

**dsregress** — Double-selection lasso linear regression

<a href="#">Description</a>	<a href="#">Quick start</a>	<a href="#">Menu</a>	<a href="#">Syntax</a>
<a href="#">Options</a>	<a href="#">Remarks and examples</a>	<a href="#">Stored results</a>	<a href="#">Methods and formulas</a>
<a href="#">Reference</a>	<a href="#">Also see</a>		

## Description

`dsregress` fits a lasso linear regression model and reports coefficients along with standard errors, test statistics, and confidence intervals for specified covariates of interest. The double-selection method is used to estimate effects for these variables and to select from potential control variables to be included in the model.

## Quick start

Estimate a coefficient for `d1` in a linear regression of `y` on `d1`, and include `x1` to `x100` as potential control variables to be selected by lassos

```
dsregress y d1, controls(x1-x100)
```

Same as above, and estimate coefficients for the levels of categorical `d2`

```
dsregress y d1 i.d2, controls(x1-x100)
```

Use cross-validation (CV) instead of a plugin iterative formula to select the optimal  $\lambda^*$  in each lasso

```
dsregress y d1 i.d2, controls(x1-x100) selection(cv)
```

Same as above, and set a random-number seed for reproducibility

```
dsregress y d1 i.d2, controls(x1-x100) selection(cv) rseed(28)
```

Specify CV for the lasso for `y` only, with the stopping rule criterion turned off

```
dsregress y d1 i.d2, controls(x1-x100) lasso(y, selection(cv), stop(0))
```

Same as above, but apply the option to the lassos for `y`, `d1`, and `i.d2`

```
dsregress y d1 i.d2, controls(x1-x100) lasso(*, selection(cv), stop(0))
```

Compute lassos beyond the CV minimum to get full coefficient paths, knots, etc.

```
dsregress y d1 i.d2, controls(x1-x100) ///
    lasso(*, selection(cv, alllambdas))
```

## Menu

Statistics > Lasso > Lasso inferential models > Continuous outcomes > Double-selection model

## Syntax

```
dsregress depvar varsofinterest [if] [in],
         controls([(alwaysvars)] othervars) [options]
```

*varsofinterest* are variables for which coefficients and their standard errors are estimated.

<i>options</i>	Description
Model	
* <u>controls</u> ([ <i>(alwaysvars)</i> ] <i>othervars</i> )	<i>alwaysvars</i> and <i>othervars</i> make up the set of control variables; <i>alwaysvars</i> are always included; lassos choose whether to include or exclude <i>othervars</i>
<u>selection</u> (plugin)	use a plugin iterative formula to select an optimal value of the lasso penalty parameter $\lambda^*$ for each lasso; the default
<u>selection</u> (cv)	use CV to select an optimal value of the lasso penalty parameter $\lambda^*$ for each lasso
<u>selection</u> (adaptive)	use adaptive lasso to select an optimal value of the lasso penalty parameter $\lambda^*$ for each lasso
<u>selection</u> (bic)	use BIC to select an optimal value of the lasso penalty parameter $\lambda^*$ for each lasso
<u>sqrlasso</u>	use square-root lassos
<u>missingok</u>	after fitting lassos, ignore missing values in any <i>othervars</i> not selected, and include these observations in the final model
SE/Robust	
<u>vce</u> ( <i>vcetype</i> )	<i>vcetype</i> may be <u>robust</u> (the default), <u>cluster</u> <i>clustvar</i> , <u>ols</u> , <u>hc2</u> , or <u>hc3</u>
Reporting	
<u>level</u> (#)	set confidence level; default is <u>level</u> (95)
<u>display_options</u>	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Optimization	
[ <u>no</u> ] <u>log</u>	display or suppress an iteration log
<u>verbose</u>	display a verbose iteration log
<u>rseed</u> (#)	set random-number seed
Advanced	
<u>lasso</u> ( <i>varlist</i> , <i>lasso_options</i> )	specify options for the lassos for variables in <i>varlist</i> ; may be repeated
<u>sqrlasso</u> ( <i>varlist</i> , <i>lasso_options</i> )	specify options for square-root lassos for variables in <i>varlist</i> ; may be repeated
<u>reestimate</u>	refit the model after using <u>lassoselect</u> to select a different $\lambda^*$
<u>noheader</u>	do not display the header on the coefficient table
<u>coeflegend</u>	display legend instead of statistics

\*`controls()` is required.

