Package ‘covTest’
February 19, 2015

Title Computes covariance test for adaptive linear modelling
Version 1.02
Depends lars, glmnet, glmpath (>= 0.97), MASS
Author Richard Lockhart, Jon Taylor, Ryan Tibshirani, Rob Tibshirani
Maintainer Rob Tibshirani <tibs@stanford.edu>
Description This package computes covariance test for the lasso.
License GPL-2
NeedsCompilation no
Repository CRAN
Date/Publication 2013-08-14 07:28:16

R topics documented:

  covTest-package .................................................. 1
  covTest .......................................................... 3
  lars.en ........................................................... 4
  lars glm .......................................................... 6
  predict.lars.en ................................................... 7
  predict.lars.glm .................................................. 8

Index

covTest-package  Compute the covariance test for adaptive linear modelling

Description

Compute the covariance test significance testing in adaptive linear modelling. Can be used with
LARS (lasso) for linear models, elastic net, binomial and Cox survival model. This package should
be considered EXPERIMENTAL. The background paper is not yet published and rigorous theory
does not yet exist for the logistic and Cox models.

Details
Very simple to use. Takes output from one of `lars`, `lars.en`, `lars.glm` and compute covariance test and p-values. Requires `lars` and `glmnet` packages. `lars.en` and `lars.glm` are included in this package. Functions are: `covTest`, `lars.en`, `lars.glm`, `predict.lars.en`, `predict.lars.glm`

Author(s)

Rob Tibshirani tibs@stanford.edu

References

A significance test for the lasso (2013). Lockhart, R., Taylor, J., Tibshirani (Ryan) and Tibshirani (Robert)

See Also

covTest, lars.glm, lars.en

Examples

```r
x = matrix(rnorm(100*10), ncol=10)
x = scale(x, TRUE, TRUE)/sqrt(99)

# Gaussian
beta = c(4, rep(0, 9))
y = x%*%beta + .4*runif(100)
a = lars(x, y)
covTest(a, x, y)

# Elastic net
a = lars.en(x, y, lambda2=1)
covTest(a, x, y)

# Logistic
y = 1*(y>0)
a = lars.glm(x, y, family="binomial")
covTest(a, x, y)

# Cox model
# y = 6*x[,2] + rnorm(100) + 10
# status = sample(c(0,1), size=length(y), replace=TRUE)
# a = lars.glm(x, y, status=status, family="cox")
```
covTest

#covTest(a,x,y,status=status)

covTest

Compute the covariance test for adaptive linear modelling

Description

This function computes covariance test for inference in adaptive linear modelling, for lasso (least angle regression) in the Gaussian case, binomial/logistic and Cox proportional hazards survival models. This package should be considered EXPERIMENTAL. The background paper is not yet published and rigorous theory does not yet exist for the logistic and Cox models. We have currently disabled the Cox option, as it is not yet reliable.

Usage

`covTest(fitobj, x, y, sigma.est = "full", status = NULL, maxp=\min(nrow(x),ncol(x)))`

Arguments

- `fitobj`: Result of a call to `lars` or `lars.en` or `lars.glm`
- `x`: N by p matrix of predictors
- `y`: N-vector of outcome values
- `sigma.est`: Estimate of error standard deviation. If a numerical value, that value if used. If "full" the (square root) of the mean squared residual from the full model is used.
- `status`: Optional N-vector of censoring indicators for Cox Proportional hazards model. 1=failed; 0=censored.
- `maxp`: Optional limit for number of steps to be analyzed.

Details

This function computes covariance test for inference in adaptive linear modelling, for lasso (least angle regression) in the Gaussian case, binomial/logistic and Cox proportional hazards survival models. It estimates p-values for each predictor entered, that account for the adaptive nature of the fitting.

