PharmaSUG 2020 - Paper DV-050

Oncology Graphs-Creation (Using SAS and R), Interpretation and QA

Taniya Muliyil, Bristol Myers Squibb Corporation

ABSTRACT

Data visualization plays a key role in analyzing and interpreting data. In oncology studies, graphs help visualize, interpret and analyze trends in data from a statistical perspective. Graphical outputs help in exploring data, identifying issues with data and in turn help improve data quality. Most commonly used statistical software for creating oncology graphs in pharmaceutical industry is SAS and R. Programmers create complex graphical outputs, but many are unable to interpret the results that these graphs display. This paper focuses on creating some common oncology graphs like spider plot, swimmer plot and waterfall plot using R and SAS, along with interpreting the results displayed by these graphs. It also discusses common QA findings that will reduce issues while generating these outputs and in turn help with statistical interpretation and analysis.

INTRODUCTION

Data visualization is very important for analysis, presenting data trends and exploring quality of data. SAS and R are two main software tools used by pharmaceutical companies to generate graphical outputs. This paper focuses on the R and SAS codes used to generate waterfall, swimmer and spider plots along with interpretation of results. It also discusses some of the data driven and programmatic checks that one needs to keep in mind while generating these outputs.

Most programmers use SAS for generating graphical outputs however; other software tools as R could also generate these outputs. This paper includes R code along with SAS for generating these graphical outputs.

GENERATING WATERFALL PLOT USING SAS

```
   proc template;
        define style tumor;
   parent=Styles.Default;

        class GraphFonts / 'GraphLabelFont'=("Arial", 10pt, normal)
                           'GraphValueFont'=("Arial", 10pt, normal)
                           'GraphTitleFont'=("Arial", 12pt, normal);
       style GraphBackground / color=WHITE;

       end;
    run;

proc template;
        define statgraph tumor;
        begingraph / border=false;
            layout overlay/
              XAXISOPTS=(display= (label) labelattrs = (family="arial"))
              YAXISOPTS=(labelattrs = (family="arial")
                tickValueAttrs =    (family="arial")
                      linearopts= (viewmin=-100 viewmax=&max.
tickvaluesequence=( start=-100 end=100 increment=25 )));

 barchart x=usubjid1 y=aval/ outlineattrs=(color=black) group=avalc
name='bars' index=groupn;

   scatterplot x=usubjid1 y=resp/ markerattrs=(color=black symbol=asterisk
size=7);
```

```
  dropline x=usubjid1 y=-30  /dropto=y discreteoffset=+0.5
 lineattrs=( color=black pattern=shortdash);

 dropline x=usubjid1 y=0  /dropto=y discreteoffset=+0.5
lineattrs=( color=black pattern=shortdash);
endlayout;

endgraph;
end;

RUN;

 PROC SGRENDER DATA=FORPLOT2 TEMPLATE=TUMOR;

    RUN;


ods graphics off;

ods listing close;
ods listing;


ODS RTF CLOSE;
```

GENERATING WATERFALL PLOT USING R

```
#Load in packages for use in the current session.

Library ("haven")

b<-ggplot(finaldata,aes(x=reorder(USUBJID,-AVAL) ,y=AVAL, color=TRTP))+
geom_bar(stat="identity", position='dodge', width=1)+
scale_fill_manual(values=c("black", "brown"))+

 Labs (list (title = "Best Reduction in Target Lesion per Investigator (%) -
All Patients\n ", x = NULL, y = "Best Reduction in Target Lesion per Investigator
(%)")) +


coord_cartesian(ylim=c(-100,100))+theme(axis.title.x=element_blank(),
axis.text.x=element_blank(),axis.ticks.x=element_blank(),

panel.border = element_blank (),panel.background = element_blank())+
geom_point(aes(x=USUBJID, y=RESP,color='black',shape=3))+


scale_color_manual(values=c("black","brown",
"blue","red"))+theme(legend.position="none")+scale_shape_identity()

b
```

