Package 'fCI'

April 14, 2017

Type Package

Title f-divergence Cutoff Index for Differential Expression Analysis in Transcriptomics and Proteomics

Version 1.4.0

Date 2016-02-03

Author Shaojun Tang

Maintainer Shaojun Tang <tangshao2008@gmail.com>

Description (f-divergence Cutoff Index), is to find DEGs in the transcriptomic & proteomic data, and identify DEGs by computing the difference between the distribution of fold-changes for the control-control and remaining (non-differential) case-control gene expression ratio data. fCI provides several advantages compared to existing methods.

License GPL (>= 2)

Depends R (>= 3.1),FNN, psych, gtools, zoo, rgl, grid, VennDiagram

Suggests knitr, rmarkdown, BiocStyle

VignetteBuilder knitr

NeedsCompilation no

biocViews Proteomics

R topics documented:

`call.npci` *the s4 class function*

Description

the s4 class function

Usage

```
call.npci(.Object)
```

Arguments

`.Object`	the fCI object

Details

The S4 method will compute DEGs and save the results to the original s4 object .Object

Value

NA No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("See README")
```

---

call.npci-methods ~~ *Methods for Function* call.npci ~~

Description

~~ Methods for function call.npci ~~

Methods

```
signature(.Object = "NPCI")
```

compute *the generic function 'compute' for s4 class*

Description

the generic function 'compute' for s4 class

Usage

```
compute(.Object)
```

Arguments

`.Object`

Details

TBD

Value

`NA` No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("See README")
```

compute-methods *~~ Methods for Function* `compute` *~~*

Description

~~ Methods for function `compute` ~~

Methods

```
signature(.Object = "NPCI")
```

`deg.pairwise.fold.change` *find targets that have a consistent fold change in the same direction (either up- or down-regulation)*

Description

find targets that have a consistent fold change in the same direction

Usage

```
deg.pairwise.fold.change(pairwise.wt.up.down.fold, pairwise.df.up.down.fold,
  d = 1, min.fold = 1.2)
```

Arguments

`pairwise.wt.up.down.fold`	a list of numeric values representing the fold changes between control replicates for every gene
`pairwise.df.up.down.fold`	a list of numeric values representing the fold changes between case and control replicates for every gene
`d`	the dimensionality of the database, if the dataset is from proteogenomics, then d=2
`min.fold`	minimum fold change to declare a gene to be dysregulated, by default, min.fold=2

Details

TBD

Value

`expression ratio`
: a dataframe of fCI gene expression ratios (folds) with none zero values defined by given control-control index (i.e. 1 & 2) and control-case index (i.e. 3&4)

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
wt.fold.changes=list(c(1.2,1.3,1.5,1.6))
df.fold.changes=list(c(1.1,1.3,1.4,1.6))
deg.pairwise.fold.change(wt.fold.changes,df.fold.changes)
```

`deg.up.down.info` *find targets and their detailed expression changes*

Description

given expression matrix, find targets and their detailed expression changes

Usage

```
deg.up.down.info(wt.index.in.list, df.index.in.list, npci,
use.normalization = FALSE, target.ratio = 0.5)
```

Arguments

`wt.index.in.list`	a list of numeric values representing the column indexes for control samples
`df.index.in.list`	a list of numeric values representing the column indexes for experimental samples
`npci`	the object npci
`use.normalization`	a boolean value indicating if the normalization will be applied or not
`target.ratio`	a numeric value indicating the expected fold changes, i.e, 1.5

Details

TBD

Value

`expression ratio`	a dataframe of fCI gene expression ratios (folds) defined by control-control index and control-case index

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("TBC")
```

`deseq.median.ratio.normalization` *data matrix normalization method*

Description

normalize expression matrix by first replicate's median gene expression values

Usage

```
deseq.median.ratio.normalization(npci.data)
```

Arguments

`npci.data`	a data frame containing non-zero numeric values (the data frame must contain more than one row and one column)

Details

TBD

Value

`data.frame`	a new dataframe with each column having the same median value

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
udata=data.frame(matrix(sample(3:100, 6*4), 6,4))
normalized.udata=deseq.median.ratio.normalization(udata)
```

`divergence.multivariate.distributions` *estimate fCI divergence for given samples of aritrary dimensions*

Description

estimate fCI divergence for given samples of aritrary dimensions

Usage

