Title: Generating Forest and Kaplan Meier graphs for Regulatory Submission: Comparison of SAS and R

Abstract:

Statistical graphs are the essential component of the clinical trials which are included in the submission package to the regulatory agencies. As we all know R language is being adopted in the pharmaceutical companies along with the traditional SAS which is still in use. In this paper we would like to discuss some of the prominent clinical graphs such as Forest plot and Kaplan-Meier plot using both SAS and R. A side-by-side comparison of both the graphs in SAS and R using their respective syntax, key differences and features that are available in both languages are highlighted in this paper.

Introduction:

When we have the knowledge and softwares available for both SAS and R, we have the luxury to choose them based on the graphs that we need to plot. With the upgraded versions of Graphic Template Language (GTL), SAS is getting user friendly to accomplish the desired high quality clinical graphs while R being an open source and is easy to adapt. ggplot2 package from R is a robust library which has the features to plot most of the clinical graphs. Additionally, there are many dedicated R packages to create specific graphs such as ‘survival’ for Kaplan-Meier, ‘forestplot’ package for Forest plots and there is scope to create more advanced customized plots in the future.

We are using GTL for SAS which might be embedded in macros and for R, we are covering all the graphs using ggplot2 package. We will distinguish outputs using these three factors: Time, Quality and Resources.

Forest plot:

For data, we are going to use the dummy data and some of the rows are displayed below.

```
infile datalines dlm=',' ;
input subgroup & $ nleft nright diff & $ diffrem & $;
cards;
Baseline Headache Status: Presence  ,       ,   ,  ,
Drug B vs Placebo                   ,  8    , 12, 27.0% (-70.1%, 30.2%)
Drug C vs Placebo                   ,  9    , 12, -15.9% (-60.7%, 32.0%)
Drug A + Drug B vs Placebo          , 18    , 12, -31.9% (-66.7%, 10.0%)
Drug A + Drug C vs Placebo          , 17    , 12, -20.2% (-58.5%, 1.1%)....
```

SAS Syntax: Parts of GTL template code is displayed below.

Step1: Following code will make the required rows bold.

```
discreteattrmap name="id" / ignorecase = true;
 value '1' / textattrs = (color=black size=10 weight=bold);
 value '2' / textattrs = (color=black size=7 weight=normal);
enddiscreteattrmap ;

discreteattrvar attrvar = sid var = id attrmap = "id";
```

Step 2: Following code is the important part of template code.

