lassoknots — Display knot table after lasso estimation

Description

lassoknots shows a table of knots after a lasso. Knots are the values of λ at which variables in the model change.

lassoknots displays the names of the variables added or removed as models are fit for successive λ's. When using cross-validation (CV) to select λ^*, lassoknots will display values of the CV function.

lassoknots also displays measures of fit. After viewing measures of fit, you can select an alternative λ^* using lassoselect.

When telasso, ds, po, and xpo commands fit models using selection(cv), selection(adaptive), or selection(bic) (see [LASSO] **lasso options**), lassoknots can be used to show the CV function (for cv and adaptive) or the BIC function or other measures of fit for each of the lassos computed.

lassoknots does work after selection(plugin) but only shows measures for the single λ^* estimated by the plugin formula.

Quick start

Show knot table after lasso, sqrtlasso, and elasticnet

```
lassoknots
```

Same as above, but show number of nonzero coefficients, out-of-sample R^2, and variables added or removed after a linear model

```
lassoknots, display(nonzero osr2 variables)
```

Same as above, but show in-sample R^2 and CV mean-prediction error in addition to out-of-sample R^2

```
lassoknots, display(osr2 r2 cvmpe)
```

After lasso logit, lasso probit, or lasso poisson, show out-of-sample mean-deviance ratio, in-sample deviance ratio, and Bayes information criterion (BIC)

```
lassoknots, display(cvdevratio devratio bic)
```

After a lasso fit with selection(adaptive), show knot tables for all adaptive steps

```
lassoknots, steps
```

After a ds or po estimation with selection(cv) or selection(adaptive), show the knot table for the lasso for the dependent variable y

```
lassoknots, for(y)
```

After poivregress, show the knot table for the lasso for the prediction of the endogenous variable whatup

```
lassoknots, for(pred(whatup))
```

After xporegress with option resample, show the knot table for the lasso for x for the 4th cross-fit fold of the 9th resample

```
lassoknots, for(x) xfold(4) resample(9)
```

After telasso estimation with selection(cv) or selection(adaptive), show the knot table for the lasso for the outcome variable y at treatment level 1

```
lassoknots, for(y) tlevel(1)
```

Menu

Statistics > Postestimation

Syntax

After lasso, sqrtlasso, *and* elasticnet

lassoknots [, *options*]

After ds *and* po

lassoknots, for(*varspec*) [*options*]

After xpo *without* resample

lassoknots, for(*varspec*) xfold(#) [*options*]

After xpo *with* resample

lassoknots, for(*varspec*) xfold(#) resample(#) [*options*]

After telasso *for the outcome variable*

lassoknots, for(*varspec*) tlevel(#) [*options*]

After telasso *for the treatment variable*

lassoknots, for(*varspec*) [*options*]

After telasso *for the outcome variable with cross-fitting but without* resample

lassoknots, for(*varspec*) tlevel(#) xfold(#) [*options*]

After telasso *for the treatment variable with cross-fitting but without* resample

lassoknots, for(*varspec*) xfold(#) [*options*]

After telasso *for the outcome variable with cross-fitting and* resample

lassoknots, for(*varspec*) tlevel(#) xfold(#) resample(#) [*options*]

After `telasso` *for the treatment variable with cross-fitting and* `resample`

`lassoknots, for(`*varspec*`) xfold(#) resample(#)` [*options*]

varspec is *varname*, except after `poivregress` and `xpoivregress`, when it is either *varname* or `pred(`*varname*`)`.

options	Description
`display(`*di_opts*`)`	specify what to display; maximum of three *di_opts* options
`alllambdas`	show all λ's
`steps`	show all adaptive steps; `selection(adaptive)` only
`nolstretch`	do not stretch the width of the table to accommodate long variable names
* `for(`*varspec*`)`	lasso for *varspec*; `telasso`, `ds`, `po`, and `xpo` commands only
* `xfold(#)`	lasso for the #th cross-fit fold; `xpo` commands and `telasso` with `xfolds` only
* `resample(#)`	lasso for the #th resample; `xpo` commands and `telasso` with `resample` only
* `tlevel(#)`	lasso for the outcome model with the treatment level #; `telasso` only

*`for(`*varspec*`)` is required for all `ds`, `po`, and `xpo` commands and for `telasso`.
`xfold(#)` is required for all `xpo` commands and for `telasso` when the option `xfolds(#)` was specified.
`resample(#)` is required for `xpo` and for `telasso` when the option `resample(#)` was specified.
`tlevel(#)` is required for the outcome model in `telasso`.
`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

di_opts	Description
`nonzero`	number of nonzero coefficients
`variables`	names of variables added or removed
`cvmd`	CV mean deviance (the CV function)
`cvdevratio`	CV mean-deviance ratio
`devratio`	in-sample deviance ratio
`bic`	BIC
`l1`	relative ℓ_1-norm of coefficients
`l2`	relative ℓ_2-norm squared of coefficients
Linear models only	
`cvmpe`	CV mean-prediction error (the CV function)
`osr2`	out-of-sample R^2
`r2`	in-sample R^2

Options

`display(`*di_opts*`)` specifies what to display in the knot table. A maximum of three *di_opts* options can be specified. For lassos fit using `selection(cv)` or `selection(adaptive)`, the default is `display(nonzero cvmpe variables)` for linear models and `display(nonzero cvmd variables)` for logit, probit, Poisson, and Cox models. For lassos fit using `selection(plugin)` or `selection(bic)`, the default is `display(nonzero r2 variables)` for linear models and `display(nonzero devratio variables)` for logit, probit, Poisson, and Cox models. The full set of *di_opts* is the following.