WATERFALL PLOT INTERPRETATION AND QA

Waterfall plot outlines the best change i.e. maximum reduction or minimum increase from baseline over a period. Reduction in tumor size across all time points can be effectively analyzed using waterfall plot. Figure1 represents best reduction from baseline in target lesions for all subjects randomized to study treatment. Vertical bars measure change from baseline in the sum of target lesion measurement. Vertical bars in between 0-100% indicate no reduction/increase in tumor size whereas subjects below zero mark

indicate tumor reduction. Vertical bars are grouped by best overall response (NE=Not Applicable, PD=Progressive Disease, SD=Stable Disease, PR=Partial Response, CRU=Complete response Unconfirmed). Best tumor reduction can be calculated from time of randomization for randomized trials or time of first treatment to progression/death .The Y-axis represent a trend from lowest reduction to the highest reduction .Figure1 shows a reference line for 30% reduction in tumor size. Asterisks on the bottom indicate confirmed responders based on the definition of overall response as stated in the protocol and statistical analysis plan. For large sample size, it is difficult to distinguish individual bars or subject information .However depiction of the trend or pattern of tumor reduction across different subjects is still achievable.

WATERFALL PLOT - BEST REDUCTION FROM BASELINE OF TARGET LESIONS GROUPED BY BEST OVERALL RESPONSE

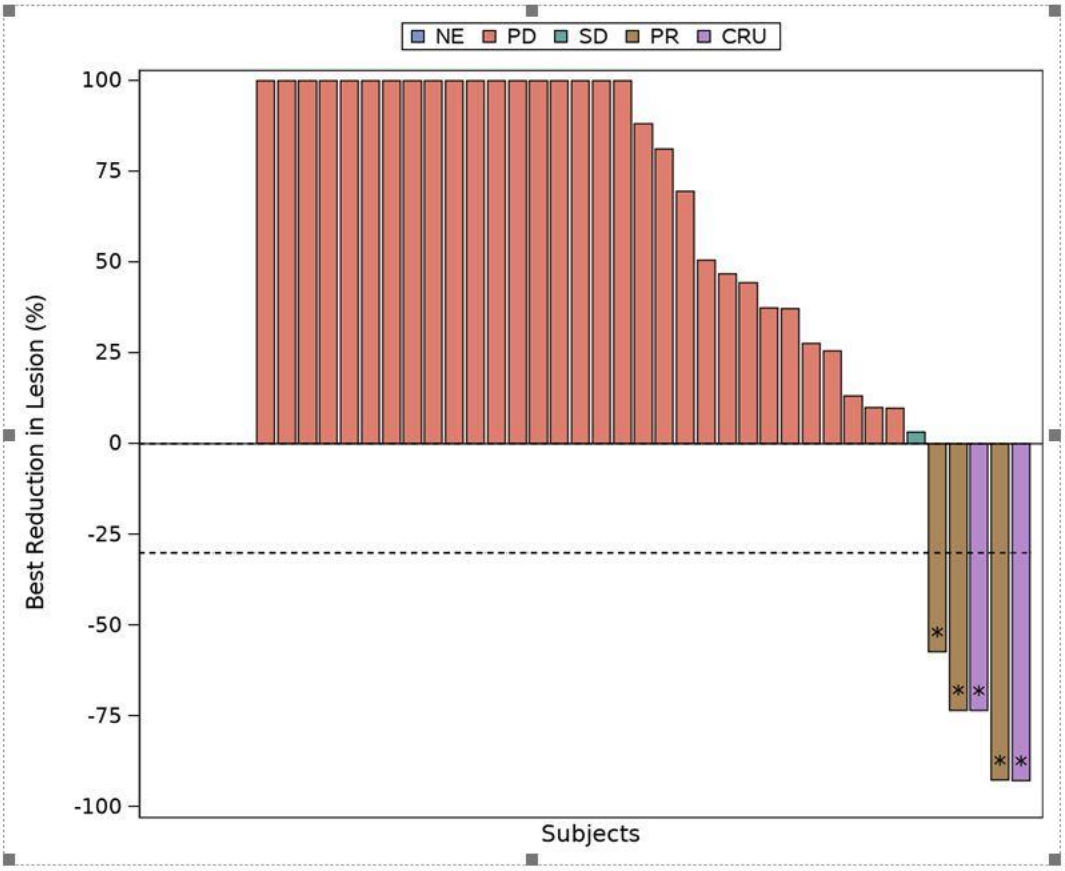

Figure 1

Good programming practice while generating waterfall plot is to use the below checks to confirm the data.

