# Package 'DiffCorr'

August 25, 2023

**Type** Package

**Title** Analyzing and Visualizing Differential Correlation Networks in Biological Data

**Version** 0.4.3

**Date** 2023-08-20

**Maintainer** Atsushi Fukushima <afukushima@gmail.com>

**Depends** pcaMethods, igraph, fdrtool, multtest

**Description** A method for identifying pattern changes between 2 experimental conditions in correlation networks (e.g., gene co-expression networks), which builds on a commonly used association measure, such as Pearson's correlation coefficient. This package includes functions to calculate correlation matrices for high-dimensional dataset and to test differential correlation, which means the changes in the correlation relationship among variables (e.g., genes and metabolites) between 2 experimental conditions.

**License** GPL (> 3)

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2023-08-25 09:40:06 UTC

**RoxygenNote** 7.2.3

**Author** Atsushi Fukushima [aut, cre],
Kozo Nishida [aut]

## R topics documented:

---

`AraMetLeaves` *A metabolite data set from Arabidopsis leaves by GC-TOF/MS*

---

## Description

A metabolite data set. The Arabidopsis metabolome of the aerial regions of individual WT plants and the mto1 and tt4 mutants were analyzed by GC-TOF/MS.

## Details

50 samples (WT, n = 17; mto1, n = 13; and tt4, n = 20, biological replicates).

A matrix containing 59 metabolites (rows) and 50 observations (columns).

MetaboLights accession no.: MTBLS40

For comparisons with data from roots (Fukushima et al. 2011) we selected 59 commonly-detected metabolites in both datasets using MetMask http://metmask.sourceforge.net.

## Author(s)

Atsushi Fukushima

## References

Miyako Kusano, Atsushi Fukushima et al. BMC Syst Biol 2007 1:53

## AraMetRoots *A metabolite data set from Arabidopsis roots by GC-TOF/MS*

### Description

A metabolite data set. The Arabidopsis metabolome of the roots of individual WT plants and the mto1 and tt4 mutants were analyzed by GC-TOF/MS.

### Details

53 root samples (WT, n = 17; mto1 n = 16; and tt4, n = 20, biological replicates).

A matrix containing 59 metabolites (rows) and 53 observations (columns).

MetaboLights accession no.: MTBLS45

For comparisons with data from aerial parts (Kusano, Fukushima et al. 2007) we selected 59 commonly-detected metabolites in both datasets using MetMask http://metmask.sourceforge.net.

### Author(s)

Atsushi Fukushima

### References

Atsushi Fukushima et al. BMC Syst Biol 2011 5:1.

## cluster.molecule *Hierarchical clustering of molecules*

### Description

Cluster molecules

### Usage

```
cluster.molecule(
  data,
  method = "pearson",
  linkage = "average",
  absolute = FALSE
)
```

### Arguments

| | |
|---|---|
| data | matrix or data frame |
| method | c("pearson", "spearman", "kendall", "euclidean", "maximum", "manhattan", "canberra", "binary", or "minkowski") |
| linkage | c("average", "ward", "single", "complete", "mcquitty", "median", "centroid") |
| absolute | if TRUE, then 1-\|COR\| else 1-COR, default is FALSE |

### Value

an object of class 'hclust'

### Author(s)

Atsushi Fukushima

### Examples

```
cluster.molecule(as.matrix(t(iris[,1:4])), "pearson", "average")
```

---

## comp.2.cc.fdr *Export differential correlations between two conditions*

---

### Description

Export differential correlations of comparison of two correlation matrices

### Usage

```
comp.2.cc.fdr(
  output.file = "res.txt",
  data1,
  data2,
  method = "pearson",
  p.adjust.methods = "local",
  threshold = 0.05
)
```

### Arguments

| | |
|---|---|
| output.file | can specify file name of the results exported |
| data1 | data matrix under condition 1 |
| data2 | data matrix under condition 2 |
| method | c("pearson", "spearman", "kendall") |
| p.adjust.methods | c("local", holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none") |
| threshold | a threshold of significance levels of differential correlation |