```
divergence.multivariate.distributions(null.data, diff.data, choice = 2)
```

Arguments

`null.data`	the empirical null dataset (a dataframe of none-zero ratio values)
`diff.data`	the case-control dataset (a dataframe of none-zero ratio values)
`choice`	choice=1 => cross entropy choice=2 => Helligan distance choice=3 => KL distance

Details

TBD

Value

`divergences`	The estimated divergence given control-control and case- control expression ratios

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
null.data=data.frame(matrix(sample(seq(from=0.1,to=10, by=0.01), 100), 100,1))
diff.data=data.frame(matrix(sample(seq(from=0.1,to=10, by=0.01), 100), 100,1))
divergence.multivariate.distributions(null.data, diff.data, choice = 2)
```

fCI-class *Class* "fCI"

Description

The main Class that defines the slots values

Objects from the Class

Objects can be created by calls of the form new("fCI", ...).

Slots

sample.data.file: Object of class "character" ~~

distance.matrix: Object of class "matrix" ~~

sample.data.normalized: Object of class "data.frame" ~~

attr.info: Object of class "data.frame" ~~

null.data.start: Object of class "matrix" ~~

diff.data.start: Object of class "matrix" ~~

expr.by.fold: Object of class "matrix" ~~

fold.cutoff.list: Object of class "list" ~~

rank.index.to.be.removed: Object of class "list" ~~

diff.gene.ids: Object of class "list" ~~

wt.index: Object of class "numeric" ~~

df.index: Object of class "numeric" ~~

ctr.indexes: Object of class "numeric" ~~

trt.indexes: Object of class "numeric" ~~

method.option: Object of class "numeric" ~~

use.ratio: Object of class "logical" ~~

percent.genes.to.scan: Object of class "numeric" ~~

num.genes.to.skip.each: Object of class "numeric" ~~

use.fold.change: Object of class "logical" ~~

wt.comb: Object of class "list" ~~

df.comb: Object of class "list" ~~

diff.ids: Object of class "list" ~~

result: Object of class "numeric" ~~

indexes.reconsidered: Object of class "numeric" ~~

center.by.gaussian.kernel: Object of class "logical" ~~

symmetric.fold: Object of class "logical" ~~

pairwise.diff.gene.ids: Object of class "list" ~~

Methods

No methods defined with class "fCI" in the signature.

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
showClass("fCI")
```

`fCI.call.by.index` *top level function call to find targets based on expression data and control & case indexes*

Description

top level function call to find targets based on expression data and control & case indexes

Usage

```
fCI.call.by.index(wt.indexes, df.indexes, data.file, use.normalization = FALSE,
  npci=NULL, short.report=TRUE)
```

Arguments

`wt.indexes`	The wild type sample column indexes in the matrix, i.e. 1,2
`df.indexes`	The diseases type sample column indexes in the matrix, i.e. 3,4
`data.file`	The expression matrix
`use.normalization`	boolean value whether you want the data to be normalized or not
`npci`	the fCI object
`short.report`	whether you want to have a report summary

Details

TBD

Value

`rtable`	A data frame of the detected targets

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
wt.indexes=1:2
df.indexes=3:4
data.file=data.frame(matrix(sample(3:100, 100*4, replace=TRUE), 100,4))
use.normalization=FALSE
npci=NULL
short.report=TRUE
fCI.call.by.index(wt.indexes, df.indexes, data.file, use.normalization,
  npci, short.report)
```

`fci.data` *data frame of gene expression*

Description

This data set gives the gene expression values for multiple control and case samples.

Usage

```
fci.data
```

Format

a matrix containing 1043 genes and 4 samples.

