

# Kaplan-Meier プロットに付加情報を追加するマクロの作成

○長島 健悟<sup>1</sup>, 佐藤 泰憲<sup>2,3</sup>

<sup>1</sup> 城西大学 薬学部 薬科学科

<sup>2</sup> 千葉大学 医学部

<sup>3</sup> ハーバード大学 公衆衛生大学院 生物統計部門

## A SAS macro for extended Kaplan-Meier plots

Kengo Nagashima<sup>1</sup>, Yasunori Sato<sup>2,3</sup>

<sup>1</sup> Department of Pharmaceutical Technochemistry, Josai University

<sup>2</sup> School of Medicine, Chiba University

<sup>3</sup> Department of Biostatistics, Harvard School of Public Health

### 要旨

生存時間データの解析結果を報告する際には、生存曲線(多くの場合 Kaplan-Meier プロット)、生存期間中央値、リスク集合の大きさ、生存時間の群間比較の検定(例えば、log-rank 検定の  $P$  値)、ハザード比などを示すことが望ましい。実際に、主要医学雑誌や新薬の承認申請などに生存時間データの解析結果を示す場合、Kaplan-Meier プロットと上に挙げた項目等を合わせて示すことが要求される。したがって、統計解析ソフトウェアで簡便に出力できる事が望まれる。

SAS 9.2 の LIFETEST Procedure では、ODS Graph を利用することで、リスク集合のサイズを出力は可能だが、それ以外の付加情報を追加することは難しい。また、SAS 9.1.3 では Kaplan-Meier プロットにリスク集合の大きさを含めた付加情報を追加することができない。そこで、これらの付加情報を追加することができる SAS マクロを作成した。

本マクロは、GPLOT Procedure の annotate option を用いて実装しており、上に挙げた項目(生存期間中央値、リスク集合の大きさ、log-rank 検定の  $P$  値、ハザード比)以外の付加情報を加えたい場合にも、容易に拡張が可能である。また、SAS 9.2 の ODS Graph によるリスク集合のサイズの出力と、本マクロの出力の比較を行った結果についても紹介する。

Kaplan-Meier plot; Number at risk; LIFETEST Procedure; GPLOT Procedure; ODS Graph; TEMPLATE Procedure

## 1 %km\_data() マクロ

本稿では、作成したマクロおよび SAS 9.2 でのグラフ描画例を紹介する。すべてのグラフ描画例を実行する場合には、付録のプログラム 7 およびプログラム 8 を事前に読み込んでおく必要がある。

### 1.1 マクロの機能と構成

本マクロは現在 (Ver 2.1.1) 以下の機能を提供している。

1. 付加情報の出力。生存期間中央値、リスク集合の大きさ、生存関数の差の検定 (log-rank test, Wilcoxon test, Likelihood ratio test), Cox の比例ハザードモデルに基づくハザード比
2. 生存関数の信頼区間の出力
3. 打ち切り記号の拡張

%km\_data() マクロを実行すると、GPLOT Procedure 用のデータセットと annotate option 用のデータセットを生成し、それらを用いてグラフ描画を行う構成となっている。マクロ実行の際に必要な引数(入力データセット名、付加情報に関するオプションなど)を表 1 に示す。

表 1: %km\_data() マクロの引数

引数名	説明	引数名	説明
data	入力データセット名	Base	リスク集合の大きさの縦軸出力位置。0を設定すると横軸に接する。
time	[data] 内の生存時間変数名	Step	リスク集合の大きさの表示間隔。重なってしまう場合は値を変える。
stra	[data] 内の層の変数名	Label	リスク集合の大きさの一番上の行に出力されるラベル。
censor	[data] 内の打ち切りの変数名	Test	生存関数の差の検定の出力 (0: 出力しない, 1: 出力する)
censurv	変数 [censor] の打ち切りを表わす値	TestX	生存関数の差の検定の出力位置 (横軸)。データエリアのパーセンテージで指定します。100 が一番右。(以下の出力位置指定も同様)
out	出力データセット名	TestY	生存関数の差の検定の出力位置 (縦軸)。データエリアのパーセンテージで指定します。100 が一番上。(以下の出力位置指定も同様)
anno	出力する annotate データセット名	Type	検定の種類 (logrank: log-rank test, wilcoxon: generalized wilcoxon test, likelihoodratio: likelihood ratio test)
CI	信頼区間の出力 (0: 出力しない, 1: 出力する)	HR	ハザード比の出力 (0: 出力しない, 1: 出力する)。対照群には param = ref = first を指定します。
censEXT	打ち切り記号の拡張 (0: 拡張しない, 1: 拡張する)	HRX	生存関数の差の検定の出力位置 (横軸)。
cHeight	拡張打ち切り記号の高さ	HRY	生存関数の差の検定の出力位置 (縦軸)。
cWidth	拡張打ち切り記号の太さ	MST	生存期間中央値と 95%信頼区間の出力 (0: 出力しない, 1: 出力する)
Size	付加情報の文字サイズ	MlabX	生存期間中央値の層ラベルの出力位置 (横軸)。
afont	付加情報のフォント	MmedX	生存期間中央値の出力位置 (横軸)。
atrisk	リスク集合の大きさの出力 (0: 出力しない, 1: 出力する)	MciX	生存期間中央値の 95%信頼区間の出力位置 (横軸)。
atriskorder	リスク集合の大きさの計算刻み幅	MSTY	生存期間中央値の出力位置 (縦軸)。