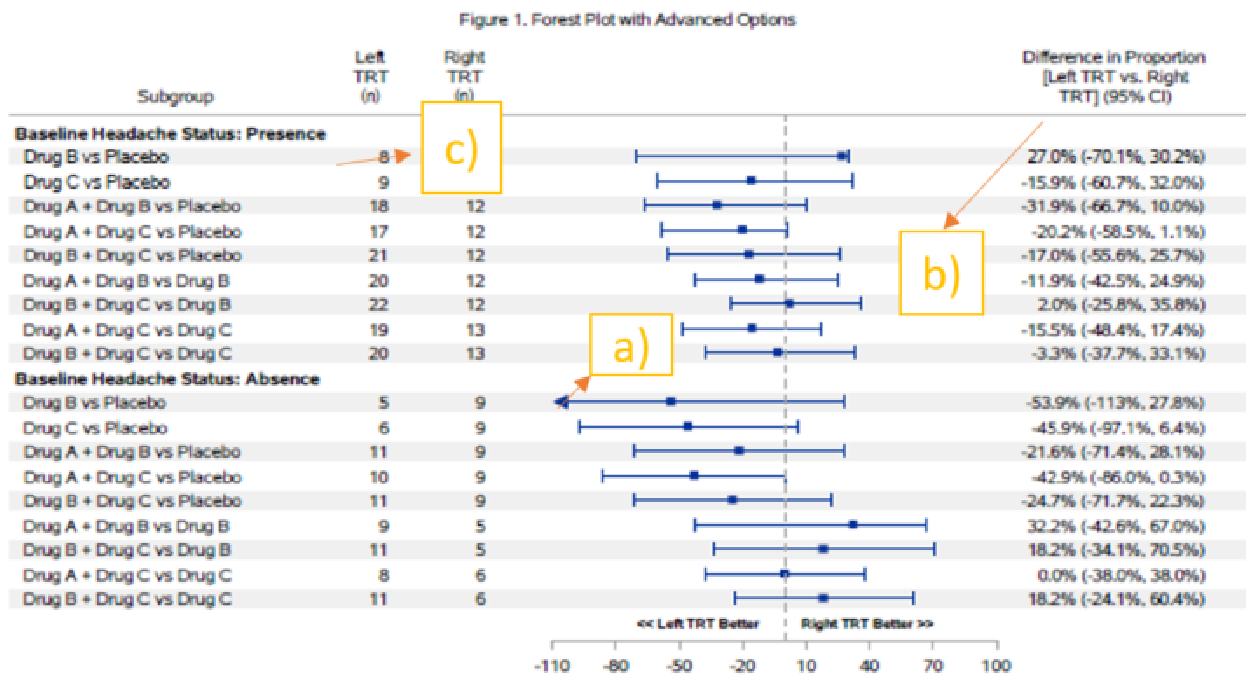
```
        highlowplot y=Subgroup  low= &vlcl high=&vucl /highcap=highcap lowcap=lowcap lineattrs=(pattern=1
thickness=0.5);
    scatterplot y=Subgroup x=&mv /  markerattrs=(symbol=&msymbol) ;
    scatterplot y=Subgroup x=&mv /  markerattrs=(size=0) xaxis=x2;
    referenceline x=0 / lineattrs = (pattern=&reflinep);
    textplot x=xl y=Subgroup text = text / position=center contributeoffsets=none textattrs=(weight=bold);
```

NOTE: innermargin option is used to place the text columns on the sides of the graph (which is not displayed here).

R Syntax: Load the required libraries including ggplot2, grid etc. before we run the following code.

```
plot <- ggplot(fp, aes(x = ldiff, y = obs)) +
     geom_hline(aes(yintercept = gray_id),colour = "gray95",size = 7,show.legend=F) +
     geom_errorbarh(aes(xmax = ucl, xmin = lcl, height = .2),color = "blue") +
     geom_errorbarh(data = subset(fp, fp$lcl <= -110), aes(xmax = ucl, xmin = -110, height = .2),color = "blue") +
     geom_point(shape = 22, fill = "blue") + geom_hline(yintercept = 0.6, linetype = 1) +
     geom_point(data = subset(fp, fp$lcl <= -110), aes(x = -109, y = obs), shape = 18, size=3, color = "blue") +
     geom_segment(x= 1,y = -1,xend = 1,yend = -50, linetype = 2) +
     geom_text(aes(y = 21,x = -40,label = "<< Left TRT Better", fontface = "plain"),size=3) +
     geom_text(aes(y = 21,x = 40,label = "Right TRT Better >>", fontface = "plain"),size=3) +
     xlab(" ") +  ylab(" ") + theme_classic() + scale_y_continuous(trans = "reverse") +
     scale_x_continuous(limits = c(-110, 100), breaks = seq(-110,100,30)) +
     theme(axis.text.y = element_blank(), axis.title.y = element_blank(), axis.ticks.y = element_blank(),
      axis.line.y = element_blank()) + theme(plot.margin = margin(2.5, -1, -7.5, -9))

data_left <- ggplot(fp, aes(y = obs)) +
            geom_hline(aes(yintercept = gray_id),colour="gray95",size=7,show.legend=F) +
            geom_text(data=subset(fp, obs %in% c(1,11) ),aes(x = 0, label = subgroup, fontface="bold"), hjust = 0) +
            geom_text(data=subset(fp,!(obs %in% c(1,11)) ),aes(x = 0, label = subgroup), hjust = 0) +
            geom_text(aes(x = 30, label = nleft)) + geom_text(aes(x = 45, label = nright), hjust = 1) +
            geom_hline(yintercept = 0.6, linetype = 1) + scale_colour_identity() +
            scale_y_continuous(trans="reverse") + theme_void() + theme(plot.margin = margin(5, -1, 35, 1))

data_right <- ggplot(fp, aes(y = obs)) +
             geom_hline(aes(yintercept  = gray_id),colour="gray95",size=7,show.legend=F) +
             geom_text(aes(x = 0, label = diff), hjust=0.5 ) +
             scale_colour_identity() + scale_y_continuous(trans="reverse")  +
             geom_hline(yintercept = 0.6, linetype = 1) + theme_void() +
             theme(plot.margin = margin(5, 0, 35, 0))

fpfinal <- grid.arrange(data_left, plot, data_right, top = "Figure 1. Forest Plot with Advanced Options", ncol = 3)

ggsave(fpfinal, file=".//forestplot.pdf", width = 12, height=6, dpi=300)
```

