Package ‘iDINGO’

October 12, 2017

Type Package

Title Integrative Differential Network Analysis in Genomics

Version 1.0.2

Date 2017-10-12

Author Min Jin Ha <mjha@mdanderson.org>, Caleb A. Class <caclass@mdanderson.org>

Maintainer Caleb A. Class <caclass@mdanderson.org>

Imports mvtnorm, glasso, parallel, GGMridge, visNetwork, scales

Depends igraph


License GPL-2

LazyLoad yes

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2017-10-12 20:55:29 UTC

R topics documented:

iDINGO-package .................................................. 2
brca ................................................................. 2
dingo ............................................................. 3
extendedBIC ....................................................... 4
gbm ................................................................. 5
Greg.em ........................................................... 6
idingo .............................................................. 7
plotNetwork ....................................................... 8
scaledMat ........................................................ 9
scoring.boot ...................................................... 10
scoring.boot.parallel ......................................... 11
iDINGO-package  iDINGO: Integrative Differential Network Analysis in Genomics

Description

This package jointly estimates group-specific partial correlations in a multi-level/platform data set.

Details

This package jointly estimates group-specific partial correlations in a multi-level/platform data set, considering the directionality of effects between platforms using a chain graph model.

Author(s)

Min Jin Ha, Caleb Class, Veerabhadran Baladandayuthapani, and Kim-Anh Do

brca  Modified TCGA Breast Cancer data

Description

Modified TCGA Breast Cancer data.

Usage

data(brca)

Format

Three data frames, with columns as standardized miRNA, gene, and protein expressions. One vector with the two classes of the samples (tumor or normal tissue). Also one iDINGO fit object, obtained by running the example in the idingo manual entry.
**Fit DINGO model**

**Description**

This function fits a DINGO model and calculates edge-wise differential scores for all pairwise edges among p variables.

**Usage**

```
dingo(dat, x, rhoarray=NULL, diff.score=T, B=100, verbose=T, cores=1)
```

**Arguments**

- `dat`: nxp data with colnames as genename
- `x`: a length n vector representing a binary covariate
- `rhoarray`: a vector representing candidate tuning parameters of glasso for fitting global network model. If it is one value, then we use the value as the tuning parameter. It is set by NULL as default and we select 100 candidate values.
- `diff.score`: a logical value. If TRUE, edge-wise differential scores are calculated from bootstrap standard error. Otherwise, we fit Steps 1 and 2 of DINGO model to get group specific GGMs (partial correlations)
- `B`: the number of bootstrap samples to calculate differential scores.
- `verbose`: if TRUE, lists the procedure
- `cores`: the number of cores to run in parallel for bootstrapping, set to 1 as a default. If more cores are specified than the recommended maximum (the number of cores detected minus 1), this value will be replaced by the recommended value.

**Value**

- `genepair`: a p(p-1)/2 x 2 matrix indicating all pairs of genes
- `levels.x`: a length 2 vector indicating levels of the binary covariate x, the first element is for group 1 and the second element is for group 2
- `R1`: a length p(p-1)/2 vector indicating partial correlations for group 1 and the order is corresponding to the order of genepair
- `R2`: a length p(p-1)/2 vector indicating partial correlations for group 2 and the order is corresponding to the order of genepair
- `boot.diff`: a p(p-1)/2 x boot.B matrix indicating bootstrapped difference, Fisher’s Z transformed R1 - R2. The rows are corresponding to the order of gene pair and the columns are corresponding to the bootstrap samples
- `diff.score`: a p(p-1)/2 vector of differential score corresponding to genepair
- `p.val`: a p(p-1)/2 vector of corrected p-values corresponding to genepair
- `rho`: selected tuning parameter of glasso fit
extendedBIC

P
q by p matrix of Global component of the DINGO model

Q
p by 2 matrix of the coefficient parameter of the local group specific component L(x) of the DINGO model.

Psi
p by p diagonal matrix of the noise covariance parameter of the local group specific component L(x) of the DINGO model.

step.times
a length 3 vector containing the elapsed time for Step 1, Step 2, and Bootstrap Scoring, respectively.