ここで、本マクロを利用する際の注意点について述べる。ハザード比の出力においては、層の変数の値をラベルとして出力しているため、出力したいラベルは data step 上で編集しておく必要がある。また、リスク集合のラベルについても各自で設定する必要があるため、群を取り違えないように注意する必要がある。n を層の数とすると、出力されるデータセット [out] には、生存関数の推定値を示す変数 SV1-SVn, 生存関数の信頼区間の下限と上限を示す変数 SL1-SLn, SU1-SUn, 打ち切りを示す変数 CSDF1-CSDFn が含まれる。色の設定については、グローバルマクロ変数 color1-colorn (打ち切り記号拡張用), scolor1-scolorn (信頼区間用) を設定する必要がある。これら [out] に含まれる変数と、annotate データセット [anno] を組み合わせることで、GPLOT Procedure を用いて多彩なグラフを出力することができる。

なお、本マクロは SAS 9.2 上で作成されており、SAS 9.1.3 上では動作確認のみを行った。異なるバージョン間で、出力が微妙に異なるため、複数のバージョンが共存する環境では注意が必要である。

## 1.2 実行例

%km\_data() マクロを用いたグラフの出力例を示す。次節で示す ODS Graph 形式のグラフにハザード比の出力を加えたプログラム 1 を作成し、その出力を図 1 に示した。実行例では、SAS のアウトプットフォントサイズが 12 pt の場合に適切な出力が得られるように調整した。

また、信頼区間の出力や打ち切り記号の拡張を利用しない場合も考えられる。その場合は、変数 SL1-SLn, SU1-SUn や変数 CSDF1-CSDFn を plot statement に指定し、symbol statement で記号や線種を適宜変更する事で対応できる (詳細は Web ページで公開予定のサンプルプログラムに収載した)。

### プログラム 1: %km\_data() マクロによるグラフ出力

```
/* *****  
/* Color setting */  
%global color1 color2 color3 scolor1 scolor2 scolor3;  
  
%let color1 = cx445694;  
%let color2 = cxA23A2E;  
%let color3 = cx01665E;  
  
%let scolor1 = cxD4D9E8;  
%let scolor2 = cxF1CECE;  
%let scolor3 = cxD1E4E3;  
  
/*  
CI: fill, Censor: needle, Atrisk: show,  
Logrank test, Hazardratio  
*/  
%km_data(  
  D1, T, GroupC, Censor, 1,  
  out = graph, anno = anno,  
  
  CI = 1,  
  censEXT = 1,  
  
  Size = 2,  
  atrisk = 1, atriskorder = 0 to 12.5 by 2.5, Step = 5,  
  Label = "No. at risk (1st entry: high, 2nd: middle, 3rd: low)",  
  Test = 1, TestX = 98, TestY = 97, Type = logrank,  
  HR = 1, HRX = 98, HRY = 92  
);  
/* *****  
/* Graph output setting */  
goptions reset = all;  
goptions vsize = 12 in hsize = 19 in htitle = 3.5  
  htext = 3.5;  
options linesize = 98 pagesize = 200;  
  
filename grafout "&Path.km_data.emf";  
goptions device = emf gsfname = grafout gsfmode = replace;  
goptions ftext = "Times New Roman";  
  
proc gplot data = Graph;  
  plot (Sv1 Sv2 Sv3) * T /  
  anno = anno /* autovref cautovref = cxE9DECA */  
  overlay skipmiss noframe legend = legend1  
  haxis = axis1 vaxis = axis2;  
  axis1 label = ('Months after entry')  
  major=(w=2 height=0.7) w=2 minor = none  
  order = (0 to 12.5 by 2.5) offset = (2, 7);  
  axis2 label = (a=90 'Proportion of overall survival')  
  major=(w=2 height=1) w=2 minor = none  
  order = (0 to 1 by 0.2) offset = (2, 2);  
  legend1 label = none position = (inside)  
  mode = protect origin = (4, 1.5)  
  value = (h = 2 "high-risk" "middle-risk" "low-risk");  
  symbol1 i = steplj c="&color1." w = 5;  
  symbol2 i = steplj c="&color2." w = 5;  
  symbol3 i = steplj c="&color3." w = 5;  
run; quit;
```

