lassoinfo — Display information about lasso estimation results

Description

lassoinfo displays basic information about the lasso or lassos fit by all commands that fit lassos.

Quick start

After any command that fits lassos

    lassoinfo
dsregress was run and the results stored under the name mygreatmodel using estimates store; show information about all the lassos in mygreatmodel

    lassoinfo mygreatmodel

As above, but three models were stored

    lassoinfo mygreatmodel mygoodmodel myfairmodel

After an xpo command, show information about every single lasso fit

    lassoinfo, each

Menu

Statistics > Postestimation

Syntax

For all lasso estimation results

    lassoinfo [ namelist ]

For xpo estimation results

    lassoinfo [ namelist ] [ , each ]

namelist is a name of a stored estimation result, a list of names, _all, or *. _all and * mean the same thing. See [R] estimates store.

Option

each applies to xpo models only. It specifies that information be shown for each lasso for each cross-fit fold to be displayed. If resample was specified, then information is shown for each lasso for each cross-fit fold in each resample. By default, summary statistics are shown for the lassos.
Remarks and examples

lassoinfo is intended for use after ds, po, and xpo commands to see basic information about the lassos they fit. It is a good idea to always run lassoinfo after these commands to see how many variables were selected in each lasso.

Running lassoinfo is a first step toward doing a sensitivity analysis. The lassos listed by lassoinfo can be examined using coefpath, cvplot, lassocoef, lassoknots, and lassoselect.

Example 1: lasso

lassoinfo works after lasso, sqrlasso, and elasticnet, but it does not display much useful information for these commands.

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/r16/fakesurvey_vl
(Fictitious Survey Data with vl)
. vl rebuild
Rebuilding vl macros ...
(output omitted)
```

We fit the lasso.

```
. 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

<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
<th>lambda</th>
<th>no. of nonzero coef.</th>
<th>R-squared</th>
<th>CV mean prediction error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>first lambda</td>
<td>.9090511</td>
<td>0</td>
<td>0.0010</td>
<td>18.33331</td>
</tr>
<tr>
<td>23</td>
<td>lambda before</td>
<td>.1174085</td>
<td>58</td>
<td>0.3543</td>
<td>11.82553</td>
</tr>
<tr>
<td>* 24</td>
<td>selected lambda</td>
<td>.1069782</td>
<td>64</td>
<td>0.3547</td>
<td>11.81814</td>
</tr>
<tr>
<td>25</td>
<td>lambda after</td>
<td>.0974746</td>
<td>66</td>
<td>0.3545</td>
<td>11.8222</td>
</tr>
<tr>
<td>28</td>
<td>last lambda</td>
<td>.0737359</td>
<td>80</td>
<td>0.3487</td>
<td>11.92887</td>
</tr>
</tbody>
</table>

* lambda selected by cross-validation.

lassoinfo tells us nothing new.

```
. lassoinfo
Estimate: active
Command: lasso
```

<table>
<thead>
<tr>
<th>Depvar</th>
<th>Selection Model</th>
<th>Selection Method</th>
<th>Selection Criterion</th>
<th>Lambda</th>
<th>No. of selected variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>q104</td>
<td>linear</td>
<td>cv</td>
<td>CV min.</td>
<td>.1069782</td>
<td>64</td>
</tr>
</tbody>
</table>
Replaying the command gives more information.

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

<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
<th>lambda</th>
<th>0</th>
<th>R-squared</th>
<th>error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>first lambda</td>
<td>.9090511</td>
<td>0</td>
<td>0.0010</td>
<td>18.33331</td>
</tr>
<tr>
<td>23</td>
<td>lambda before</td>
<td>.1174085</td>
<td>58</td>
<td>0.3543</td>
<td>11.82553</td>
</tr>
<tr>
<td>*24</td>
<td>selected lambda</td>
<td>.1069782</td>
<td>64</td>
<td>0.3547</td>
<td>11.81814</td>
</tr>
<tr>
<td>25</td>
<td>lambda after</td>
<td>.0974746</td>
<td>66</td>
<td>0.3545</td>
<td>11.82222</td>
</tr>
<tr>
<td>28</td>
<td>last lambda</td>
<td>.0737359</td>
<td>80</td>
<td>0.3487</td>
<td>11.92887</td>
</tr>
</tbody>
</table>
```

* lambda selected by cross-validation.

Example 2: dsregress

lassoinfo gives important information after the ds, po, and xpo commands.

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

