Package ‘space’

October 4, 2019

Version 0.1-1.1

Date 2008-3-7

Title Sparse PArtial Correlation Estimation

Author Jie Peng <jie@wald.ucdavis.edu> , Pei Wang <pwang@fhcrc.org>,
Nengfeng Zhou <nfzhou@umich.edu>, Ji Zhu <jizhu@umich.edu>.

Maintainer Pei Wang <pwang@fhcrc.org>

Description Partial correlation estimation with joint sparse
regression model.

License GPL (>= 2)

Repository CRAN

Date/Publication 2019-10-04 07:13:29 UTC

NeedsCompilation yes

Depends R (>= 2.10)

R topics documented:

`space.joint` *A function to estimate partial correlations using the Joint Sparse Regression Model*

Description

A function to estimate partial correlations using the Joint Sparse Regression Model

Usage

```
space.joint(Y.m, lam1, lam2=0, sig=NULL, weight=NULL,iter=2)
```

Arguments

`Y.m` numeric matrix. Columns are for variables and rows are for samples. Missing values are not allowed. It's recommended to first standardize each column to have mean 0 and l_2 norm 1.

`lam1` numeric value. This is the l_1 norm penalty parameter. If the columns of Y.m have norm one, then the suggested range of lam1 is: $O(n^{3/2}\Phi^{-1}(1-\alpha/(2p^2)))$ for small α such as 0.1.

`lam2` numeric value. If not specified, lasso regression is used in the Joint Sparse Regression Model (JSRM). Otherwise, elastic net regression is used in JSRM and *lam2* serves as the l_2 norm penalty parameter.

`sig` numeric vector. Its length should be the same as the number of columns of `Y.m`. It is the vector of σ^{ii} (the diagonal of the inverse covariance matrix). If not specified, σ^{ii} will be estimated during the model fitting with initial values *rep(1,p)*. The number of the iteration of the model fitting (*iter*) will then be at least 2. Note, the scale of *sig* does not matter.

`weight` numeric value or vector. It specifies the weights or the type of weights used for each regression in JSRM. The default value is NULL, which means all regressions will be weighted equally in the joint model. If *weight*$=1$, residue variances will be used for weights. If *weight*$=2$, the estimated degree of each variable will be used for weights. Otherwise, it should be a positive numeric vector, whose length is equal to the number of columns of *Y.m*.

`iter` integer. It is the total number of interactions in JSRM for estimating σ^{ii} and partial correlations. When *sig*$=NULL$ and/or *weight*$=NULL$ or 2, *iter* should be at least 2.

Details

`space.joint` uses a computationally efficient approach for selecting non-zero partial correlations under the high-dimension-low-sample-size setting (Peng and et.al., 2007).

Value

A list with two components

`ParCor` the estimated partial correlation matrix.

`sig.fit` numeric vector of the estimated diagonal σ^{ii}.

Author(s)

J. Peng, P. Wang, Nengfeng Zhou, Ji Zhu

References

J. Peng, P. Wang, N. Zhou, J. Zhu (2007), Partial Correlation Estimation by Joint Sparse Regression Model.

Meinshausen, N., and Buhlmann, P. (2006), High Dimensional Graphs and Variable Selection with the Lasso, Annals of Statistics, 34, 1436-1462.

Examples

```
############################################################################################
############################ (A) The simulated Hub.net example in Peng et. al. (2007).
############################################################################################
data(spaceSimu)

n=nrow(spaceSimu$Y.data)
p=ncol(spaceSimu$Y.data)
true.adj=abs(spaceSimu$ParCor.true)>1e-6

#################### view the network corresponding to the parcial correlation matrix in the simulation example
########### the following code can run only if the "igraph" is installed in the system.
#library(igraph)
#plot.adj=true.adj
#diag(plot.adj)=0
#temp=graph.adjacency(adjmatrix=plot.adj, mode="undirected")
#temp.degree=apply(plot.adj, 2, sum)
#V(temp)$color=(temp.degree>9)+3
#plot(temp, vertex.size=3, vertex.frame.color="white",layout=layout.fruchterman.reingold, vertex.label=NA, edge.


#################### estimate the parcial correlation matrix with various methods
alpha=1
l1=1/sqrt(n)*qnorm(1-alpha/(2*p^2))
iter=3

########### the values of lam1 were selected to make the results of different methods comparable.
#### 1. MB method
result1=space.neighbor(spaceSimu$Y.data, lam1=l1*0.7, lam2=0)
fit.adj=abs(result1$ParCor)>1e-6
sum(fit.adj==1)/2                  ##total number of edges detected
sum(fit.adj[true.adj==1]==1)/2  ##total number of true edges detected

#### 2. Joint method with no weight
result2=space.joint(spaceSimu$Y.data, lam1=l1*n*1.56, lam2=0, iter=iter)
fit.adj=abs(result2$ParCor)>1e-6
sum(fit.adj==1)/2                  ##total number of edges detected
sum(fit.adj[true.adj==1]==1)/2  ##total number of true edges detected

#### 3. Joint method with residue variance based weights
result3=space.joint(spaceSimu$Y.data, lam1=l1*n*1.86, lam2=0, weight=1, iter=iter)
fit.adj=abs(result3$ParCor)>1e-6
sum(fit.adj==1)/2                  ##total number of edges detected
sum(fit.adj[true.adj==1]==1)/2  ##total number of true edges detected
```

```
#### 4. Joint method with degree based weights
result4=space.joint(spaceSimu$Y.data, lam1=l1*n*1.61, lam2=0, weight=2, iter=iter)
fit.adj=abs(result4$ParCor)>1e-6
sum(fit.adj==1)/2                 ##total number of edges detected
sum(fit.adj[true.adj==1]==1)/2  ##total number of true edges detected
```

