2013-04-01
11	8	23	107.0	MODERATE	APPLICATION SITE DERMATITIS		2013-03-26
12	9	75	76.0	MILD	HEADACHE		2013-05-17
13	10	23	107.0	MODERATE	APPLICATION SITE DERMATITIS		2013-03-26
14	11	85	107.0	MODERATE	PRURITUS		2013-05-27

Figure 4 – Data Set for Adverse Events Timeline Graph

Note the following features in the data set above:

- The first three observations are added to ensure that all values are displayed in the legend.
- For observations that have a missing “aeendate”, the value of “Highcap” is set to ‘FILLEDARROW’.

The Adverse Events Timeline Graph by AESEQ:

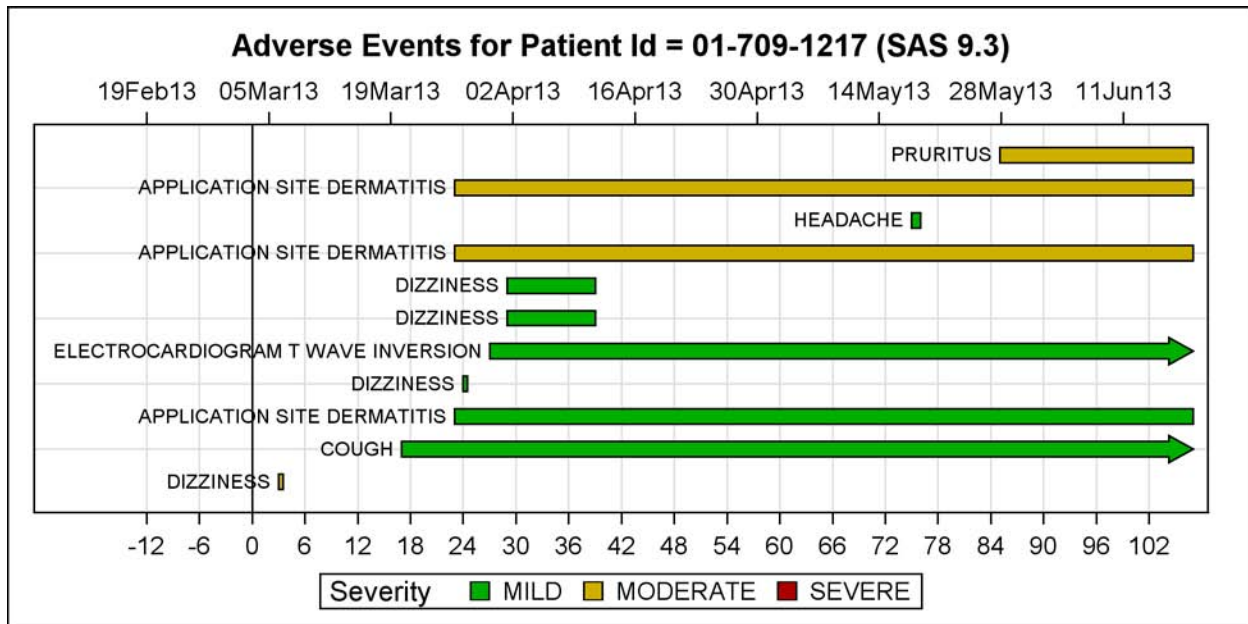


Figure 5 – Adverse Events Timeline Graph

SAS 9.3 SGPLOT Procedure code:

```
title "Adverse Events for Patient Id = &pid (SAS 9.3)";
proc sgplot data=AE_Cap dattrmap=attrmap;
  format aestdate date7.;
  refline 0 / axis=x lineattrs=(color=black);
  highlow y=aeseq low=aestdy high=aeendy / type=bar group=aesev
    lineattrs=(color=black pattern=solid) barwidth=0.8
    lowlabel=aedecod highcap=highcap attrid=Severity;
  scatter y=aeseq x=aestdate / x2axis markerattrs=(size=0);
  xaxis grid display=(nolabel) offsetmax=0.02
    values=(&minday2 to &maxday by 2);
  x2axis display=(nolabel) offsetmax=0.02 values=(&mindate2 to &maxdate);
  yaxis grid display=(noticks novalues nolabel);
run;
```

Figure 6 - SGPLOT Code for AE Timeline by AESEQ

For the graph shown in Figure 5, the start date for the graph was set at -12 as required by the user. Here are the salient features of the graph:

- All bars representing the adverse events are drawn using the HIGHLOW bar plot by “AESEQ”. If an adverse event is repeated, it is drawn again, as is the case for “Dizziness”.
- The HIGHLOW plot can have a “Line” or “Bar” type appearance. Here we set TYPE to BAR. This creates bars with fill colors and outlines. The range of each of the plot bars is set by the LOW and HIGH parameters. For this graph, we use the start and end day.
- Some events do not have an end date, indicating they are continuing past the duration of this graph. Such bars have an end cap. The HIGHLOW bar can have end caps at both ends, as specified by the LOWCAP and HIGHCAP options. In the DATA step, we have set the HighCap variable with the right value as seen in Figure 4.
- The name of the adverse event is displayed at the low end of the bar using the LOWLABEL option.
- The bars are grouped by “aesev”, the adverse event severity. The AE data set can have three severity levels of MILD, MODERATE, and SEVERE. For the case with the GROUP option, each bar is drawn using one of the GraphData style elements. More about this in the section labeled **Attribute Maps** below.
- The HIGHLOW plot is associated with the default X (bottom) and Y (left) axes, where the X axis shows the elapsed days since admission to the hospital. For this graph, we also want to see the actual dates for the events. To do that, we have used the SCATTER plot of AESTDATE by AESEQ, and associated this plot to the X2 (top) axis. This displays the data and the X2 axis at the top. The marker itself is set to a size of zero, so nothing extra is actually plotted in the graph. We did this only to obtain the top X2 axis.
- The SGPLOT axes use their own heuristics to determine a “good” range for each axis and to label the tick values with nice round numbers. So, each axis will decide to use its own values, possibly causing them to not be aligned with each other. To ensure correct alignment, we compute the min and max values for the day and date axes that are matched in macro variables, and apply these to each axis.
- We follow the principles of effective graphics and suppress the axis to reduce clutter in the graph.

