Description Remarks and examples References Also see

Description

Stata's survival analysis routines are used to compute sample size, power, and effect size and to declare, convert, manipulate, summarize, and analyze survival data. Survival data are time-to-event data, and survival analysis is full of jargon: truncation, censoring, hazard rates, etc. See the glossary in this manual. For a good Stata-specific introduction to survival analysis, see Cleves, Gould, and Marchenko (2016).

To learn how to effectively analyze survival analysis data using Stata, we recommend NetCourse 631, *Introduction to Survival Analysis Using Stata*; see https://www.stata.com/netcourse/nc631.html.

All the commands documented in this manual are listed below, and they are described in detail in their respective manual entries. While most commands for survival analysis are documented here, some are documented in other manuals. The commands for computing sample size, power, and effect size for survival analysis are documented in the *Stata Power, Precision, and Sample-Size Reference Manual* with the other `power` commands. The command for longitudinal or panel-data survival analysis is documented with the other panel-data commands in the *Stata Longitudinal-Data/Panel-Data Reference Manual*. The command for multilevel survival analysis is documented with the other multilevel commands in the *Stata Multilevel Mixed-Effects Reference Manual*. The commands for estimating treatment effects from observational survival-time data are documented in the *Stata Causal Inference and Treatment-Effects Estimation Reference Manual*. The commands for model selection and prediction using lasso and elastic net are documented in the *Stata Lasso Reference Manual*.

Declaring and converting count data

`ctset`	Declare data to be count-time data
`cttost`	Convert count-time data to survival-time data

Converting snapshot data

`snapspan`	Convert snapshot data to time-span data

Declaring and summarizing survival-time data

`stset`	Declare data to be survival-time data
`stdescribe`	Describe survival-time data
`stsum`	Summarize survival-time data

Manipulating survival-time data

`stvary`	Report variables that vary over time
`stfill`	Fill in by carrying forward values of covariates
`stgen`	Generate variables reflecting entire histories
`stsplit`	Split time-span records
`stjoin`	Join time-span records
`stbase`	Form baseline dataset

Obtaining summary statistics, confidence intervals, tables, etc.

sts	Generate, graph, list, and test the survivor and related functions
stir	Report incidence-rate comparison
stci	Confidence intervals for means and percentiles of survival time
strate	Tabulate failure rate
stptime	Calculate person-time, incidence rates, and SMR
stmh	Calculate rate ratios with the Mantel–Haenszel method
stmc	Calculate rate ratios with the Mantel–Cox method
ltable	Display and graph life tables

Fitting regression models

stcox	Cox proportional hazards model
estat concordance	Compute the concordance probability
estat phtest	Test Cox proportional-hazards assumption
stphplot	Graphically assess the Cox proportional-hazards assumption
stcoxkm	Graphically assess the Cox proportional-hazards assumption
streg	Parametric survival models
stintreg	Parametric models for interval-censored survival-time data
estat gofplot	Graphically assess goodness of fit after streg, stcox, stintreg, or stintcox
stintcox	Cox proportional hazards model for interval-censored data
stmgintcox	Marginal Cox PH model for interval-censored multiple-event data
estat common	Estimate average effects of covariates across all events
stintphplot	Graphically assess the Cox proportional-hazards assumption for interval-censored data
stintcoxnp	Graphically assess the Cox proportional-hazards assumption for interval-censored data
stcrreg	Competing-risks regression
xtstreg	Random-effects parametric survival models
mestreg	Multilevel mixed-effects parametric survival models
stcurve	Plot the survivor or related function after streg, stcox, and more
stteffects	Treatment-effects estimation for observational survival-time data
fmm: streg	Finite mixtures of parametric survival models
bayes: streg	Bayesian parametric survival models
bayes: mestreg	Bayesian multilevel parametric survival models

Prediction and model selection

lasso cox	Lasso selection of covariates in Cox proportional hazards models
elasticnet cox	Elastic net selection of covariates in Cox proportional hazards models

