**6. Estimation: (ii) discrete time models (logistic mainly)**

1. **Aim**

The aim of this lesson is to illustrate how to use Stata to estimate multivariate discrete time (grouped data) survival time models of the type discussed in Lesson 2.

1. **Introduction:**

Stata does not have a set of specialist commands for estimating the discrete time proportional odds or proportional hazards models. But they are very easy to estimate nonetheless. All one has to do is re-organise the data set, define some new variables (to specify the baseline hazard function in particular), and then apply logit or cloglog regression.

Derivation of predicted survival times (median durations etc) is a little more fiddly because there are no closed-form formulae for these except in special cases, but the **predict** command makes things relatively straightforward.

The illustrations concerning discrete time models use the Cancer data set in the same way as Lesson 5 about continuous time models did. This is done deliberately in order to highlight the similarities and differences between the modelling (and the estimates).

This lesson first discusses how to re-organise the data set and define the new variables, necessary for estimation of both proportional odds and proportional hazard models. Then I discuss estimation. The discrete time models are estimated by maximum likelihood using **logit** and **cloglog** (or **logistic** and **glm**: see below). We will focus here on the discrete logistic (proportional odds) model. Estimation of the discrete complementary log-log (proportional hazard) model is very similar:

1. **Data reorganisation and creation of new variables**

Revise the material discussing this in Lesson 3. Recall that our ‘easy estimation’ methods for the discrete models are based on application of standard binary dependent variable models to re-organised data.

The data set must be re-organised so that, for each person, there are as many data rows as there are time intervals at risk of the event occurring for each person. We need to go from the simple data set discussed earlier, with one row of data per person, to another data set in which each person contributes *Ti* rows, where *Ti* is the number of time periods (e.g. months) *i* was at risk of the event. In effect an unbalanced panel data set-up is required.

We also require a unique identifier variable for each subject (if it doesn’t already exist), plus a spell month identifier variable for each subject. The binary dependent variable also needs to be created. If subject *i*’s survival time is censored, the binary dependent variable is equal to 0 for all of *i*’s spell months; if subject *i*’s survival time is not censored, the binary dependent variable is equal to 0 for all but the last of *i*’s spell months (month 1,..., *Ti*–1) and equal to 1 for the last month (month *Ti*). To emphasize that time now refers to discrete intervals, I use the notation *j* for elapsed duration rather than *t* as in Lesson 5.

Let us illustrate this using the Cancer data set (as in Lesson 3), with the drug variable recoded into two categories (as before):

. use cancer

(Patient Survival in Drug Trial)

. ge id = \_n

. lab var id "subject identifier"

. recode drug 1=0 2/3=1

(48 changes made)

. lab var drug "receives drug?"

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

. lab val drug drug

Now we do the episode splitting, producing a data set in person-month format, exactly we did in Lesson 3 (see the discussion there explaining the Stata code used).

. expand studytim

(696 observations created)

. bysort id: ge j = \_n

. \* spell month identifier, by subject

. lab var j "spell month"

. bysort id: ge dead = died==1 & \_n==\_N

. lab var dead "binary depvar for discrete hazard model"

Remember that we do *not* have to **stset** the data for estimation, because we do not use the **st** commands – they are for the continuous time case.

1. **Choose the functional form for the baseline hazard function**

The final step prior to estimation is to choose a functional form for the baseline hazard function. We do this by defining new time-varying covariates which are functions of survival time *t* per person (variable t in the illustration). Lesson 3 briefly discussed this. Here we consider several alternative specifications (from the many!): log(time), polynomial in time, piece-wise constant, and fully non-parametric.