Legend Entries: Sometimes it is necessary to display all the values of severity shown in the legend even if all the values are not in a particular data set, as in this case. To ensure this, we have added dummy observations to the data, ensuring that all severity values are in the data. The observations have missing values for start and end days, hence are not drawn, but are included in the legend. If this is not needed, you can omit the dummy observations.

Attribute Maps: When the GROUP option is used, each observation is drawn using one of the 12 GRAPHDATA style elements in the active style in the order in which they are encountered in the data. So, the first group value uses the GraphData1 style element, the second group uses GraphData2, and so on.

One way to ensure that the group values have the colors you want is to derive a custom style from one of the SAS shipped styles and set the values in the GraphData elements to the colors you want. Then, add the dummy

observations as discussed earlier to ensure that the group values are encountered in the order that you want.

A better way is to define a discrete attribute map, which specifies the visual attributes for each group by value. Figure 7 shows the attributes data set that specifies the fill color to be used with each of the severity values for the group variable. This data set is provided to the procedure using the DATTRMAP procedure option.

```
proc sgplot data=AE_Cap dattrmap=attrmap;
```

The “ID” column value of “SEVERITY” is provided to the HEATMAP statement, which identifies which values from the attribute map are to be used with the group variable on the plot statement.

Obs	id	value	fillcolor
1	Severity	MILD	Green
2	Severity	MODERATE	Yellow
3	Severity	SEVERE	DarkRed

Figure 7

Adverse Events by AE Name

The graph in Figure 5 shows each adverse event by sequence, and the same adverse event (like DIZZINESS) can be repeated. You can also display the events by name. This plots all adverse events of the same name in one horizontal location as shown in Figure 8 below.

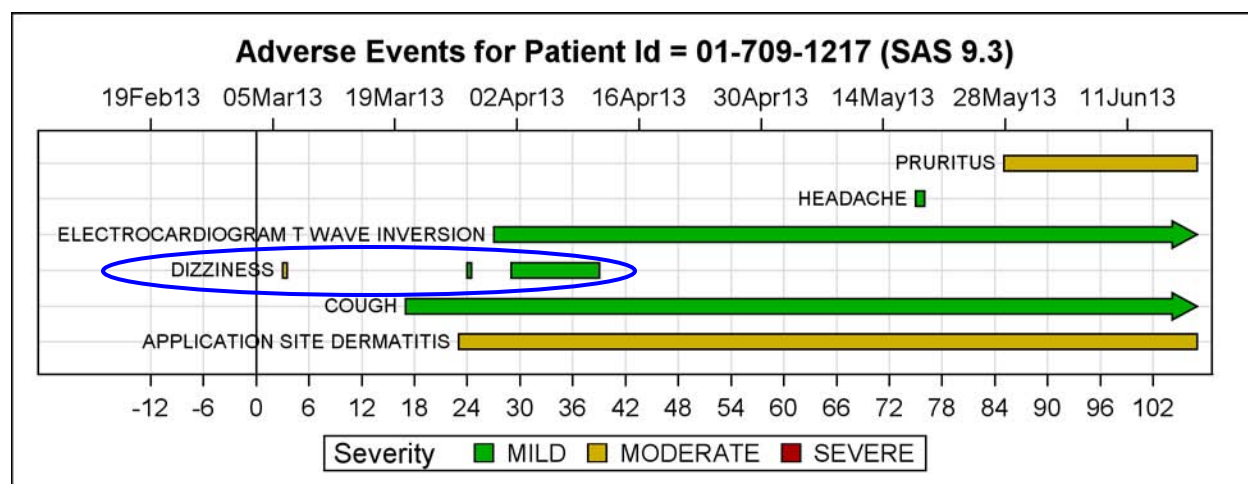


Figure 8

To ensure that each adverse event for DIZZINESS is not repeatedly labeled, we have created a new column “aename”, which has the value of “aecode” for the first occurrence of an adverse event name and missing for all subsequent occurrences of the same adverse event.

SAS 9.3 SGPLOT Procedure code:

```
title "Adverse Events for Patient Id = &pid (SAS 9.3)";
proc sgplot data=ae_by_name dattrmap=attrmap;
  format aestdate date7.;
  refline 0 / axis=x lineattrs=(color=black);
  highlow y=aecode low=aestdy high=aehendy / type=bar group=aesev
    lineattrs=(color=black pattern=solid) barwidth=0.8
    lowlabel=aename highcap=aehicap attrid=Severity;
  scatter y=aecode x=aestdate / x2axis markerattrs=(size=0);
  xaxis grid display=(nolabel) offsetmax=0.02 values=(&minday2 to &maxday by 2);
  x2axis display=(nolabel) offsetmax=0.02 values=(&mindate2 to &maxdate);
  yaxis grid display=(noticks novalues nolabel);
run;
```

Figure 9 - SGPLOT Code for AE Timeline by AENAME

THE CONCOMITANT MEDICATIONS TIMELINE

For the concomitant medications graph, we will use the CM data set. This data set contains data of concomitant medications for all the patients in the study. We will extract the relevant information for one patient (ID = 01-709-1217). The data needed to draw the graph is shown in Figure 10.

