Package 'PDSCE'

February 19, 2015

Type Package

Title Positive definite sparse covariance estimators

Version 1.2

Date 2012-06-12

Author Adam J. Rothman

Maintainer Adam J. Rothman <arothman@umn.edu>

Depends R (>= 2.10)

Description A package to compute and tune some positive definite and sparse covariance estimators

License GPL-2

NeedsCompilation yes

Repository CRAN

Date/Publication 2013-06-13 07:39:01

R topics documented:

`PDSCE-package` *Positive definite sparse covariance estimators*

Description

A package to compute and tune some positive definite and sparse covariance estimators

Details

The main functions are `pdsoft`, `pdsoft.cv`, `band.chol`, and `band.chol.cv`.

Author(s)

Adam J. Rothman

Maintainer: Adam J. Rothman <arothman@umn.edu>

References

Rothman, A. J., Levina, E., and Zhu, J. (2010). A new approach to Cholesky-based covariance regularization in high dimensions. Biometrika 97(3): 539-550.

Rothman, A. J. (2012). Positive definite estimators of large covariance matrices. Biometrika 99(3): 733-740

`band.chol` *Computes the banded covariance estimator by banding the covariance Cholesky factor*

Description

Computes the k-banded covariance estimator by k-banding the covariance Cholesky factor as described by Rothman, Levina, and Zhu (2010).

Usage

```
band.chol(x, k, centered = FALSE, method = c("fast", "safe"))
```

Arguments

`x`	A data matrix with n rows and p columns. The rows are assumed to be a realization of n independent copies of a p-variate random vector.
`k`	The banding parameter (the number of sub-diagonals to keep as non-zero). Should be a non-negative integer.
`centered`	Logical: `centered=TRUE` should be used if the columns of `x` have already been centered.
`method`	The method to use. The default is `method="fast"`, which uses the Grahm-Schmidt style algorithm and must have $k \leq \min(n-2, p-1)$. Alternatively, `method="safe"` uses an inverse or generalized inverse to compute estimates of the regression coefficients and is more numerically stable (and capable of handling $k \in \{0, \ldots, p-1\}$ regardless of n).

Details

`method="fast"` is much faster than `method="safe"`. See Rothman, Levina, and Zhu (2010).

Value

The banded covariance estimate (a p by p matrix).

Author(s)

Adam J. Rothman

References

Rothman, A. J., Levina, E., and Zhu, J. (2010). A new approach to Cholesky-based covariance regularization in high dimensions. Biometrika 97(3): 539-550.

See Also

band.chol.cv

Examples

```
set.seed(1)
n=50
p=20
true.cov=diag(p)
true.cov[cbind(1:(p-1), 2:p)]=0.4
true.cov[cbind(2:p, 1:(p-1))]=0.4
eo=eigen(true.cov, symmetric=TRUE)
z=matrix(rnorm(n*p), nrow=n, ncol=p)
x=z%*% tcrossprod(eo$vec*rep(eo$val^(0.5), each=p),eo$vec)
sigma=band.chol(x=x, k=1)
sigma
```

---

band.chol.cv *Banding parameter selection for banding the covariance Cholesky factor.*

Description

Selects the banding parameter and computes the banded covariance estimator by banding the covariance Cholesky factor as described by Rothman, Levina, and Zhu (2010).

Usage

```
band.chol.cv(x, k.vec = NULL, method = c("fast", "safe"), nsplits = 10,
             n.tr = NULL, quiet = TRUE)
```

Arguments

x	A data matrix with n rows and p columns. The rows are assumed to be a realization of n independent copies of a p-variate random vector.
k.vec	An optional vector of candidate banding parameters (the possible number of subdiagonals to keep as non-zero). The default is the long vector `0:min(n-2, p-1)`.
method	The method to use. The default is `method="fast"`, which uses the Grahm-Schmidt style algorithm and must have $k \leq \min(n-2, p-1)$. Alternatively, `method="safe"` uses an inverse or generalized inverse to compute estimates of the regression coefficients and is more numerically stable (and capable of handling $k \in \{0, \ldots, p-1\}$ regardless of n).
nsplits	Number of random splits to use for banding parameter selection.
n.tr	Optional number of cases to use in the training set. The default is the nearest integer to $n(1 - 1/\log(n))$. The value must be in $\{3, \ldots, n-2\}$.
quiet	Logical: `quiet=TRUE` suppresses the printing of progress updates.

