Title

stata.com

stcox — Cox proportional hazards model

Syntax

```
stcox [varlist] [if] [in] [, options]
```

options	Description
Model	
`estimate`	fit model without covariates
`strata(`*varnames*`)`	strata ID variables
`shared(`*varname*`)`	shared-frailty ID variable
`offset(`*varname*`)`	include *varname* in model with coefficient constrained to 1
`breslow`	use Breslow method to handle tied failures; the default
`efron`	use Efron method to handle tied failures
`exactm`	use exact marginal-likelihood method to handle tied failures
`exactp`	use exact partial-likelihood method to handle tied failures
Time varying	
`tvc(`*varlist*`)`	time-varying covariates
`texp(`*exp*`)`	multiplier for time-varying covariates; default is `texp(_t)`
SE/Robust	
`vce(`*vcetype*`)`	*vcetype* may be `oim`, `robust`, `cluster` *clustvar*, `bootstrap`, or `jackknife`
`noadjust`	do not use standard degree-of-freedom adjustment
Reporting	
`level(#)`	set confidence level; default is `level(95)`
`nohr`	report coefficients, not hazard ratios
`noshow`	do not show st setting information
display_options	control column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Maximization	
maximize_options	control the maximization process; seldom used
`coeflegend`	display legend instead of statistics

You must `stset` your data before using `stcox`; see [ST] **stset**.

varlist may contain factor variables; see [U] **11.4.3 Factor variables**.

`bootstrap`, `by`, `fp`, `jackknife`, `mfp`, `mi estimate`, `nestreg`, `statsby`, `stepwise`, and `svy` are allowed; see [U] **11.1.10 Prefix commands**.

`vce(bootstrap)` and `vce(jackknife)` are not allowed with the `mi estimate` prefix; see [MI] **mi estimate**.

`estimate`, `shared()`, `efron`, `exactm`, `exactp`, `tvc()`, `texp()`, `vce()`, and `noadjust` are not allowed with the `svy` prefix; see [SVY] **svy**.

`fweights`, `iweights`, and `pweights` may be specified using `stset`; see [ST] **stset**. Weights are not supported with `efron` and `exactp`. Also weights may not be specified if you are using the `bootstrap` prefix with the `stcox` command.

`coeflegend` does not appear in the dialog box.

See [U] **20 Estimation and postestimation commands** for more capabilities of estimation commands.

Menu

Statistics > Survival analysis > Regression models > Cox proportional hazards model

Description

`stcox` fits, via maximum likelihood, proportional hazards models on st data. `stcox` can be used with single- or multiple-record or single- or multiple-failure st data.

Options

Model

`estimate` forces fitting of the null model. All Stata estimation commands redisplay results when the command name is typed without arguments. So does `stcox`. What if you wish to fit a Cox model on $\mathbf{x}_j\boldsymbol{\beta}$, where $\mathbf{x}_j\boldsymbol{\beta}$ is defined as 0? Logic says that you would type `stcox`. There are no explanatory variables, so there is nothing to type after the command. Unfortunately, this looks the same as `stcox` typed without arguments, which is a request to redisplay results.

To fit the null model, type `stcox, estimate`.

`strata(`*varnames*`)` specifies up to five strata variables. Observations with equal values of the strata variables are assumed to be in the same stratum. Stratified estimates (equal coefficients across strata but with a baseline hazard unique to each stratum) are then obtained.

`shared(`*varname*`)` specifies that a Cox model with shared frailty be fit. Observations with equal value of *varname* are assumed to have shared (the same) frailty. Across groups, the frailties are assumed to be gamma-distributed latent random effects that affect the hazard multiplicatively, or, equivalently, the logarithm of the frailty enters the linear predictor as a random offset. Think of a shared-frailty model as a Cox model for panel data. *varname* is a variable in the data that identifies the groups. `shared()` is not allowed in the presence of delayed entries or gaps.

Shared-frailty models are discussed more in *Cox regression with shared frailty*.

`offset(`*varname*`)`; see [R] **estimation options**.

`breslow`, `efron`, `exactm`, and `exactp` specify the method for handling tied failures in the calculation of the log partial likelihood (and residuals). `breslow` is the default. Each method is described in *Treatment of tied failure times*. `efron` and the exact methods require substantially more computer time than the default `breslow` option. `exactm` and `exactp` may not be specified with `tvc()`, `vce(robust)`, or `vce(cluster` *clustvar*`)`.

Time varying

`tvc(`*varlist*`)` specifies those variables that vary continuously with respect to time, that is, time-varying covariates. This is a convenience option used to speed up calculations and to avoid having to `stsplit` (see [ST] **stsplit**) the data over many failure times.

`texp(`*exp*`)` is used in conjunction with `tvc(`*varlist*`)` to specify the function of analysis time that should be multiplied by the time-varying covariates. For example, specifying `texp(ln(_t))` would cause the time-varying covariates to be multiplied by the logarithm of analysis time. If `tvc(`*varlist*`)` is used without `texp(`*exp*`)`, Stata understands that you mean `texp(_t)` and thus multiplies the time-varying covariates by the analysis time.

Both `tvc(`*varlist*`)` and `texp(`*exp*`)` are explained more in the section on *Cox regression with continuous time-varying covariates* below.

SE/Robust

`vce(`*vcetype*`)` specifies the type of standard error reported, which includes types that are derived from asymptotic theory (`oim`), that are robust to some kinds of misspecification (`robust`), that allow for intragroup correlation (`cluster` *clustvar*), and that use bootstrap or jackknife methods (`bootstrap`, `jackknife`); see [R] ***vce_option***.

`noadjust` is for use with `vce(robust)` or `vce(cluster` *clustvar*`)`. `noadjust` prevents the estimated variance matrix from being multiplied by $N/(N-1)$ or $g/(g-1)$, where g is the number of clusters. The default adjustment is somewhat arbitrary because it is not always clear how to count observations or clusters. In such cases, however, the adjustment is likely to be biased toward 1, so we would still recommend making it.

Reporting

`level(#)`; see [R] **estimation options**.

`nohr` specifies that coefficients be displayed rather than exponentiated coefficients or hazard ratios. This option affects only how results are displayed and not how they are estimated. `nohr` may be specified at estimation time or when redisplaying previously estimated results (which you do by typing `stcox` without a variable list).

`noshow` prevents `stcox` from showing the key st variables. This option is seldom used because most people type `stset, show` or `stset, noshow` to set whether they want to see these variables mentioned at the top of the output of every st command; see [ST] **stset**.

display_options: `noomitted`, `vsquish`, `noemptycells`, `baselevels`, `allbaselevels`, `nofvlabel`, `fvwrap(#)`, `fvwrapon(`*style*`)`, `cformat(%`*fmt*`)`, `pformat(%`*fmt*`)`, `sformat(%`*fmt*`)`, and `nolstretch`; see [R] **estimation options**.

Maximization

maximize_options: `iterate(#)`, `[no]log`, `trace`, `tolerance(#)`, `ltolerance(#)`, `nrtolerance(#)`, and `nonrtolerance`; see [R] **maximize**. These options are seldom used.

The following option is available with `stcox` but is not shown in the dialog box:

`coeflegend`; see [R] **estimation options**.

Remarks and examples

stata.com

Remarks are presented under the following headings:

What follows is a summary of what can be done with `stcox`. For a complete tutorial, see Cleves et al. (2010), which devotes three chapters to this topic.

In the Cox proportional hazards model (Cox 1972), the hazard is assumed to be

$$h(t) = h_0(t)\exp(\beta_1 x_1 + \cdots + \beta_k x_k)$$

The Cox model provides estimates of $\beta_1, \ldots, \beta_k$ but provides no direct estimate of $h_0(t)$—the baseline hazard. Formally, the function $h_0(t)$ is not directly estimated, but it is possible to recover an estimate of the cumulative hazard $H_0(t)$ and, from that, an estimate of the baseline survivor function $S_0(t)$.

`stcox` fits the Cox proportional hazards model; that is, it provides estimates of $\boldsymbol{\beta}$ and its variance–covariance matrix. Estimates of $H_0(t)$, $S_0(t)$, and other predictions and diagnostics are obtained with `predict` after `stcox`; see [ST] **stcox postestimation**. For information on fitting a Cox model to survey data, see Cleves et al. (2010, sec. 9.5), and for information on handling missing data, see Cleves et al. (2010, sec. 9.6).

`stcox` with the `strata()` option will produce stratified Cox regression estimates. In the stratified estimator, the hazard at time t for a subject in group i is assumed to be

$$h_i(t) = h_{0i}(t)\exp(\beta_1 x_1 + \cdots + \beta_k x_k)$$

That is, the coefficients are assumed to be the same, regardless of group, but the baseline hazard can be group specific.

Regardless of whether you specify `strata()`, the default variance estimate is to calculate the conventional, inverse matrix of negative second derivatives. The theoretical justification for this estimator is based on likelihood theory. The `vce(robust)` option instead switches to the robust measure developed by Lin and Wei (1989). This variance estimator is a variant of the estimator discussed in [U] **20.21 Obtaining robust variance estimates**.

`stcox` with the `shared()` option fits a Cox model with shared frailty. A *frailty* is a group-specific latent random effect that multiplies into the hazard function. The distribution of the frailties is gamma with mean 1 and variance to be estimated from the data. Shared-frailty models are used to model within-group correlation. Observations within a group are correlated because they share the same frailty.

We give examples below with uncensored, censored, time-varying, and recurring failure data, but it does not matter in terms of what you type. Once you have `stset` your data, to fit a model you type `stcox` followed by the names of the explanatory variables. You do this whether your dataset has single or multiple records, includes censored observations or delayed entry, or even has single or multiple failures. You use `stset` to describe the properties of the data, and then that information is available to `stcox`—and all the other st commands—so that you do not have to specify it again.

