Napoleon Plot for PharmaSUG

Kriss Harris, SAS Specialists Ltd, Hertfordshire, United Kingdom

ABSTRACT

Do you want to produce a very useful plot assessing treatment safety? Okay, do you want to produce a plot that for each subject shows the number of treatment cycles, the number of days on treatment, the doses that were received, whether the subject has discontinued treatment, and the cohort the subject is in? This paper will demonstrate how to do the above in SAS® 9.4.

INTRODUCTION

A Napoleon Plot is very useful at assessing treatment safety. From looking at a Napoleon Plot you can determine the number of cycles and days that each subject has been exposed to treatment, if a subject has needed to reduce the dose of their treatment for safety reasons, and which subjects are still on treatment or the reasons why a subject has discontinued.

The very useful Napoleon Plot was introduced to me in R, approximately 2 years ago and was created by the author Anna McGlothlin. I was intrigued by it, and challenged myself to produce a version within SAS. Admittedly I did struggle to produce the Napoleon Plot in SAS 9.2, and this was because I found it difficult to show that a subjects dose had been reduced between visits. However, when I received SAS 9.4 earlier this year that I was able to persevere with creating the Napoleon Plot. I was able to finally produce the Napoleon Plot because of the DISCRETEATTRMAP facilities that are present in SAS 9.4 (and also in SAS 9.3) but not in SAS 9.2. The improvements in my technical knowledge of Graph Template Language, and the motivation I had to produce an instance of the plot within SAS also helped me to produce the Napoleon Plot in SAS.
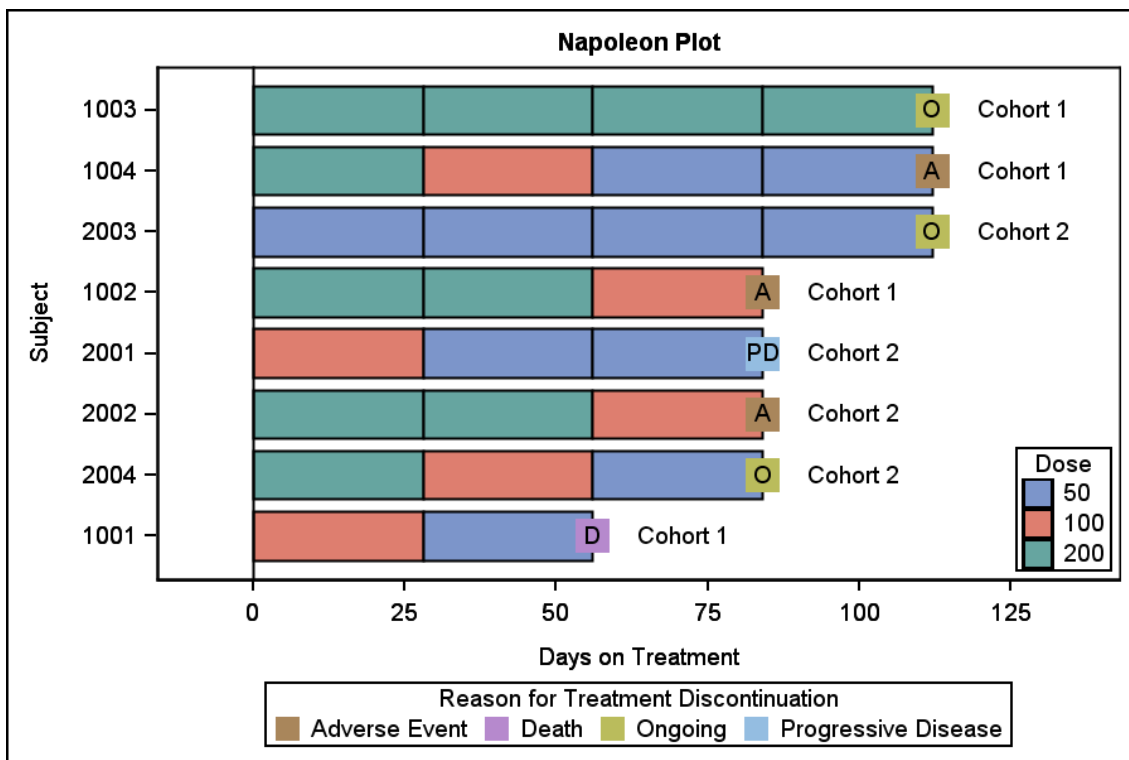
Figure 1: Napoleon Plot