Sample size and power determination for survival analysis

power cox	Sample size, power, and effect size for the Cox proportional hazards model
power exponential	Sample size and power for the exponential test
power logrank	Sample size, power, and effect size for the log-rank test

Converting survival-time data

sttocc	Convert survival-time data to case–control data
sttoct	Convert survival-time data to count-time data

Programmer's utilities

st_*	Survival analysis subroutines for programmers

Remarks and examples

Remarks are presented under the following headings:

Introduction

All but one entry in this manual deals with the analysis of survival data, which is used to measure the time to an event of interest such as death or failure. Survival data can be organized in two ways. The first way is as *count data*, which refers to observations on populations, whether people or generators, with observations recording the number of units at a given time that failed or were lost because of censoring. The second way is as *survival-time*, or *time-span*, data. In survival-time data, the observations represent periods and typically contain three variables that record the start time of the period, the end time, and an indicator of whether failure or right-censoring occurred at the end of the period. The representation of the response of these three variables makes survival data unique in terms of implementing the statistical methods in the software. Such representation is specific to *right-censored survival-time data*. *Interval-censored survival-time data* are represented by two time variables that record the endpoints of time intervals in which failures are known to have occurred. Throughout the manual, when we refer to survival-time data, we will assume right-censored survival-time data. We will refer to interval-censored data explicitly.

Survival data may also be organized as *snapshot data* (a small variation of the survival-time format), in which observations depict an instance in time rather than an interval. When you have snapshot data, you simply use the snapspan command to convert it to survival-time data before proceeding.

Stata commands that begin with ct are used to convert count data to survival-time data. Survival-time data are analyzed using Stata commands that begin with st, known in our terminology as st commands. You can express all the information contained in count data in an equivalent survival-time dataset, but the converse is not true. Thus Stata commands are made to work with survival-time data because it is the more general representation.

All st commands, except stintreg, stintcox, and stmgintcox, are designed for right-censored survival-time data. The stintreg, stintcox, and stmgintcox commands analyze more general interval-censored survival-time (event-time) data.

Declaring and converting count data

Count data must first be converted to survival-time data before Stata's st commands can be used. Count data can be thought of as aggregated survival-time data. Rather than having observations that are specific to a subject and a period, you have data that, at each recorded time, record the number lost because of failure and, optionally, the number lost because of right-censoring.

ctset is used to tell Stata the names of the variables in your count data that record the time, the number failed, and the number censored. You ctset your data before typing cttost to convert it to survival-time data. Because you ctset your data, you can type cttost without any arguments to perform the conversion. Stata remembers how the data are ctset.

Converting snapshot data

Snapshot data are data in which each observation records the status of a given subject at a certain point in time. Usually you have multiple observations on each subject that chart the subject's progress through the study.

Before using Stata's survival analysis commands with snapshot data, you must first convert the data to survival-time data; that is, the observations in the data should represent intervals. When you convert snapshot data, the existing time variable in your data is used to record the end of a time span, and a new variable is created to record the beginning. Time spans are created using the recorded snapshot times as breakpoints at which new intervals are to be created. Before converting snapshot data to time-span data, you must understand the distinction between *enduring variables* and *instantaneous variables*. Enduring variables record characteristics of the subject that endure throughout the time span, such as sex or smoking status. Instantaneous variables describe events that occur at the end of a time span, such as failure or censoring. When you convert snapshots to intervals, enduring variables obtain their values from the previous recorded snapshot or are set to missing for the first interval. Instantaneous variables obtain their values from the current recorded snapshot because the existing time variable now records the end of the span.

Stata's snapspan makes this whole process easy. You specify an ID variable identifying your subjects, the snapshot time variable, the name of the new variable to hold the beginning times of the spans, and any variables that you want to treat as instantaneous variables. Stata does the rest for you.