In extracting this data set, we have taken the following steps:

- Created a shortened name for the dosage units. For example, we use “Mg” instead of “Milligrams”.
- Concatenated the treatment, dosage, and shortened units into one medication name.
- Created one observation for each unique medication name.
- Added arrowheads for low cap and high cap if the medication continues past the study dates.
- Added start day values for each adverse event by severity.

Obs	cmstdate	CMMED	cmseq	cmstdy	cmendy	cmlocap	cmhicap	severe	moderate	mild
1	2013-03-30	BECONASE 1 SPRAY	0	27	107		FilledArrow	.	3	.
2	2013-02-16	GINKGO BILOBA 1 Tab	1	-12	-1	FilledArrow		.	.	17
3	2013-03-30	HYDROCORTISONE 1 Unit	2	24	107		FilledArrow	.	23	.
4	2013-02-16	HYTRIN 5 Mg	3	-7	107		FilledArrow	.	.	24
5	2013-05-25	IBUPROFEN 500 Mg	4	75	84			.	.	27
6	2013-02-16	LANOXIN 0.25 Mg	5	-7	107		FilledArrow	.	.	29
7	2013-02-16	MULTIVITAMIN 1 Tab	6	-7	107		FilledArrow	.	.	75
8	2013-03-30	VICKS FORMULA 44D 1 Tsp	7	17	28			.	85	.

Figure 10 – Data Set for Concomitant Medications

Now, each medication is plotted in the graph using the HIGHLOW plot as shown in Figure 11 below.

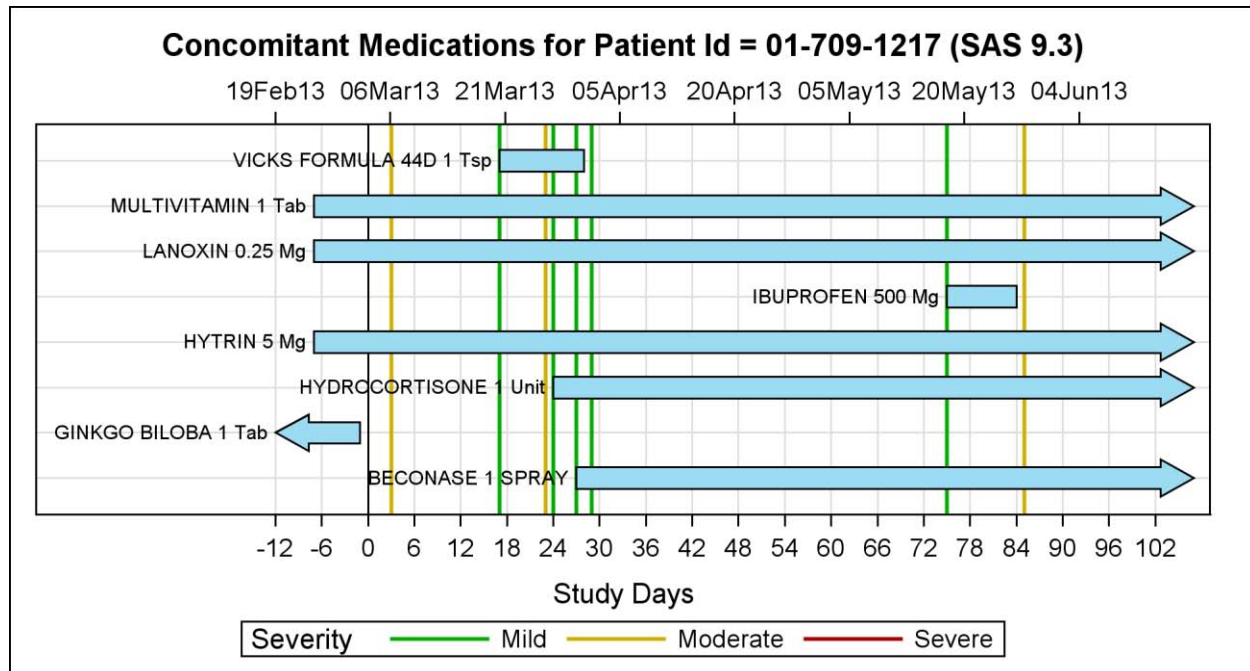


Figure 11 – Concomitant Medications Timeline

SAS 9.3 SGPLOT Procedure code:

```
title "Concomitant Medications for Patient Id = &pid (SAS 9.3)";
proc sgplot data=cmmeds noautolegend nocycleattrs;
  refline 0 / axis=x lineattrs=(thickness=1 color=black);

  refline mild / axis=x lineattrs=(pattern=solid thickness=2 color=cx00af00)
    name='mi' legendlabel='Mild';
  refline moderate / axis=x lineattrs=(pattern=solid thickness=2 color=cxcfaf00)
    name='mo' legendlabel='Moderate';
  refline severe / axis=x lineattrs=(pattern=solid thickness=2 color=cxaf0000)
    name='se' legendlabel='Severe';

  highlow y=cmseq low=cmstdy high=cmendy / type=bar barwidth=0.8 lowlabel=cmmed
    lowcap=cmlocap highcap=cmhicap fillattrs=(color=cx9BDCF2);
  scatter y=cmseq x=cmstdate / markerattrs=(size=0) x2axis;

  yaxis display=(nolabel noticks novalues) min=0;
  xaxis grid label='Study Days' offsetmax=0.02 values=(&minday2 to &maxday by 2);
  x2axis notimesplit display=(nolabel) offsetmax=0.02 tickvalueformat=date7.
    values=(&mindate2 to &maxdate);
  keylegend 'mi' 'mo' 'se' / title='Severity';
run;
```

Figure 12 - SGPLOT Code for Concomitant Medications Timeline

Synchronizing the X axes for AE and Concomitant Medications Graphs

Individually, the AE timeline graph shown in Figure 8 and the Concomitant Medications graph shown in Figure 11 each have the correct X axes. Each axis has an offset of the amount needed to allow the longest names to be drawn correctly. However, the lengths of the labels in each graph are different. Therefore, the X axes of the graphs are not aligned with each other.

