**5. Estimation: (i) continuous time models**

1. **Aim**

The aim of this lesson is to illustrate how to use Stata to estimate multivariate continuous time survival time models. These include the parametric models (with hazard functions of the type discussed in Lesson 2) and the semi-parametric Cox model.

1. **Introduction**

Stata provides an extensive suite of estimators. Parametric regression survival-time models (including the piece-wise constant exponential model) are estimated by maximum likelihood using **streg**. Models corresponding to six types of parametric distribution can be estimated: Exponential, Weibull, Log-logistic, Gompertz, Lognormal, and Generalised Gamma. We will focus on the first three (discussed in Lesson 2).

To use these programs, you must **stset** the data first, as discussed in Lesson 3.

I discuss and illustrate **streg** and **stcox** in turn, using the Cancer data set assumed to be **stset** already. At the end I ask you, as an exercise, to repeat parts of the analysis with alternative models or with different data sets.

Note that typing **streg** by itself after estimating a model with **streg**, or typing **stcox** by itself after estimation with **stcox**, will result in the previous estimates being replayed on the screen.

1. **Estimation using streg (and plotting fitted curves with stcurv)**

The different parametric models estimated by **streg** share a common command syntax – the different distributions are chosen via option specifications. See **help streg** for the full command syntax and all the options available. We will ignore the **frailty(.)** option the moment. Frailty (unobserved heterogeneity) models are considered separately in Lesson 7.

The basic syntax is

streg [varlist], dist(distname) nohr time tr nolog

**dist(distname)** specifies the survival model to be estimated. **distname** is one of the following: **exponential**, **weibull**, **gompertz**, **lognormal**, **loglogistic** or **gamma**. Abbreviations are allowed (to the minimum, as underlined).

As Stata’s on-line help says (this is text modified from Stata version 17, which still applies):

‘nohr’ specifies that coefficients rather than exponentiated coefficients are to be displayed or, said differently, coefficients rather than hazard ratios. This option is valid only for models with a proportional hazard ratio parameterization: exponential, Weibull, and Gompertz.

‘hr’, which can be specified when the model is estimated or when redisplaying results, states that the underlying log relative hazard coefficients are to be displayed. This option affects only how results are displayed, not how they are estimated.

‘time’ specifies that the model is to be estimated in the accelerated failure-time metric rather than the log relative-hazard metric. This option is only valid for the exponential and Weibull models since they have both a hazard ratio and an accelerated failure-time parameterization. For these two models, in the log relative-hazard metric, estimates of (B,s) are produced and in the accelerated failure-time metric, estimates of (-B\*s,s) are produced.

Regardless of metric, the likelihood function is the same and models are equally appropriate viewed in either metric; it is just a matter of changing interpretation. ‘time’ must be specified when the model is estimated.

‘tr’ is appropriate only for the log-logistic, lognormal, and gamma models, or for the exponential and Weibull models when estimated in the log expected time metric. ‘tr’ specifies that exponentiated coefficients are to be displayed, which have the interpretation of time ratios. ‘tr’ may be specified when the model is estimated or when results are redisplayed

‘nolog’ prevents streg from showing the iteration log.

**stcurve** and **predict** are commands used after having run **streg**. See below.  
Recall that for models which can be written in the *proportional hazards* (PH) metric, the

hazard function for person *i* can be written  
*hi*(*t*, *Xi*) = *h*0(*t*).λ*i*, where λ*i* ≡ exp(β′*Xi*), or

log[*hi*(*t Xi*)] = log[*h*0(*t*)] + *Xi*β

where *h*0(*t*) is the baseline hazard, *Xi* is a vector of individual characteristics, and β is a vector regression coefficients and includes an intercept term. In a PH model, λ*i* scales the baseline hazard multiplicatively by the same amount at each value of *t*.

For PH models Stata reports estimates for covariate *k* of either β*k* (use the **nohr** option) or of the ‘hazard ratio’, exp(β*k*), for which you use the **hr** option. The PH form is referred to as the ‘log relative hazard’ in Stata output.

The empirical illustration uses the Cancer data set, which has already been **stset**. Recall that there are two variables in the data set which are available to be used as covariates: age and drug. I shall recode the drug variable from three categories into a simpler binary variable summarising whether the subjects receive the drug or not.

