Title

stata.com

stteffects wra — Survival-time weighted regression adjustment

Description

`stteffects wra` estimates the average treatment effect (ATE), the average treatment effect on the treated (ATET), and the potential-outcome means (POMs) from observational survival-time data with random time to censoring. Estimation is by weighted regression adjustment (WRA). WRA estimators use inverse-probability-of-censoring adjusted regression coefficients to compute averages of treatment-level predicted outcomes. Contrasts of these averages estimate the treatment effects. WRA uses estimated weights from a time-to-censoring model to account for censored survival times instead of including a term in the likelihood function. `stteffects wra` offers several choices for the functional forms of the outcome model and the time-to-censoring model. Binary and multivalued treatments are accommodated.

See [TE] **stteffects intro** for an overview of estimating treatment effects from observational survival-time data.

Quick start

Specify `time` as observed failure time and `fail` as failure indicator

```
stset time, failure(fail)
```

ATE from a Weibull model for `time` on `x1` and `x2` with binary treatment `treat2` and a Weibull model on `x1` and `x2` for censoring

```
stteffects wra (x1 x2) (treat2) (x1 x2)
```

As above, but estimate the ATET

```
stteffects wra (x1 x2) (treat2) (x1 x2), atet
```

ATE of `treat2` using a gamma model for `time` and a gamma censoring model

```
stteffects wra (x1 x2, gamma) (treat2) (x1 x2, gamma)
```

ATE for each level of three-valued treatment `treat3`

```
stteffects wra (x1 x2) (treat3) (x1 x2)
```

As above, and specify that `treat3` = 3 is the control level using the value label "MyControl" for 3

```
stteffects wra (x1 x2) (treat3) (x1 x2), control("MyControl")
```

Menu

Statistics > Treatment effects > Survival outcomes > Weighted regression adjustment

Syntax

```
steffects wra (omvarlist [, omoptions]) (tvar) (cmvarlist [, cmoptions])
    [if] [in] [, stat options]
```

omvarlist specifies the variables that predict the survival-time variable in the outcome model.

tvar must contain integer values representing the treatment levels.

cmvarlist specifies the variables that predict censoring in the censoring model.

omoptions	Description
Model	
weibull	Weibull; the default
exponential	exponential
gamma	two-parameter gamma
lnormal	lognormal
ancillary(*avarlist* [, noconstant])	specify variables used to model ancillary parameter
noconstant	suppress constant from outcome model

cmoptions	Description
Model	
weibull	Weibull; the default
exponential	exponential
gamma	two-parameter gamma
lnormal	lognormal
ancillary(*avarlist* [, noconstant])	specify variables used to model ancillary parameter
noconstant	suppress constant from censoring model

stat	Description
Stat	
ate	estimate average treatment effect in population; the default
atet	estimate average treatment effect on the treated
pomeans	estimate potential-outcome means

options	Description
SE/Robust	
`vce(`*vcetype*`)`	*vcetype* may be `robust`, `cluster` *clustvar*, `bootstrap`, or `jackknife`
Reporting	
`level(#)`	set confidence level; default is `level(95)`
`aequations`	display auxiliary-equation results
`noshow`	do not show st setting information
display_options	control columns and 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
`iterinit(#)`	specify starting-value iterations; seldom used
Advanced	
`pstolerance(#)`	set tolerance for the overlap assumption
`osample(`*newvar*`)`	identify observations that violate the overlap assumption
`control(#` | *label*`)`	specify the level of *tvar* that is the control
`tlevel(#` | *label*`)`	specify the level of *tvar* that is the treatment
`coeflegend`	display legend instead of statistics

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

omvarlist, *cmvarlist*, and *avarlist* may contain factor variables; see [U] **11.4.3 Factor variables**.

`bootstrap`, `by`, `collect`, `jackknife`, and `statsby` are allowed; see [U] **11.1.10 Prefix commands**.