`nonzero` specifies that the number of nonzero coefficients be shown.

`variables` specifies that the names of variables added or removed at each knot be shown.

`cvmd` specifies that the CV mean deviance be shown. These are the values of the CV function that are searched for a minimum. For linear models, it is the same as the CV mean-prediction error given by `cvmpe`. `cvmd` is available only for lassos fit using `selection(cv)` or `selection(adaptive)`.

`cvdevratio` specifies that the CV mean-deviance ratio be shown. The CV mean-deviance ratio is an estimate of out-of-sample goodness of fit. As a measure of prediction performance, it is superior to `devratio`, the in-sample deviance ratio. It is typically between 0 and 1, but in some cases, it may be outside this range. For linear models, it is the same as out-of-sample R^2 given by `osr2`. `cvdevratio` is available only for lassos fit using `selection(cv)` or `selection(adaptive)`.

`devratio` specifies that the in-sample deviance ratio be shown. The in-sample deviance ratio is an indicator of in-sample goodness of fit. The in-sample deviance generalizes the in-sample R^2 to nonlinear models. As a measure of prediction performance, it is inferior to `cvdevratio`, the CV mean-deviance ratio. The in-sample deviance ratio is a poor measure of prediction performance because it does not capture the cost of including additional covariates for prediction. It is always between 0 and 1. For linear models, it is the same as in-sample R^2 given by `r2`.

`bic` specifies that the BIC be shown. Note that the BIC can be displayed for lassos fit using `selection(cv)` and `selection(adaptive)`, but the CV measures—`cvmd`, `cvdevratio`, and `cvmpe`—are not available for lassos fit using `selection(bic)`.

`l1` specifies that the relative ℓ_1-norm of coefficients be shown.

`l2` specifies that relative ℓ_2-norm squared of coefficients be shown.

Linear models only

`cvmpe` specifies that the CV mean-prediction error be shown. These are the values of the CV function that are searched for a minimum. `cvmpe` is available only for lassos fit using `selection(cv)` or `selection(adaptive)`.

`osr2` specifies that the out-of-sample R^2 be shown. The out-of-sample R^2 is an estimate of out-of-sample goodness of fit. As a measure of prediction performance, it is superior to `r2`, the in-sample R^2. It is typically between 0 and 1, but in some cases, it may be outside this range.

`r2` specifies that the in-sample deviance ratio be shown. The in-sample deviance ratio is an indicator of in-sample goodness of fit. As a measure of prediction performance, it is inferior to `osr2`, the out-of-sample R^2. The in-sample R^2 is a poor measure of prediction performance because it does not capture the cost of including additional covariates for prediction. It is always between 0 and 1.

alllambdas specifies that all λ's are to be shown, not just the knots. Measures at λ's that are not knots change slightly because the coefficient estimates change slightly. λ's that are not knots can be selected as λ^* by lassoselect; however, this is typically not done.

steps applies to selection(adaptive) only. When specified, λ's for all adaptive steps are shown. By default, λ's for only the last adaptive step are shown.

nolstretch specifies that the width of the table not be automatically widened to accommodate long variable names. When nolstretch is specified, names are abbreviated to make the table width no more than 79 characters. The default, lstretch, is to automatically widen the table up to the width of the Results window. To change the default, use set lstretch off.

for(*varspec*) specifies a particular lasso after telasso or after a ds, po, or xpo estimation command fit using the option selection(cv), selection(adaptive), or selection(bic). For all commands except poivregress and xpoivregress, *varspec* is always *varname*.

For the ds, po, and xpo commands except poivregress and xpoivregress, *varspec* is either *depvar*, the dependent variable, or one of *varsofinterest* for which inference is done.

For poivregress and xpoivregress, *varspec* is either *varname* or pred(*varname*). The lasso for *depvar* is specified with its *varname*. Each of the endogenous variables have two lassos, specified by *varname* and pred(*varname*). The exogenous variables of interest each have only one lasso, and it is specified by pred(*varname*).

For telasso, *varspec* is either the outcome variable or the treatment variable.