Author(s)
Min Jin HA mjha@mdanderson.org, Caleb CLASS caclass@mdanderson.org

Examples
data(gbm)
# Run DINGO (the first column, 'x', contains the group data).
# This may take 5-10 minutes.
## Not run: fit <- dingo(gbm[,1], gbm$x, diff.score = TRUE, B = 100, cores = 2)

extendedBIC

Extended bayesian information criteria for gaussian graphical models

Description
Extended bayesian information criteria for gaussian graphical models

Usage
extendedBIC(gamma, omegahat, S, n)

Arguments
gamma
a tuning parameter taking a scalar in [0,1] and leading to stronger penalization of large graphs

omegahat
a p x p matrix indicating an estimates of precision (inverse covariance) matrix

S
a p x p matrix indicating sample covariance matrix

n
a scalar indicating sample size

Value
Extended BIC penalized by the size of graphs

Author(s)
Min Jin Ha <mjha@mdanderson.org>
gbm

References

Examples

library(glasso)
data(gbm)
x = gbm[,1]
Y = gbm[,-1]

# Estimating inverse covariance matrix using GLasso #
S = cov(Y)

rhoarray = exp(seq(log(0.001),log(1),length=100))
BIC = rep(0,length(rhoarray))
for (rh in 1:length(rhoarray)) {
  fit.gl1 = glasso(S,rho=rhoarray[rh])
  BIC[rh] = extendedBIC(gamma=0,omegahat=fit.gl1$wi,S=S,n=nrow(Y))
}
rho = rhoarray[which.min(BIC)]
fit.gl2 = glasso(S,rho=rho)
Omega = fit.gl2$wi

gbm

Modified TCGA Glioblastoma data

Description
Modified TCGA Glioblastoma data.

Usage
data(gbm)

Format
A data frame with first column as a covariate and other columns as standardized gene expressions.
Description

This function fits the covariance regression model by Hoff and Niu (2012) using EM algorithm with the restriction of diagonal matrix for the noise variance.

Usage

```r
greg.em(formula, data = NULL, R = 1, tol = 1e-10, itmax = 1000, verbose = F)
```

Arguments

- **formula**: an object of class "formula" used in model.frame function
- **data**: a data frame used in model.frame function
- **R**: rank of the model
- **tol**: a stopping criterion
- **itmax**: maximum number of iteration
- **verbose**: If true, estimation results for each iteration are printed

Value

- **A**: MLE of the baseline covariance matrix
- **B**: MLE of the regression coefficients

Author(s)

Min Jin Ha <mjha@mdanderson.org>

References


Examples

```r
library(glasso)
data(gbm)
x = gbm[,1]Y = as.matrix(gbm[,2])p = ncol(Y)
# Estimating inverse covariance matrix using GLasso #S = cov(Y)
w.upper = which(upper.tri(S))
rhooarray = exp(seq(log(0.001),log(1),length=100))BIC = rep(0,length(rhooarray))
```
for (rh in 1:length(rhoarray)) {
    fit.gl1 = glasso(S,rho=rhoarray[rh])
    BIC[rh] = extendedBIC(gamma=0, omegahat=fit.gl1$wi, S=S, n=nrow(Y))
}
rho = rhoarray(which.min(BIC))
fit.gl2 = glasso(S,rho=rho)
Omega = fit.gl2$wi

# Fitting (Covariance Regression on transformed data)
diag.Omega = diag(Omega)
P = -Omega/diag.Omega
diag(P) = 0

tY = Y
mdat = apply(tY,2,mean)
sdat = apply(tY,2,sd)
std.tY = t((t(tY) - mdat)/sdat)
smat = diag(sdat)

## rank 1 covariance regression
fit.g = Greg.em(std.tY~x,R=1)

---

**idingo**  

**Fit iDINGO model**

---

**Description**

This function fits the iDINGO model and calculates edge-wise differential scores for all pairwise edges among p variables between multiple platforms.

**Usage**

```r
idingo(dat,dat2=NULL,dat3=NULL,x,plats=NULL,rhoarray=NULL,
       diff.score=T,B=100,verbose=T,cores=1)
```

**Arguments**

- `dat`  
  *n* x *p* dataframe/matrix with colnames as genename

- `dat2`  
  Second *n* x *p* dataframe/matrix with colnames as genename (optional)

- `dat3`  
  Third *n* x *p* dataframe/matrix with colnames as genename (optional)

- `x`  
  A length *n* vector representing a binary covariate

- `plats`  
  A length 1-3 vector (corresponding to the number of data sets submitted, with names for the platforms/levels of the data, such as "microRNA" or "RNAseq". This is optional, and default names "platN" will be used if names are not provided.