For the log(time) and cubic polynomial specifications, the new variables are:

. ge lnj = ln(j)

. ge j2 = j^2

. ge j3 = j^3

For a non-parametric baseline, we need to create duration-interval-specific dummy variables, one for each spell month at risk. The maximum survival time in the Cancer data set is 39, so we need 39 dummy variables. There are 2 methods (at least) of creating them. First, you: can use **tabulate, generate(.)** where the argument of the generate option is the common prefix (‘stub’) of the 39 new variables, ‘d’ in this case:

This creates 39 dummy variables called d1–d39. This is confirmed by describing the data in compact form using the **ds** command (cf. the more verbose **describe** command):

Alternatively one can use the **forvalues** command (see **help forvalues**) to generate the variables in a loop. The local macro ‘x’ is set equal to 1 the first time the loop and generates durat1 = 1 if the spell month identifier is equal to one, sets durat1 = 0 otherwise. The ‘x’ then increments to 2 and generates durat2 = 1 if the spell month identifier is equal to 2, it is set equal to 0 otherwise. And then ‘x’ increments by 1 again, and generates another variable, and so on, with ‘x’ =39 being the last loop.

The 39 dummy variables called durat1–durat39 are created. In fact we will use d1–d39, so we can

. drop durat\*

to save on memory usage.

An example of a different piece-wise constant specification was already shown in Lessons 3 and 5. Let us assume that the interval (discrete) hazard is constant in months 1–8, 9–17, and 18+. The dummy variables we require are:

. ge e1 = j < 9

. ge e2 = j >= 9 & j <= 17

. ge e3 = j >= 18 & j <.

An alternative allowing more pieces, six in fact, would be the following:

. ge dur1 = d1+d2+d3+d4+d5+d6

. ge dur2 = d7+d8+d9+d10+d11+d12

. ge dur3 = d13+d14+d15+d16+d17+d18

. ge dur4 = d19+d20+d21+d22+d23+d24

. ge dur5 = d25+d26+d27+d28+d29+d30

. ge dur6 = d31+d32+d33+d34+d35+d36+d37+d38+d39

The reason for the splitting of survival times at the particular points chosen will become apparent shortly (it ensures there are events occurring within each of the time intervals so defined). See also Ex. 6.1.

To conserve memory space after creating these variables, you will find it useful to **compress** the data in order to conserve disk space.

. compress

If you wish to estimate a model with fully non-parametric baseline hazard, then it is essential to check whether events occur at each value of *j* (i.e. the variable ‘j’ that we created). The hazard cannot be estimated for values of *j* with no events (exactly as with the non-parametric baseline hazard in the Cox model).

If there are duration intervals with no events, then either one must refine the grouping on the survival time dimension (this was the rationale for creating the variables dur\* above), or else one must drop the relevant person months from the estimation. (Cf. the discussion of identification of the logit model in the Reference Manuals under ‘perfect predictors’.)

Checking whether there are events within each of the intervals is straightforward. Cross- tabulate the spell month identifier with the censoring variable.

1. **Estimation**

For ML estimation of the discrete time logistic model we use **logit**. The basic syntax is

logit depvar varlist, [or noconstant]

‘depvar’ is the (new) event variable – dead in the illustration – and ‘varlist’ refers to the explanatory variables (covariates) together with the variables used to summarise the baseline hazard function.

If **logit** is used without the **or** option, coefficients are reported; with the **or** option, odds ratios (exponentiated coefficients) are reported. For an alternative way of getting the latter directly, see **help logistic**. The **noconstant** option means estimate a model without a constant term –

we mainly use this for estimating models with a fully non-parametric baseline hazard. Note that odds ratios of hazard rates refer to ratios of form [*h*1/(1–*h*1)] / [*h*0/(1–*h*0)] for the one unit change in an explanatory variable from zero to one. I personally find these difficult to interpret. On the other hand, as *h* → 0, the odds ratio tends to the hazard ratio *h*1/*h*0, which does have a ready interpretation.

For ML estimation of the discrete time complementary log-log model we use **cloglog**. The basic syntax is

cloglog depvar varlist, [or noconstant]

**depvar** and **varlist** and the **noconstant** options are as for the logit model. To produce exponentiated coefficients, simply use the **eform** option. Recall that the exponentiated coefficients can be interpreted as hazard ratios since the cloglog model is the discrete time proportional hazards model.