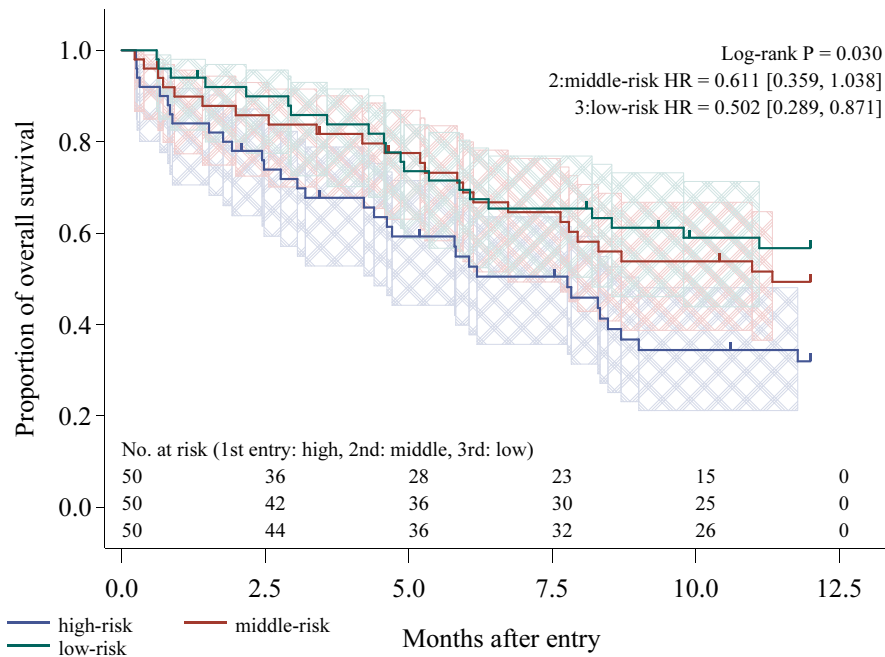


図 1: プログラム 1 の出力 (%km\_data()) マクロ

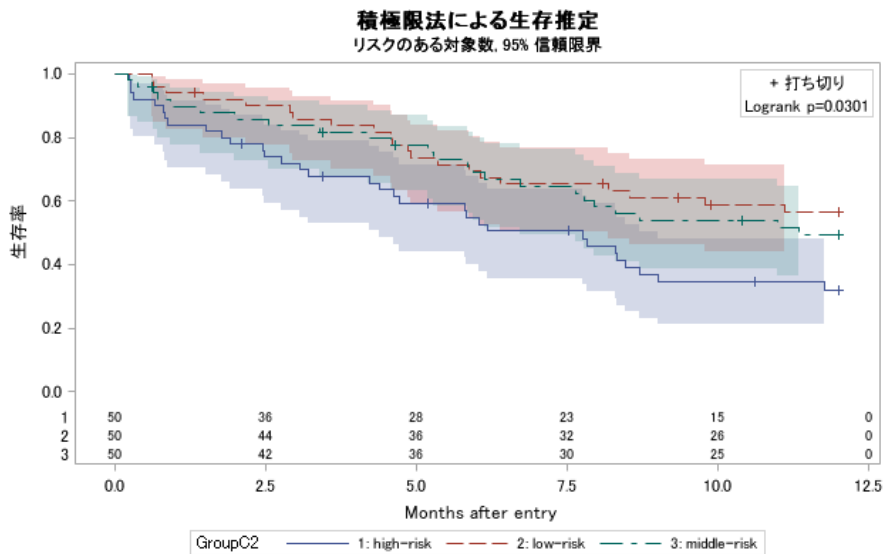


図 2: プログラム 2 の出力 (SAS 9.2 ODS Graph)

## 2 LIFETEST Procedure (SAS 9.2) との比較

### 2.1 ODS Graph

SAS 9.2 では、ODS Graph を利用した高品質なグラフが出力可能である。以下では、LIFETEST Procedure の出力と、作成したマクロの出力についてそれぞれの特徴を述べる。なお、SAS 9.1.3 では付加情報を加えることができないため、%km\_data() マクロを利用すると簡便だろう。

ODS Graph を利用したプログラム 2 を作成し、その出力を図 2 に示した。

## プログラム 2: ODS Graph によるグラフ出力

```
ods listing gpath = "&Path." style = Statistical sge = on;
ods graphics on /
  antialias = on border = off scale = on imagename = "Lifetest_ods" width = 6.33333333 in height = 4 in;
proc lifetest data = D1 plots=(survival(atrisk=(0 to 12.5 by 2.5) test c1));
  time T * Censor(1);
  strata GroupC;
run;
ods graphics off;
ods listing close; ods listing;
```