*varsofinterest*, *alwaysvars*, and *othervars* may contain factor variables. Base levels of factor variables cannot be set for *alwaysvars* and *othervars*. See [U] 11.4.3 Factor variables.

`collect` is allowed; see [U] 11.1.10 Prefix commands.

`reestimate`, `noheader`, and `coeflegend` do not appear in the dialog box.

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

## Options

Model

`controls`( [*alwaysvars*] *othervars*) specifies the set of control variables, which control for omitted variables. Control variables are also known as confounding variables. `dsregress` fits lassos for *devar* and each of the *varsofinterest*. *alwaysvars* are variables that are always to be included in these lassos. *alwaysvars* are optional. *othervars* are variables that each lasso will choose to include or exclude. That is, each lasso will select a subset of *othervars*. The selected subset of *othervars* may differ across lassos. `controls()` is required.

`selection`(*plugin* | *cv* | *adaptive* | *bic*) specifies the selection method for choosing an optimal value of the lasso penalty parameter  $\lambda^*$  for each lasso or square-root lasso estimation. Separate lassos are estimated for *devar* and each variable in *varsofinterest*. Specifying `selection()` changes the selection method for all of these lassos. You can specify different selection methods for different lassos using the option `lasso()` or `sqrtlasso()`. When `lasso()` or `sqrtlasso()` is used to specify a different selection method for the lassos of some variables, they override the global setting made using `selection()` for the specified variables.

`selection(plugin)` is the default. It selects  $\lambda^*$  based on a “plugin” iterative formula dependent on the data. See [LASSO] lasso options.

`selection(cv)` selects the  $\lambda^*$  that gives the minimum of the CV function. See [LASSO] lasso options.

`selection(adaptive)` selects  $\lambda^*$  using the adaptive lasso selection method. It cannot be specified when `sqrtlasso` is specified. See [LASSO] lasso options.

`selection(bic)` selects the  $\lambda^*$  that gives the minimum of the BIC function. See [LASSO] lasso options.

`sqrtlasso` specifies that square-root lassos be done rather than regular lassos. The option `lasso()` can be used with `sqrtlasso` to specify that regular lasso be done for some variables, overriding the global `sqrtlasso` setting for these variables. See [LASSO] lasso options.

`missingok` specifies that, after fitting lassos, the estimation sample be redefined based on only the nonmissing observations of variables in the final model. In all cases, any observation with missing values for *devar*, *varsofinterest*, *alwaysvars*, and *othervars* is omitted from the estimation sample for the lassos. By default, the same sample is used for calculation of the coefficients of the *varsofinterest* and their standard errors.

When `missingok` is specified, the initial estimation sample is the same as the default, but the sample used for the calculation of the coefficients of the *varsofinterest* can be larger. Now observations with missing values for any *othervars* not selected will be added to the estimation sample (provided there are no missing values for any of the variables in the final model).

`missingok` may produce more efficient estimates when data are missing completely at random. It does, however, have the consequence that estimation samples can change when selected variables differ in models fit using different selection methods. That is, when *othervars* contain missing

values, the estimation sample for a model fit using the default `selection(plugin)` will likely differ from the estimation sample for a model fit using, for example, `selection(cv)`.

## SE/Robust

`vce(vctype)` specifies the type of standard error reported. The default is `vce(robust)`, which is robust to some kinds of misspecification. Also available are `vce(cluster clustvar)`, which allows for intragroup correlation; `vce(o1s)`, which specifies the standard variance estimator for ordinary least-squares regression; and `vce(hc2)` and `vce(hc3)`, which specify alternative bias corrections for the robust variance calculation. See [R] [vce\\_option](#) and [Options](#) in [R] [regress](#).

When `vce(cluster clustvar)` is specified, all lassos also account for clustering. For each lasso, this affects how the log-likelihood function is computed and how the sample is split in cross-validation; see [Methods and formulas](#) in [LASSO] [lasso](#). Specifying `vce(cluster clustvar)` may lead to different selected controls and therefore to different point estimates for your variable of interest when compared to the estimation that ignores clustering.

## 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](#).

## Optimization

`[no]` `log` displays or suppresses a log showing the progress of the estimation. By default, one-line messages indicating when each lasso estimation begins are shown. Specify `verbose` to see a more detailed log.

