Package ‘relaxo’

February 20, 2015

Type Package
Title Relaxed Lasso
Version 0.1-2
Date 2012-06-01
Author Nicolai Meinshausen
Maintainer Nicolai Meinshausen <meinshausen@stats.ox.ac.uk>
Depends lars
Imports graphics, utils, stats
Description Relaxed Lasso is a generalisation of the Lasso shrinkage technique for linear regression. Both variable selection and parameter estimation is achieved by regular Lasso, yet both steps do not necessarily use the same penalty parameter. The results include all standard Lasso solutions but allow often for sparser models while having similar or even slightly better predictive performance if many predictor variables are present. The package depends on the LARS package.
License GPL
URL http://www.stats.ox.ac.uk/~meinshau/
Repository CRAN
Date/Publication 2012-06-01 16:06:47
NeedsCompilation no

R topics documented:
cvrelaxo ................................................................. 2
plot.relaxo ............................................................ 3
predict.relaxo ....................................................... 4
relaxo ................................................................. 5

Index 7
Cross validation for "Relaxed Lasso"

Description
Compute the "Relaxed Lasso" solution with minimal cross-validated L2-loss.

Usage
```r
cvrelaxo(X, y, k = 5, phi = seq(0, 1, length = 10), max.steps = min(2 * length(y), 2 * ncol(X)), fast = TRUE, keep.data = TRUE, warn = TRUE)
```

Arguments
- `X` as in function `relaxo`
- `y` as in function `relaxo`
- `k` Number of folds. Defaults to 5.
- `phi` as in function `relaxo`
- `max.steps` as in function `relaxo`
- `fast` as in function `relaxo`
- `keep.data` as in function `relaxo`
- `warn` as in function `relaxo`

Details
The plot method is not useful for result of `cvrelaxo` (as no path of solutions exists).

Value
An object of class `relaxo`, for which print and predict methods exist.

Author(s)
Nicolai Meinshausen <nicolai@stat.berkeley.edu>

References

See Also
See also `relaxo` for computation of the entire solution path.
Examples

```r
data(diabetes)

## Center and scale variables
  x <- scale(diabetes$x)
  y <- scale(diabetes$y)

## Compute "Relaxed Lasso" solution and plot results
  object <- relaxo(x, y)
  plot(object)

## Compute cross-validated solution with optimal predictive performance and print relaxation parameter phi and penalty parameter lambda of the found solution
  cvobject <- cvrelaxo(x, y)
  print(cvobject$phi)
  print(cvobject$lambda)

## Compute fitted values and plot them versus actual values
  fitted.values <- predict(cvobject)
  plot(fitted.values, y)
  abline(c(0, 1))
```

---

**plot.relaxo**  
*Plot of a Relaxed Lasso Object*

**Description**

Plot of Relaxed Lasso solutions for various relaxation parameters

**Usage**

```r
## S3 method for class 'relaxo'
plot(x, type = "l", lty = 1, main = NULL, xlab = "|beta|/max|beta| (phi=1)", ylab = expression("coefficients * beta[j]")), plotphi = unique(x$phi), NNN)
```

**Arguments**

- `x` A object of class relaxo
- `type` Plot type
- `lty` Line style
- `main` Title of plot
- `xlab` Label of x-axis
- `ylab` Label of y-axis
- `plotphi` Which values of the relaxation parameter phi should be plotted?
- `...` Arguments passed on to the plot function
Value

No return value

Author(s)

Nicolai Meinshausen <nicolai@stat.berkeley.edu>

References


predict.relaxo  

predict method for class relaxo

Description

Prediction with Relaxed Lasso

Usage

## S3 method for class 'relaxo'
predict(object, newX = NULL, lambda = NULL, phi = NULL,...)

Arguments

- object An object of class relaxo
- newX A data frame or matrix containing new data. If not given, the prediction for the original training data is returned.
- lambda The penalty parameter for variable selection.
- phi The relaxation parameter in the interval [0,1].
- ... other arguments (currently ignored).

Value

A numerical vector, containing the predictions for the new data points (or the fitted values if newX=NULL).

Author(s)

Nicolai Meinshausen <nicolai@stat.berkeley.edu>

References

relaxo

**Relaxed Lasso (relaxo)**

**Description**

Computes all "Relaxed Lasso" solutions.

**Usage**

```
relaxo(X, Y, phi = seq(0, 1, length = 4), max.steps = min(2*length(Y), 2 * ncol(X)), fast = TRUE, keep.data = TRUE, warn = TRUE)
```

**Arguments**

- **X**: n x p-dimensional matrix or data frame containing the predictor variables; columns are supposed to be scaled and centered.
- **Y**: n-dimensional numerical response vector; supposed to be centered to mean 0.
- **phi**: Relaxation parameter in [0,1]. A value of phi=1 corresponds to the regular Lasso solutions; a value of phi=0 computes the OLS estimates on the set of variables selected by the Lasso.
- **max.steps**: Maximal number of steps the LARS algorithm is run.
- **fast**: Should the estimates be computed in approx. the same time as the LARS algorithm? If fast=TRUE, minor deviations from the original Relaxed Lasso solution can occur.
- **keep.data**: Should the data be kept for later usage e.g. (when computing predicted values for the training data)?
- **warn**: If TRUE, warnings are given if the predictor variables X are not centered and scaled or if the reponse variable is not centered? 

**Value**

An object of class relaxo, for which plot and predict methods are available.

**Author(s)**

Nicolai Meinshausen <nicolai@stat.berkeley.edu>

**References**


**See Also**

See also `cvrelaxo` for computation of the cross-validated solution with optimal predictive performance.
Examples

```r
data(diabetes)

## Center and scale variables
x <- scale(diabetes$x)
y <- as.numeric(scale(diabetes$y))

## Compute "Relaxed Lasso" solution and plot results
object <- relaxo(x, y)
plot(object)

## Compute cross-validated solution with optimal
## predictive performance and print relaxation parameter phi and
## penalty parameter lambda of the found solution
cvobject <- cvrelaxo(x, y)
print(cvobject$phi)
print(cvobject$lambda)

## Compute fitted values and plot them versus actual values
fitted.values <- predict(cvobject)
plot(fitted.values, y)
abline(c(0, 1))
```
Index

*Topic models
  cvrelaxo, 2
  plot.relaxo, 3
  predict.relaxo, 4
  relaxo, 5

  cvrelaxo, 2, 5
  plot.relaxo, 3
  predict.relaxo, 4
  relaxo, 2, 5