Details

`method="fast"` is much faster than `method="safe"`. See Rothman, Levina, and Zhu (2010).

Value

A list with

sigma	the covariance estimate at the selected banding parameter
best.k	the selected banding parameter
cv.err	the vector of validation errors, one for each entry in `k.vec`
k.vec	the vector of candidate banding parameters
n.tr	The number of cases used for the training set

Author(s)

Adam J. Rothman

References

Rothman, A. J., Levina, E., and Zhu, J. (2010). A new approach to Cholesky-based covariance regularization in high dimensions. Biometrika 97(3): 539-550.

See Also

band.chol

Examples

```
set.seed(1)
n=50
p=20
true.cov=diag(p)
true.cov[cbind(1:(p-1), 2:p)]=0.4
true.cov[cbind(2:p, 1:(p-1))]=0.4
eo=eigen(true.cov, symmetric=TRUE)
z=matrix(rnorm(n*p), nrow=n, ncol=p)
x=z%*% tcrossprod(eo$vec*rep(eo$val^(0.5), each=p),eo$vec)
cv.out=band.chol.cv(x=x)
plot(cv.out$k.vec, cv.out$cv.err)
cv.out$best.k
cv.out$sigma
```

`pdsoft` *A permutation invariant positive definite and sparse covariance matrix estimate*

Description

Computes the sparse and positive definite covariance matrix estimator proposed by Rothman (2012).

Usage

```
pdsoft(s, lam, tau = 1e-04, init = c("soft", "diag", "dense", "user"),
       s0 = NULL, i0 = NULL, standard = TRUE, tolin = 1e-08,
       tolout = 1e-08, maxitin = 10000, maxitout = 1000, quiet = TRUE)
```

Arguments

`s`	A realization of the p by p sample covariance matrix. More generally, any symmetric p by p matrix with positive diagonal entries.
`lam`	The tuning parameter λ on the penalty $\lambda \sum_{i \neq j} \|\sigma_{ij}\|$. Could be either a scalar or a p by p symmetric matrix with an irrelevant diagonal. When a matrix is used, the penalty has the form $\sum_{i \neq j} \lambda_{ij} \|\sigma_{ij}\|$.
`tau`	The logarithmic barrier parameter. The default is `tau=1e-4`, which works well when `standard=TRUE`.
`init`	The type of initialization. The default option `init="soft"` uses a positive definite version of the soft thresholded covariance or correlation estimate, depending on `standard`. The second option `init="diag"` uses diagonal starting values. The third option `init="dense"` uses the closed-form solution when `lam=0`. The fourth option `init="user"` allows the user to specify the starting point (one must then specify `s0` and `i0`, ensuring that `i0` is positive definite).
`s0`	Optional user supplied starting point for $\Sigma^{(0)}$; see Rothman (2012)
`i0`	Optional user supplied starting point for $\Omega^{(0)}$; see Rothman (2012)

standard	Logical: standard=TRUE first computes the observed sample correlation matrix from s, then computes the sparse correlation matrix estimate, and finally rescales to return the sparse covariance matrix estimate. The strongly recommended default is standard=TRUE.
tolin	Convergence tolerance for the inner loop of the algorithm that solves the lasso regression.
tolout	Convergence tolerance for the outer loop of the algorithm.
maxitin	Maximum number of inner-loop iterations allowed
maxitout	Maximum number of outer-loop iterations allowed
quiet	Logical: quiet=TRUE suppresses the printing of progress updates.

Details

See Rothman (2012) for the objective function and more information.

Value

A list with

sigma	covariance estimate
omega	inverse covariance estimate
theta	correlation matrix estimate, will be NULL if standard=FALSE
theta.inv	inverse correlation matrix estimate, will be NULL if standard=FALSE

Note

So long as s is symmetric with positive diagonal entries and init is not set to "user" (or init is set to "user" and i0 as a positive definite matrix), then omega is positive definite. If tolin and tolout are too large, or maxitin and maxitout are too small, then sigma may be indefinite.

