

Customizing Survival Plot by %Survivalplot Macro Zhong Yan, i3, Indianapolis, IN

Abstract

Several macros were reported to generate Kaplan-Meier plots. However, the existing macros typically draw up to two groups with limited flexibility for the summary statistics and curve style. This paper introduces the new macro %Survivalplot. The major features are: 1. It can draw up to nine survival curves with median survival time, number of events and censored subjects, and the number of patients at risk at each time; 2. For two group comparison, it can also show hazard ratio and the log-rank test p value; 3. User can easily update the color and line pattern for each curve; 4. User has the flexibility to control the location of the summary statistics and its font size on the graph; 5. The plot data can be exported to an excel file; 6. User can enter up to four titles and four footnotes.

Introduction

PROC LIFETEST procedure is often used to create a simple Kaplan-Meier survival plot. However, sometimes the investigators prefer to customize the line pattern and color for each curve, or request to show summary statistics on the plot, for example, median survival time, number of censored or event subjects, hazard ratio for two treatment comparisons. In these cases, a simple PROC LIFETEST is not enough to do the magic.

In the past, papers have been published to introduce macros to customize the Kaplan-Meier survival plot using PROC GPLOT procedure. However the reported macros typically worked on one or two groups with limited flexibility to display the summary statistics and control the curve patterns and colors at the same time.

During exploratory analysis, investigators might be interested in visualizing three or more plots on the same graph with key summary statistics displaying. Based on exploratory analysis, the investigators might need to focus on two of the groups for further comparison.

The %Survivalplot macro has been developed to overcome the above limits. User can easily customize up to nine Kaplan-Meier survival plots on a single graph with summary statistics displaying, or select any two groups for further analysis. This paper will describe the structure and algorithm of the macro and show some examples for the usage.

Simulated Survival Data Preparation

The code below was used to generate simulated survival datasets. The datasets will be used as the input for the %Survivalplot macro.

```
%macro prepdata(group=, md=, cutoff=, pat=, seed=123);

  data all(keep=group time censor);
    %do i= 1 %to &group;
      group=&i;
      %let md&i =%scan(&md, &i, "|");
      lambda&i =&&md&i /(log(2));

      %do pat = 1 %to &pat;
        t = ranexp(&seed)*lambda&i;
        x=floor(10*ranuni(0));
        if x>9 then censor=1;
        if t<=&cutoff then do;
          censor =0;
          time=t;
        end;
        if t>&cutoff then do;
          censor =1;
          time=&cutoff;
        end;
        output;
      %end;
    %end;
run;
```

```

data all2;
  set all(where=(group<=2));
run;

%mend prepdata;

%prepdata(group=9,md=18|15|14|13.8|19|16|17|16.5|15.5,cutoff=24,pat=100,seed=123);

```

The above code assumes survival time to be exponentially distributed. The study cutoff time is set to 24 months from the starting. The simulated time to event datasets “all” and “all2” contain nine and two groups of patients respectively. Each group contains 100 patients. Both datasets contain 3 required variables: group, time, and censor. The group variable contains the numeric group ID (for example, 1 and 2). The time variable stores the event time or the censor time from the beginning of the study. The censor variable is a flag to indicate if it is an event or a censor. In our examples, 1 is assumed to be censored and 0 indicates event.

group	censor	time
2	0	16.000214176
2	0	11.801752519
2	0	1.5553314175
2	1	24

Figure 1. Structure of the Data

Macro Parameters

The table below describes the macro %survivalplot parameters.

Table 1. Macro Parameters

Parameter	Required?	Default Value	Description
ds	Yes		time to event dataset name
timevar	No	time	time variable name in the dataset ds
sensorvar	No	censor	censor variable name in the dataset ds
sensorflagind	No	1	the sensorvar value that indicates censored
groupvar	No		Group variable name in ds with sequential numeric numbers for values, for 2 group comparisons, interest group must have value 1 and reference group must have value 2. Blank meant all observations are in the same group
timeunit	No	Month	the time unit
timeatck	Yes	6	the time interval in the x-axis to draw a tick mark and calculate the number of patients at risk
ypercentscale	No	N	the tick number format on Y-axis (the default value N means in decimal format, Y means in percentage)
ylabel	No	Probability of Survival	Y-axis label
figpathname	Yes		figure path and file name (excluding file extension)
figtype	No	cgm	type of output figures (valid values: cgm, eps)
annolabel	No	Group 1 Group 2 ... Group9	
risklabel	No	G1 G2 ... G9	
curve_pattern	No	1 20 1 20 1 20 1 20 1	
curve_color	No	black black gray gray pink pink green green red	
fonttype	No	simplex	font type of the graph text