The cloglog model (both with or without exponentiated coefficients) can also be fitted using the **glm** command: see Exercise 6.1.

See **help logit** and **help cloglog** for the full command syntax and all the options available. As with all Stata’s estimation commands, estimation output can be re-played by simply re- issuing the command name again.

To take just a single illustration of the **logit** model, consider the following specification which is a discrete time analogue of the piece-wise constant exponential model considered in Lessons 3 and 5. In that case, we assumed that the (continuous time) hazard rate was constant between survival times (0, 8], (8, 17], and (17, ∞) where the numbers refer to exact dates. Now we are assuming that the interval (discrete) hazard is constant in months 1–8, 9–17, and 18+; we created the necessary dummy variables corresponding to these intervals earlier in the Lesson. They are the variables called e1, e2, and e3. For the reasons discussed in the previous Lesson, we include only two of these as regressors and also include a constant term. (Alternatively one could include all three dummies, but then one must also exclude the constant term using the **nocons** option.)

Time as **logarithmic function**

logit, or

Logistic regression Number of obs = 744

LR chi2(3) = 35.41

Prob > chi2 = 0.0000

Log likelihood = -111.16102 Pseudo R2 = 0.1374

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

dead | Odds ratio Std. err. z P>|z| [95% conf. interval]

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

drug | .1004883 .0440996 -5.24 0.000 .0425171 .2375022

age | 1.133561 .044389 3.20 0.001 1.049816 1.223988

lnj | 1.970237 .5087869 2.63 0.009 1.187701 3.268358

\_cons | .0000354 .0000843 -4.30 0.000 3.31e-07 .003777

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

Note: \_cons estimates baseline odds.

. logistic dead drug age lnj, nolog

Logistic regression Number of obs = 744

LR chi2(3) = 35.41

Prob > chi2 = 0.0000

Log likelihood = -111.16102 Pseudo R2 = 0.1374

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

dead | Odds ratio Std. err. z P>|z| [95% conf. interval]

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

drug | .1004883 .0440996 -5.24 0.000 .0425171 .2375022

age | 1.133561 .044389 3.20 0.001 1.049816 1.223988

lnj | 1.970237 .5087869 2.63 0.009 1.187701 3.268358

\_cons | .0000354 .0000843 -4.30 0.000 3.31e-07 .003777

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

Note: \_cons estimates baseline odds.

Time as **piece-wise constant**

logit dead drug age e2 e3, nolog

Logistic regression Number of obs = 744

LR chi2(4) = 32.39

Prob > chi2 = 0.0000

Log likelihood = -112.66968 Pseudo R2 = 0.1257

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

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

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

drug | -2.179088 .4349997 -5.01 0.000 -3.031672 -1.326505

age | .1130895 .0380588 2.97 0.003 .0384956 .1876833

e2 | .5014397 .4645604 1.08 0.280 -.4090819 1.411961

e3 | 1.181932 .5274663 2.24 0.025 .1481172 2.215747

\_cons | -8.610281 2.187815 -3.94 0.000 -12.89832 -4.322242

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

. logit, or

Logistic regression Number of obs = 744

LR chi2(4) = 32.39

Prob > chi2 = 0.0000

Log likelihood = -112.66968 Pseudo R2 = 0.1257

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

dead | Odds ratio Std. err. z P>|z| [95% conf. interval]

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

drug | .1131446 .0492179 -5.01 0.000 .0482349 .2654033

age | 1.119732 .0426156 2.97 0.003 1.039246 1.206451

e2 | 1.651097 .7670341 1.08 0.280 .6642598 4.103997

e3 | 3.260668 1.719892 2.24 0.025 1.159649 9.168256

\_cons | .0001822 .0003987 -3.94 0.000 2.50e-06 .0132701

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

Note: \_cons estimates baseline odds.

The estimates imply that the odds of dying for drug recipients is about one-tenth the odds for placebo recipients. For age, the estimate implies that the odds of dying increase by about 13% with each extra year of age. But ask yourself if you understand what these changes in odds mean. I find it easier to understand hazard ratios.