Cox regression with uncensored data

▷ Example 1

We wish to analyze an experiment testing the ability of emergency generators with a new-style bearing to withstand overloads. For this experiment, the overload protection circuit was disabled, and the generators were run overloaded until they burned up. Here are our data:

```
. use http://www.stata-press.com/data/r13/kva
(Generator experiment)
. list
```

	failtime	load	bearings
1.	100	15	0
2.	140	15	1
3.	97	20	0
4.	122	20	1
5.	84	25	0
6.	100	25	1
7.	54	30	0
8.	52	30	1
9.	40	35	0
10.	55	35	1
11.	22	40	0
12.	30	40	1

Twelve generators, half with the new-style bearings and half with the old, were allocated to this destructive test. The first observation reflects an old-style generator (`bearings` = 0) under a 15-kVA overload. It stopped functioning after 100 hours. The second generator had new-style bearings (`bearings` = 1) and, under the same overload condition, lasted 140 hours. Paired experiments were also performed under overloads of 20, 25, 30, 35, and 40 kVA.

We wish to fit a Cox proportional hazards model in which the failure rate depends on the amount of overload and the style of the bearings. That is, we assume that `bearings` and `load` do not affect the shape of the overall hazard function, but they do affect the relative risk of failure. To fit this model, we type

```
. stset failtime
  (output omitted)
. stcox load bearings
        failure _d:  1 (meaning all fail)
  analysis time _t:  failtime
Iteration 0:   log likelihood = -20.274897
Iteration 1:   log likelihood = -10.515114
Iteration 2:   log likelihood = -8.8700259
Iteration 3:   log likelihood = -8.5915211
Iteration 4:   log likelihood = -8.5778991
Iteration 5:   log likelihood =  -8.577853
Refining estimates:
Iteration 0:   log likelihood =  -8.577853
```

```
Cox regression -- Breslow method for ties
No. of subjects =           12                  Number of obs   =        12
No. of failures =           12
Time at risk    =          896
                                                LR chi2(2)      =     23.39
Log likelihood  =    -8.577853                  Prob > chi2     =    0.0000

------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        load |    1.52647   .2188172     2.95   0.003     1.152576    2.021653
    bearings |   .0636433   .0746609    -2.35   0.019     .0063855    .6343223
------------------------------------------------------------------------------
```

We find that after controlling for overload, the new-style bearings result in a lower hazard and therefore a longer survivor time.

Once an `stcox` model has been fit, typing `stcox` without arguments redisplays the previous results. Options that affect the display, such as `nohr`—which requests that coefficients rather than hazard ratios be displayed—can be specified upon estimation or when results are redisplayed:

```
. stcox, nohr
Cox regression -- Breslow method for ties
No. of subjects =           12                  Number of obs   =        12
No. of failures =           12
Time at risk    =          896
                                                LR chi2(2)      =     23.39
Log likelihood  =    -8.577853                  Prob > chi2     =    0.0000

------------------------------------------------------------------------------
          _t |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        load |   .4229578   .1433485     2.95   0.003     .1419999    .7039157
    bearings |  -2.754461   1.173115    -2.35   0.019    -5.053723   -.4551981
------------------------------------------------------------------------------
```

◁

❑ Technical note

`stcox`'s iteration log looks like a standard Stata iteration log up to where it says "Refining estimates". The Cox proportional-hazards likelihood function is indeed a difficult function, both conceptually and numerically. Until Stata says "Refining estimates", it maximizes the Cox likelihood in the standard way by using double-precision arithmetic. Then just to be sure that the answers are accurate, Stata switches to quad-precision routines (double double precision) and completes the maximization procedure from its current location on the likelihood.

❑

Cox regression with censored data

▷ Example 2

We have data on 48 participants in a cancer drug trial. Of these 48, 28 receive treatment (`drug` = 1) and 20 receive a placebo (`drug` = 0). The participants range in age from 47 to 67 years. We wish to analyze time until death, measured in months. Our data include 1 observation for each patient. The variable `studytime` records either the month of their death or the last month that they were known to be alive. Some of the patients still live, so together with `studytime` is `died`, indicating their health status. Persons known to have died—"noncensored" in the jargon—have `died` = 1, whereas the patients who are still alive—"right-censored" in the jargon—have `died` = 0.

Here is an overview of our data:

```
. use http://www.stata-press.com/data/r13/drugtr
(Patient Survival in Drug Trial)
. st
-> stset studytime, failure(died)
     failure event:  died != 0 & died < .
obs. time interval:  (0, studytime]
 exit on or before:  failure
. summarize
```

Variable	Obs	Mean	Std. Dev.	Min	Max
studytime	48	15.5	10.25629	1	39
died	48	.6458333	.4833211	0	1
drug	48	.5833333	.4982238	0	1
age	48	55.875	5.659205	47	67
_st	48	1	0	1	1
_d	48	.6458333	.4833211	0	1
_t	48	15.5	10.25629	1	39
_t0	48	0	0	0	0

We typed `stset studytime, failure(died)` previously; that is how `st` knew about this dataset. To fit the Cox model, we type

```
. stcox drug age
        failure _d:  died
  analysis time _t:  studytime
Iteration 0:   log likelihood = -99.911448
Iteration 1:   log likelihood = -83.551879
Iteration 2:   log likelihood = -83.324009
Iteration 3:   log likelihood = -83.323546
Refining estimates:
Iteration 0:   log likelihood = -83.323546
Cox regression -- Breslow method for ties
No. of subjects =           48                 Number of obs   =        48
No. of failures =           31
Time at risk    =          744
                                               LR chi2(2)      =     33.18
Log likelihood  =   -83.323546                 Prob > chi2     =    0.0000
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
drug	.1048772	.0477017	-4.96	0.000	.0430057	.2557622
age	1.120325	.0417711	3.05	0.002	1.041375	1.20526

We find that the drug results in a lower hazard—and therefore a longer survivor time—controlling for age. Older patients are more likely to die. The model as a whole is statistically significant.

The hazard ratios reported correspond to a one-unit change in the corresponding variable. It is more typical to report relative risk for 5-year changes in age. To obtain such a hazard ratio, we create a new age variable such that a one-unit change indicates a 5-year change:

```
. replace age = age/5
age was int now float
(48 real changes made)

. stcox drug age, nolog

        failure _d:  died
  analysis time _t:  studytime

Cox regression -- Breslow method for ties

No. of subjects =           48                  Number of obs   =        48
No. of failures =           31
Time at risk    =          744
                                                LR chi2(2)      =     33.18
Log likelihood  =   -83.323544                  Prob > chi2     =    0.0000

------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        drug |   .1048772   .0477017    -4.96   0.000     .0430057    .2557622
         age |   1.764898   .3290196     3.05   0.002     1.224715    2.543338
------------------------------------------------------------------------------
```

Treatment of tied failure times

The proportional hazards model assumes that the hazard function is continuous and, thus, that there are no tied survival times. Because of the way that time is recorded, however, tied events do occur in survival data. In such cases, the partial likelihood must be modified. See *Methods and formulas* for more details on the methods described below.

Stata provides four methods for handling tied failures in calculating the Cox partial likelihood through the `breslow`, `efron`, `exactm`, and `exactp` options. If there are no ties in the data, the results are identical, regardless of the method selected.

Cox regression is a series of comparisons of those subjects who fail to those subjects at risk of failing; we refer to the latter set informally as a *risk pool*. When there are tied failure times, we must decide how to calculate the risk pools for these tied observations. Assume that there are 2 observations that fail in succession. In the calculation involving the second observation, the first observation is not in the risk pool because failure has already occurred. If the two observations have the same failure time, we must decide how to calculate the risk pool for the second observation and in which order to calculate the two observations.

There are two views of time. In the first, time is continuous, so ties should not occur. If they have occurred, the likelihood reflects the marginal probability that the tied-failure events occurred before the nonfailure events in the risk pool (the order that they occurred is not important). This is called the exact marginal likelihood (option `exactm`).

In the second view, time is discrete, so ties are expected. The likelihood is changed to reflect this discreteness and calculates the conditional probability that the observed failures are those that fail in the risk pool given the observed number of failures. This is called the exact partial likelihood (option `exactp`).

Let's assume that there are five subjects—e_1, e_2, e_3, e_4, and e_5—in the risk pool and that subjects e_1 and e_2 fail. Had we been able to observe the events at a better resolution, we might have seen that e_1 failed from risk pool $e_1 + e_2 + e_3 + e_4 + e_5$ and then e_2 failed from risk pool $e_2 + e_3 + e_4 + e_5$. Alternatively, e_2 might have failed first from risk pool $e_1 + e_2 + e_3 + e_4 + e_5$, and then e_1 failed from risk pool $e_1 + e_3 + e_4 + e_5$.

The Breslow method (option `breslow`) for handling tied values simply says that because we do not know the order, we will use the largest risk pool for each tied failure event. This method assumes that both e_1 and e_2 failed from risk pool $e_1 + e_2 + e_3 + e_4 + e_5$. This approximation is fast and is the default method for handling ties. If there are many ties in the dataset, this approximation will not be accurate because the risk pools include too many observations. The Breslow method is an approximation of the exact marginal likelihood.

The Efron method (option `efron`) for handling tied values assumes that the first risk pool is $e_1 + e_2 + e_3 + e_4 + e_5$ and the second risk pool is either $e_2 + e_3 + e_4 + e_5$ or $e_1 + e_3 + e_4 + e_5$. From this, Efron noted that the e_1 and e_2 terms were in the second risk pool with probability 1/2 and so used for the second risk pool $.5(e_1 + e_2) + e_3 + e_4 + e_5$. Efron's approximation is a more accurate approximation of the exact marginal likelihood than Breslow's but takes longer to calculate.

The exact marginal method (option `exactm`) is a misnomer in that the calculation performed is also an *approximation* of the exact marginal likelihood. It is an approximation because it evaluates the likelihood (and derivatives) by using 15-point Gauss–Laguerre quadrature. For small-to-moderate samples, this is slower than the Efron approximation, but the difference in execution time diminishes when samples become larger. You may want to consider the quadrature when deciding to use this method. If the number of tied deaths is large (on average), the quadrature approximation of the function is not well behaved. A little empirical checking suggests that if the number of tied deaths is larger (on average) than 30, the quadrature does not approximate the function well.

When we view time as discrete, the exact partial method (option `exactp`) is the final method available. This approach is equivalent to computing conditional logistic regression where the groups are defined by the risk sets and the outcome is given by the death variable. This is the slowest method to use and can take a significant amount of time if the number of tied failures and the risk sets are large.

Cox regression with discrete time-varying covariates

▷ Example 3

In [ST] **stset**, we introduce the Stanford heart transplant data in which there are one or two records per patient depending on whether they received a new heart.

This dataset (Crowley and Hu 1977) consists of 103 patients admitted to the Stanford Heart Transplantation Program. Patients were admitted to the program after review by a committee and then waited for an available donor heart. While waiting, some patients died or were transferred out of the program, but 67% received a transplant. The dataset includes the year the patient was accepted into the program along with the patient's age, whether the patient had other heart surgery previously, and whether the patient received a transplant.

In the data, `posttran` becomes 1 when a patient receives a new heart, so it is a time-varying covariate. That does not, however, affect what we type to fit the model:

```
. use http://www.stata-press.com/data/r13/stan3, clear
(Heart transplant data)
. stset t1, failure(died) id(id)
  (output omitted)
```

```
. stcox age posttran surg year
         failure _d:  died
   analysis time _t:  t1
                 id:  id
Iteration 0:   log likelihood = -298.31514
Iteration 1:   log likelihood =  -289.7344
Iteration 2:   log likelihood = -289.53498
Iteration 3:   log likelihood = -289.53378
Iteration 4:   log likelihood = -289.53378
Refining estimates:
Iteration 0:   log likelihood = -289.53378
Cox regression -- Breslow method for ties
No. of subjects =          103                     Number of obs   =       172
No. of failures =           75
Time at risk    =      31938.1
                                                   LR chi2(4)      =     17.56
Log likelihood  =   -289.53378                     Prob > chi2     =    0.0015
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
age	1.030224	.0143201	2.14	0.032	1.002536	1.058677
posttran	.9787243	.3032597	-0.07	0.945	.5332291	1.796416
surgery	.3738278	.163204	-2.25	0.024	.1588759	.8796
year	.8873107	.059808	-1.77	0.076	.7775022	1.012628

We find that older patients have higher hazards, that patients tend to do better over time, and that patients with prior surgery do better. Whether a patient ultimately receives a transplant does not seem to make much difference.

◁

Cox regression with continuous time-varying covariates

The basic proportional hazards regression assumes the relationship

$$h(t) = h_0(t)\exp(\beta_1 x_1 + \cdots + \beta_k x_k)$$

where $h_0(t)$ is the baseline hazard function. For most purposes, this model is sufficient, but sometimes we may wish to introduce variables of the form $z_i(t) = z_i g(t)$, which vary continuously with time so that

$$h(t) = h_0(t)\exp\{\beta_1 x_1 + \cdots + \beta_k x_k + g(t)(\gamma_1 z_1 + \cdots + \gamma_m z_m)\} \tag{1}$$

where $z_1, \ldots, z_m$ are the time-varying covariates and where estimation has the net effect of estimating, say, a regression coefficient, γ_i, for a covariate, $g(t)z_i$, which is a function of the current time.

The time-varying covariates $z_1, \ldots, z_m$ are specified by using the `tvc(`*varlist*`)` option, and $g(t)$ is specified by using the `texp(`*exp*`)` option, where t in $g(t)$ is analysis time. For example, if we want $g(t) = \log(t)$, we would use `texp(log(_t))` because `_t` stores the analysis time once the data are `stset`.

Because the calculations in Cox regression concern themselves only with the times at which failures occur, the above results could also be achieved by `stsplit`ting the data at the observed failure times and manually generating the time-varying covariates. When this is feasible, `tvc()` merely represents a more convenient way to accomplish this. However, for large datasets with many distinct failure times, using `stsplit` may produce datasets that are too large to fit in memory, and even if this were not so, the estimation would take far longer to complete. For these reasons, the `tvc()` and `texp()` options described above were introduced.

▷ Example 4

Consider a dataset consisting of 45 observations on recovery time from walking pneumonia. Recovery time (in days) is recorded in the variable `time`, and there are measurements on the covariates `age`, `drug1`, and `drug2`, where `drug1` and `drug2` interact a choice of treatment with initial dosage level. The study was terminated after 30 days, so those who had not recovered by that time were censored (`cured` = 0).

```
. use http://www.stata-press.com/data/r13/drugtr2
. list age drug1 drug2 time cured in 1/12, separator(0)
```

	age	drug1	drug2	time	cured
1.	36	0	50	20.6	1
2.	14	0	50	6.8	1
3.	43	0	125	8.6	1
4.	25	100	0	10	1
5.	50	100	0	30	0
6.	26	0	100	13.6	1
7.	21	150	0	5.4	1
8.	25	0	100	15.4	1
9.	32	125	0	8.6	1
10.	28	150	0	8.5	1
11.	34	0	100	30	0
12.	40	0	50	30	0

Patient 1 took 50 mg of drug number 2 and was cured after 20.6 days, whereas patient 5 took 100 mg of drug number 1 and had yet to recover when the study ended and so was censored at 30 days.

We run a standard Cox regression after `stset`ting the data:

```
. stset time, failure(cured)

     failure event:  cured != 0 & cured < .
obs. time interval:  (0, time]
 exit on or before:  failure
------------------------------------------------------------------------------
       45  total observations
        0  exclusions
------------------------------------------------------------------------------
       45  observations remaining, representing
       36  failures in single-record/single-failure data
    677.9  total analysis time at risk and under observation
                                            at risk from t =         0
                                 earliest observed entry t =         0
                                      last observed exit t =        30
```

```
. stcox age drug1 drug2
        failure _d:  cured
  analysis time _t:  time
Iteration 0:   log likelihood = -116.54385
Iteration 1:   log likelihood = -102.77311
Iteration 2:   log likelihood = -101.92794
Iteration 3:   log likelihood = -101.92504
Iteration 4:   log likelihood = -101.92504
Refining estimates:
Iteration 0:   log likelihood = -101.92504
Cox regression -- Breslow method for ties
No. of subjects =           45                     Number of obs   =        45
No. of failures =           36
Time at risk    =  677.9000034
                                                   LR chi2(3)      =     29.24
Log likelihood  =   -101.92504                     Prob > chi2     =    0.0000
------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |   .8759449   .0253259    -4.58   0.000     .8276873    .9270162
       drug1 |   1.008482   .0043249     1.97   0.049     1.000041    1.016994
       drug2 |    1.00189   .0047971     0.39   0.693     .9925323    1.011337
------------------------------------------------------------------------------
```

The output includes p-values for the tests of the null hypotheses that each regression coefficient is 0 or, equivalently, that each hazard ratio is 1. That all hazard ratios are apparently close to 1 is a matter of scale; however, we can see that drug number 1 significantly increases the risk of being cured and so is an effective drug, whereas drug number 2 is ineffective (given the presence of age and drug number 1 in the model).

Suppose now that we wish to fit a model in which we account for the effect that as time goes by, the actual level of the drug remaining in the body diminishes, say, at an exponential rate. If it is known that the half-life of both drugs is close to 2 days, we can say that the actual concentration level of the drug in the patient's blood is proportional to the initial dosage times, $\exp(-0.35t)$, where t is analysis time. We now fit a model that reflects this change.

```
. stcox age, tvc(drug1 drug2) texp(exp(-0.35*_t)) nolog
        failure _d:  cured
  analysis time _t:  time
Cox regression -- Breslow method for ties
No. of subjects =           45                     Number of obs   =        45
No. of failures =           36
Time at risk    =  677.9000034
                                                   LR chi2(3)      =     36.98
Log likelihood  =   -98.052763                     Prob > chi2     =    0.0000
------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
main         |
         age |   .8614636    .028558    -4.50   0.000     .8072706    .9192948
-------------+----------------------------------------------------------------
tvc          |
       drug1 |   1.304744   .1135967     3.06   0.002     1.100059    1.547514
       drug2 |   1.200613   .1113218     1.97   0.049     1.001103    1.439882
------------------------------------------------------------------------------
Note: variables in tvc equation interacted with exp(-0.35*_t)
```

The first equation, `rh`, reports the results (hazard ratios) for the covariates that do not vary over time; the second equation, `t`, reports the results for the time-varying covariates.

As the level of drug in the blood system decreases, the drug's effectiveness diminishes. Accounting for this serves to unmask the effects of both drugs in that we now see increased effects on both. In fact, the effect on recovery time of drug number 2 now becomes significant.