Figure 1 above was created in GTL using the DISCRETEATTRMAP, DISCRETEATTRVAR, LAYOUT OVERLAY, BARCHARTPARM, SCATTERPLOT and DISCRETELEGEND statements. Figure 1 is created in four steps. Step 1 creates the stacked bar charts and the dose legend. Step 2 creates the square markers on the stacked bar chart that

denote the reason for treatment discontinuation. Step 2 also creates the reason for treatment discontinuation legend. Step 3 plots the initials of the word for the reason for treatment discontinuation, and step 4 produces the cohort information that each subject is in. Figures 2 – 5 below show the four steps that were taken. You can see that in Figure 3 the additional statements that are used in GTL compared to Figure 2 are DiscreteAttrMap, DiscreteAttrVar, Scatterplot and DiscretedLegend. The additional statements create the reason for treatment discontinuation legend, and display the markers of reason for treatment discontinuation on the plot. The red ovals on Figure 3 highlight the additional items on the plot compared to Figure 2. Similarly, the red ovals identify the additional items on the plot in Figure 4 and Figure 5.
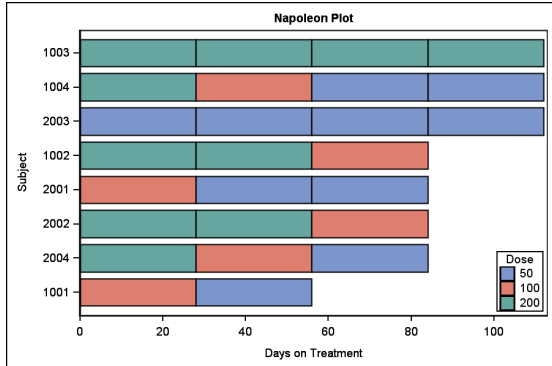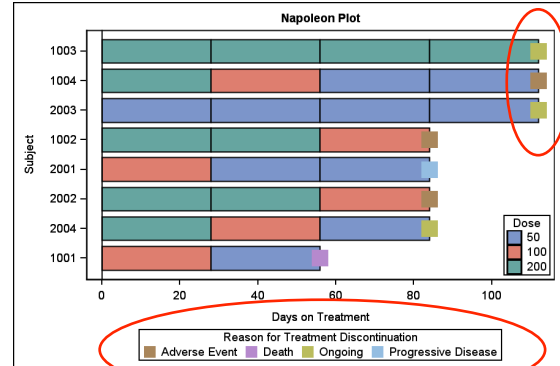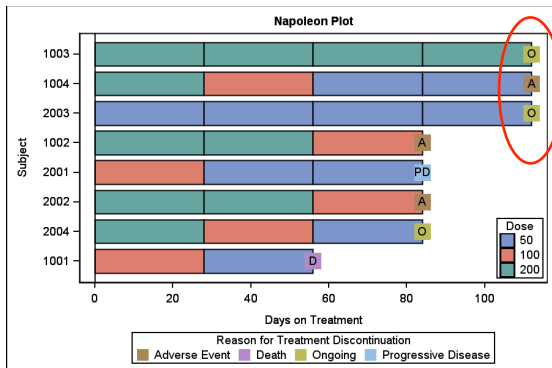
Figure 2 : Napoleon Plot – Step 1

Figure 3: Napoleon Plot – Step 2

Additional Statements:

DiscreteAttrMap
DiscreteAttrVar
Scatterplot
DiscreteLegend

Figure 4: Napoleon Plot – Step 3

Additional Statements:

Scatterplot

Figure 5: Napoleon Plot – Final Step
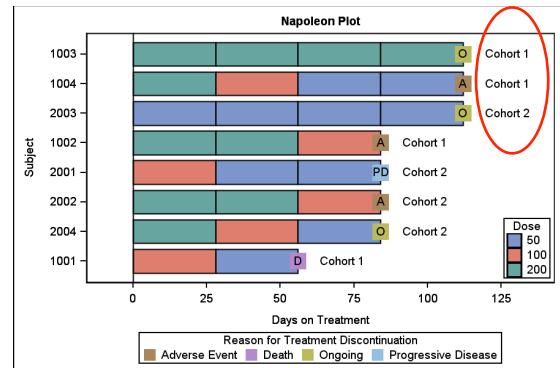
Additional Statements:

Scatterplot

THE PLOT

Assuming Study Data Tabulation Model (SDTM) data is being used, the data that is used in the Napoleon Plot can be extracted from the DS, EX, SV, and DM domains.

The table below is an example of the data that is used in Figure 2. In order to clearly identify subjects that have been dose reduced the bar charts need to be stacked at each visit, and hence the data should be in a format to allow that as shown below. Subject 1004 is an example of dose reduction as their dose has decreased from 200 at visit 1 to 100 at visit 2.

Table 1: Subset of Data used in Figure 2

	subjid	cycledays	dose	visit_dose	order_subject
1	1003	28	200	1-200	1
2	1003	28	200	2-200	1
3	1003	28	200	3-200	1
4	1003	28	200	4-200	1
5	1004	28	200	1-200	2
6	1004	28	100	2-100	2
7	1004	28	50	3-50	2
8	1004	28	50	4-50	2

Figure 2, shows the number of cycles and the number of days each subject has been on treatment. As mentioned above it also shows the dose the subject is on, and hence can show if a subject has had their dose reduced. Now I am going to describe the GTL code that was used to create the figure.

DISCRETE ATTRMAP AND DISCRETEATTRVAR

```
DiscreteAttrMap name="Dose_Group";

      Value "50" /fillattrs=(color= GraphData1:color) lineattrs=(color=black
pattern=solid);
      Value "100" /fillattrs=(color= GraphData2:color) lineattrs=(color=black
pattern=solid);
      Value "200" /fillattrs=(color= GraphData3:color) lineattrs=(color=black
pattern=solid);

EndDiscreteAttrMap;
DiscreteAttrVar attrvar=id_dose_group var=DOSE attrmap="Dose_Group";
```

The DISCRETEATTRMAP statement above was used to assign the attributes for the treatment doses (and the doses within each visit). For example patients that were dosed with 50mg of treatment are displayed with bar charts that are filled in blue with a solid black outline. The color = **GraphData1:color** creates the blue color and this was used because this color is used within the Default Style Template, and is the first color that is used within grouped data. There are twelve colors that the Default Style Template has specified to use with groped items, and these colors have been researched extensively by the graphics development team in SAS. These colors were chosen because they are easy to see and differentiate well between the groups, meaning that the groups are more easily distinguished when viewed on plots. Therefore, when I specify colors, I tend to use these colors first. These colors can be accessed by using **GraphData1:color**, **GraphData2:color**, **GraphData3:color**, etc. The Default Style Template can be viewed in the log after running this code:

```
proc template;
source styles.default;
run;
```

The section of the template that specifies the actual group colors are **gdata1**, **gdata2**, etc. This section of the template is located within the **GraphColors** class.

The DISCRETEATTRVAR statement was used to link the graphical properties of the DISCRETEATTRMAP that was created, namely **"Dose Group"** with the variable in the dataset named **Dose**. Therefore when a dose in the dataset matches a value in the DISCRETEATTRMAP, the specified attributes for the dose will be used on the graph.

```
DiscreteAttrMap name="Visit_Dose_Group";

      Value "1-50"  "2-50"  "3-50"  "4-50"     /fillattrs=(color= GraphData1:color)
lineattrs=(color=black pattern=solid);
      Value "1-100" "2-100" "3-100" "4-100" /fillattrs=(color= GraphData2:color)
lineattrs=(color=black pattern=solid);
      Value "1-200" "2-200" "3-200" "4-200" /fillattrs=(color= GraphData3:color)
lineattrs=(color=black pattern=solid);

EndDiscreteAttrMap;
DiscreteAttrVar attrvar=id_visit_dose_group var=VISIT_DOSE
attrmap="Visit_Dose_Group";
```

