# Title

stata.com

**estimates table —** Compare estimation results


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


## Description

`estimates table` displays a table of coefficients and statistics for one or more sets of estimation results.

## Quick start

Display a table of coefficients for stored estimates `m1` and `m2`

```
estimates table m1 m2
```

Add standard errors

```
estimates table m1 m2, se
```

Same as above, but display coefficients and standard errors to 2 decimal places

```
estimates table m1 m2, b(%7.2f) se(%7.2f)
```

Add $p$-values

```
estimates table m1 m2, b(%7.2f) se(%7.2f) p(%4.3f)
```

Add sample size and adjusted $R^2$

```
estimates table m1 m2, b(%7.2f) se(%7.2f) p(%4.3f) stats(N r2_a)
```

Replace variable names with labels

```
estimates table m1 m2, b(%7.2f) se(%7.2f) p(%4.3f) stats(N r2_a) ///
     varlabel
```

Table of coefficients with stars to denote significance

```
estimates table m1 m2, b(%7.2f) star
```

Display coefficients in exponentiated form

```
estimates table m3 m4, b(%7.2f) se(%7.2f) eform
```

Display only a subset of variables and reorder variables in table

```
estimates table m1 m2, keep(v2 v1 v3 _cons)
```

## Menu

Statistics > Postestimation

# Syntax

```
estimates table [namelist] [, options]
```

where *namelist* is a name, a list of names, `_all`, or `*`. A name may be `.`, meaning the current (active) estimates. `_all` and `*` mean the same thing.

| *options* | Description |
|---|---|
| **Main** | |
| `stats(`*scalarlist*`)` | report *scalarlist* in table |
| `star[(`*#1 #2 #3*`)]` | use stars to denote significance levels |
| **Options** | |
| `keep(`*coeflist*`)` | report coefficients in order specified |
| `drop(`*coeflist*`)` | omit specified coefficients from table |
| `equations(`*matchlist*`)` | match equations of models as specified |
| **Numerical formats** | |
| `b[(`%*fmt*`)]` | how to format coefficients, which are always reported |
| `se[(`%*fmt*`)]` | report standard errors and use optional format |
| `t[(`%*fmt*`)]` | report $t$ or $z$ and use optional format |
| `p[(`%*fmt*`)]` | report $p$-values and use optional format |
| `stfmt(`%*fmt*`)` | how to format scalar statistics |
| **General format** | |
| `varwidth(`*#*`)` | use *#* characters to display variable names and statistics |
| `modelwidth(`*#*`)` | use *#* characters to display model names |
| `eform` | display coefficients in exponentiated form |
| `varlabel` | display variable labels rather than variable names |
| `newpanel` | display statistics in separate table from coefficients |
| `style(oneline)` | put vertical line after variable names; the default |
| `style(columns)` | put vertical line separating every column |
| `style(noline)` | suppress all vertical lines |
| `coded` | display compact table |
| **Reporting** | |
| *display_options* | control row spacing, line width, and display of omitted variables and base and empty cells |
| `title(`*string*`)` | title for table |

`title()` does not appear in the dialog box.

where

- A *scalarlist* is a list of any of or all the names of scalars stored in `e()`, plus `aic`, `bic`, and `rank`.
- *#1 #2 #3* are three numbers such as `.05 .01 .001`.
- A *coeflist* is a list of coefficient names, each name of which may be simple (for example, `price`), an equation name followed by a colon (for example, `mean:`), or a full name (for example, `mean:price`). Names are separated by blanks.
- A *matchlist* specifies how equations from different estimation results are to be matched. If you need to specify a *matchlist*, the solution is usually `1`, as in `equations(1)`. The full syntax is

$$\textit{matchlist} \;:=\; \textit{term} \;\big[\,\texttt{,}\;\textit{term}\;\ldots\,\big]$$

$$\textit{term} \;:=\; \big[\,\textit{eqname}\;\texttt{=}\,\big]\;\texttt{\#:\#}\ldots\texttt{:\#}$$
$$\big[\,\textit{eqname}\;\texttt{=}\,\big]\;\texttt{\#}$$

  See `equations()` under *Options* below.