- Compare reduction in tumor size to subject's overall response. Thirty percent reduction in solid tumor based on RECIST criteria v1.1 is considered. Subject is not considered a responder if maximum reduction is 10%, for example. Also, as the waterfall plot only summarizes best changes from baseline for target lesions so it only provides information on target lesion response, even if subjects had reduction beyond certain threshold for the target lesions, they may not be responders taking into account the non-target lesions and presence of new lesions.
- Subjects with missing baseline information mentioned in footnote
- The number of subjects illustrated on the waterfall plot should equal to number of subjects

with atleast 1 valid baseline and post baseline measurement.

Above-mentioned checks will help to ensure data and programming quality.

GENERATING SWIMMER PLOT USING SAS

```
proc template;
    define style bars1;

        style GraphData1 / color = GREY96 fillpattern = "Lx1";
        style GraphData2 / color = GREYFA fillpattern = "Rxx1";
        style GraphData3 / color = GREYBE fillpattern = "Xx4";
        style GraphData4 / color = GREYE6 fillpattern = "Lxx4";
        style GraphData5 / color = GREYAA fillpattern = "Rxx4";
        style GraphData6 / color = GREY96 fillpattern = "X5";
        style GraphData7 / color = GREYFA fillpattern = "L5";
    end;

define statgraph swimplot ;
    begingraph / designwidth=7in designheight=6.7 in border=off;
    dynamic bar_width;
    layout lattice / columns=1 rowgutter=1 ;
    cell;

    layout overlay/ yAXISOPTS=(display= (label ticks tickvalues)
label="&YAxisLbl." labelattrs = (family="albany amt")

  tickValueAttrs=(family="albany amt" size=10pt ))
   xAXISOPTS=(label="Study Day" labelattrs = (family="albany amt")
tickValueAttrs = (family="albany amt" size=10pt)

  linearopts= (viewmin=0 viewmax=3000 tickvaluesequence=( start=0  end=3000
increment=100)));

   barchart y=pfs_day  x=usubjid1 / name = "bars1" group = avalc index = aval
                                    orient=horizontal barwidth=bar_width
                                    display=(fillpattern fill outline);


   scatterplot x=cnsr_day     y=usubjid1/ markerattrs=(color=black
symbol=circlefilled   size=12) name="a1" legendlabel="PFS Censored" ;

   scatterplot x=cnsr os day  y=usubjid1/ markerattrs=(color=black
symbol=triangle       size=12) name="a2" legendlabel="OS Censored" ;
 scatterplot x=os_day   y=usubjid1/ markerattrs=(color=black
symbol=trianglefilled size=12) name="a3" legendlabel="Death" ;

 discretelegend  "a1" "a2" "a3" "a4" "a5" / across=2 valign=bottom ValueAttrs
= (family="albany amt" size=10pt) ;

 discretelegend  "bars1" / valign=top across=6 ValueAttrs = (family="albany
amt" size=10pt);

    endlayout;
    endcell;
```

```
    endlayout;
    endgraph;
    end;

run;

goptions reset = global
         device = png300 gsfname = outputs goutmode = replace
         fileonly fileclose = graph
         gunit = cells noborder cback = white
         colors = (black)
         hsize = 7.0in vsize = 7.4in
         htitle = 10pt htext = 10pt
         ftitle = "albany amt" ftext = "albany amt"
         display;


ods graphics on /labelmax=300;
ods _all_ close;

proc sgrender data=figures.&tabname  template=swimplot ;
run;
```