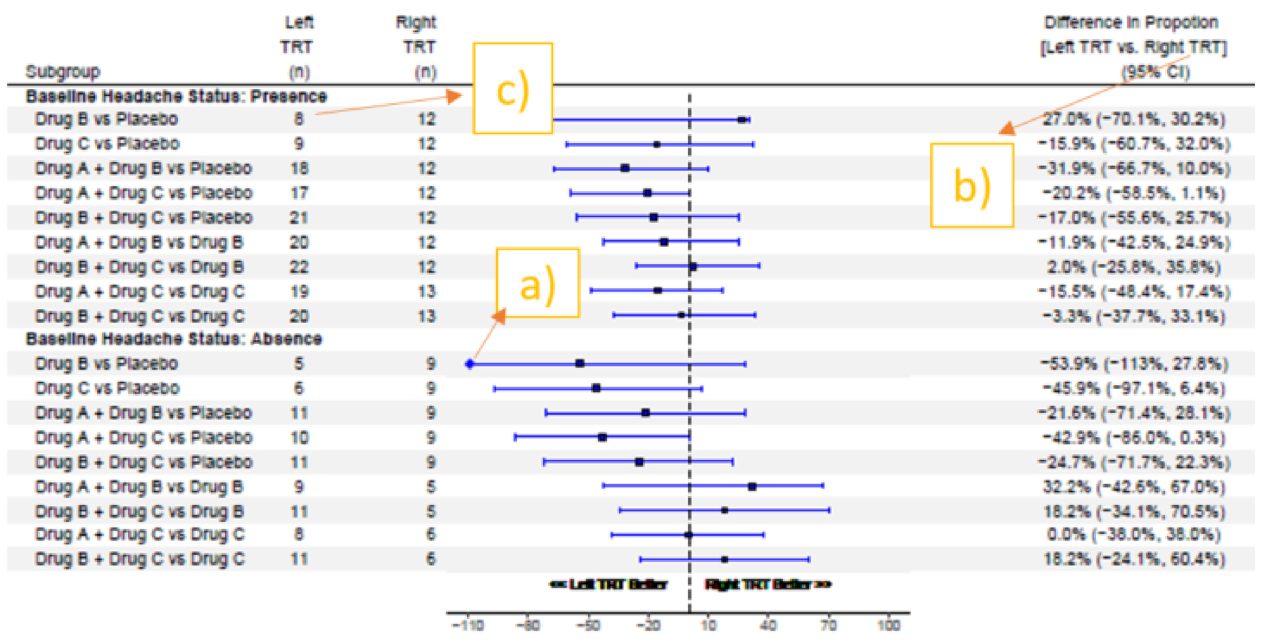
output: SAS



Subgroup	Left TRT (n)	Right TRT (n)	Difference in Proportion [Left TRT vs. Right TRT] (95% CI)
Baseline Headache Status: Presence			
Drug B vs Placebo	8		27.0% (-70.1%, 30.2%)
Drug C vs Placebo	9		-15.9% (-60.7%, 32.0%)
Drug A + Drug B vs Placebo	18	12	-31.9% (-66.7%, 10.0%)
Drug A + Drug C vs Placebo	17	12	-20.2% (-58.5%, 1.1%)
Drug B + Drug C vs Placebo	21	12	-17.0% (-55.6%, 25.7%)
Drug A + Drug B vs Drug B	20	12	-11.9% (-42.5%, 24.9%)
Drug B + Drug C vs Drug B	22	12	2.0% (-25.8%, 35.8%)
Drug A + Drug C vs Drug C	19	13	-15.5% (-48.4%, 17.4%)
Drug B + Drug C vs Drug C	20	13	-3.3% (-37.7%, 33.1%)
Baseline Headache Status: Absence			
Drug B vs Placebo	5	9	-53.9% (-113%, 27.8%)
Drug C vs Placebo	6	9	-45.9% (-97.1%, 6.4%)
Drug A + Drug B vs Placebo	11	9	-21.6% (-71.4%, 28.1%)
Drug A + Drug C vs Placebo	10	9	-42.9% (-86.0%, 0.3%)
Drug B + Drug C vs Placebo	11	9	-24.7% (-71.7%, 22.3%)
Drug A + Drug B vs Drug B	9	5	32.2% (-42.6%, 67.0%)
Drug B + Drug C vs Drug B	11	5	18.2% (-34.1%, 70.5%)
Drug A + Drug C vs Drug C	8	6	0.0% (-38.0%, 38.0%)
Drug B + Drug C vs Drug C	11	6	18.2% (-24.1%, 60.4%)

output: R

Figure 1. Forest Plot with Advanced Options



Subgroup	Left TRT (n)	Right TRT (n)	Difference In Propotion [Left TRT vs. Right TRT] (95% CI)
Baseline Headache Status: Presence			
Drug B vs Placebo	8	12	27.0% (−70.1%, 30.2%)
Drug C vs Placebo	9	12	−15.9% (−60.7%, 32.0%)
Drug A + Drug B vs Placebo	18	12	−31.9% (−66.7%, 10.0%)
Drug A + Drug C vs Placebo	17	12	−20.2% (−58.5%, 1.1%)
Drug B + Drug C vs Placebo	21	12	−17.0% (−55.6%, 25.7%)
Drug A + Drug B vs Drug B	20	12	−11.9% (−42.5%, 24.9%)
Drug B + Drug C vs Drug B	22	12	2.0% (−25.8%, 35.8%)
Drug A + Drug C vs Drug C	19	13	−15.5% (−48.4%, 17.4%)