This by itself is not a problem because each graph has its own correctly positioned axes. However, it would be preferable if we could align the X axes of the two graphs so that the graphs can be read easily when placed together in a stack.

There are two ways to achieve such synchronization as described below:

- Place the two graphs in a single container of two rows with uniform or common column axes.
 - We use the LAYOUT LATTICE statement, with two rows.
 - Each row contains the individual graphs.
 - The column axis is made uniform, so all the axis parameters are made uniform.
 - Preferably, common external horizontal axes can apply to both graphs.
 - We must use GTL to create this graph.
- Draw the data for both the adverse events and the medications in both graphs.
 - In this case, the axis in each graph contains data for the adverse events and medication.
 - The axes are correctly aligned to account for the longest data labels.
 - In the AE graph, make the medications transparent (invisible).
 - In the Con Meds graph, make the adverse events transparent.
 - Now, GTL is not required, and we can create each graph using the SGPLOT procedure.
 - We still have two separate graphs, but their X axes are uniform.

Figure 13 below shows both the adverse events and the medications in one graph. The statement drawing the medications is assigned a transparency of 1, so only the adverse events are displayed. Now you can see that the Adverse Events graph in Figure 13 has an X axis that is correctly aligned with the X axis of the medications graph shown in Figure 14, as indicated by the blue arrow.

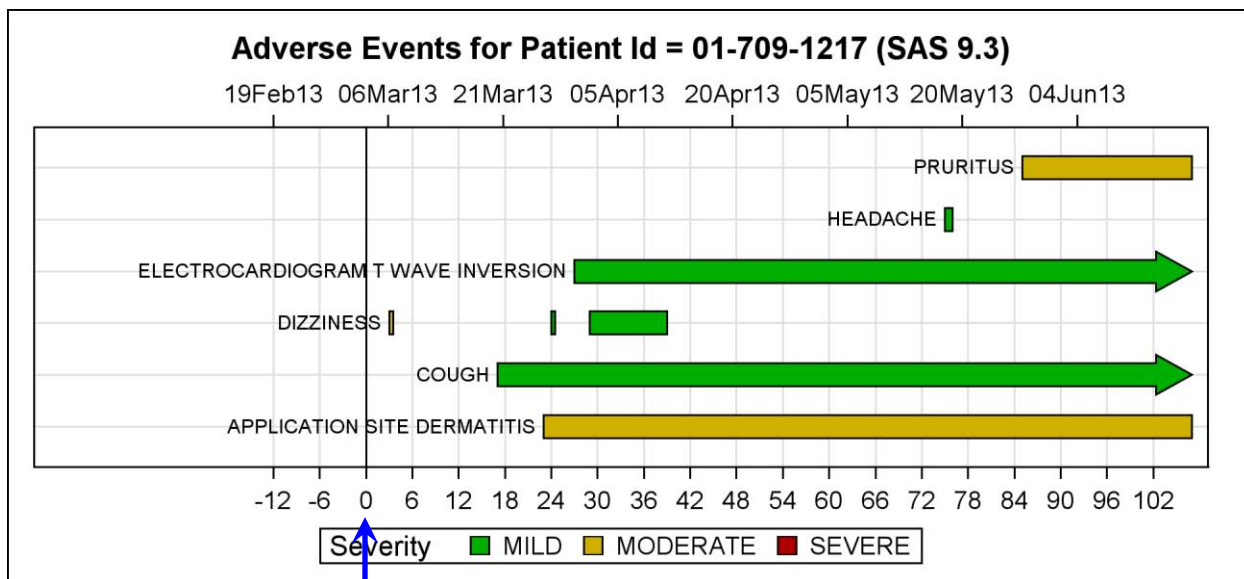


Figure 13 – Adverse Events Timeline Graph with Invisible Medications.