❑ Technical note

The interpretation of hazard ratios requires careful consideration here. For the first model, the hazard ratio for, say, `drug1` is interpreted as the proportional change in hazard when the dosage level of `drug1` is increased by one unit. For the second model, the hazard ratio for `drug1` is the proportional change in hazard when the blood concentration level—that is, `drug1*`$\exp(-0.35t)$—increases by 1. ❑

Because the number of observations in our data is relatively small, for illustrative purposes we can `stsplit` the data at each recovery time, manually generate the blood concentration levels, and refit the second model.

```
. generate id=_n
. streset, id(id)
  (output omitted)
. stsplit, at(failures)
(31 failure times)
(812 observations (episodes) created)
. generate drug1emt = drug1*exp(-0.35*_t)
. generate drug2emt = drug2*exp(-0.35*_t)
. stcox age drug1emt drug2emt
         failure _d:  cured
   analysis time _t:  time
                 id:  id
Iteration 0:   log likelihood = -116.54385
Iteration 1:   log likelihood = -99.321912
Iteration 2:   log likelihood =  -98.07369
Iteration 3:   log likelihood =  -98.05277
Iteration 4:   log likelihood = -98.052763
Refining estimates:
Iteration 0:   log likelihood = -98.052763
Cox regression -- Breslow method for ties
No. of subjects =           45                     Number of obs   =       857
No. of failures =           36
Time at risk    =  677.9000034
                                                   LR chi2(3)      =     36.98
Log likelihood  =   -98.052763                     Prob > chi2     =    0.0000
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
age	.8614636	.028558	-4.50	0.000	.8072706	.9192948
drug1emt	1.304744	.1135967	3.06	0.002	1.100059	1.547514
drug2emt	1.200613	.1113218	1.97	0.049	1.001103	1.439882

We get the same answer. However, this required more work both for Stata and for you.

◁

Above we used `tvc()` and `texp()` to demonstrate fitting models with time-varying covariates, but these options can also be used to fit models with *time-varying coefficients*. For simplicity, consider a version of (1) that contains only one fixed covariate, x_1, and sets $z_1 = x_1$:

$$h(t) = h_0(t)\exp\{\beta_1 x_1 + g(t)\gamma_1 x_1\}$$

Rearranging terms results in

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

Given this new arrangement, we consider that $\beta_1 + \gamma_1 g(t)$ is a (possibly) time-varying coefficient on the covariate x_1, for some specified function of time $g(t)$. The coefficient has a time-invariant component, β_1, with γ_1 determining the magnitude of the time-dependent deviations from β_1. As such, a test of $\gamma_1 = 0$ is a test of time invariance for the coefficient on x_1.

Confirming that a coefficient is time invariant is one way of testing the proportional-hazards assumption. Proportional hazards implies that the relative hazard (that is, β) is fixed over time, and this assumption would be violated if a time interaction proved significant.

▷ Example 5

Returning to our cancer drug trial, we now include a time interaction on `age` as a way of testing the proportional-hazards assumption for that covariate:

```
. use http://www.stata-press.com/data/r13/drugtr, clear
(Patient Survival in Drug Trial)
. stcox drug age, tvc(age)
  (output omitted)
Cox regression -- Breslow method for ties
No. of subjects =           48                     Number of obs    =        48
No. of failures =           31
Time at risk    =          744
                                                   LR chi2(3)       =     33.63
Log likelihood  =   -83.095036                     Prob > chi2      =    0.0000
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
main						
drug	.1059862	.0478178	-4.97	0.000	.0437737	.2566171
age	1.156977	.07018	2.40	0.016	1.027288	1.303037
tvc						
age	.9970966	.0042415	-0.68	0.494	.988818	1.005445

```
Note: variables in tvc equation interacted with _t
```

We used the default function of time, $g(t) = t$, although we could have specified otherwise with the `texp()` option. The estimation results are presented in terms of hazard ratios, and so 0.9971 is an estimate of $\exp(\gamma_{\text{age}})$. Tests of hypotheses, however, are in terms of the original metric, and so 0.494 is the significance for the test of $H_0 : \gamma_{\text{age}} = 0$ versus the two-sided alternative. With respect to this specific form of misspecification, there is not much evidence to dispute the proportionality of hazards when it comes to age.

◁

Robust estimate of variance

By default, stcox produces the conventional estimate for the variance–covariance matrix of the coefficients (and hence the reported standard errors). If, however, you specify the vce(robust) option, stcox switches to the robust variance estimator (Lin and Wei 1989).

The key to the robust calculation is using the efficient score residual for each subject in the data for the variance calculation. Even in simple single-record, single-failure survival data, the same subjects appear repeatedly in the risk pools, and the robust calculation needs to account for that.

▷ Example 6

Refitting the Stanford heart transplant data model with robust standard errors, we obtain

```
. use http://www.stata-press.com/data/r13/stan3
(Heart transplant data)
. stset t1, failure(died) id(id)

                id:  id
     failure event:  died != 0 & died < .
obs. time interval:  (t1[_n-1], t1]
 exit on or before:  failure

------------------------------------------------------------------------------
       172  total observations
         0  exclusions
------------------------------------------------------------------------------

       172  observations remaining, representing
       103  subjects
        75  failures in single-failure-per-subject data
   31938.1  total analysis time at risk and under observation
                                             at risk from t =         0
                                  earliest observed entry t =         0
                                       last observed exit t =      1799

. stcox age posttran surg year, vce(robust)

        failure _d:  died
  analysis time _t:  t1
                id:  id

Iteration 0:   log pseudolikelihood = -298.31514
Iteration 1:   log pseudolikelihood =  -289.7344
Iteration 2:   log pseudolikelihood = -289.53498
Iteration 3:   log pseudolikelihood = -289.53378
Iteration 4:   log pseudolikelihood = -289.53378
Refining estimates:
Iteration 0:   log pseudolikelihood = -289.53378

Cox regression -- Breslow method for ties

No. of subjects      =          103                 Number of obs    =       172
No. of failures      =           75
Time at risk         =      31938.1
                                                    Wald chi2(4)     =     19.68
Log pseudolikelihood =   -289.53378                 Prob > chi2      =    0.0006
                                       (Std. Err. adjusted for 103 clusters in id)
------------------------------------------------------------------------------
             |               Robust
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |   1.030224   .0148771     2.06   0.039     1.001474    1.059799
    posttran |   .9787243   .2961736    -0.07   0.943     .5408498    1.771104
     surgery |   .3738278   .1304912    -2.82   0.005     .1886013    .7409665
        year |   .8873107   .0613176    -1.73   0.084     .7749139     1.01601
------------------------------------------------------------------------------
```

Note the word `Robust` above `Std. Err.` in the table and the phrase "Std. Err. adjusted for 103 clusters in id" above the table.

The hazard ratio estimates are the same as before, but the standard errors are slightly different.

◁

❑ Technical note

In the previous example, `stcox` knew to specify `vce(cluster id)` for us when we specified `vce(robust)`.

To see the importance of `vce(cluster id)`, consider simple single-record, single-failure survival data, a piece of which is

t0	t	died	x
0	5	1	1
0	9	0	1
0	8	0	0

and then consider the absolutely equivalent multiple-record survival data:

id	t0	t	died	x
1	0	3	0	1
1	3	5	1	1
2	0	6	0	1
2	6	9	0	1
3	0	3	0	0
3	3	8	0	0

Both datasets record the same underlying data, and so both should produce the same numerical results. This should be true regardless of whether `vce(robust)` is specified.

In the second dataset, were we to ignore `id`, it would appear that there are 6 observations on 6 subjects. The key ingredients in the robust calculation are the efficient score residuals, and viewing the data as 6 observations on 6 subjects produces different score residuals. Let's call the 6 score residuals s_1, s_2, …, s_6 and the 3 score residuals that would be generated by the first dataset S_1, S_2, and S_3. $S_1 = s_1 + s_2$, $S_2 = s_3 + s_4$, and $S_3 = s_5 + s_6$.

That residuals sum is the key to understanding the `vce(cluster` *clustvar*`)` option. When you specify `vce(cluster id)`, Stata makes the robust calculation based not on the overly detailed s_1, s_2, …, s_6 but on $S_1 + S_2$, $S_3 + S_4$, and $S_5 + S_6$. That is, Stata sums residuals within clusters before entering them into subsequent calculations (where they are squared), so results estimated from the second dataset are equal to those estimated from the first. In more complicated datasets with time-varying regressors, delayed entry, and gaps, this action of summing within cluster, in effect, treats the cluster (which is typically a subject) as a unified whole.

Because we had `stset` an `id()` variable, `stcox` knew to specify `vce(cluster id)` for us when we specified `vce(robust)`. You may, however, override the default clustering by specifying `vce(cluster` *clustvar*`)` with a different variable from the one you used in `stset, id()`. This is useful in analyzing multiple-failure data, where you need to `stset` a pseudo-ID establishing the time from the last failure as the onset of risk.

❑

Cox regression with multiple-failure data

▷ Example 7

In [ST] **stsum**, we introduce a multiple-failure dataset:

```
. use http://www.stata-press.com/data/r13/mfail
. stdescribe
```

Category	total	per subject: mean	min	median	max
no. of subjects	926				
no. of records	1734	1.87257	1	2	4
(first) entry time		0	0	0	0
(final) exit time		470.6857	1	477	960
subjects with gap	0				
time on gap if gap	0	.	.	.	.
time at risk	435855	470.6857	1	477	960
failures	808	.8725702	0	1	3

This dataset contains two variables—`x1` and `x2`—which we believe affect the hazard of failure.

If we simply want to analyze these multiple-failure data as if the baseline hazard remains unchanged as events occur (that is, the hazard may change with time, but time is measured from 0 and is independent of when the last failure occurred), we can type

```
. stcox x1 x2, vce(robust)
Iteration 0:   log pseudolikelihood = -5034.9569
Iteration 1:   log pseudolikelihood = -4978.4198
Iteration 2:   log pseudolikelihood = -4978.1915
Iteration 3:   log pseudolikelihood = -4978.1914
Refining estimates:
Iteration 0:   log pseudolikelihood = -4978.1914
Cox regression -- Breslow method for ties
No. of subjects      =          926                 Number of obs    =      1734
No. of failures      =          808
Time at risk         =       435855
                                                    Wald chi2(2)     =    152.13
Log pseudolikelihood =   -4978.1914                 Prob > chi2      =    0.0000
                                        (Std. Err. adjusted for 926 clusters in id)
```

_t	Haz. Ratio	Robust Std. Err.	z	P>|z|	[95% Conf.	Interval]
x1	2.273456	.1868211	9.99	0.000	1.935259	2.670755
x2	.329011	.0523425	-6.99	0.000	.2408754	.4493951

We chose to fit this model with robust standard errors—we specified `vce(robust)`—but you can estimate conventional standard errors if you wish.