GENERATING SWIMMER PLOT USING R

```
Data Reading and Manipulation Packages
Library ("haven")

X1<- ggplot (swimmer, aes (USUBJID1, PFS_DAY)) +
geom_bar (stat="identity", aes (fill=factor (AVALC)), width=0.3) +
geom_point (data=swimmer, aes (USUBJID1, CNSR_DAY), shape=18, size=5) +
geom_point (data=swimmer, aes( USUBJID1,CNSR_OS_DAY),shape=17,size=5)+
geom_point(data=swimmer, aes (USUBJID1, OS_DAY), shape=6, size=5) +
geom_point(data=swimmer, aes (USUBJID1, ADA_SAMPLE1), shape=18, size=5) +


scale_color_manual (values = c("green","yellow","red")) +
scale_shape_identity ()+


coord_flip () +scale_y_continuous (limits = c (0, 3000), breaks = seq (0,
3000, by = 100)) + theme (legend. Position = "top", legend.
Title=element_blank ()) + labs(x="X positive subjects", y="Study Day")

X1
```

SWIMMER PLOT INTERPRETATION AND QA

Swimmer plot can graphically provide key insight to subject's response to treatment over the duration of study. It can provide critical information about the duration of response, best overall response, progression free survival, progression date, death date and many other response related information.

The swimmer plot in Figure2 provides information about the best overall response (Complete response or CR, Partial Response or PR, Stable Disease or SD) along with the below information:

1. PFS (Progression Free Survival) censored time point- PFS censored time point can be defined as last assessment date for subjects who did not experience progression or death.

2. OS (Overall Survival) censored time point-OS censored time point can be defined as the last date for which the subject was known to be alive without documentation of death.

3. Death day for six subjects (Pt1-Pt6).

Horizontal bar shown below indicates progression free survival period and grouped based on their best overall response as shown in the legend above the graph.

SWIMMER PLOT OF BIOMARKER OCCURRENCE IN RELATIONSHIP TO PFS, OS and OR

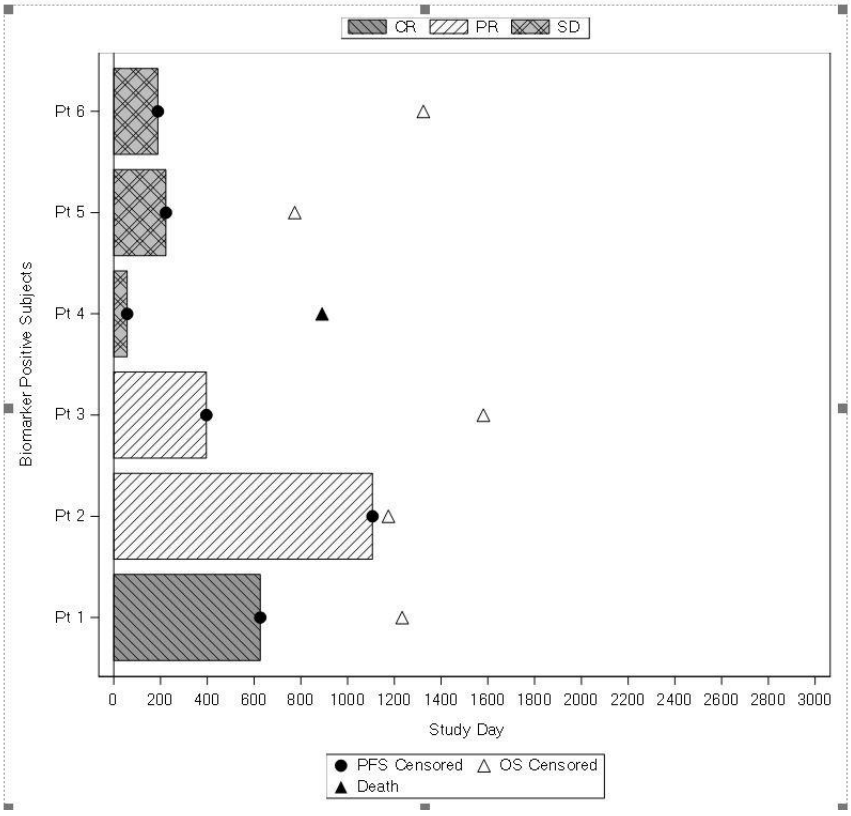

Figure 2

Good programming practice while generating swimmer plot is to use the following checks while generating the graph:

- Swimmer plot can best represent individual subject results if sample size is small. If the population size is very big then it is advisable to subset the population based on different conditions like displaying data just for the responders.
- If many analysis time points would be, represented then appropriate symbols ,the endpoints can be easily distinguished using appropriate symbols
- While plotting duration of response in swimmer plots, it is important to represent start and end of

each response clearly.