## Value

a text file

## Author(s)

Atsushi Fukushima

## References

Fukushima, A. Gene (2013) 518, 209-214

## Examples

```
## Not run:
data(AraMetRoots)
AraMetRoots[AraMetRoots==0] <- NA
AraMetRootsImp <- completeObs(pca(log2(AraMetRoots), nPcs=3, method="ppca"))
comp.2.cc.fdr(output.file="res.txt", AraMetRootsImp[,1:17], method="spearman",
              AraMetRootsImp[,18:37], threshold=0.05)

## End(Not run)
```

---

# compcorr *Compare two correlation coefficients*

---

## Description

Compare two correlation coefficients using Fisher's Z-transformation

## Usage

```
compcorr(n1, r1, n2, r2)
```

## Arguments

| | |
|---|---|
| n1 | sample size under condition 1 |
| r1 | correlation coefficient under condition 1 |
| n2 | sample size under condition 2 |
| r2 | correlation coefficient under condition 1 |

## Value

list of result (diff and p-value)

## Author(s)

Atsushi Fukushima

## References

http://www.fon.hum.uva.nl/Service/Statistics/Two_Correlations.html http://support.sas.com/ctx/samples/index.jsp?sid=494 http://support.sas.com/ctx/samples/index.jsp?sid=494

## Examples

```
compcorr(10, 0.1, 10, 0.9)
```

---

`cor.dist` *Additional distance functions correlation distance (1-r)*

---

## Description

Additional distance functions Correlation distance (1-r)

## Usage

```
cor.dist(data, methods = "pearson", absolute = FALSE)
```

## Arguments

| | |
|---|---|
| `data` | a data matrix ([data.frame object] row: metabolites, col: samples or replicates) |
| `methods` | a character string indicating which correlation coefficient is to be calculated. One of "pearson" (default), "spearman", or "kendall" can be abbreviated. |
| `absolute` | TRUE means that absolute value of the correlation coefficient is used (Default: FALSE). |

## Details

These functions were originally from 'hybridHclust' package. We modified the functions slightly. See also the reference manual in detail.

## Value

the resulting correlation matrix

## Author(s)

Atsushi Fukushima

## Examples

```
cor.dist(as.matrix(t(iris[,1:4])))
```

## cor2.test *Correlation Test*

### Description

Correlation Test

### Usage

```
cor2.test(n, r, method = c("pearson", "kendall", "spearman"))
```

### Arguments

| | |
|---|---|
| n | the number of samples |
| r | the correlation coefficient |
| method | "pearson" and "spearman" can be used. |

### Value

p-value

### Author(s)

Atsushi Fukushima

### References

http://aoki2.si.gunma-u.ac.jp/R/cor2.html

### Examples

```
cor2.test(30, 0.6)
```

## generate_g *Generating graph from data matrix*

### Description

Generating graph from data matrix

## Usage

```
generate_g(
  data,
  method = "pearson",
  cor.thr = 0.6,
  neg.flag = 1,
  node.col = "red",
  node.size = 7,
  edge.col = "blue",
  edge.width = 3
)
```

## Arguments

| | |
|---|---|
| data | data matrix or data frame |
| method | c("Pearson", "Spearman", "Kendall") |
| cor.thr | a threshold of correlation coefficient (default: r >= 0.6) |
| neg.flag | flag where uses or not negative correlations |
| node.col | specifies color of nodes in a graph (default: red) |
| node.size | specifies size of nodes in a graph (default: 7) |
| edge.col | specifies color of edges in a graph (default: blue) |
| edge.width | specifies width of edges in a graph (default: 3) |

## Value

igraph object

## Author(s)

Atsushi Fukushima

## Examples

```
library(igraph)
mat <- matrix(runif(100), nr=10)
rownames(mat) <- as.character(1:10)
generate_g(mat)
```

`get.eigen.molecule` *Get eigen molecule*

## Description

Get eigen molecule

## Usage

```
get.eigen.molecule(data, groups, whichgroups = NULL, methods = "svd", n = 10)
```

## Arguments

| | |
|---|---|
| `data` | a data matrix ([data.frame object] row: molecules, col: samples or replicates) |
| `groups` | a vector of group memberships as returned by cutree |
| `whichgroups` | a vector of group numbers to examine |
| `methods` | c("svd", "nipals", "rnipals", "bpca", "ppca"). See also pca() function in pcaMethods package |
| `n` | top n principal components |

## Value

the resulting vector.