Weights are not allowed with the `bootstrap` prefix; see [R] **bootstrap**.

`fweights`, `iweights`, and `pweights` may be specified using `stset`; see *Weights* under *Remarks and examples* in [ST] **stset**. However, weights may not be specified if you are using the `bootstrap` prefix.

`coeflegend` does not appear in the dialog box.

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

Options

Model

`ancillary(`*avarlist* [`, noconstant`]`)` specifies the variables used to model the ancillary parameter. By default, the ancillary parameter does not depend on covariates. Specifying `ancillary(`*avarlist*`, noconstant)` causes the constant to be suppressed in the model for the ancillary parameter.

`ancillary()` may be specified for the model for survival-time outcome, for the model for the censoring variable, or for both. If `ancillary()` is specified for both, the varlist used for each model may be different.

`noconstant`; see [R] **Estimation options**.

Stat

stat is one of three statistics: `ate`, `atet`, or `pomeans`. `ate` is the default.

`ate` specifies that the average treatment effect be estimated.

`atet` specifies that the average treatment effect on the treated be estimated.

`pomeans` specifies that the potential-outcome means for each treatment level be estimated.

SE/Robust

`vce(`*vcetype*`)` specifies the type of standard error reported, which includes types 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***.

Reporting

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

`aequations` specifies that the results for the outcome-model or treatment-model parameters be displayed. By default, the results for these auxiliary parameters are not displayed.

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

display_options: `noci`, `nopvalues`, `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`, and `from(`*init_specs*`)`; see [R] **Maximize**. These options are seldom used.

init_specs is one of

matname `[, skip copy]`

`[, #` ... `], copy`

`iterinit(#)` specifies the maximum number of iterations used to calculate the starting values. This option is seldom used.

Advanced

`pstolerance(#)` specifies the tolerance used to check the overlap assumption. The default value is `pstolerance(1e-5)`. `stteffects` will exit with an error if an observation has an estimated propensity score smaller than that specified by `pstolerance()`.

`osample(`*newvar*`)` specifies that indicator variable *newvar* be created to identify observations that violate the overlap assumption.

`control(#|`*label*`)` specifies the level of *tvar* that is the control. The default is the first treatment level. You may specify the numeric level *#* (a nonnegative integer) or the label associated with the numeric level. `control()` may not be specified with the statistic `pomeans`. `control()` and `tlevel()` may not specify the same treatment level.

`tlevel(#|label)` specifies the level of *tvar* that is the treatment for the statistic `atet`. The default is the second treatment level. You may specify the numeric level # (a nonnegative integer) or the label associated with the numeric level. `tlevel()` may only be specified with statistic `atet`. `tlevel()` and `control()` may not specify the same treatment level.

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

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

Remarks and examples

stata.com

If you are not familiar with the framework for treatment-effects estimation from observational survival-time data, please see [TE] **stteffects intro**.

Weighted regression-adjustment (WRA) estimators use estimated weights to account for censoring when estimating outcome-regression parameters. The estimated outcome-regression parameters are used to compute averages of treatment-level predicted outcomes. Contrasts of these averages estimate the treatment effects.

WRA estimators use a three-step approach to estimating treatment effects:

1. They estimate the parameters of a time-to-censoring model and compute inverse-probability-of-censoring weights.
2. Using the estimated inverse-probability-of-censoring weights, they use weighted maximum likelihood estimators for the outcome for each treatment level and obtain the treatment-specific predicted mean outcomes for each subject. The inverse-probability-of-censoring weights account for right-censored survival times.
3. They compute the means of the treatment-specific predicted mean outcomes. Contrasts of these averages provide the estimates of the ATEs. By restricting the computations of the means to the subset of treated subjects, we can obtain the ATETs.

WRA estimators differ from RA estimators in that WRA estimators use weights to account for observations lost to censoring while RA estimators use an additional term in the likelihood function. A model for the time to censoring is used to estimate the weights.