This option is required after telasso and after the ds, po, and xpo commands.

xfold(#) specifies a particular lasso after an xpo estimation command or after telasso when the option xfolds(#) was specified. For each variable to be fit with a lasso, K lassos are done, one for each cross-fit fold, where K is the number of folds. This option specifies which fold, where $\# = 1, 2, \ldots, K$. xfold(#) is required after an xpo command and after telasso when the option xfolds(#) was specified.

resample(#) specifies a particular lasso after an xpo estimation command or after telasso fit using the option resample(#). For each variable to be fit with a lasso, $R \times K$ lassos are done, where R is the number of resamples and K is the number of cross-fitting folds. This option specifies which resample, where $\# = 1, 2, \ldots, R$. resample(#), along with xfold(#), is required after an xpo command and after telasso with resampling.

tlevel(#) specifies the lasso for the outcome variable at the specified treatment level after telasso. This option is required to refer to the outcome model after telasso.

Remarks and examples

Remarks are presented under the following headings:

Introduction

When a lasso is fit over a grid of λ's, it starts with the smallest λ that produces a model with no selected variables. This initial λ is the largest λ in the grid. Lasso steps to the next λ and fits a model for it. One or more variables are selected for this second λ (if no variables were selected, it would be the starting λ). Lasso steps to the third λ, and more variables may be selected, or the model may have the same variables as the model for the second λ.

In this way, lasso iterates across the grid of λ values. λ's at which the selected variables change are called "knots". Variables are not only added at a knot but also sometimes removed. Typically, when a variable is removed, one or more variables are added in its place. Usually, the number of nonzero coefficients increases monotonically as λ gets smaller but not always. Occasionally, the net number of variables in the model goes down, rather than up, in an iteration to a smaller λ.

`lassoknots` displays a table of the knots, showing the names of variables that enter and leave the models. The option `alllambdas` can be specified to display all the λ's in the grid. To view all variables selected at a particular λ, you can use `lassoselect` to select that λ and then `lassocoef` to list the variables and, optionally, the coefficients.

Selection methods `selection(cv)`, `selection(adaptive)`, `selection(bic)`, and `selection(none)` fit models for each λ in the grid. The method `selection(plugin)` calculates λ^* using a formula so there is only one λ.

Measures of fit

`lassoknots` will also display other measures. The methods `selection(cv)` and `selection(adaptive)` use CV. When CV is performed, `lassoknots` by default displays the number of nonzero coefficients, the CV function, and the names of variables that enter or leave the model.

Optionally, there are five other measures that can be displayed. For linear models, they are in-sample R^2 (`r2`), estimates of out-of-sample R^2 (`osr2`), the BIC (`bic`), relative ℓ_1-norm of coefficients (`l1`), and relative ℓ_2-norm squared of coefficients (`l2`).

For nonlinear models, in place of the R^2 measures, there are the analogous measures, the in-sample deviance ratio (`devratio`) and estimates of out-of-sample deviance ratio (`cvdevratio`).

The in-sample measures, BIC, and relative norms are available regardless of whether CV was done.

The out-of-sample R^2 and out-of-sample deviance ratio are not computed on out-of-sample data, but rather they are estimates of what these measures would be on out-of-sample data. The CV procedure provides these estimates.

In-sample measures versus estimates of out-of-sample measures

Estimates of out-of-sample measures are superior to in-sample measures.

Consider a linear lasso. The set of covariates that produces the smallest out-of-sample MSE is the set that produces the best predictions. CV is used to estimate out-of-sample MSE and select the set that produces the smallest estimate.

In contrast, we should not use in-sample MSE to select the set of covariates. In-sample MSE systematically underestimates out-of-sample prediction error. In-sample MSE can be made smaller and smaller simply by including more covariates (as long as they are not collinear with covariates already in the model). In-sample MSE does not capture the cost of including more covariates.

For the same reason, estimates of out-of-sample R^2 are superior to in-sample R^2 for linear models. For logit, probit, and Poisson models, estimates of out-of-sample deviance ratios are superior to in-sample deviance ratios.

See Hastie, Tibshirani, and Friedman (2009, sec. 7.2) for an introduction to a comparison of in-sample and out-of-sample measures of the predictive ability of a model.

BIC

Information criteria, like the BIC, have a term that penalizes for each additional parameter. Selecting the set of covariates that minimizes the BIC is another way to select a set of covariates that will predict well out of sample. Zhang, Li, and Tsai (2010) show that the λ selected by minimizing the BIC will select a set of covariates close to the true set under the conditions described in their article.

In practice, the BIC is more informative than the in-sample measures reported by `lassoknots` for `selection(plugin)` and `selection(none)`.

Examples

▷ Example 1: lasso linear

Here is an example using `lasso` from [LASSO] **lasso examples**. We load the data and make the `vl` variable lists active.

```
. use https://www.stata-press.com/data/r19/fakesurvey_vl
(Fictitious survey data with vl)

. vl rebuild
Rebuilding vl macros ...
  (output omitted)
```