GENERATING SPIDER PLOT USING SAS

```
proc template;
            define statgraph tumor_&i.;
            begingraph / border=false;
             layout lattice/ border=false rows=3
             rowweights=(.85 .0 .15);
                layout overlay/
                XAXISOPTS=(labelattrs = (family="arial") tickvalueattrs=(
                    %if &&_max_&i.>80 %then size = 8pt;
                    %else size = 10pt;)

      linearopts=(viewmin=0 viewmax=&&_max_&i. tickvalueformat=ticks_fmt.
                    tickvaluesequence=( start=0 end=24 increment=6 )
                    TICKVALUEFITPOLICY=NONE))


     YAXISOPTS=(labelattrs = (size = 7pt family="arial") tickValueAttrs =
(family="arial")

  linearopts= (viewmin=-100 viewmax=100 tickvaluesequence=( start=-100
end=100 increment=25 ))
  label="Percent Change from Baseline in Target Lesions Tumor Burden (%)") ;

    seriesplot x=x_line y=y_line/ group=usubjid1 index=grpx_groupn
                    %if &legend. ne 1 %then %do;
                        lineattrs=(color=black pattern=solid)
                    %end;
                    ;
    scatterplot x=x_line y=y_new / name = "NEWL"    legendlabel = ": 1st
occurrence of new lesion" markerattrs=(symbol=plus size=9);

referenceline y= -50 / lineattrs=(color=black pattern=shortdash);

scatterplot x=x_line y=y_off /name = "OFFTRT"  legendlabel = ": OFFTRT"
markerattrs=(symbol=circlefilled size=6) datalabel=usubjid1;

 dropline x=1000 y=-50  /dropto=y lineattrs=(color=black pattern=shortdash);
 dropline x=1000 y=0  /dropto=y lineattrs=(color=black pattern=shortdash);
endlayout;


%do;
 layout lattice/ rows=2 rowweights=(0.5 0.5);
 layout overlay;
   discretelegend "NEWL" "OFFTRT" "100%" /
border=false halign=center valign=bottom valueattrs=(size=10pt color=black)
DISPLAYCLIPPED = TRUE
order=columnmajor down=2;
  endlayout;
   layout overlay;
    discretelegend "LINE"/border=false halign=center valign=bottom
valueattrs=(size=10pt color=black) DISPLAYCLIPPED = TRUE
   order=columnmajor down=2;
```

```
   endlayout;
   endlayout;
   %end;

  ods graphics on /  reset=all height=6.5in width=8.0in imagefmt=png;
    ods graphics / antialiasmax=100000;

    ods listing close;

proc sgrender data=figures.x template=tumor_&i.;
run;
```

GENERATING SPIDER PLOT USING R

```
#Load in packages for use in the current session.

Library ("haven")


p<- ggplot(adfetm,aes(x=X_LINE,y=Y_LINE,group=USUBJID1))+coord_cartesian

(ylim = c(-100,-75,-50,-25,0,25,50,75,100))

p<-p+ geom_line (aes (color=USUBJID1)) + scale_shape_identity () +geom_point
(aes(x=X_LINE, y=Y_NEW, color='blue', shape=3)) +

geom_point (aes(x=X_LINE, y=Y_OFF, color='blue', shape=19)) +

theme (legend.position = "none") +geom_text (data = adfetm, aes (X_LINE,
Y_OFF, label = USUBJID1), nudge_y=4, nudge_x=1, check_overlap=TRUE, size=3)+


geom_hline (yintercept=-50, linetype="dashed", color = "blue") +

ggtitle ("Plot of Tumor Burden change per Investigator\n All response
evaluable subjects") + ylab ("Percent Change from Baseline in Target Lesions
Tumor Burden (%)")+xlab("Time since First Treatment Date (Weeks)")

p1 <- p +scale_y_continuous(breaks=c(-100,-75,-50,0,50,75,100)) +
scale_x_continuous(breaks=c(0,6,12,18,24,30,36))

p1
```

SPIDER PLOT INTERPRETATION AND QA

Spider plot portraits changes in tumor burden over time for all subjects relative to their baseline measurement. The major difference between waterfall plot and spider plot is that waterfall plot shows overall increase or decrease in tumor burden based on baseline measurement whereas spider plot shows changes in tumor burden across each time point. Spider plot is especially useful if we need to see subject data across time points.

