

tebalance density — Covariate balance density

Description	Quick start	Menu	Syntax
Options	Remarks and examples	Stored results	Reference
Also see			

Description

`tebalance density` produces kernel density plots that are used to check for covariate balance after estimation by a `teffects` inverse-probability-weighted estimator, a `teffects` matching estimator, or an `stteffects` inverse-probability-weighted estimator.

Quick start

- Kernel density plot of the propensity score after `teffects psmatch`
`tebalance density`
- Kernel density plot of `x1` after a `teffects` command or an `stteffects` command
`tebalance density x1`
- As above, but rescale the kernel bandwidth by a factor of 2
`tebalance density x1, bwidth(*2)`

Menu

Statistics > Treatment effects > Balance > Graphs

Syntax

Density plots for the propensity score

```
tebalance density [ , options ]
```

Density plots for a covariate

```
tebalance density varname [ , options ]
```

<i>options</i>	Description
<hr/>	
Main	
<code>kernel(<i>kernel</i>)</code>	specify the kernel function; default is <code>kernel(epanechnikov)</code>
<code>bwidth(*#)</code>	rescale default bandwidth
<code>line#opts(<i>line_options</i>)</code>	twoway line options for density line number #
<code>twoway_options</code>	any options other than <code>by()</code> documented in [G-3] <i>twoway_options</i>
<code>byopts(<i>byopts</i>)</code>	how subgraphs are combined, labeled, etc.

collect is allowed; see [\[U\] 11.1.10 Prefix commands](#).

<i>kernel</i>	Description
<hr/>	
<code>triangle</code>	triangle kernel function; the default
<code>epanechnikov</code>	Epanechnikov kernel function
<code>epan2</code>	alternative Epanechnikov kernel function
<code>biweight</code>	biweight kernel function
<code>cosine</code>	cosine trace kernel function
<code>gaussian</code>	Gaussian kernel function
<code>parzen</code>	Parzen kernel function
<code>rectangle</code>	rectangle kernel function

Options

Main

`kernel(kernel)` specifies the kernel function for use in calculating the kernel density estimates. The default kernel is the `kernel(epanechnikov)`.

`bwidth(*#)` specifies the factor by which the default bandwidths are to be rescaled. A bandwidth is the half-width of the kernel, the width of the density window around each point. Each kernel density plot has its own bandwidth, and by default, each kernel density plot uses its own optimal bandwidth; see [\[R\] **kdensity**](#). `bwidth()` rescales each plot's optimal bandwidth by the specified amount.

`line#opts(line_options)` specifies the line pattern, width, color, and overall style of density line number #. The line numbers are in the same order as the treatment levels specified in `e(tlevels)`.

`twoway_options` are any of the options documented in [\[G-3\] *twoway_options*](#), excluding `by()`. These include options for titling the graph (see [\[G-3\] *title_options*](#)) and for saving the graph to disk (see [\[G-3\] *saving_option*](#)). `tebalance density` uses `by()` to differentiate between raw and weighted or matched samples, and some `twoway_options` will be repeated for `by` graph and might be better specified as `byopts()`.

`byopts(by_option)` is as documented in [G-3] [by_option](#). `byopts()` affects how the subgraphs are combined, labeled, etc. `byopts()` generally affects the entire graph, and some *by_option* may be better specified as *twoway_options*; see [G-3] [twoway_options](#).

Remarks and examples

[stata.com](#)

When the distribution of a covariate does not vary over the treatment levels, the covariate is said to be balanced. `tebalance density` produces kernel density plots of a covariate over treatment levels for the raw data and the weighted or matched sample produced by `teffects` or `stteffects`. If the weighted-sample or matched-sample kernel density plots of the covariate are the same over the treatment levels, the covariate is balanced in the weighted or matched sample.

After all `teffects` commands except `teffects ra`, `stteffects ipw`, and `stteffects ipwra`,

```
. tebalance density varname [, options]
```

produces kernel density plots to check whether *varname* is balanced.

After all `teffects` commands except `teffects ra`, `teffects nnmatch`, `stteffects ipw`, and `stteffects ipwra`,

```
. tebalance density [, options]
```

produces kernel density plots to check whether the propensity score estimated by `teffects` or `teffects` is balanced. Our discussion of the use of `tebalance density` and interpretation of its results for a covariate below also apply to a propensity score.

We recommend that you read [TE] [tebalance](#) before proceeding; it provides an introduction to covariate balance and an overview of the implemented methods. See [TE] [stteffects intro](#) for a discussion of survival-time features.

► Example 1: Checking covariate balance after `psmatch`

Using an extract from the data used by Cattaneo (2010), we use `teffects psmatch` to estimate the effect of a mother's smoking behavior (`mbsmoke`) on the birthweight of her child (`bweight`), controlling for marital status (`mmarried`), the mother's age (`mage`), whether the mother had a prenatal doctor's visit in the baby's first trimester (`prenatal1`), and whether this baby is the mother's first child (`fbaby`).