annosize	No	0.7	the font size of the annotation for the summary statistics on the graph
anno_xstart	No	45 or 52	starting position in percentage of the X-axis for the first column of the annotate dataset
anno_ystart	No	92 or 85	starting position in percentage of the Y-axis for the first row of the annotate dataset
anno_xinterval	No	17 or 10	interval in percentage of the X-axis for the neighboring two columns of the annotate dataset
anno_yinterval	No	3	interval in percentage of the Y-axis for the neighboring two rows of the annotate dataset
href	No		vertical reference line on the graph based on the value provided (default is not to draw reference line)
vref	No		horizontal reference line on the graph based on the value provided (default is not to draw reference line)
showhr	No	N	choice if the hazard ratio is needed to show on the graph when the input data has only two groups
tiephreg	No	Breslow	the tie handling method that will be used to calculate hazard ratio when the dataset only contains two groups
showlogrankp	No	N	show logrank test p value or not (valid value: Y, N)
needxlsout	No	N	if need to output plot data to excel file (valid value: Y, N)
title1	No		title1
title2	No		title2
title3	No		title3
title4	No		title4
footnote1	No		footnote1
footnote2	No		footnote2
footnote3	No		footnote3
footnote4	No		footnote4
titlefootsize	No	2	font size of the titles and footnotes
debug	No	0	0 (default) value means to clean up intermediate datasets and also turn off sas options mprint/mlogic/symbolgen. 1 means to keep the intermediate datasets and turn on sas options mprint/mlogic/symbolgen

