Package ‘ddpca’

October 13, 2022

Type Package

Title Diagonally Dominant Principal Component Analysis

Version 1.1

Date 2019-09-14

Author Tracy Ke [aut],
Lingzhou Xue [aut],
Fan Yang [aut, cre]

Maintainer Fan Yang <fyang1@uchicago.edu>

Description

Efficient procedures for fitting the DD-PCA (Ke et al., 2019, <arXiv:1906.00051>) by decomposing a large covariance matrix into a low-rank matrix plus a diagonally dominant matrix. The implementation of DD-PCA includes the convex approach using the Alternating Direction Method of Multipliers (ADMM) and the non-convex approach using the iterative projection algorithm. Applications of DD-PCA to large covariance matrix estimation and global multiple testing are also included in this package.

Imports RSpectra, Matrix, quantreg, MASS

NeedsCompilation no

Repository CRAN

Date/Publication 2019-09-14 20:50:02 UTC

R topics documented:

ddpca-package *Diagonally Dominant Principal Component Analysis*

Description

Efficient procedures for fitting the DD-PCA (Ke et al., 2019, <arXiv:1906.00051>) by decomposing a large covariance matrix into a low-rank matrix plus a diagonally dominant matrix. The implementation of DD-PCA includes the convex approach using the Alternating Direction Method of Multipliers (ADMM) and the non-convex approach using the iterative projection algorithm. Applications of DD-PCA to large covariance matrix estimation and global multiple testing are also included in this package.

Details

The DESCRIPTION file:

Index of help topics:

This package contains DDPCA_nonconvex and DDPCA_convex function, which decomposes a positive semidefinite matrix into a low rank component, and a diagonally dominant component using either nonconvex approach or convex approach.

Note

Please cite the reference paper to cite this R package.

Author(s)

Tracy Ke [aut], Lingzhou Xue [aut], Fan Yang [aut, cre]

Maintainer: Fan Yang <fyang1@uchicago.edu>

References

Ke, Z., Xue, L. and Yang, F., 2019. Diagonally Dominant Principal Component Analysis. Journal of Computational and Graphic Statistics, under review.

DDHC *DD-HC test*

Description

Combining DDPCA with orthodox Higher Criticism for detecting sparse mean effect.

Usage

```
DDHC(X, known_Sigma = NA, method = "nonconvex", K = 1, lambda = 3,
max_iter_nonconvex = 15 ,SDD_approx = TRUE, max_iter_SDD = 20, eps = NA,
rho = 20, max_iter_convex = 50, alpha = 0.5, pvalcut = NA)
```

Arguments

`X`	A $n \times p$ data matrix, where each row is drawn i.i.d from $\mathcal{N}(\mu, \Sigma)$
`known_Sigma`	The true covariance matrix of data. Default NA. If NA, then Σ will be estimated from data matrix X.
`method`	Either "convex" or "noncovex", indicating which method to use for DDPCA.
`K`	Argument in function `DDPCA_nonconvex`. Need to be specified when `method = "nonconvex"`
`lambda`	Argument in function `DDPCA_convex`. Need to be specified when `method = "convex"`
`max_iter_nonconvex`	Argument in function `DDPCA_nonconvex`.
`SDD_approx`	Argument in function `DDPCA_nonconvex`.
`max_iter_SDD`	Argument in function `DDPCA_nonconvex`.
`eps`	Argument in function `DDPCA_nonconvex`.
`rho`	Argument in function `DDPCA_convex`.
`max_iter_convex`	Argument in function `DDPCA_convex`.
`alpha`	Argument in function `HCdetection`.
`pvalcut`	Argument in function `HCdetection`.

Details

See reference paper for more details.

Value

Returns a list containing the following items

`H`	0 or 1 scalar indicating whether H_0 the global null is rejected (1) or not rejected (0). The use of `H` is not recommended as it's approximately valid only when `p` is sufficiently large and mean effect in alternative is really sparse.
`HCT`	DD-HC Test statistic

Author(s)

Fan Yang <fyang1@uchicago.edu>

References

Ke, Z., Xue, L. and Yang, F., 2019. Diagonally Dominant Principal Component Analysis. Journal of Computational and Graphic Statistics, under review.

