

[Postestimation commands](#)

[Remarks and examples](#)

[Also see](#)

[predict](#)

[Stored results](#)

[estat](#)

[Methods and formulas](#)

[procoverlay](#)

[References](#)

Postestimation commands

The following postestimation commands are of special interest after `procrustes`:

| Command | Description |
|---------------------------------|--|
| estat compare | fit statistics for orthogonal, oblique, and unrestricted transformations |
| estat mvreg | display multivariate regression resembling unrestricted transformation |
| estat summarize | display summary statistics over the estimation sample |
| procoverlay | produce a Procrustes overlay graph |

The following standard postestimation commands are also available:

| Command | Description |
|-----------------------------|-------------------------------|
| * estimates | cataloging estimation results |
| predict | predictions and residuals |

*All estimates subcommands except `table` and `stats` are available; see [\[R\] estimates](#).

predict

Description for predict

predict creates new variables containing predictions such as fitted values, unstandardized residuals, and residual sum of squares.

Menu for predict

Statistics > Postestimation

Syntax for predict

```
predict [type] {stub* | newvarlist} [if] [in], [statistic]
```

| statistic | Description |
|-----------|--|
| fitted | fitted values $\mathbf{1} \mathbf{c}' + \boldsymbol{\rho} \mathbf{X} \mathbf{A}$; the default (specify # _y vars) |
| residuals | unstandardized residuals (specify # _y vars) |
| q | residual sum of squares over the target variables (specify one var) |

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

Options for predict

Main

fitted, the default, computes fitted values, that is, the least-squares approximations of the target (*varlist_y*) variables. You must specify the same number of new variables as there are target variables.

residuals computes the raw (unstandardized) residuals for each target (*varlist_y*) variable. You must specify the same number of new variables as there are target variables.

q computes the residual sum of squares over all variables, that is, the squared Euclidean distance between the target and transformed source points. Specify one new variable.

estat

Description for estat

`estat compare` displays a table with fit statistics of the three transformations provided by `procrustes`: orthogonal, oblique, and unrestricted. The two additional `procrustes` analyses are performed on the same sample as the original `procrustes` analysis and with the same options. F tests comparing the models are provided.

`estat mvreg` produces the `mvreg` (see [MV] [mvreg](#)) output related to the unrestricted Procrustes analysis (the `transform(unrestricted)` option of `procrustes`).

`estat summarize` displays summary statistics over the estimation sample of the target and source variables ([varlist_y](#) and [varlist_x](#)).

Menu for estat

Statistics > Postestimation

Syntax for estat

Table of fit statistics

```
estat compare [ , detail ]
```

Comparison of `mvreg` and `procrustes` output

```
estat mvreg [ , mvreg_options ]
```

Display summary statistics

```
estat summarize [ , labels noheader noweights ]
```

`collect` is allowed with `estat compare` and `estat summarize`; see [U] [11.1.10 Prefix commands](#).

Options for estat

`detail`, an option with `estat compare`, displays the standard `procrustes` output for the two additional transformations.

mvreg_options, allowed with `estat mvreg`, are any of the options allowed by `mvreg`; see [MV] [mvreg](#).

The constant is already suppressed if the Procrustes analysis suppressed it.

`labels`, `noheader`, and `noweights` are the same as for the generic `estat summarize` command; see [R] [estat summarize](#).

procoverlay

Description for procoverlay

procoverlay displays a plot of the target variables overlaid with the fitted values derived from the source variables. If there are more than two target variables, multiple plots are shown in one graph.

Menu for procoverlay

Statistics > Multivariate analysis > Procrustes overlay graph

Syntax for procoverlay

| procoverlay [<i>if</i>] [<i>in</i>] [, <i>procoverlay_options</i>] | |
|--|---|
| <i>procoverlay_options</i> | Description |
| Main | |
| <u>autoaspect</u> | adjust aspect ratio on the basis of the data; default aspect ratio is 1 |
| <u>targetopts</u> (<i>target_opts</i>) | affect the rendition of the target |
| <u>sourceopts</u> (<i>source_opts</i>) | affect the rendition of the source |
| Y axis, X axis, Titles, Legend, Overall | |
| <i>twoway_options</i> | any options other than by() documented in [G-3] <i>twoway_options</i> |
| By | |
| <u>byopts</u> (<i>by_option</i>) | affect the rendition of combined graphs |
| <i>target_opts</i> | Description |
| Main | |
| <u>no</u> label | removes the default observation label from the target |
| <i>marker_options</i> | change look of markers (color, size, etc.) |
| <i>marker_label_options</i> | change look or position of marker labels |
| <i>source_opts</i> | Description |
| Main | |
| <u>no</u> label | removes the default observation label from the source |
| <i>marker_options</i> | change look of markers (color, size, etc.) |
| <i>marker_label_options</i> | change look or position of marker labels |