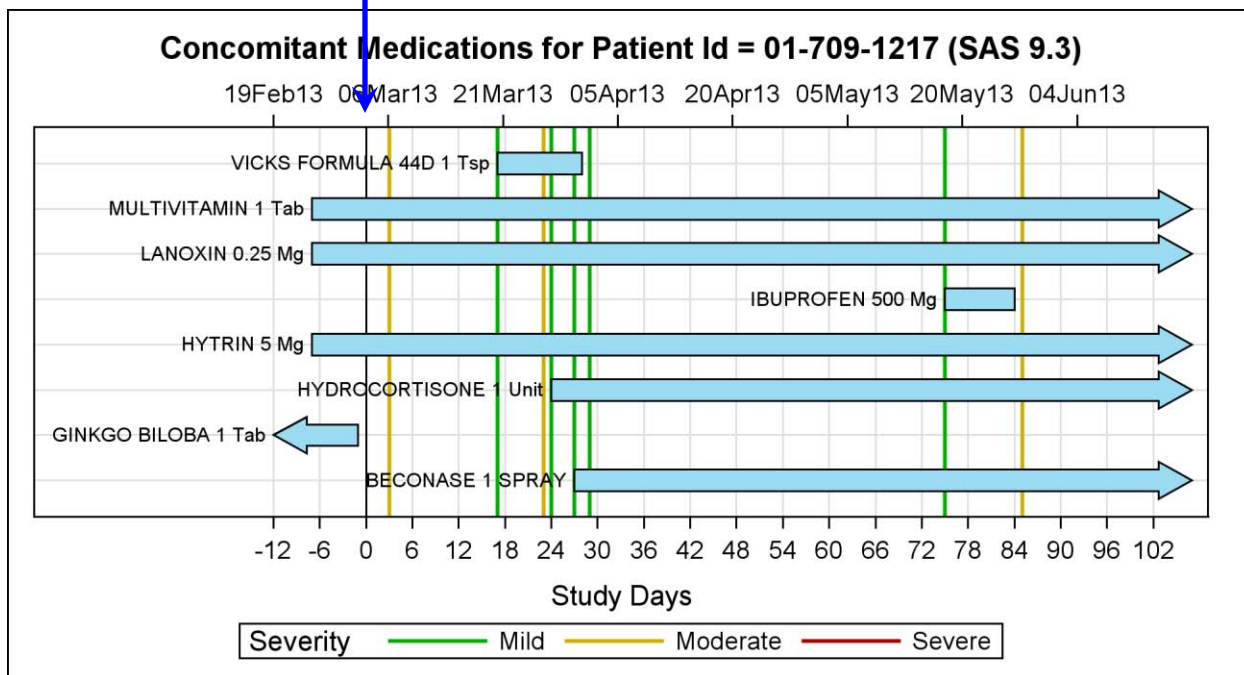


Figure 14 – Concomitant Medications Graph

Note the reference lines for each adverse event, color coded by severity, in Figure 14. These are now correctly aligned with the start of each adverse event in Figure 13.

THE VITAL STATISTICS GRAPH

For the vital signs graph, we will use the VS data set. In this case, we want to plot the Systolic, Diastolic, and Pulse values over time as a single paneled graph using the SGPPANEL procedure. The data is processed from the VS data set, and we extract the data shown in Figure 15.

We could use VSTEST as the class variable in the PANELBY statement. However, the panels are sorted alphabetically by class values, and "Pulse" will be shown in the middle of the two blood pressure measures. So, we use the "ID" variable with values on 1, 2, and 3 as the PANELBY variable so that the panel is sorted in the desired order. A format is used to convert ID values to VSTEST names.

id	VSDY	VSSTRESN	vstpt
Systolic	-15	132	AFTER LYING DOWN FOR 5 MINUTES
Systolic	-1	146	AFTER LYING DOWN FOR 5 MINUTES
Systolic	1	130	AFTER LYING DOWN FOR 5 MINUTES
Systolic	15	122	AFTER LYING DOWN FOR 5 MINUTES
Systolic	16	132	AFTER LYING DOWN FOR 5 MINUTES
Systolic	27	130	AFTER LYING DOWN FOR 5 MINUTES

Figure 15 – Data Set for Vital Signs

In addition to the data shown in Figure 15, we also add the events for drawing the reference lines for the Mild, Moderate and Severe AE events. The SGPPANEL program is shown in Figure 17.