- `rhoarray`  
  A vector representing candidate tuning parameters of glasso for fitting global network model. If it is one value, then we use the value as the tuning parameter. It is set by NULL as default and we select 100 candidate values.
diff.score a logical value. If TRUE, edge-wise differential scores are calculated from bootstrap standard error. Otherwise, we fit Steps 1 and 2 of DINGO model to get group specific GGMs (partial correlations)

B the number of bootstrap samples to calculate differential scores.

verbose if TRUE, lists the procedure

cores the number of cores to run in parallel for bootstrapping, set to 1 as a default. If more cores are specified than the recommended maximum (the number of cores detected minus 1), this value will be replaced by the recommended value.

Value

genepair a p(p-1)/2 x 2 matrix indicating all pairs of genes

levels.x a length 2 vector indicating levels of the binary covariate x, the first element is for group 1 and the second element is for group 2

R1 a length p(p-1)/2 vector indicating partial correlations for group 1 and the order is corresponding to the order of genepair

R2 a length p(p-1)/2 vector indicating partial correlations for group 2 and the order is corresponding to the order of genepair

diff.score a p(p-1)/2 vector of differential score corresponding to genepair

p.val a p(p-1)/2 vector of corrected p-values corresponding to genepair

Author(s)

Caleb CLASS caclass@mdanderson.org, Min Jin HA mjha@mdanderson.org

Examples

data(brca)

# Run iDINGO with microRNA, RNA, and protein data.
# Generally, we recommend a minimum of 100 bootstraps.
## Not run: fit <- idingo(brca$mirna, dat2 = brca$rna, dat3 = brca$prot,
## x = brca$class, plats = c("microRNA", "RNA", "Protein"),
## diff.score = TRUE, B = 20, cores = 2)
## End(Not run)

plotNetwork

Plot differential network

Description

This function plots the differential network from a completed DINGO or iDINGO model.

Usage

plotNetwork(fit, threshold=0.05, thresh.type="p.val", layout="circular",
legend.pos="left")
scaledMat

scale a square matrix

Description

scale a square matrix to have unit diagonal elements.

Usage

scaledMat(x)
**scoring.boot**

**Arguments**

- **x**: a square matrix with positive diagonal elements

**Value**

- scaled matrix of x

**Author(s)**

Min Jin Ha mjha@mdanderson.org

---

**Description**

This function calculates standard errors for edge-wise partial correlation differences obtained from DINGO model.

**Usage**

```r
scoring.boot(stddat,z,Omega,A,B,boot.N=100,verbose=T)
```

**Arguments**

- **stddat**: standardized nxp data with colnames as genename
- **z**: a length n vector representing a binary covariate
- **Omega**: a p x p precision matrix for std dat which implies the global network
- **A**: p x p matrix of the MLE for the baseline covariance matrix which is obtained from A value of the Greg.em function.
- **B**: p x 2 matrix of the MLE for the regression coefficient which is obtained from B value of the Greg.em function
- **boot.B**: a scalar indicating the number of bootstraps
- **verbose**: if TRUE, lists the bootstrap replications

**Value**

- **genepair**: a p(p-1)/2 x 2 matrix indicating all pairs of genes
- **levels.z**: a length 2 vector indicating levels of the binary covariate z, the first element is for group 1 and the second element is for group 2
- **R1**: a length p(p-1)/2 vector indicating partial correlations for group 1 and the order is corresponding to the order of genepair
- **R2**: a length p(p-1)/2 vector indicating partial correlations for group 2 and the order is corresponding to the order of genepair
scoring.boot.parallel

boot.diff  a p(p-1)/2 x boot.B matrix indicating bootstrapped difference, Fisher's Z transformed R1 - R2. The rows are corresponding to the order of gene pair and the columns are corresponding to the bootstrap samples

diff.score  a p(p-1)/2 vector of differential score corresponding to genepair

p.val  a p(p-1)/2 vector of corrected p-values corresponding to genepair

Author(s)

Min Jin HA mjha@mdanderson.org

scoring.boot.parallel  Calculating differential score with parallel bootstrap scoring

Description

This function calculates standard errors for edge-wise partial correlation differences obtained from DINGO model. Bootstrapping is done in parallel using parSapply from the "parallel" library.

Usage

scoring.boot.parallel(stddat,z,Omega,A,B,boot.B=100,verbose=T,cores=1)

Arguments

stddat  standardized n x p data with colnames as genename
z  a length n vector representing a binary covariate
Omega  a p x p precision matrix for std dat which implies the global network
A  p x p matrix of the MLE for the baseline covariance matrix which is obtained from A value of the Greg.em function.
B  p x 2 matrix of the MLE for the regression coefficient which is obtained from B value of the Greg.em function
boot.B  a scalar indicating the number of bootstraps
verbose  if TRUE, lists the bootstrap replications
cores  the number of cores to run in parallel for bootstrapping, set to 1 as a default.