We fit a lasso linear model.

```
. lasso linear q104 $idemographics $ifactors $vlcontinuous, rseed(1234)

10-fold cross-validation with 100 lambdas ...
Grid value 1:     lambda = .9090511   no. of nonzero coef. =   0
Folds: 1...5....10   CVF = 18.33331
  (output omitted)
Grid value 28:    lambda = .0737359   no. of nonzero coef. =  80
Folds: 1...5....10   CVF = 11.92887
... cross-validation complete ... minimum found

Lasso linear model                          No. of obs         =        914
                                            No. of covariates =        277
Selection: Cross-validation                 No. of CV folds    =         10

---------------------------------------------------------------------------
             |                                No. of      Out-of-      CV mean
             |                               nonzero       sample   prediction
          ID |     Description      lambda     coef.    R-squared        error
-------------+-------------------------------------------------------------
           1 |    first lambda    .9090511         0      -0.0010     18.33331
          23 |   lambda before    .1174085        58       0.3543     11.82553
        * 24 |  selected lambda   .1069782        64       0.3547     11.81814
          25 |    lambda after    .0974746        66       0.3545      11.8222
          28 |     last lambda    .0737359        80       0.3487     11.92887
---------------------------------------------------------------------------
* lambda selected by cross-validation.
```

We run `lassoknots`.

```
. lassoknots

-------------------------------------------------------------------------------
      |                No. of   CV mean |
      |               nonzero     pred. |  Variables (A)dded, (R)emoved,
   ID |    lambda        coef.    error |      or left (U)nchanged
------+---------------------------------+--------------------------------------
    2 | .8282935          2   18.24362  | A 0.q19     0.q88
    3 | .7547102          4   17.99053  | A 0.q85     3.q156
    5 | .6265736          7   17.26211  | A 0.q48     0.q73     0.q101
    6 | .5709106         11    16.7744  | A 4.q38     q31       q76
      |                                 |   q139
    7 | .5201924         15   16.19275  | A 0.q5      2.q34     0.q43
      |                                 |   0.q50
    8 |   .47398         16   15.58941  | A q22
   11 | .3585485         19   14.07708  | A 0.q41     0.q56     2.q84
   12 |  .326696         22   13.69483  | A 3.q16     0.q89     0.q118
   13 | .2976732         25    13.3281  | A 0.q91     age       0.gender
   14 | .2712288         26   12.99274  | A 3.q38
   16 | .2251789         32   12.48904  | A 0.q3      0.q49     0.q150
      |                                 |   2.q155    0.q160    q111
   18 | .1869475         34   12.15245  | A 2.q6      3.q78
   19 | .1703396         39   12.03358  | A 0.q14     0.q33     0.q126
      |                                 |   0.q147    0.q149
   20 | .1552071         42   11.94361  | A 0.q25     0.q82     1.q110
   21 | .1414189         46   11.88652  | A 0.q96     q20       3.q110
      |                                 |   1.q134
   22 | .1288556         50   11.84693  | A 0.q32     0.q102    1.q105
      |                                 |   0.q122
   23 | .1174085         58   11.82553  | A 0.q4      0.q7      1.q34
      |                                 |   0.q40     3.q84     q53
      |                                 |   q93       2.q134
 * 24 | .1069782         64   11.81814  | A 0.q51     0.q55     0.q75
      |                                 |   0.q77     q63       0.q115
   25 | .0974746         66    11.8222  | A 3.q6      0.q117
   26 | .0888152         70   11.84669  | A 0.q59     3.q95     q21
      |                                 |   0.q125
   27 | .0809251         72   11.88463  | A 0.q100    4.q155
   28 | .0737359         80   11.92887  | A 0.q13     0.q30     0.q68
      |                                 |   q52       q70       2.q110
      |                                 |   0.q153    0.q159
-------------------------------------------------------------------------------
* lambda selected by cross-validation.
```

The table ends at the 28th λ. The default grid had 100 λ's. The iteration over the λ grid ended after a minimum of the CV function was found. There are other cases in which the iteration ends before the end of the grid is reached. See *The CV function* in [LASSO] **lasso** and [LASSO] **lasso fitting** for details.