ODS Graph を用いたプログラムは、詳細な設定を行う必要が無く、非常に簡単に高品質なグラフが出力可能である事が分かる。しかし、フォントサイズが小さく、軸ラベルは固定されているため、このまま用いる事ができない場合もあるだろう。ODS LISTING statement で `sge = on` オプションを指定し `sge` 形式のグラフを出力すれば、ODS Graphics Editor でグラフを直接編集できる。だが、複数のグラフを出力する場合には不向きであり、TEMPLATE Procedure を用いて雛形を書き換える必要がある。また、SAS 9.2 TS2M0 では、ODS Graph の出力はファイル形式に抛らずラスター画像に変換されてしまう。ベクトル画像を出力したい場合、現状では ODS Graph は利用できない。

### 2.2 TEMPLATE Procedure

グラフの雛形を書き換えるためには、プログラム 3 を実行して、ログ画面に出力される雛形を編集する必要がある。雛形をコピーし、TEMPLATE Procedure で適切な部分を書き換えると、オリジナルの雛形を作成することができる。スペースの都合上全文は省略するが、プログラム 4 のような形式で実行することができる。また、プログラム 5 やプログラム 6 を実行すると、デフォルトの雛形に戻ることができる。

### プログラム 3: グラフ雛形の表示

```
proc template;
  source Stat.Lifetest.Graphics.ProductLimitSurvival;
run;
```

### プログラム 4: オリジナル雛形によるグラフ出力

```
proc template;
  define style Styles.MyStatistical;
  parent = styles.Statistical;
  style GraphFonts /
    'GraphTitleFont'=("Times New Roman",24pt, bold)
    'GraphFootnoteFont'=("Times New Roman",24pt, italic)
    'GraphLabelFont'=("Times New Roman",24pt)
    'GraphValueFont'=("Times New Roman", 24pt)
    'GraphDataFont'=("Times New Roman", 24pt)
    'GraphUnicodeFont'=(" <MTsans-serif-unicode> ", 24pt)
    'GraphAnnoFont'=("Times New Roman", 24pt);
  end;
  define statgraph
  Stat.Lifetest.Graphics.ProductLimitSurvival;
  .
  .
  .
  if (PLOTATRISK=1)
    innermargin / align=bottom;
    blockplot x=TATRISK block=ATRISK /
      repeatedvalues=true display=(values) valuealign=start
      valuefitpolicy=truncate labelposition=left
      labelattrs=GRAPHVALUETEXT valueattrs=GRAPHDATATEXT
      (size=20pt) includemissingclass=false;
    .
    .
  end;
run;
ods listing gpath = "&Path." style = MyStatistical sge = on;
ods graphics on /
  border = off scale = on
  imagename = "Lifetest_original_ods"
  height = 12 in width = 19 in;
proc lifetest data = D1 plots=(survival(atrisk=(0 to 12.5 by 2.5) test c1));
  time T * Censor(1);
  strata GroupC2;
run;
ods graphics off;
ods listing close; ods listing;
```

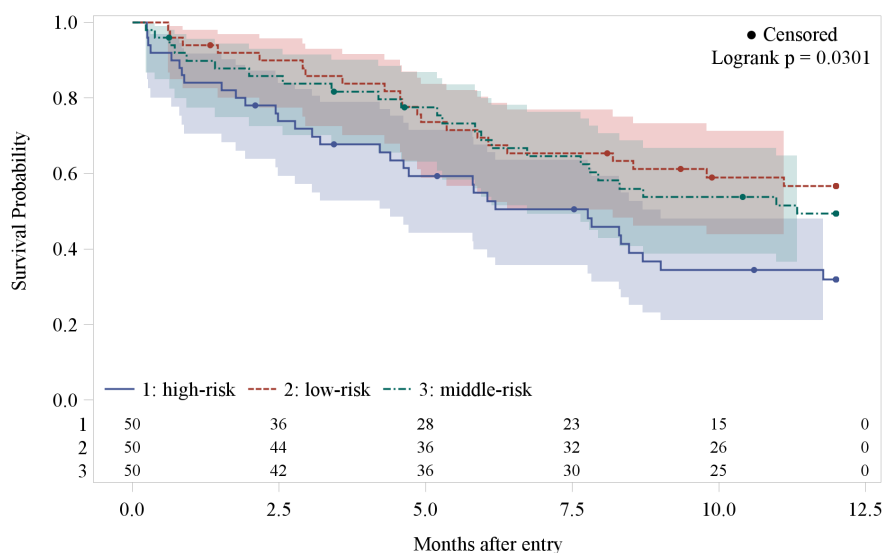


図 3: プログラム 4 の出力 (SAS 9.2 ODS Graph オリジナル雛形)

#### プログラム 5: デフォルトの雛形に戻す方法

```
proc template;
  delete Stat.Lifetest.Graphics.ProductLimitSurvival;
run;
```

#### プログラム 6: 全ての雛形を初期化する方法

```
ods path sashelp.tmplmst(read);
proc datasets library=sasuser;
  delete templat(mentype=itemstor);
run;
ods path sasuser.templat(update) sashelp.tmplmst(read);
```