Value

`dataframe` A data frame of expression values

Source

software.steen.org

References

http://software.steenlab.org/fCI/

figures *generic function to draw figures of the current analysis*

Description

generic function to draw figures of the current analysis

Usage

```
figures(.Object)
```

Arguments

`.Object`

Details

TBD

Value

`NA` No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("See README")
```

figures-methods *generate figures for empirical null and case-control distributions*

Description

~~ Methods for function `figures` ~~

Methods

```
signature(.Object = "NPCI")
```

`find.fci.targets` *identify differentially expressed genes*

Description

identify differentially expressed genes

Usage

```
find.fci.targets(.Object, wt.indexes, df.indexes, data.file, use.normalization)
```

Arguments

`.Object`	the fCI object
`wt.indexes`	The wild type sample column indexes in the matrix, i.e. 1,2
`df.indexes`	The diseases type sample column indexes in the matrix, i.e. 3,4
`data.file`	The expression matrix
`use.normalization`	boolean value whether you want the data to be normalized or not

Details

TBD

Value

`NA`	No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
fci=new("NPCI")
fci.data=data.frame(matrix(sample(3:100, 1043*6, replace=TRUE), 1043,6))
targets=find.fci.targets(fci, c(1,2,3), c(4,5,6), fci.data)
head(show.targets(targets))
```

find.fci.targets-methods

~~ *Methods for Function* `find.fci.targets` ~~

Description

~~ Methods for function `find.fci` ~~

Methods

`signature(.Object = "NPCI")` the built-in method to compute fCI DEGs.

find.mid.point *find the middle value of the density distribution*

Description

find the middle value of the density distribution

Usage

```
find.mid.point(Y)
```

Arguments

`Y`

Details

TBD

Value

`position` The value the separates density into two halves

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
Y=density(sample(1:100, 50), bw=0.5)
find.mid.point(Y)
```

`get.fold.large.step` *generate fold change cutoff values for fCI divergence computation*

Description

generate fold change cutoff with a large step of 0.5 fold

Usage

```
get.fold.large.step()
```

Details

TBD

Value

`fold_values` A vector of predefined fold changes for fCI computation

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
get.fold.large.step()
```

`get.npci.data` *return a fCI object given the gene expression data*

Description

return a fCI object given the gene expression data

Usage

```
get.npci.data(sample.data.normalized, wt.index, df.index)
```

Arguments

`sample.data.normalized`

`wt.index`

`df.index`

Details

TBD

Value

`expression ratio`
: a dataframe of fCI gene expression ratios (folds) defined by control-control index and control-case index

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
sample.data.normalized=data.frame(matrix(sample(3:100, 100*4, replace=TRUE),
  100,4))
wt.index=c(1,2)
df.index=c(1,3)
get.npci.data(sample.data.normalized, wt.index, df.index)
```

get.npci.distance.matrix *generate the divergence estimation based of fold change cutoff values*

Description

generate the divergence estimation based of fold change cutoff values

Usage

```
get.npci.distance.matrix(npci.data, null.data.start, diff.data.start, choice = 2, rank.index.to.b
```

Arguments

npci.data	
null.data.start	
diff.data.start	
choice	
rank.index.to.be.removed	
expr.by.fold	
ctr.indexes	
trt.indexes	
use.intersect	
symmetric.fold	
fold.cutoff.list	

Details

TBD

Value

divergence	A matrix of computed divergences

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
data.file=data.frame(matrix(sample(3:100, 100*4, replace=TRUE), 100,4))
wt.index=c(1,2)
df.index=c(1,3)
npci=new("NPCI")
npci@wt.index=wt.index
npci@df.index=df.index
npci@sample.data.normalized=data.file
npci=initialize(npci)
npci=normalization(npci)
npci=populate(npci)

null.data.start=npci@null.data.start
diff.data.start=npci@diff.data.start
choice=2
rank.index.to.be.removed=npci@rank.index.to.be.removed
expr.by.fold=npci@expr.by.fold
ctr.indexes=npci@wt.index
trt.indexes=npci@df.index
use.intersect=FALSE
symmetric.fold=TRUE
fold.cutoff.list=npci@fold.cutoff.list

get.npci.distance.matrix(npci.data, null.data.start, diff.data.start,
  choice = 2, rank.index.to.be.removed, expr.by.fold, ctr.indexes, trt.indexes,
  use.intersect, symmetric.fold, fold.cutoff.list)
```

---

get.outline.index *find the outline genes of a given distribution*

Description

find the outline genes of a given distribution

Usage

```
get.outline.index(values)
```

Arguments

values

Details

TBD

Value

`indexes` remove the index of values that are outliners based on the t-test with alpha=0.05