The option `alllambdas` shows all the λ's for which models were fit. In this case, the first 28 λ's in the grid.

```
. lassoknots, alllambdas

-------------------------------------------------------------------------------
        |                    No. of   CV mean
        |                   nonzero     pred.  |    Variables (A)dded, (R)emoved,
     ID |     lambda          coef.     error  |          or left (U)nchanged
--------+--------------------------------------+-------------------------------
      1 |   .9090511             0  18.33331   | U
      2 |   .8282935             2  18.24362   | A 0.q19     0.q88
      3 |   .7547102             4  17.99053   | A 0.q85     3.q156
      4 |   .6876638             4   17.6434   | U
      5 |   .6265736             7  17.26211   | A 0.q48     0.q73     0.q101
      6 |   .5709106            11   16.7744   | A 4.q38     q31       q76
        |                                      |   q139
      7 |   .5201924            15  16.19275   | A 0.q5      2.q34     0.q43
        |                                      |   0.q50
      8 |     .47398            16  15.58941   | A q22
      9 |   .4318729            16  15.01285   | U
     10 |   .3935065            16  14.50648   | U
     11 |   .3585485            19  14.07708   | A 0.q41     0.q56     2.q84
     12 |    .326696            22  13.69483   | A 3.q16     0.q89     0.q118
     13 |   .2976732            25   13.3281   | A 0.q91     age       0.gender
     14 |   .2712288            26  12.99274   | A 3.q38
     15 |   .2471336            26  12.71385   | U
     16 |   .2251789            32  12.48904   | A 0.q3      0.q49     0.q150
        |                                      |   2.q155    0.q160    q111
     17 |   .2051746            32  12.30196   | U
     18 |   .1869475            34  12.15245   | A 2.q6      3.q78
     19 |   .1703396            39  12.03358   | A 0.q14     0.q33     0.q126
        |                                      |   0.q147    0.q149
     20 |   .1552071            42  11.94361   | A 0.q25     0.q82     1.q110
     21 |   .1414189            46  11.88652   | A 0.q96     q20       3.q110
        |                                      |   1.q134
     22 |   .1288556            50  11.84693   | A 0.q32     0.q102    1.q105
        |                                      |   0.q122
     23 |   .1174085            58  11.82553   | A 0.q4      0.q7      1.q34
        |                                      |   0.q40     3.q84     q53
        |                                      |   q93       2.q134
   * 24 |   .1069782            64  11.81814   | A 0.q51     0.q55     0.q75
        |                                      |   0.q77     q63       0.q115
     25 |   .0974746            66   11.8222   | A 3.q6      0.q117
     26 |   .0888152            70  11.84669   | A 0.q59     3.q95     q21
        |                                      |   0.q125
     27 |   .0809251            72  11.88463   | A 0.q100    4.q155
     28 |   .0737359            80  11.92887   | A 0.q13     0.q30     0.q68
        |                                      |   q52       q70       2.q110
        |                                      |   0.q153    0.q159
-------------------------------------------------------------------------------
* lambda selected by cross-validation.
```

The λ's that are not knots have a `U` for unchanged in the variables column. At these λ's, the variables in the model do not change, but their coefficient estimates do. In this example, the selected λ^* is a knot, but frequently the selected λ^* will not be a knot.

We display the number of nonzero coefficients again, but this time with estimates of out-of-sample R^2 and in-sample R^2.

```
. lassoknots, display(nonzero osr2 r2)
```

ID	lambda	No. of nonzero coef.	Out-of-sample R-squared	In-sample R-squared
2	.8282935	2	0.0039	0.0102
3	.7547102	4	0.0177	0.0278
5	.6265736	7	0.0575	0.0707
6	.5709106	11	0.0841	0.1051
7	.5201924	15	0.1159	0.1414
8	.47398	16	0.1488	0.1790
11	.3585485	19	0.2314	0.2635
12	.326696	22	0.2523	0.2861
13	.2976732	25	0.2723	0.3090
14	.2712288	26	0.2906	0.3288
16	.2251789	32	0.3181	0.3610
18	.1869475	34	0.3365	0.3870
19	.1703396	39	0.3430	0.3981
20	.1552071	42	0.3479	0.4081
21	.1414189	46	0.3510	0.4176
22	.1288556	50	0.3532	0.4263
23	.1174085	58	0.3543	0.4342
* 24	.1069782	64	0.3547	0.4418
25	.0974746	66	0.3545	0.4486
26	.0888152	70	0.3532	0.4546
27	.0809251	72	0.3511	0.4598
28	.0737359	80	0.3487	0.4647

```
* lambda selected by cross-validation.
```

In-sample R^2 is significantly larger than the estimates of out-of-sample R^2. As we discussed in *In-sample measures versus estimates of out-of-sample measures* above, in-sample R^2 should not be used for assessing fit. It is, however, occasionally useful for exposing problems with the specification of the set of potential covariates. For example, suppose our dependent variable is log-income and we accidentally include income as a potential covariate. It will no doubt be selected, and we will see an R^2 of 1 or close to it. Seeing that, we realize we made a mistake in the specification of potential variables.

We run `lassoknots` again to display BIC and the relative norms of the coefficient vectors.

```
. lassoknots, display(l1 l2 bic)
```

ID	lambda	BIC	Relative L1 length	Relative L2 length
2	.8282935	5262.546	0.0084	0.0013
3	.7547102	5259.79	0.0244	0.0060
5	.6265736	5238.991	0.0696	0.0313
6	.5709106	5231.834	0.1066	0.0544
7	.5201924	5221.257	0.1449	0.0840
8	.47398	5187.164	0.1903	0.1195
11	.3585485	5108.273	0.3092	0.2504
12	.326696	5100.274	0.3492	0.2982
13	.2976732	5090.95	0.3948	0.3487
14	.2712288	5071.186	0.4375	0.4001
16	.2251789	5067.137	0.5179	0.4999
18	.1869475	5042.754	0.5959	0.5949
19	.1703396	5060.244	0.6344	0.6398
20	.1552071	5065.277	0.6734	0.6834
21	.1414189	5077.835	0.7133	0.7259
22	.1288556	5091.401	0.7543	0.7677
23	.1174085	5133.245	0.7955	0.8091
* 24	.1069782	5161.662	0.8388	0.8503
25	.0974746	5164.198	0.8805	0.8904
26	.0888152	5181.477	0.9213	0.9286
27	.0809251	5186.25	0.9606	0.9651
28	.0737359	5232.569	1.0000	1.0000

* lambda selected by cross-validation.