### 3 考察と補足

本稿では、`%km_data()` マクロおよび ODS Graph による付加情報を追加した Kaplan-Meier プロットの比較を行った。どちらの方法でも高品質なグラフを作成することができるが、より詳細な拡張が必要な場合や、ベクトル画像が必要な場合には `%km_data()` マクロを有効に利用できると思われる。

`%km_data()` マクロの本体、図 4 のグラフを描画できるサンプルプログラムおよびスペースの都合上省略したプログラムについては Web ページ (<http://www.josai.ac.jp/~nagasima/>) 上で公開予定であり、ライセンス (GPLv3; <http://www.gnu.org/licenses/>) で認める範囲内で自由に利用できる。

### 参考文献

- [1] SAS Institute Inc. *SAS 9.2 Macro Language: Reference*. Cary, NC, USA: SAS Institute Inc., 2009.
- [2] SAS Institute Inc. *SAS/STAT(R) 9.2 User's Guide, Second Edition*. Cary, NC, USA: SAS Institute Inc., 2009.
- [3] SAS Institute Inc. *SAS/GRAPH(R) 9.2: Graph Template Language User's Guide, Second Edition*. Cary, NC, USA: SAS Institute Inc., 2008.

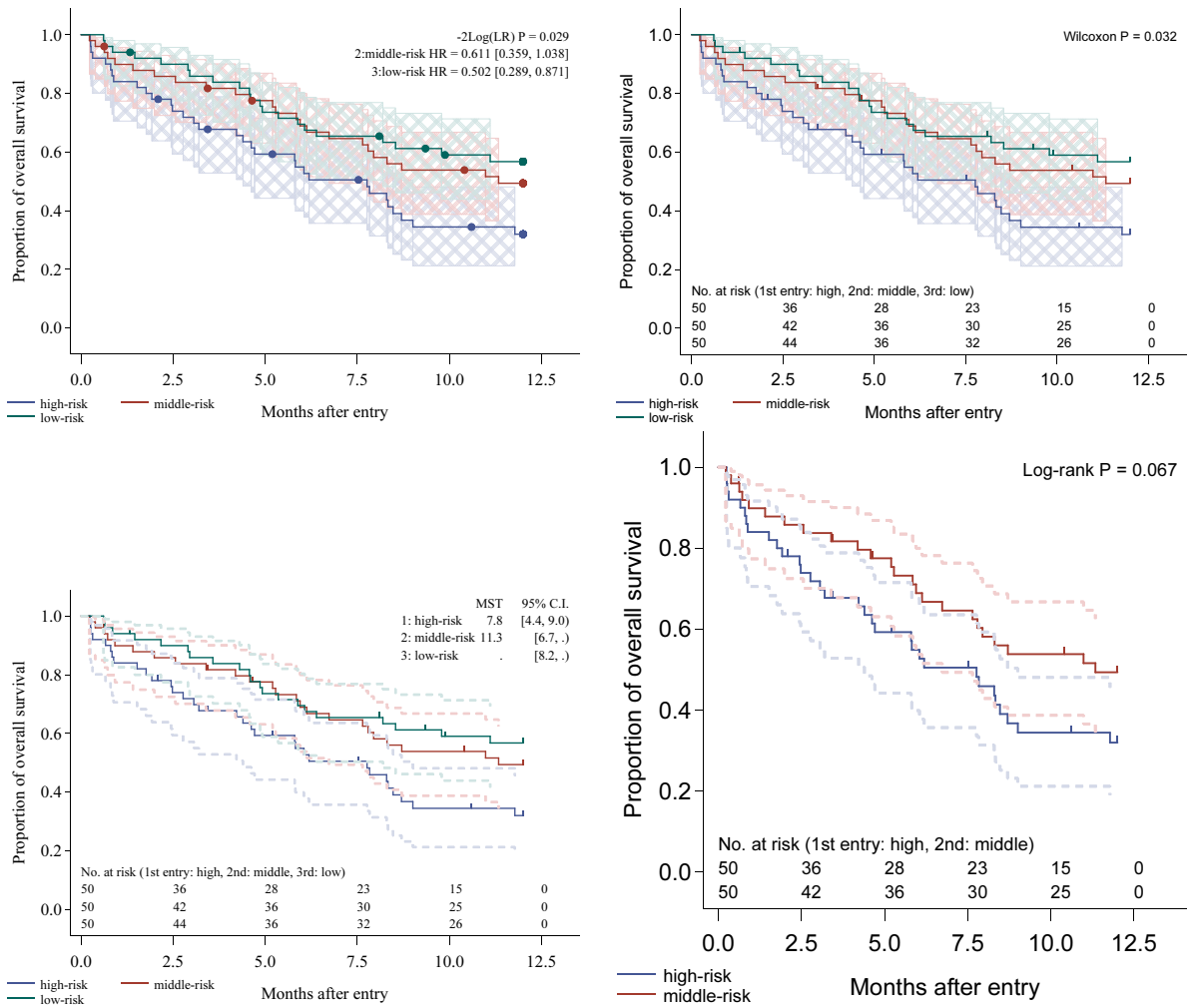


図 4: その他の実行例 (%km\_data()) マクロ

## A サンプルプログラム, 初期化プログラム, およびマクロ本体

### プログラム 7: 初期化, マクロの読み込みおよび仮想データの生成

```

option linesize = 200 pagesize = 5000 mprint;
dm 'log; clear; output; clear;';

/* setting path */
%let execpath = " ";
%let Path = " ";
%macro setexecpath;
%let execpath = %sysfunc(getoption(sysin));
%if %length(&execpath) = 0 %then
%let execpath = %sysget(sas_execfilepath);
data _null_;
do i = length("&execpath") to 1 by -1;
if substr("&execpath", i, 1) = "\" then do;
call symput("Path", substr("&execpath", 1, i));
stop;
end;
end;
run;
%mend setexecpath;
%setexecpath;
libname Out "&Path";

data D1;
length GroupC $20.;
call streaminit(32789238);
tc = 12; /* withdrawal time */
h1 = 0.100; n1 = 50; /* h1: hazard of group 1 */
h2 = 0.075; n2 = 50; /* h2: hazard of group 2 */
h3 = 0.050; n3 = 50; /* h3: hazard of group 3 */
/* E(T)=1/h */
array h[*] h1-h3; array n[*] n1-n3;
do Group = 1 to 3;
do i = 1 to n[Group];
T = rand('exponential') / h[Group]; Censor = 0;
if T > tc then do; T = tc; Censor = 1; end;

```

```

if rand('Uniform') > 0.9 then do; Censor = 1; end;
/* lost to follow-up (random) */
if Group = 1 then GroupC = '1: high-risk';
if Group = 2 then GroupC = '2: middle-risk';
if Group = 3 then GroupC = '3: low-risk';
output;
end;
end;
label T = "Months after entry";
keep T Censor Group GroupC;
run;

```

## プログラム 8: %km\_data() マクロ

```

/*****
Copyright (C) 2010, Kengo NAGASHIMA.

This program is free software: you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation, either
