Package ‘lars’

February 20, 2015

Type Package

Version 1.2

Date 2013-04-23

Title Least Angle Regression, Lasso and Forward Stagewise

Author Trevor Hastie <hastie@stanford.edu> and Brad Efron
<brad@stat.stanford.edu>

Maintainer Trevor Hastie <hastie@stanford.edu>

Description Efficient procedures for fitting an entire lasso sequence
with the cost of a single least squares fit. Least angle
regression and infinitesimal forward stagewise regression are
related to the lasso, as described in the paper below.

Depends R (>= 2.10)

License GPL-2

URL http://www.stat.stanford.edu/~hastie/Papers/#LARS

NeedsCompilation yes

Repository CRAN

Date/Publication 2013-04-24 09:46:12

R topics documented:

cv.lars ................................................................. 2
diabetes .................................................................. 3
lars ...................................................................... 4
plot.lars ............................................................... 5
predict.lars ............................................................ 7
summary.lars ......................................................... 8

Index 10
cv.lars

**Computes K-fold cross-validated error curve for lars**

**Description**

Computes the K-fold cross-validated mean squared prediction error for lars, lasso, or forward stagewise.

**Usage**

```r
cv.lars(x, y, K = 10, index, trace = FALSE, plot.it = TRUE, se = TRUE, type = c("lasso", "lar", "forward.stagewise", "stepwise"), mode = c("fraction", "step"), ...)```

**Arguments**

- `x`: Input to lars
- `y`: Input to lars
- `K`: Number of folds
- `index`: Abscissa values at which CV curve should be computed. If `mode="fraction"` this is the fraction of the saturated `|beta|`. The default value in this case is `index=seq(from = 0, to = 1, length =100)`. If `mode="step"`, this is the number of steps in lars procedure. The default is complex in this case, and depends on whether `N>p` or not. In principal it is `index=1:p`. Users can supply their own values of index (with care).
- `trace`: Show computations?
- `plot.it`: Plot it?
- `se`: Include standard error bands?
- `type`: Type of lars fit, with default "lasso"
- `mode`: This refers to the index that is used for cross-validation. The default is "fraction" for `type="lasso"` or type="forward.stagewise". For `type="lar"` or `type="stepwise"` the default is "step"
- `...`: Additional arguments to lars

**Value**

Invisibly returns a list with components (which can be plotted using `plotCVlars`)

- `index`: As above
- `cv`: The CV curve at each value of index
- `cv.error`: The standard error of the CV curve
- `mode`: As above
diabetes

Author(s)
Trevor Hastie

References

Examples
data(diabetes)
attach(diabetes)
cv.lars(x2,y,trace=TRUE,max.steps=80)
detach(diabetes)

---

<table>
<thead>
<tr>
<th>diabetes</th>
<th>Blood and other measurements in diabetics</th>
</tr>
</thead>
</table>

Description
The diabetes data frame has 442 rows and 3 columns. These are the data used in the Efron et al "Least Angle Regression" paper.

Format
This data frame contains the following columns:

- x a matrix with 10 columns
- y a numeric vector
- x2 a matrix with 64 columns

Details
The x matrix has been standardized to have unit L2 norm in each column and zero mean. The matrix x2 consists of x plus certain interactions.

Source

References
Efron, Hastie, Johnstone and Tibshirani (2003) "Least Angle Regression" (with discussion) *Annals of Statistics*
Description

These are all variants of Lasso, and provide the entire sequence of coefficients and fits, starting from zero, to the least squares fit.

Usage

\[
\text{lars}(x, y, \text{type} = \text{c("lasso", "lar", "forward.stagewise", "stepwise")}, \text{trace} = \text{FALSE}, \text{normalize} = \text{TRUE}, \text{intercept} = \text{TRUE}, \text{Gram}, \text{eps} = \cdot \text{Machine$double.eps}, \text{max.steps}, \text{use.Gram})
\]

Arguments

- **x**: matrix of predictors
- **y**: response
