Self-Study Material 2: Stata for Windows

Entering data into Stata

1) Enter data directly into Stata.
   Open the **Data Editor** window by clicking on the spreadsheet icon on the toolbar. Columns corresponds to variables, rows correspond to observations. Enter the data directly in the spreadsheet then click column names Var1, Var2 to change the variable names. Then close the data editor.

2) Open Stata format data file
   Click the **open(use)** icon on the toolbar or use command "**use** filename"

3) Import data from text (ASCII) files.
   To open an ASCII (or text) file in Stata, go into the **File** pull-down menu and select **Import**. From the submenu, you can choose to import data created by a spreadsheet, in fixed format, in fixed format with a dictionary, or unformatted. Or use "**infile var1 var2 using filename**" or "**insheet using filename**" for data created by a spreadsheet.

Basic commands

Note: Stata is case-sensitive: small letter not capital letters for commands. Up cases can be used for variable names.

log: to create a log file to store all the used commands and output excluding graphs
cmdlog: to make a record of what you type during your Stata session
#review: to display the last few lines typed at the terminal. For example, #review by itself lists the last 5 lines and "#review 10" would list the last 10 lines.
describe: to describe a dataset
list: to list the contents of a dataset
codebook: to list detailed contents of a dataset
summarize: to calculate and displays a variety of univariate summary statistics
generate: to generate new variable
replace: to change the contents of an existing variable

Save files

To save data set click **Save** icon on the tool bar
To save log file, use "log" or click **Begin Log** icon on the toolbar
To save graphs, after you create a graph, go to the **File** menu and click **Save Graph**

One can copy selected Stata output and Stata graphs either from **Edit** menu or using Ctrl+c into Word file. For wide table use **Curier New** font with size 8, 9 or 10 so that the table columns will align properly.

Write a do-file

To include a series of commands so that you can repeat the analysis later without typing commands again.

Stata help

Help menu is very useful. You can search contents, State command. Or search internet.

Survival Analysis

Data description: The first data we're going to use in this lab is **myelomatosis data** (Peto et al. 1977), which includes 25 patients diagnosed with myelomatosis. These patients are randomly assigned to two treatments.

DUR: contains the time in days from the point of randomization to either death or censoring.
STATUS: coded 1 if uncensored, 0 if censored.
TREAT: indicates the treatment patients are randomly assigned to.
RENAL: renal functioning at the time of randomization, coded 1 if normal, 0 if impaired.

1). Input data into STATA.

If data file is already in STATA format, you can double click on it to open. The file you download from course website, myel.dta, is the myelomatosis data in STATA format.

2). Declare data to be survival-time data

In order to do survival analysis in STATA, we first need to let STATA know that the data we are dealing with is survival-time data. To declare data to be survival-time data, we use the following command

```
stset dur, failure(status==1)

     failure event:  status == 1
```

```
obs. time interval:  (0, dur]
 exit on or before:  failure

------------------------------------------------------------------------------
       25  total obs.
        0  exclusions
------------------------------------------------------------------------------
       25  obs. remaining, representing
       17  failures in single record/single failure data
    15334  total analysis time at risk, at risk from t =         0
                             earliest observed entry t =         0
                                  last observed exit t =      2240
```

When the analysis is done and if you want to go back to normal mode, we can use command
`stset, clear`
to make STATA forget that the data are survival-time data.

3). Describe survival-time data

stdes presents a brief description of the survival-time data. It doesn't do any analysis, but only summarizes the information.

Here is the description for myel.dta

```
. stdes

         failure _d:  status == 1
   analysis time _t:  dur

                                   |-------------- per subject --------------|
Category                   total        mean         min     median        max
------------------------------------------------------------------------------
no. of subjects               25
no. of records                25           1           1          1          1

(first) entry time                         0           0          0          0
(final) exit time                     613.36           8        210       2240

subjects with gap              0
time on gap if gap             0
time at risk               15334      613.36           8        210       2240

failures                      17         .68           0          1          1
------------------------------------------------------------------------------
```

Here we set dur to be the time variable and use status to indicate censor (=0) or not (=1). The output of `stdes` tells us that there are 25 objects at total. We have single record for each object. The table listed mean/minimum/median/maximum for the exit time and time at risk for each object. There are 8 objects censored.

