Package 'miRComb'

June 25, 2015

Type Package

Title An R package for analyzing miRNA-mRNA interactions

Version 0.6.4

Date 2015-05-20

Author Maria Vila-Casadesús, Juanjo Lozano

Maintainer Maria Vila-Casadesús <maria.vila@ciberehd.org>

Description Find miRNA targets combining both biological and theoretical information.

License GPL-3

Depends
R (>= 3.0), methods, gplots, gtools, network, WriteXLS, Hmisc,RankProd, GOstats, limma, scatterplot3d, RamiGO, circlize,VennDiagram, xtable, ReactomePA, DESeq, miRData (>= 0.4)

Imports AnnotationDbi

Encoding UTF-8

URL http://mircomb.sourceforge.net

R topics documented:

information-package 2
addCorrelation 3
addCorrelation.R 4
addDatabase 6
addDiffexp 7
addFoldchanges 8
addNet 9
addScore 10
addSig 11
boxplotCorrelation 12
boxplotSamples 13

combinePval . . . 13
corObject-class . . . 15
correctPval . . . 16
data.obj . . . 17
GOanalysis . . . 18
miRNA . . . 19
mkReport . . . 20
mRNA . . . 21
openCytoscape . . . 21
pearson . . . 22
pheno.miRNA . . . 23
pheno.mRNA . . . 24
plot3d . . . 25
plotCircos . . . 26
plotCordist . . . 27
plotCorrelation . . . 28
plotDensity . . . 29
plotGO . . . 30
plotHeatmap . . . 30
plotNetwork . . . 31
plotPca . . . 32
plotVolcano . . . 33
removeSamp . . . 34
selSubsetCor . . . 35
selSubsetExprs . . . 35
summary.corObject . . . 36
topTable . . . 37
writeCsv . . . 38
writeExcel . . . 39
writeSif . . . 40

Index **41**

`information-package` *Find miRNA targets combining both biological and theoretical information.*

Description

Set of functions and databases useful for computing the miRNA targets from miRNA and mRNA expression data. It is based on the principle that miRNA targets need to be correlated and also be predicted on a database to be true target.

Details

Package:	information
Type:	Package
Version:	0.6.4
Date:	2015-06-18
License:	GPL-3

Author(s)

Maria Vila-Casadesús, Juanjo Lozano

Mantainer: Maria Vila-Casadesús: <maria.vila@ciberehd.org>

Contributor: Pau Erola (C++ code)

References

http://mircomb.sourceforge.net

Example data adapted from: Affò S, Dominguez M, Lozano JJ, Sancho-Bru P et al. *Transcriptome analysis identifies TNF superfamily receptors as potential therapeutic targets in alcoholic hepatitis*. Gut 2013 Mar;62(3):452-60. Doi: 10.1136/gutjnl-2011-301146.

See Also

miRData, methods, gplots, gtools, network, WriteXLS, Hmisc, RankProd, GOstats, limma, scatterplot3d, RamiGO, circlize, VennDiagram, xtable, ReactomePA, DESeq

---

addCorrelation *Correlate miRNA and mRNA expression*

Description

The function correlates miRNA and mRNA expression from a corObject and fills the `cor` and `pval` slots.

Usage

```
addCorrelation(obj, method = "pearson", subset.miRNA = obj@sig.miRNA,
  subset.mRNA = obj@sig.mRNA, common = NULL, alternative = "two.sided")
```

Arguments

obj	a corObject
method	method used for computing correlation: "pearson" or "spearman".
subset.miRNA	Optional, character vector with the names of the miRNAs to correlate.
subset.mRNA	Optional, character vector with the names of the mRNAs to correlate.
common	Optional, character vector with the names of the samples to correlate (the samples must appear in both miRNA and mRNA datasets.)
alternative	specification of the alternative hypothesis: "two-sided", "less" or "greater".

Value

corObject the slots cor and pval filled.

Note

addCorrelation.R is the slow version of this funcion, but has the option to compute if there are any influential samples.

See Also

corObject-class,cor, addCorrelation.R

Examples

```
data(data.obj)

data.obj.correlated<-addCorrelation.R(data.obj, method="pearson", alternative="less",
 subset.miRNA=c("hsa-miR-21","hsa-miR-200c"), subset.mRNA=c("A1BG","A1CF"))

data.obj.correlated@cor
data.obj.correlated@pval
```

---

addCorrelation.R *Correlation, old version*

Description

Correlation, old version

Usage

```
addCorrelation.R(obj, method = "pearson", subset.miRNA = obj@sig.miRNA,
 subset.mRNA = obj@sig.mRNA, common = NULL, d.influences = FALSE,
 alternative = "two.sided")
```

Arguments

obj	a corObject.
method	method used for computing correlation: "pearson" or "spearman".
subset.miRNA	Optional, character vector with the names of the miRNAs to correlate.
subset.mRNA	Optional, character vector with the names of the mRNAs to correlate.
common	Optional, character vector with the names of the samples to correlate (the samples must appear in both miRNA and mRNA datasets.)
d.influences	compute a matrix with the Cook's Distance of each sample in each miRNA-mRNA correlatio.
alternative	specification of the alternative hypothesis: "two-sided", "less" or "greater".