See Also

IHCDD, HCdetection, DDPCA_convex, DDPCA_nonconvex

Examples

```
library(MASS)
n = 200
p = 200
k = 3
rho = 0.5
a = 0:(p-1)
Sigma_mu = rho^abs(outer(a,a,'-'))
Sigma_mu = (diag(p) + Sigma_mu)/2 # Now Sigma_mu is a symmetric diagonally dominant matrix
B = matrix(rnorm(p*k),nrow = p)
Sigma = Sigma_mu + B %*% t(B)
X = mvrnorm(n,rep(0,p),Sigma)
results = DDHC(X,K = k)
print(results$H)
print(results$HCT)
```

DDPCA_convex — *Diagonally Dominant Principal Component Analysis using Convex approach*

Description

This function decomposes a positive semidefinite matrix into a low rank component, and a diagonally dominant component by solving a convex relaxation using ADMM.

Usage

```
DDPCA_convex(Sigma, lambda, rho = 20, max_iter_convex = 50)
```

Arguments

Sigma	Input matrix of size $n \times n$
lambda	The parameter in the convex program that controls the rank of the low rank component
rho	The parameter used in each ADMM update.
max_iter_convex	Maximal number of iterations of ADMM update.

Details

This function decomposes a positive semidefinite matrix `Sigma` into a low rank component `L` and a symmetric diagonally dominant component `A`, by solving the following convex program

$$\text{minimize} \quad 0.5 * \|\Sigma - L - A\|^2 + \lambda \|L\|_*$$

$$\text{subject to} \quad A \in SDD$$

where $\|L\|_*$ is the nuclear norm of `L` (sum of singular values) and `SDD` is the symmetric diagonally dominant cone.

Value

A list containing the following items

`L`	The low rank component
`A`	The diagonally dominant component

Author(s)

Fan Yang <fyang1@uchicago.edu>

References

Ke, Z., Xue, L. and Yang, F., 2019. Diagonally Dominant Principal Component Analysis. Journal of Computational and Graphic Statistics, under review.

See Also

`DDPCA_nonconvex`

Examples

```
library(MASS)
p = 30
n = 30
k = 3
rho = 0.5
a = 0:(p-1)
Sigma_mu = rho^abs(outer(a,a,'-'))
Sigma_mu = (diag(p) + Sigma_mu)/2 # Now Sigma_mu is a symmetric diagonally dominant matrix
mu = mvrnorm(n,rep(0,p),Sigma_mu)
B = matrix(rnorm(p*k),nrow = p)
F = matrix(rnorm(k*n),nrow = k)
Y = mu + t(B %*% F)
Sigma_sample = cov(Y)
result = DDPCA_convex(Sigma_sample,lambda=3)
```

DDPCA_nonconvex *Diagonally Dominant Principal Component Analysis using Nonconvex approach*

Description

This function decomposes a positive semidefinite matrix into a low rank component, and a diagonally dominant component using an iterative projection algorithm.

Usage

```
DDPCA_nonconvex(Sigma, K, max_iter_nonconvex = 15, SDD_approx = TRUE,
max_iter_SDD = 20, eps = NA)
```

Arguments

Sigma	Input matrix of size $n \times n$
K	A positive integer indicating the rank of the low rank component.
max_iter_nonconvex	Maximal number of iterations of the iterative projection algorithm.
SDD_approx	If set to TRUE, then the projection onto SDD cone step in each iteration will be replaced by projection onto DD cone followed by symmetrization. This approximation will reduce the computational cost, but the output matrix A may only be approximately diagonally dominant.
max_iter_SDD,eps	Arguments in function ProjSDD. Matters only when SDD_approx = False.

Details

This function performs iterative projection algorithm to decompose a positive semidefinite matrix Sigma into a low rank component L and a symmetric diagonally dominant component A. The projection onto the set of low rank matrices is done via eigenvalue decomposition, while the projection onto the symmetric diagonally dominant (SDD) cone is done via function ProjSDD, unless SDD_approx = TRUE where an approximation is used to speed up the algorithm.