- **type**: One of "lasso", "lar", "forward.stagewise" or "stepwise". The names can be abbreviated to any unique substring. Default is "lasso".
- **trace**: If TRUE, lars prints out its progress
- **normalize**: If TRUE, each variable is standardized to have unit L2 norm, otherwise it is left alone. Default is TRUE.
- **intercept**: if TRUE, an intercept is included in the model (and not penalized), otherwise no intercept is included. Default is TRUE.
- **Gram**: The X'X matrix; useful for repeated runs (bootstrap) where a large X'X stays the same.
- **eps**: An effective zero
- **max.steps**: Limit the number of steps taken; the default is 8 * \(\min(m, n-\text{intercept})\), with m the number of variables, and n the number of samples. For type="lar" or type="stepwise", the maximum number of steps is \(\min(m, n-\text{intercept})\). For type="lasso" and especially type="forward.stagewise", there can be many more terms, because although no more than \(\min(m, n-\text{intercept})\) variables can be active during any step, variables are frequently dropped and added as the algorithm proceeds. Although the default usually guarantees that the algorithm has proceeded to the saturated fit, users should check.
- **use.Gram**: When the number m of variables is very large, i.e. larger than N, then you may not want LARS to precompute the Gram matrix. Default is use.Gram=TRUE

Details

LARS is described in detail in Efron, Hastie, Johnstone and Tibshirani (2002). With the "lasso" option, it computes the complete lasso solution simultaneously for ALL values of the shrinkage parameter in the same computational cost as a least squares fit. A "stepwise" option has recently been added to LARS.
Value
A "lars" object is returned, for which print, plot, predict, coef and summary methods exist.

Author(s)
Brad Efron and Trevor Hastie

References

See Also
print, plot, summary and predict methods for lars, and cv.lars

Examples
```r
data(diabetes)
par(mfrow=c(2,2))
attach(diabetes)
object <- lars(x,y)
plot(object)
object2 <- lars(x,y,type="lar")
plot(object2)
object3 <- lars(x,y,type="for") # Can use abbreviations
plot(object3)
detach(diabetes)
```

Description
Produce a plot of a lars fit. The default is a complete coefficient path.

Usage
```r
## S3 method for class 'lars'
plot(x, xvar= c("norm", "df", "arc.length", "step"), breaks = TRUE, plottype = c("coefficients", "Cp"), omit.zeros = TRUE, eps = 1e-10, ...)
```
plot.lars

Arguments

- **x**: lars object
  - The type of x variable against which to plot. `xvar=norm` (default) plots against the L1 norm of the coefficient vector, as a fraction of the maximal L1 norm. `xvar=step` plots against the step number (which is essentially degrees of freedom for LAR; not for LASSO or Forward Stagewise). `xvar=arc.length` plots against the arc.length of the fitted vector; this is useful for a LAR object, because the L1 norm of its coefficient vector need not be monotone in the steps. `xvar=df` plots against the estimated df, which is the size of the active set at each step.

- **breaks**: If TRUE, then vertical lines are drawn at each break point in the piecewise linear coefficient paths

- **plottype**: Either `coefficients` (default) or `Cp`. The coefficient plot shows the path of each coefficient as a function of the norm fraction or Df. The Cp plot shows the Cp curve.

- **omit.zeros**: When the number of variables is much greater than the number of observations, many coefficients will never be nonzero; this logical (default TRUE) avoids plotting these zero coefficients

- **eps**: Definition of zero above, default is `1e-10`

- **...**: Additional arguments for generic plot. Can be used to set xlimits, change colors, line widths, etc

Details

The default plot uses the fraction of L1 norm as the xvar. For forward stagewise and LAR, coefficients can pass through zero during a step, which causes a change of slope of L1 norm vs arc-length. Since the coefficients are piecewise linear in arc-length between each step, this causes a change in slope of the coefficients.

Value

- NULL

Author(s)

- Trevor Hastie

References

Examples