```
. use https://www.stata-press.com/data/r16/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 cross-validation to select $\lambda^*$'s in the lassos.

```
. 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

<table>
<thead>
<tr>
<th>Variable</th>
<th>Number of obs</th>
<th>Number of controls</th>
<th>Number of selected controls</th>
<th>Wald chi2(2)</th>
<th>Prob &gt; chi2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>914</td>
<td>274</td>
<td>119</td>
<td>10.96</td>
<td>0.0042</td>
</tr>
</tbody>
</table>

```
. `lassoinfo`
```

```
| Variable   | Coef.  | Std. Err. | z      | P>|z| | [95% Conf. Interval] |
|------------|--------|-----------|--------|------|---------------------|
| q104       | .6003918 | .2848483  | 2.11   | 0.035 | .0420994 1.158684   |
| q41 (yes)  | -.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.

`lassoinfo` shows us how many variables were selected in each lasso.

```
. lassoinfo
```

```
Estimate: active
Command: `dsregress`
```

```
<table>
<thead>
<tr>
<th>Variable</th>
<th>Model</th>
<th>Selection method</th>
<th>Selection criterion</th>
<th>lambda</th>
<th>No. of selected variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>q104</td>
<td>linear</td>
<td>cv</td>
<td>CV min.</td>
<td>.1132914</td>
<td>59</td>
</tr>
<tr>
<td>1bn.q41</td>
<td>linear</td>
<td>cv</td>
<td>CV min.</td>
<td>.0137972</td>
<td>64</td>
</tr>
<tr>
<td>q22</td>
<td>linear</td>
<td>cv</td>
<td>CV min.</td>
<td>.1648102</td>
<td>45</td>
</tr>
</tbody>
</table>
```
lassoinfo also gives useful information after fitting the model using the default selection(plugin).

. 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

<table>
<thead>
<tr>
<th></th>
<th>Number of obs</th>
<th>Number of controls</th>
<th>Number of selected controls</th>
<th>Wald chi2(2)</th>
<th>Prob &gt; chi2</th>
</tr>
</thead>
<tbody>
<tr>
<td>q104</td>
<td>914</td>
<td>274</td>
<td>29</td>
<td>18.72</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

Wald chi2(2) = 18.72
Prob > chi2 = 0.0001

| Variable | Coef.   | Std. Err. | z       | P>|z|     | [95% Conf. Interval] |
|----------|---------|-----------|---------|---------|---------------------|
| q104     | .8410538| .2691082  | 3.13    | 0.002   | .3136114 1.368496   |
| q41      | yes     | .0878443  | .0310435| -2.83   | 0.005   | -.1486884 -.0270001|
| 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

Estimate: active
Command: dsregress

<table>
<thead>
<tr>
<th>Variable</th>
<th>Selection</th>
<th>No. of selected</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Model</td>
<td>method</td>
</tr>
<tr>
<td>q104</td>
<td>linear</td>
<td>plugin</td>
</tr>
<tr>
<td>1bn.q41</td>
<td>linear</td>
<td>plugin</td>
</tr>
<tr>
<td>q22</td>
<td>linear</td>
<td>plugin</td>
</tr>
</tbody>
</table>

See [LASSO] lassoselect, where we continue this example and do a sensitivity analysis to examine the differences between the lassos fit using cross-validation and the lassos fit using the plugin estimator.
Example 3: poivregress

We want to show you some differences that arise when you fit models containing endogenous variables using poivregress and xpoivregress.

We will not describe the data or the model here. See [LASSO] Inference examples.