The relative norms are relative to the coefficient vector for the last λ. If we were using BIC to select λ^*, we would have chosen λ at ID $= 18$.

◁

▷ Example 2: lasso logit

We fit a lasso logit model using the same data as in the previous example.

```
. lasso logit q106 $idemographics $ifactors $vlcontinuous, rseed(1234)
10-fold cross-validation with 100 lambdas ...
Grid value 1:      lambda = .0886291   no. of nonzero coef. =    0
Folds: 1...5....10   CVF = 1.386903
  (output omitted)
Grid value 27:     lambda = .0078899   no. of nonzero coef. =   87
Folds: 1...5....10   CVF = 1.103886
... cross-validation complete ... minimum found
Lasso logit model                           No. of obs         =        914
                                            No. of covariates  =        277
Selection: Cross-validation                 No. of CV folds    =         10
```

ID	Description	lambda	No. of nonzero coef.	Out-of-sample dev. ratio	CV mean deviance
1	first lambda	.0886291	0	-0.0004	1.386903
23	lambda before	.0114469	68	0.2102	1.094953
* 24	selected lambda	.01043	76	0.2103	1.09471
25	lambda after	.0095034	79	0.2091	1.096417
27	last lambda	.0078899	87	0.2037	1.103886

* lambda selected by cross-validation.

The default `lassoknots` gives a table that is the same as that for a linear model, except that instead of CV mean-prediction error, CV mean deviance is shown. The CV function for logit (and probit and Poisson) is the CV mean deviance.

```
. lassoknots
```

ID	lambda	No. of nonzero coef.	CV mean deviance	Variables (A)dded, (R)emoved, or left (U)nchanged
2	.0807555	3	1.38295	A 0.q90 2.q134 0.q142
3	.0735814	5	1.37237	A 0.q8 q53
4	.0670447	8	1.357427	A 0.q68 0.q77 q22
5	.0610886	9	1.33969	A 0.q46
6	.0556616	12	1.319525	A 0.q13 2.q16 2.q95
7	.0507168	14	1.299571	A 1.q84 q20
8	.0462113	18	1.279802	A 0.q29 0.q133 0.q140 1.q144
(output omitted)				
23	.0114469	68	1.094953	A 0.q26 0.q73 0.q118
* 24	.01043	76	1.09471	A 0.q4 q1 0.q50 2.q65 3.q65 0.q83 q24 1.q155
25	.0095034	79	1.096417	A q76 0.q108 0.q122
26	.0086591	83	1.09945	A 2.q6 0.q64 0.q100 q132
27	.0078899	87	1.103886	A 0.q58 0.q74 0.q113 q103

* lambda selected by cross-validation.

We can look at in-sample CV deviance ratio and estimates of out-of-sample CV deviance ratio. These are analogous to the linear in-sample R^2 and out-of-sample R^2. The in-sample CV deviance ratio is always between 0 and 1. The estimates of out-of-sample CV deviance ratio are usually, but not always, between 0 and 1.

```
. lassoknots, display(cvdevratio devratio bic)
```

ID	lambda	Out-of-sample dev. ratio	In-sample deviance ratio	BIC
2	.0807555	0.0024	0.0057	1287.176
3	.0735814	0.0100	0.0180	1285.111
4	.0670447	0.0208	0.0323	1287.477
5	.0610886	0.0336	0.0488	1273.364
6	.0556616	0.0482	0.0657	1272.417
7	.0507168	0.0626	0.0835	1263.5
8	.0462113	0.0768	0.1022	1267.165
(output omitted)				
23	.0114469	0.2102	0.3209	1330.907
* 24	.01043	0.2103	0.3297	1374.27
25	.0095034	0.2091	0.3379	1384.306
26	.0086591	0.2069	0.3461	1401.188
27	.0078899	0.2037	0.3535	1419.149

```
* lambda selected by cross-validation.
```

◁

▷ Example 3: dsregress

We load the data used in [LASSO] **lasso examples**. See that entry for details about the data.

```
. use https://www.stata-press.com/data/r19/fakesurvey_vl, clear
(Fictitious survey data with vl)
. vl rebuild
Rebuilding vl macros ...
```
(output omitted)

We are going to fit a `dsregress` model with `q104` as our dependent variable and variables of interest `q41` and `q22`. These variables of interest are currently in the variable lists `factors` and `vlcontinuous`, which we will use to specify the control variables. So we need to move them out of these variable lists.

```
. vl modify factors = factors - (q41)
note: 1 variable removed from $factors.
. vl move (q22) vlother
note: 1 variable specified and 1 variable moved.
```
(output omitted)
```
. vl rebuild
Rebuilding vl macros ...
```
(output omitted)

After we moved the variables out of the variable lists, we typed `vl rebuild` to update the variable list `ifactors` created from `factors`. See [D] **vl** for details.