Details

Slow version of the correlation function. Works on large datasets but is much more slow. Moreover, Kendall correlation is specially slow. Use always addCorrelation function if it is possible.

If TRUE, a 3-dimension matrix is added to the info slot, labeled "influenting.sample". First dimension: miRNA names; second dimension: mRNA names; third dimension: sample names; fill: Cook's Distance for a specific sample in a specific miRNA-mRNA linear regression (defined by the dimension label-names).

Value

corObject the slots "cor" and "pval" filled. Optionally, a matrix named "influenting.sample" is added to the info slot.

Note

This function can take a long time to complete when is applied to large datasets.

See Also

corObject-class,cor, addCorrelation, cooks.distance

Examples

```
data(data.obj)

data.obj.correlated<-addCorrelation.R(data.obj, method="pearson", alternative="less",
 subset.miRNA=c("hsa-miR-21","hsa-miR-200c"), subset.mRNA=c("A1BG","A1CF"),
 d.influences=TRUE)

data.obj.correlated@cor
data.obj.correlated@pval
data.obj.correlated@info[["influenting.sample"]]
```

addDatabase *Intersect correlations with an external database.*

Description

For each miRNA-mRNA pair, add if this pair has been predicted as miRNA-target according to the desired external databases.

Usage

```
addDatabase(obj, database = "microCosm_v5_18_numeric", pval.ref=1, dat.sum=1)
```

Arguments

obj	a corObject with a cytofile slot already defined.
database	"microCosm_v5_18_numeric", or a character vector including: "microCosm_v5_18", "targetScan_v6.2_18" or a character with the name of the data.frame containing the database.
pval.ref	only for "microCosm_v5_18_numeric": p value to set if no information is given.
dat.sum	only if "microCosm_v5_18_numeric" is not selected. For future purposes, the minimun number of concurrences across databases that determine that a miRNA-mRNA pair is bioinformatically predicted. By default: 1.

Details

If the database is a customised data.frame, the row names must follow the format "miRNA_name:mRNA_name" (check head(microCosm_v5_18) or head(targetScan_v6.2_18))

Value

corObject The same corObject with new columns added. The name and content of the column is dependent on the database selected:

- microCosm_v5_18_numeric: pval.database: p value of the microCosm database. If there is no p value in the database, then the value pval.ref is assigned
- A character vector including the names of the databases: the following columns are created:
  - For each database: column called "dat.xxx", where "xxx" is the name of the database. The values are "0" (no target) or "1" (target).
  - A column called "dat.sum" which sums, for each row, the values of all "dat.xxx" columns.

References

http://www.targetscan.org/

http://www.ebi.ac.uk/enright-srv/microcosm/htdocs/targets/v5/

See Also

data(microCosm_v5_18), data(targetScan_v6.2_18)

Examples

```
## load databases
data(microCosm_v5_18)
data(targetScan_v6.2_18)

## load corObject example
data(data.obj)

## numeric example
data.obj.datadded.numeric <- addDatabase(data.obj, "microCosm_v5_18_numeric")
head(data.obj.datadded.numeric@net)

## non-numeric example, multiple databases
data.obj.datadded.multiple <- addDatabase(data.obj,
 c("microCosm_v5_18", "targetScan_v6.2_18"))
head(data.obj.datadded.multiple@net)
```

---

addDiffexp *Calculate differential expression*

Description

Calculate differential expression from a corObject

Usage

```
addDiffexp(obj, dataset, classes, method.dif = "t.test", method.adj = "BH",
 var.t.test = FALSE, trend = FALSE)
```

Arguments

obj	corObject
dataset	"miRNA" or "mRNA"
classes	column name of the column to compare in the phenodata slot. The column must contain "0" (reference) and "1" (case). Missing values are also allowed.
method.dif	"t.test", "wilcoxon", "limma" or "rankprod"
method.adj	Multiple testing correction method used (only for "t.test", "wilcoxon" or "limma". One of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none".
var.t.test	TRUE or FALSE (default). If TRUE, considers equality of variances in a T-test.
trend	if TRUE and method.dif="limma", use "limma-trend" method. (Recommended for log-normalised counts)

Value

corObject — with a diffexp.miRNA or diffexp.mRNA slot added. The diffexp is a data.frame with:

- FC: foldchange
- logratio: logratio
- meanExp: mean value log2-expression for the probe across all samples.
- pval: *p* values. In RankProd the minimum of both *p* values is reported.
- adj.pval: *p* values corrected for multiple testing. In the RankProd the pfp (estimated percentage of false positives, which are, in theory, equivalent to False Discovery Rate) are added.

