Fitting interval-censored Cox model with time-varying covariates in Stata

Xiao Yang

StataCorp LLC

2023 Canadian Stata Conference
August 4, 2023

Outline

What are interval-censored event-time data?

Brief introduction to the algorithm

stintcox's new features

- Using the tvc() option to create TVCs
- Testing the PH assumption using tvc()
- Fitting stintcox with multiple-record data
- Producing new postestimation graphs

References

Table of Contents

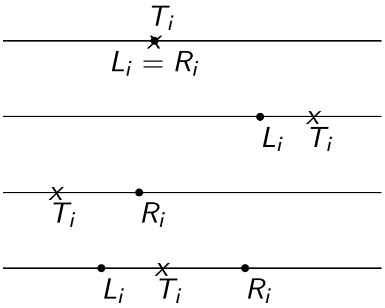
What are interval-censored event-time data?

- The event of interest is not always observed exactly but is known only to occur within some time interval. For example, cancer recurrence, time of COVID infection, etc.
- Interval-censored event-time data arise in many areas, including medical, epidemiological, economic, financial, and sociological studies.
- There are four types of censoring: left-censoring, right-censoring, interval-censoring, and no censoring.
- Data are usually stored in two formats.
- Ignoring interval-censoring may lead to biased estimates.

Types of censoring

For each subject i, event time T_i is not always exactly observed.
$(L_i, R_i]$ denotes the interval in which T_i is observed.

No censoring
$L_i = R_i = T_i$
Right-censoring
$(L_i, R_i = +\infty)$
Left-censoring
$(L_i = 0, R_i]$
Interval-censoring
$(L_i, R_i]$



Data formats

Single-record-per-subject (single-record) format:

- contains one record for a subject
- contains lower and upper endpoints of the event-time interval
- censoring type is determined by the event-time interval
- covariates are time-independent

	id	ltime	rtime	x1	x2	x3
1.	101	0	6	17	22	0
2.	102	4	9	12	22	1
3.	103	13	.	13	22	0

Data formats

Multiple-record-per-subject (multiple-record) format:

- typically contains multiple records for a subject
- contains an examination time and an event status for each record
- censoring type and the event-time interval can be determined by the examination time and event status
- easily records time-varying covariates

	id	time	status	x1	x2	x3
1.	101	6	1	17	22	0
2.	102	4	0	12	22	1
3.	102	6	0	12	22	0
4.	102	9	1	12	22	1
5.	103	13	0	13	22	0

Methods for analyzing interval-censored data

- Simple imputation methods
- Nonparametric maximum-likelihood estimation
- Parametric regression models – `stintreg`
- **Semiparametric Cox proportional hazards model –** `stintcox`

Table of Contents

What is Cox proportional hazards model?

- The Cox proportional hazards model was first introduced by Cox in 1972 and was used routinely to analyze uncensored and right-censored event-time data.

$$h(t; \mathbf{x}) = h_0(t)\exp(\mathbf{x}'\beta)$$

- It does not require parameterization of the baseline hazard function.
- Also, under the proportional-hazard assumption, the hazard ratios are constant over time.

$$\frac{h(t; \mathbf{x_i})}{h(t; \mathbf{x_j})} = \frac{h_0(t)\exp(\mathbf{x_i}'\beta)}{h_0(t)\exp(\mathbf{x_j}'\beta)} = \exp(\mathbf{x_i} - \mathbf{x_j})'\beta$$

Cox model's challenge for interval-censored data

- Cox model is challenging for interval-censored event-time data because none of the event times are observed exactly. In particular, the traditional partial-likelihood approach is not applicable.
- Several authors have proposed spline methods to fit the Cox model to interval-censored data and those method have their limitations.
- The direct maximum-likelihood optimization using the Newton-Raphson algorithm is highly unstable.
- Zeng et al. (2016) developed a genuine EM algorithm for efficient nonparametric maximum-likelihood estimation (NPMLE) method to fit the Cox model for interval-censored data.