Value

A list containing the following items

L	The low rank component
A	The diagonally dominant component

Author(s)

Fan Yang <fyang1@uchicago.edu>

References

Ke, Z., Xue, L. and Yang, F., 2019. Diagonally Dominant Principal Component Analysis. Journal of Computational and Graphic Statistics, under review.

See Also

DDPCA_convex

Examples

```
library(MASS)
p = 200
n = 200
k = 3
rho = 0.5
a = 0:(p-1)
Sigma_mu = rho^abs(outer(a,a,'-'))
Sigma_mu = (diag(p) + Sigma_mu)/2 # Now Sigma_mu is a symmetric diagonally dominant matrix
mu = mvrnorm(n,rep(0,p),Sigma_mu)
B = matrix(rnorm(p*k),nrow = p)
F = matrix(rnorm(k*n),nrow = k)
Y = mu + t(B %*% F)
Sigma_sample = cov(Y)
result = DDPCA_nonconvex(Sigma_sample,K=k)
```

HCdetection *Higher Criticism for detecting rare and weak signals*

Description

This function takes a bunch of p-values as input and ouput the Higher Criticism statistics as well as the decision (rejection or not).

Usage

```
HCdetection(p, alpha = 0.5, pvalcut = NA)
```

Arguments

p	A vector of size n containing p-values from data
alpha	A number between 0 and 1. The smallest alpha*n p-values will be used to calculate the HC statistic. Default is 0.5.
pvalcut	A number between 0 and 1. Those small p-values (smaller than pvalcut) will be taken away to avoid heavy tails of test statistic. Set it to NA is equivalent to setting it to $1/n$.

Details

This function is an adaptation of the Matlab code here http://www.stat.cmu.edu/~jiashun/Research/software/HC/

Value

Returns a list containing the following items

H	0 or 1 scalar indicating whether H_0 the global null is rejected (1) or not rejected (0)
HCT	Higher Criticism test statistic

Author(s)

Fan Yang <fyang1@uchicago.edu>

References

Donoho, D. and Jin, J., Higher criticism for detecting sparse heterogeneous mixtures. Ann. Statist. 32 (2004), no. 3, 962–994.

Ke, Z., Xue, L. and Yang, F., 2019. Diagonally Dominant Principal Component Analysis. Journal of Computational and Graphic Statistics, under review.

Examples

```
n = 1e5
data = rnorm(n)
p = 2*(1 - pnorm(abs(data)))
result = HCdetection(p)
print(result$H)
print(result$HCT)
```

IHCDD *IHC-DD test*

Description

Combining Innovated Higher Criticism with DDPCA for detecting sparse mean effect.

Usage

```
IHCDD(X, method = "nonconvex", K = 1, lambda = 3, max_iter_nonconvex = 15,
SDD_approx = TRUE, max_iter_SDD = 20, eps = NA, rho = 20, max_iter_convex = 50,
alpha = 0.5, pvalcut = NA)
```

Arguments

X	A $n \times p$ data matrix, where each row is drawn i.i.d from $\mathcal{N}(\mu, \Sigma)$
method	Either "convex" or "noncovex", indicating which method to use for DDPCA.
K	Argument in function DDPCA_nonconvex. Need to be specified when method = "nonconvex"
lambda	Argument in function DDPCA_convex. Need to be specified when method = "convex"
max_iter_nonconvex	Argument in function DDPCA_nonconvex.
SDD_approx	Argument in function DDPCA_nonconvex.
max_iter_SDD	Argument in function DDPCA_nonconvex.
eps	Argument in function DDPCA_nonconvex.
rho	Argument in function DDPCA_convex.
max_iter_convex	Argument in function DDPCA_convex.
alpha	Argument in function HCdetection.
pvalcut	Argument in function HCdetection.

Details

See reference paper for more details.

Value

Returns a list containing the following items

H	0 or 1 scalar indicating whether H_0 the global null is rejected (1) or not rejected (0). Not recommended for use.
HCT	IHC-DD Test statistic

Author(s)

Fan Yang <fyang1@uchicago.edu>

References

Ke, Z., Xue, L. and Yang, F., 2019. Diagonally Dominant Principal Component Analysis. Journal of Computational and Graphic Statistics, under review.