Figure3 below shows the percent change in tumor burden during the first 24 weeks of treatment. The graph below shows trend of increase/decrease of tumor burden from start of treatment to 24 weeks post treatment. First occurrence of new lesion and the off treatment period for each subject has been plotted in the graph shown below.

SPIDER PLOT OF TUMOR BURDEN CHANGE

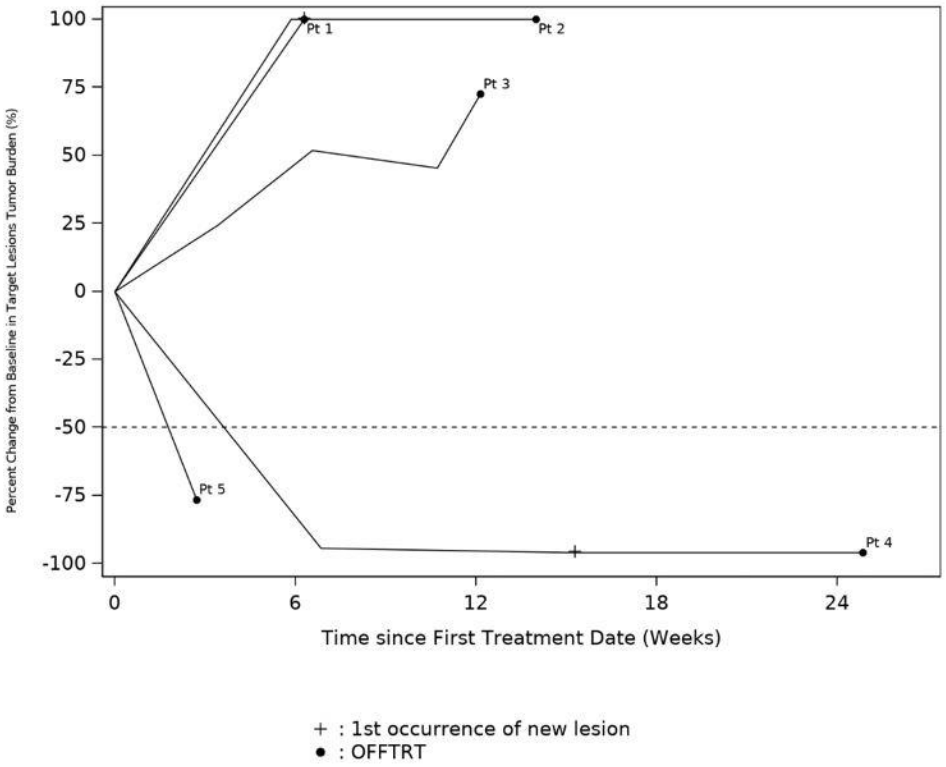

Figure 3

Good programming practice while generating spider plot is to use the below checks while generating the graph.

- Spider plot is very useful if the sample size is small. For large populations it is advisable to subset based on analysis requirements. For example, population can be subset based on responders.
- The number of subjects shown on the spider plot should equal to number of subjects with atleast 1 valid baseline and post baseline measurement.

CONCLUSION

Graphs provide key insight into data and serve as an important tool for data analysis. It is important to know different software tools to create graphical outputs and not rely on just one .Adhering to good programming practice can help minimize programming errors, offer help in checking quality of data, which in turn help generate high quality outputs for data interpretation.

REFERENCES

Matange, Sanjay (2012) ”Clinical Graphs Using SG Procedures Course Notes” Proceedings of the Pharmasug 2012 Conference. Cary, NC: SAS Institute.

Dmitrienko, Alex, Christy Chuang-Stein, and Ralph D’Agostino. 2007. Pharmaceutical Statistics Using SAS A Practical Guide, Cary, NC: SAS Institute Inc.

CONTACT INFORMATION

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

Taniya Muliyil
Bristol Myers Squibb
taniya.muliyil@bms.com