We fit our dsregress model using the default plugin selection method.

```
. dsregress q104 i.q41 q22, controls(($idemographics) $ifactors $vlcontinuous)
Estimating lasso for q104 using plugin
Estimating lasso for 1bn.q41 using plugin
Estimating lasso for q22 using plugin
Double-selection linear model         Number of obs               =        914
                                      Number of controls          =        274
                                      Number of selected controls =         33
                                      Wald chi2(2)                =      18.72
                                      Prob > chi2                 =     0.0001
```

q104	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
q41						
Yes	.8410538	.2691082	3.13	0.002	.3136114	1.368496
q22	-.0878443	.0310435	-2.83	0.005	-.1486884	-.0270001

```
Note: Chi-squared test is a Wald test of the coefficients of the variables
      of interest jointly equal to zero. Lassos select controls for model
      estimation. Type lassoinfo to see number of selected variables in each
      lasso.
```

lassoinfo shows the lassos that dsregress fit.

```
. lassoinfo
    Estimate: active
     Command: dsregress
```

Variable	Model	Selection method	lambda	No. of selected variables
q104	linear	plugin	.1467287	18
1bn.q41	linear	plugin	.1467287	16
q22	linear	plugin	.1467287	15

The knot table for the lasso for the dependent variable q104 can be seen using the for(q104) option. We also show BIC and in-sample R^2.

```
. lassoknots, display(nonzero r2 bic) for(q104)
```

ID	lambda	No. of nonzero coef.	In-sample R-squared	BIC
* 1	.1467287	14	0.1623	5191.862

```
* lambda selected by plugin assuming heteroskedastic errors.
```

A lasso fit with plugin fits only one model for one λ. So that is all we get from lassoknots.

If we wanted to see the same table for the variable of interest i.q41, we would type

```
. lassoknots, display(nonzero r2 bic) for(1bn.q41)
```

In the for() option, we specify the variable name for the lasso exactly as it is shown in lassoinfo.

We run dsregress again, this time specifying `selection(cv)`.

```
. dsregress q104 i.q41 q22,
> controls(($idemographics) $ifactors $vlcontinuous)
> selection(cv) rseed(1234)
Estimating lasso for q104 using cv
Estimating lasso for 1bn.q41 using cv
Estimating lasso for q22 using cv
Double-selection linear model         Number of obs               =        914
                                      Number of controls          =        274
                                      Number of selected controls =        123
                                      Wald chi2(2)                =      10.96
                                      Prob > chi2                 =     0.0042
```

q104	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
q41						
Yes	.6003918	.2848483	2.11	0.035	.0420994	1.158684
q22	-.0681067	.0306219	-2.22	0.026	-.1281246	-.0080888

```
Note: Chi-squared test is a Wald test of the coefficients of the variables
      of interest jointly equal to zero. Lassos select controls for model
      estimation. Type lassoinfo to see number of selected variables in each
      lasso.
```

`lassoknots` now shows knots up to the minimum and slightly passed it.

```
. lassoknots, display(nonzero cvmpe osr2) for(q104)
```

ID	lambda	No. of nonzero coef.	CV mean pred. error	Out-of-sample R-squared
1	.864369	4	17.9727	0.0187
2	.7875809	6	17.88282	0.0236
3	.7176144	7	17.64713	0.0365
4	.6538635	8	17.32777	0.0539
5	.595776	12	16.87904	0.0784
6	.5428489	14	16.3203	0.1089
7	.4946237	15	15.74852	0.1401
8	.4506827	18	15.2143	0.1693
(output omitted)				
22	.1225221	52	12.02453	0.3435
* 23	.1116376	59	12.02148	0.3436
24	.10172	62	12.02571	0.3434
25	.0926835	71	12.03785	0.3427
26	.0844497	76	12.0626	0.3414
27	.0769474	80	12.09713	0.3395
27	.0769474	80	12.09713	0.3395

```
* lambda selected by cross-validation.
```

For a sensitivity analysis that uses `lassoselect` after `lassoknots`, see [LASSO] **lassoselect**.

◁

Stored results

`lassoknots` stores the following in `r()`:

Matrices
`r(table)` matrix containing the values displayed

Methods and formulas

Methods and formulas are presented under the following headings:

Overview

All the reported statistics depend on the p-dimensional coefficient vector $\widehat{\boldsymbol{\beta}}_\lambda$, which is the penalized estimate of $\boldsymbol{\beta}$ for given penalty value λ.

We present the formulas in the context of `lasso`, but formulas for `elasticnet` and `sqrtlasso` are the same, although the context would have some subtle differences that we can safely ignore.

Statistics that measure the size of the coefficient vector

Option `display(nonzero)` displays the number of nonzero coefficients, which is given by

$$\texttt{nonzero} = \sum_{j=1}^{p} d_j$$

$$d_j = \begin{cases} 1 & \text{if } \widehat{\beta}_{\lambda,j} \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

Option `display(l1)` displays the sum of the absolute values of the coefficients, which is known as the ℓ_1-norm:

$$\texttt{l1} = \sum_{j=1}^{p} |\widehat{\beta}_{\lambda,j}|$$

Option `display(l2)` displays the sum of the squared values of the coefficients, which is the square of the ℓ_2-norm:

$$\texttt{l2} = \sum_{j=1}^{p} \widehat{\beta}_{\lambda,j}^2$$

Statistics that measure fit

All statistics that measure fit are functions of the observation-level contributions of either the squared prediction error, spe_i, or the log likelihood, ℓ_i.