See Also

DDHC, HCdetection, DDPCA_convex, DDPCA_nonconvex

Examples

```
library(MASS)
n = 200
p = 200
k = 3
rho = 0.5
a = 0:(p-1)
Sigma_mu = rho^abs(outer(a,a,'-'))
Sigma_mu = (diag(p) + Sigma_mu)/2 # Now Sigma_mu is a symmetric diagonally dominant matrix
B = matrix(rnorm(p*k),nrow = p)
Sigma = Sigma_mu + B %*% t(B)
X = mvrnorm(n,rep(0,p),Sigma)
results = IHCDD(X,K = k)
print(results$H)
print(results$HCT)
```

`ProjDD` *Projection onto the Diagonally Dominant Cone*

Description

Given a matrix C, this function outputs the projection of C onto the cones of diagonally domimant matrices.

Usage

```
ProjDD(C)
```

Arguments

`C`	A $n \times n$ matrix

Details

This function projects the input matrix C of size $n \times n$ onto the cones of diagonally domimant matrices defined as

$$\{A = (a_{ij})_{1 \leq i \leq n, 1 \leq j \leq n} : a_{jj} \geq \sum_{k \neq j} |a_{jk}| \quad \text{for all} \quad 1 \leq j \leq n\}$$

The algorithm is described in Mendoza, M., Raydan, M. and Tarazaga, P., 1998. Computing the nearest diagonally dominant matrix.

Value

A $n \times n$ diagonally dominant matrix

Author(s)

Fan Yang <fyang1@uchicago.edu>

References

Mendoza, M., Raydan, M. and Tarazaga, P., 1998. Computing the nearest diagonally dominant matrix. Numerical linear algebra with applications, 5(6), pp.461-474.

Ke, Z., Xue, L. and Yang, F., 2019. Diagonally Dominant Principal Component Analysis. Journal of Computational and Graphic Statistics, under review.

See Also

ProjSDD

Examples

```
ProjDD(matrix(runif(100),nrow=10))
```

ProjSDD *Projection onto the Symmetric Diagonally Dominant Cone*

Description

Given a matrix C, this function outputs the projection of C onto the cones of symmetric diagonally domimant matrices using Dykstra's projection algorithm.

Usage

```
ProjSDD(A, max_iter_SDD = 20, eps = NA)
```

Arguments

A	Input matrix of size $n \times n$
max_iter_SDD	Maximal number of iterations of the Dykstra's projection algorithm
eps	The iterations will stop either when the Frobenious norm of difference matrix between two updates is less than `eps` or after `max_iter_SDD` steps. If set to `NA`, then no check will be done during iterations and the iteration will stop after `max_iter_SDD` steps. Default is `NA`.

Details

This function projects the input matrix C of size $n \times n$ onto the cones of symmetric diagonally domimant matrices defined as

$$\{A = (a_{ij})_{1 \le i \le n, 1 \le j \le n} : a_{ij} = a_{ji}, a_{jj} \ge \sum_{k \ne j} |a_{jk}| \quad \text{for all} \quad 1 \le j \le n, 1 \le i \le n\}$$

It makes use of Dykstra's algorithm, which is a variation of iterative projection algorithm. The two key steps are projection onto the diagonally domimant cone by calling function `ProjDD` and projection onto the symmetric matrix cone by simple symmetrization.

More details can be found in Mendoza, M., Raydan, M. and Tarazaga, P., 1998. Computing the nearest diagonally dominant matrix.

Value

A $n \times n$ symmetric diagonally dominant matrix

Author(s)

Fan Yang <fyang1@uchicago.edu>

References

Mendoza, M., Raydan, M. and Tarazaga, P., 1998. Computing the nearest diagonally dominant matrix. Numerical linear algebra with applications, 5(6), pp.461-474.

Ke, Z., Xue, L. and Yang, F., 2019. Diagonally Dominant Principal Component Analysis. Journal of Computational and Graphic Statistics, under review.

See Also

ProjDD

Examples

```
ProjSDD(matrix(runif(100),nrow=10))
```

Index