`verbose` displays a verbose log showing the iterations of each lasso estimation. This option is useful when doing `selection(cv)` or `selection(adaptive)`. It allows you to monitor the progress of the lasso estimations for these selection methods, which can be time consuming when there are many *othervars* specified in `controls()`.

`rseed(#)` sets the random-number seed. This option can be used to reproduce results for `selection(cv)` and `selection(adaptive)`. The default selection method `selection(plugin)` does not use random numbers. `rseed(#)` is equivalent to typing `set seed #` prior to running `dsregress`. See [R] [set seed](#).

## Advanced

`lasso(varlist, lasso_options)` lets you set different options for different lassos, or advanced options for all lassos. You specify a *varlist* followed by the options you want to apply to the lassos for these variables. *varlist* consists of one or more variables from *depvar* or *varsofinterest*. `_all` or `*` may be used to specify *depvar* and all *varsofinterest*. This option is repeatable as long as different variables are given in each specification. *lasso\_options* are `selection(...)`, `grid(...)`, `stop(#)`, `tolerance(#)`, `dtolerance(#)`, and `cvtolerance(#)`. When `lasso(varlist, selection(...))` is specified, it overrides any global `selection()` option for the variables in *varlist*. It also overrides the global `sqrtlasso` option for these variables. See [LASSO] [lasso options](#).

`sqrtlasso(varlist, lasso_options)` works like the option `lasso()`, except square-root lassos for the variables in *varlist* are done rather than regular lassos. *varlist* consists of one or more variables from *depvar* or *varsofinterest*. This option is repeatable as long as different variables are given in each specification. *lasso\_options* are `selection(...)`, `grid(...)`, `stop(#)`, `tolerance(#)`,

`dtolerance(#)`, and `cvtolerance(#)`. When `sqrtlasso(varlist, selection(...))` is specified, it overrides any global `selection()` option for the variables in *varlist*. See [LASSO] [lasso options](#).

The following options are available with `dsregress` but are not shown in the dialog box:

`reestimate` is an advanced option that refits the `dsregress` model based on changes made to the underlying lassos using `lassoselect`. After running `dsregress`, you can select a different  $\lambda^*$  for one or more of the lassos estimated by `dsregress`. After selecting  $\lambda^*$ , you type `dsregress, reestimate` to refit the `dsregress` model based on the newly selected  $\lambda$ 's.

`reestimate` may be combined only with reporting options.

`noheader` prevents the coefficient table header from being displayed.

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

## Remarks and examples

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

`dsregress` performs double-selection lasso linear regression. This command estimates coefficients, standard errors, and confidence intervals and performs tests for variables of interest while using lassos to select from among potential control variables.

The linear regression model is

$$\mathbf{E}[y|\mathbf{d}, \mathbf{x}] = \mathbf{d}\alpha' + \mathbf{x}\beta'$$

where  $\mathbf{d}$  are the variables for which we wish to make inferences and  $\mathbf{x}$  are the potential control variables from which the lassos select. `dsregress` reports estimated coefficients for  $\alpha$ . However, double-selection does not provide estimates of the coefficients on the control variables ( $\beta$ ) or their standard errors. No estimation results can be reported for  $\beta$ .

For an introduction to the double-selection lasso method for inference, as well as the partialing-out and cross-fit partialing-out methods, see [LASSO] [Lasso inference intro](#).

Examples that demonstrate how to use `dsregress` and the other lasso inference commands are presented in [LASSO] [Inference examples](#). In particular, we recommend reading [1 Overview](#) for an introduction to the examples and to the `v1` command, which provides tools for working with the large lists of variables that are often included when using lasso methods. See [2 Fitting and interpreting inferential models](#) for examples of fitting inferential lasso linear models and comparisons of the different methods available in Stata.

If you are interested in digging deeper into the lassos that are used to select controls, see [5 Exploring inferential model lassos](#) in [LASSO] [Inference examples](#).