The above DISCRETEATTRMAP statement was used to define attributes of each treatment dose within each visit. This is because a patient could receive a lesser dose at a subsequent visit if it was deemed safer. A key part of the Napoleon Plot is to be able to identify colors that have had a reduced dose, and the DISCRETEATTRMAP created above allows this.

In essence each dose is denoted by specific attributes regardless of the visit and hence is the same color that was specified for the overall dose value, for example if a patient received 50mg at visit 1, 2, 3 or 4 then this patient would have a bar chart filled in blue with a solid black outline at visits 1, 2, 3 and 4. Another way this could have been accomplished is by using the GROUP=index option. However, using the DISCRETEATTRMAP statement instead of the GROUP=index option is more simpler when producing Napoleon Plots.

BARCHARTPARM AND DISCRETE LEGEND

```
layout overlay / xaxisopts=(label="Days on Treatment" labelattrs=(size = 12px)
linearopts = (viewmin = 0) tickvalueattrs =(size=10px))
                yaxisopts=(label="Subject" labelattrs=(size = 12px) reverse = true
tickvalueattrs =(size=10px) type=discrete);

   BarChartParm X=order_subject Y=cycledays /  barwidth=0.8 orient=horizontal
Group=id_visit_dose_group datalabel = cohort;


   DiscreteLegend "Dose_Group"/ type = fill autoalign = (bottomright) across =1
down = 4 location = inside title = "Dose" titleattrs=(size=12px)
valueattrs=(size=10px);


endlayout;
```

In Figure 2, to plot the bar chart and the dose legend the BARCHARTPARM and the DISCRETELEGEND statements were used within the LAYOUT OVERLAY. REVERSE = TRUE is used within the LAYOUT OVERLAY to ensure that the subjects are sorted in ascending order, when they are displayed on the y-axis. The reason for this is because ORIENT = HORIZONTAL is used in the BARCHARTPARM statement, and this essentially flips the x and y-axis and reverse the order of the categorical variable. X = ORDER_SUBJECT was used to order the subjects by the maximum days on treatment. GROUP = ID_VISIT_DOSE_GROUP is essentially specifying that the colors to be used on the graph come from the DISCRETEATTRMAP created above and named VISIT_DOSE_GROUP.

FIGURE 3 – ADDITIONAL CODE

In addition to Figure 2, Figure 3 also has the reason for treatment discontinuation displayed on the bar chart, and on the legend.

Table 2: Subset of Data used in Figure 3

	subjid	cycledays	dose	visit_dose	order_subject	result	length
1	1003	28	200	1-200	1	O	112
2	1003	28	200	2-200	1	O	112
3	1003	28	200	3-200	1	O	112
4	1003	28	200	4-200	1	O	112
5	1004	28	200	1-200	2	A	112
6	1004	28	100	2-100	2	A	112
7	1004	28	50	3-50	2	A	112
8	1004	28	50	4-50	2	A	112

The variables **result** and **length** were used in Figure 3 to display the reason for treatment discontinuation and the legend. Length is the cumulative sum of the cycle days for each subject.

```
DiscreteAttrMap name="Response_Markers";

      Value "A" "Adverse Event" /markerattrs=(color=cxA9865B symbol=squarefilled);
      Value "D" "Death" /markerattrs=(color=cxB689CD symbol=squarefilled);
      Value "O" "Ongoing" /markerattrs=(color=cxBABC5C symbol=squarefilled);
      Value "PD" "Progressive Disease"/markerattrs=(color=cx94BDE1
symbol=squarefilled);

EndDiscreteAttrMap;
DiscreteAttrVar attrvar=id_response_markers var=result attrmap="Response_Markers";
```

The above DISCRETEATTRMAP statement was used to specify the colors and the shape of the reason for treatment discontinuation, if applicable. If applicable is included in the sentence because the value "Ongoing" or "O" means that the subject has not discontinued treatment.