Author(s)

Adam J. Rothman

References

Rothman, A. J. (2012). Positive definite estimators of large covariance matrices. Biometrika 99(3): 733-740

See Also

pdsoft.cv

Examples

```
set.seed(1)
n=50
p=20
true.cov=diag(p)
true.cov[cbind(1:(p-1), 2:p)]=0.4
true.cov[cbind(2:p, 1:(p-1))]=0.4
eo=eigen(true.cov, symmetric=TRUE)
z=matrix(rnorm(n*p), nrow=n, ncol=p)
x=z%*% tcrossprod(eo$vec*rep(eo$val^(0.5), each=p),eo$vec)
s=cov(x)*(n-1)/n
output=pdsoft(s=s, lam=0.3)
output$sigma
```

`pdsoft.cv` *Tuning parameter selection and computation for the positive definite and sparse covariance matrix estimator*

Description

Computes and selects the tuning parameter for the sparse and positive definite covariance matrix estimator proposed by Rothman (2012).

Usage

```
pdsoft.cv(x, lam.vec = NULL, standard = TRUE,
          init = c("diag", "soft", "dense"), tau = 1e-04,
          nsplits = 10, n.tr = NULL, tolin = 1e-08, tolout = 1e-08,
          maxitin = 10000, maxitout = 1000, quiet = TRUE)
```

Arguments

`x`	A data matrix with n rows and p columns. The rows are assumed to be a realization of n independent copies of a p-variate random vector.
`lam.vec`	An optional vector of candidate lasso-type penalty tuning parameter values. The default for `standard=TRUE` is `seq(from=0, to=1, by=0.05)` and the default for `standard=FALSE` is `seq(from=0, to=m, length.out=20)`, where `m` is the maximum magnitude of the off-diagonal entries in `s`. Both of these default choices are far from excellent and are time consuming, particularly for values close to zero. The user should consider refining this set by increasing its resolution in a narrower range.
`standard`	Logical: `standard=TRUE` first computes the observed sample correlation matrix from `s`, then computes the sparse correlation matrix estimate, and finally rescales to return the sparse covariance matrix estimate. The strongly recommended default is `standard=TRUE`.

init	The type of initialization used for the estimate computed at the maximum element in lam.vec. Subsequent initializations use the final iterates for sigma and omega at the previous value in lam.vec. The default option init="diag" uses diagonal starting values. The second option init="soft" uses a positive definite version of the soft thresholded covariance or correlation estimate, depending on standard. The third option init="dense" uses the closed-form solution when lam=0.
tau	The logarithmic barrier parameter. The default is tau=1e-4, which works well when standard=TRUE with the default choices for the convergence tolerances.
nsplits	The number of random splits to use for the tuning parameter selection.
n.tr	Optional number of cases to use in the training set. The default is the nearest integer to $n(1 - 1/\log(n))$. The value must be in $\{3, \ldots, n-2\}$.
tolin	Convergence tolerance for the inner loop of the algorithm that solves the lasso regression.
tolout	Convergence tolerance for the outer loop of the algorithm.
maxitin	Maximum number of inner-loop iterations allowed
maxitout	Maximum number of outer-loop iterations allowed
quiet	Logical: quiet=TRUE suppresses the printing of progress updates.

Details

See Rothman (2012) for the objective function and more information.

Value

A list with

sigma	covariance estimate at the selected tuning parameter
omega	inverse covariance estimate at the selected tuning parameter
best.lam	the selected value of the tuning parameter
cv.err	a vector of the validation errors, one for each element in lam.vec
lam.vec	the vector of candidate tuning parameter values
n.tr	the number of cases used for the training set

Note

It is always the case that omega is positive definite. If tolin and tolout are too large, or maxitin and maxitout are too small, then sigma may be indefinite.

Author(s)

Adam J. Rothman

References

Rothman, A. J. (2012). Positive definite estimators of large covariance matrices. Biometrika 99(3): 733-740

See Also

pdsoft

Examples

```
set.seed(1)
n=50
p=20
true.cov=diag(p)
true.cov[cbind(1:(p-1), 2:p)]=0.4
true.cov[cbind(2:p, 1:(p-1))]=0.4
eo=eigen(true.cov, symmetric=TRUE)
z=matrix(rnorm(n*p), nrow=n, ncol=p)
x=z%*% tcrossprod(eo$vec*rep(eo$val^(0.5), each=p),eo$vec)
output=pdsoft.cv(x=x)
plot(output$lam.vec, output$cv.err)
output$best.lam
output$sigma
```

Index