4). Summarize survival-time data

stsum presents summary statistics -- time at risk, incidence rate, number of subjects, and the 25th, 50th, and 75th percentiles of survival time.

```
. stsum

         failure _d:  status == 1
   analysis time _t:  dur

         |               incidence       no. of    |------ Survival time -----|
         | time at risk     rate        subjects        25%       50%       75%
---------+---------------------------------------------------------------------
   total |        15334   .0011086            25         63       210         .
```

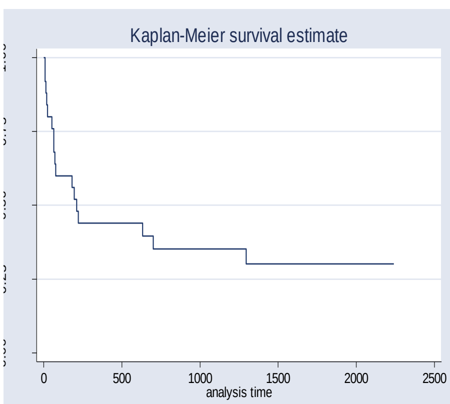
5). Generate, graph, and list the survivor and cumulative hazard functions

sts graph graphs the estimated survivor or Nelson-Aalen cumulative hazard function.

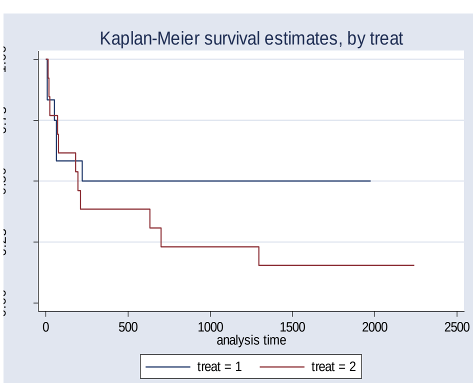
sts list lists the estimated survivor and related functions.

sts generate creates new variables containing the estimated survivor function, the Nelson-Aalen cumulative hazard function, or related functions.

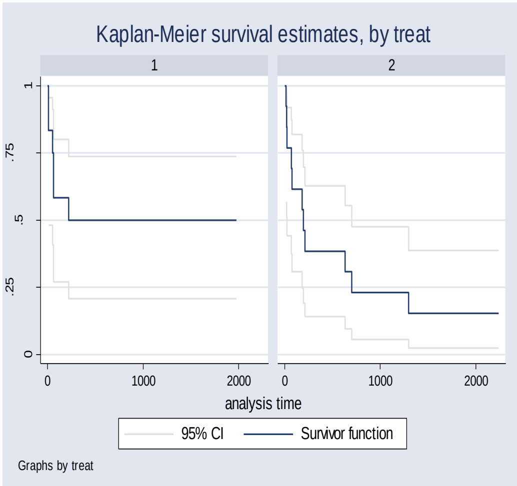
```
sts graph /*graphs the estimated survivor function*/
```



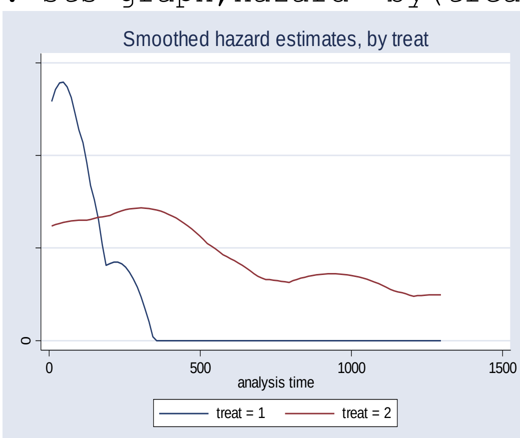

```
. sts graph, by(treat)
```



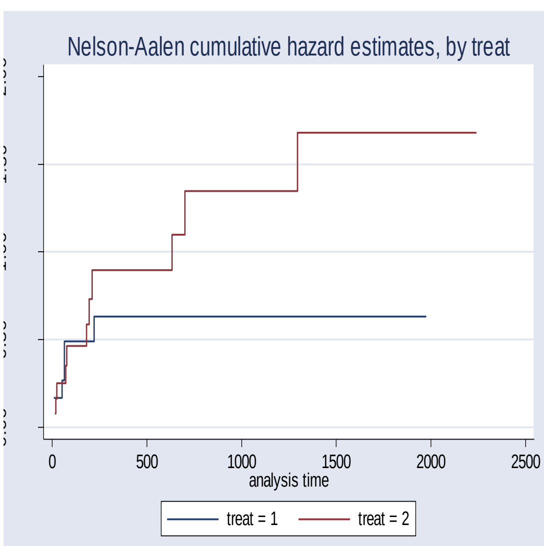
```
. sts graph,  by(treat) gwood
```




```
. sts graph,hazard  by(treat)
```




```
. sts graph,na by(treat)
```



```
. sts list, by(treat)
```
*K-M survival functions

```
         failure _d:  status
   analysis time _t:  dur
```

