stcrmix and Timing of Events with Stata

Christophe Kolodziejczyk, VIVE

August 30, 2017

Introduction

- I will present a Stata command to estimate mixed proportional hazards competing risks models (`stcrmix`).
- This implemention follows closely Gaure et. al.'s implementation which has actually been used in some of their other previous papers. Simen Gaure has written an R-package (crmph).
- Reference: Gaure, Simen & Roed, Knut & Zhang, Tao, 2007. "Time and causality: A Monte Carlo assessment of the timing-of-events approach," Journal of Econometrics, Elsevier, vol. 141(2), pages 1159-1195, December.
- can be used to estimate timing of events models.

Outline

- I will briefly present the model generally and two of its variants (in continous and discrete time).
- I will talk about the non-parametric maximum likelihood estimator (NPMLE).
- I will review the likelihood function for the two variants
- In light of these likelihood I will then present how to set up the data

The model in a nutshell

- competing risks: duration models with several destination processes competing against each other.
- Timing of Events model in Stata
- Timing of events model to evaluate treatment effects on duration processes

- Allows to model unobserved heterogeneity Idenfication: proportional hazard and no-anticipation assumptions.
- Typical application: Evaluation of Active Labor Market Programs (ALMP). Unemployed are at risk of participating to different treatments. Participation to treatment is not random. They can possibly transit to different destinations, i.e. programs.

The model in a nutshell

- competing risks: duration models with several destination processes competing against each other.
- Timing of Events model in Stata
- Timing of events model to evaluate treatment effects on duration processes
- Treatment effects are also modelled as duration process

- Allows to model unobserved heterogeneity
- Typical application: Evaluation of Active Labor Market Programs (ALMP). Unemployed are at risk of participating to different treatments. Participation to treatment is not random. They can possibly transit to different destinations, i.e. programs.

Model

Continous Time

- the hazard rate is equal to

$$\theta_j := \exp(x_j\beta_j + u_j)$$

- T_j is the duration of process j with d_j and indicator of failure for the same process.
- The contribution to the likelihood is equal to

$$\ell = \prod_{j=1}^{J} S_j\left(T_j|\mathbf{x},\mathbf{u};\theta_j\right)\cdot\theta_j\left(T_j|\mathbf{x},\mathbf{u};\theta_j\right)^{d_j} \qquad (1)$$

Model:Discrete Time

- Continous time model is generally an approximation of discrete time model. Duration data are discrete even with weekly data (transition can occur within a week).
- The continous time model should in theory be easier to estimate.
- The hazard rate is equal to

$$\theta_k := \exp(x_k \beta_k + u_k)$$

- Duration data are typically splitted

Model: Discrete Time

Likelihood

- Spell for individual i is divided in T subspells
- Let us define $d_{k,t}$ an indicator which takes value 1 if transition k occurs during subspell t (interval-censored data) and l_t the subspell's length.
- d_t is an indicator for whether a transition occured during subspell t. $d_t = (\sum_{k \in K} d_{k,t}) > 0$.
- We define the sum of the hazards in subspell t as $\theta_t = \sum_{k \in K} \theta_{k,t}$.
- Finally the contribution for an individual with several transitions is equal to

$$\ell_i = \prod_{t \in T} \left\{ \exp\left(-l_t \theta_t\right)^{1-d_t} \prod_{k \in K} \left[\left(1 - \exp\left(-l_t \theta\right)\right) \frac{\theta_{k,t}}{\theta_t} \right]^{d_{k,t}} \right\}$$

The NPMLE

- wrongly named non-parametric; rather a flexible parametric model
- Finite mixture model where unobserved heterogeneity is modelled as a discrete finite distribution.
- Another mixture formulation could be the use of a copula.

$$\begin{aligned}\ell &= \ln \sum_{j=1}^{J} p_j f\left(y|x;\theta^{(j)}\right)\\ &= \ln \sum_{j=1}^{J} \exp\left\{\ln p_j + \ln f\left(y|x;\theta^{(j)}\right)\right\}\end{aligned}$$

Direct maximization

- Given a fixed number of heterogeneity points. Mazimize in two-(or three) steps
- First maximize with respect to the heterogeneity mass-points
- Then use this solution as initial values when you try to maximize the likelihood with respect to the whole set of parameters.
- The program computes the gradient and the Hessian analytically. Makes it faster and improves the numerical stability of the model.

