Biostatistics III: Survival analysis for epidemiologists
Computing notes for SAS users

Paul W. Dickman

November 2008

http://www.pauldickman.com/survival/sas/

Contents

1 Notes on survival analysis using SAS

These notes describe how some of the methods described in the course can be implemented in SAS. Data sets in SAS format and SAS code for reproducing some of the exercises are available on the web `http://www.pauldickman.com/survival/sas/`. Note that there is a separate handout (`relative_survival_using_sas.pdf`) describing how to estimate and model relative survival using SAS.

1.1 Kaplan-Meier, life table, and log-rank test using PROC LIFETEST

The LIFETEST procedure can compute nonparametric estimates of the survivor function (using either the actuarial or Kaplan-Meier method) and test the equality of survival distributions across strata. The following code produces Kaplan-Meier estimates (and a corresponding plot) of the observed survival rates for the sample of 35 patients with colon carcinoma. That is, it reproduces the estimates shown in the lecture notes. The code is available in `example_lifetest.sas`.

```
proc lifetest data=biostat3.example plots=(s);
time surv_mm*status(0,4);
run;
```

- Note that we must specify the variable containing survival time, `surv_mm`, the status variable, `status`, and the codes in the status variable which indicate censored survival times (0=alive and 4=lost to follow-up).
- The `plots=(s)` option requests plots of the survivor function.
- By default, a symbol is plotted to indicate censored observations. This can be prevented by specifying `censoredsymbol=none`.

The following code produces the corresponding actuarial estimates of the all-cause survivor function for the sample of 35 patients diagnosed with colon carcinoma. The `width=12` option specifies annual intervals for the life table.

```
proc lifetest data=biostat3.example plots=(s)
     nocens graphics method=act width=12;
time surv_mm*status(0,4);
run;
```

Stratified estimates can be made using the `STRATA` statement. The following code estimates the survivor function separately for males and females and calculates several tests for differences in survival between the 2 groups (including the log-rank test). A subsetting `WHERE` statement is used to restrict the analysis to patients diagnosed during 1985 and later (note that '`ge`' is SAS notation for 'greater than or equal to ($\geq$)').

```
proc lifetest data=biostat3.example plots=(s);
time surv_mm*status(0,4);
where yydx ge 85;
strata sex;
run;
```

The complete SAS code for these analyses is in the file `example_lifetest.sas`.

1.2 High-resolution graphics options

The quality of the graphics output can be enhanced by resetting the values of some SAS graphics options (`goptions`). For example,

```
goptions noprompt gunit=percent rotate=landscape
device=win ftext="Arial" htext=3 htitle=4;

symbol1 c=black v=none line=1;  /*solid line*/
symbol2 c=black v=none line=20; /*dashed line*/
```

1.3 Tabulating mortality rates

In exercise 6 we tabulated CHD rates for each of 3 categories of energy intake. To repeat this in SAS we use the following procedure:

1. Use PROC SUMMARY to calculate the number of events and person-time at risk in each exposure group and save this to a SAS data set (I've used a format to define the grouping);
2. In DATA step, calculate the rate (events/person-time) and the corresponding CI;
3. Use PROC PRINT to print the results.

```
proc format;
value energy
low-2500='<2500'
2500-3000='2500-3000'
3000-high='3000+';
run;

proc summary data=biostat3.diet nway;
var chd y;
class energy;
output out=rates(drop=_type_ _freq_) sum=chd y;
format energy energy.;
run;

data rates;
set rates;
_rate=1000*(chd/y);
ci_low=_rate/exp(1.96*sqrt(1/chd));
ci_high=_rate*exp(1.96*sqrt(1/chd));
run;

proc print noobs;
title 'Table of cases, person-years, and rates per 1000 person-years';
var energy chd y _rate ci_low ci_high;
```

```
format y 8.1 _rate ci_low ci_high 7.2;
run;
```

The code is available in the file `diet_rates_poisson_regression.sas`.

