Package ‘CVTuningCov’

February 19, 2015

Type Package

Title Regularized Estimators of Covariance Matrices with CV Tuning

Version 1.0

Date 2014-07-31

Author Binhuan Wang

Maintainer Binhuan Wang <binhuan.wang@nyumc.org>

Description This is a package for selecting tuning parameters based on cross-validation (CV) in regularized estimators of large covariance matrices. Four regularized methods are implemented: banding, tapering, hard-thresholding and soft-thresholding. Two types of matrix norms are applied: Frobenius norm and operator norm. Two types of CV are considered: K-fold CV and random CV. Usually K-fold CV use K-1 folds to train a model and the rest one fold to validate the model. The reverse version trains a model with 1 fold and validates with the rest with K-1 folds. Random CV randomly splits the data set to two parts, a training set and a validation set with user-specified sizes.

Suggests MASS

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2014-08-15 06:56:38

R topics documented:

`CVTuningCov-package` *Select Tuning Parameters based on CV in Regularized Estimators of Covariance Matrices*

Description

This is a package for selecting tuning parameters based on cross-validation (CV) in regularized estimators of large covariance matrices. Four regularized methods are implemented: banding, tapering, hard-thresholding and soft-thresholding. Two types of matrix norms are applied: Frobenius norm and operator norm. Two types of CV are considered: K-fold CV and random CV. Usually K-fold CV use K-1 folds to train a model and the rest one fold to validate the model. The reverse version trains a model with 1 fold and validates with the rest with K-1 folds. Random CV randomly splits the data set to two parts, a training set and a validation set with user-specified sizes.

Details

Package:	CVTuningCov
Type:	Package
Version:	1.0
Date:	2014-07-31
License:	GPL-2

Author(s)

Binhuan Wang

Maintainer: Binhuan Wang <binhuan.wang@nyumc.org>

References

Fang, Y., Wang, B. and Feng, Y. (2013). Tuning parameter selection in regularized estimations of large covariance matrices. Available at: http://arxiv.org/abs/1308.3416.

Examples

```
library(MASS);
n <- 50;
p <- 50;
fold <- 3;
k.grid <- seq(0,2*(p-1),by=1);
Sigma <- AR1(p, rho=0.6);
X <- mvrnorm(n,rep(0,p),Sigma);
CV.F.fit <- regular.CV(X,k.grid, method='Tapering',fold=fold,norm='F');
CV.F.fit$CV.k;
```

AR1 *Covariance Matrix with AR(1) Structure*

Description

Generate Covariance Matrix with an Autoregression (1) Structrue

Usage

```
AR1(p,rho=0.5)
```

Arguments

p	the dimension of a covariance matrix.
rho	the default value is 0.5.

Value

a `p*p` matrix.

Author(s)

Binhuan Wang

Examples

```
p <- 5;
Sigma <- AR1(p, rho=0.9);
Sigma;
```

banding *A Banding Operator on A Matrix*

Description

Generate a banding operator with given dimention and tuning parameter. Multiplying it on a covariance matrix by componentwise product can provide a regularized estimator with the banding method.

Usage

```
banding(p, k = 1)
```

Arguments

p	the dimension of a covariance matrix.
k	the default value is 1.

Value

a p*p matrix.

Author(s)

Binhuan Wang

References

Bickel, P and Levina, E, Regularized estimation of large covariance matrices, Annals of Statistics, 36, 199-227 (2008).

Examples

```
p <- 5;
W <- banding(p,k=2) ;
W;
```

F.norm2 *The Squared Frobenius Norm*

Description

Calculate the squared Frobenius norm of a matrix

Usage

```
F.norm2(A)
```

Arguments

A	a matrix

Value

a scalar of the squared Frobenius norm.

Author(s)

Binhuan Wang

Examples

```
A<-matrix(1:9,3,3);
F.norm2(A);
```

hard.thresholding *Hard-thresholding Operator on A Covariance Matrix*

Description

Apply hard-thresholding operator on an input covariance matrix with a tuning parameter.

Usage

```
hard.thresholding(Sigma, c = 0.5)
```

Arguments

Sigma	a covariance matrix with dimension p*p.
c	the default value is 0.5.

Value

a p*p covariance matrix after hard-thresholding operation.

Author(s)

Binhuan Wang

References

Bickel, P and Levina, E, Covariance regularization by thresholding, Annals of Statistics, 36, 2577-2604 (2008).

Examples

```
p <- 5;
Sigma <- AR1(p, rho=0.6);
hard.Sigma<-hard.thresholding(Sigma,c=0.5);
hard.Sigma;
```

L2.norm2 *The Squared Operator Norm*

Description

Calculate the squared operator norm of a matrix

Usage

```
L2.norm2(A)
```

Arguments

A	a matrix

Value

a scalar of the squared operator norm.

Author(s)

Binhuan Wang

Examples

```
A<-matrix(1:9,3,3);
L2.norm2(A);
```

---

random.CV *Select Tuning Parameter for Regularized Covariance Matrix by Random CV*

Description

Apply a random cross-validation (CV) to select tuning parameters for regualrized covariance matrix with banding, tapering, soft-thresholding or hard-thresholding method under the Frobenius norm or the operator norm. The random CV randomly splits the data set to two parts, a training set and a validation set with user-specifed sizes, and repeats the process for multiple times.

