Hour 4: Discrete time duration model, piecewise constant, complementary log-log estimation and model misspecification

Although the time itself is continuous, in real life we seldom have the survival data in a continuous form. More often, we have data that comes in a discrete, or time interval style.

It is important to understand how the discreteness stems from. One source of discreteness is due to the sampling practice, such as unemployment register data. Another source could be genuine discreteness of event, i.e. things just happen at certain point of time (exhaustion of unemployment benefits). We here will focus on the first source, in which the underlying process is continuous, while the data we work on is discrete due to sampling practice.

It is better known as the grouped-hazard duration model.

$$p(t-1<T<t \mid T>t-1)=\frac{S(t-1)-S(t)}{S(t-1)}$$

$$=1-\frac{S(t)}{S(t-1)}$$

$$=1-\exp\left(-\int_{t-1}^{t} h(s)ds\right)\approx h(t) \text{ for } \int_{t-1}^{t} h(s)ds \text{ "small"}$$

With proportionality assumption, this can be further reduced to a complementary log-log form

$$p(t-1<T<t \mid T>t-1)=1-\exp\left(-\exp(x_t'\beta+\lambda_t)\right)$$

Notice that the baseline λ_t has an additive form to structural covariates. Thus if the observational data is discrete, we can use flexible non-parametric specification for baseline, and estimate it as if they are covariates.

Stata does not make an effort especially for discrete time duration analysis. Most of the commands for continuous time survival data work seemingly, but no guarantee for correctness of results.

Exercise 4: discrete data with underlying Weibull distributed baseline.

In this exercise, we are going to use a dataset (discrete_weibull.dta) generated with the same underlying Weibull baseline as in continous_weibull.dta, but we divide the duration into 12 periods.

Period	Baseline b(t)
1	0
2	-0.14379
3	-0.19625

4	-0.23026
5	-0.25554
6	-0.27568
7	-0.29243
8	-0.30677
9	-0.3193
10	-0.33044
11	-0.34046
12	-0.34956

Where b(t) is the log of integrated Weibull hazard

$$b(t_d) = \ln\left(\int_{d-1}^{d} \alpha t^{\alpha-1} dt\right) = \ln\left(t^{\alpha}\Big|_{a-1}^{d}\right),\ where\ \alpha = 0.9,\ d = [1,12]$$

So in this case we have a so-called piecewise constant baseline, where baseline is constant within unit time intervals.

Recall that the full Weibull hazard formulation is (with scale parameter)

$$h(t) = b(t)\exp(X'\beta) = \lambda^{\alpha}\alpha t^{\alpha-1}\exp(X'\beta)$$

And the discrete_weibull.dta is simulated with scale parameter set to 0.1, such that the duration would not be too short (otherwise we will have problem with episode splitting). Then the $h(t) = b(t)\exp(X'\beta) = \lambda^{\alpha}\alpha t^{\alpha-1}\exp(X'\beta) = \alpha t^{\alpha-1}\exp(X'\beta + \ln(\lambda^{\alpha}))$.

The probability of transition (note, this is not the hazard rate, but probability!) in each interval [d-1, d] is thus:

$$p(d-1 < t < d \mid t > d-1) = 1 - \exp\left(-\exp(x'\beta + b(t_d) + const\right)$$

where the const is just the scaling factor

$\ln(\lambda^{\alpha})$ = -2.072326584 when $\alpha = 0.9,\ \lambda = 0.1$.

1. open data file discrete_weibull.dta.
2. describe the variables
3. do a summarizing statistics

We notice that there are 12 periods for duration variable t, so we need to define 12 dummies for piecewise constant baseline. Each individual contribute different numbers of periods to the overall durations. Thus we need to split each individual's duration spell to subspells, so that they sum up to the original total length. Then for each subspell we define the corresponding dummy to reflect which subspell/period this is referring to. Each subspell will have the censoring indicator to be 0, meaning censored, except the last subspell for individual will retain its original censoring status.

4. episode splitting by expand: **expand t;** (duplicate each observation by t, the total duration period. Thus each subobservation generated reflects one period of the original duration, and they sum up to original t.

5. let's have a look of data at this stage. You see that every variable is duplicated, and the censoring indicator is 1 for each. We will have to correct that.
   sort id;
   by id: replace d=0 if _n!=_N; /* Note the use of index variable ***/**

6. we will need to define each subspell's id.
   by id: gen epid=_n;
   note that epid[_n]-epid[_n-1]=1 is the exact length of subspell.
7. list some variables to check the data now.

8. Since we are assuming that for each subspell the duration baseline is constant (piecewise constant) we will have to define a set of dummies to indicate which period the subspell actually is corresponding to. This can be done in several ways, the easiest is however
   tabulate epid, gen(dur)
   this command runs the frequency statistics of epid, which is the id for subspells, for each individual. Tabulate can at the same time create a set of dummies which are corresponding to the actual subspell in total duration period. This can be checked by list some variables: **list id d epid dur1-dur12 in 1/5;**

9. now that we have the data ready, we can do some estimation. Note here, we haven't used the stset command. This is because we are not going to use Stata's st estimation command. Due to the complementary log-log form of grouped hazard formulation, it is straight forward to estimate with the **cloglog** command, treating censoring indicator d as binary response variable.
   cloglog depvar indepvar, options;
   here:
   cloglog d x y dur2-dur12; /* note here we drop dur1 to estimate with a constant, then the constant is in fact sum of baseline hazard of dur1 which is equal to 0 , and the scale factor */

   alternatively suppress the constant (which means scale the dur1-dur12),
   cloglog d x y dur1-dur12, noconstant

   we can see that the coefficients for x and y are not bad, but the piecewise constant duration dummies are all insignificant. This could be due to the fact that we have a handful observations for each dur# dummy. But we observe some falling tendency from dur1-dur12, which indicates the falling baseline of Weibull.

10. What if we do this with logit, that is we simplify the duration model, and treat the censoring indicator d as binary response variable and do
    logit d x y dur1-dur12, noconstant
    This give pretty biased estimates.

11. what if we discard the discreteness of data and treat the duration as continuous when in fact it is discrete?

stset epid, id(id) failure(d);
(optionally do some statistics with the data)

suppose we want fit the data with stcox?
stcox x y, nohr;
or what if we fit it with continuous Weibull?
streg x y, dist(w) nohr;

You can see that the results are pretty much nonsense. This should draw your attention that, when empirical data present some discreteness feature, either due to sampling practice, or due to the nature of data generating process, fitting the data with an arbitrary distribution model might result in severe bias in estimates.

Røed and Zhang (2002) has demonstrated that even though the underlying DGP is pure continuous, if the data observed is discrete, fitting with a theoretically correct model will still produce non- neglectable bias, which is termed “time aggregation bias”. So one must be careful before make any assumption of duration dependence.

12. fitting data without episode splitting (expand command) yields exact same (wrong) results.