version 3 of the License, or any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied
warranty of MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE. See the GNU General Public License
for more details.

You should have received a copy of the GNU General Public
License along with this program.
If not, see <http://www.gnu.org/licenses/>
*****/

/*****
Macro name: %kmdata()
Author: Kengo NAGASHIMA
Version: 2.1.3
Last Updated Date: 07/09/2010
System Recommends:
  Windows XP 32 bit, SAS 9.2,
  set up SAS output font size = 12pt
System Requirements:
  Windows XP / Vista / 7, SAS 9.1.3 or later,
  Base SAS, SAS/STAT, SAS/GRAPH
*****/

/*****
Make graph datasets for extended Kaplan-Meier plots
data: a input dataset name
time: a survival time variable name
stra: a stratum variable name
censor: a censor variable name
censorv: censored variable code
out: output graph dataset name
anno: output annotate dataset name
afont: no. at risk text font
CI: 0 = none
    1 = show confidence interval
Pattern: confidence interval filling pattern
        see [Annotate Dictionary STYLE Variable]
censEXT: 0 = none
        1 = censor extension (needle)
cHeight: extended censor height
cWidth: extended censor width
Size: font size of at risk, test and hazard ratio
Step: at risk, MST & HR interval step size
atrisk: 0 = none
        1 = no. at risk
atriskorder: no. at risk order
Base: at risk base position
Label: at risk label
Test: 0 = none
        1 = show homogeneity test
TestX: = x position (% of data area)
TestY: = y position (% of data area)
Type: logrank = log-rank test
      wilcoxon = generalized wilcoxon test
      likelihoodratio = likelihood ratio test
HR: 0 = none
    1 = show hazard ratio
HRX: x position (% of data area)
HRY: y position (% of data area)
MST: 0 = none
    1 = show median survival time
MlabX: x position of strata label (% of data area)
MmedX: x position of median (% of data area)
MciX: x position of confidence interval (% of data area)
MSTY: y position (% of data area)
*****/
%macro km_data(
  data, time, stra, censor, censorv,
  out = graph, anno = anno,

  CI = 0, Pattern = x4,

  censEXT = 0, cHeight = 0.025, cWidth = 5,

  Size = 3, Step = 4,
  afont = "'Times New Roman'",