Usage

```
random.CV(X, k.grid = 0.5, method = "Tapering", test.size = 5, norm = "F",
boot.num = 50, seed = 10323)
```

Arguments

X	input data matrix with dimension n*p. n indicates the sample size and p indicates the dimension of the corresponding random vector.
k.grid	the default value is 0.5.
method	the regularized method, which can be "Banding", "Tapering", "HardThresholding" or "SoftThresholding". the default value is "Tapering".
test.size	the size of the validation set, which should be < n.
norm	the norms which can be used to measure the estimation accuracy, which can be the Frobenius norm "F" or the operator norm "L2".
boot.num	the number of random split. The default value is 50.
seed	the default value is 10323.

Value

A list including elements:

CV.k	the optimal tuning parameter selected by the random CV.
k.grid	the vector of tuning parameters
CV.pre.error	a vector denoting predicting errors by random CV at each element of tuning parameters based on the selected norm.

Author(s)

Binhuan Wang

Examples

```
library(MASS);
n <- 50;
p <- 50;
fold <- 3;
k.grid <- seq(0,2*(p-1),by=1);
Sigma <- AR1(p, rho=0.6);
X <- mvrnorm(n,rep(0,p),Sigma);
CV.F.fit <- random.CV(X,k.grid, method='Tapering',test.size = 10,norm='F');
CV.F.fit$CV.k;
```

---

regular.CV — *Select Tuning Parameter for Regularized Covariance Matrix by K-fold CV*

Description

Apply K-fold cross-validation (CV) to select tuning parameters for regualrized covariance matrix with banding, tapering, soft-thresholding or hard-thresholding method under the Frobenius norm or the operator norm. Two versions of K-fold CV are applied: 1) the regular one, K-1 folds are used to train and 1 fold is used to validate; 2) the reverse one, 1 fold is used to train and K-1 folds are used to validate.

Usage

```
regular.CV(X, k.grid = 0.5, method = "Tapering", fold = 5, norm = "F", seed=10323)
```

Arguments

X	input data matrix with dimension n*p. n indicates the sample size and p indicates the dimension of the corresponding random vector.
k.grid	the default value is 0.5.
method	the regularized method, which can be "Banding", "Tapering", "HardThresholding" or "SoftThresholding". the default value is "Tapering".

fold	the number of folds in K-fold CV. integers only. the default number is 5.
norm	the norms which can be used to measure the estimation accuracy, which can be the Frobenius norm "F" or the operator norm "L2".
seed	the default value is 10323.

Value

A list including elements:

CV.k	a 2-dimensional vector denoting the optimal tuning parameters selected by K-fold CV with its first component as the one from regular version and its second component as the one from the reverse version.
k.grid	the vector of tuning parameters
CV.pre.error	a matrix with two columns denoting predicting errors by K-fold CV at each element of tuning parameters based on the selected norm with its first column demoting the values from regular version and the second column denoting the values from the reverse version. The number of rows equals the length of the vector for tuning parameters.

Author(s)

Binhuan Wang

Examples

```
library(MASS);
n <- 50;
p <- 50;
fold <- 3;
k.grid <- seq(0,2*(p-1),by=1);
Sigma <- AR1(p, rho=0.6);
X <- mvrnorm(n,rep(0,p),Sigma);
CV.F.fit <- regular.CV(X,k.grid, method='Tapering',fold=fold,norm='F');
CV.F.fit$CV.k;
```

---

soft.thresholding *Soft-thresholding Operator on A Covariance Matrix*

Description

Apply soft-thresholding operator on an input covariance matrix with a tuning parameter.

Usage

```
soft.thresholding(Sigma, c = 0.5)
```

Arguments

`Sigma`	a covariance matrix with dimension `p*p`.
`c`	the default value is 0.5.

Value

a `p*p` covariance matrix after soft-thresholding operation.

Author(s)

Binhuan Wang

References

Rothman, A, Levina, E and Zhu, J, A new approach to Cholesky-based covariance regularization in high dimensions, Biometrika, 97, 539-550 (2010).

Examples

```
p <- 5;
Sigma <- AR1(p, rho=0.6);
soft.Sigma<-soft.thresholding(Sigma,c=0.5);
soft.Sigma;
```

`tapering` *A Tapering Operator on A Matrix*

Description

Generate a tapering operator with given dimention and tuning parameter. Multiplying it on a covariance matrix by componentwise product can provide a regularized estimator with the tapering method.

Usage

```
tapering(p, k = 1)
```

Arguments

`p`	the dimension of a covariance matrix.
`k`	the tuning parameter of the tapering method. The default value is 1.

Value

a `p*p` matrix.

Author(s)

Binhuan Wang

References

Cai, T, Zhang, CH and Zhou, H, Optimal rates of convergence for covariance matrix estimation, Annals of Statistics, 38, 2118-2144 (2010).

Examples

```
p <- 5;
W <- tapering(p,k=2) ;
W;
```

Index