A genuine model for stintcox

- Suppose that the observed data consist of $(t_{li}, t_{ui}, \mathbf{x}_i)$ for $i = 1, \ldots, n$, where t_{li} and t_{ui} define the observed time interval and $\mathbf{x}_i$ records covariate values for a subject i.
- Under the NPMLE approach, the baseline cumulative hazard function H_0 is regarded as a step function with nonnegative jumps $h_1, \ldots, h_m$ at $t_1, \ldots, t_m$, respectively, where $t_1 < \cdots < t_m$ are the distinct time points for all $t_{li} > 0$ and $t_{ui} < \infty$ for $i = 1, \ldots, n$.
- The observed-data likelihood function is

$$\prod_{i=1}^{n} \exp\left\{-\sum_{t_k \le t_{li}} h_k \exp(\mathbf{x}_i \beta)\right\} \left[1 - \exp\left\{-\sum_{t_{li} < t_k \le t_{ui}} h_k \exp(\mathbf{x}_i \beta)\right\}\right]^{I(t_{ui} < \infty)} \tag{1}$$

A genuine model for stintcox (cont.)

- Let W_{ik} $(i = 1, \ldots, n; k = 1, \ldots, m)$ be independent latent Poisson random variables with means $h_k \exp(\mathbf{x}_i \beta)$. Define $A_i = \sum_{t_k \le t_{li}} W_{ik}$ and $B_i = I(t_{ui} < \infty) \sum_{t_{li} < t_k \le t_{ui}} W_{ik}$. The likelihood for the observed data $(t_{li}, t_{ui}, \mathbf{x}_i, A_i = 0, B_i > 0)$ is

$$\prod_{i=1}^{n} \prod_{t_k \le t_{li}} \Pr(W_{ik} = 0) \Big\{ 1 - \Pr\Big(\sum_{t_{li} < t_k \le t_{ui}} W_{ik} = 0 \Big) \Big\}^{I(t_{ui} < \infty)} \qquad (2)$$

- (1) and (2) are exactly equal. The maximization of a weighted sum of Poisson log-likelihood functions is strictly concave and has a closed-form solution for h_k's.

A genuine model for stintcox (cont.)

- We maximize (2) through an EM algorithm treating W_{ik} as missing data.
  - In the E-step, we evaluate the posterior means of W_{ik}.
  - In the M-step, we update $\boldsymbol{\beta}$ and h_k for $k = 1, ..., m$.
- This method allows a completely arbitrary baseline hazard function, and the results are consistent, asymptotically normal, and asymptotically efficient.

Table of Contents

stintcox's highlights

Stata 17 introduced the stintcox command for fitting a semiparametric Cox model to single-record interval-censored data.

- Provides four methods for standard-error computation.
- Provides standard-error computation on replay.
- Provides options to control the tradeoff between the execution speed and accuracy of the results.
- Supports two ways to choose the time intervals to be estimated for baseline hazard contributions.
- Supports stratification.
- Supports various postestimation features after fitting stintcox

stintcox's new features

Stata 18 extended the functionality of `stintcox` command:

- Fits multiple-record formats
- Supports time-varying covariates (TVCs):
  - created automatically as deterministic functions of time using the `tvc()` option
  - use the `tvc()` option to test the proportional-hazards assumption
  - Supplied directly in a multiple-record data format
- Supports robust and cluster standard-error computation
- Produces goodness-of-fit plots
- Provides predictions with TVCs
- Plots functions with TVCs

Basic syntax

Single-record-per-subject data format

```
. stintcox [<indepvars>], interval(t_l t_u) ...
```

Multiple-record-per-subject data format

```
. stintcox [<indepvars>], id() time() status() ...
```

- st setting the data is not necessary and will be ignored.
- *indepvars* is optional. You can fit a Cox model without any covariates.

Motivating example background

Modified Bangkok IDU Preparatory Study

It is a cohort study of injecting drug users in Thailand.