## Author(s)

Atsushi Fukushima

## Examples

```
library(pcaMethods)
data(golub, package = "multtest")
hc.mol1 <- cluster.molecule(golub[1:100, 1:27], "pearson", "average")
g1 <- cutree(hc.mol1, h=0.6)
res1 <- get.eigen.molecule(golub[1:100,], g1)
```

## get.eigen.molecule.graph *Getting graph from eigengene module list*

### Description

Getting graph from eigengene module list

### Usage

```
get.eigen.molecule.graph(eigen.list, label = "Module")
```

### Arguments

| | |
|---|---|
| eigen.list | the resulting vector from get.eigen.molecule |
| label | a label of module extracted (default: "Module") |

### Value

igraph object

### Author(s)

Atsushi Fukushima

### Examples

```
library(pcaMethods)
library(igraph)
data(golub, package = "multtest")
hc.mol1 <- cluster.molecule(golub[, 1:27], "pearson", "average")
g1 <- cutree(hc.mol1, h=0.4)
res1 <- get.eigen.molecule(golub, g1)
g1.eigen <- get.eigen.molecule.graph(res1)
```

## get.lfdr *Getting local false discovery rate (lfdr)*

### Description

Getting local false discovery rate (lfdr) using 'fdrtool' package

### Usage

```
get.lfdr(r)
```

### Arguments

| | |
|---|---|
| r | a vector of correlation coefficient under condition |

### Value

list of lfdr

### Author(s)

Atsushi Fukushima

### References

Strimmer, K. Bioinformatics (2008) 24, 1461-1462

### Examples

```
library("fdrtool")
data(pvalues)
get.lfdr(pvalues)
```

---

## get.min.max *Get minimum and maximum*

---

### Description

Get minimum and maximum

### Usage

```
get.min.max(d)
```

### Arguments

| | |
|---|---|
| d | data matrix or data frame |

### Value

list object of minimum value or maximum value in a data

### Author(s)

Atsushi Fukushima

### Examples

```
get.min.max(iris[,1:2])
```

`plotClusterMolecules` *Plot cluster molecules*

## Description

Plot cluster molecules

## Usage

```
plotClusterMolecules(
  data,
  groups = NULL,
  group.no = NULL,
  title = NULL,
  ylim = NULL,
  order = NULL,
  scale.center = FALSE,
  scale.scale = FALSE,
  frame = "white",
  col = NULL,
  bottom.mar = 5,
  xlab = "Samples",
  ylab = "Relative abundance"
)
```

## Arguments

| | |
|---|---|
| `data` | data matrix or data frame |
| `groups` | a vector of group memberships as returned by cutree |
| `group.no` | the group number to be plotted |
| `title` | a title for the graph |
| `ylim` | a vector indicating the upper and lower limit for the y-axis |
| `order` | whether or not to order the columns of the data matrix |
| `scale.center` | unless NULL, each row is scaled using scale |
| `scale.scale` | unless NULL, each row is scaled using scale. |
| `frame` | the color of the frame that is drawn as the background of the plot |
| `col` | If NULL, all genes will be drawn in the default color (blue). If the text "random" is given, then a set of colors will be generated by |
| `bottom.mar` | The size of the bottom margin of the plots as sent in par(mar=c(...)) |
| `xlab` | a lalel of x axis (defalt: "Samples") |
| `ylab` | a lalel of y axis (defalt: "Relative abundance") |

## Value

a graph

## Author(s)

Atsushi Fukushima

## References

this function was originally from Watson M (2005) BMC Bioinformatics 7:509

## Examples

```
library(pcaMethods)
data(golub, package = "multtest")
hc.mol1 <- cluster.molecule(golub[, 1:27], "pearson", "average")
g1 <- cutree(hc.mol1, h=0.4)
plotClusterMolecules(golub[,1:27], g1, 3)
```

---

# plotDiffCorrGroup *Plot DiffCorr group*

---

## Description

Plot DiffCorr group

## Usage

```
plotDiffCorrGroup(
  data,
  groups1 = NULL,
  groups2 = NULL,
  group1.no = NULL,
  group2.no = NULL,
  g1,
  g2,
  g1.order = NULL,
  g2.order = NULL,
  title1 = NULL,
  title2 = NULL,
  ...
)
```