Drug B + Drug C vs Drug C	20	13	−3.3% (−37.7%, 33.1%)
Baseline Headache Status: Absence			
Drug B vs Placebo	5	9	−53.9% (−113%, 27.8%)
Drug C vs Placebo	6	9	−45.9% (−97.1%, 6.4%)
Drug A + Drug B vs Placebo	11	9	−21.6% (−71.4%, 28.1%)
Drug A + Drug C vs Placebo	10	9	−42.9% (−86.0%, 0.3%)
Drug B + Drug C vs Placebo	11	9	−24.7% (−71.7%, 22.3%)
Drug A + Drug B vs Drug B	9	5	32.2% (−42.6%, 67.0%)
Drug B + Drug C vs Drug B	11	5	18.2% (−34.1%, 70.5%)
Drug A + Drug C vs Drug C	8	6	0.0% (−38.0%, 38.0%)
Drug B + Drug C vs Drug C	11	6	18.2% (−24.1%, 60.4%)

Conclusion:

Differences: Some of the observed limitations are indicated below.

a) SAS is using the left filled triangle where R doesn't have that symbol so we have to go with a diamond symbol in the above plot (for values which are going beyond the minimum value i.e. -110). With R, we have tried Unicode U+25c4 and it looked good in the plot window but we have problems when outputting with ggsave.
b) SAS is wrapping the labels of the last column which is not the problem with R as we are passing the values as row values rather than labels of x-axis.
c) In SAS every column is allotted some space so second column will start only after the first column allotted space which is not the case with R, which can help us save some space.

Time: To create the above plot, SAS took less time when compared to R. And also when SAS code is embedded In sophisticated macros can save more time.

Quality: Almost similar but SAS as an edge over R as we are creating different parts of the output, every time and in the end we are attaching all of them to get the desired output which might reduce the quality.

Resources: R has an advantage as it is an open source software and free updates are available from time to time.