- 1124 subjects were initially negative for HIV-1 virus.
- They were followed and tested for HIV approximately every four months.
- The event of interest was time to HIV-1 seropositivity.
- We want to identify the factors that influence time to HIV infection.
- Data are stored in both formats:
  - single-record dataset contains all baseline covariates;
  - multiple-record dataset contains both baseline covariates as well as time-varying covariates.

Single-record-per-subject data

```
. list id ltime rtime age_mean male needle inject jail ///
> if id >= 271 & id <= 274, noobs
```

id	ltime	rtime	age_mean	male	needle	inject	jail
271	22.00	.	-6.46	Yes	Yes	No	No
272	3.80	9.41	8.54	No	No	No	Yes
273	20.66	.	-11.46	Yes	Yes	No	No
274	0.00	3.87	-4.46	Yes	Yes	Yes	Yes

Multiple-record-per-subject data

```
. list id time is_seropos age_mean male needle inject jail_vary ///
> if id >= 271 & id <=274, sepby(id) noobs abbreviate(10) compress
```

id	time	is_seropos	age_mean	male	needle	inject	jail_vary
271	4.89	No	-6.46	Yes	Yes	No	No
271	9.31	No	-6.46	Yes	Yes	No	No
271	13.38	No	-6.46	Yes	Yes	No	Yes
271	17.97	No	-6.46	Yes	Yes	No	Yes
271	22.00	No	-6.46	Yes	Yes	No	No
272	3.80	No	8.54	No	No	No	Yes
272	9.41	Yes	8.54	No	No	No	No
273	3.93	No	-11.46	Yes	Yes	No	No
273	8.00	No	-11.46	Yes	Yes	No	No
273	12.07	No	-11.46	Yes	Yes	No	Yes
273	15.97	No	-11.46	Yes	Yes	No	Yes
273	20.66	No	-11.46	Yes	Yes	No	Yes
274	3.87	Yes	-4.46	Yes	Yes	Yes	Yes

Fitting stintcox with single-record data

First, we fit a Cox model with time-independent covariates using the single-record data.

```
. stintcox age_mean i.male i.needle i.inject i.jail, interval(ltime rtime)
note: using adaptive step size to compute derivatives.
```

(iteration output omitted)

```
Interval-censored Cox regression                    Number of obs     =  1,124
Baseline hazard: Reduced intervals                         Uncensored =      0
                                                        Left-censored =     41
Event-time interval:                                   Right-censored =    991
  Lower endpoint: ltime                                Interval-cens. =     92
  Upper endpoint: rtime
                                                    Wald chi2(5)      =  17.10
Log likelihood = -597.56443                         Prob > chi2       = 0.0043
--more--
```

Fitting stintcox with single-record data (cont.)

	Haz. ratio	OPG std. err.	z	P>|z|	[95% conf.	interval]
age_mean	.9684341	.0126552	-2.45	0.014	.9439452	.9935582
male						
Yes	.6846949	.1855907	-1.40	0.162	.4025073	1.164717
needle						
Yes	1.275912	.2279038	1.36	0.173	.8990401	1.810768
inject						
Yes	1.250154	.2414221	1.16	0.248	.8562184	1.825334
jail						
Yes	1.567244	.3473972	2.03	0.043	1.014982	2.419998

Note: Standard error estimates may be more variable for small datasets and datasets with low proportions of interval-censored observations.

Using the tvc() option

- tvc() specifies the variables to be included in the model as an interaction with a function of time to form time-varying covariates.
- It is a convenience tool to speed up calculations and avoid splitting the data over many analysis times.
- Option texp() is used in conjunction with tvc() to specify the function of time that multiplies covariates specified in the tvc() option, i.e., texp(log(_t)).
- Option lrphtest is used in conjunction with tvc() to performs the likelihood-ratio test between the full model and the model without specifying option tvc().
- tvc() is also useful for testing the proportional-hazards (PH) assumption.

