

# Package ‘iDINGO’

September 6, 2019

**Type** Package

**Title** Integrative Differential Network Analysis in Genomics

**Version** 1.0.3

**Date** 2019-09-04

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

**Maintainer** Caleb A. Class <caiclass@mdanderson.org>

**Imports** mvtnorm, glasso, parallel, GGMrIDGE, visNetwork, scales

**Depends** igraph

**Description** Fits covariate dependent partial correlation matrices for integrative models to identify differential networks between two groups. The methods are described in Class et. al., (2018) <doi:10.1093/bioinformatics/btx750> and Ha et. al., (2015) <doi:10.1093/bioinformatics/btv4

**License** GPL-2

**LazyLoad** yes

**RoxygenNote** 6.1.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2019-09-06 09:20:02 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

Sigmax . . . . .	12
single.boot . . . . .	13
trans.Fisher . . . . .	14

<b>Index</b>	<b>15</b>
--------------	-----------

---

iDINGO-package	<i>iDINGO: Integrative Differential Network Analysis in Genomics</i>
----------------	--

---

### Description

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

### Details

This packages 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	<i>Modified TCGA Breast Cancer data</i>
------	---

---

### 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.

dingo

*Fit DINGO model***Description**

This function fit 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	n x p data with colnames as gene name
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 $p(p-1)/2 \times 2$ 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 $p(p-1)/2 \times \text{boot.B}$ 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

P	p 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>

## References

Foygel, R. and Drton, M. (2010). Extended bayesian information criteria for gaussian graphical models. arXiv preprint arXiv:1011.6640 .

## 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.g11 = glasso(S,rho=rhoarray[rh])
  BIC[rh] = extendedBIC(gamma=0,omegahat=fit.g11$wi,S=S,n=nrow(Y))
}
rho = rhoarray[which.min(BIC)]
fit.g12 = glasso(S,rho=rho)
Omega = fit.g12$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**

```
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**

Hoff, P. D. and Niu, X. (2012) A covariance regression model. *Statistica Sinica*, 22, 729-753.

**Examples**

```
library(glasso)
data(gbm)
x = gbm[,1]
Y = as.matrix(gbm[,-1])
p = ncol(Y)
# Estimating inverse covariance matrix using GLasso #
S = cov(Y)
w.upper = 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.g11 = glasso(S,rho=rhoarray[rh])
  BIC[rh] = extendedBIC(gamma=0,omegahat=fit.g11$wi,S=S,n=nrow(Y))
}
rho = rhoarray[which.min(BIC)]
fit.g12 = glasso(S,rho=rho)
Omega = fit.g12$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

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

## Arguments

<code>dat</code>	nxp dataframe/matrix with colnames as genename
<code>dat2</code>	Second nxp dataframe/matrix with colnames as genename (optional)
<code>dat3</code>	Third nxp dataframe/matrix with colnames as genename (optional)
<code>x</code>	a length $n$ vector representing a binary covariate
<code>plats</code>	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.
<code>rhoarray</code>	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 [caiclass@mdanderson.org](mailto:caiclass@mdanderson.org), Min Jin HA [mjha@mdanderson.org](mailto: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")
```



**Arguments**

fit	output from running dingo() or idingo()
threshold	a numeric value containing the threshold for which edges will be included in the differential network plot. If 'thresh.type' is 'p.val', all edges with p-values below this threshold will be included in the plot. If 'thresh.type' is 'diff.score', all edges with absolute differential scores above this threshold will be included in the plot.
thresh.type	either 'p.val' or 'diff.score', defining which variable is used as threshold for edge inclusion.
layout	either 'circular' or one of igraph's supported layouts. If 'circular', dingo networks will be plotted in a circle, and idingo networks will be plotted as a cylinder (with each platform/level as a separate circle).
legend.pos	Legend position for multi-platform networks, in c("left", "right"). Legend is not included for single-platform networks.

**Value**

visNet	a network plot, using igraph and visNetwork
--------	---

**Note**

To calculate differential scores and p-values for use in network plot thresholding, diff.score must be set to TRUE in dingo() or idingo().

**Author(s)**

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

**Examples**