Kaplan Meier plot:

For data, we are going to use sashelp.bmt dataset and modify to our needs.

```
proc format;
   invalue trtfrm
         'AML-High Risk' =1
         'AML-Low Risk'  =2
         'ALL'           =3
        ;
run;

data final;
   set sashelp.bmt;
   subjid=_n_;
   aval=(t/365.25);
   avalM=(t/365.25/12);
   censor=status;
   trtn=input(group,trtfrm.);
run;
```

SAS Syntax: Parts of GTL template code is displayed below.

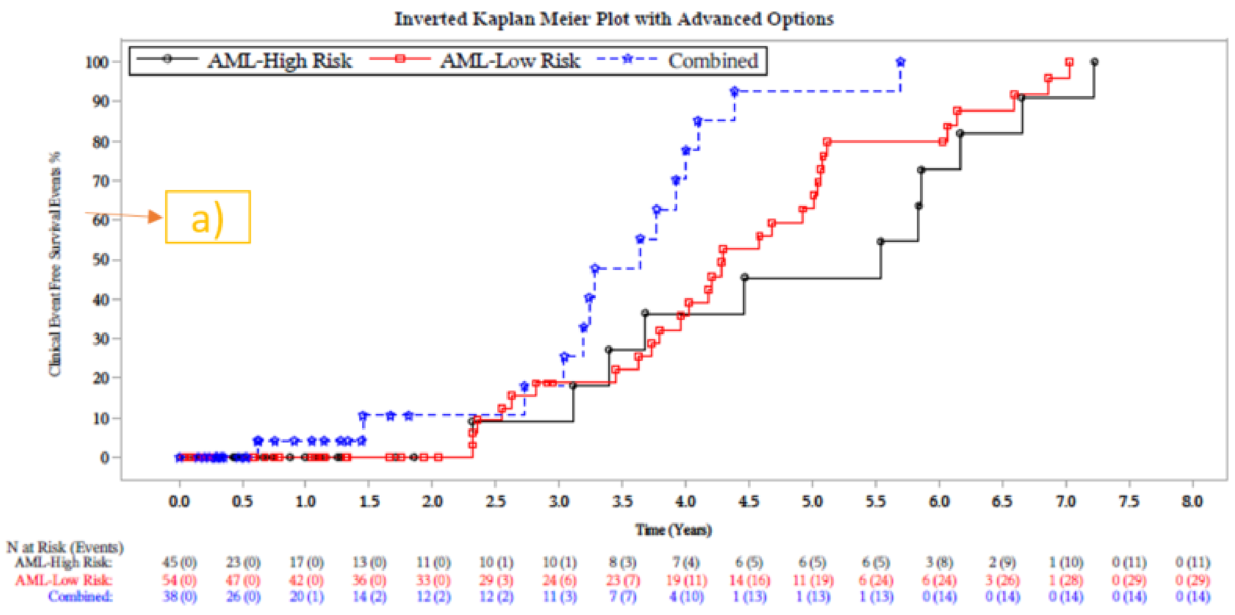
Following code is the important part of proc template.

```
  layout lattice / columndatarange=unionall rowweights=&rowweights.;
    layout overlay / walldisplay=(outline)
     yaxisopts=(offsetmin=0.05 offsetmax=0.05
     linearopts=(tickvaluesequence=(start=0 end=100 increment=10) tickvaluepriority=true) label=" ")
     xaxisopts=(label="&xlabel" labelattrs=(size=8 weight=Normal) offsetmin=0.05 offsetmax=0.03
     linearopts=(tickvaluesequence=(start=&xlv end=&xuv increment=&xinc)));
    stepplot y=survival x=aval/group=trta display=(markers) name="step" markerattrs=(size=7);annotate;
  discretelegend "step" / location=inside across=&lablcols valueattrs=(size=12) autoalign=(topleft) border=TRUE
   borderattrs=(thickness=1 color=black); endlayout;

     layout overlay;
       entry halign=left textattrs=(size=9.5) 'N at Risk (Events)';
     endlayout;

     layout overlay / walldisplay=none border=false yaxisopts=(display=none)
      xaxisopts=(display=none offsetmin=0.05 offsetmax=0.03
            linearopts=(tickvaluesequence=(start=&xlv end=&xuv increment=&xinc) tickvaluepriority=true));
                 %if left_fail^='' %then %do;
                          axistable value=left_fail x=aval/class=trtb colorgroup=trtb labelattrs=(size=8)
                                    valuejustify=right labeljustify=right valueattrs=(size=8);
                 %end;
                endlayout;
   endlayout;
```

