Package ‘UnifQuantReg’

May 13, 2014

Type Package

Title Uniformly Adaptive-LASSO Quantile Regression

Version 1.0

Date 2014-05-12

Author Limin Peng, Jinfeng Xu and Qi Zheng

Maintainer Qi Zheng <qi.zheng@emory.edu>

Description Uniformly adaptive-LASSO quantile regression for model selection

License GPL (>= 2)

Dependes lars, quantreg, MASS, R(>=3.0)

R topics documented:

`UnifQuantReg-package`

A package conducting uniformly adaptive-LASSO quantile regression for model selection

---

Description

The packages provides

(1) Adaptive-LASSO quantile regression at a single quantile level: rqalasso();

(2) Uniformly Adaptive-LASSO quantile regression at at multiple specified quantile levels: rqalasso();

(3) Uniformly Adaptive-LASSO quantile regression at a specified compact set: urqalasso().

Details

Package:	UnifQuantReg
Type:	Package
Version:	1.0
Date:	2014-05-07
License:	GPL (>= 2)

Author(s)

Limin Peng, Jinfeng Xu and Qi Zheng

Maintainer: Qi Zheng <qi.zheng@emory.edu>

`lsalasso` *Adaptive-LASSO linear regression with BIC tuning parameter*

Description

Adaptive-LASSO from Zou (2006) + BIC tuning parameter selector from Wang et al (2007)

Usage

```
lsalasso(x, y)
```

Arguments

`x`	the predictor matrix
`y`	the response vector

Details

It returns an object including fit, coeff, lars objects and so on.

Value

The function returns a list containing

`fit`	The fitted y's
`coeff`	The estimated regression coefficients
`st`	The fitted model size, i.e. the number of nonzero coefficients
`mse`	The mse of the fitted model
`object`	lars object
`bic2`	The BIC values of steps in the lars object
`step.bic2`	The step with the minimum BIC

Author(s)

http://www4.stat.ncsu.edu/~boos/var.select/lasso.adaptive.html

References

Zou (2006, JASA) Adaptive LASSO and its oracle property

Wang et al (2007, Biometrika) Tuning parameter selectors for the smoothly clipped absolute deviation method

Examples

```
library(lars)
library(MASS)
p=8;                                          # model dim
n=100;                                        #sample size
tol=1e-6;
######## covariance matrix
Sigma=matrix(0,p,p);
pho=0.5;
J=seq(1,p,1);
for (i in 1:p){Sigma[i,]=pho^(abs(i-J))};


######## covariate matrix
Z=mvrnorm(n=n , rep(0,p), Sigma,  empirical = FALSE); # Generate covariates
Z=pmax(pmin(Z, 3), -3);                               # Truncate covariates
X=cbind(rep(1,n),Z);                                  # Add intercept

######## regression coefficients
beta=rep(0,p+1);
beta[2]=3;
beta[3]=1.5;
beta[6]=2;

######## errors and response variable
epsilon=rnorm(n,0,2);                         # sqrt(2) * standard normal quantile
Y=X%*%beta+epsilon;
lsalasso(Y,Z);
```

---

rqalasso — *Adaptive_LASSO model selection and parameter estimation based on Quantile regression*

Description

This is a function to conduct model selection and parameter estimation using adaptive-LASSO quantile regression with BIC tuning parameter selection

Usage

```
rqalasso(y, x, tau=0.5, len=3*sqrt(nrow(x)))
```

Arguments

`y`	the response vector
`x`	the predictor matrix
`tau`	the quantile level, the default is 0.5. (1) it can be a single quantile, (2) a vector of quantile levels
`len`	the grid size for the refined BIC search, the default is 3*sqrt(n)

Details

1. If tau is a single quantile, then an adaptive-LASSO quantile regression will be implemented. If tau is a vector of quantile, then the uniformly adaptive-LASSO quantile regression (Peng et al, 2014) will be implmented at the specified quantile levels.

2. Two-fold BIC search are included: (Step 1): the initial search is conducted on the grid { 1/n, 2/n, 4/n, ... } until the estimated coeffs are small enough. (Step 2): the next search will conducted around two grid points with the smallest BIC from (Step 1). It can be seen as a refined BIC search.

Value

The function returns the estimated coefficients at specified quantile levels.

Author(s)

Limin Peng, Jinfeng Xu and Qi Zheng

References

Peng et al (2014, Statistics and Computing) Shrinkage estimation of varying covariate effects based on quantile regression

Examples

```
library(MASS)
p=8;                                            # model dim
n=100;                                          # sample size
tol=1e-6;
######## covariance matrix
Sigma=matrix(0,p,p);
pho=0.5;
J=seq(1,p,1);
for (i in 1:p){Sigma[i,]=pho^(abs(i-J))};


######## covariate matrix
Z=mvrnorm(n=n , rep(0,p), Sigma,  empirical = FALSE); # Generate covariates
Z=pmax(pmin(Z, 3), -3);                               # Truncate covariates
X=cbind(rep(1,n),Z);                                  # Add intercept

######## regression coefficients
beta=rep(0,p+1);
beta[2]=3;
beta[3]=1.5;
beta[6]=2;

######## errors and response variable
```

```
epsilon=rnorm(n,0,2);                          # sqrt(2) * standard normal quantile
Y=X%*%beta+epsilon;



rqalasso(Y,Z,0.5,100);
rqalasso(Y,Z,len=100);
J=c(0.25,0.5,0.75);
rqalasso(Y,Z, J, 100);
```

`urqalasso` *Adaptive-LASSO with uniform Weights*

Description

The function will provide a model containing all relevant variables and estimate their coefficients across quantile leves specified.

Usage

```
urqalasso(y, x, tau = c(0.1, 0.9), len=3*sqrt(nrow(x)))
```

Arguments

`y`	the response vector
`x`	the predictor matrix
`tau`	a compact set of quantile levels. It should be of the form (a1, b1, a2 ,b2, a3, b3 ....)
`len`	the length of BIC grid

Details

1. The tau vector must have even elements and the elements are in an increasing order. Otherwise, the the function will stop and report an error. The compact set will be treated as Union([a1, b1], [a2, b2] . . .).

2. Two-fold BIC search are included: (Step 1): the initial search is conducted on the grid { 1/n, 2/n, 4/n, . . . } until the estimated coeffs are small enough. (Step 2): the next search will conducted around two grid points with the smallest BIC from (Step 1). It can be seen as a refined BIC search.

Value

The function returns a list containing

`tau`	all quantile levels in the given compact set which have distinc solutions
`coeff`	the corresponding solutions

Note

1. It may take some time for the function to compute all possible solutions at a given compact set. If only few quantile levels are of interest, please use the function rqalasso().

2. If the given compact set is very small, the function will return an error, " Error, please specify a larger set". User probably may consider using rqalasso() or enlarge the compact set.

Author(s)

Limin Peng, Jinfeng Xu and Qi Zheng

References

Peng et al (2014, Statistics and Computing) Shrinkage estimation of varying covariate effects based on quantile regression

Examples

```
library(SMPracticals)
data(pollution);
Y=pollution[,16];
X=pollution[,-16];
urqalasso(Y,X);
```

Index