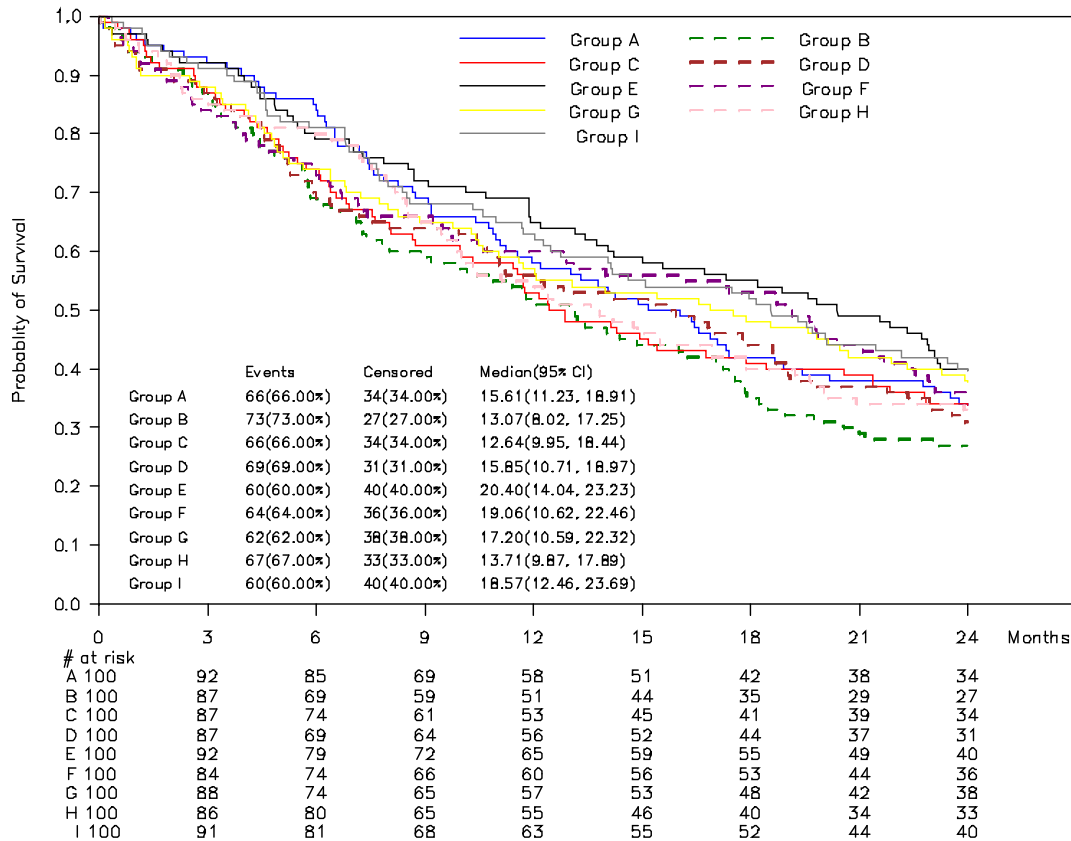
Macro Usage Examples

The macro %survivalplot can be used to generate customized Kaplan-Meier plot with the displaying of summary statistics. It can output the graph to a pre-specified location in cgm or eps format. It can also output the plot data to an external excel spreadsheet. Below are three examples to show the usage of the macro.

Example1 – Customizing the Curves and the Summary Statistics Location

This example used dataset “all” as input. In the macro call below, the parameter “timevar” indicates the variable name that stores survival time. The parameter “censorvar” indicates the variable name that stores censoring information.

“censorflagind” indicates the value of the censor variable that represents for the censored subject. “figpathname” stores the path and the file name of the output figure. “annolabel” contains the label for each group that will be displayed for summary statistics. “risklabel” stores the labels that will be displayed below X-axis to indicate the number of patients at risk at each time spot for each group. The parameter “curve_color” specifies unique color for each curve. The line pattern of each curve can also be customized (see example 3). The location of the summary statistics (number of events and censored subjects, median survival and its 95% confidence interval) on the graph was determined by the four parameters with names starting with “anno_”. “anno_xstart” and “anno_ystart” indicate the start location of the first column and the first row of the summary statistics on the graph, which are in percentages of the X-axis or Y-axis to the origin. “anno_xinterval” and “anno_yinterval” indicates the column or row interval in percentages of the X-axis and Y-axis respectively. In this example, title is not used and only one footnote is displayed with default font size. In figure 2, the summary statistics is located at the bottom left corner. Without specifying the location of the summary statistics, it will be displayed at the top right corner of the graph by default. “timeatck=3” means to draw tick markers on the X-axis every three months.



Kaplan-Meier Survival Plot

Figure 2. Customizing Curves and Summary Statistics Location

```
%kmpplot(ds=all,
    timevar=time,
    censorvar=censor,
    censorflagind=1,
    groupvar=group,
    timeunit=month,
    timeatck=3,
    ylabel=Probability of Survival,
    figpathname=c:\temp\fig2,
    figtype=cgm,
    annolabel=Group A|Group B|Group C|Group D|Group E|Group F|Group G|Group H|
              Group I,
    risklabel=A|B|C|D|E|F|G|H|I,
    curve_color=blue|green|red|brown|black|purple|yellow|pink|gray,
    annosize=0.7,
    anno_xstart=3,
```

```

anno_xinterval=12,
anno_ystart=40,
anno_yinterval=4,
footnotel=Kaplan-Meier Survival Plot);

```

Example2 – Displaying Hazard Ratio for Two Group Comparisons

For two group comparisons, the macro can calculate the hazard ratio and its 95% confidence interval. Log rank test can be performed to produce p value. The summary statistics was shown on the top right corner of the graph (see figure 3). In the macro call, the values of the parameters “showhr” and “showlogrankp” control whether or not the hazard ratio and the p value from the log rank test will be displayed on the graph. “tiephreg” indicates the tie handling method when hazard ratio is calculated. If it is not provided, SAS default method will be used.