R Syntax: Load the required libraries including ggplot2, patchwork etc. before we run the following code.

```
km <- ggplot(surv, aes(group=trta, y=SURVIVAL,x=aval,colour=trta,linetype=trta)) +
   geom_step() + geom_point(aes(shape=as.factor(trta))) +
   scale_shape_manual("", values=c(1,0,11)) + scale_color_manual("", values=c("black","red","blue")) +
   scale_linetype_manual("", values=c(1,1,2)) + scale_y_continuous(breaks=seq(0,100,10),limits=c(0,100)) +
   scale_x_continuous(breaks=seq(0,8,0.5), limits=c(0,8)) +
   xlab("\nTime (Years)") + ylab("Clinical Events Free Survival Events%") +
   labs(subtitle ="Inverted Kaplan Meier Plot with Advanced Options") + theme_bw() +
   theme(legend.justification=c(0,1), legend.position=c(0.01, 0.99),
     legend.direction = "horizontal", legend.text.align = 1,
     plot.subtitle = element_text(hjust = 0.5),
     legend.box.margin = margin(t = 0, r = 0, b = 0, l = 0, unit = "pt"),
     legend.background = element_rect(linetype = 1, size = 0,color = 1),
     legend.key = element_blank(),legend.spacing.x = unit(0.3, 'cm'),
     panel.grid = element_blank(), text=element_text(family="Times"))

atrisk <- ggplot(subset(surv, !is.na(left_fail)), aes(color=trtb)) +
       geom_text(aes(label=left_fail, x=aval,y=trtb), show.legend=F, size=3) +
       scale_x_continuous(breaks=seq(0,8,0.5), limits=c(0,8)) +
       scale_color_manual("", values=c("black","red","blue")) +
       ylab(" ") + xlab(" ") + theme_bw() + scale_y_discrete(limits = rev) +
       labs(subtitle = "N at Risk (Events)") +
       theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
       axis.line= element_blank(), axis.text.x = element_blank(), axis.ticks = element_blank(),
       panel.border=element_blank(), plot.title.position = "plot",
       plot.subtitle = element_text(hjust = 0.007,vjust = -1),axis.text.y = element_text(color="black"),
       plot.caption = element_text(hjust = 0, family="Times"),
       text=element_text(family="Times"))

km2 <- km + atrisk + plot_layout(ncol=1, heights=c(1,0.20))

ggsave(km2, file=".//KMplot.pdf", width = 10, height=5, dpi=300)
```