WRA estimators require more assumptions than RA estimators. Specifically, they require that the censoring time be random and that the time-to-censoring model be well specified. The implemented WRA estimators also require that the time-to-censoring process not vary by treatment level. The RA estimator and the likelihood-adjusted-censoring version of the inverse-probability-weighted RA estimator do not require these extra assumptions, because they use a likelihood term instead of weights to adjust for the data lost to censoring; see [TE] **stteffects ra** and [TE] **stteffects ipwra**.

Here we note only a few entry points to the vast literature on weighted estimators. Imbens (2004), Imbens and Wooldridge (2009), Robins and Rotnitzky (2006), Wooldridge (2002, 2007), Cameron and Trivedi (2005, chap. 25), Wooldridge (2010, chap. 21), and Vittinghoff et al. (2012, chap. 9) provide excellent general introductions to estimating ATEs and to WRA estimators in particular.

Like `streg` and other survival-time commands, `stteffects wra` uses the outcome variable and the failure indicator computed by, and optionally weights specified with, `stset`. `stteffects wra` is not appropriate for data with time-varying covariates, also known as multiple-record survival-time data, or for delayed-entry data.

Example 1: Estimating the ATE

Suppose we wish to study the effect of smoking on the time to a second heart attack among women aged 45–55 years. In our fictional `sheart` dataset, `atime` is the observed time in years to a second heart attack or censoring, and `fail` is the 0/1 indicator that a second heart attack was observed. (When `fail` is 1, `atime` records the time to the second heart attack; when `fail` is 0, `atime` records a censored observation of the time to a second heart attack.) We previously `stset` these data; see *A quick tour of the estimators* in [TE] **stteffects intro**.

The treatment, smoking, is stored in the 0/1 indicator `smoke`. These data also contain age at the time of the first heart attack (`age`), and indices of the level of exercise (`exercise`), diet quality (`diet`), and education (`education`) prior to the first heart attack.

We can use `stteffects wra` to estimate the ATE by WRA. We model the mean survival time using the default Weibull outcome model with `age`, `exercise`, `diet`, and `education` as covariates, and we specify that `smoke` is the treatment variable. We also specify the default Weibull time-to-censoring model and include `age`, square of `age`, `exercise`, and `education`.

```
. use https://www.stata-press.com/data/r17/sheart
(Time to second heart attack (fictional))
. stteffects wra  (age exercise diet education)
>                 (smoke)
>                 (age c.age#c.age exercise diet education)
        Failure _d: fail
  Analysis time _t: atime
Iteration 0:   EE criterion =  4.096e-18
Iteration 1:   EE criterion =  1.406e-29
Survival treatment-effects estimation           Number of obs     =      2,000
Estimator      : weighted regression adjustment
Outcome model  : Weibull
Treatment model: none
Censoring model: Weibull
```

_t	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
ATE						
smoke (Smoker vs Nonsmoker)	-2.374174	.6017498	-3.95	0.000	-3.553582	-1.194766
POmean						
smoke Nonsmoker	4.302131	.5528943	7.78	0.000	3.218478	5.385784

When every woman smoked in the population of women aged 45–55 years who have had a heart attack, the average time to a second heart attack is estimated to be 2.37 years less than when no women in the subpopulation of interest smoked. The estimated average time to a second heart attack when no women in the subpopulation of interest smoked is 4.30 years.