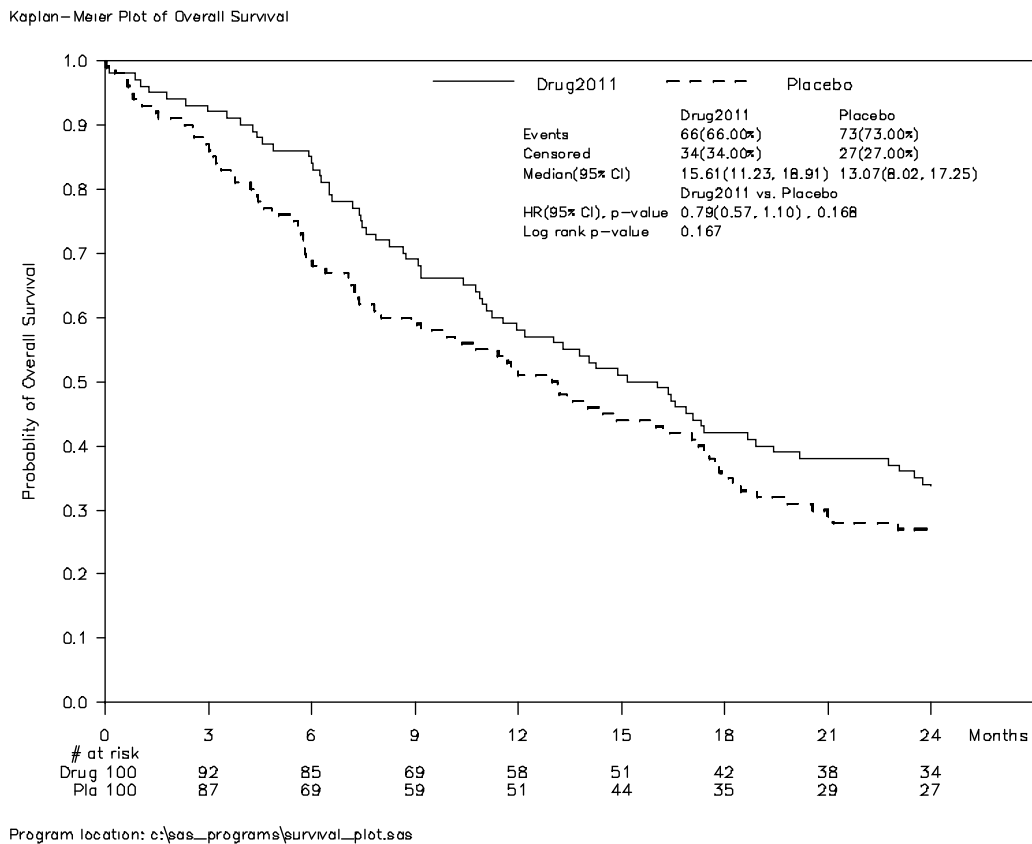


Figure 3. Display Hazard Ratio for Two Group Comparison

```

%kmplot(ds=all2,
timevar=time,
censorvar=censor,
censorflagind=1,
groupvar=group,
timeunit=month,
timeatck=3,
ypercentscale=,
ylabel=Probablity of Overall Survival,
figpathname=c:\temp\fig3,
figtype=cgm,
annolabel=Drug2011|Placebo,
risklabel=Drug|Pla,
annosize=0.7,
showhr=Y,
tiephreg=efron,
showlogrankp=Y,
title1=Kaplan-Meier Plot of Overall Survival,

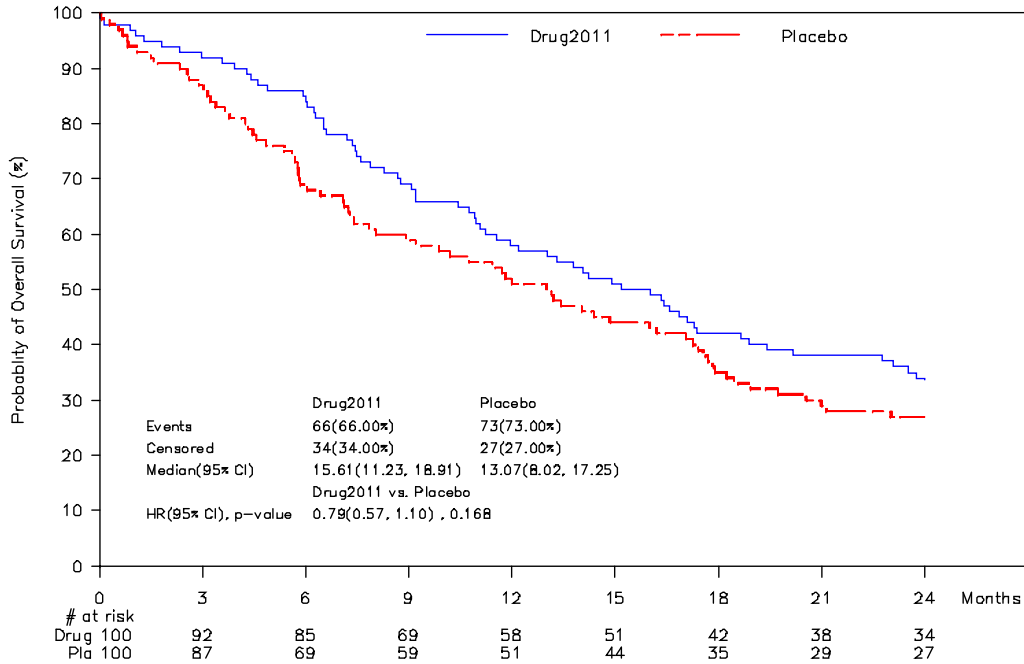
```