Time	Beg. Total	Fail	Net Lost	Survivor Function	Std. Error	[95% Conf.	Int.]
treat=1							
8	12	2	0	0.8333	0.1076	0.4817	0.9555
52	10	1	0	0.7500	0.1250	0.4084	0.9117
63	9	2	0	0.5833	0.1423	0.2701	0.8009
220	7	1	0	0.5000	0.1443	0.2085	0.7361
365	6	0	1	0.5000	0.1443	0.2085	0.7361
852	5	0	1	0.5000	0.1443	0.2085	0.7361
1296	4	0	1	0.5000	0.1443	0.2085	0.7361
1328	3	0	1	0.5000	0.1443	0.2085	0.7361
1460	2	0	1	0.5000	0.1443	0.2085	0.7361
1976	1	0	1	0.5000	0.1443	0.2085	0.7361
treat=2							
13	13	1	0	0.9231	0.0739	0.5664	0.9888
18	12	1	0	0.8462	0.1001	0.5122	0.9591
23	11	1	0	0.7692	0.1169	0.4421	0.9191
70	10	1	0	0.6923	0.1280	0.3734	0.8718
76	9	1	0	0.6154	0.1349	0.3083	0.8184
180	8	1	0	0.5385	0.1383	0.2477	0.7599
195	7	1	0	0.4615	0.1383	0.1916	0.6964
210	6	1	0	0.3846	0.1349	0.1405	0.6280
632	5	1	0	0.3077	0.1280	0.0950	0.5543
700	4	1	0	0.2308	0.1169	0.0558	0.4746
1296	3	1	0	0.1538	0.1001	0.0248	0.3878
1990	2	0	1	0.1538	0.1001	0.0248	0.3878
2240	1	0	1	0.1538	0.1001	0.0248	0.3878

```
. sts list, by(treat) compare
```
*lists the estimated survivor and related functions.

```
         failure _d:  status == 1
   analysis time _t:  dur
```

treat		Survivor Function 1	2
time	8	0.8333	1.0000
	287	0.5000	0.3846
	566	0.5000	0.3846
	845	0.5000	0.2308
	1124	0.5000	0.2308
	1403	0.5000	0.1538
	1682	0.5000	0.1538
	1961	0.5000	0.1538
	2240	.	0.1538

`sts generate surv=s, by(treat)`

creates new variables containing the Kaplan-Meier survivor function. Surv is the name of the new generated variable, which contains Kaplan-Meier survivor function.

```
. sts gen surv=s, by(treat)

. list surv
```

```
          surv
  1.  .83333333
  2.  .53846154
  3.  .30769231
  4.         .5
  5.        .75
  6.  .15384615
  7.         .5
  8.  .58333333
  9.  .46153846
 10.  .61538462
 11.  .69230769
 12.  .83333333
 13.  .92307692
 14.  .15384615
 15.         .5
 16.  .84615385
 17.  .23076923
 18.         .5
 19.         .5
 20.  .38461538
 21.  .58333333
 22.         .5
 23.  .15384615
 24.         .5
 25.  .76923077
```

You can also generate hazard function, Greenwood pointwise standard error, or confidence interval of survival function, …… by

```
sts generate haz=h, by(treat)
sts generate gse=se, by(treat)
sts generate lowerbound=lb, by(treat)
sts generate upperbound=ub, by(treat)
```

Then plot them using graph command

6. Two Sample Testing in Stata

sts test -- Test equality of survivor functions

sts test *varlist* [*if*] [*in*] [**,** *options*]

options description

logrank perform log-rank test of equality; the default

cox perform Cox test of equality

wilcoxon perform Wilcoxon-Breslow-Gehan test of equality (The weights w=n the number of subjects at risk at each interval.)

tware perform Tarone-Ware test of equality (This test is the same as the Wilcoxon test, with the exception that the weight function w= $n^{1/2}$)

peto perform Peto-Peto-Prentice test of equality (The only difference between the Wilcoxon test and this one is that the weight function is approximately equal to the K-M survival Function)

fh(*p q***)** perform generalized Fleming-Harrington test of equality

trend test trend of the survivor function across three or more ordered groups

strata(*varlist*) perform stratified test on *varlist*, displaying overall test results

detail display individual test results; modifies strata()

`. sts test treat` tests the equality of the survivor function across two different treatment groups.

```
. sts test treat

         failure _d:  status == 1
   analysis time _t:  dur


Log-rank test for equality of survivor functions

      |   Events         Events
treat |  observed       expected
------+-------------------------
1     |         6           8.34
2     |        11           8.66
------+-------------------------
Total |        17          17.00

            chi2(1) =       1.31
            Pr>chi2 =     0.2519
sts test treat, wilcoxon

         failure _d:  status == 1
   analysis time _t:  dur


Wilcoxon (Breslow) test for equality of survivor functions

      |   Events         Events        Sum of
treat |  observed       expected        ranks
------+--------------------------------------
1     |         6           8.34          -18
2     |        11           8.66           18
------+--------------------------------------
Total |        17          17.00            0

            chi2(1) =       0.25
            Pr>chi2 =     0.6178
```

P-value of less than 0.05 is an evidence that there is significant difference between two survival functions.