# Options

Main

`stats(`*scalarlist*`)` specifies one or more scalar statistics to be displayed in the table. *scalarlist* may contain

| | |
|---|---|
| `aic` | Akaike's information criterion |
| `bic` | Schwarz's Bayesian information criterion |
| `rank` | rank of `e(V)` (# of free parameters in model) |

along with the names of any scalars stored in `e()`. The specified statistics do not have to be available for all estimation results being displayed.

For example, `stats(N ll chi2 aic)` specifies that `e(N)`, `e(ll)`, `e(chi2)`, and AIC be included. In Stata, `e(N)` records the number of observations; `e(ll)`, the log likelihood; and `e(chi2)`, the chi-squared test that all coefficients in the first equation of the model are equal to zero.

`star` and `star(`*#1 #2 #3*`)` specify that stars (asterisks) are to be used to mark significance. The second syntax specifies the significance levels for one, two, and three stars. If you specify simply `star`, that is equivalent to specifying `star(.05 .01 .001)`, which means one star (*) if $p < 0.05$, two stars (**) if $p < 0.01$, and three stars (***) if $p < 0.001$.

The `star` and `star()` options may not be combined with `se`, `t`, or `p` option.

Options

`keep(`*coeflist*`)` and `drop(`*coeflist*`)` are alternatives; they specify coefficients to be included or omitted from the table. The default is to display all coefficients.

If `keep()` is specified, it specifies not only the coefficients to be included but also the order in which they appear.

A *coeflist* is a list of coefficient names, each name of which may be simple (for example, `price`), an equation name followed by a colon (for example, `mean:`), or a full name (for example, `mean:price`). Names are separated from each other by blanks.

When full names are not specified, all coefficients that match the partial specification are included. For instance, `drop(_cons)` would omit `_cons` for all equations.

`equations(`*matchlist*`)` specifies how the equations of the models in *namelist* are to be matched. The default is to match equations by name. Matching by name usually works well when all results were fit by the same estimation command. When you are comparing results from different estimation commands, however, specifying `equations()` may be necessary.

The most common usage is `equations(1)`, which indicates that all first equations are to be matched into one equation named `#1`.

*matchlist* has the syntax

*term* [ , *term* ... ]

where *term* is

[ *eqname* = ] #:#...:# (syntax 1)

[ *eqname* = ] # (syntax 2)

In syntax 1, each # is a number or a period (.). If a number, it specifies the position of the equation in the corresponding model; `1:3:1` would indicate that equation 1 in the first model matches equation 3 in the second, which matches equation 1 in the third. A period indicates that there is no corresponding equation in the model; `1:.:1` indicates that equation 1 in the first matches equation 1 in the third.

In syntax 2, you specify just one number, say, `1` or `2`, and that is shorthand for `1:1...:1` or `2:2...:2`, meaning that equation 1 matches across all models specified or that equation 2 matches across all models specified.

Now that you can specify a *term*, you can put that together into a *matchlist* by separating one term from the other by commas. In what follows, we will assume that three names were specified,

```
. estimates table alpha beta gamma, ...
```

`equations(1)` is equivalent to `equations(1:1:1)`; we would be saying that the first equations match across the board.

`equations(1:.:1)` would specify that equation 1 matches in models `alpha` and `gamma` but that there is nothing corresponding in model `beta`.

`equations(1,2)` is equivalent to `equations(1:1:1, 2:2:2)`. We would be saying that the first equations match across the board and so do the second equations.

`equations(1, 2:.:2)` would specify that the first equations match across the board, that the second equations match for models `alpha` and `gamma`, and that there is nothing equivalent to equation 2 in model `beta`.