In the VALUE statement, SYMBOL = SQUAREFILLED denotes that the shape of the symbol will be square.

```
Scatterplot X = length Y = order_subject / markerattrs = (size = 10px) group
=id_response_markers;

DiscreteLegend "Response_Markers"/ type = marker exclude = ("A" "D" "O" "PD")
location = outside valign = bottom down = 2 across = 4 title="Reason for Treatment
Discontinuation" titleattrs=(size=10px) valueattrs=(size=8px);
```

The above SCATTERPLOT statement was used to display the square symbols at the end of the bar charts. The DISCRETELEGEND statement was used to display the legend on the plot.

FIGURE 4 – ADDITIONAL CODE

In addition to the previous figure, Figure 4 also displays the initials, of the reason for treatment discontinuation.

Table 3: Subset of data used in Figure 4

	subjid	cycledays	dose	visit_dose	order_subject	result	length	result_label
1	1003	28	200	1-200	1	O	112	O
2	1003	28	200	2-200	1	O	112	
3	1003	28	200	3-200	1	O	112	
4	1003	28	200	4-200	1	O	112	
5	1004	28	200	1-200	2	A	112	A
6	1004	28	100	2-100	2	A	112	
7	1004	28	50	3-50	2	A	112	
8	1004	28	50	4-50	2	A	112	

The **result_label** variable in Table 3 above in conjunction with the SCATTERPLOT statement below (used with the previous LAYOUT OVERLAY) display the initials. If you compare the variables result and result_label, the reason you will notice that the only difference is result_label has some missing values, i.e. only the first row per subject has a value present for result_label. The reason for this is to display the initials more clearly on the plot. If the variable result was used instead of result_label in the scatterplot then the initials would appear quite blurry.

```
Scatterplot X = length Y = order_subject / markercharacter = result_label
markercharacterattrs=(size=10px);
```

FIGURE 5 – ADDITIONAL CODE

In addition to the previous figure, Figure 5 – the Napoleon Plot, also displays the cohort information.

Table 4: Subset of data used in Figure 5

	subjid	cycledays	dose	visit_dose	order_subject	result	length	result_label	cohort_label_x	cohort_label
1	1003	28	200	1-200	1	O	112	O	127	Cohort 1
2	1003	28	200	2-200	1	O	112		127	
3	1003	28	200	3-200	1	O	112		127	
4	1003	28	200	4-200	1	O	112		127	
5	1004	28	200	1-200	2	A	112	A	127	Cohort 1
6	1004	28	100	2-100	2	A	112		127	
7	1004	28	50	3-50	2	A	112		127	
8	1004	28	50	4-50	2	A	112		127	

The variables **cohort_label_x** and **cohort_label** were used with the SCATTERPLOT statement below to produce the cohort information. The variable cohort_label_x comprises of the variable length + 15, and when used with the option

```
Scatterplot X = cohort_label_x Y = order_subject / markercharacter = cohort_label
markercharacterattrs=(size=9px);
```

MARKERCHARACTER = COHORT_LABEL displays the cohort that each subject is in.

CONCLUSION

The Napoleon Plot is a very useful plot for assessing treatment safety at each interim. For each subject the extent of exposure, the reason for treatment discontinuation and the assigned cohort is displayed. Another way to use the Napoleon Plot is to exclusively use it to assess the Maximum Tolerated Dose (MTD). That is, display the adverse events on the plot that contribute to determining if the MTD has been met.

The drawback of the method used to display subjects that had been dosed reduced on the Napoleon Plot is that the treatment cycles have to be entered manually in the VALUE statement within the DISCRETEATTRMAP statement. If you recall in our example we had:

```
Value "1-50" "2-50" "3-50" "4-50" …;
```

Therefore if there were too many treatment cycles in the study then each cycle would have to be entered manually in the VALUE statement.