Options for procoverlay

| |
|---|
| Main |
| autoaspect specifies that the aspect ratio be automatically adjusted based on the range of the data to be plotted. This option can make some procoverlay plots more readable. By default, procoverlay uses an aspect ratio of one, producing a square plot. |

As an alternative to `autoaspect`, the *twoway_option* `aspectratio()` can be used to override the default aspect ratio. `procoverlay` accepts the `aspectratio()` option as a suggestion only and will override it when necessary to produce plots with balanced axes, that is, where distance on the *x* axis equals distance on the *y* axis.

twoway_options, such as `xlabel()`, `yscale()`, `ylabel()`, and `yscale()`, should be used with caution. These *axis_options* are accepted but may have unintended side effects on the aspect ratio. See [G-3] *twoway_options*.

`targetopts(target_opts)` affects the rendition of the target plot. The following *target_opts* are allowed:

`no label` removes the default target observation label from the graph.

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] *marker_options*.

marker_label_options specify if and how the markers are to be labeled; see [G-3] *marker_label_options*.

`sourceopts(source_opts)` affects the rendition of the source plot. The following *source_opts* are allowed:

`no label` removes the default source observation label from the graph.

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] *marker_options*.

marker_label_options specify if and how the markers are to be labeled; see [G-3] *marker_label_options*.

Y axis, X axis, Titles, Legend, Overall

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*). See `autoaspect` above for a warning against using options such as `xlabel()`, `yscale()`, `ylabel()`, and `yscale()`.

By

`byopts(by_option)` is documented in [G-3] *by_option*. This option affects the appearance of the combined graph and is ignored, unless there are more than two target variables specified in `procrustes`.

Remarks and examples

The examples in [MV] *procrustes* demonstrated a Procrustes transformation of a historical map, produced by John Speed in 1610, to a modern map. Here we demonstrate the use of *procrustes* postestimation tools in assessing the accuracy of Speed's map. Example 1 of [MV] *procrustes* performed the following analysis:

```
. use https://www.stata-press.com/data/r19/speed_survey
(Data on Speed's Worcestershire map (1610))

. procrustes (survey_x survey_y) (speed_x speed_y)
(output omitted)
```

See example 1 of [MV] *procrustes*. The following examples are based on this *procrustes* analysis.

► Example 1: Predictions

Did John Speed get the coordinates of the towns right—up to the location, scale, and orientation of his map relative to the modern map? In [example 1](#) of [\[MV\] procrustes](#), we demonstrated how the optimal transformation from the historical coordinates to the modern (true) coordinates can be estimated by `procrustes`.

It is possible to “predict” the configuration of 20 cities on Speed’s historical map, optimally transformed (rotated, dilated, and translated) to approximate the true configuration. `predict` with the `fitted` option expects the same number of variables as the number of target (dependent) variables (`survey_x` and `survey_y`).

```
. predict fitted_x fitted_y
(option fitted assumed; fitted values)
```

We omitted the `fitted` option because it is the default.

It is often useful to also compute the (squared) distance between the true location and the transformed location of the historical map. This can be seen as a quality measure—the larger the value, the more Speed erred in the location of the respective town.

```
. predict q, q
```

We now list the target data (`survey_x` and `survey_y`, the values from the modern map), the fitted values (`fitted_x` and `fitted_y`, produced by `predict`), and the squared distance between them (`q`, produced by `predict` with the `q` option).

```
. list name survey_x survey_y fitted_x fitted_y q, sep(0) noobs
```

| name | survey_x | survey_y | fitted_x | fitted_y | q |
|------|----------|----------|----------|----------|----------|
| Alve | 1027 | 725 | 1037.117 | 702.9464 | 588.7149 |
| Arro | 1083 | 565 | 1071.682 | 562.6791 | 133.4802 |
| Astl | 787 | 677 | 783.0652 | 674.5216 | 21.62482 |
| Beck | 976 | 358 | 978.8665 | 366.3761 | 78.37637 |
| Beng | 1045 | 435 | 1055.245 | 431.6015 | 116.51 |
| Crad | 736 | 471 | 725.8594 | 476.5895 | 134.075 |
| Droi | 893 | 633 | 890.5839 | 633.6066 | 6.205747 |
| Ecki | 922 | 414 | 929.4932 | 411.1757 | 64.12465 |
| Eves | 1037 | 437 | 1036.887 | 449.2707 | 150.5827 |
| Hall | 828 | 579 | 825.1494 | 575.9836 | 17.22464 |
| Hanb | 944 | 637 | 954.6189 | 643.6107 | 156.4629 |
| Inkb | 1016 | 573 | 1004.869 | 577.1111 | 140.7917 |
| Kemp | 848 | 490 | 845.7215 | 490.8959 | 5.994327 |
| Kidd | 826 | 762 | 836.8665 | 760.5699 | 120.1264 |
| Mart | 756 | 598 | 745.2623 | 597.5585 | 115.4937 |
| Stud | 1074 | 632 | 1072.622 | 634.3164 | 7.264294 |
| Tewk | 891 | 324 | 898.4571 | 318.632 | 84.42448 |
| UpSn | 943 | 544 | 939.3932 | 545.8247 | 16.33858 |
| Upto | 852 | 403 | 853.449 | 400.9419 | 6.335171 |
| Worc | 850 | 545 | 848.7917 | 547.7881 | 9.233305 |