Declaring and summarizing survival-time data

Stata does not automatically recognize survival-time data, so you must declare your survival-time data to Stata by using stset. Every st command, except stintreg, stintcox, and stmgintcox, relies on the information that is provided when you stset your data. Survival-time data come in different forms.

For example, your time variables may be dates, time measured from a fixed date, or time measured from some other point unique to each subject, such as enrollment in the study. You can also consider the following questions. What is the onset of risk for the subjects in your data? Is it time zero? Is it enrollment in the study or some other event, such as a heart transplant? Do you have censoring, and if so, which variable records it? What values does this variable record for censoring/failure? Do you have delayed entry? That is, were some subjects at risk of failure before you actually observed them? Do you have simple data and wish to treat everyone as entering and at risk at time zero?

Whatever the form of your data, you must first `stset` it before analyzing it, and so if you are new to Stata's st commands, we highly recommend that you take the time to learn about `stset`. It is really easy once you get the hang of it, and [ST] **stset** has many examples to help. For more discussion of `stset`, see Cleves, Gould, and Marchenko (2016, chap. 6).

Once you `stset` the data, you can use `stdescribe` to describe the aspects of your survival data. For example, you will see the number of subjects you were successful in declaring, the total number of records associated with these subjects, the total time at risk for these subjects, time gaps for any of these subjects, any delayed entry, etc. You can use `stsum` to summarize your survival data, for example, to obtain the total time at risk and the quantiles of time-to-failure in analysis-time units.

Manipulating survival-time data

Once your data have been `stset`, you may want to clean them up a bit before beginning your analysis. Suppose that you had an enduring variable and `snapspan` recorded it as missing for the interval leading up to the first recorded snapshot time. You can use `stfill` to fill in missing values of covariates, either by carrying forward the values from previous periods or by making the covariate equal to its earliest recorded (nonmissing) value for all time spans. You can use `stvary` to check for time-varying covariates or to confirm that certain variables, such as sex, are not time varying. You can use `stgen` to generate new covariates based on functions of the time spans for each given subject. For example, you can create a new variable called `eversmoked` that equals one for all of a subject's observations, if the variable `smoke` in your data is equal to one for *any* of the subject's time spans. Think of `stgen` as just a convenient way to do things that could be done using `by` *subject_id*`:` with survival-time data.

`stsplit` is useful for creating data that have multiple records per subject from data that have one record per subject. Suppose that you have already `stset` your data and wish to introduce a time-varying covariate. You would first need to `stsplit` your data so that separate time spans could be created for each subject, allowing the new covariate to assume different values over time within a subject. `stjoin` is the opposite of `stsplit`. Suppose that you have data with multiple records per subject but then realize that the data could be collapsed into single-subject records with no loss of information. Using `stjoin` would speed up any subsequent analysis using the st commands without changing the results.

`stbase` can be used to set every variable in your multiple-record st data to the value at baseline, defined as the earliest time at which each subject was observed. It can also be used to convert st data to cross-sectional data.

Obtaining summary statistics, confidence intervals, tables, etc.

Stata provides several commands for nonparametric analysis of survival data that can produce a wide array of summary statistics, inference, tables, and graphs. `sts` is a truly powerful command, used to obtain nonparametric estimates, inference, tests, and graphs of the survivor function, the cumulative hazard function, and the hazard function. You can compare estimates across groups, such as smoking

versus nonsmoking, and you can adjust these estimates for the effects of other covariates in your data. sts can present these estimates as tables and graphs. sts can also be used to test the equality of survivor functions across groups.

stir is used to estimate incidence rates and to compare incidence rates across groups. stci is the survival-time data analog of ci means and is used to obtain confidence intervals for means and percentiles of time to failure. strate is used to tabulate failure rates. stptime is used to calculate person-time and standardized mortality/morbidity ratios (SMRs). stmh calculates rate ratios by using the Mantel–Haenszel method, and stmc calculates rate ratios by using the Mantel–Cox method.

ltable displays and graphs life tables for individual-level or aggregate data.