Value

- `results`: Table of covariance test values and p-values, for each predictor entered
- `sigma`: Estimate of sigma used
- `null.dist`: Null distribution used to compute p-values

Note

This function requires the `lars` R library (for the Gaussian case), and the `glmpath` function for the logistic and Cox model.
Author(s)

Rob Tibshirani

References

A significance test for the lasso (2013). Lockhart, R., Taylor, J., Tibshirani (Ryan) and Tibshirani (Robert)

See Also

lars, lars.en, lars.glm

Examples

```r
set.seed(1234)
x = matrix(rnorm(100*10), ncol=10)
x = scale(x, TRUE, TRUE)/sqrt(99)
beta = c(4, rep(0,9))
y = x%*%beta + 4*rnorm(100)

# Gaussian
a = lars(x, y)
covTest(a, x, y)

# EN
a = lars.en(x, y, lambda2=1)
covTest(a, x, y)

# logistic
y = 1*(y>0)
a = lars.glm(x, y, family="binomial")
covTest(a, x, y)

# Cox model
y = 6*x[,2]+rnorm(100)+10
# status = sample(c(0,1), size = length(y), replace = TRUE)
a = lars.glm(x, y, status = status, family = "cox")
covTest(a, x, y, status = status)
```

lars.en

Function to fit least angle regression path of solution for the elastic net.

Description

Function to fit least angle regression path of solution for the elastic net.
Usage

lars.en(x, y, lambda2, normalize=TRUE)

Arguments

x  N by p matrix of predictors
y  N-vector of outcome values
lambda2  Value of L2 penalty parameter
normalize  Should columns of x be standardized?

Details

This function estimates the least angle regression path of solution for L1-penalized (lasso) logistic regression and the Cox proportional hazards model, using the R functions enpath and coxpath. These latter functions use the predictor-corrector strategy devised by Park and Hastie (2007).

Value

beta  Matrix whose rows of contain the estimated coefficients for each lambda value
larsobj  Result of call to lars on augmented data
mx  Column means of x
sdx  Column standard deviations of x
normalize  Value of normalize argument in call to lars.en
lambda  Values of lambda used
lambda2  Value of lambda2 used
act  Actions (predictor added) at each step
maxp  Maximum number of predictors entered
call  Call to lars.en

Author(s)

Rob Tibshirani

References

Zou, H. and Hastie, Trevor (2005) Regularization and Variable Selection via the Elastic Net. JRSSB 301-320,


See Also

predict.lars, covTest
Examples

```r
set.seed(432)
x = matrix(rnorm(100*10), ncol=10)
x = scale(x, TRUE, TRUE)/sqrt(99)
beta = c(3, rep(0, 9))
y = x %*% beta + 4 * rnorm(100)
a = lars.en(x, y, lambda2 = .5)
```

---

**lars.glm**

Function to fit least angle regression path of solution for L1-penalized (lasso) logistic regression and the Cox proportional hazards model.

---

**Description**

Function to fit least angle regression path of solution for L1-penalized (lasso) logistic regression and the Cox proportional hazards model.

**Usage**

```r
lars.glm(x, y, status = NULL, family = c("binomial", "cox"), standardize = TRUE, frac.arclength = .1)
```

**Arguments**

- **x**: N by p matrix of predictors
- **y**: N-vector of outcome values
- **status**: Optional N-vector of censoring indicators for Cox Proportioanl hazards model. 1=failed; 0=censored.
- **family**: "binomial" or "cox"
- **standardize**: Should predictor be standardized? Default TRUE
- **frac.arclength**: Step length parameter for glmpath. See help file for glmpath for details

**Details**

This function estimates the least angle regression path of solution for L1-penalized (lasso) logistic regression and the Cox proportional hazards model, using the R functions glmpath and coxpath written by Park and Hastie. These latter functions use the predictor-corrector strategy devised by Park and Hastie (2007). An additional L2 penalty can be added for stability.