Direct maximization: choice of algorithm

- Combination of BFGS and Newton-Raphson
- Switching between algorithms can be effective (but not always) in getting out of a situation where the optimizer gets stuck.
- Stata's version of Newton-Raphson (NR) is quite effective, but it requires to compute the Hessian which can be costly depending on the scale of the problem.
- BFGS is less costly since in computes an approximation of the Hessian based on the gradient, but it is slower in finding a solution, i.e. you need more iteration. But still it can be faster in finding the solution.
- You may use the BHHH/Fisher scoring instead of NR (based on gradient hence less costly). BHHH uses the outer-product of the gradient. To be combined with BFGS.

Finding new heterogeneity mass-points

- Find mass-points which will likely give an improvement in the likelihood
- Simulated Annealing to find a positive Gateaux derivative.

$$\frac{LL\left[\theta^1; (p(1-\rho), \rho)\right] - LL(\theta^0; p)}{\rho} > 0$$

- Simulated annealing: derivative free method to find global optimum of a function or at least a reasonably close solution at a non-prohibitive cost. Slow but robust (or robust but slow).
- Heckman and Singer (1984) in the single transition case proposed to find a m.p. which maximizes the Gateaux derivative. Use grid search. Gaure et al. adivse against it.

When is it finished?

- Repeat the process of finding heterogeneity mass-points until no further improvement in the likelihood.
- Add heterogeneity points one at a time. Otherwise you end up with numerical problems.
- A popular formulation is to estimate n points for each transition and estimate the probability of each combination of m.p. It is fine with 2 heterogeneity points (still challenging though...), but with 3 heterogeneity points and 2 transitions you have to estimate 8 probabilities.

Estimation problems and possible solutions

- Large (negative) values for the mass-points. Solution: treat these parameters as constants during maximization.
- Defect (very small) hazards. Problem occurs when number of points becomes large (7). Risk set is set to zero for these observations.
- Small probabilities of the heterogeneity mass-points ($\leq$ 0.000001 f.e.). Solution: average these points with the next adjacent point.

Estimation problems and possible solutions

- Numerical problems can occur when evaluating $1 - exp(-x)$. I have written a function to solve this problem. We need a function for $\log(1 + x)$ as well. In the C-standard library these are called `expm1()` and `log1p()`. They don't exist in Mata.
- There are a few tricks to make the likelihood numerically more stable (`logSumOfExp()`).
- The likelihood can have regions with (many) local optima which makes it almost look as if it is flat. Obviously it is a problem with quasi-Newton methods.
- One problem with the Newton-Raphson is that the step length may be too long giving you absurd paramaters. Use Trust-region method to limit the step length. Not officially implemented in Stata-Mata.

What can we do with the command (in theory)

- Estimate the full model with any number of transitions and a number of m.p . which maximizes the likelihood function.
- Direct maximization given a number of points of heterogeneity.
- We can also estimate a variant of the model where we fix the number of m.p. and estimate probabilities associated to each combination of m.p. across processes.
- Mixed proportional hazard (single transition)
- Model with no unobserved heterogeneity (degenerate). Gives actually the initial values when finding the parameters for 2 mass-points.

Data set-up

```
. list id t transType d1 d2 exit treat in 1/15 , sepby(id)

     +-----------------------------------------------+
     |  id    t   transT~e   d1   d2   exit   treat |
     |-----------------------------------------------|
  1. |   1    5          0    0    0      0       0 |
  2. |   1    6          2    0    0      0       1 |
  3. |   1    7          0    1    0      0       . |
  4. |   1    8          1    1    0      1       . |
     |-----------------------------------------------|
  5. |   2   25          0    0    0      0       0 |
  6. |   2   26          1    0    0      1       0 |
     |-----------------------------------------------|
  7. |   3    1          2    0    0      0       1 |
  8. |   3    6          0    1    0      0       . |
  9. |   3   13          0    0    1      0       0 |
 10. |   3   14          1    0    1      1       0 |
     |-----------------------------------------------|
 11. |   4    2          0    0    0      0       0 |
 12. |   4    3          2    0    0      0       1 |
 13. |   4    8          0    1    0      0       . |
 14. |   4    9          2    0    1      0       1 |
 15. |   4   14          0    1    0      0       . |
```

The syntax of the command I

```
stcrmix
( depvar = [ indepvars ] )
( depvar = [ indepvars ] ) ...
[ if ] , time(varname) ident(varname) [ np(numlist)
trace(string) from(string) technique(string) first fullmax
model(string) direct maxiter(integer 200) uval(numlist
min=2 max=2) ]
```

Note: Options for modelling the baseline hazards. You can specify step-wise baseline hazards to avoid the splitting of the sample in order to gain speed.–> Only gradient-based. Consider working on other approximation of time-dependencies such as splines.