We load the data.

```
. use https://www.stata-press.com/data/r16/mroz2, clear
```

set vl variable lists,

```
. vl create vars = (kidslt6 kidsge6 age husage city exper)
note: $vars initialized with 6 variables
. vl substitute vars2 = c.vars c.vars#c.vars
. vl create iv = (huseduc motheduc fatheduc)
note: $iv initialized with 3 variables
. vl substitute iv2 = c.iv c.iv#c.iv
```

and fit our model using poivregress.

```
. poivregress lwage (educ = $iv2), controls($vars2) selection(cv) rseed(12345)
```

```
Estimating lasso for lwage using cv
note: city dropped because of collinearity with another variable
Estimating lasso for educ using cv
note: city dropped because of collinearity with another variable
Estimating lasso for pred(educ) using cv
note: city dropped because of collinearity with another variable
```

```
Partialing-out IV linear model
```

```

<table>
<thead>
<tr>
<th>Number of obs</th>
<th>428</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of controls</td>
<td>27</td>
</tr>
<tr>
<td>Number of instruments</td>
<td>9</td>
</tr>
<tr>
<td>Number of selected controls</td>
<td>16</td>
</tr>
<tr>
<td>Number of selected instruments</td>
<td>4</td>
</tr>
</tbody>
</table>

| Wald chi2(1) | 11.10 |
| Prob > chi2  | 0.0009 |

```

| lwage | Robust
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Coef.</td>
<td>Std. Err.</td>
</tr>
<tr>
<td>-------</td>
<td>------------</td>
</tr>
<tr>
<td>educ</td>
<td>.0765154</td>
</tr>
</tbody>
</table>
```

Endogenous: educ

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.

```
. estimates store poivregresscv
```
We stored our estimation results using `estimates store`, and here we use `lassoinfo` with the name used to store them.

```
. lassoinfo poivregresscv
   Estimate: poivregresscv
   Command: poivregress

<table>
<thead>
<tr>
<th>Variable</th>
<th>Model</th>
<th>Selection method</th>
<th>Selection criterion</th>
<th>lambda</th>
<th>No. of selected variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>lwage</td>
<td>linear</td>
<td>cv</td>
<td>CV min.</td>
<td>.0353704</td>
<td>3</td>
</tr>
<tr>
<td>educ</td>
<td>linear</td>
<td>cv</td>
<td>CV min.</td>
<td>.0530428</td>
<td>10</td>
</tr>
<tr>
<td>pred(educ)</td>
<td>linear</td>
<td>cv</td>
<td>CV min.</td>
<td>.013186</td>
<td>12</td>
</tr>
</tbody>
</table>
```

Note that we have two lassos for `educ` labeled by `lassoinfo` as `educ` and `pred(educ)`. `poivregress` and `xpoivregress` perform two lassos for each endogenous variable, one for the endogenous variable and one for its prediction. `lassoinfo` shows us how to refer to each of these lassos in other postestimation commands using the `for()` option. In this example, we would type `for(educ)` and `for(pred(educ))`, respectively.

> **Example 4: `xporegress`**

The `xpo` commands fit many lassos. For each lasso fit by a `po` command, the corresponding `xpo` command fits `xfolds(#) × resample(#)` lassos. `lassoinfo` can be used to get information about these lassos.

We will not describe the data or the model here. See [LASSO] Inference examples.

We load the data,

```
. use https://www.stata-press.com/data/r16/breathe, clear
   (Nitrogen dioxide and attention)
```

set vl variable lists,