The estimates are the same as those we obtained using Stata.

```
Table of cases, person-years, and rates per 1000 person-years

ENERGY       CHD        Y   _RATE  CI_LOW   CI_HIGH

<2500         16     946.6   16.90   10.35     27.59
2500-3000     22    2017.3   10.91    7.18     16.56
3000+          8    1639.8    4.88    2.44      9.76
```

```
. strate eng3, per(1000)

Estimated rates (per 1000) and lower/upper bounds of 95% confidence intervals
(337 records included in the analysis)

  +-------------------------------------------------+
  | eng3    D        Y      Rate     Lower     Upper |
  |-------------------------------------------------|
  | 1500   16   0.9466   16.9020   10.3547   27.5892 |
  | 2500   22   2.0173   10.9059    7.1810   16.5629 |
  | 3000    8   1.6398    4.8787    2.4398    9.7555 |
  +-------------------------------------------------+
```

1.4 Poisson regression in SAS

The best way to estimate Poisson regression models in SAS is using PROC GENMOD (a procedure for fitting generalised linear models). PROC GENMOD, however, does not report the rate ratio directly, only the estimated beta parameters (log rate ratios). We need to save the parameter estimates as a SAS data set, exponentiate them in a data step to obtain estimated rate ratios and then print them using PROC PRINT.

Most Stata commands for fitting multiplicative models (including the `glm` command that fits generalised linear models) contain an option where the user can specify whether the estimates should be presented on the original scale or the exponential scale (i.e., as relative risks). An additional annoyance with SAS is that we first need to calculate, in a data step, a variable containing the natural logarithm of person-time which is specified as an offset.

Continuing from the example in the previous section, following is the SAS code for fitting a Poisson regression model to estimate the effect of energy intake (in 3 categories). Such a model was fitted using Stata in exercise 6.

```
data biostat3.diet2;
set biostat3.diet;
ln_y=log(y);
label ln_y='Natural log of person-time';
run;

proc genmod data=biostat3.diet2;
title1 'Poisson regression model to estimate the effect of high energy intake';
class energy;
model chd = energy / error=poisson link=log offset=ln_y type3;
make 'ParameterEstimates' out=parmest;
format energy energy.;
run;

data parmest;
set parmest;
irr=exp(estimate);
low_irr=exp(estimate-1.96*stderr);
hi_irr=exp(estimate+1.96*stderr);
run;

proc print data=parmest label noobs;
title2 'Estimated rate ratios and 95% CIs';
var parameter level1 estimate stderr irr low_irr hi_irr;
format estimate stderr irr low_irr hi_irr 7.4;
run;
```

The code is available in the file `diet_rates_poisson_regression.sas`.

It is also possible to estimate Poisson regression models in the framework of parametric survival models using PROC LIFEREG. We still, however, have to save the parameter estimates to a date set and exponentiate them in a data step in order to obtain the rate ratio estimates.

1.5 Cox regression using PROC PHREG

The Cox proportional hazards model is estimated in SAS using the PHREG procedure. An annoyance with PROC PHREG (prior to version 9) is that it does not contain a CLASS statement. As such, dummy variables must be created in a data step in order to model categorical variables.

The following SAS code (available in `melanoma_phreg.sas`.) was used to fit a proportional hazards model to the localised melanoma data.

```
proc phreg data=melanoma(where=(stage=1));
model surv_mm*status(0,2,4) = sex age_gr2
      age_gr3 age_gr4 year8594 / risklimits;
Age: Test age_gr2=age_gr3=age_gr4=0;
run;
```

- Note the use of the `where` data step option to restrict the analysis to localised cases.
- Note the similarity with PROC LIFETEST in the method of defining the outcome variable and censoring indicator.
- The `model` statement has been split over two lines in the above example in order to fit the code on the page.
- The `risklimits` option requests confidence limits for the estimated hazard ratios (i.e. $\exp(\hat{\beta})$).
- The last line before the `run` statement specifies a Wald test that all age parameters are equal to zero. The parameter for the first age group is zero since it is the reference group. This provides a test of an overall association due to age. Under the null hypothesis, H_0: all age parameters are equal to zero, the test statistic has a χ^2_3 distribution.
- A corresponding likelihood ratio test can be performed by fitting a second model, without the three age variables, and then calculating (by hand) twice the difference in the log likelihoods of the two models.