◁

Stored results

stteffects wra stores the following in e():

Scalars

e(N)	number of observations
e(n*j*)	number of observations for treatment level *j*
e(N_clust)	number of clusters
e(k_eq)	number of equations in e(b)
e(k_levels)	number of levels in treatment variable
e(treated)	level of treatment variable defined as treated
e(control)	level of treatment variable defined as control
e(converged)	1 if converged, 0 otherwise

Macros

e(cmd)	stteffects
e(cmdline)	command as typed
e(dead)	_d
e(depvar)	_t
e(tvar)	name of treatment variable
e(subcmd)	wra
e(omodel)	outcome model: weibull, exponential, gamma, or lognormal
e(cmodel)	censoring model: weibull, exponential, gamma, or lognormal
e(stat)	statistic estimated: ate, atet, or pomeans
e(wtype)	weight type
e(wexp)	weight expression
e(title)	title in estimation output
e(clustvar)	name of cluster variable
e(tlevels)	levels of treatment variable
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. err.
e(properties)	b V
e(estat_cmd)	program used to implement estat
e(predict)	program used to implement predict
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 estimators

Functions

e(sample)	marks estimation sample

In addition to the above, the following is stored in r():

Matrices

r(table)	matrix containing the coefficients with their standard errors, test statistics, *p*-values, and confidence intervals

Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Methods and formulas

The methods and formulas for the WRA estimators implemented in stteffects wra are given in *Methods and formulas* of [TE] **stteffects ipwra**.

References

Angrist, J. D., and J.-S. Pischke. 2009. *Mostly Harmless Econometrics: An Empiricist's Companion*. Princeton, NJ: Princeton University Press.

Bai, X., A. A. Tsiatis, and S. M. O'Brien. 2013. Doubly robust estimators of treatment-specific survival distributions in observational studies with stratified sampling. *Biometrics* 69: 830–839. https://doi.org/10.1111/biom.12076.

Cameron, A. C., and P. K. Trivedi. 2005. *Microeconometrics: Methods and Applications*. New York: Cambridge University Press.

Cattaneo, M. D. 2010. Efficient semiparametric estimation of multi-valued treatment effects under ignorability. *Journal of Econometrics* 155: 138–154. https://doi.org/10.1016/j.jeconom.2009.09.023.

Cattaneo, M. D., D. M. Drukker, and A. D. Holland. 2013. Estimation of multivalued treatment effects under conditional independence. *Stata Journal* 13: 407–450.

Guo, S., and M. W. Fraser. 2015. *Propensity Score Analysis: Statistical Methods and Applications*. 2nd ed. Thousand Oaks, CA: SAGE.

Imbens, G. W. 2004. Nonparametric estimation of average treatment effects under exogeneity: A review. *Review of Economics and Statistics* 86: 4–29. https://doi.org/10.1162/003465304323023651.

Imbens, G. W., and J. M. Wooldridge. 2009. Recent developments in the econometrics of program evaluation. *Journal of Economic Literature* 47: 5–86. https://doi.org/10.1257/jel.47.1.5.

Robins, J. M., and A. Rotnitzky. 2006. Inverse probability weighting in survival analysis. In *Survival and Event History Analysis*, ed. N. Keiding and P. K. Andersen, 266–271. Chichester, UK: Wiley.

Rosenbaum, P. R., and D. B. Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika* 70: 41–55. https://doi.org/10.2307/2335942.

Rubin, D. B. 1974. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology* 66: 688–701. https://doi.org/10.1037/h0037350.

Tsiatis, A. A. 2006. *Semiparametric Theory and Missing Data*. 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.

Wooldridge, J. M. 2002. Inverse probability weighted M-estimators for sample selection, attrition, and stratification. *Portuguese Economic Journal* 1: 117–139. https://doi.org/10.1007/s10258-002-0008-x.

——. 2007. Inverse probability weighted estimation for general missing data problems. *Journal of Econometrics* 141: 1281–1301. https://doi.org/10.1016/j.jeconom.2007.02.002.

——. 2010. *Econometric Analysis of Cross Section and Panel Data*. 2nd ed. Cambridge, MA: MIT Press.

Also see

[TE] **stteffects postestimation** — Postestimation tools for stteffects

[TE] **stteffects intro** — Introduction to treatment effects for observational survival-time data

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

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

[U] **20 Estimation and postestimation commands**