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
values=rnorm(100)
get.outline.index(values)
```

`get.protein.fold.step` *generate fold-change cutoff on proteomics data (with large steps of 0.2-0.5 fold)*

Description

generate fold-change cutoff on proteomics data (with large steps of 0.2-0.5 fold)

Usage

```
get.protein.fold.step()
```

Details

TBD

Value

`folds` returning a vector of recommended fold ratios for proteomic study

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
get.protein.fold.step()
```

`get.rank.combinations` *fold change values*

Description

identify the fold change value indexes beyond the fCI estimation

Usage

```
get.rank.combinations(rank.index.to.be.removed, symmetric.fold)
```

Arguments

`rank.index.to.be.removed`	a list of integers representing the genes to be removed because it exceeds the predefined fold change, i.e 1.2 fold
`symmetric.fold`	a boolean value indicating the upregulation and downregulation are treatedly equally

Details

TBD

Value

`combinations`	a data frame of gene indexes

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
rank.index.to.be.removed=list(sample(1:100, 20))
symmetric.fold=TRUE
get.rank.combinations(rank.index.to.be.removed, symmetric.fold)
```

`get.rna.fold.step` *generate fCI fold-change cutoff values for typical RNA-Seq data*

Description

generate fCI fold-change cutoff values for typical RNA-Seq data

Usage

```
get.rna.fold.step()
```

Details

TBD

Value

`folds` a vector of fold changes fCI used for divergence computation

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
get.rna.fold.step()
```

initialize-methods ~~ *Methods for Function* initialize ~~

Description

~~ Methods for function initialize ~~

Methods

signature(.Object = "NPCI") this s4 class generic method initialize the fCI object once it is made

intersect.of.lists *find the common values of all vectors of a list*

Description

find the common values of all vectors of a list

Usage

```
intersect.of.lists(vectorlist)
```

Arguments

vectorlist	a list of list values which we want to use to find common values

Details

TBD

Value

intersects	the common values of lists

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("this function will be disabled!")
```

`is.installed` *package*

Description

test if a package is installed in the R library

Usage

```
is.installed(mypkg)
```

Arguments

`mypkg`	a R library name, such as FNN

Details

TBD

Value

`installation`	boolean value indicating the installation

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
is.installed('fCI')
```

multi.dimensional.fci.data

data frame of gene expression

Description

This data set gives the gene expression values for 14204 genes and the control and case samples were generated at two time points (bivariate data).

Usage

```
fci.data
```

Format

a matrix containing 14204 genes and 8 samples.

Value

dataframe	A data frame of expression values

Source

software.steen.org

References

http://software.steenlab.org/fCI/

normalization

generic function to normalize gene expression matrix

Description

generic function to normalize gene expression matrix

Usage

```
normalization(.Object)
```

Arguments

.Object	the predefined class object (i.e fCI=new("NPCI"))

Details

TBD

Value

NA	No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("See README")
```

---

normalization-methods ~~ *Methods for Function* normalization ~~

Description

~~ Methods for function normalization ~~

Methods

`signature(.Object = "NPCI")` the built-in method for fCI data normalization, by default, the data is normalized according to mean excluding the top 5 and bottom 5 percent.

NPCI-class *Class* "NPCI"

Description

The main Class that defines the slots values

Objects from the Class

Objects can be created by calls of the form `new("NPCI", ...)`.

Slots

`sample.data.file`: Object of class "character" ~~

`distance.matrix`: Object of class "matrix" ~~

`sample.data.normalized`: Object of class "data.frame" ~~

`attr.info`: Object of class "data.frame" ~~

`null.data.start`: Object of class "matrix" ~~

`diff.data.start`: Object of class "matrix" ~~

`expr.by.fold`: Object of class "matrix" ~~

`fold.cutoff.list`: Object of class "list" ~~

`rank.index.to.be.removed`: Object of class "list" ~~

`diff.gene.ids`: Object of class "list" ~~

`wt.index`: Object of class "numeric" ~~

`df.index`: Object of class "numeric" ~~

`ctr.indexes`: Object of class "numeric" ~~

`trt.indexes`: Object of class "numeric" ~~

`method.option`: Object of class "numeric" ~~

`use.ratio`: Object of class "logical" ~~

`percent.genes.to.scan`: Object of class "numeric" ~~

`num.genes.to.skip.each`: Object of class "numeric" ~~

`use.fold.change`: Object of class "logical" ~~

`wt.comb`: Object of class "list" ~~

`df.comb`: Object of class "list" ~~

`diff.ids`: Object of class "list" ~~

`result`: Object of class "numeric" ~~

`indexes.reconsidered`: Object of class "numeric" ~~

`center.by.gaussian.kernel`: Object of class "logical" ~~

`symmetric.fold`: Object of class "logical" ~~

`pairwise.diff.gene.ids`: Object of class "list" ~~

Methods

No methods defined with class "NPCI" in the signature.

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/NPCI/

See Also

TBD

Examples