. recode drug 1=0 2/3=1

(drug: 48 changes made)

. lab var drug "receives drug?"

. lab def drug 0 "placebo" 1 "drug"

. lab val drug drug

1. **Estimation of the Weibull model**

The Weibull model estimates are:

streg drug age, dist(weibull) nolog nohr

Failure \_d: died

Analysis time \_t: studytime

Weibull PH regression

No. of subjects = 48 Number of obs = 48

No. of failures = 31

Time at risk = 744

LR chi2(2) = 35.39

Log likelihood = -42.931335 Prob > chi2 = 0.0000

------------------------------------------------------------------------------

\_t | Coefficient Std. err. z P>|z| [95% conf. interval]

-------------+----------------------------------------------------------------

drug | -2.196936 .4087791 -5.37 0.000 -2.998129 -1.395744

age | .1202027 .0371599 3.23 0.001 .0473707 .1930348

\_cons | -10.58396 2.326271 -4.55 0.000 -15.14337 -6.024553

-------------+----------------------------------------------------------------

/ln\_p | .5204297 .1389037 3.75 0.000 .2481834 .792676

-------------+----------------------------------------------------------------

p | 1.682751 .2337403 1.281695 2.209301

1/p | .5942651 .0825456 .452632 .7802168

------------------------------------------------------------------------------

The **nohr** option meant that coefficient estimates were shown. We can show the corresponding hazard ratio estimates by simple replaying the command and adding the **hr** option (if we had wished, instead we could have used the **nohr** option and replayed using the **hr** option). Interpretation of the estimates follows the display of the estimates in hazard ratio form.

The estimates suggest that the hazard rate is increasing over time at a decreasing rate: note that 1 < *p* < 2 (see Lesson 2). In the Weibull model, the ratio of the hazard rate at survival time *t* to the hazard rate at survival time *u*, given the same *X*, is given by (*t*/*u*)α–1. Thus according to our model estimates, the ratio of the hazard rate at time 10 to that at time 5 is 1.6, and the ratio of the hazard rate at time 30 to that at time 5 is 3.4:

. di "h(10,X)/h(5,X) = " (10/5)^(e(aux\_p)-1)

h(10,X)/h(5,X) = 1.6051972

. di "h(30,X)/h(5,X) = " (30/5)^(e(aux\_p)-1)

h(30,X)/h(5,X) = 3.3984681

The coefficient estimates indicate that those receiving the drug have lower hazard rates *ceteris paribus* (i.e. lower conditional death rates and hence longer survival times). Note the negative (and statistically significant) coefficient for drug in the **nohr** representation and the hazard ratio for drug less that one in the **hr** representation: 0.11 = exp(–2.2). The estimates imply that, at each survival time, the hazard rate for those who received the drug is only 11% of the hazard rate for those who received the placebo. The output also shows that there is a positive association between age and the hazard rate: older people die earlier. In fact a one year rise in age is associated with a 13% higher hazard rate.

The elasticity of the hazard rate with respect to a one unit change in the *k*th explanatory variable is given by β*kXik*; for age, it is therefore (0.1202027)\*age*i*. (If the explanatory variable had instead been ln(age) rather than age, the estimated coefficient on ln(age) would be the elasticity of the hazard with respect to age.) Here are the elasticities:

. \* Elasticity of hazard w.r.t. age (age covariate in levels) = b\_age \* age

. ge elas\_age = \_b[age]\*age

Observe the way in which we can retrieve and refer to the estimated model coefficients: **\_b[*something*]** refers to the estimated coefficient on the explanatory variable with name *something* in the last regression that was run. (One can also refer to many other estimates after running regressions, including estimated standard errors, log-likelihood values, and so on: see the User’s Guide.)