This paper shows how to produce the Napoleon Plot in SAS 9.4. You can also use it to produce a Napoleon Plot in SAS 9.3 with minor modifications, such as changing the DPI to IMAGE_DPI in ODS LISTING. Currently I am writing a program that can produce the Napoleon Plot in SAS 9.2 as I understand there are users still on this version that can benefit from producing a Napoleon Plot in SAS 9.2.

REFERENCES

Kuhfeld, Warren. February 2010. *Statistical Graphics in SAS: An Introduction to the Graph Template Language and the Statistical Graphics Procedures*. Cary, North Carolina. SAS Institute.

SAS(R) 9.4 ODS Graphics: Procedures Guide. (2013, November). Available at http://support.sas.com/documentation/cdl/en/grstatproc/64978/HTML/default/viewer.htm#n0yjdd910dh59zn1toodgupaj4v9.htm

ACKNOWLEDGMENTS

I would like to thank Devi Sekar of GlaxoSmithKline for reviewing this paper and for her valuable suggestions. Additionally, I would like to thank Sharon Carroll for the support she gave me whilst preparing the paper, and I would

like to thank Anna McGlothlin for creating the Napoleon Plot in R, and therefore enabling me with something to work towards.

RECOMMENDED READING

- Matange, Sanjay. November 2013. *Getting Started with the Graph Template Language in SAS: Examples, Tips, and Techniques for Creating Custom Graphs*. Cary, North Carolina. SAS Institute.

CONTACT INFORMATION

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

> Name: Kriss Harris
> Enterprise: SAS Specialists Ltd.
> E-mail: italjet125@yahoo.com
> Web: http://www.krissharris.co.uk
> Twitter: https://twitter.com/krissharris

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.