Fitting regression models

Stata has commands for fitting both semiparametric and parametric regression models to survival data. stcox fits the Cox proportional hazards model and predict after stcox can be used to retrieve estimates of the baseline survivor function, the baseline cumulative hazard function, and the baseline hazard contributions. predict after stcox can also calculate a myriad of Cox regression diagnostic quantities, such as martingale residuals, efficient score residuals, and Schoenfeld residuals. stcox has four options for handling tied failures. stcox can be used to fit stratified Cox models, where the baseline hazard is allowed to differ over the strata, and it can be used to model multivariate survival data by using a *shared-frailty* model, which can be thought of as a Cox model with random effects. After stcox, you can use estat phtest to test the proportional-hazards assumption or estat concordance to compute the concordance probability. With stphplot and stcoxkm, you can graphically assess the proportional-hazards assumption.

stintcox fits the Cox proportional hazards model for interval-censored data, and stmgintcox fits the marginal Cox proportional hazards model for interval-censored multiple-event data. predict can be used after stintcox and stmgintcox to obtain estimates of the baseline survivor function, the baseline cumulative hazard function, and the baseline hazard contributions. predict can also calculate martingale-like residuals and Cox–Snell-like residuals after stintcox and stmgintcox. stintcox can be used to fit stratified Cox models, where the baseline hazard is allowed to differ over the strata; similarly, stmgintcox can fit stratified marginal Cox models. Additionally, you can use stintphplot and stintcoxnp to graphically assess the proportional-hazards assumption for your interval-censored data. You do not need to fit a model with stintcox or stmgintcox before using these graphical tools.

Stata offers six parametric regression models for survival data: exponential, Weibull, lognormal, loglogistic, Gompertz, and generalized gamma. All six models are fit using streg for right-censored data and stintreg for interval-censored data, and you can specify the model you want with the distribution() option. All of these models, except for the exponential, have ancillary parameters that are estimated (along with the linear predictor) from the data. By default, these ancillary parameters are treated as constant, but you may optionally model the ancillary parameters as functions of a linear predictor. Stratified models may also be fit using streg and stintreg. You can also fit frailty models with streg and specify whether you want the frailties to be treated as spell-specific or shared across groups of observations.

stcrreg fits a semiparametric regression model for survival data in the presence of competing risks. Competing risks impede the failure event under study from occurring. An analysis of such competing-risks data focuses on the *cumulative incidence function*, the probability of failure in the presence of competing events that prevent that failure. stcrreg provides an analogue to stcox for such data. The baseline *subhazard function*—that which generates failures under competing risks—is left unspecified, and covariates act multiplicatively on the baseline subhazard.

You can also fit parametric survival models to clustered and hierarchical or multilevel data by using the xtstreg or mestreg command, respectively.

xtstreg fits random-intercept parametric survival models to clustered survival data. Random intercepts are assumed to be normally distributed. A random-intercept model with Gaussian intercepts can be viewed as a shared-frailty model with lognormal frailty. xtstreg supports five distributions: exponential, loglogistic, Weibull, lognormal, and gamma, which you can specify using the distribution() option. Several predictions, such as mean, median, or survivor or hazard functions, can be obtained by using predict after fitting a model with xtstreg.

mestreg fits multilevel mixed-effects parametric survival models. It supports five distributions: exponential, loglogistic, Weibull, lognormal, and gamma, which you can specify using the distribution() option. mestreg allows for multiple levels of random effects and for random coefficients. Marginal or conditional predictions for several statistics and functions of interest, such as mean, median, or survival or hazard functions, can be obtained by using predict after fitting a model with mestreg.