```
data(brca)

# Plot the iDINGO result using a p-value threshold of 0.01.
plotNetwork(brca$fit, threshold = 0.01, thresh.type = "p.val")
```

---

scaledMat	<i>scale a square matrix</i>
-----------	------------------------------

---

**Description**

scale a square matrix to have unit diagonal elements.

**Usage**

```
scaledMat(x)
```

**Arguments**

x a square matrix with positive diagonal elements

**Value**

scaled matrix of x

**Author(s)**

Min Jin Ha mjha@mdanderson.org

---

scoring.boot

*Calculating differential score*

---

**Description**

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

**Usage**

```
scoring.boot(stddat, z, Omega, A, B, boot.B=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  $p(p-1)/2 \times 2$  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

---

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 nxp data with colnames as genename
z	a length n vector representing a binary covariate
Omega	a $p \times p$ precision matrix for std dat which implies the global network
A	$p \times p$ matrix of the MLE for the baseline covariance matrix which is obtained from A value of the Greg.em function.
B	$p \times 2$ 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 $p(p-1)/2 \times 2$ 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

<code>boot.diff</code>	a $p(p-1)/2$ x <code>boot.B</code> 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
<code>diff.score</code>	a $p(p-1)/2$ vector of differential score corresponding to <code>genepair</code>
<code>p.val</code>	a $p(p-1)/2$ vector of corrected p-values corresponding to <code>genepair</code>

**Author(s)**

Min Jin HA [mjha@mdanderson.org](mailto:mjha@mdanderson.org), Caleb CLASS [caclass@mdanderson.org](mailto:caclass@mdanderson.org)

---

Sigmax

*group specific covariance matrices*

---

**Description**

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

**Usage**

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

**Arguments**

<code>P</code>	a $p \times p$ matrix specifying global component
<code>Q</code>	the coefficient parameter matrix of covariance regression model using <code>Greg.em</code> function
<code>Psi</code>	the diagonal error variance matrix of covariance regression model using <code>Greg.em</code> function
<code>x</code>	a vector specifying group. This must be corresponding to the design matrix of <code>Greg.em</code> function

**Value**

group specific precision matrix

**Author(s)**

Min Jin Ha <[mjha@mdanderson.org](mailto:mjha@mdanderson.org)>

**Examples**

```

library(glasso)
data(gbm)
x = gbm[,1]
Y = as.matrix(gbm[,-1])
p = ncol(Y)
# Estimating inverse covariance matrix using GLasso #
S = cov(Y)
w.upper = 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.g11 = glasso(S,rho=rhoarray[rh])
  BIC[rh] = extendedBIC(gamma=0,omegahat=fit.g11$wi,S=S,n=nrow(Y))
}
rho = rhoarray[which.min(BIC)]
fit.g12 = glasso(S,rho=rho)
Omega = fit.g12$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)
```

**Arguments**

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

**Value**

<code>boot.diff</code>	the difference for this bootstrap
------------------------	-----------------------------------

**Author(s)**

Min Jin HA [mjha@mdanderson.org](mailto:mjha@mdanderson.org), Caleb CLASS [caclass@mdanderson.org](mailto:caclass@mdanderson.org)

---

`trans.Fisher`

*Fisher's Z-transformation*

---

**Description**

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

**Arguments**

<code>x</code>	a vector having entries between -1 and 1
----------------	--

**Value**

Fisher's Z-transformed values

**Author(s)**

Min Jin HA [mjha@mdanderson.org](mailto:mjha@mdanderson.org)

# Index

\*Topic **datasets**

brca, [2](#)

gbm, [5](#)

\*Topic **package**

iDINGO-package, [2](#)

brca, [2](#)

dingo, [3](#)

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](#)