```
showClass("NPCI")
```

`npci.gene.by.pvalues` *find most signficantly change fCI targets*

Description

identify the genes that change most significantly using inverse of log ratio the smaller the results, the more signficant the changes.

Usage

```
npci.gene.by.pvalues(npci.data, gene.indexes, ctr.indexes, trt.indexes)
```

Arguments

`npci.data`	a data frame containing non-zero numeric values (the data frame must contain more than one row and one column)
`gene.indexes`	the row ids of genes used for p-value calculation
`ctr.indexes`	The wild type sample column indexes in the matrix, i.e. 1,2
`trt.indexes`	The experimental sample column indexes in the matrix, i.e. 1,2

Details

TBD

Value

`pvalues`	a vector of pvalues

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
npci.data=data.frame(matrix(sample(3:100, 100*4, replace=TRUE), 100,4))
gene.indexes=sample(1:97, 25)
ctr.indexes=c(1,2)
trt.indexes=c(3,4)
npci.gene.by.pvalues(npci.data, gene.indexes, ctr.indexes, trt.indexes)
```

---

npci.index.reconsidered

find targets that have little evidence to be differentially expressed

---

Description

the function will be depreciated

Usage

```
npci.index.reconsidered(npci.data, expr.by.fold, null.data.start, diff.data.start, gene.indexes,
```

Arguments

npci.data	a data frame containing non-zero numeric values (the data frame must contain more than one row and one column)
expr.by.fold	a 1xN matrix of case-control fold changes for every gene of the total N genes
null.data.start	a Nx1 matrix of control-control fold changes
diff.data.start	a Nx1 matrix of case-control fold changes
gene.indexes	the genes used for differential expression analysis.
ctr.indexes	the control sample column indexes
trt.indexes	the case sample column indexes
left.fold	the minimum fold changes for downregulation
right.fold	the minimum fold changes for upregulation

Details

TBD

Value

values	genes wrongly considered as differentially expressed

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
data.file=data.frame(matrix(sample(3:100, 100*4, replace=TRUE), 100,4))
wt.index=c(1,2)
df.index=c(1,3)
npci=new("NPCI")
npci@wt.index=wt.index
npci@df.index=df.index
npci@sample.data.normalized=data.file
npci=initialize(npci)
npci=normalization(npci)
npci=populate(npci)
npci=compute(npci)
npci=summarize(npci)

npci.data=npci@sample.data.normalized
null.data.start=npci@null.data.start
diff.data.start=npci@diff.data.start
choice=2
rank.index.to.be.removed=npci@rank.index.to.be.removed
expr.by.fold=npci@expr.by.fold

ctr.indexes=1:2
trt.indexes=3:4
use.intersect=FALSE
symmetric.fold=TRUE
fold.cutoff.list=npci@fold.cutoff.list
gene.indexes=npci@diff.gene.ids
left.fold=2
right.fold=2
```

---

npci.index.to.be.removed

gene indexes that will be considered as targets

Description

This function will be depreciated.

Usage