```



```

atrisk = 0, atriskorder = 0 to 0,
Base = 0, Label = "No. at risk",

Test = 0, TestX = 90, TestY = 100, Type = logrank,
HR = 0, HRX = 90, HRY = 95,
MST = 0, MlabX = 60, MmedX = 80, MciX = 98, MSTY = 95
);
%if "&stra." = "" %then %do;
%let stra = dstra;
data &data.;
  set &data.; dstra = 1;
%end;
ods exclude all;
proc lifetest data = &data. outsurv = &out.;
  time &time. * &censor.(&censrv.);
  strata &stra.;
  ods output
    HomTests = Test CensoredSummary = CS
    ProductLimitEstimates = PLE Quartiles = QEST;
%if "&sysver." = "9.2" %then proc phreg;
%else %if "&sysver." = "9.1" %then proc tphreg;
data = &data.;
  class &stra. / param = ref ref = first;
  model &time. * &censor.(&censrv.) = &stra. / rl;
  ods output ParameterEstimates = HazardRatio;
run;
ods select all;
proc sort data = &out.; by &stra.;
data &out.;
  set &out.; retain CSDF 0;
  if Survival = . then Survival = CSDF;
  else CSDF = Survival;
  drop &stra.;
data &out.;
  set
    &out.(where=(CENSOR_ ^= 1) drop=CSDF)
    &out.(where=(CENSOR_ = 1));
  drop CENSOR_;
proc sort data = &out.; by &time.;
data _null_;
  set CS(where=(control_var ^= "-"));
  call symput("StraN", cat(_N_));
data &out.;
merge
%do i = 1 %to &StraN.;
  &out.(where=(STRATUM=&i.)
    rename=(Survival=SV&i. SDF_LCL=SL&i. SDF_UCL=SU&i.)
    drop=CSDF)
  &out.(where=(STRATUM=&i. & CSDF&i.^=.)
    rename=(CSDF=CSDF&i. SDF_LCL=SL&i. SDF_UCL=SU&i.)
    drop=Survival)
%end; ;
by &time.; drop STRATUM;
proc sort data = &out.; by &time.;
run;

/* Censor extension */
%if "&censEXT." = "1" %then
  %km_censEXT(&out., &time., &StraN., &cHeight.,
    &cWidth., cens);
/* No. at risk */
%if "&atrisk." = "1" %then
  %km_atrisk(PLE, &stra., &time., &StraN.,
    &atriskorder., &Base., &Step., &Size., &label.,
    &afont., atrisk);
/* Homogeneity test */
%if "&Test." = "1" %then
  %km_hontest(Test, &TestX., &TestY., &Type.,
    &Size., &afont., HTEST);
/* Hazard ratio */
%if "&HR." = "1" %then
  %km_hr(HazardRatio, &HRX., &HRY., &StraN., &Step.,
    &Size., &afont., HR);
/* Confidence interval */
%if "&CI." = "1" %then
  %km_ci(&out., &time., &StraN., &Pattern., CI);
/* Quantile estimate */
%if "&MST." = "1" %then
  %km_qest(QEST, &stra., &MlabX., &MmedX., &MciX.,
    &MSTY., &StraN., &Step., &Size., &afont., QE);

%if %eval(&censEXT.+&atrisk.+&test.+&hr.+&ci.)
  ^= 0 %then %do;
data &anno.;
  length when $1. style $30.;
  set
    %if &CI. = 1 %then CI;
    %if &censEXT. = 1 %then cens;
    %if &atrisk. = 1 %then atrisk;
    %if &test. = 1 %then HTEST;
    %if &hr. = 1 %then HR;
    %if &MST. = 1 %then QE;
  ;
  retain when 'B';
run;
%end;

proc datasets lib = work;
  delete CS PLE QEST HazardRatio Test
  %if &CI. = 1 %then CI;