In [ST] **stsum**, we discuss analyzing this dataset as the time since last failure. We wished to assume that the hazard function remained unchanged with failure, except that one restarted the same hazard function. To that end, we made the following changes to our data:

```
. stgen nf = nfailures()
. egen newid = group(id nf)
```

```
. sort newid t
. by newid: replace t = t - t0[1]
(808 real changes made)
. by newid: gen newt0 = t0 - t0[1]
. stset t, id(newid) failure(d) time0(newt0) noshow
                id:  newid
     failure event:  d != 0 & d < .
obs. time interval:  (newt0, t]
 exit on or before:  failure
------------------------------------------------------------------------------
     1734  total observations
        0  exclusions
------------------------------------------------------------------------------
     1734  observations remaining, representing
     1734  subjects
      808  failures in single-failure-per-subject data
   435444  total analysis time at risk and under observation
                                         at risk from t =         0
                              earliest observed entry t =         0
                                   last observed exit t =       797
```

That is, we took each subject and made many `newid` subjects out of each, with each subject entering at time 0 (now meaning the time of the last failure). `id` still identifies a real subject, but Stata thinks the identifier variable is `newid` because we `stset, id(newid)`. If we were to fit a model with `vce(robust)`, we would get

```
. stcox x1 x2, vce(robust) nolog
Cox regression -- Breslow method for ties
No. of subjects      =         1734                 Number of obs    =      1734
No. of failures      =          808
Time at risk         =       435444
                                                    Wald chi2(2)     =     88.51
Log pseudolikelihood =   -5062.5815                 Prob > chi2      =    0.0000
                                (Std. Err. adjusted for 1734 clusters in newid)
------------------------------------------------------------------------------
          _t | Haz. Ratio   Robust Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          x1 |   2.002547   .1936906     7.18   0.000     1.656733    2.420542
          x2 |   .2946263   .0569167    -6.33   0.000     .2017595    .4302382
------------------------------------------------------------------------------
```

Note carefully the message concerning the clustering: standard errors have been adjusted for clustering on `newid`. We, however, want the standard errors adjusted for clustering on `id`, so we must specify the `vce(cluster` *clustvar*`)` option:

```
. stcox x1 x2, vce(cluster id) nolog
Cox regression -- Breslow method for ties
No. of subjects      =         1734                 Number of obs    =      1734
No. of failures      =          808
Time at risk         =       435444
                                                    Wald chi2(2)     =     93.66
Log pseudolikelihood =   -5062.5815                 Prob > chi2      =    0.0000
                                  (Std. Err. adjusted for 926 clusters in id)
------------------------------------------------------------------------------
          _t | Haz. Ratio   Robust Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          x1 |   2.002547   .1920151     7.24   0.000     1.659452    2.416576
          x2 |   .2946263   .0544625    -6.61   0.000     .2050806    .4232709
------------------------------------------------------------------------------
```

That is, if you are using `vce(robust)`, you must remember to specify `vce(cluster` *clustvar*`)` for yourself when

1. you are analyzing multiple-failure data and
2. you have reset time to time since last failure, so what Stata considers the subjects are really subsubjects.

◁

Stratified estimation

When you type

```
. stcox xvars, strata(svars)
```

you are allowing the baseline hazard functions to differ for the groups identified by *svars*. This is equivalent to fitting separate Cox proportional hazards models under the constraint that the coefficients are equal but the baseline hazard functions are not.

▷ Example 8

Say that in the Stanford heart experiment data, there was a change in treatment for all patients, before and after transplant, in 1970 and then again in 1973. Further assume that the proportional-hazards assumption is not reasonable for these changes in treatment—perhaps the changes result in short-run benefit but little expected long-run benefit. Our interest in the data is not in the effect of these treatment changes but in the effect of transplantation, for which we still find the proportional-hazards assumption reasonable. We might fit our model to account for these fictional changes by typing

```
. use http://www.stata-press.com/data/r13/stan3, clear
(Heart transplant data)
. generate pgroup = year
. recode pgroup min/69=1 70/72=2 73/max=3
(pgroup: 172 changes made)
```

```
. stcox age posttran surg year, strata(pgroup) nolog
        failure _d:  died
  analysis time _t:  t1
                id:  id
Stratified Cox regr. -- Breslow method for ties
No. of subjects =          103                     Number of obs   =       172
No. of failures =           75
Time at risk    =      31938.1
                                                   LR chi2(4)      =     20.67
Log likelihood  =   -213.35033                     Prob > chi2     =    0.0004
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
age	1.027406	.0150188	1.85	0.064	.9983874	1.057268
posttran	1.075476	.3354669	0.23	0.816	.583567	1.982034
surgery	.2222415	.1218386	-2.74	0.006	.0758882	.6508429
year	.5523966	.1132688	-2.89	0.004	.3695832	.825638

Stratified by pgroup

Of course, we could obtain the robust estimate of variance by also including the `vce(robust)` option. ◁

Cox regression as Poisson regression

▷ Example 9

In example 2, we fit the following Cox model to data from a cancer drug trial with 48 participants:

```
. use http://www.stata-press.com/data/r13/drugtr, clear
(Patient Survival in Drug Trial)
. summarize
```

Variable	Obs	Mean	Std. Dev.	Min	Max
studytime	48	15.5	10.25629	1	39
died	48	.6458333	.4833211	0	1
drug	48	.5833333	.4982238	0	1
age	48	55.875	5.659205	47	67
_st	48	1	0	1	1
_d	48	.6458333	.4833211	0	1
_t	48	15.5	10.25629	1	39
_t0	48	0	0	0	0

```
. stcox drug age
  (output omitted)
Cox regression -- Breslow method for ties
No. of subjects =           48                     Number of obs   =        48
No. of failures =           31
Time at risk    =          744
                                                   LR chi2(2)      =     33.18
Log likelihood  =   -83.323546                     Prob > chi2     =    0.0000
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
drug	.1048772	.0477017	-4.96	0.000	.0430057	.2557622
age	1.120325	.0417711	3.05	0.002	1.041375	1.20526

In what follows, we discuss baseline hazard functions. Thus for clarity, we first fit the same model with an alternate age variable so that "baseline" reflects someone in the control group who is 50 years old and not a newborn; see *Making baseline reasonable* in [ST] **stcox postestimation** for more details.

```
. generate age50 = age - 50
. stcox drug age50
  (output omitted)
Cox regression -- Breslow method for ties
No. of subjects =           48                     Number of obs    =        48
No. of failures =           31
Time at risk    =          744
                                                   LR chi2(2)       =     33.18
Log likelihood  =   -83.323546                     Prob > chi2      =    0.0000
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
drug	.1048772	.0477017	-4.96	0.000	.0430057	.2557622
age50	1.120325	.0417711	3.05	0.002	1.041375	1.20526

Because stcox does not estimate a baseline hazard function, our model and hazard ratios remain unchanged.

Among others, Royston and Lambert (2011, sec. 4.5) show that you can obtain identical hazard ratios by fitting a Poisson model on the above data after splitting on all observed failure times.

Because these data have already been stset, variable _t0 contains the beginning of the time span (which, for these simple data, is time zero for everyone), variable _t contains the end of the time span, and variable _d indicates failure (_d == 1) or censoring (_d == 0).

As we did in example 4, we can split these single-record observations at each observed failure time, thus creating a dataset with multiple records per subject. To do so, we must first create an ID variable that identifies each observation as a unique patient:

```
. generate id = _n
. streset, id(id)
-> stset studytime, id(id) failure(died)
                id:  id
     failure event:  died != 0 & died < .
obs. time interval:  (studytime[_n-1], studytime]
 exit on or before:  failure
------------------------------------------------------------------------------
       48  total observations
        0  exclusions
------------------------------------------------------------------------------
       48  observations remaining, representing
       48  subjects
       31  failures in single-failure-per-subject data
      744  total analysis time at risk and under observation
                                             at risk from t =         0
                                  earliest observed entry t =         0
                                       last observed exit t =        39
. stsplit, at(failures) riskset(interval)
(21 failure times)
(534 observations (episodes) created)
```

The output shows that we have 21 unique failure times and that we created 534 new observations for a total of $48 + 534 = 582$ observations. Also created is the `interval` variable, which contains a value of `1` for those records that span from time zero to the first failure time, `2` for those records that span from the first failure time to the second failure time, all the way up to a value of `21` for those records that span from the 20th failure time to the 21st failure time. To see this requires a little bit of sorting and data manipulation:

```
. gsort _t -_d
. by _t: generate tolist = (_n==1) & _d
. list _t0 _t interval if tolist
```

	_t0	_t	interval
1.	0	1	1
49.	1	2	2
95.	2	3	3
140.	3	4	4
184.	4	5	5
226.	5	6	6
266.	6	7	7
303.	7	8	8
340.	8	10	9
371.	10	11	10
400.	11	12	11
426.	12	13	12
450.	13	15	13
473.	15	16	14
494.	16	17	15
517.	17	22	16
532.	22	23	17
545.	23	24	18
556.	24	25	19
566.	25	28	20
576.	28	33	21

Thus for example, interval 16 ranges from time 17 to time 22.

For this newly created multiple-record dataset, our Cox model fit will be identical because we have not added any information to the data. If you do not believe us, feel free to now try the following command:

```
. stcox drug age50
```

At this point, it would seem that making the dataset bigger is a needless waste of space, but what it grants us is the ability to directly estimate the baseline hazard function in addition to the hazard ratios we previously obtained. We accomplish this by using Poisson regression.

Poisson regression models event counts, and so we use our event counter for these data, the failure indicator `_d`, as the response variable. That `_d` is only valued as zero or one should not bother you—it is still a count variable. We need to treat time spanned as the amount of exposure a subject had toward failing; the longer the interval, the greater the exposure. As such, we create a variable that records the length of each time span and include it as an `exposure()` variable in our Poisson model. We also include indicator variables for each of the 21 time intervals, with no base level assumed; we use the `ibn.` factor-variable specification and the `noconstant` option:

```
. generate time_exposed = _t - _t0
. poisson _d ibn.interval drug age50, exposure(time_exposed) noconstant irr
Iteration 0:   log likelihood = -1239.0595
Iteration 1:   log likelihood = -114.23986
Iteration 2:   log likelihood = -100.13556
Iteration 3:   log likelihood = -99.938857
Iteration 4:   log likelihood = -99.937354
Iteration 5:   log likelihood = -99.937354
Poisson regression                              Number of obs   =        573
                                                Wald chi2(23)   =     224.18
Log likelihood = -99.937354                     Prob > chi2     =     0.0000
```

_d	IRR	Std. Err.	z	P>|z|	[95% Conf.	Interval]
interval						
1	.0360771	.0284092	-4.22	0.000	.0077081	.1688562
2	.0215286	.0225926	-3.66	0.000	.0027526	.1683778
3	.0228993	.0240269	-3.60	0.000	.0029289	.1790349
4	.0471539	.0366942	-3.92	0.000	.0102596	.2167234
5	.0596354	.045201	-3.72	0.000	.0134999	.2634375
6	.0749754	.0561057	-3.46	0.001	.017296	.3250055
7	.0396981	.0406826	-3.15	0.002	.0053267	.2958558
8	.1203377	.0744625	-3.42	0.001	.0357845	.4046762
9	.0276002	.0283969	-3.49	0.000	.003674	.207341
10	.1120012	.083727	-2.93	0.003	.0258763	.4847777
11	.1358135	.1024475	-2.65	0.008	.0309642	.5956972
12	.1007666	.1040271	-2.22	0.026	.0133221	.7621858
13	.0525547	.0540884	-2.86	0.004	.0069915	.395051
14	.1206462	.1250492	-2.04	0.041	.0158215	.919984
15	.1321868	.1357583	-1.97	0.049	.0176599	.9894363
16	.0670895	.0503478	-3.60	0.000	.0154122	.2920415
17	.5736017	.4415411	-0.72	0.470	.1268766	2.59322
18	.4636009	.5113227	-0.70	0.486	.0533731	4.026856
19	.5272168	.5810138	-0.58	0.561	.0608039	4.571377
20	.2074545	.2292209	-1.42	0.155	.023791	1.80898
21	.2101074	.2344194	-1.40	0.162	.0235909	1.871275
drug	.1048772	.0477017	-4.96	0.000	.0430057	.2557622
age50	1.120325	.0417711	3.05	0.002	1.041375	1.20526
ln(time_e~d)	1	(exposure)				

The incidence-rate ratios from `poisson` (obtained with the `irr` option) are identical to the hazard ratios we previously obtained. Additionally, the incidence-rate ratio for each of the 21 intervals is an estimate of the baseline hazard function for that time interval.

`poisson` gives us an estimated baseline hazard function (the hazard for someone aged 50 in the control group) as a piecewise-constant function. If we had continued to use `stcox`, estimating the baseline hazard function would have required that we apply a kernel smoother to the estimated baseline contributions; see example 3 of [ST] **stcox postestimation** for details. In other words, estimating a baseline hazard after `stcox` is not easy, and it requires choosing a kernel function and bandwidth. As such, the title of this section is technically a misnomer; the models are not exactly the same, only the "hazard ratios" are. Using `poisson` instead of `stcox` carries the added assumption that the baseline hazard is constant between observed failures. Making this assumption buys you the ability to directly estimate the baseline hazard.

There also exists a duality between the Poisson model and the exponential model as fit by `streg`; see [ST] **streg**. A defining property of the Poisson distribution is that waiting times between events are distributed as exponential. Thus we can fit the same piecewise-constant hazard model with

```
. streg ibn.interval drug age50, dist(exponential) noconstant
```

which we invite you to try.

Of course, if you are willing to assume the hazard is piecewise constant, then perhaps you do not need it to change over all 21 observed failure times, and thus perhaps you would want to collapse some intervals. Better still, why not just use `streg` without the indicator variables for `interval`, assume the baseline hazard is some smooth function, and reduce your 21 parameters to one or two estimated shape parameters? The advantages to this fully parametric approach are that you get a parsimonious model and smooth hazard functions that you can estimate at any time point. The disadvantage is that you now carry the stringent assumption that your hazard follows the chosen functional form. If you choose the wrong function, then your hazard ratios are, in essence, worthless.

The two extremes here are the model that makes no assumption about the baseline hazard (the Cox model) and the model that makes the strongest assumptions about the baseline hazard (the fully parametric model). Our piecewise-constant baseline hazard model represents a compromise between Cox regression and fully parametric regression. If you are interested in other ways you can compromise between Cox and parametric models, we recommend you read Royston and Lambert (2011), which is entirely devoted to that topic. There you will find information on (among other things) Royston–Parmar models (Royston and Parmar 2002; Lambert and Royston 2009), proportional-odds models, scaled-probit models, the use of cubic splines and fractional polynomials, time-dependent effects, and models for relative survival.

◁

Cox regression with shared frailty

A shared-frailty model is the survival-data analog to regression models with random effects. A *frailty* is a latent random effect that enters multiplicatively on the hazard function. In a Cox model, the data are organized as $i = 1, \ldots, n$ groups with $j = 1, \ldots, n_i$ observations in group i. For the jth observation in the ith group, the hazard is

$$h_{ij}(t) = h_0(t)\alpha_i \exp(\mathbf{x}_{ij}\boldsymbol{\beta})$$

where α_i is the group-level frailty. The frailties are unobservable positive quantities and are assumed to have mean 1 and variance θ, to be estimated from the data. You can fit a Cox shared-frailty model by specifying `shared(`*varname*`)`, where *varname* defines the groups over which frailties are shared. `stcox, shared()` treats the frailties as being gamma distributed, but this is mainly an issue of computational convenience; see *Methods and formulas*. Theoretically, any distribution with positive support, mean 1, and finite variance may be used to model frailty.

Shared-frailty models are used to model within-group correlation; observations within a group are correlated because they share the same frailty. The estimate of θ is used to measure the degree of within-group correlation, and the shared-frailty model reduces to standard Cox when $\theta = 0$.

For $\nu_i = \log\alpha_i$, the hazard can also be expressed as

$$h_{ij}(t) = h_0(t) \exp(\mathbf{x}_{ij}\boldsymbol{\beta} + \nu_i)$$

and thus the log frailties, ν_i, are analogous to random effects in standard linear models.

▷ Example 10

Consider the data from a study of 38 kidney dialysis patients, as described in McGilchrist and Aisbett (1991). The study is concerned with the prevalence of infection at the catheter insertion point. Two recurrence times (in days) are measured for each patient, and each recorded time is the time from initial insertion (onset of risk) to infection or censoring:

```
. use http://www.stata-press.com/data/r13/catheter, clear
(Kidney data, McGilchrist and Aisbett, Biometrics, 1991)
. list patient time infect age female in 1/10
```

	patient	time	infect	age	female
1.	1	16	1	28	0
2.	1	8	1	28	0
3.	2	13	0	48	1
4.	2	23	1	48	1
5.	3	22	1	32	0
6.	3	28	1	32	0
7.	4	318	1	31.5	1
8.	4	447	1	31.5	1
9.	5	30	1	10	0
10.	5	12	1	10	0

Each patient (`patient`) has two recurrence times (`time`) recorded, with each catheter insertion resulting in either infection (`infect==1`) or right-censoring (`infect==0`). Among the covariates measured are `age` and sex (`female==1` if female, `female==0` if male).

One subtlety to note concerns the use of the generic term *subjects*. In this example, the subjects are taken to be the individual catheter insertions, not the patients themselves. This is a function of how the data were recorded—the onset of risk occurs at catheter insertion (of which there are two for each patient), and not, say, at the time of admission of the patient into the study. We therefore have two subjects (insertions) within each group (patient).

It is reasonable to assume independence of patients but unreasonable to assume that recurrence times within each patient are independent. One solution would be to fit a standard Cox model, adjusting the standard errors of the estimated hazard ratios to account for the possible correlation by specifying `vce(cluster patient)`.

We could instead model the correlation by assuming that the correlation is the result of a latent patient-level effect, or frailty. That is, rather than fitting a standard model and specifying `vce(cluster patient)`, we could fit a frailty model by specifying `shared(patient)`:

```
. stset time, fail(infect)
  (output omitted)
. stcox age female, shared(patient)
        failure _d:  infect
  analysis time _t:  time
Fitting comparison Cox model:
Estimating frailty variance:
Iteration 0:   log profile likelihood = -182.06713
Iteration 1:   log profile likelihood =  -181.9791
Iteration 2:   log profile likelihood = -181.97453
Iteration 3:   log profile likelihood = -181.97453
Fitting final Cox model:
Iteration 0:   log likelihood = -199.05599
Iteration 1:   log likelihood = -183.72296
Iteration 2:   log likelihood = -181.99509
Iteration 3:   log likelihood = -181.97455
Iteration 4:   log likelihood = -181.97453
Refining estimates:
Iteration 0:   log likelihood = -181.97453
Cox regression --
        Breslow method for ties                 Number of obs      =        76
        Gamma shared frailty                    Number of groups   =        38
Group variable: patient
No. of subjects =           76                  Obs per group: min =         2
No. of failures =           58                                 avg =         2
Time at risk    =         7424                                 max =         2
                                                Wald chi2(2)       =     11.66
Log likelihood  =   -181.97453                  Prob > chi2        =    0.0029
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
age	1.006202	.0120965	0.51	0.607	.9827701	1.030192
female	.2068678	.095708	-3.41	0.001	.0835376	.5122756
theta	.4754497	.2673108				

```
Likelihood-ratio test of theta=0: chibar2(01) =     6.27 Prob>=chibar2 = 0.006
Note: standard errors of hazard ratios are conditional on theta.
```

From the output, we obtain $\widehat{\theta} = 0.475$, and given the standard error of $\widehat{\theta}$ and likelihood-ratio test of H_0: $\theta = 0$, we find a significant frailty effect, meaning that the correlation within patient cannot be ignored. Contrast this with the analysis of the same data in [ST] **streg**, which considered both Weibull and lognormal shared-frailty models. For Weibull, there was significant frailty; for lognormal, there was not.