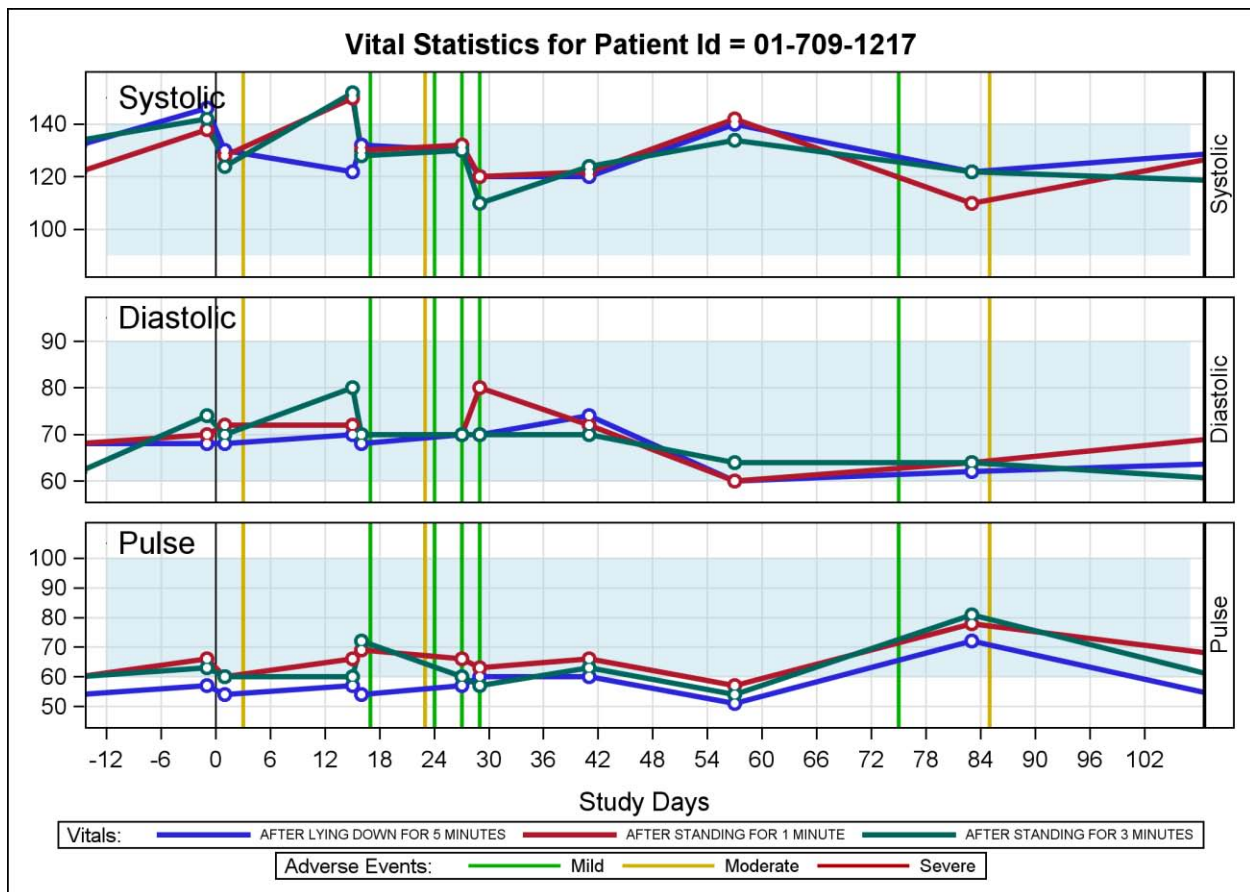


Figure 16 – Vital Signs Graph

SAS 9.3 SG PANEL Procedure code:

```
title "Vital Statistics for Patient Id = &pid";
proc sgpanel data=vs_with_events noautolegend nocycleattrs;
  panelby id / onepanel layout=rowlattice uniscale=column novarname spacing=10;
  band x=x lower=low upper=high / transparency=0.6 fill nooutline
      fillattrs=(color=lightblue);
  refline 0 / axis=x lineattrs=(thickness=1 color=black);
  refline mild / axis=x lineattrs=(thickness=2 color=cx00af00) name='mi'
      legendlabel='Mild';
  refline moderate / axis=x lineattrs=(thickness=2 color=cxcfaf00) name='mo'
      legendlabel='Moderate';
  refline severe / axis=x lineattrs=(thickness=2 color=cxaf0000) name='se'
      legendlabel='Severe';
  series x=vsdy y=vsstresn / group=vstpt lineattrs=(pattern=solid thickness=3)
      nomissinggroup name='bp' markerattrs=(symbol=circlefilled size=11);
  scatter x=vsdy y=vsstresn / markerattrs=(symbol=circlefilled size=11)
      group=vstpt;
  scatter x=vsdy y=vsstresn / markerattrs=(symbol=circlefilled size=5 color=white)
      group=vstpt;
  highlow y=yhigh low=xlow high=xhigh / highlabel=id labelattrs=(size=12);
  keylegend 'bp' / title='Vitals:' across=3 valueattrs=(size=5)
      titleattrs=(size=8);
  keylegend 'mi' 'mo' 'se' / title='Adverse Events:' across=3 valueattrs=(size=7)
      titleattrs=(size=8);
  rowaxis grid display=(nolabel) offsetmax=0.1;
  colaxis grid label='Study Days' offsetmax=0.02 values=(&minday2 to &maxday by 2);
run;
```

Figure 17 - SG PANEL Code for Vital Signs Timeline

- The VSTEST value is displayed for each graph in the cell heading along the right side. I have included this inside each graph at the top left for ease of reading. A HIGHLOW plot is used to do this. With SAS 9.3, there is no way to turn off the cell heading on the left side. This feature will be available in SAS 9.4.
- The ID variable with values of 1, 2, and 3 are used as the class variable in the PANELBY statement. This is done to ensure that the values are displayed in the preferred order. A format is used to convert those to Systolic, Diastolic, and Pulse.
- For each ID, the relevant "normal" values are added to the data set. These values are used to draw the bands that represent the normal levels.
- A color-coded reference line is drawn at the starting point of each adverse event.
- Two separate legends are displayed, one for the VS name and one for the AE severity.

THE LABS GRAPH

For the labs graph, we will use the LB data set. In this case, we want to plot all the lab values over time. The data set contains over 30 tests. To show the process, the graph is created for only a few of the lab tests.

The data for lab value by test and day is extracted from the LB data set by lab test name for the subject as shown in Figure 18. The data includes the normal high and low values for the test.

In addition to the data shown in Figure 18, we also add the events for drawing the reference lines for the Mild, Moderate and Severe AE events. The graph for three of the labs is shown in Figure 19, and the SG PANEL program is shown in Figure 20.

Obs	lbtestcd	LBDY	LBSTRESN	LBSTNRLO	LBSTNRHI	lbstnrmid
190	CREAT	-16	132.60	71	141	106
191	CREAT	16	150.28	71	141	106
192	CREAT	27	132.60	71	141	106
193	CREAT	41	132.60	71	141	106
194	CREAT	57	132.60	71	141	106
195	CREAT	83	132.60	71	141	106
196	CREAT	114	132.60	71	141	106

Figure 18 – Data Set for Lab Tests

In the graph below, we have displayed the portion of the lab test that lies outside the normal range for the test in red color. To do this, we first displayed the lab values as a band plot with a red fill color. Then, we overdraw an opaque band of the normal values. The red color shows only wherever the test values exceed the normal.