footnote1=Program location: c:\sas_programs\survival_plot.sas);

Example3 –Easy Update to New Graphical Look

This example shows how easy it is to update the look of the graph. Compared to figure 3, the figure 4 has the line pattern and color both updated. The location of the summary statistics is now on the bottom left corner. The p value is not showing anymore. The tick mark labels of the Y-axis have been updated. There are more footnotes. The font size for the title and footnotes is bigger.

Kaplan–Meier Plot of Overall Survival



Program location: c:\sas_programs\survival_plot.sas
 Output location: c:\sas_programs\groups2_a.cgm

Figure 4. New Graphical Look

```
%kmplot(ds=all2,
  timevar=time,
  censorvar=censor,
  censorflagind=1,
  groupvar=group,
  timeunit=month,
  timeatck=3,
  ypercentscale=Y,
  ylabel=Probability of Overall Survival (%),
  figpathname=c:\temp\fig4,
  figtype=cgm,
  annolabel=Drug2011|Placebo,
  risklabel=Drug|Pla,
  curve_pattern=1|30,
  curve_color=blue|red,
  anno_xstart=5,
  anno_xinterval=18,
  anno_ystart=30,
  anno_yinterval=4,
  showhr=Y,
  tiephreg=efron,
  showlogrankp=N,
```

```

annosize=0.7,
title1=Kaplan-Meier Plot of Overall Survival,
footnote1=Program location: c:\sas_programs\survival_plot.sas);
footnote2=Output location: c:\sas_programs\groups2_a.cgm,
titlefontsize=4,
footfontsize=4);

```

Macro Algorithm Flowchart

The flow chart below describes the macro %survivalplot algorithm.