The estimated ν_i are not displayed in the coefficient table but may be retrieved postestimation by using `predict` with the `effects` option; see [ST] **stcox postestimation** for an example.

◁

In shared-frailty Cox models, the estimation consists of two steps. In the first step, the optimization is in terms of θ only. For fixed θ, the second step consists of fitting a standard Cox model via penalized log likelihood, with the ν_i introduced as estimable coefficients of dummy variables identifying the groups. The penalty term in the penalized log likelihood is a function of θ; see *Methods and formulas*. The final estimate of θ is taken to be the one that maximizes the penalized log likelihood. Once the optimal θ is obtained, it is held fixed, and a final penalized Cox model is fit. As a result, the standard errors of the main regression parameters (or hazard ratios, if displayed as such) are treated as conditional on θ fixed at its optimal value.

With gamma-distributed frailty, hazard ratios decay over time in favor of the *frailty effect* and thus the displayed "Haz. Ratio" in the above output is actually the hazard ratio only for $t = 0$. The degree of decay depends on θ. Should the estimated θ be close to 0, the hazard ratios do regain their usual interpretation; see Gutierrez (2002) for details.

❑ Technical note

The likelihood-ratio test of $\theta = 0$ is a boundary test and thus requires careful consideration concerning the calculation of its p-value. In particular, the null distribution of the likelihood-ratio test statistic is not the usual χ^2_1 but is rather a 50:50 mixture of a χ^2_0 (point mass at zero) and a χ^2_1, denoted as $\overline{\chi}^2_{01}$. See Gutierrez, Carter, and Drukker (2001) for more details.

❑

❑ Technical note

In [ST] **streg**, shared-frailty models are compared and contrasted with *unshared* frailty models. Unshared-frailty models are used to model heterogeneity, and the frailties are integrated out of the conditional survivor function to produce an unconditional survivor function, which serves as a basis for all likelihood calculations.

Given the nature of Cox regression (the baseline hazard remains unspecified), there is no Cox regression analog to the unshared parametric frailty model as fit using `streg`. That is not to say that you cannot fit a shared-frailty model with 1 observation per group; you can as long as you do not fit a null model.

❑

Stored results

`stcox` stores the following in `e()`:

Scalars

`e(N)`	number of observations
`e(N_sub)`	number of subjects
`e(N_fail)`	number of failures
`e(N_g)`	number of groups
`e(df_m)`	model degrees of freedom
`e(r2_p)`	pseudo-R-squared
`e(ll)`	log likelihood
`e(ll_0)`	log likelihood, constant-only model
`e(ll_c)`	log likelihood, comparison model
`e(N_clust)`	number of clusters
`e(chi2)`	χ^2
`e(chi2_c)`	χ^2, comparison model
`e(risk)`	total time at risk
`e(g_min)`	smallest group size
`e(g_avg)`	average group size
`e(g_max)`	largest group size
`e(theta)`	frailty parameter
`e(se_theta)`	standard error of θ
`e(p_c)`	significance, comparison model
`e(rank)`	rank of `e(V)`

Macros

e(cmd)	cox or stcox_fr
e(cmd2)	stcox
e(cmdline)	command as typed
e(depvar)	_t
e(t0)	_t0
e(texp)	function used for time-varying covariates
e(ties)	method used for handling ties
e(strata)	strata variables
e(shared)	frailty grouping variable
e(clustvar)	name of cluster variable
e(offset)	linear offset variable
e(chi2type)	Wald or LR; type of model χ^2 test
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. Err.
e(method)	requested estimation method
e(datasignature)	the checksum
e(datasignaturevars)	variables used in calculation of checksum
e(properties)	b V
e(estat_cmd)	program used to implement estat
e(predict)	program used to implement predict
e(footnote)	program used to implement the footnote display
e(marginsnotok)	predictions disallowed by margins
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices

e(b)	coefficient vector
e(V)	variance–covariance matrix of the
e(V_modelbased)	model-based variance estimators

Functions

e(sample)	marks estimation sample

Methods and formulas

The proportional hazards model with explanatory variables was first suggested by Cox (1972). For an introductory explanation, see Hosmer, Lemeshow, and May (2008, chap. 3, 4, and 7), Kahn and Sempos (1989, 193–198), and Selvin (2004, 412–442). For an introduction for the social scientist, see Box-Steffensmeier and Jones (2004, chap. 4). For a comprehensive review of the methods in this entry, see Klein and Moeschberger (2003). For a detailed development of these methods, see Kalbfleisch and Prentice (2002). For more Stata-specific insight, see Cleves et al. (2010), Dupont (2009), and Vittinghoff et al. (2012).

Let $\mathbf{x}_i$ be the row vector of covariates for the time interval $(t_{0i}, t_i]$ for the ith observation in the dataset $i = 1, \ldots, N$. stcox obtains parameter estimates, $\widehat{\boldsymbol{\beta}}$, by maximizing the partial log-likelihood function

$$\log L = \sum_{j=1}^{D} \Big[\sum_{i \in D_j} \mathbf{x}_i \boldsymbol{\beta} - d_j \log \Big\{ \sum_{k \in R_j} \exp(\mathbf{x}_k \boldsymbol{\beta}) \Big\} \Big]$$

where j indexes the ordered failure times $t_{(j)}$, $j = 1, \ldots, D$; D_j is the set of d_j observations that fail at $t_{(j)}$; d_j is the number of failures at $t_{(j)}$; and R_j is the set of observations k that are at risk at time $t_{(j)}$ (that is, all k such that $t_{0k} < t_{(j)} \leq t_k$). This formula for log$L$ is for unweighted data and handles ties by using the Peto–Breslow approximation (Peto 1972; Breslow 1974), which is the default method of handling ties in `stcox`.

If `strata(`*varnames*`)` is specified, then the partial log likelihood is the sum of each stratum-specific partial log likelihood, obtained by forming the ordered failure times $t_{(j)}$, the failure sets D_j, and the risk sets R_j, using only those observations within that stratum.

The variance of $\widehat{\boldsymbol{\beta}}$ is estimated by the conventional inverse matrix of (negative) second derivatives of log$L$, unless `vce(robust)` is specified, in which case the method of Lin and Wei (1989) is used. That method treats efficient score residuals as analogs to the log-likelihood scores one would find in fully parametric models; see *Methods and formulas* in [ST] **stcox postestimation** for how to calculate efficient score residuals. If `vce(cluster` *clustvar*`)` is specified, the efficient score residuals are summed within cluster before the sandwich (robust) estimator is applied.

Tied values are handled using one of four approaches. The log likelihoods corresponding to the four approaches are given with weights (`exactp` and `efron` do not allow weights) and offsets by

$$\log L_{\text{breslow}} = \sum_{j=1}^{D} \sum_{i \in D_j} \left[w_i(\mathbf{x}_i\boldsymbol{\beta} + \text{offset}_i) - w_i \log \left\{ \sum_{\ell \in R_j} w_\ell \exp(\mathbf{x}_\ell\boldsymbol{\beta} + \text{offset}_\ell) \right\} \right]$$

$$\log L_{\text{efron}} = \sum_{j=1}^{D} \sum_{i \in D_j} \left[\mathbf{x}_i\boldsymbol{\beta} + \text{offset}_i - d_j^{-1} \sum_{k=0}^{d_j - 1} \log \left\{ \sum_{\ell \in R_j} \exp(\mathbf{x}_\ell\boldsymbol{\beta} + \text{offset}_\ell) - kA_j \right\} \right]$$

$$A_j = d_j^{-1} \sum_{\ell \in D_j} \exp(\mathbf{x}_\ell\boldsymbol{\beta} + \text{offset}_\ell)$$

$$\log L_{\text{exactm}} = \sum_{j=1}^{D} \log \int_0^\infty \prod_{\ell \in D_j} \left\{ 1 - \exp\left(-\frac{e_\ell}{s} t \right) \right\}^{w_\ell} \exp(-t) dt$$

$$e_\ell = \exp(\mathbf{x}_\ell\boldsymbol{\beta} + \text{offset}_\ell)$$

$$s = \sum_{\substack{k \in R_j \\ k \notin D_j}} w_k \exp(\mathbf{x}_k\boldsymbol{\beta} + \text{offset}_k) = \text{sum of weighted nondeath risk scores}$$

$$\log L_{\text{exactp}} = \sum_{j=1}^{D} \left\{ \sum_{i \in R_j} \delta_{ij}(\mathbf{x}_i\boldsymbol{\beta} + \text{offset}_i) - \log f(r_j, d_j) \right\}$$

$$f(r, d) = f(r-1, d) + f(r-1, d-1) \exp(\mathbf{x}_k\boldsymbol{\beta} + \text{offset}_k)$$

$$k = r^{\text{th}} \text{ observation in the set } R_j$$

$$r_j = \text{cardinality of the set } R_j$$

$$f(r, d) = \begin{cases} 0 & \text{if } r < d \\ 1 & \text{if } d = 0 \end{cases}$$

where δ_{ij} is an indicator for failure of observation i at time $t_{(j)}$.

Calculations for the exact marginal log likelihood (and associated derivatives) are obtained with 15-point Gauss–Laguerre quadrature. The `breslow` and `efron` options both provide approximations of the exact marginal log likelihood. The `efron` approximation is a better (closer) approximation, but the `breslow` approximation is faster. The choice of the approximation to use in a given situation should generally be driven by the proportion of ties in the data.