If `equations()` is specified, equations not matched by position are matched by name.

Numerical formats

`b(%`*fmt*`)` specifies how the coefficients are to be displayed. You might specify `b(%9.2f)` to make decimal points line up. There is also a `b` option, which specifies that coefficients are to be displayed, but that is just included for consistency with the `se`, `t`, and `p` options. Coefficients are always displayed.

`se`, `t`, and `p` specify that standard errors, $t$ or $z$ statistics, and significance levels are to be displayed. The default is not to display them. `se(%`*fmt*`)`, `t(%`*fmt*`)`, and `p(%`*fmt*`)` specify that each is to be displayed and specifies the display format to be used to format them.

`stfmt(%fmt)` specifies the format for displaying the scalar statistics included by the `stats()` option.

General format

`varwidth(#)` specifies the number of character positions used to display the names of the variables and statistics. The default is 12.

`modelwidth(#)` specifies the number of character positions used to display the names of the models. The default is 12.

`eform` displays coefficients in exponentiated form. For each coefficient, exp($\beta$) rather than $\beta$ is displayed, and standard errors are transformed appropriately. Display of the intercept, if any, is suppressed.

`varlabel` specifies that variable labels be displayed instead of variable names.

`newpanel` specifies that the statistics be displayed in a table separated by a blank line from the table with coefficients rather than in the style of another equation in the table of coefficients.

`style(`*stylespec*`)` specifies the style of the coefficient table.

`style(oneline)` specifies that a vertical line be displayed after the variables but not between the models. This is the default.

`style(columns)` specifies that vertical lines be displayed after each column.

`style(noline)` specifies that no vertical lines be displayed.

`coded` specifies that a compact table be displayed. This format is especially useful for comparing variables that are included in a large collection of models.

Reporting

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

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

`title(`*string*`)` specifies the title to appear above the table.

## Remarks and examples

stata.com

If you type `estimates table` without arguments, a table of the most recent estimation results will be shown:

```
. use http://www.stata-press.com/data/r15/auto
(1978 Automobile Data)
. regress mpg weight displ
  (output omitted)
. estimates table
```

| Variable | active |
|---|---|
| weight | -.00656711 |
| displacement | .00528078 |
| _cons | 40.084522 |

The real use of `estimates table`, however, is for comparing estimation results, and that requires using it after `estimates store`:

```
. regress mpg weight displ
  (output omitted)
. estimates store base
. regress mpg weight displ foreign
  (output omitted)
. estimates store alt
. qreg mpg weight displ foreign
  (output omitted)
. estimates store qreg
. estimates table base alt qreg, stats(r2)
```

| Variable | base | alt | qreg |
|---|---|---|---|
| weight | -.00656711 | -.00677449 | -.00595056 |
| displacement | .00528078 | .00192865 | .00018552 |
| foreign | | -1.6006312 | -2.1326005 |
| _cons | 40.084522 | 41.847949 | 39.213348 |
| r2 | .6529307 | .66287957 | |

## Stored results

`estimates table` stores the following in `r()`:

Macros
- `r(names)` — names of results used

Matrices
- `r(coef)` — matrix $M$: $n \times 2{*}m$
  $M[i, 2j-1]$ = $i$th parameter estimate for model $j$;
  $M[i, 2j]$ = variance of $M[i, 2j-1]$; $i$=1,...,$n$; $j$=1,...,$m$
- `r(stats)` — matrix $S$: $k \times m$ (if option `stats()` specified)
  $S[i, j]$ = $i$th statistic for model $j$; $i$=1,...,$k$; $j$=1,...,$m$

## References

Gallup, J. L. 2012. A new system for formatting estimation tables. *Stata Journal* 12: 3–28.

Weiss, M. 2010. Stata tip 90: Displaying partial results. *Stata Journal* 10: 500–502.

## Also see

[R] **estimates** — Save and manipulate estimation results