Value

genepair  a p(p-1)/2 x 2 matrix indicating all pairs of genes
levels.z  a length 2 vector indicating levels of the binary covariate z, the first element is for group 1 and the second element is for group 2
R1  a length p(p-1)/2 vector indicating partial correlations for group 1 and the order is corresponding to the order of genepair
R2  a length p(p-1)/2 vector indicating partial correlations for group 2 and the order is corresponding to the order of genepair
boot.diff  a p(p-1)/2 x boot.B matrix indicating bootstrapped difference, Fisher’s Z transformed R1 - R2. The rows are corresponding to the order of gene pair and the columns are corresponding to the bootstrap samples

diff.score a p(p-1)/2 vector of differential score corresponding to genepair

p.val    a p(p-1)/2 vector of corrected p-values corresponding to genepair

Author(s)

Min Jin HA mjha@mdanderson.org, Caleb CLASS caclass@mdanderson.org

---

**Sigmax**

*group specific covariance matrices*

**Description**

From parameters of DINGO model, group specific covariance matrices are obtained

**Usage**

```r
Sigmax(P = NULL, Q, Psi, x)
```

**Arguments**

- **P**
  a p x p matrix specifying global component

- **Q**
  the coefficient parameter matrix of covariance regression model using Greg.em function

- **Psi**
  the diagonal error variance matrix of covariance regression model using Greg.em function

- **x**
  a vector specifying group. This must be corresponding to the design matrix of Greg.em function

**Value**

group specific precision matrix

**Author(s)**

Min Jin Ha <mjha@mdanderson.org>
Examples

```r
library(glasso)
data(gbm)
x = gbm[,1]
Y = as.matrix(gbm[,1])
p = ncol(Y)
# Estimating inverse covariance matrix using GLasso #
S = cov(Y)
upr = which(upper.tri(S))

rhoarray = exp(seq(log(0.001),log(1),length=100))
BIC = rep(0,length(rhoarray))
for (rh in 1:length(rhoarray)) {
  fit.gl = glasso(S,rho=rhoarray[rh])
  BIC[rh] = extendedBIC(gamma=0,omegahat=fit.gl$wi,S=S,n=nrow(Y))
}
rho = rhoarray[which.min(BIC)]
fit.gl2 = glasso(S,rho=rho)
Omega = fit.gl2$wi

# Fitting (Covariance Regression on transformed data)
diag.Omega = diag(Omega)
P = -Omega/diag.Omega
diag(P) = 0
tY = Y
mdat = apply(tY,2,mean)
sdat = apply(tY,2,sd)
std.tY = t((t(tY) - mdat)/sdat)
smat = diag(sdat)

## rank 1 covariance regression
fit.g = Greg.em(std.tY~x,R=1)
## obtain covariance matrix of Y when x=1
sigmaX1 = Sigmax(Q=fit.g$B,P=P,Psi=fit.g$A,x=c(1,1))
```

---

**single.boot**

*Calculating differential score for a single bootstrap*

**Description**

This function calculates the edge-wise partial correlation difference for a single bootstrap.

**Usage**

```
single.boot(i, z, n, tY.org, P, levels.z, w.upper)
```
trans.Fisher

Arguments

- i: iteration number. This is not used within this function, but necessary for parSapply within scoring.boot.parallel function.
- z: a length n vector representing a binary covariate
- n: the number of rows in data
- tY.org: the transformed standardized data
- P: the global correlation component
- levels.z: the levels of the covariates
- w.upper: the upper triangular of Omega

Value

- boot.diff: the difference for this bootstrap

Author(s)

Min Jin HA mjha@mdanderson.org, Caleb CLASS caclass@mdanderson.org

Description

Fisher’s Z-transformation of (partial) correlation.

Arguments

- x: a vector having entries between -1 and 1

Value

Fisher’s Z-transformed values

Author(s)

Min Jin HA mjha@mdanderson.org
Index

*Topic datasets
  brca, 2
  gbm, 5

*Topic package
  iDINGO-package, 2

brca, 2

mingo, 3

ding. 4

extendedBIC, 4

gbm, 5
Greg.em, 6

iDINGO (iDINGO-package), 2
iDiNGO, 7
iDINGO-package, 2

plotNetwork, 8

scaledMat, 9
scoring.boot, 10
scoring.boot.parallel, 11
Sigmax, 12
single.boot, 13

trans.Fisher, 14