For shared-frailty models, the data are organized into G groups with the ith group consisting of n_i observations, $i = 1, \ldots, G$. From Therneau and Grambsch (2000, 253–255), estimation of θ takes place via maximum profile log likelihood. For fixed θ, estimates of $\boldsymbol{\beta}$ and $\nu_1, \ldots, \nu_G$ are obtained by maximizing

$$\log L(\theta) = \log L_{\text{Cox}}(\boldsymbol{\beta}, \nu_1, \ldots, \nu_G) + \sum_{i=1}^{G} \left[\frac{1}{\theta} \{\nu_i - \exp(\nu_i)\} + \left(\frac{1}{\theta} + D_i\right) \left\{1 - \log\left(\frac{1}{\theta} + D_i\right)\right\} - \frac{\log\theta}{\theta} + \log\Gamma\left(\frac{1}{\theta} + D_i\right) - \log\Gamma\left(\frac{1}{\theta}\right) \right]$$

where D_i is the number of death events in group i, and $\log L_{\text{Cox}}(\boldsymbol{\beta}, \nu_1, \ldots, \nu_G)$ is the standard Cox partial log likelihood, with the ν_i treated as the coefficients of indicator variables identifying the groups. That is, the jth observation in the ith group has log relative hazard $\mathbf{x}_{ij}\boldsymbol{\beta} + \nu_i$. The estimate of the frailty parameter, $\widehat{\theta}$, is chosen as that which maximizes $\log L(\theta)$. The final estimates of $\boldsymbol{\beta}$ are obtained by maximizing $\log L(\widehat{\theta})$ in $\boldsymbol{\beta}$ and the ν_i. The ν_i are not reported in the coefficient table but are available via `predict`; see [ST] **stcox postestimation**. The estimated variance–covariance matrix of $\widehat{\boldsymbol{\beta}}$ is obtained as the appropriate submatrix of the variance matrix of $(\widehat{\boldsymbol{\beta}}, \widehat{\nu}_1, \ldots, \widehat{\nu}_G)$, and that matrix is obtained as the inverse of the negative Hessian of $\log L(\widehat{\theta})$. Therefore, standard errors and inference based on $\widehat{\boldsymbol{\beta}}$ should be treated as conditional on $\theta = \widehat{\theta}$.

The likelihood-ratio test statistic for testing H_0: $\theta = 0$ is calculated as minus twice the difference between the log likelihood for a Cox model without shared frailty and $\log L(\widehat{\theta})$ evaluated at the final $(\widehat{\boldsymbol{\beta}}, \widehat{\nu}_1, \ldots, \widehat{\nu}_G)$.

David Roxbee Cox (1924–) was born in Birmingham, England. He earned master's and PhD degrees in mathematics and statistics from the universities of Cambridge and Leeds, and he worked at the Royal Aircraft Establishment, the Wool Industries Research Association, and the universities of Cambridge, London (Birkbeck and Imperial Colleges), and Oxford. He was knighted in 1985. Sir David has worked on a wide range of theoretical and applied statistical problems, with outstanding contributions in areas such as experimental design, stochastic processes, binary data, survival analysis, asymptotic techniques, and multivariate dependencies. In 2010, Sir David was awarded the Copley Medal, the Royal Society's highest honor.

Acknowledgment

We thank Peter Sasieni of the Wolfson Institute of Preventive Medicine, London, for his statistical advice and guidance in implementing the robust variance estimator for this command.

References

Andersson, T. M.-L., and P. C. Lambert. 2012. Fitting and modeling cure in population-based cancer studies within the framework of flexible parametric survival models. *Stata Journal* 12: 623–638.

Box-Steffensmeier, J. M., and B. S. Jones. 2004. *Event History Modeling: A Guide for Social Scientists*. Cambridge: Cambridge University Press.

Breslow, N. E. 1974. Covariance analysis of censored survival data. *Biometrics* 30: 89–99.

Cleves, M. A. 1999. ssa13: Analysis of multiple failure-time data with Stata. *Stata Technical Bulletin* 49: 30–39. Reprinted in *Stata Technical Bulletin Reprints*, vol. 9, pp. 338–349. College Station, TX: Stata Press.

Cleves, M. A., W. W. Gould, R. G. Gutierrez, and Y. V. Marchenko. 2010. *An Introduction to Survival Analysis Using Stata*. 3rd ed. College Station, TX: Stata Press.

Cox, D. R. 1972. Regression models and life-tables (with discussion). *Journal of the Royal Statistical Society, Series B* 34: 187–220.

——. 1975. Partial likelihood. *Biometrika* 62: 269–276.

Cox, D. R., and D. Oakes. 1984. *Analysis of Survival Data*. London: Chapman & Hall/CRC.

Cox, D. R., and E. J. Snell. 1968. A general definition of residuals (with discussion). *Journal of the Royal Statistical Society, Series B* 30: 248–275.

Crowley, J., and M. Hu. 1977. Covariance analysis of heart transplant survival data. *Journal of the American Statistical Association* 72: 27–36.

Cui, J. 2005. Buckley–James method for analyzing censored data, with an application to a cardiovascular disease and an HIV/AIDS study. *Stata Journal* 5: 517–526.

Dupont, W. D. 2009. *Statistical Modeling for Biomedical Researchers: A Simple Introduction to the Analysis of Complex Data*. 2nd ed. Cambridge: Cambridge University Press.

Fleming, T. R., and D. P. Harrington. 1991. *Counting Processes and Survival Analysis*. New York: Wiley.

Gutierrez, R. G. 2002. Parametric frailty and shared frailty survival models. *Stata Journal* 2: 22–44.

Gutierrez, R. G., S. L. Carter, and D. M. Drukker. 2001. sg160: On boundary-value likelihood-ratio tests. *Stata Technical Bulletin* 60: 15–18. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 269–273. College Station, TX: Stata Press.

Hills, M., and B. L. De Stavola. 2012. *A Short Introduction to Stata for Biostatistics: Updated to Stata 12*. London: Timberlake.

Hinchliffe, S. R., D. A. Scott, and P. C. Lambert. 2013. Flexible parametric illness-death models. *Stata Journal* 13: 759–775.

Hosmer, D. W., Jr., S. A. Lemeshow, and S. May. 2008. *Applied Survival Analysis: Regression Modeling of Time to Event Data*. 2nd ed. New York: Wiley.

Jenkins, S. P. 1997. sbe17: Discrete time proportional hazards regression. *Stata Technical Bulletin* 39: 22–32. Reprinted in *Stata Technical Bulletin Reprints*, vol. 7, pp. 109–121. College Station, TX: Stata Press.

Kahn, H. A., and C. T. Sempos. 1989. *Statistical Methods in Epidemiology*. New York: Oxford University Press.

Kalbfleisch, J. D., and R. L. Prentice. 2002. *The Statistical Analysis of Failure Time Data*. 2nd ed. New York: Wiley.

Klein, J. P., and M. L. Moeschberger. 2003. *Survival Analysis: Techniques for Censored and Truncated Data*. 2nd ed. New York: Springer.

Lambert, P. C., and P. Royston. 2009. Further development of flexible parametric models for survival analysis. *Stata Journal* 9: 265–290.

Lin, D. Y., and L. J. Wei. 1989. The robust inference for the Cox proportional hazards model. *Journal of the American Statistical Association* 84: 1074–1078.

McGilchrist, C. A., and C. W. Aisbett. 1991. Regression with frailty in survival analysis. *Biometrics* 47: 461–466.

Newman, S. C. 2001. *Biostatistical Methods in Epidemiology*. New York: Wiley.

Parner, E. T., and P. K. Andersen. 2010. Regression analysis of censored data using pseudo-observations. *Stata Journal* 10: 408–422.

Peto, R. 1972. Contribution to the discussion of paper by D. R. Cox. *Journal of the Royal Statistical Society, Series B* 34: 205–207.

Reid, N. M. 1994. A conversation with Sir David Cox. *Statistical Science* 9: 439–455.

Royston, P. 2001. Flexible parametric alternatives to the Cox model, and more. *Stata Journal* 1: 1–28.

——. 2006. Explained variation for survival models. *Stata Journal* 6: 83–96.

——. 2007. Profile likelihood for estimation and confidence intervals. *Stata Journal* 7: 376–387.

Royston, P., and P. C. Lambert. 2011. *Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model*. College Station, TX: Stata Press.

Royston, P., and M. K. B. Parmar. 2002. Flexible parametric proportional-hazards and proportional-odds models for censored survival data, with application to prognostic modelling and estimation of treatment effects. *Statistics in Medicine* 21: 2175–2197.

Schoenfeld, D. A. 1982. Partial residuals for the proportional hazards regression model. *Biometrika* 69: 239–241.

Selvin, S. 2004. *Statistical Analysis of Epidemiologic Data*. 3rd ed. New York: Oxford University Press.

Sterne, J. A. C., and K. Tilling. 2002. G-estimation of causal effects, allowing for time-varying confounding. *Stata Journal* 2: 164–182.

Therneau, T. M., and P. M. Grambsch. 2000. *Modeling Survival Data: Extending the Cox Model*. New York: Springer.

Vittinghoff, E., D. V. Glidden, S. C. Shiboski, and C. E. McCulloch. 2012. *Regression Methods in Biostatistics: Linear, Logistic, Survival, and Repeated Measures Models*. 2nd ed. New York: Springer.

Also see

[ST] **stcox postestimation** — Postestimation tools for stcox

[ST] **stcurve** — Plot survivor, hazard, cumulative hazard, or cumulative incidence function

[ST] **stcox PH-assumption tests** — Tests of proportional-hazards assumption

[ST] **stcrreg** — Competing-risks regression

[ST] **streg** — Parametric survival models

[ST] **sts** — Generate, graph, list, and test the survivor and cumulative hazard functions

[ST] **stset** — Declare data to be survival-time data

[MI] **estimation** — Estimation commands for use with mi estimate

[SVY] **svy estimation** — Estimation commands for survey data

[U] 20 Estimation and postestimation commands