Testing the PH assumption using tvc()

- One way of testing the PH assumption for a covariate (say, x_1) is to test whether the coefficient associated with that covariate is time invariant.
- This can be accomplished by including an interaction between this covariate and a function of time ($g(t)$) in the model and testing whether the corresponding coefficient equals zero ($\gamma_1 = 0$).

$$
\begin{aligned}
h(t) &= h_0(t)\exp\{\beta_1 x_1 + \gamma_1 g(t) x_1\} \\
&= h_0(t)\exp\left[\{\beta_1 + \gamma_1 g(t)\} x_1\right]
\end{aligned}
$$

Example: testing the PH assumption

We now include all covariates in option tvc() to additionally include their interactions with the analysis time in the model. Thus we can test the PH assumption individually and globally:

```
. stintcox age_mean i.male i.needle i.inject i.jail, interval(ltime rtime) ///
> tvc(age_mean i.male i.needle i.inject i.jail) nohr
note: using adaptive step size to compute derivatives.
```

(iteration output omitted)

```
Interval-censored Cox regression                    Number of obs      =  1,124
Baseline hazard: Reduced intervals                         Uncensored =      0
                                                        Left-censored =     41
Event-time interval:                                   Right-censored =    991
  Lower endpoint: ltime                                Interval-cens. =     92
  Upper endpoint: rtime
                                                    Wald chi2(10)      =  31.99
Log likelihood = -590.43386                         Prob > chi2        = 0.0004
--more--
```

Example: testing the PH assumption (cont.)

	Coefficient	OPG std. err.	z	P>|z|	[95% conf.	interval]
main						
age_mean	-.0310177	.0233817	-1.33	0.185	-.076845	.0148097
male						
Yes	-1.271583	.4604788	-2.76	0.006	-2.174105	-.3690615
needle						
Yes	-.1819587	.3297493	-0.55	0.581	-.8282554	.464338
inject						
Yes	.6852961	.3431924	2.00	0.046	.0126513	1.357941
jail						
Yes	-.529615	.4021087	-1.32	0.188	-1.317734	.2585036

--more--

Example: testing the PH assumption (cont.)

tvc						
age_mean	-.000129	.0017099	-0.08	0.940	-.0034804	.0032224
male						
Yes	.0884102	.042994	2.06	0.040	.0041434	.1726769
needle						
Yes	.0358545	.0238562	1.50	0.133	-.0109027	.0826118
inject						
Yes	-.0361192	.0228754	-1.58	0.114	-.0809541	.0087157
jail						
Yes	.0916036	.0348915	2.63	0.009	.0232176	.1599896

```
Notes: Standard error estimates may be more variable for small datasets and
       datasets with low proportions of interval-censored observations.
       Variables in tvc equation interacted with _t.
Wald test that [tvc] = 0: chi2(5) = 13.3282             Prob > chi2 = 0.0205
```

Fitting stintcox with multiple-record data

Fit a Cox model using multiple-record data, including the time-varying covariate jail_vary

```
. stintcox age_mean i.male i.needle i.inject i.jail_vary, id(id) time(time) ///
> status(is_seropos)
note: time-varying covariates detected in the data; using method nearleft to
      impute their values between examination times.
note: using adaptive step size to compute derivatives.
```

(iteration output omitted)

```
Interval-censored Cox regression                     Number of obs      =  6,453
Baseline hazard: Reduced intervals                   Number of subjects =  1,124
                                                             Uncensored =      0
ID variable: id                                           Left-censored =     41
Examination time: time                                   Right-censored =    991
Status indicator: is_seropos                             Interval-cens. =     92

                                                     Wald chi2(5)       =  17.03
Log likelihood = -598.34887                          Prob > chi2        = 0.0044
--more--
```

Fitting stintcox with multiple-record data (cont.)