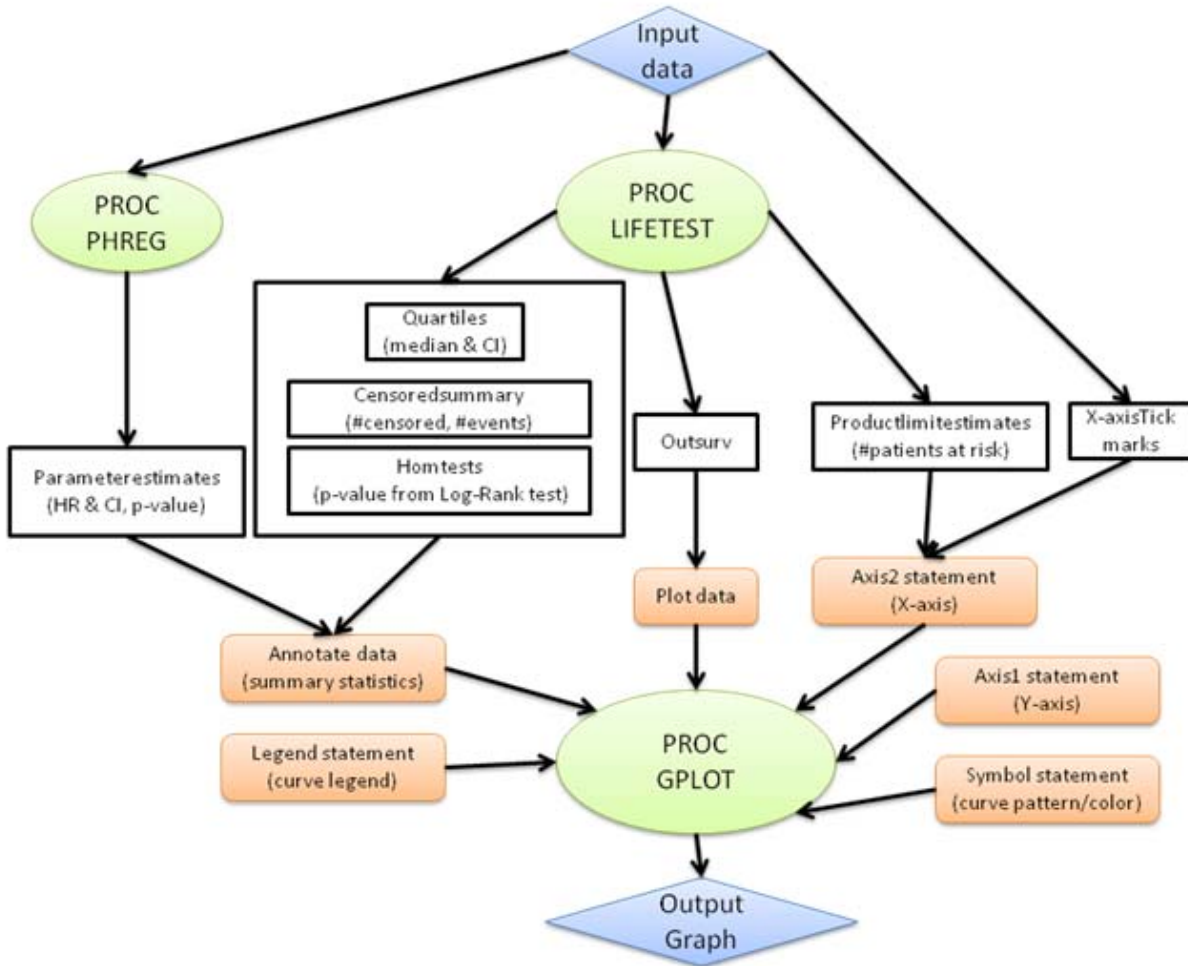


Figure 4. Algorithm Flow Chart

Examples of the Code Pieces

Using ODS OUTPUT to Store Analysis Results from PROC LIFETEST and PROC PHREG

The code shows the survival analysis using PROC LIFETEST and PROC PHREG. The analysis results are stored in a set of datasets using ODS OUTPUT. The macro variable "timelist" was calculated based on the macro "timeatck" and the maximum time from the input dataset. The code for the calculation is shown in example2.

```

data ds(keep=group time censor );
  set &ds;
  time=&timevar;

```

```

censor=&cursorvar;

%if "&groupvar" eq "" %then %do;
    group=1;
%end;
%else %do;
    group=&groupvar;
%end;

run;

proc sql noprint;
    select count(distinct group) into :ngroups
    from ds;
quit;

ods output productlimitestimates=estimate quartiles=quart homTests=logrank
    censoredSummary=censor;

proc lifetest data=ds outsurv = outsurv timelist=&timelist;
    time time*censor(&cursorflagind) ;
    strata group;
run;

%if %upcase("&showhr")="Y" and &ngroups eq 2 %then %do;

    data ds;
        set ds;
        if group=1 then hrgroup=1; *Interest group;
        if group=2 then hrgroup=0; *Reference group;
    run;

    ods output ParameterEstimates=pest(drop=df);

    proc phreg data=ds;
        model time * censor(&cursorflagind) = hrgroup /
            %if "&tiephreg" ne '' %then %do; ties=&tiePhreg %end; r1;
    run;

    data _null_;
        set pest;
        hr=compress(put(round(hazardratio, 0.01), 5.2)) || "(" ||
            compress(put(round(hrlowercl, 0.01), 4.2)) || ", " ||
            compress(put(round(hruppercl, 0.01), 4.2)) || ")";
        call symput('hr', hr);

        if probchisq < 0.001 then pv ="<0.001";
        else pv=compress(put(probchisq, 5.3));

        call symput('pv', pv);
    run;

%end;