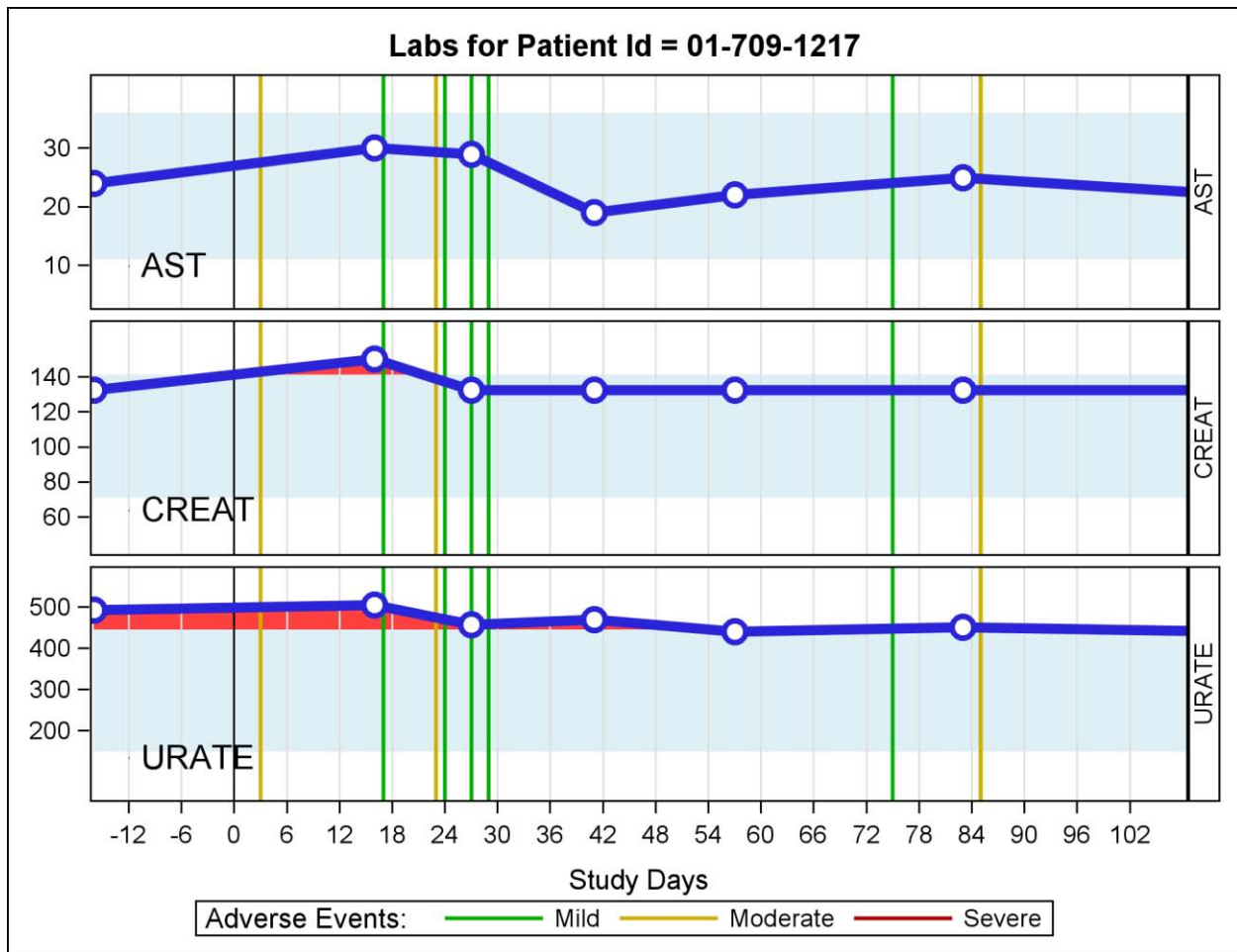


Figure 19 – Lab Test Graph

SAS 9.3 SG PANEL Procedure code:

```
title "Labs for Patient Id = &pid";
proc sgpanel data=lb2(where=(lbtestcd in ('AST' 'CREAT' 'URATE')) noautolegend;
panelby lbtestcd / columns=1 uniscale=column spacing=5 rows=3 layout=rowlattice
novarname /*rowheaderpos=left*/;
band x=lbdy lower=lbstnrmid upper=lbstresn / transparency=0.5
fillattrs=(color=red) fill outline;
band x=lbdy lower=lbstresn upper=lbstnrmid / transparency=0.5
fillattrs=(color=red) fill outline;
band x=lbdy lower=lbstnrlo upper=lbstnrhi / transparency=0
fillattrs=(color=white) fill;
band x=lbdy lower=lbstnrlo upper=lbstnrhi / transparency=0.6
fillattrs=(color=lightblue) fill;
refline -12 to 102 by 6 / axis=x lineattrs=graphgridlines;
refline 0 / axis=x lineattrs=(thickness=1 color=black);
refline mild / axis=x lineattrs=(thickness=2 color=cx00af00) name='mi'
legendlabel='Mild';
refline moderate / axis=x lineattrs=(thickness=2 color=cxcfaf00) name='mo'
legendlabel='Moderate';
refline severe / axis=x lineattrs=(thickness=2 color=cxaf0000) name='se'
legendlabel='Severe';
series x=lbdy y=lbstresn / markerattrs=graphdata1(symbol=circlefilled size=19px)
markers lineattrs=graphdata1(pattern=solid thickness=5);
scatter x=lbdy y=lbstresn /
markerattrs=(symbol=circlefilled size=11px color=white);
highlow y=yhigh low=xlow high=xhigh / highlabel=lbtestcd labelattrs=(size=12);
keylegend 'mi' 'mo' 'se' / title='Adverse Events:' across=3;
rowaxis display=(nolabel);
colaxis label='Study Days' offsetmax=0.02 values=(&minday2 to &maxday by 2);
run;
```

Figure 20 - SG PLOT Code for Lab Test Panel

Note the following salient points in this program:

- The LBTESTCD value is displayed for each graph in the panel along the right side. I have included this inside each graph at the bottom left for ease of reading. We use a HIGHLOW plot with HIGHLABEL to display this name. With SAS 9.3, there is no way to turn off the class value on the left side. This feature will be available in SAS 9.4.
- For each lab test, the relevant “normal” values are added to the data set. These values are used to draw the bands that represent the normal levels.
- A color-coded reference line is drawn at the starting point of each adverse event.
- The LBTESTCD name is displayed in the upper left of each cell for ease of readability. A HIGHLOW plot is used to draw the label. A feature to display insets will be supported in SAS 9.4.
- A legend is displayed for the AE severity.
- The portion of the lab results that exceed the normal values is shown in red. To do this, we first draw the lab values as a band plot from the lab value to the mid-level of the band. Then, we draw over this with an opaque band of the normal values. This band hides the part of the results that are within normal. Values that are outside the normal are displayed in red.

FREQUENCY OF MEDICATIONS PLOT

Finally, we create a frequency plot for all medications from the CM data set using the FREQ procedure. We will merge this data with the medications data set by treatment.