In addition, you can perform treatment-effects estimation for observational survival-time data by using stteffects. stteffects estimates average treatment effects, average treatment effects on the treated, and potential-outcome means using observational survival-time data. The available estimators are regression adjustment, inverse-probability weighting, and double-robust methods that combine regression adjustment and inverse-probability weighting; see [CAUSAL] **stteffects intro** for details.

stcurve plots the survivor, failure, hazard, or cumulative hazard function after stcox, streg, stintreg, stintcox, stmgintcox, stcrreg, mestreg, or xtstreg. stcurve also plots the cumulative subhazard or cumulative incidence function after stcrreg. Covariates, by default, are held fixed at their mean values, but you can specify other values if you wish. stcurve is useful for comparing these functions across different levels of covariates.

estat gofplot creates a goodness of fit plot after streg, stcox, stintreg, stintcox, or stmgintcox. This graph consists of the estimated cumulative hazard function for the Cox–Snell residuals plotted against the residuals themselves.

Prediction and model selection

Stata provides commands to select covariates and fit models using lasso and elastic net. Lasso is a solution to a penalized optimization problem, where the penalty is used to force some covariated to be omitted from the model. For more information on the lasso penalty, see [LASSO] **lasso**. Elastic net is similar to lasso; it uses a different penalty that performs better when groups of variables that are highly correlated. The results from lasso and elastic net are useful for prediction and model selection.

lasso cox selects covariates using lasso and fits a Cox proportional hazards model. After this command, predictions are available by using the standard predict postestimation command.

elasticnet cox selects covariates using elastic net and fits a Cox proportional hazards model. After this command, predictions are available by using the standard predict postestimation command.

Sample size and power determination for survival analysis

Stata has commands for computing sample size, power, and effect size for survival analysis using the log-rank test, the Cox proportional hazards model, and the exponential test comparing exponential hazard rates.

power logrank computes sample size, power, or effect size for survival analysis comparing survivor functions in two groups by using the log-rank test. The command supports unbalanced designs and provides options to account for administrative censoring, uniform accrual, and withdrawal of subjects from the study.

power cox computes sample size, power, or effect size for survival analyses that use Cox proportional hazards (PH) models. The results are obtained for the test of the effect of one covariate (binary or continuous) on time to failure adjusted for other predictors in a PH model. The command can account for the dependence between the covariate of interest and other model covariates, and it can adjust computations for censoring and for withdrawal of subjects for the study.

power exponential computes sample size or power for survival analysis comparing two exponential survivor functions by using parametric tests for the difference between hazards or, optionally, for the difference between log hazards. It accommodates unequal allocation between the two groups, flexible accrual of subjects into the study, and group-specific losses to follow-up. The accrual distribution may be chosen to be uniform or truncated exponential over a fixed accrual period.

The commands allow automated production of customizable tables and graphs; see [PSS-2] **power** for details.

Converting survival-time data

Stata has commands for converting survival-time data to case–control and count data. These commands are rarely used, because most of the analyses are performed using data in the survival-time format. sttocc is useful for converting survival data to case–control data suitable for estimation with clogit. sttoct is the opposite of cttost and will convert survival-time data to count data.

Programmer's utilities

Stata also provides routines for programmers interested in writing their own st commands. These are basically utilities for setting, accessing, and verifying the information saved by stset. For example, st_is verifies that the data have in fact been stset and gives the appropriate error if not. st_show is used to preface the output of a program with key information on the st variables used in the analysis. Programmers interested in writing st code should see [ST] **st_is**.

References

Bower, H., T. M.-L. Andersson, M. J. Crowther, and P. C. Lambert. 2022. Flexible parametric survival analysis with multiple timescales: Estimation and implementation using stmt. *Stata Journal* 22: 679–701.

Cleves, M. A., W. W. Gould, and Y. V. Marchenko. 2016. *An Introduction to Survival Analysis Using Stata*. Rev. 3rd ed. College Station, TX: Stata Press.

Crowther, M. J. 2022. Simulating time-to-event data from parametric distributions, custom distributions, competing-risks models, and general multistate models. *Stata Journal* 22: 3–24.

Also see

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

[ST] **Intro** — Introduction to survival analysis manual

Stata Power, Precision, and Sample-Size Reference Manual

Stata Meta-Analysis Reference Manual