```
npci.index.to.be.removed(expr.by.fold, d, symmetric.fold, max.rank,
l.max.rank, r.max.rank)
```

Arguments

expr.by.fold	a 1xN matrix of fold change between case and control for every genes in N genes
d	the dimension of the data, if RNA-Seq or LC-MS/MS data, d=1
symmetric.fold	a booleam valuable indicating whether to use the same fold change cutoff for upregulation and downregulation
max.rank	the maximum fold change, i.e 3 fold
l.max.rank	the maximum fold change for downregulation, i.e 1.5 fold
r.max.rank	the maximum fold change for upregulation, i.e 1.5 fold

Details

TBD

Value

indexes	gene (indexes) considered as differentially expressed

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("Function to be discarded!")
```

`npci.venn.diagram` *generate venn diagram for multiple fCI analysis*

Description

plot the overlap differentially expressed genes by pairwise fCI analysis

Usage

```
npci.venn.diagram(diff.gene.ids, i = 1, k = 1)
```

Arguments

`diff.gene.ids`	gene ids for genes that are differentially expressed
`i`	number of comparisons for fCI analysis, i,e 1 or 2
`k`	number of genes for fCI analysis

Details

TBD

Value

`figure`	the venn diagram plot

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
targets.run1=c(2:10)
targets.run2=c(1:8)
targets.run3=c(6:12)
diff.gene.ids=list(targets.run1, targets.run2, targets.run3)
npci.venn.diagram(diff.gene.ids)
```

`pairwise.change.occupancy` *find the targets whose fold changes occur consistently (upregulated or downregulated) in all fCI analysis*

Description

find the targets whose fold changes occur consistently (upregulated or downregulated) in all fCI analysis

Usage

```
pairwise.change.occupancy(common.ids, pairwise.index,
  pairwise.up.down, target.ratio)
```

Arguments

`common.ids`	the gene ids that are differentially expressed
`pairwise.index`	a list of the genes ids that differentially expressed in each of the fCI analysis
`pairwise.up.down`	a list of up regulatio (+1) or downregulation (-1) for each gene in fCI analysis
`target.ratio`	the expected fold changes

Details

TBD

Value

`consistent targets`	Gene (indexes) that are consistently changed in fCI pairwise analysis
`direction`	Gene (indexes) that are consistently upregulated (if < 0) or upregulated (if > 0)

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
common.ids=6:13
pairwise.index=list(c(4:13), c(6:15))
pairwise.up.down=list(c(sample(c(-1,1), 10, replace=TRUE)),
                      c(sample(c(-1,1), 10, replace=TRUE)))
target.ratio=0.5
pairwise.change.occupancy(common.ids, pairwise.index,
  pairwise.up.down, target.ratio)
```

---

populate *generic function to populate the fCI object based on provided data*

Description

generic function to populate the fCI object based on provided data

Usage

```
populate(.Object)
```

Arguments

`.Object`

Details

TBD

Value

`NA`	No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("See README")
```

populate-methods ~~ *Methods for Function* populate ~~

Description

~~ Methods for function populate ~~

Methods

signature(.Object = "NPCI") after fCI object is initialized, popular the slot values for the object

report.target.summary *generate the results (gene ids) in the data frame*

Description

generate the results (gene ids) in the data frame

Usage

```
report.target.summary(pairwise.diff.gene.ids)
```

Arguments

pairwise.diff.gene.ids	a list of the the differentially expression genes (its index) for each pairwise fCI analysis.

Details

TBD

Value

NA	No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("See README")
```

`setfCI` *the generic function 'setfCI' for s4 class*

Description

the generic function 'setfCI' for s4 class

Usage

```
setfCI(.Object, wt.index, df.index, fold.cutoff.list,
  center.distribution)
```

Arguments

`.Object`	the fCI object
`wt.index`	the control sample column ids, such as c(1,2)
`df.index`	the case sample column ids, such as c(1,2)
`fold.cutoff.list`	the predefined fold change cut-off such as list(seq(from=1.1, to=3.0, by=0.1))
`center.distribution`	a boolean value showing that if the users want to center the distribution or not

Details

TBD

Value

`NA`	No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
fci=new("NPCI")
fci=setfCI(fci, 7:8, 11:12, seq(from=1.1,to=3,by=0.1), TRUE)
```

`setfCI-methods` *~~ Methods for Function* `setfCI` *~~*

Description

~~ Methods for function `setfCI` ~~

Methods

```
signature(.Object = "NPCI")
```

`show.targets` *display the gene ids that are identified to be differentially regulated*

Description

display the gene ids that are identified to be differentially regulated

Usage

```
show.targets(.Object)
```

Arguments

`.Object`	the class object, for example, fCI=new("NPCI")

Details

TBD

Value

`NA`	No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("See README")
```

`show.targets-methods` ~~ *Methods for Function* `show.targets` ~~

Description

~~ Methods for function `show.targets` ~~

Methods

`signature(.Object = "NPCI")` the built-in method to show the fCI final DEGs.