The contribution of observation i to a statistic can be calculated using a single-sample calculation or using CV. The CV version estimates the out-of-sample equivalent. The single-sample versions are in-sample measures that do not reliably estimate their out-of-sample equivalents.

CV measures of fit

When CV is performed, CV versions of spe_i and ℓ_i are available. Here is how we compute these observation-level quantities.

1. The data are partitioned into K folds.
2. For each value of λ,
   a. the coefficients are estimated on the observations not in fold k using λ.
   b. for each observation i in fold k, the fit measures spe_i and ℓ_i are computed using the penalized coefficient estimates.

Single-sample measures of fit

The single-sample measures of fit are computed as follows.

1. For each value of λ,
   a. the coefficients are estimated on all the observations using λ.
   b. for each observation i the fit measures spe_i and ℓ_i are computed using the penalized coefficient estimates.

Deviance formulas

The CV version of ℓ_i is used in the formulas for `cvmd` and `cvdevratio`. The single-sample version of ℓ_i is used in the formula for `devratio`.

For all models, the deviance, D_i, for the ith observation is given by

$$D_i = -2(\ell_i - \ell_{\text{saturated}})$$

where ℓ_i is the value of the log-likelihood function at observation i, and $\ell_{\text{saturated}}$ is the value of the saturated log-likelihood function. Formulas for the ℓ_i and for the $\ell_{\text{saturated}}$ are given below. The penalized coefficient estimates are used in these computations.

The mean deviance $\overline{D}$ is given by

$$\overline{D} = \frac{1}{N}\sum_{i=1}^{N} D_i$$

The formula for the deviance ratio D_2 is

$$D_2 = 1 - \frac{\overline{D}}{D_{\text{null}}}$$

where the D_{null} is the null deviance and is given by

$$D_{\text{null}} = \frac{1}{N}\sum_{i=1}^{N} -2(\ell_{0,i} - \ell_{\text{saturated}})$$

and $\ell_{0,i}$ is the ith observation of the log likelihood from the model that includes only a constant term.

Saturated log likelihood

For linear, logit, and probit models, the log-likelihood function of the saturated model is zero. For the Poisson model,

$$\ell_{\text{saturated}} = \frac{1}{N}\sum_{i=1}^{N}(-y_i + y_i \ln y_i)$$

For the Cox model,

$$\ell_{\text{saturated}} = -\frac{1}{N}\sum_{j=1}^{N_f} d_j \log\left(d_j\right)$$

where j indexes the ordered failure times $t_{(j)}$, $j = 1, \dots, N_f$; D_j is the set of observations that fail at $t_{(j)}$; and d_j is the number of observations in D_j.

Prediction error formulas

These formulas are used only for linear models. The squared prediction error for the ith observation is given by

$$\text{spe}_i = \left(y_i - \mathbf{x}_i\widehat{\boldsymbol{\beta}}_\lambda\right)^2$$

where y_i is the ith observation of the dependent variable and $\mathbf{x}_i\widehat{\boldsymbol{\beta}}_\lambda$ is the predicted mean of y_i conditional on $\mathbf{x}_i$.

For `cvmpe` and `osr2`, the CV version of spe_i is used. For `r2`, the single-sample version of spe_i is used.

R^2 is given by

$$R^2 = 1 - \frac{\text{MSE}}{\text{MSE}_{\text{null}}}$$

where the mean squared error (MSE) is given by

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\text{spe}_i$$

and the MSE of the null model is given by

$$\text{MSE}_{\text{null}} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \overline{y})^2$$

where $\overline{y}$ is the sample average of y.

BIC formula

BIC is given by

$$\text{BIC} = -2\ell + k \ln N$$

where $\ell = \sum_{i=1}^{N} \ell_i$, $k = \texttt{nonzero} + 1$ is the number of coefficients in the model including the constant term, and each ℓ_i is always calculated using the single-sample methods.

References

Hastie, T. J., R. J. Tibshirani, and J. H. Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd ed. New York: Springer. https://doi.org/10.1007/978-0-387-84858-7.

Zhang, Y., R. Li, and C.-L. Tsai. 2010. Regularization parameter selections via generalized information criterion. *Journal of the American Statistical Association* 105: 312–323. https://doi.org/10.1198/jasa.2009.tm08013.

Also see

[LASSO] **lasso** — Lasso for prediction and model selection

[LASSO] **lasso fitting** — The process (in a nutshell) of fitting lasso models

[LASSO] **lassocoef** — Display coefficients after lasso estimation results

[LASSO] **lassoselect** — Select lambda after lasso

[LASSO] **lasso inference postestimation** — Postestimation tools for lasso inferential models

[CAUSAL] **telasso postestimation** — Postestimation tools for telasso