More generally, hazard rate ratios at each survival time are related to absolute differences in characteristics: *h*(*t*,*X*1)/*h*(*t*,*X*2) = exp[β′(*X*1–*X*2)]. Thus a ten year difference in age, other things equal, is associated with a hazard rate ratio of some 3.3. Some one aged *y*+10 and who is receiving the drug has a hazard ratio that is 37% of some one aged *y* who gets the placebo:

. di "h(t;age=y+10,drug=x)/h(t;age=y,drug=x) = " exp(\_b[age]\*10)

h(t;age=y+10,drug=x)/h(t;age=y,drug=x) = 3.3268546

. di "h(t;age=y+10,drug=1)/h(t;age=y,drug=0) = " exp(\_b[age]\*10 + \_b[drug]) h(t;age=y+10,drug=1)/h(t;age=y,drug=0) = .36975709

Let us now look at the estimated hazard and survivor functions graphically. We can do this using **stcurv**, run after **streg**. For example:



Note the monotonically rising hazard. The corresponding survival curve is as follows

stcurv, survival title("Cancer data, at sample means") saving(streg1,replace)



Observe that **stcurv** can also be used for out-of-sample projections, i.e. showing what the estimated functions look like at survival times beyond the range that exists in the estimation sample. To do this use the **range(# #)** option to **stcurve**. Here’s what the previous hazard and survival curves look like if the analysis time axis is extended to 50.

The last graphs were, by default, drawn with the covariates set at their mean values. This does not make a lot of intuitive sense for categorical covariates. Compare instead the survivor curves for persons with drug = 0 and drug = 1 (and mean age), making use of the **at(.)** option.

. stcurv, survival title("Cancer data:drug=0") at(drug=0) /// > saving(streg5,replace)

stcurv, survival title("Cancer data:drug=0") at(drug=0) ///

saving(streg5,replace)

stcurv, survival title("Cancer data:drug=1") at(drug=1) ///

saving(streg6,replace)

\* put both curves on one graph

stcurv, survival title("Survival, Cancer data: drug=0,1") at(drug=0) at(drug=1) ///

saving(streg5a,replace)



Let us now *calculate* the median and mean survival times, using the formulae discussed in Lesson 2. To do this we have to specify the values of the covariate vector (*X*), and thence can derive λ*i*. (In Chapter 2, we simulated values for a particular value of λ.)

The code below shows first how to calculate the Weibull median and mean for the case when the covariates are set at the sample average values. Second it shows how to calculate the median and mean for each person in the sample – we can then examine the values for particular covariate combinations using **list**.

In both cases, the derivations use the **predict** command after **streg**. The **xb** option with **predict** generates a new variable equal to the estimate β′*Xi* for each person *i*. The other calculations also use other automatically saved results, such as the mean after a **summarize**, **r(mean)**, and the estimate of the Weibull shape parameter *p* as **e(aux\_p)**. I have also used a couple of local macros to hold scalar results to use in other calculations (see **help macro**). In fact, the calculation of mean and mean medians can be done directly using **predict**.

[Note: after all ‘**e**stimation class’ commands, examples of which are mostly regression commands (including **streg**, **logit**, **cloglog**), Stata saves a variety of results in objects with names **e(*something*)**. You can find the full list of saved results by typing **ereturn list** after an estimation command. Examples include **e(b)** which is a vector containing the parameter estimates, and **e(V)** which is a matrix containing the variance-covariance matrix of the parameter estimates. Different commands save extra results relevant to their model; e.g. after a Weibull regression, **e(aux\_p)** contains the shape parameter α. Results are also saved after commands like **summarize** and **tabulate** in objects with names like **r(*something*)**. You can find the full list of saved results by typing **return list** after one of these ‘rclass’ commands. E.g. after a **summarize**, **r(mean)** contains the estimate of the mean. For a more complete discussion of saved results, see the Manuals. Finally, observe that virtually all estimation commands may be followed with a **predict** command that generates predictions for the observations in memory, based on the parameter estimates of the most recent model. The sorts of things that one can predict depends, of course, on the command. See the Manual entries for the relevant command about **predict** for that command.]

First is the code for predictions for the case when the covariates are set at the sample average values. The trick here is to note that our calculations require the value β′*Xm* where *Xm* is a vector containing the sample mean values of the characteristics. Instead of first calculating *Xm* and then β′*Xm*, we take advantage of the fact that β′*Xm* is equal to the mean of the individual β′*Xi* for each subject *i* in the sample. But I know that **predict** will produce the β′*Xi* so all we have to do is generate that and take its mean. Then we can feed the result into our calculation of the mean and median spell lengths.

predict xb, xb

. su xb

Variable | Obs Mean Std. dev. Min Max

-------------+---------------------------------------------------------

xb | 48 2.786804 .8510832 1.064673 4.091489

Now, second, we examine how to calculate the estimated mean and median survival time for every person in the sample.