```r
data(diabetes)
attach(diabetes)
object <- lars(x,y)
plot(object)
detach(diabetes)
```

### predict.lars

Make predictions or extract coefficients from a fitted lars model

**Description**

While lars() produces the entire path of solutions, predict.lars allows one to extract a prediction at a particular point along the path.

**Usage**

```r
## S3 method for class 'lars'
predict(object, newx, s, type = c("fit", "coefficients"), mode = c("step", "fraction", "norm", "lambda"), ...)
## S3 method for class 'lars'
coef(object, ...)
```

**Arguments**

- `object`: A fitted lars object
- `newx`: If type="fit", then newx should be the x values at which the fit is required. If type="coefficients", then newx can be omitted.
- `s`: a value, or vector of values, indexing the path. Its values depends on the mode= argument. By default (mode="step"), s should take on values between 0 and p (e.g., a step of 1.3 means .3 of the way between step 1 and 2.)
- `type`: If type="fit", predict returns the fitted values. If type="coefficients", predict returns the coefficients. Abbreviations allowed.
- `mode`: Mode="step" means the s= argument indexes the lars step number, and the coefficients will be returned corresponding to the values corresponding to step s. If mode="fraction", then s should be a number between 0 and 1, and it refers to the ratio of the L1 norm of the coefficient vector, relative to the norm at the full LS solution. Mode="norm" means s refers to the L1 norm of the coefficient vector. Mode="lambda" uses the lasso regularization parameter for s; for other models it is the maximal correlation (does not make sense for lars/stepwise models). Abbreviations allowed.
- `...`: Any arguments for `predict.lars` should work for `coef.lars`
Details

LARS is described in detail in Efron, Hastie, Johnstone and Tibshirani (2002). With the "lasso" option, it computes the complete lasso solution simultaneously for ALL values of the shrinkage parameter in the same computational cost as a least squares fit.

Value

Either a vector/matrix of fitted values, or a vector/matrix of coefficients.

Author(s)

Trevor Hastie

References


See Also

print, plot, lars, cv.lars

Examples

data(diabetes)
attach(diabetes)
object <- lars(x,y,type="lasso")
### make predictions at the values in x, at each of the
### steps produced in object
fits <- predict.lars(object, x, type="fit")
### extract the coefficient vector with L1 norm=4.1
coef4.1 <- coef(object, s=4.1, mode="norm") # or
coeff4.1 <- predict(object, s=4.1, type="coef", mode="norm")
detach(diabetes)

summary.lars

Summary method for lars objects

Description

Produce an anova-type summary for a lars object.

Usage

## S3 method for class 'lars'
summary(object, sigma2=NULL, ...)

summary.lars

Summary method for lars objects

Description

Produce an anova-type summary for a lars object.

Usage

## S3 method for class 'lars'
summary(object, sigma2=NULL, ...)
Arguments

object  lars object
sigma2  optional variance measure (for p>n)
...  Additional arguments for summary generic

Details

An anova summary is produced, with Df, RSS and Cp for each step. Df is tricky for some models, such as forward stagewise and stepwise, and is not likely to be accurate. When p>n, the user is responsible for supplying sigma2.

Value

An anova object is returned, with rownames the step number, and with components:

Df  Estimated degree of freedom
Rss  The Residual sum of Squares
Cp  The Cp statistic

Author(s)

Brad Efron and Trevor Hastie

References


See Also

lars, and print, plot, and predict methods for lars, and cv.lars

Examples

data(diabetes)
attach(diabetes)
object <- lars(x,y)
summary(object)
detach(diabetes)
Index

*Topic datasets
diabetes, 3

*Topic hplot
plot.lars, 5

*Topic methods
plot.lars, 5
predict.lars, 7

*Topic regression
cv.lars, 2
lars, 4
predict.lars, 7
summary.lars, 8

coef.lars (predict.lars), 7
cv.lars, 2
diabetes, 3

lars, 4

plot.lars, 5
predict.lars, 7

summary.lars, 8