time	Haz. ratio	OPG std. err.	z	P>|z|	[95% conf.	interval]
age_mean	.9714605	.012757	-2.20	0.027	.9467762	.9967884
male						
Yes	.6678044	.1816576	-1.48	0.138	.3918353	1.138138
needle						
Yes	1.271409	.2275426	1.34	0.180	.8952546	1.805609
inject						
Yes	1.370672	.2575405	1.68	0.093	.9484142	1.980928
jail_vary						
Yes	1.440966	.2916178	1.81	0.071	.9691488	2.142481

```
Time varying: jail_vary
Note: Standard error estimates may be more variable for small datasets and
      datasets with low proportions of interval-censored observations.
```

Using tvcovimpute() option

- Use `tvcovimpute()` to specify how to impute unobserved covariate values between two examination times for time-varying covariates.
- The imputation methods include `nearleft` (default), `nearright`, `nearest`, or `first`.

```
. stintcox age_mean i.male i.needle i.inject i.jail_vary, id(id) time(time) ///
> status(is_seropos) tvcovimpute(nearright)
note: time-varying covariates detected in the data; using method nearright to
      impute their values between examination times.
note: using adaptive step size to compute derivatives.
```

(iteration output omitted)

```
Interval-censored Cox regression                    Number of obs      =  6,453
Baseline hazard: Reduced intervals                  Number of subjects =  1,124
                                                            Uncensored =      0
ID variable: id                                          Left-censored =     41
Examination time: time                                  Right-censored =    991
Status indicator: is_seropos                            Interval-cens. =     92

                                                    Wald chi2(5)       =  18.41
Log likelihood = -597.00103                         Prob > chi2        = 0.0025
```

Using tvcovimpute() option (cont.)

time	Haz. ratio	OPG std. err.	z	P>|z|	[95% conf.	interval]
age_mean	.9726438	.0126471	-2.13	0.033	.9481692	.9977502
male						
Yes	.6561992	.1780502	-1.55	0.121	.3855444	1.116856
needle						
Yes	1.267405	.228118	1.32	0.188	.890654	1.803523
inject						
Yes	1.367475	.252569	1.69	0.090	.9521488	1.963966
jail_vary						
Yes	1.640746	.3346384	2.43	0.015	1.100106	2.44708

```
Time varying: jail_vary
Note: Standard error estimates may be more variable for small datasets and
      datasets with low proportions of interval-censored observations.
```

Postestimation features after stintcox

stintcox provides several postestimation features after estimation:

- Predictions of hazard ratios, linear predictions, and standard errors with support for TVCs
- Predictions of baseline survivor, baseline cumulative hazard, and baseline hazard contribution functions
- Prediction of martingale-like residuals and Cox–Snell-like residuals
- goodness-of-fit plot
- Plots for survivor, hazard, and cumulative hazard functions

Producing Goodness-of-fit (GOF) plot

- `estat gofplot` is used to assess the goodness of fit of the model visually.
- It plots the Cox–Snell-like residuals versus the estimated cumulative hazard function corresponding to these residuals.
- The estimated cumulative hazards are calculated using the self-consistency algorithm proposed by Turnbull (1976).
- The Cox–Snell-like residuals form the 45° reference line. If the model fits the data well, the plotted estimated cumulative hazards should be close to the reference line.

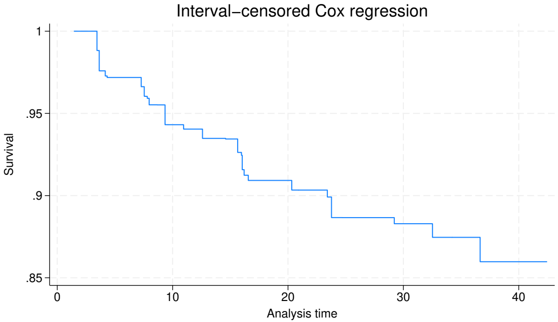
Goodness-of-fit (GOF) plot

```
. estat gofplot
```

Graph survivor function

- Use stcurve to plot the estimated survivor function.
- By default, stcurve evaluates the functions at the overall means of covariates.