```

```

%if &censEXT. = 1 %then cens;
%if &atrisk. = 1 %then atrisk;
%if &test. = 1 %then HTEST;
%if &hr. = 1 %then HR;
%if &MST. = 1 %then QE;
;
run; quit;
%mend km_data;

/*****
/* censor extension */
%macro km_censEXT(
  data, time, StraN, cHeight, cWidth, out
);
data _null_;
  set &data.;
  %if "&StraN." = "1" %then
    call symput("exht", &cHeight. * (1 - CSDF1) );
  %else
    call symput("exht", &cHeight. * (1 -
      min(of CSDF1 - CSDF&StraN.)) );;
data &out.;
  length color function $10.;
  retain xsys ysys '2' size &cWidth.; set &data.;
  %do i = 1 %to &StraN.;
    if CSDF&i. ^= . then do;
      color = "&&color&i.";
      function='move'; x=&time.; y=CSDF&i.; output;
      function='draw'; x=&time.; y=CSDF&i. + &exht.; output;
    end;
  %end;
  keep xsys ysys size color function x y;
run;
%mend km_censEXT;

/*****
/* No. at risk */
%macro km_atrisk(
  data, stra, time, StraN, order,
  base, step, size, label, afont, out
);
data &out.;
  set &data.; by STRATUM &time.;
  do x = &order.;
    if &time. = 0 & Censor = . then do; leave; end;
    if &time. < x then do; leave; end;
  end;
data &out.;
  length color function $10.; set &out.; by STRATUM x;
  retain xsys '2' ysys '1' color 'black' position '3'
  function 'label' size &size.;
  %do i = &StraN. %to 1 %by -1;
    if STRATUM = &i. then y =
      (&base. + &step. * (&StraN. - 1)) - &step. * (&i. - 1);
  %end;
  text = cat(Left);
  if last.x then output;
data dummy;
  length color function $10.;
  retain xsys '2' ysys '1' color 'black' position '3'
  function 'label' size &size.;
  do STRATUM = 1 to &StraN.;
  do x = &order.;
    %do i = &StraN. %to 1 %by -1;
      if STRATUM = &i. then y =
        (&base. + &step. * (&StraN. - 1)) - &step. * (&i. - 1);
    %end;
    output;
  end; end;
data &out.;
  merge dummy &out.; by STRATUM x;
  format xtext $20.; retain xtext '';
  if text ^= "" then xtext = text;
  else text = xtext;
data Label;
  length color function $10. text $100.;
  retain xsys '2' ysys '1' color 'black' position '3'
  function 'label' size &size.;
  x = 0; y = &base. + &step. * &StraN.; text = &label.; output;
data &out.; set &out. Label; retain style &afont.;
  keep xsys ysys color position function size x y text style;
proc datasets lib = work; delete dummy Label;
run;
%mend km_atrisk;

/*****
/* homogeneity test */
%macro km_homtest(
  Test, TestX, TestY, Type, Size, afont, out
);
data &out.;
  %if "&Type" = "logrank" %then set &Test.(firstobs=1 obs=1);
  %else %if "&Type" = "wilcoxon" %then
    set &Test.(firstobs=2 obs=2);
  %else %if "&Type" = "likelihoodratio" %then
    set &Test.(firstobs=3 obs=3);;

```

```

retain xsys '1' ysys '1' color 'black' position '7'
function 'label' size &size. style &afont.;
x = &TestX.; y = &TestY.;
%if "&Type" = "logrank" %then
text = cat("Log-rank P = ", put(ProbChiSq, PVALUE5.4));
%else %if "&Type" = "wilcoxon" %then
text = cat("Wilcoxon P = ", put(ProbChiSq, PVALUE5.4));
%else %if "&Type" = "likelihoodratio" %then
text = cat("-2Log(LR) P = ", put(ProbChiSq, PVALUE5.4));;
keep xsys ysys color position function size style x y text;
run;
%mend km_homtest;

/*****
/* hazard ratio */
%macro km_hr(
HazardRatio, HRX, HRY, StrAN, Step, Size, afont, out
);
data &out.;
set &HazardRatio.;
length text $200.;
retain xsys '1' ysys '1' color 'black' position '7' function 'label'
size &size. style &afont.;
x = &HRX.;
y = &HRY. - &Step. * (&n_ - 1);
text = cat(
compress(ClassVal0), " HR = ",
compress(put(HazardRatio, 8.3)), " [",
compress(put(HRLowerCL, 8.3)), " ",
compress(put(HRUpperCL, 8.3)), "]"
);
keep xsys ysys color position function size style x y text;
run;
%mend km_hr;

/*****
/* confidence interval */
%macro km_ci(
ConfidenceInterval, time, StrAN, pattern, out
);
%do i = 1 %to &StrAN.;
data &out.&i.;
length color function $10.;
retain xsys ysys '2' style "&pattern." line 0 tmp .;
set &ConfidenceInterval.(firstobs=2);
where SL&i. ^= .;
color = "&scolor&i.";
function = 'bar'; x = &time.; y = tmp; output;
tmp = SU&i.;
function = 'move'; x = &time.; y = SL&i.; output;
keep xsys ysys style line color function x y;
%end;
data &out.;
set
%do i = 1 %to &StrAN.;
&out.&i.(firstobs=2)
%end;;
%do i = 1 %to &StrAN.;
proc datasets lib = work;
delete &out.&i.;
%end;;
run;
%mend km_ci;

/*****
/* Quantile estimate + 95% CI */
%macro km_qest(
Est, stra, MlabX, MmedX, MciX, MSTY, StrAN, Step,
Size, afont, out
);
data &out.;
length text $200.;
set &Est.;
retain xsys '1' ysys '1' color 'black' function 'label'
size &size. style &afont.;
%do i = 1 %to &StrAN.;
if (STRATUM = &i. & Percent = 50) then do;
position = '4';
%if "&i." = "1" %then %do;
x = &MmedX.; y = &MSTY.; text = "MST"; output;
x = &MciX.; y = &MSTY.; text = "95% C.I."; output;
%end;
y = &MSTY. - &Step. * &i.;
x = &MmedX.; text = put(Estimate, 6.1); output;
x = &MciX.;
text = cat("[" , compress(put(LowerLimit, 6.1)),
" , ", compress(put(UpperLimit, 6.1)), " )"); output;
position = '6';
x = &MlabX.; text = trim(&stra.); output;
%end;
%end;
keep xsys ysys color position function size style x y text;
run;
%mend km_qest;

```