. \* median duration for each person in sample

. ge mediand = (ln(2)\*exp(-xb))^(1/e(aux\_p))

. \* expected (mean) duration for each person in sample

. ge meand = exp(-xb/e(aux\_p))\*exp(lngamma(1+1/e(aux\_p)))

In fact, Stata allows you to calculate these variables directly, using **predict** after **streg**, rather than calculating them by hand. Here’s how:

. predict mediandS, median time

. predict meandS, mean time

Finally, let’s compare the estimated means and medians for two (hypothetical) persons, call them *i* and *j*, each of which has an age equal to the sample mean age, but one received the drug and the other didn’t. (These comparisons parallel those that we undertook in Lesson 2.) First we drop the previous variables, then we find the mean age using **summarize** and place its value into a **local** macro that we can refer to later. The next steps compute β′*Xi* and β′*Xj* for the two individuals *i* and *j*, and then finally we substitute these values into the formula for the mean and median for the Weibull model.

We now have the mean age. Now follows the calculations for the placebo recipient.

. local xb0 = \_b[\_cons] + \_b[age]\*`meana' + \_b[drug]\*0

. di "Mean age = " `meana' " ,\_b[\_cons] + \_b[age]\*(mean age) + \_b[drug]\*0 = " `

> xb0'

Mean age = 55.875 ,\_b[\_cons] + \_b[age]\*(mean age) + \_b[drug]\*0 = -3.8676331

. di "Pred. Median [mean(age), drug=0] = " (ln(2)\*exp(-`xb0'))^(1/e(aux\_p)) Pred. Median [mean(age), drug=0] = 8.0092224

. di "Pred. Mean [mean(age),drug=0] = " exp(-`xb0'/e(aux\_p))\*exp(lngamma(1+1/e( > aux\_p)))  
Pred. Mean [mean(age),drug=0] = 8.8915291

Here are the calculations for the drug recipient.

. local xb0 = \_b[\_cons] + \_b[age]\*`meana' + \_b[drug]\*1

. di "Mean age = " `meana' " ,\_b[\_cons] + \_b[age]\*(mean age) + \_b[drug]\*1 = " `

> xb0'

Mean age = 55.875 ,\_b[\_cons] + \_b[age]\*(mean age) + \_b[drug]\*1 = -6.0645694

. di "Pred. Median [mean(age), drug=1] = " (ln(2)\*exp(-`xb0'))^(1/e(aux\_p)) Pred. Median [mean(age), drug=1] = 29.552145

. di "Pred. Mean [mean(age),drug=1] = " exp(-`xb0'/e(aux\_p))\*exp(lngamma(1+1/e( > aux\_p)))  
Pred. Mean [mean(age),drug=1] = 32.807649

The results highlight again the very large difference in the survival time distribution between drug and placebo recipients. Observe too the difference between the mean and median durations.

Exactly the same principles as described here could be used if you had a model with a large number of explanatory variables rather than simply two.

The **predict** command after **streg** can be used to create other types of variables.

1. **Estimation of the piece-wise constant exponential model using streg**

The parametric models that we have considered make strong assumptions about the shape of the hazard function, and the Cox model makes none. Sometimes an in between approach is more appealing, in which we fit a semi-parametric hazard. The piece-wise constant exponential model is the model most commonly used for doing this (in a continuous time modelling framework). The hazard is assumed constant within pre-specified survival time intervals but the constants may differ for different intervals.

The model is simple to estimate using **streg, dist(exponential)** but first requires some reorganisation of the data and creation of some time-varying covariates.

Recall that the exponential model is  
*hi*(*t*) = *h*0.λ*i*, where λ*i* ≡ exp(β′*Xi*), or

log[*hi*(*t*)] = log(*h*0) + β′*Xi* since, in this case *h*0(*t*) = *h*0, a constant.

We can generalise this specification to have a constant hazard within each of *K* intervals along the survival time axis:

log[*hi*(*t*)] = log(*h*01) + β′*Xi*, *t* ∈ (0, τ1] log[*hi*(*t*)] = log(*h*02) + β′*Xi*, *t* ∈ (τ1, τ2] ...  
log[*hi*(*t*)] = log(*h*0K) + β′*Xi*, *t* ∈ (τK-1, τK]

All we need to estimate the model is to generate variables which allow the constant term in the hazard regression to differ from interval to interval. This we do by changing the organisation of the data set (**expand**ing it or **stsplit**ting it) and specifying the variables using appropriate dummy variables. Reread the relevant section in Lesson 3.