APPENDIX

```
/* Napoleon plot */

data nap;
input subjid 1-4 cycledays 6-7 cumcycledays 9-11  visit 13 length 15-17 dose 19-21
result $23-24 cohort $26-33;
cards;
1001 28 28   1 56   100 D   Cohort 1
1001 28 56   2 56   50  D   Cohort 1
1002 28 28   1 84   200 A   Cohort 1
1002 28 56   2 84   200 A   Cohort 1
1002 28 84   3 84   100 A   Cohort 1
1003 28 28   1 112 200 O   Cohort 1
1003 28 56   2 112 200 O   Cohort 1
1003 28 84   3 112 200 O   Cohort 1
1003 28 112 4 112 200 O   Cohort 1
1004 28 28   1 112 200 A   Cohort 1
1004 28 56   2 112 100 A   Cohort 1
1004 28 84   3 112 50  A   Cohort 1
1004 28 112 4 112 50  A   Cohort 1
2001 28 28   1 84   100 PD Cohort 2
2001 28 56   2 84   50  PD Cohort 2
2001 28 56   3 84   50  PD Cohort 2
2002 28 28   1 84   200 A   Cohort 2
2002 28 56   2 84   200 A   Cohort 2
2002 28 84   3 84   100 A   Cohort 2
2003 28 28   1 112 50  O   Cohort 2
2003 28 56   2 112 50  O   Cohort 2
2003 28 84   3 112 50  O   Cohort 2
2003 28 112 4 112 50  O   Cohort 2
2004 28 28   1 84   200 O   Cohort 2
2004 28 56   2 84   100 O   Cohort 2
2004 28 84   3 84   50  O   Cohort 2
;
run;

data nap2;
set nap;
by subjid visit;
```

```
retain classvisit;
if first.subjid then do;
      classvisit = visit;
      cohort_label = cohort;
      result_label = result;
end;
visit_dose = catx("-",visit, dose);
label_x = length + 2;
cohort_label_x = length + 15;
run;

proc sort data= nap2;
by descending length subjid;
run;

data nap3;
set nap2;
by descending length subjid;
retain order_subject 0;
if first.subjid then order_subject = order_subject + 1;
run;
proc sql;
create table format_dataset as
select distinct order_subject as start, subjid as label, "subjfmt" as fmtname
from nap3;
quit;

proc format cntlin=format_dataset;
run;
proc template;
source styles.default;
run;

*** Note when referencing the DiscreteattrMap directly in the discrete legend you need
to add in the type ***;


proc template;
define statgraph napplot;
begingraph;

DiscreteAttrMap name="Dose_Group";

      Value "50" /fillattrs=(color= GraphData1:color) lineattrs=(color=black
pattern=solid);
      Value "100" /fillattrs=(color= GraphData2:color) lineattrs=(color=black
pattern=solid);
      Value "200" /fillattrs=(color= GraphData3:color) lineattrs=(color=black
pattern=solid);

EndDiscreteAttrMap;
DiscreteAttrVar attrvar=id_dose_group var=DOSE attrmap="Dose_Group";


DiscreteAttrMap name="Visit_Dose_Group";

      Value "1-50" "2-50" "3-50" "4-50"      /fillattrs=(color= GraphData1:color)
lineattrs=(color=black pattern=solid);
      Value "1-100" "2-100" "3-100" "4-100" /fillattrs=(color= GraphData2:color)
lineattrs=(color=black pattern=solid);
      Value "1-200" "2-200" "3-200" "4-200" /fillattrs=(color= GraphData3:color)
lineattrs=(color=black pattern=solid);
```

```
EndDiscreteAttrMap;
DiscreteAttrVar attrvar=id_visit_dose_group var=VISIT_DOSE attrmap="Visit_Dose_Group";


DiscreteAttrMap name="Response_Markers";

       Value "A" "Adverse Event" /markerattrs=(color= GraphData4:color
symbol=squarefilled);
       Value "D" "Death" /markerattrs=(color= GraphData5:color symbol=squarefilled);
       Value "O" "Ongoing" /markerattrs=(color= GraphData6:color symbol=squarefilled);
       Value "PD" "Progressive Disease"/markerattrs=(color= GraphData7:color
symbol=squarefilled);

EndDiscreteAttrMap;
DiscreteAttrVar attrvar=id_response_markers var=result attrmap="Response_Markers";


EntryTitle "Napoleon Plot" / textattrs = (size=14px);
layout overlay / xaxisopts=(label="Days on Treatment" labelattrs=(size = 12px)
linearopts = (viewmin = 0) tickvalueattrs =(size=10px))
                 yaxisopts=(label="Subject" labelattrs=(size = 12px) reverse = true
tickvalueattrs =(size=10px) type=discrete);

   BarChartParm X=order_subject Y=cycledays /  barwidth=0.8 orient=horizontal
Group=id_visit_dose_group datalabel = cohort;


   Scatterplot X = length Y = order_subject / markerattrs = (size = 10px) group
=id_response_markers;
   Scatterplot X = length Y = order_subject / markercharacter = result_label
markercharacterattrs=(size=10px);
   Scatterplot X = cohort_label_x Y = order_subject / markercharacter = cohort_label
markercharacterattrs=(size=9px);


   DiscreteLegend "Dose_Group"/ type = fill autoalign = (bottomright) across =1 down =
4 location = inside title = "Dose" titleattrs=(size=12px) valueattrs=(size=10px);
   DiscreteLegend "Response_Markers"/ type = marker exclude = ("A" "D" "O" "PD")
location = outside valign = bottom down = 2 across = 4 title="Reason for Treatment
Discontinuation" titleattrs=(size=10px) valueattrs=(size=8px);

* The above statement produces the legend for the Legend "Response_Markers". Exclude
is used because "Response_Markers" has 8 items. *;

endlayout;


endgraph;
end;
run;

** Maxlegend area is very important here, as we are displaying 2 legends **;

ods graphics on / reset = all maxlegendarea = 30 imagename ="nappharmaSUG" width = 3in
height  = 2in ;
ods listing gpath = "C:\Users\Kriss\PharmaSUG 2014" dpi = 300;
goptions reset=all border;

proc sgrender data = nap3 template = napplot;
format order_subject subjfmt.;
run;

ods listing close;
```