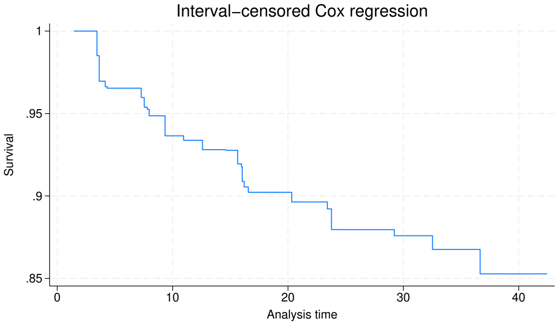
```
. stcurve, survival
note: function evaluated at overall means of covariates.
```



Graph survivor function with TVCs

- Use option attmeans to evaluate the function at time-specific means.

```
. stcurve, survival attmeans
note: function evaluated at time-specific means of covariates.
```



Graph survivor function using frame

We can also use option `atframe()` to specify your own TVC values to be used to evaluate the survivor function.

- Suppose we want to plot the survivor curve for an individual with the same covariate pattern as subject 2.
- We create a new frame called `id2` and use `frame put` to copy the relevant information to the new frame.
- We `list` the data in frame `id2`.

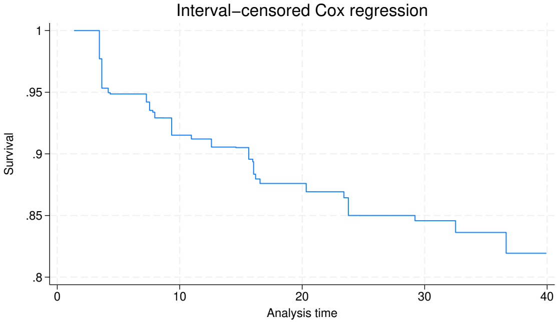
```
. frame put time age_mean male needle inject jail_vary if id==2, into(id2)
. frame id2: list
```

	time	age_mean	male	needle	inject	jail_vary
1.	4.13	-6.46	Yes	No	Yes	Yes
2.	8.26	-6.46	Yes	No	Yes	No
3.	12.30	-6.46	Yes	No	Yes	No
4.	16.07	-6.46	Yes	No	Yes	No
5.	20.10	-6.46	Yes	No	Yes	No
6.	24.26	-6.46	Yes	No	Yes	No

Graph survivor function using frame (cont.)

- Use option atframe() to graph the survivor curve for this particular profile,

```
. stcurve, survival atframe(id2)
note: function evaluated at specified values of selected covariates and
      overall means of other covariates (if any).
note: covariate values from frame id2 used to evaluate function.
```



Conclusions for stintcox

- Fits a genuine semiparametric Cox proportional hazards model with two formats of interval-censored data.
- Supports different methods for standard error computation; also support VCE computation on replay.
- Suppors creating TVCs automatically and testing the PH assumption.
- Provides diagnostic measures, predictions, and much more after fitting the model.
- Provides convenient graphical tools for assessing the goodness of fit of the model, and for plotting the survivor, cumulative hazards, and hazard functions.
- Supports TVCs with predictions and graphs.

Table of Contents

References

Farrington, C. P. (2000). Residuals for proportional hazards models with interval-censored survival data. *Biometrics 56*, 473–482.

Turnbull, B. W. (1976). The empirical distribution function with arbitrarily grouped censored and truncated data. *Journal of the Royal Statistical Society, Series B 38*, 290–295.

Zeng, D., F. Gao, and D. Lin (2017). Maximum likelihood estimation for semiparametric regression models with multivariate interval-censored data. *Biometrika 104*, 505–525.

Zeng, D., L. Mao, and D. Lin (2016). Maximum likelihood estimation for semiparametric transformation models with interval-censored data. *Biometrika 103*, 253–271.

More resources

https://www.stata.com/manuals/ststintcox.pdf
https://www.stata.com/manuals/ststintcoxpostestimation.pdf
https://www.stata.com/manuals/ststintcoxph-assumptionplots.pdf
https://www.stata.com/manuals/stestatgofplot.pdf
https://www.stata.com/manuals/ststcurve.pdf

Thank you!