

## newey — Regression with Newey–West standard errors

|             |                      |                |                      |
|-------------|----------------------|----------------|----------------------|
| Description | Quick start          | Menu           | Syntax               |
| Options     | Remarks and examples | Stored results | Methods and formulas |
| References  | Also see             |                |                      |

## Description

`newey` produces Newey–West standard errors for coefficients estimated by OLS regression. The error structure is assumed to be heteroskedastic and possibly autocorrelated up to some lag.

## Quick start

OLS regression of `y` on `x1` and `x2` with Newey–West standard errors robust to heteroskedasticity and first-order autocorrelation using `tsset` data

```
newey y x1 x2, lag(1)
```

With heteroskedasticity-robust standard errors

```
newey y x1 x2, lag(0)
```

## Menu

Statistics > Time series > Regression with Newey-West std. errors

## Syntax

```
newey depvar [indepvars] [if] [in] [weight], lag(#) [options]
```

| <i>options</i>               | Description  |
|------------------------------|--|
| <hr/>                        |  |
| Model                        |  |
| * <code>lag(#)</code>        | set maximum lag order of autocorrelation   |
| <code>noconstant</code>      | suppress constant term   |
| Reporting                    |  |
| <code>level(#)</code>        | set confidence level; default is <code>level(95)</code>  |
| <code>display_options</code> | control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling |
| <code>coeflegend</code>      | display legend instead of statistics   |

\*`lag(#)` is required.

You must `tsset` your data before using `newey`; see [TS] `tsset`.

`indepvars` may contain factor variables; see [U] 11.4.3 Factor variables.

`depvar` and `indepvars` may contain time-series operators; see [U] 11.4.4 Time-series varlists.

`by`, `rolling`, and `statsby` are allowed; see [U] 11.1.10 Prefix commands.

`aweight`s are allowed; see [U] 11.1.6 `weight`.

`coeflegend` does not appear in the dialog box.

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

## Options

### Model

`lag(#)` specifies the maximum lag to be considered in the autocorrelation structure. If you specify `lag(0)`, the output is the same as `regress, vce(robust)`. `lag()` is required.

`noconstant`; see [R] [estimation options](#).

### Reporting

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

`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](#).

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

`coeflegend`; see [R] [estimation options](#).

## Remarks and examples

[stata.com](http://www.stata.com)

The Huber/White/sandwich robust variance estimator (see [White \[1980\]](#)) produces consistent standard errors for OLS regression coefficient estimates in the presence of heteroskedasticity. The Newey–West (1987) variance estimator is an extension that produces consistent estimates when there is autocorrelation in addition to possible heteroskedasticity.

The Newey–West variance estimator handles autocorrelation up to and including a lag of  $m$ , where  $m$  is specified by stipulating the `lag()` option. Thus, it assumes that any autocorrelation at lags greater than  $m$  can be ignored.

If `lag(0)` is specified, the variance estimates produced by `newey` are simply the Huber/White/sandwich robust variances estimates calculated by `regress, vce(robust)`; see [R] [regress](#).

### ► Example 1

`newey, lag(0)` is equivalent to `regress, vce(robust)`:

```
. use http://www.stata-press.com/data/r15/auto
(1978 Automobile Data)
```

```
. regress price weight displ, vce(robust)
```

|                   |  |               |   |        |
|-------------------|--|---------------|---|--------|
| Linear regression |  | Number of obs | = | 74     |
|                   |  | F(2, 71)      | = | 14.44  |
|                   |  | Prob > F      | = | 0.0000 |
|                   |  | R-squared     | = | 0.2909 |
|                   |  | Root MSE      | = | 2518.4 |

|  | price        | Coef.    | Robust<br>Std. Err. | t    | P> t  | [95% Conf. Interval] |
|--|--------------|----------|---------------------|------|-------|----------------------|
|  | weight       | 1.823366 | .7808755            | 2.34 | 0.022 | .2663445 3.380387    |
|  | displacement | 2.087054 | 7.436967            | 0.28 | 0.780 | -12.74184 16.91595   |
|  | _cons        | 247.907  | 1129.602            | 0.22 | 0.827 | -2004.455 2500.269   |

```
. generate t = _n
```

```

. tsset t
   time variable: t, 1 to 74
             delta: 1 unit

. newey price weight displ, lag(0)
Regression with Newey–West standard errors   Number of obs   =       74
maximum lag: 0                               F( 2,          71) =      14.44
                                              Prob > F         =      0.0000

```

| price        | Newey–West |           | t    | P> t  | [95% Conf. Interval] |          |
|--------------|------------|-----------|------|-------|----------------------|----------|
|              | Coef.      | Std. Err. |      |       |                      |          |
| weight       | 1.823366   | .7808755  | 2.34 | 0.022 | .2663445             | 3.380387 |
| displacement | 2.087054   | 7.436967  | 0.28 | 0.780 | -12.74184            | 16.91595 |
| _cons        | 247.907    | 1129.602  | 0.22 | 0.827 | -2004.455            | 2500.269 |

Because `newey` requires the dataset to be `tsset`, we generated a dummy time variable `t`, which in this example played no role in the estimation.