The syntax of the command II

```
stcrmix (exit = d1 d2 x1 x2) (treat = x1 x2) , id(id)
time(time) evaltype(gf2) method(trust) technique(bfgs 60
nr 10) fullmax np(1 10) maxiter(300)
```

Simulated data

- Data generating processes (DGP): interval censored data.
- Timing-of-events. Second process is a treatment. Once individuals transit to the second process they are in treatment. I assume that treatment has a positive effect of exiting the first process. The two processes compete against each other. If individuals transit to the first process, they leave the study.
- No post-treatment effect.
- Time to censoring is random.
- No time dependence.
- But unobserved heterogeneity
- Solution after 10 points of heterogeneity.

Simulations

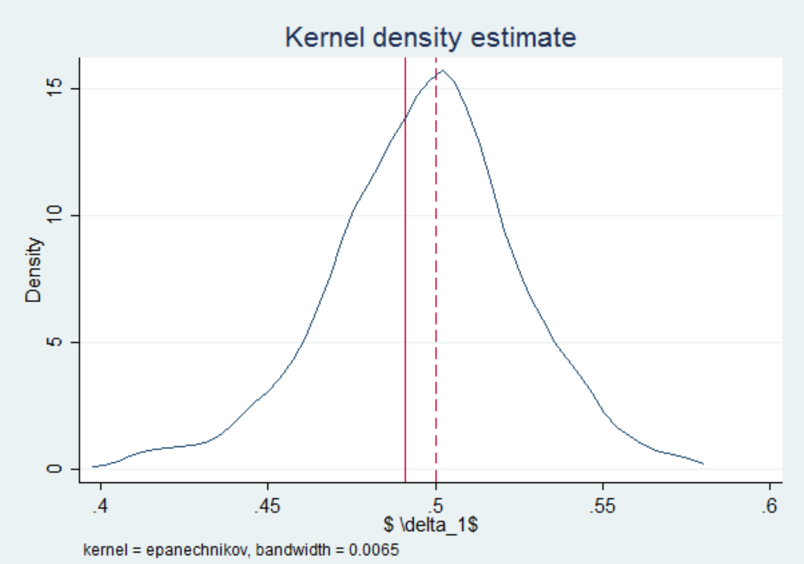
- # individuals : 50000
- Unobserved heterogeneity: bivariate normal (0,1) with $\rho = -0.25$
- Two covariates: 1 normal variate and one dummy.
- parameters for the Monte-Carlo simulations $\delta_1 = 0.5$, $b_1 = (1, -1), b_2 = (-1, 1)$
- 500 samples

Some results: average og estimated paramaters

	Average	s.e.	low	high
δ	.4966	.001262	.4942	.4991
$x_{n,1}$	1.009	.0005229	1.008	1.01
$x_{d,1}$	-1.011	.0007952	-1.012	-1.009
$x_{n,2}$	-1.009	.0005612	-1.01	-1.007
$x_{d,2}$	1.009	.000858	1.007	1.01
n_{MP}	9.974	.007727	9.959	9.989
N observations	114788	9.811	114768	114807
N individuals	50000	0	50000	50000
Log-likelihood	-188374	20.58	-188415	-188334
Run-time in minutes	14.03	1.081	11.9	16.15
Observations	496			

- Estimate of ρ not computed

Some results: distribution of the estimates treatment effect



Further work

- Some work needed for the simulations to get results from NPMLE which takes less time.
- the number of covariates is an issue since it makes the model slower to estimate and thereby limit the number of processes you can estimate.
- In the context of ALMP we would like to evaluate many different employment programs.
- Some work necessary on how to compute the likelihood, notably the issue of defect risks.

Summary

- I have presented `stcrmix`: a Stata command which purpose is to estimate competing risk models with unobserved heterogeneity.
- I presented it in the context of the timing-of-events model.
- Have performed simulations. DGP is a TOE model. The model seems to estimate the parameters consistently.
- Need further work to get the full NPMLE for the 500 samples.