1.5.1 Time-varying covariates in PROC PHREG

Following is an example of how we allow the effect of age at diagnosis to depend on time since diagnosis. We estimate two sets of hazard ratios for age, one for the interval up to 2 years following diagnosis and one set for the interval 2 years or more subsequent to diagnosis. The code is available in `melanoma_phreg.sas`.

```
proc phreg data=melanoma(where=(stage=1));
model surv_yy*status(0,2,4) = sex age_gr2-age_gr4 t_age2-t_age4
         year8594 t_yr8594 / risklimits;
t_yr8594=0; t_age2=0; t_age3=0; t_age4=0;
if surv_yy ge 2 then do;
t_yr8594=year8594; t_age2=age_gr2; t_age3=age_gr3; t_age4=age_gr4; end;
Age: Test age_gr2=age_gr3=age_gr4=0;
t_by_age: Test t_age2=t_age3=t_age4=0;
run;
```

1.6 'Splitting' person-time using the LEXIS macro

SAS does not contain a command analogous to `stsplit` in Stata for 'splitting' person-time. A large number of user-written SAS macros exist for performing this task, many of them described in the epidemiology literature [1, 2, 3, 4]. I use the LEXIS macro written by Bendix Carstensen and available from his web site [5]. The syntax of the macro is very similar to the syntax of the Stata `stsplit` command.

Following is an example of how the macro is used to split the diet data by attained age.

```
%lexis(
origin = dob,
entry = doe,
exit = dox,
fail = chd,
scale = 365.25,
data = biostat3.diet,
out = biostat3.diet_split,
breaks = %str(40 to 70 by 5),
other = %str(format ageband ageband. hieng hieng.;),
left = ageband
) ;
```

Note that a new data set (`biostat3.diet_split`) is created. The variable `ageband` (which contains the left cutpoint) is added to the dataset. We then fit a Poisson regression model (with `ageband` as an explanatory variable) using PROC GENMOD (see section 1.4). The SAS code above was extracted from `diet_splitting_by_age.sas` which contains a fully worked example of splitting the diet data by attained age, estimating CHD incidence rates for each classification of attained age, and estimating incidence rate ratios using Poisson regression.

1.6.1 Splitting on more than one timescale

To split on more than one timescale we simply apply the lexis macro multiple times. For example, if we wanted to study both attained age and time since entry in the diet data we would split the data in `biostat3.diet_split` by time since entry.

1.7 Cox regression with late entry and a time-varying exposure

The code in `brv_phreg.sas` uses PROC PHREG to fit Cox proportional hazards models to the bereavement data (reproducing the analyses in exercise 13). The code illustrates how to model data with late entry (using age as the timescale) and how to model a time-varying exposure (bereavement). To model the time-varying exposure we split the data in an identical manner to Stata. It is also possible to create time-varying covariates using data step statements within the PHREG procedure (see section 1.5.1).

```
/* split the data by bereavement status */
data new;
set biostat3.brv;
if (dosp < dox) then do;
brv=1; doe_tmp=doe; doe=dosp; output;
brv=0; doe=doe_tmp; dox=dosp; fail=0; output;
end;
else do; brv=0; output; end;
format sex sex.;
run;

/* Create variables for age at entry and age at exit */
/* (we will use age as the timescale) */
data new;
set new;
agee=doe-dob;
agex=dox-dob;
brv_m=brv*(sex=1);
brv_f=brv*(sex=2);
run;

/* Estimate the crude effect of brv */
proc phreg data=new;
model agex*fail(0)=brv / entry=agee risklimits;
run;

/* Estimate the effect of brv separately for each gender */
proc phreg data=new;
model agex*fail(0)=sex brv_m brv_f / entry=agee risklimits;
run;

/* Stratified Cox model (separate baseline for each gender) */
proc phreg data=new;
model agex*fail(0)=brv / entry=agee risklimits;
strata sex;
run;
```

References

[1] Wood J, Richardson D, Wing S. A simple program to create exact person-time data in cohort analyses. *Int J Epidemiol* 1997;**26**:395–9.

[2] Sun J, Shibata E, Kamijima M, Toida M, Takeuchi Y. An efficient SAS program for exact stratification of person-years. *Comput Biol Med* 1997;**27**:49–53.

[3] Yaari S, Goldbourt U. A SAS program for evaluating person-years of risk in cohort studies. *Comput Biol Med* 1989;**19**:353–9.

[4] Pearce N, Checkoway H. A simple computer program for generating person-time data in cohort studies involving time-related factors. *American Journal of Epidemiology* 1987; **125**:1085–91.

[5] Carstensen B. Lexis macro for splitting person-time in sas, 2004. `http://www.biostat.ku.dk/~bxc/Lexis/`.