```
. vl set
   (output omitted)
. vl move (siblings_old siblings_young) vlcontinuous
   note: 2 variables specified and 2 variables moved
   (output omitted)
. vl create mycontinuous = vlcontinuous - (react no2_class)
   note: $mycontinuous initialized with 10 variables
. vl substitute mycontrols = i.vl_categorical mycontinuous
```
and fit our model using \texttt{xporegress} with the options \texttt{xfolds(3)} and \texttt{resample(2)}.

\begin{verbatim}
. xporegress react no2_class, controls($mycontrols) xfolds(3) resample(2) 
> selection(cv) rseed(12345)
\end{verbatim}

Resample 1 of 2 ... 
Cross-fit fold 1 of 3 ... 

Resample 1 of 2 ... 
Cross-fit fold 2 of 3 ... 

Resample 1 of 2 ... 
Cross-fit fold 3 of 3 ... 

Resample 2 of 2 ... 
Cross-fit fold 1 of 3 ... 

Resample 2 of 2 ... 
Cross-fit fold 2 of 3 ... 

Resample 2 of 2 ... 
Cross-fit fold 3 of 3 ... 

Cross-fit partialing-out 
linear model

<table>
<thead>
<tr>
<th>Variable</th>
<th>Selection method</th>
<th>min</th>
<th>median</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>react</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>no2_class</td>
<td></td>
<td>11</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>react</td>
<td></td>
<td>9</td>
<td>15</td>
<td>19</td>
</tr>
</tbody>
</table>

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 \texttt{lassoinfo} to see number of selected variables in each lasso.

For each cross-fit fold and each resample, \texttt{xporegress} fits lassos. So it fit six lassos for the dependent variable, \texttt{react}, and six for the variable of interest, \texttt{no2_class}. \texttt{lassoinfo} summarizes the numbers of variables selected across these six lassos for \texttt{react} and \texttt{no2_class}.

\begin{verbatim}
. lassoinfo
Estimate: active
Command: xporegress
\end{verbatim}
Specifying the option `each` gives us information on each lasso.

```
. lassoinfo, each
  Estimate: active
  Command: xporegress
```

<table>
<thead>
<tr>
<th>Depvar</th>
<th>Model</th>
<th>Selection</th>
<th>Resample</th>
<th>xfold</th>
<th>No. of</th>
<th>Selection</th>
<th>lambda</th>
<th>var.</th>
</tr>
</thead>
<tbody>
<tr>
<td>no2_class</td>
<td>linear</td>
<td>cv</td>
<td>1</td>
<td>1</td>
<td>CV min.</td>
<td>.2663004</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>no2_class</td>
<td>linear</td>
<td>cv</td>
<td>1</td>
<td>2</td>
<td>CV min.</td>
<td>.2860957</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>no2_class</td>
<td>linear</td>
<td>cv</td>
<td>1</td>
<td>3</td>
<td>CV min.</td>
<td>.2887414</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>no2_class</td>
<td>linear</td>
<td>cv</td>
<td>2</td>
<td>1</td>
<td>CV min.</td>
<td>.2337636</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>no2_class</td>
<td>linear</td>
<td>cv</td>
<td>2</td>
<td>2</td>
<td>CV min.</td>
<td>.2824076</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>no2_class</td>
<td>linear</td>
<td>cv</td>
<td>2</td>
<td>3</td>
<td>CV min.</td>
<td>.2515777</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>react</td>
<td>linear</td>
<td>cv</td>
<td>1</td>
<td>1</td>
<td>CV min.</td>
<td>6.07542</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>react</td>
<td>linear</td>
<td>cv</td>
<td>1</td>
<td>2</td>
<td>CV min.</td>
<td>1.704323</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>react</td>
<td>linear</td>
<td>cv</td>
<td>1</td>
<td>3</td>
<td>CV min.</td>
<td>3.449884</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>react</td>
<td>linear</td>
<td>cv</td>
<td>2</td>
<td>1</td>
<td>CV min.</td>
<td>6.034922</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>react</td>
<td>linear</td>
<td>cv</td>
<td>2</td>
<td>2</td>
<td>CV min.</td>
<td>4.31785</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>react</td>
<td>linear</td>
<td>cv</td>
<td>2</td>
<td>3</td>
<td>CV min.</td>
<td>4.096779</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

See [LASSO] lassocoef for an example where we list the variables selected by each lasso.

**Stored results**

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

- **Macros**
  - `r(names)`
    - names of estimation results displayed
- **Matrices**
  - `r(table)`
    - matrix containing the numerical values displayed

**Also see**

- [LASSO] lassoselect — Select lambda after lasso
- [LASSO] lasso postestimation — Postestimation tools for lasso for prediction
- [LASSO] lasso inference postestimation — Postestimation tools for lasso inferential models