`space.neighbor` *A function to estimate partial correlations using the neighborhood selection approach*

Description

A function to estimate partial correlations using the neighborhood selection approach

Usage

```
space.neighbor(Y.m, lam1, lam2=0)
```

Arguments

`Y.m`	numeric matrix. Each column is for one variable and each row is for one sample. Missing values are not allowed. It's recommended to first standardize each column to have mean 0 and norm 1.
`lam1`	numeric value. This is the l_1 norm penalty parameeter. If the columns of Y.m have norm one, then the suggested range of lam1 is: $O(n^{1/2}\Phi^{-1}(1-\alpha/(2p^2)))$ for small α such as 0.1.
`lam2`	numeric value. If not specified, lasso regression is used in the neighborhood selection. Otherwise, elastic net regression is used and *lam2* serves as the l_2 norm penalty parameter.

Details

`space.neighbor` estimate partial correlations using the neighborhood selection approach (Meinshausen and Buhlmann, 2006).

Value

A list with two components

`ParCor`	the estimated partial correlation matrix.
`sig.fit`	numeric vector of the estimated σ^{ii}

Author(s)

J. Peng, P. Wang, N. Zhou, J. Zhu

References

J. Peng, P. Wang, N. Zhou, J. Zhu (2007). Partial Correlation Estimation by Joint Sparse Regression Model.

Meinshausen, N., and Buhlmann, P. (2006), High Dimensional Graphs and Variable Selection with the Lasso, Annals of Statistics, 34, 1436-1462.

Examples

```
##############################################################################################
############################ (A) The simulated Hub.net example in Peng et. al. (2007).
##############################################################################################
data(spaceSimu)

n=nrow(spaceSimu$Y.data)
p=ncol(spaceSimu$Y.data)
true.adj=abs(spaceSimu$ParCor.true)>1e-6

#################### view the network corresponding to the parcial correlation matrix in the simulation example
#################### view the network corresponding to the parcial correlation matrix in the simulation example
########### the following code can run only if the "igraph" is installed in the system.
#library(igraph)
#plot.adj=true.adj
#diag(plot.adj)=0
#temp=graph.adjacency(adjmatrix=plot.adj, mode="undirected")
#temp.degree=apply(plot.adj, 2, sum)
#V(temp)$color=(temp.degree>9)+3
#plot(temp, vertex.size=3, vertex.frame.color="white",layout=layout.fruchterman.reingold, vertex.label=NA, edge.


#################### estimate the parcial correlation matrix with various methods
alpha=1
l1=1/sqrt(n)*qnorm(1-alpha/(2*p^2))
iter=3


########### the values of lam1 were selected to make the results of different methods comparable.
#### 1. MB method
result1=space.neighbor(spaceSimu$Y.data, lam1=l1*0.7, lam2=0)
fit.adj=abs(result1$ParCor)>1e-6
sum(fit.adj==1)/2                  ##total number of edges detected
sum(fit.adj[true.adj==1]==1)/2  ##total number of true edges detected

#### 2. Joint method with no weight
result2=space.joint(spaceSimu$Y.data, lam1=l1*n*1.56, lam2=0, iter=iter)
fit.adj=abs(result2$ParCor)>1e-6
sum(fit.adj==1)/2                  ##total number of edges detected
sum(fit.adj[true.adj==1]==1)/2  ##total number of true edges detected

#### 3. Joint method with residue variance based weights
result3=space.joint(spaceSimu$Y.data, lam1=l1*n*1.86, lam2=0, weight=1, iter=iter)
fit.adj=abs(result3$ParCor)>1e-6
```

```
sum(fit.adj==1)/2                 ##total number of edges detected
sum(fit.adj[true.adj==1]==1)/2  ##total number of true edges detected

#### 4. Joint method with degree based weights
result4=space.joint(spaceSimu$Y.data, lam1=l1*n*1.61, lam2=0, weight=2, iter=iter)
fit.adj=abs(result4$ParCor)>1e-6
sum(fit.adj==1)/2                 ##total number of edges detected
sum(fit.adj[true.adj==1]==1)/2  ##total number of true edges detected
```

---

spaceSimu *A simulated data example for inferring the partial correlation matrix*

Description

A list containing a simulated data example for package `space`

Usage

```
data(spaceSimu)
```

Details

`Y.data` is a simulated array data with columns corresponding to samples and rows corresponding genes/clones. `ParCor.true` is the true partical correlation matrix used to simulate `Y.data`.

Value

spaceSimu is a list of two components:

`Y.data`	a numeric matrix consisting of 500 rows and 250 columns.
`ParCor.true`	a numeric matrix consisting of 500 rows and 500 columns.

References

J. Peng, P. Wang, N. Zhou, J. Zhu (2007). Partial Correlation Estimation by Joint Sparse Regression Model.

Index