```

Create Dataset for X-axis Tick Marks

The code below creates dataset "times" that stores X-axis tick marks for time spots. A macro variable "timelist" is also created to store time spots that need to show the number of patients at risk.

```

%* maxtime: the maximum survival time from the input dataset;
proc sql noprint;
    select ceil(max(time)) format=3.0 into :maxtime
    from ds;
quit;

%* Calculate total tick marks on the x axis;

```



```

data _null_;
    nticks=int(&maxtime/&timeatick) ;
    call symput('nticks',trim(left(put(nticks,3.0))));
run;

%let tickn=%eval(&nticks*&timeatick);

/* in case of last value being a multiple of &timeatick;
%let tickn1 = %eval(&tickn+&timeatick);

* Calculate time at each tick for each group;
* Assume ngroups are <= 9 to fit the current ij format;
* Assume nticks are <100 to fit the current ij format;

data times(rename=(i=stratum));
    length timelist $100.;

    do i=1 to &ngroups;
        timelist="";

        do j=0 to &nticks ;
            time = j * &timeatick;
            flag=1;

            tmp=compress(put(time, 4.));
            timelist= strip(timelist) || ' ' || tmp;

            if j<=9 then jj="0" || compress(put(j,1.0));
            else if j<100 then jj=compress(put(j,2.0));

            ij=compress(put(i,1.0)) || jj;

            output;
            call symput('tick' || left(put(j,3.0)), trim(left(put(time,3.0))));
        end;

    end;

    drop j jj;
    call symput('timelist', timelist);
run;

proc sort data=times;
    by stratum time;
run;

```

Legend Statement

The code below shows the legend statement that will be used by PROC GPLOT to define the labels for each curve.

```

data _null_;
    down=ceil(&ngroups/2);
    call symput('down',put(down, 2.));
run;

%let legend_stat=legend1 mode=share position=(top center inside) offset =(7)
across=2 down=&down label=none Value=(;

data _null_;
    legdlabell= '' || tranwrd("&annolabel", '|', ' ' ' ') || '';
    call symput('legdlabell', legdlabell);
run;

%let legend_stat=%cmpres(&legend_stat &legdlabell %str());

```

Symbol Statement

The symbol statement will be used by PROC GPLOT to customize the curve styles (colors and line patterns).

```
%let symbol_stat=;

%do i=1 %to &ngroups;

    %let pattern&i=%scan(&curve_pattern, &i, "|");
    %let color&i=%scan(&curve_color, &i, "|");
    %let line&i=&&pattern&i width=2 ci= &&color&i;
    %let symbol_stat=&symbol_stat symbol&i color=black line= &&line&i
        interpol=join value=none %str(;;);

%end;
```

Axis2 Statement

Below code shows how X-axis tick marks and the number of patients at risks are matched with the time spots on the X-axis. The result is stored in the macro variable "ax2_stat" and will be used by PROC GPLOT.

```
%do i= 1 %to &ngroups;

    %let header&i=%scan(&risklabel, &i, "|");
    %let anno&i=%scan(&annolabel, &i, "|");

%end;

%let ax2_stat = axis2 offset=(0,0) label=none minor = none order=(0 to &tickn by
&timeatck &tickn1) value=(;

%do i=0 %to &nticks;

    %let tickstat&i = %str(tick=) %eval(&i+1) j=c "&&tick&i" j=c ;

    %if &i=0 %then %do;
        %let tickstat&i=&&tickstat&i '# at risk' j=c "&header1" " &risk100";
        %do j=2 %to &ngroups;
            %let tickstat&i=&&tickstat&i j=c "&&header&j" " &&risk&j.00";
        %end;
    %end;
    %else %do;
        %if &i < 10 %then %do;
            %let tmp=0&i;
        %end;
        %else %do;
            %let tmp=&i;
        %end;
        %let tickstat&i=&&tickstat&i ' ' j=c "&&risk1&tmp";
        %do j=2 %to &ngroups;
            %let tickstat&i=&&tickstat&i j=c "&&risk&j&tmp";
        %end;
    %end;

%end;

%end;

data _null_;
    unit="&timeunit";

    if length(strip(unit)) eq 1 then newunit=unit;
    else do;
        letter1=upcase(substr(unit,1,1));
        letterrest=lowcase(substr(unit,2));
        len=length(unit);
        letterlast=lowcase(substr(unit,len,1));
        if letterlast eq "s" then newunit= compress(letter1) || compress(letterrest);
        else newunit=compress(letter1) || compress(letterrest) || "s";
    end;
```

```

end;
    call symput('_unit', newunit);
run;

%let tickstat&i=%str(tick=) %eval(&i+1) j=c "&_unit") %str(;);

%do i=0 %to &nticks+1;
    %let ax2_stat=&ax2_stat &&tickstat&i;
%end;