The frequencies are plotted using the HBAR statement in the SG PLOT procedure. The start and end dates are shown with each bar using the SCATTER statement with the MARKERCHAR option. The code is shown in Figure 21.

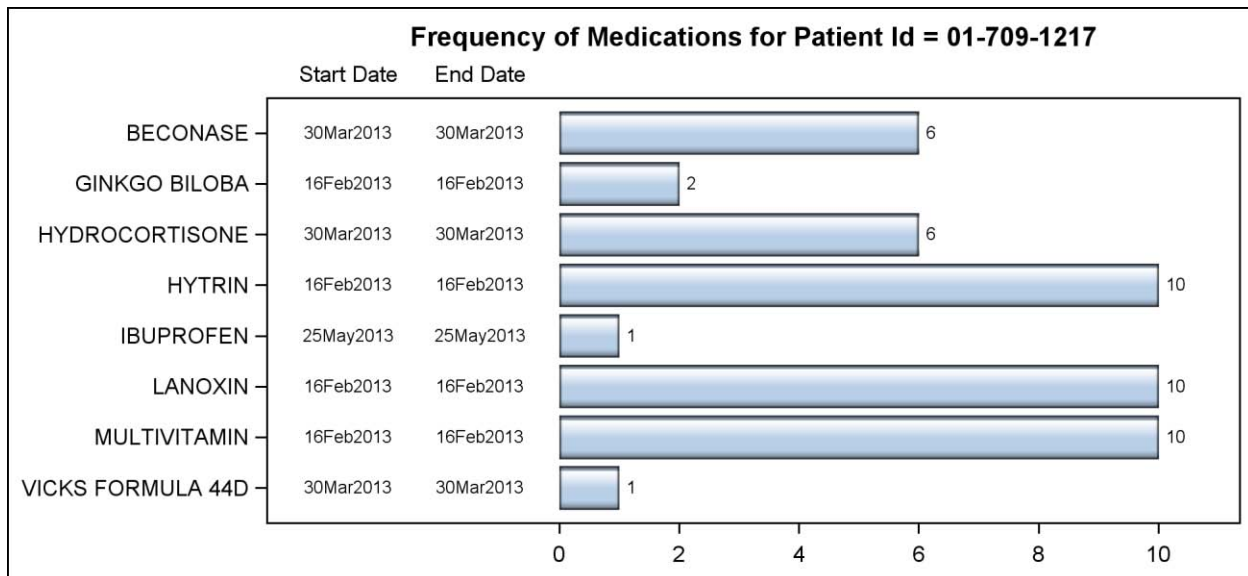


Figure 21 - Frequency of Medications Graph

SAS 9.3 SGPLOT Procedure code:

```
proc freq data=cm(keep=cmtrt) noprint;
  tables cmtrt / out=cmtrtf;
run;

/*--Merge frequency data with dates--*/
data cm3;
  format startdate enddate date9.;
  set meds(keep=cmtrt startdate enddate) cmtrtf;
  by cmtrt;
  startlabel='Start Date';
  endlabel='End Date';
run;
/*proc print; run;*/

/*--Medications frequency chart--*/
ods graphics / reset width=6.5in height=3in imagename="MedFreq" ;
title "Frequency of Medications for Patient Id = &pid";
proc sgplot data=cm3 noautolegend;
  hbarparm category=cmtrt response=count / dataskin=gloss datalabel;
  scatter y=cmtrt x=startlabel / markerchar=startdate x2axis;
  scatter y=cmtrt x=endlabel / markerchar=startdate x2axis;
  xaxis display=(nolabel) offsetmin=0.3;
  x2axis display=(nolabel noticks) offsetmax=0.78;
  yaxis display=(nolabel);
run;
```

Figure 22 - SGPLOT Code for Frequency of Medications

CONCLUSIONS

Development of pharmaceutical drugs and medical devices continues to accelerate, with new products entering safety and efficacy trials every year. The problem is that the data generated by the various trials is huge, and there is important information in the data that is hidden, or worse, possibly misinterpreted. How can we present the data to the clinicians in a way that is easily understood?

The SG Procedures provide the tools you need to create the graphs that deliver this data clearly. Here we have used the SAS 9.3 SGPLOT and SGPanel procedures to illustrate how to create graphs for Patient Profile reports.

You can download a copy of the code and data used in this paper from the SAS Technical Papers and Presentations site at <http://support.sas.com/rnd/papers/index.html>. Find the entry for "Patient Profile Graphs using SAS®" under the section for SAS Presentations at SAS Global Forum 2013, and download the examples. The code in this paper was tested using SAS® 9.3 software.

REFERENCES

Faye Yeh and Melvin Munsaka, (2012). "Programmed Assisted Patient Narratives (PANs) with New SAS® ODS and Graphical Approaches". MWSUG 2012, Paper PH04.

Sanjay Matange 2012. *Graphically Speaking: [A Better AE Timeline](#) using the third maintenance release for SAS 9.2*

RECOMMENDED READING

Sanjay Matange, 2012. "[Quick Results with SAS® ODS Graphics Designer](#)" *Proceedings of the SAS Global Forum 2012 Conference*. Cary, NC: SAS Institute Inc. Available at <http://support.sas.com/resources/papers/proceedings12/153-2012.pdf>.

CONTACT INFORMATION

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

Name:	Sanjay Matange
Enterprise:	SAS Institute Inc.
Address:	S3014 SAS Campus Dr.
City, State ZIP:	Cary, NC 27513
Work Phone:	(919) 531- 6753
Fax:	(919) 531-4444
E-mail:	Sanjay.Matange@sas.com
Web:	http://blogs.sas.com/content/graphicallyspeaking/

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.