We see that Speed especially erred in the location of Alvechurch—it is off by no less than $\sqrt{588} = 24$ miles, whereas the average error is about 8 miles. In a serious analysis of this dataset, we would check the data on Alvechurch, and, if we found it to be in order, consider whether we should actually drop Alvechurch from the analysis. In this illustration, we ignore this potential problem.

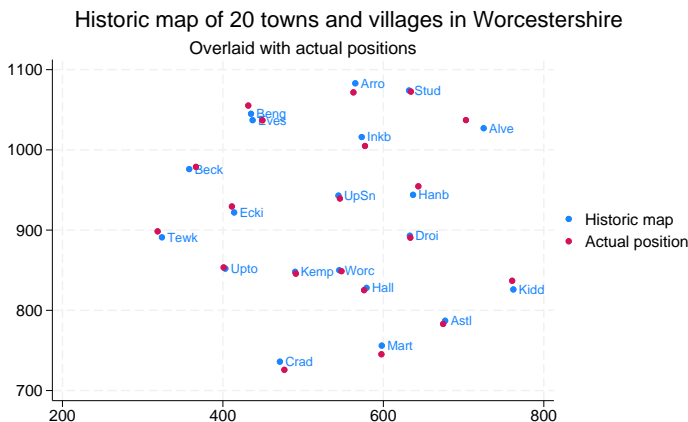
► Example 2: Procrustes overlay graph

Although the numerical information convinces us that Speed's map is generally accurate, a plot will convey this message more convincingly. `procoverlay` produces a plot that contains the target (survey) coordinates and the Procrustes-transformed historical coordinates. We could just type

```
. procoverlay
```

However, we decide to set several options to produce a presentation-quality graph. The suboption `mlabel()` of `target()` (or of `source()`) adds labels, identifying the towns. Because the target and source points are so close, there can be no confusing how they are matched. Displaying the labels twice in the plot is not helpful for this dataset. Therefore, we choose to label the target points, but not the source points using the `nolabel` suboption of `source()`. We preserve the equivalence of the x and y scale while using as much of the graphing region as possible with the `autoaspect` option. The `span` suboption of `title()` allows the long title to extend beyond the graph region if needed. We override the default legend by using the `legend()` option.

```
. procoverlay, target(mlabel(name)) source(nolabel) autoaspect
> title(Historic map of 20 towns and villages in Worcestershire, span)
> subtitle(Overlaid with actual positions)
> legend(label(1 Historic map) label(2 Actual position))
```



► Example 3: estat

`estat` offers three specific facilities after `procrustes`. These can all be seen as convenience tools that accomplish simple analyses, ensuring that the same variables and the same observations are used as in the Procrustes analysis.

The variables involved in the Procrustes analysis can be summarized over the estimation sample, for instance, to gauge differences in scales and location of the target and source variables.

```
. estat summarize
```

| Estimation sample procrustes | | | Number of obs = | | 20 |
|------------------------------|--------|-----------|-----------------|------|----|
| Variable | Mean | Std. dev. | Min | Max | |
| target | | | | | |
| survey_x | 916.7 | 106.6993 | 736 | 1083 | |
| survey_y | 540.1 | 121.1262 | 324 | 762 | |
| source | | | | | |
| speed_x | 153.95 | 46.76084 | 78 | 220 | |
| speed_y | 133.9 | 49.90401 | 40 | 220 | |

From the summarization, the two maps have different origins and scale.

As pointed out in [MV] [procrustes](#), orthogonal and oblique Procrustes analyses can be thought of as special cases of multivariate regression (see [MV] [mvreg](#)), subject to nonlinear restrictions on the coefficient matrix. Comparing the Procrustes statistics and the transformations for each of the three classes of transformations is helpful in selecting a transformation. The `compare` subcommand of `estat` provides summary information for the optimal transformations in each of the three classes.

```
. estat compare
```

Summary statistics for three transformations

| | Procrustes | df_m | df_r | rmse |
|--------------|------------|------|------|----------|
| orthogonal | 0.0040 | 4 | 36 | 7.403797 |
| oblique | 0.0040 | 5 | 35 | 7.498294 |
| unrestricted | 0.0037 | 6 | 34 | 7.343334 |