## Arguments

| | |
|---|---|
| data | a data matrix or data frame |
| groups1 | a vector of row group membership as produced by cutree under condition 1 |
| groups2 | a vector of row group membership as produced by cutree under condition 2 |
| group1.no | the group number to be plotted (condition 1) |
| group2.no | the group number to be plotted (condition 2) |
| g1 | a vector describing the columns of the data belonging to condition 1 |
| g2 | a vector describing the columns of the data belonging to condition 2 |
| g1.order | whether or not to order the columns of the data matrix for condition 1. If "average", then the columns are ordered by the average expression value. If the name of a gene (row), then the columns are orderd according to the expression levels of that gene. If NULL, columns remain in their original order. |
| g2.order | See g1.order |
| title1 | A title for the left hand graph |
| title2 | A title for the right hand graph |
| ... | other parameters to be passed to this function |

## Value

a graph

## Author(s)

Atsushi Fukushima

## Examples

```
library(pcaMethods)
data(golub, package = "multtest")
hc.mol1 <- cluster.molecule(golub[, 1:27], "pearson", "average")
hc.mol2 <- cluster.molecule(golub[, 28:38], "pearson", "average")
g1 <- cutree(hc.mol1, h=0.4)
g2 <- cutree(hc.mol2, h=0.4)
##
plotDiffCorrGroup(golub, g1, g2, 21, 24, 1:27, 28:38,
                  scale.center=TRUE, scale.scale=TRUE,
                  ylim=c(-5,5))
```

scalingMethods *scalingMethods*

## Description

The pre-treatment methods

## Usage

```
scalingMethods(
  data,
  methods = c("auto", "range", "pareto", "vast", "level", "power")
)
```

## Arguments

| | |
|---|---|
| data | a data matrix ([data.frame object] row: molecules, col: samples or replicates) |
| methods | the chosen methods. |

## Value

the resulting data frame (or scaled data matrix)

## Author(s)

Atsushi Fukushima

## Examples

```
scalingMethods(iris[,1:4], "level")
```

uncent.cor2dist *Additional distance functions correlation distance (uncentered)*

## Description

Additional distance functions correlation distance (uncentered)

## Usage

```
uncent.cor2dist(data, i, absolute = FALSE)
```

### Arguments

| | |
|---|---|
| data | a data matrix ([data.frame object] row: metabolites, col: samples or replicates) |
| i | i-th row of data |
| absolute | TRUE means that absolute value of the correlation coefficient is used (Default: FALSE). |

### Details

These functions were originally from 'hybridHclust' package. We modified the functions slightly. See also the reference manual in detail.

### Value

the resulting correlation matrix

### Author(s)

Atsushi Fukushima

### Examples

```
uncent.cor2dist(as.matrix(t(iris[,1:4])), 1) ## NOT RUN!
```

---

## uncent.cordist *Calculating all pairwise distances using correlation distance*

---

### Description

Calculating all pairwise distances using correlation distance

### Usage

```
uncent.cordist(data, absolute = FALSE)
```

### Arguments

| | |
|---|---|
| data | a data matrix ([data.frame object] row: metabolites, col: samples or replicates) |
| absolute | TRUE means that absolute value of the correlation coefficient is used (Default: FALSE). |

### Details

These functions were originally from 'hybridHclust' package. We modified the functions slightly. See also the reference manual in detail.

### Value

the resulting correlation matrix

### Author(s)

Atsushi Fukushima

### Examples

```
uncent.cordist(as.matrix(t(iris[,1:4]))) ## NOT RUN!
```

---

## write.modules *Writing modules into a text file*

---

### Description

Writing modules into a text file

### Usage

```
write.modules(cutree.res, mod.list, outfile = "module_list.txt")
```

### Arguments

| | |
|---|---|
| cutree.res | the result of cutree function |
| mod.list | the result of get.eigen.molecule |
| outfile | file name of output |

### Value

a text file

### Author(s)

Atsushi Fukushima

### Examples

```
## Not run:
data(golub, package = "multtest")
hc.mol1 <- cluster.molecule(golub[, 1:27], "pearson", "average")
g1 <- cutree(hc.mol1, h=0.4)
res1 <- get.eigen.molecule(golub, g1)
write.modules(g1, res1)

## End(Not run)
```

# Index