## Stored results

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

### Scalars

<code>e(N)</code>	number of observations
<code>e(N_clust)</code>	number of clusters
<code>e(k_varsofinterest)</code>	number of variables of interest
<code>e(k_controls)</code>	number of potential control variables
<code>e(k_controls_sel)</code>	number of selected control variables
<code>e(df)</code>	degrees of freedom for test of variables of interest
<code>e(chi2)</code>	$\chi^2$
<code>e(p)</code>	$p$ -value for test of variables of interest
<code>e(rank)</code>	rank of $e(V)$

### Macros

<code>e(cmd)</code>	<code>dsregress</code>
<code>e(cmdline)</code>	command as typed
<code>e(depvar)</code>	name of dependent variable
<code>e(lasso_depvars)</code>	names of dependent variables for all lassos
<code>e(varsofinterest)</code>	variables of interest
<code>e(controls)</code>	potential control variables
<code>e(controls_sel)</code>	selected control variables
<code>e(model)</code>	linear
<code>e(title)</code>	title in estimation output
<code>e(clustvar)</code>	name of cluster variable
<code>e(chi2type)</code>	Wald; type of $\chi^2$ test
<code>e(vce)</code>	<i>vce</i> type specified in <code>vce()</code>
<code>e(vcetype)</code>	title used to label Std. err.
<code>e(rngstate)</code>	random-number state used
<code>e(properties)</code>	<code>b V</code>
<code>e(predict)</code>	program used to implement <code>predict</code>
<code>e(select_cmd)</code>	program used to implement <code>lassoselect</code>
<code>e(marginsnotok)</code>	predictions disallowed by <code>margins</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(V)</code>	variance–covariance matrix of the estimators

### Functions

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

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

### Matrices

<code>r(table)</code>	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

`dsregress` implements double-selection lasso as described in [Belloni, Chernozhukov, and Hansen \(2014\)](#). The regression model is

$$\mathbf{E}[y|\mathbf{d}, \mathbf{x}] = \mathbf{d}\boldsymbol{\alpha}' + \beta_0 + \mathbf{x}\boldsymbol{\beta}'$$

where  $\mathbf{d}$  contains the  $J$  covariates of interest and  $\mathbf{x}$  contains the  $p$  controls. The number of covariates in  $\mathbf{d}$  must be small and fixed. The number of controls in  $\mathbf{x}$  can be large and, in theory, can grow with the sample size; however, the number of nonzero elements in  $\boldsymbol{\beta}$  must not be too large, which is to say that the model must be sparse.

## Double-selection lasso algorithm

1. Perform a linear lasso of  $y$  on  $\mathbf{x}$ , and denote the selected variables by  $\tilde{\mathbf{x}}_y$ .

This lasso can choose the lasso penalty parameter ( $\lambda^*$ ) using the plugin estimator, adaptive lasso, or CV. The heteroskedastic plugin estimator for the linear lasso is the default.

2. For  $j = 1, \dots, J$ , perform a linear lasso of  $d_j$  on  $\mathbf{x}$ , and denote the selected controls by  $\tilde{\mathbf{x}}_j$ .

Each of these lassos can choose the lasso penalty parameter ( $\lambda_j^*$ ) using the plugin estimator, adaptive lasso, or CV. The heteroskedastic plugin estimator for the linear lasso is the default.

3. Let  $\hat{\mathbf{x}}$  be the distinct variables in the union of the variables in  $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_J$ , and  $\tilde{\mathbf{x}}_y$ .
4. Fit a linear regression of  $y$  on  $\mathbf{d}$  and  $\hat{\mathbf{x}}$ , denoting the estimated coefficient vectors by  $\hat{\boldsymbol{\alpha}}$  and  $\hat{\boldsymbol{\beta}}$ , respectively.
5. Store the point estimates  $\hat{\boldsymbol{\alpha}}$  in `e(b)` and their variance estimates (VCE) in `e(V)`.

Option `vce(robust)`, the robust estimator of the VCE for a linear regression, is the default. See [Methods and formulas](#) in [R] `regress` for details about option `vce(robust)` and the other VCE estimators available via options `vce(o1s)`, `vce(hc2)`, and `vce(hc3)`.

See [Methods and formulas](#) in [LASSO] `lasso` for details on how the lassos in steps 1 and 2 choose their penalty parameter ( $\lambda^*$ ).

## Reference

Belloni, A., V. Chernozhukov, and C. B. Hansen. 2014. Inference on treatment effects after selection among high-dimensional controls. *Review of Economic Studies* 81: 608–650. <https://doi.org/10.1093/restud/rdt044>.

## Also see

[LASSO] [lasso inference postestimation](#) — Postestimation tools for lasso inferential models

[LASSO] [poregress](#) — Partialing-out lasso linear regression

[LASSO] [xporegress](#) — Cross-fit partialing-out lasso linear regression

[R] [regress](#) — Linear 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 © 1985–2023 StataCorp LLC, College Station, TX, USA. All rights reserved.



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