(F tests comparing the models suppressed)

The Procrustes statistic is ensured to decrease (not increase) from orthogonal to oblique to unrestricted because the associated classes of transformations are getting less restrictive. The model degrees of freedom (`df_m`) of the three transformation classes are the dimension of the classes, that is, the number of “free parameters”. For instance, with orthogonal transformations between two source and two target variables, there is 1 degree of freedom for the rotation (representing the rotation angle), 2 degrees of freedom for the translation, and 1 degree of freedom for dilation (uniform scaling), that is, four in total. The residual degrees of freedom (`df_r`) are the number of observations (number of target variables times the number of observations) minus the model degrees of freedom. The root mean squared error RMSE, defined as

$$\text{RMSE} = \sqrt{\frac{\text{RSS}}{\text{df}_r}}$$

does not, unlike the Procrustes statistic, surely become smaller with the less restrictive models. In this example, in fact, the RMSE of the orthogonal transformation is smaller than that of the oblique transformation. This indicates that the additional degree of freedom allowing for skew rotations does not produce a closer fit. In this example, we see little reason to relax orthogonal transformations; very little is gained in terms of the Procrustes statistic (an illness-of-fit measure) or the RMSE. In this interpretation, we used our intuition to guide us whether a difference in fit is substantively and statistically meaningful—formal significance tests are not provided.

Finally, the unrestricted transformation can be estimated with `procrustes ..., transform(unrestricted)`. This analysis is related to a multivariate regression with the target variables as the dependent variables and the source variables as the independent variables. Although the unrestricted Procrustes analysis assumes spherical (uncorrelated homoskedastic) residuals, this restrictive assumption is not made in multivariate regression as estimated by the `mvreg` command. The comparable multivariate regression over the same estimation sample can be viewed simply by typing

```
. estat mvreg
```

Multivariate regression, similar to "procrustes ..., transform(unrestricted)"

| Equation | Obs | Parms | RMSE | "R-sq" | F | P>F |
|----------|-----|-------|----------|--------|----------|--------|
| survey_x | 20 | 3 | 7.696981 | 0.9953 | 1817.102 | 0.0000 |
| survey_y | 20 | 3 | 6.971772 | 0.9970 | 2859.068 | 0.0000 |

| | Coefficient | Std. err. | t | P> t | [95% conf. interval] | |
|----------|-------------|-----------|--------|-------|----------------------|-----------|
| survey_x | | | | | | |
| speed_x | 2.27584 | .0379369 | 59.99 | 0.000 | 2.1958 | 2.35588 |
| speed_y | .4147244 | .0355475 | 11.67 | 0.000 | .3397257 | .489723 |
| _cons | 510.8028 | 8.065519 | 63.33 | 0.000 | 493.7861 | 527.8196 |
| survey_y | | | | | | |
| speed_x | -.4129564 | .0343625 | -12.02 | 0.000 | -.485455 | -.3404579 |
| speed_y | 2.355725 | .0321982 | 73.16 | 0.000 | 2.287793 | 2.423658 |
| _cons | 288.243 | 7.305587 | 39.46 | 0.000 | 272.8296 | 303.6564 |

This analysis is seen as postestimation after a Procrustes analysis, so it does not change the “last estimation results”. We may still replay `procrustes` and use other `procrustes` postestimation commands.



Stored results

`estat compare` after `procrustes` stores the following in `r()`:

Matrices

| | |
|-----------------------|---|
| <code>r(cstat)</code> | Procrustes statistics, degrees of freedom, and RMSEs |
| <code>r(fstat)</code> | <i>F</i> statistics, degrees of freedom, and <i>p</i> -values |

`estat mvreg` does not return results.

`estat summarize` after `procrustes` stores the following in `r()`:

Matrices

| | |
|-----------------------|--|
| <code>r(stats)</code> | means, standard deviations, minimums, and maximums |
|-----------------------|--|

Methods and formulas

The predicted values for the j th variable are defined as

$$\hat{y}_j = \hat{c}_j + \hat{\rho} \mathbf{X} \hat{\mathbf{A}}[:, j]$$

The residual for y_j is simply $y_j - \hat{y}_j$. The “rowwise” quality q of the approximation is defined as the residual sum of squares:

$$q = \sum_j (y_j - \hat{y}_j)^2$$

The entries of the summary table produced by `estat compare` are described in [Methods and formulas](#) of [\[MV\] procrustes](#). The F tests produced by `estat compare` are similar to standard nested model tests in linear models.

References

See [References](#) in [\[MV\] procrustes](#).

Also see

[\[MV\] procrustes](#) — Procrustes transformation

[\[MV\] mvreg](#) — Multivariate regression

[\[U\] 20 Estimation and postestimation commands](#)

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.

For suggested citations, see the FAQ on [citing Stata documentation](#).