Suppose the estimates above lead us to wish to allow the baseline hazard to differ over three intervals (0,8], (8, 17] and (17, 39]. In Lesson 3 we showed how to split episodes and create dummy variables that linked the (new) episodes with these time intervals – we called these variables, e1, e2, and e3, respectively. We then have two possible strategies in estimation: either we include all three variables (e1, e2, and e3) as regressors in the model and exclude the constant term, or we can include a constant term but only two of the variables. (We can’t include all three variables plus a constant because that would introduce a collinearity between the regressors, and the model could not be estimated.) I prefer the second display because it allows us to look directly at how the baseline hazard for the two intervals in question differs from that of the interval corresponding to the excluded variable. (This is a presentational or interpretational issue – the models are the same models and would generate the same predictions.)

Here, first, are the estimates for the case where all three variables are included, there is no constant term, and we want coefficients displayed rather than hazard ratios.

streg, nohr

Exponential PH regression

No. of subjects = 48 Number of obs = 98

No. of failures = 31

Time at risk = 744

LR chi2(4) = 30.42

Log likelihood = -46.134703 Prob > chi2 = 0.0000

------------------------------------------------------------------------------

\_t | Coefficient Std. err. z P>|z| [95% conf. interval]

-------------+----------------------------------------------------------------

drug | -2.010659 .4069367 -4.94 0.000 -2.80824 -1.213077

age | .1031196 .0357331 2.89 0.004 .033084 .1731553

e2 | .4585195 .4406342 1.04 0.298 -.4051077 1.322147

e3 | 1.079937 .4924212 2.19 0.028 .1148094 2.045065

\_cons | -8.178818 2.071186 -3.95 0.000 -12.23827 -4.119368

------------------------------------------------------------------------------

Now, second, see what happens if, instead, you estimate the model including the last two variables and a constant term:

streg drug age e2 e3, dist(exp) nolog nohr

Failure \_d: died

Analysis time \_t: studytim

ID variable: id

Exponential PH regression

No. of subjects = 48 Number of obs = 98

No. of failures = 31

Time at risk = 744

LR chi2(4) = 30.42

Log likelihood = -46.134703 Prob > chi2 = 0.0000

------------------------------------------------------------------------------

\_t | Coefficient Std. err. z P>|z| [95% conf. interval]

-------------+----------------------------------------------------------------

drug | -2.010659 .4069367 -4.94 0.000 -2.80824 -1.213077

age | .1031196 .0357331 2.89 0.004 .033084 .1731553

e2 | .4585195 .4406342 1.04 0.298 -.4051077 1.322147

e3 | 1.079937 .4924212 2.19 0.028 .1148094 2.045065

\_cons | -8.178818 2.071186 -3.95 0.000 -12.23827 -4.119368

------------------------------------------------------------------------------

Finally here is that same model again, but now with hazard ratios displayed.

streg drug age e1 e2 e3, dist(exp) nolog

Failure \_d: died

Analysis time \_t: studytim

ID variable: id

note: e3 omitted because of collinearity.

Exponential PH regression

No. of subjects = 48 Number of obs = 98

No. of failures = 31

Time at risk = 744

LR chi2(4) = 30.42

Log likelihood = -46.134703 Prob > chi2 = 0.0000

------------------------------------------------------------------------------

\_t | Haz. ratio Std. err. z P>|z| [95% conf. interval]

-------------+----------------------------------------------------------------

drug | .1339005 .054489 -4.94 0.000 .0603111 .297281

age | 1.108624 .0396146 2.89 0.004 1.033637 1.189051

e1 | .3396168 .1672345 -2.19 0.028 .1293718 .891536

e2 | .5371822 .2686525 -1.24 0.214 .201569 1.431593

e3 | 1 (omitted)

\_cons | .000826 .0016521 -3.55 0.000 .0000164 .0416307

------------------------------------------------------------------------------

Note: \_cons estimates baseline hazard.

The hazard for the third interval (17, 39] is 2.94 times higher than the hazard for the first interval (the reference category). That it is higher is what we would expect from the non- parametric estimates. The hazard ratios for drug and age are similar to those estimated by the other PH models.