```

PROC GPLOT Procedure

Below code shows PROC GPLOT used in the macro %Survivalplot. The part of code to process titles, footnotes, href, and vref are not listed here.

```

ods listing;
filename gsf "&figpathname.&figtype";

goptions reset =all
    gsfname =gsf
    device= %if %upcase("&figtype") eq "CGM" %then %do; cgmofml %end;
           %else %if %upcase("&figtype") eq "EPS" %then %do; psepsf %end;
    cback=white ctext=black noborder ftext=&fonttype ftitle=&fonttype
    hsize=8 in vsize=6.4 in gunit=pct htext=2.2 htitle=2.2 colors =(black)
    gsflen=80 gprotocol=sasgpasc display gsfmode=replace;

proc gplot data=plot100 annotate=anno; &legend_stat; &symbol_stat;
    axis1 offset=(0,0) order = %if %upcase("&ypercentscale") eq "Y" %then %do;
        (0 to 100 by 10)
    %end;
    %else %do;
        (0 to 1 by 0.1)
    %end;

    minor=none label=(angle=90 "&ylabel" );
    &ax2_stat;

    %if %upcase("&ypercentscale") eq "Y" %then %do;
        plot survival100 *time=stratum /vaxis = axis1 haxis = axis2 legend=legend1
    %end;
    %else %do;
        plot survival*time=stratum /vaxis = axis1 haxis = axis2 legend=legend1
    %end;
run; quit;

```

Below shows the example code after replacing all of the macro variables (titles and footnotes are not included).

```

ods listing;
filename gsf "c:\tmp\groups2.cgm";
goptions reset =all gsfname =gsf device= cgmofml cback=white ctext=black noborder
ftext=simplex ftitle=simplex hsize=8 in vsize=6.4 in gunit=pct htext=2.2 htitle=2.2
colors =(black) gsflen=80 gprotocol=sasgpasc display gsfmode=replace;

proc gplot data=plot100 annotate=anno;
legend1 mode=share position=(top center inside) offset = (7) across=2 down= 1
label=none Value=("Drug2011" "Placebo" );
symbol1 color = black line = 1 width=2 ci= black interpol = join value = none ;
symbol2 color = black line = 20 width=2 ci= black interpol = join value = none ;

axis1 offset=(0,0) order = (0 to 1 by 0.1) minor=none label=(angle=90 "Probablity of
Overall Survival");
axis2 offset=(0,0) label=none minor = none order=(0 to 24 by 3 27) value=( tick= 1
j=c "0" j=c '# at risk' j=c "Drug" " 100" j=c "Pla" " 100" tick= 2 j=c "3" j=c ' '
j=c "92" j=c "87" tick= 3 j=c "6" j=c ' ' j=c "85" j=c "69" tick= 4 j=c "9" j=c ' '
j=c "69" j=c "59" tick= 5 j=c "12" j=c ' ' j=c "58" j=c "51" tick= 6 j=c "15" j=c
' ' j=c "51" j=c "44" tick=7 j=c "18" j=c ' ' j=c "42" j=c "35" tick=8 j=c "21" j=c

```

```
' ' j=c "38" j=c "29" tick=9 j=c "24" j=c ' ' j=c "0" j=c "0" tick= 10 j=c "Month");  
plot survival*time=stratum /vaxis = axis1 haxis = axis2 legend=legend1;  
run;quit;
```

Conclusion

%Survivalplot is the macro developed to create Kaplan-Meier survival plots. This paper introduced the main features of the macro, its usages through examples, and sample code pieces. The algorithm flow chart showed how the macro was organized to produce Kaplan-Meier survival plot. Through the examples it can be seen that this macro is easy to use and convenient to generate graphs with summary statistics.

Acknowledgements

I'd like to thank my Oncology teammates for their valuable support.

Contact Information

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

Zhong Yan
i3
Office: (317) 564-2858
E-mail: ohzhong@yahoo.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.