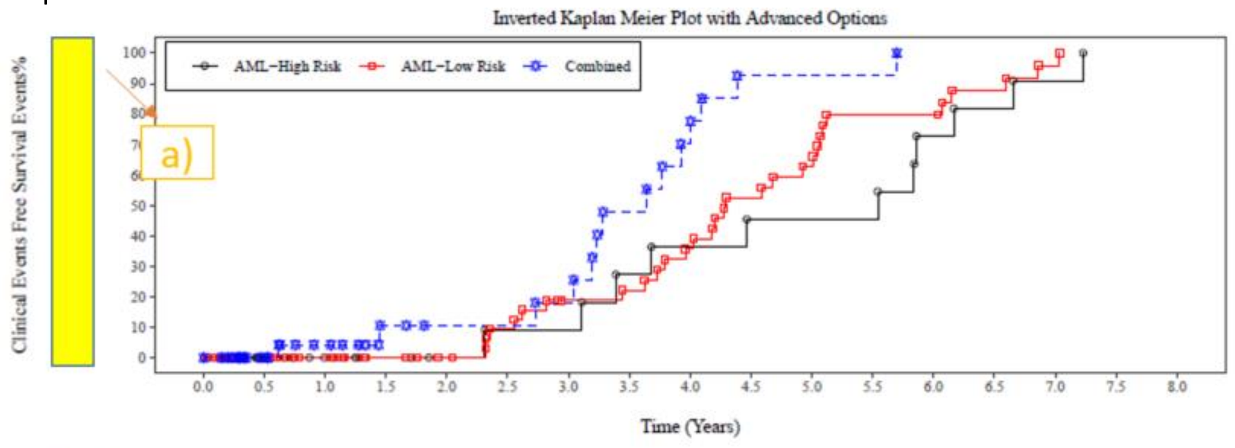
output: SAS



N at Risk (Events)

AML-High Risk:	45 (0)	23 (0)	17 (0)	13 (0)	11 (0)	10 (1)	10 (1)	8 (3)	7 (4)	6 (5)	6 (5)	6 (5)	3 (8)	2 (9)	1 (10)	0 (11)	0 (11)
AML-Low Risk:	54 (0)	47 (0)	42 (0)	36 (0)	33 (0)	29 (3)	24 (6)	23 (7)	19 (11)	14 (16)	11 (19)	6 (24)	6 (24)	3 (26)	1 (28)	0 (29)	0 (29)
Combined:	38 (0)	26 (0)	20 (1)	14 (2)	12 (2)	12 (2)	11 (3)	7 (7)	4 (10)	1 (13)	1 (13)	1 (13)	0 (14)	0 (14)	0 (14)	0 (14)	0 (14)

output: R



N at Risk (Events)

AML−High Risk:	45 (0)	23 (0)	17 (0)	13 (0)	11 (0)	10 (1)	10 (1)	8 (3)	7 (4)	6 (5)	6 (5)	6 (5)	3 (8)	2 (9)	1 (10)	0 (11)	0 (11)
AML−Low Risk:	54 (0)	47 (0)	42 (0)	36 (0)	33 (0)	29 (3)	24 (6)	23 (7)	19 (11)	14 (16)	11 (19)	6 (24)	6 (24)	3 (26)	1 (28)	0 (29)	0 (29)
Combined:	38 (0)	26 (0)	20 (1)	14 (2)	12 (2)	12 (2)	11 (3)	7 (7)	4 (10)	1 (13)	1 (13)	1 (13)	0 (14)	0 (14)	0 (14)	0 (14)	0 (14)

Conclusion:

Differences: Some of the observed limitations are indicated below.

a) Some space is produced in the R output between y-axis and its label (and that space is indicated with a rectangular box filled with yellow) which was adjusted using annotations in SAS.

Time: To create the above plot, R took less time when compared to SAS but when SAS code is embedded in the sophisticated macros which can be handy when we are outputting similar outputs and it’s well worth the time that we spend in creating the macros.

Quality: Almost similar.

Resources: R has an advantage as it is an open source software and free updates are available from time to time.

Final conclusion: In this paper we generated graphs such as Kaplan-Meier and Forest-plot using R and SAS. We provided a step-by-step process and the R code for reading the input dataset and processing the data to generate the graphs. We also provided SAS code, compared the SAS and R graph outputs and discussed challenges and limitations. We demonstrated that R can be used as an effective alternative to create the required graphs in clinical trials. Since R is open source software available in all academic institutions, new software engineers are getting access to R courses easily compared to traditional SAS software (which is predominantly used in Biostatistics/Epidemiology and public health courses). The CRO companies and mid-size pharma companies might adopt R programming for their regular use because of the flexibility that R offers and due to the budget constraints. As programmers, there is always room to adapt to changing environment by learning new languages like R, Python, Spotfire, etc.

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

Author Name: Javvaji, Girija Manohar
Labcorp Drug Development Inc. (previously Covance)
4000 Centergreen Way # 300
Cary, NC 27513
Work Phone: 224-280-4518
Email: girija.javvaji@labcorp.com

Co-Author Name: Konda, Sivasankar
Labcorp Drug Development Inc. (previously Covance)
4000 Centergreen Way # 300
Cary, NC 27513
Work Phone: 513-426-4512
Email: siva.konda@labcorp.com