**Value**

- **beta**: Matrix of estimated coefficients, with LAR steps in the rows.
- **a0**: Estimate of intercept
- **lambda0**: Raw values of lambda used
- **lambda**: Values of lambda multiplied by sdx, the standard deviation of each predictor
predict.lars.en

lambda2  Value of lambda2 (L2 penalty parameter)
act      Actions (predictor added) at each step
maxp     Maximum number of predictors entered
family   family used- "binomial" or "cox"
call     Call to lars.glm
pathobj  Result of call to glmpath or coxpath

Author(s)
Rob Tibshirani

References

See Also
covTest, predict.glm.Rd

Examples
x=matrix(rnorm(100*10),ncol=10)
x=scale(x,TRUE,TRUE)/sqrt(99)

#logistic
y=5*x[,2]+rnorm(100)
y=1*(y>0)
a=lars.glm(x,y,family="binomial")

# Cox model
y=6*x[,2]+rnorm(100)+10
status=sample(c(0,1),size=length(y),replace=TRUE)
a=lars.glm(x,y,status=status,family="cox")

predict.lars.en  Function to make predictions from lars.en fit

Description
Function to make predictions from lars.en fit

Usage
## S3 method for class 'lars.en'
predict(object, x, lambda, ...)

Arguments

- **object**: Result of call to lars.en
- **x**: N by p matrix of predictors
- **lambda**: Value of L1-regularization parameter at which predictions are desired
- **...**: additional arguments (not used)

Details

Makes predictions from an object returned by lars.en

Value

Vector of predicted values.

Author(s)

Rob Tibshirani

References


See Also

lars.en, covTest

Examples

```r
x <- matrix(rnorm(100*10), ncol = 10)
x <- scale(x, TRUE, TRUE)/sqrt(99)
y <- 4*x[,2] + rnorm(100)
a <- lars.en(x, y, lambda = 1)
yhat <- predict.lars.en(a, x, .5)
```

---

**predict.lars.glm**

Function to make predictions from lars.glm fit

Description

Function to make predictions from lars.glm fit

Usage

```r
## S3 method for class 'lars.glm'
predict(object, x, lambda, time = NULL, status = NULL, ...)
```
predict.lars.glm

Arguments

object  
Result of call to lars.glm

x  
N by p matrix of predictors

lambda  
Value of L1-regularization parameter at which predictions are desired

time  
Optional N-vector of survival times, required for Cox Proportional hazards model.

status  
Optional N-vector of censoring indicators, required for Cox Proportional hazards model. 1=failed; 0=censored.

...  
additional arguments (not used)

Value

Vector of predicted values, on the linear predictor scale.

Author(s)

Rob Tibshirani

References


See Also

lars.glm, covTest

Examples

# logistic
x=matrix(rnorm(100*10),ncol=10)
x=scale(x,TRUE,TRUE)/sqrt(99)
y=4*x[,2]+rnorm(100)
y=1*(y>0)
a=lars.glm(x,y,family="binomial")
yhat=predict.lars.glm(a,x,family="binomial")

# Cox model
#y=6*x[,2]+rnorm(100)+10
#status=sample(c(0,1),size=length(y),replace=TRUE)
a=lars.glm(x,y,status=status,family="cox")
yhat=predict.lars.glm(a,x,family="cox")
Index

*Topic **Cox model**
  - lars.glm, 6
  - predict.lars.glm, 8

*Topic **LARS**
  - covTest-package, 1
  - lars.en, 4
  - lars.glm, 6
  - predict.lars.en, 7
  - predict.lars.glm, 8

*Topic **adaptive fitting**
  - covTest, 3

*Topic **elastic net**
  - lars.en, 4
  - predict.lars.en, 7

*Topic **lasso**
  - covTest, 3
  - lars.en, 4
  - lars.glm, 6
  - predict.lars.en, 7
  - predict.lars.glm, 8

*Topic **least angle regression**
  - covTest, 3

*Topic **logistic**
  - lars.glm, 6
  - predict.lars.glm, 8

  covTest, 3
  covTest-package, 1
  covTest-packarghe (covTest-package), 1

  lars.en, 4
  lars.glm, 6

  predict.lars.en, 7
  predict.lars.glm, 8