`summarize` *result summerization*

Description

summerize the result after fCI computation is done

Usage

```
summarize(.Object)
```

Arguments

`.Object`	the class object, for exaple, fci = new("NPCI")

Details

TBD

Value

`NA`	No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
 data.file=data.frame(matrix(sample(3:100, 100*4, replace=TRUE), 100,4))
wt.index=c(1,2)
df.index=c(1,3)
npci=new("NPCI")
npci@wt.index=wt.index
npci@df.index=df.index
npci@sample.data.normalized=data.file
npci=initialize(npci)
npci=normalization(npci)
npci=populate(npci)
npci=summarize(npci)
```

---

summarize-methods *result summerization*

Description

summerize the result after fCI computation is done

Methods

```
signature(.Object = "NPCI")
```

---

total.library.size.normalization *normalize the gene expression based on the library size (summation) of the first sample replicate*

Description

normalize the gene expression based on the library size (summation) of the first sample replicate

Usage

```
total.library.size.normalization(sample.data)
```

Arguments

sample.data	a data frame of gene expression (noen-zero) with columns being the sample and rows being genes

Details

TBD

Value

dataframe	a data frame where column values were normalized by total library size

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
sample.data=data.frame(matrix(sample(3:100, 100*4, replace=TRUE), 100,4))
total.library.size.normalization(sample.data)
```

---

trim.size.normalization

normalize gene expression by exluding genes on the top 5 and bottom 5 percentage

---

Description

normalize gene expression by exluding genes on the top 5 and bottom 5 percentage

Usage

```
trim.size.normalization(sample.data)
```

Arguments

sample.data	a data frame of gene expression (noen-zero) with columns being the sample and rows being genes

Details

TBD

Value

dataframe	a data frame where column values were normalized by all genes except the top 5 percent and bottom 5 percent genes

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
sample.data=data.frame(matrix(sample(3:100, 100*4, replace=TRUE), 100,4))
trim.size.normalization(sample.data)
```

---

two.sample.log.ratio *compute the log ratios of two vectors*

Description

compute the log ratios of two vectors

Usage

```
two.sample.log.ratio(a, b)
```

Arguments

a	a vector of numeric values (value must be greater than 0)
b	a vector of numeric values (value must be greater than 0)

Details

TBD

Value

ratios	the log ratios of two vectors

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
a=10
b=2
two.sample.log.ratio(a, b)
```

---

two.sample.permutation.test

perform permuation test on two vectors

Description

perform permuation test on two vectors

Usage

```
two.sample.permutation.test(a, b)
```

Arguments

a	a vector of numeric values (value must be greater than 0)
b	a vector of numeric values (value must be greater than 0)

Details

TBD

Value

pvalue	the pvalue of permutation test

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
two.sample.permutation.test(sample(1:100, 20), sample(5:104, 20))
```

`venndiagram` *generate a venn diagram to show the differentially expression summaries accross pairwise fCI analysis*

Description

generate a venn diagram to show the differentially expression summaries accross pairwise fCI analysis

Usage

```
venndiagram(.Object)
```

Arguments

`.Object`	the class object, i.e, fci=new("NPCI")

Details

TBD

Value

`NA`	No values will be returned

Note

TBD

Author(s)

Shaojun Tang

References

http://software.steenlab.org/fCI/

See Also

TBD

Examples

```
print("See README")
```

---

venndiagram-methods ~~ *Methods for Function* venndiagram ~~

Description

~~ Methods for function venndiagram ~~

Methods

`signature(.Object = "NPCI")` generate the venn diagram to show the targets that shared among different fCI analysis

Index