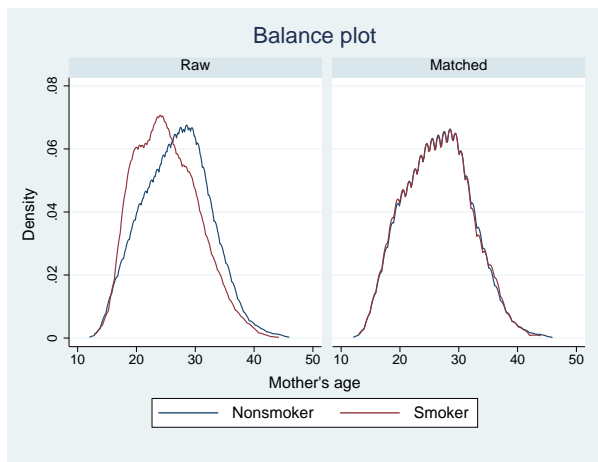
```
. use https://www.stata-press.com/data/r17/cattaneo2
(Excerpt from Cattaneo (2010) Journal of Econometrics 155: 138-154)
. teffects psmatch (bweight) (mbsmoke mmarried mage prenatal1 fbaby),
> generate(matchv)

Treatment-effects estimation      Number of obs      =      4,642
Estimator      : propensity-score matching      Matches: requested =      1
Outcome model  : matching                      min =      1
Treatment model: logit                      max =     139
```

bweight	AI robust		z	P> z	[95% conf. interval]	
	Coefficient	std. err.				
ATE						
mbsmoke						
(Smoker						
vs						
Nonsmoker)	-235.1714	27.74409	-8.48	0.000	-289.5488	-180.794

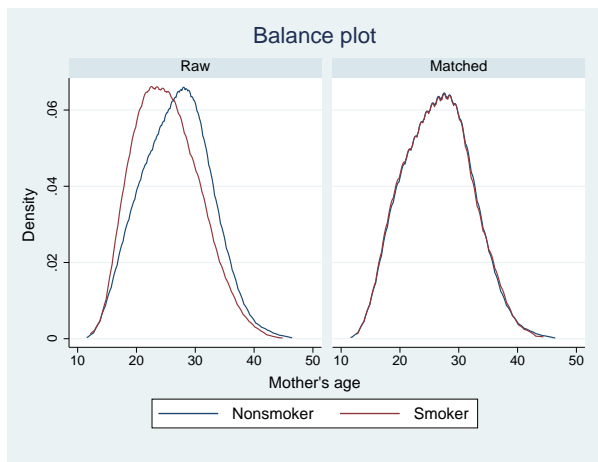
We specified the option `generate(matchv)` to speed up the `postestimation` command that produces density plots, as discussed in [example 3](#) under *Remarks and examples* in [\[TE\] tebalance](#) entry. We do not interpret the estimated effect produced by this preliminary model but rather check the specification. We begin by looking at the default density plots.

```
. tebalance density mage
```



The density plots for the matched sample are nearly indistinguishable, implying that matching on the estimated propensity score balanced the covariates. The density plots are too jagged for presentation, so we oversmooth them by scaling up the bandwidth used for each plot.

```
. tebalance density mage, bwidth(*1.5)
```



Option `bwidth()` rescales the default optimal bandwidths by the specified scale factor. Each of the four density plots has its own sample size and optimal bandwidth. Rescaling each of the four bandwidths by 1.5 produces smoother plots.

Stored results

After `teffects` or `stteffects` fits a binary treatment, `tebalance density` stores the following in `r()`:

Scalars

<code>r(bwc_adj)</code>	bandwidth for control in weighted or matched-adjusted sample
<code>r(Nc_adj)</code>	observations on control in weighted or matched-adjusted sample
<code>r(bwt_adj)</code>	bandwidth for treated in weighted or matched-adjusted sample
<code>r(Nt_adj)</code>	observations on treated in weighted or matched-adjusted sample
<code>r(bwc_raw)</code>	bandwidth for control in raw sample
<code>r(Nc_raw)</code>	observations on control in raw sample
<code>r(bwt_raw)</code>	bandwidth for treated in raw sample
<code>r(Nt_raw)</code>	observations on treated in raw sample

Macros

<code>r(kernel)</code>	name of kernel
------------------------	----------------

After `teffects` or `stteffects` fits a multivalued treatment, `tebalance density` stores the following in `r()`:

Scalars

<code>r(bw#_adj)</code>	bandwidth for treatment level # in weighted or matched-adjusted sample
<code>r(N#_adj)</code>	observations on treatment level # in weighted or matched-adjusted sample
<code>r(bw#_raw)</code>	bandwidth for treatment level # in raw sample
<code>r(N#_raw)</code>	observations on treatment level # in raw sample

Macros

<code>r(kernel)</code>	name of kernel
------------------------	----------------

Reference

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>.

Also see

- [TE] [stteffects intro](#) — Introduction to treatment effects for observational survival-time data
- [TE] [tebalance](#) — Check balance after `teffects` or `stteffects` estimation
- [TE] [teffects aipw](#) — Augmented inverse-probability weighting
- [TE] [teffects ipw](#) — Inverse-probability weighting
- [TE] [teffects ipwra](#) — Inverse-probability-weighted regression adjustment
- [TE] [teffects nnmatch](#) — Nearest-neighbor matching
- [TE] [teffects psmatch](#) — Propensity-score matching
- [TE] [teoverlap](#) — Overlap plots