◀

## ▶ Example 2

Say that we have time-series measurements on variables `usr` and `idle` and now wish to fit an OLS model but obtain Newey–West standard errors allowing for a lag of up to 3:

```

. use http://www.stata-press.com/data/r15/idle2, clear
. tsset time
   time variable: time, 1 to 30
             delta: 1 unit

. newey usr idle, lag(3)
Regression with Newey–West standard errors   Number of obs   =       30
maximum lag: 3                               F( 1,          28) =      10.90
                                              Prob > F         =      0.0026

```

| usr   | Newey–West |           | t     | P> t  | [95% Conf. Interval] |          |
|-------|------------|-----------|-------|-------|----------------------|----------|
|       | Coef.      | Std. Err. |       |       |                      |          |
| idle  | -.2281501  | .0690927  | -3.30 | 0.003 | -.3696801            | -.08662  |
| _cons | 23.13483   | 6.327031  | 3.66  | 0.001 | 10.17449             | 36.09516 |

◀

## Stored results

`newey` stores the following in `e()`:

### Scalars

|                      |                             |
|----------------------|-----------------------------|
| <code>e(N)</code>    | number of observations      |
| <code>e(df_m)</code> | model degrees of freedom    |
| <code>e(df_r)</code> | residual degrees of freedom |
| <code>e(F)</code>    | $F$ statistic               |
| <code>e(lag)</code>  | maximum lag                 |
| <code>e(rank)</code> | rank of $e(V)$              |

### Macros

|                            |  |
|----------------------------|--|
| <code>e(cmd)</code>        | <code>newey</code>   |
| <code>e(cmdline)</code>    | command as typed   |
| <code>e(depvar)</code>     | name of dependent variable                                     |
| <code>e(wtype)</code>      | weight type  |
| <code>e(wexp)</code>       | weight expression  |
| <code>e(title)</code>      | title in estimation output                                     |
| <code>e(vctype)</code>     | title used to label Std. Err.                                  |
| <code>e(properties)</code> | b V  |
| <code>e(estat_cmd)</code>  | program used to implement <code>estat</code>                   |
| <code>e(predict)</code>    | program used to implement <code>predict</code>                 |
| <code>e(asbalanced)</code> | factor variables <code>fvset</code> as <code>asbalanced</code> |
| <code>e(asobserved)</code> | factor variables <code>fvset</code> as <code>asobserved</code> |

### Matrices

|                     |  |
|---------------------|--|
| <code>e(b)</code>   | coefficient vector                           |
| <code>e(Cns)</code> | constraints matrix                           |
| <code>e(V)</code>   | variance–covariance matrix of the estimators |

### Functions

|                        |                         |
|------------------------|-------------------------|
| <code>e(sample)</code> | marks estimation sample |
|------------------------|-------------------------|

## Methods and formulas

`newey` calculates the estimates

$$\hat{\beta}_{OLS} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

$$\widehat{\text{Var}}(\hat{\beta}_{OLS}) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\hat{\Omega}\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}$$

That is, the coefficient estimates are simply those of OLS linear regression.

For `lag(0)` (no autocorrelation), the variance estimates are calculated using the White formulation:

$$\mathbf{X}'\hat{\Omega}\mathbf{X} = \mathbf{X}'\hat{\Omega}_0\mathbf{X} = \frac{n}{n-k} \sum_i \hat{e}_i^2 \mathbf{x}_i' \mathbf{x}_i$$

Here  $\hat{e}_i = y_i - \mathbf{x}_i' \hat{\beta}_{OLS}$ , where  $\mathbf{x}_i$  is the  $i$ th row of the  $\mathbf{X}$  matrix,  $n$  is the number of observations, and  $k$  is the number of predictors in the model, including the constant if there is one. The above formula is the same as that used by `regress`, `vce(robust)` with the regression-like formula (the default) for the multiplier  $q_c$ ; see [Methods and formulas](#) of [\[R\] regress](#).

For `lag(m)`,  $m > 0$ , the variance estimates are calculated using the Newey–West (1987) formulation

$$\mathbf{X}'\hat{\Omega}\mathbf{X} = \mathbf{X}'\hat{\Omega}_0\mathbf{X} + \frac{n}{n-k} \sum_{l=1}^m \left(1 - \frac{l}{m+1}\right) \sum_{t=l+1}^n \hat{e}_t \hat{e}_{t-l} (\mathbf{x}_t' \mathbf{x}_{t-l} + \mathbf{x}_{t-l}' \mathbf{x}_t)$$

where  $\mathbf{x}_t$  is the row of the  $\mathbf{X}$  matrix observed at time  $t$ .

Whitney K. Newey (1954– ) earned degrees in economics at Brigham Young University and MIT. After a period at Princeton, he returned to MIT as a professor in 1990. His interests in theoretical and applied econometrics include bootstrapping, nonparametric estimation of models, semiparametric models, and choosing the number of instrumental variables.

Kenneth D. West (1953– ) earned a bachelor's degree in economics and mathematics at Wesleyan University and then a PhD in economics at MIT. After a period at Princeton, he joined the University of Wisconsin in 1988. His interests include empirical macroeconomics and time-series econometrics.

## References

- Hardin, J. W. 1997. [sg72: Newey–West standard errors for probit, logit, and poisson models](#). *Stata Technical Bulletin* 39: 32–35. Reprinted in *Stata Technical Bulletin Reprints*, vol. 7, pp. 182–186. College Station, TX: Stata Press.
- Newey, W. K., and K. D. West. 1987. A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica* 55: 703–708.
- Wang, Q., and N. Wu. 2012. [Long-run covariance and its applications in cointegration regression](#). *Stata Journal* 12: 515–542.
- White, H. L., Jr. 1980. A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity. *Econometrica* 48: 817–838.

## Also see

- [TS] [newey postestimation](#) — Postestimation tools for newey
- [TS] [arima](#) — ARIMA, ARMAX, and other dynamic regression models
- [TS] [forecast](#) — Econometric model forecasting
- [TS] [tsset](#) — Declare data to be time-series data
- [R] [regress](#) — Linear regression
- [U] [20 Estimation and postestimation commands](#)