References

http://www.bioconductor.org/packages/release/bioc/html/limma.html

http://www.bioconductor.org/packages/release/bioc/html/RankProd.html

http://www.bioconductor.org/packages/release/bioc/html/voom.html

See Also

package:limma, package:RankProd

Examples

```
data(miRNA)
data(mRNA)
data(pheno.miRNA)
data(pheno.mRNA)

minimal<-new("corObject",dat.miRNA=miRNA,dat.mRNA=mRNA,
pheno.miRNA=pheno.miRNA,pheno.mRNA=pheno.mRNA)

minimal.diffexp<-addDiffexp(minimal, "miRNA", classes="DvH",
 method.dif="limma")
head(minimal.diffexp@diffexp.miRNA)
```

---

addFoldchanges — *Add foldchanges to a cytofile*

Description

Adds information regarding to miRNA and mRNA differential expression on the cytofile slot of a corObject

Usage

```
addFoldchanges(obj)
```

Arguments

obj	a corObject with a cytofile, diffexp.miRNA and diffexp.mRNA slots already defined.

Value

corObject — The same corObject, but with the following columns added in the cytofile slot:

- logratio.miRNA
- logratio.mRNA
- meanExp.miRNA
- meanExp.mRNA

See Also

addDiffexp

Examples

```
## obtain minimal net slot
data(data.obj)
data.obj@net <- data.obj@net[,-c(5:ncol(data.obj@net))]
head(data.obj@net)

## add the foldchanges from diffexp.miRNA and diffexp.mRNA slots
data.obj<-addFoldchanges(data.obj)
head(data.obj@net)
```

---

addNet *Create a net slot*

Description

Creates and fills the net slot from a corObject.

Usage

```
addNet(obj)
```

Arguments

obj	a corObject with the slots pval and cor already defined

Details

The net slot is a data.frame where each row represents a potential miRNA-mRNA interaction. The row names are the concatenation of the miRNA and mRNA name (format "miRNA:mRNA") and the columns contain all the available information for each pair. Net is sorted by miRNA name and then by mRNA name.

Value

A data.frame with the cytofile slot filled.

Examples

```
data(data.obj)
data.obj@net <- data.frame()

## create the minimal net slot
data.obj<-addNet(data.obj)
head(data.obj@net)
```

---

addScore *Create a score*

Description

Create a score, needed for plot.network.corObject function

Usage

```
addScore(obj)
```

Arguments

obj	a corObject, with a net slot containing logratio.miRNA and logratio.mRNA columns.

Value

A corObject in which a column containing the score values (score) has been added to the net slot.

See Also

addFoldchanges, addDiffexp

Examples

```
data(data.obj)
data.obj@net$score<-NULL
head(data.obj@net)

data.obj<-addScore(data.obj)
head(data.obj@net)
```

addSig *Select significant miRNAs or mRNAs*

Description

Select significant miRNAs or mRNAs that will be used for correlationg their expression.

Usage

```
addSig(obj, dataset, FC=NA, logratio=foldchange2logratio(FC), pval=NA,
  adj.pval=NA, min.meanExp=NA, up=FALSE, dw=FALSE, manual=NULL)
```

Arguments

obj	a corObject.
dataset	"miRNA" or "mRNA"
FC	minimum FoldChange (in absolute value).
logratio	minimum logratio (in absolute value).
pval	maximum uncorrected *p* value.
adj.pval	maximum corrected *p* value.
min.meanExp	minimum mean expression.
up	Select only upregulated items (TRUE or FALSE).
dw	Select only downregulated items (TRUE or FALSE).
manual	character vector with miRNA or mRNA names.

Value

corObject	The same corObject with the slots sig.miRNA or sig.mRNA including the names of the miRNAs or mRNAs that will be used for correlation.

See Also

addDiffexp

Examples

```
data(data.obj)

## select the significant miRNAs and mRNAs
data.obj<-addSig(data.obj, "miRNA", adj.pval=0.05)
data.obj<-addSig(data.obj, "mRNA", adj.pval=0.05, FC=1.5)
```

`boxplotCorrelation` *Bopxlot, full*

Description

Combines boxplot with correlation

Usage

```
boxplotCorrelation(obj, miRNA, mRNA, col.color = 1, pos.leg = "topright",
 colors = c("turquoise", "violet"), ...)
```

Arguments

`obj`	a `corObject`
`miRNA`	character with the name of the miRNA to be plotted.
`mRNA`	character with the name of the miRNA to be plotted.
`col.color`	numeric. Number of the column in the `pheno.miRNA` slot that will be used for coloring.
`colors`	character vector indicating the colors to be used.
`pos.leg`	legend position.
`...`	other parameters

Value

It returns a plot.

- Top-right: boxplot for the miRNA
- Bottom-left: boxplot for the mRNA
- Bottom-right: plot of the correlation (see `plotCorrelation`.)

See Also

`plotCorrelation`

Examples

```
data(data.obj)
boxplotCorrelation(data.obj, "hsa-miR-200c", "TTF2", pos.leg="topleft")
```

boxplotSamples *Boxplots of samples expression*

Description

Plot boxplots of the miRNA or mRNA expression, for each sample. It is possible to colour the samples according to a phenotypical description.

Usage

```
boxplotSamples(obj, subset, col.color = 1, las = 1, colors = c("turquoise", "violet"))
```

Arguments

obj	a corObject
subset	"miRNA" or "mRNA"
col.color	number or name of the phenotype column which define the grouping variables.
las	las parameter
colors	character vector indicating the colors to be used.

See Also

plotCordist, boxplotCorrelation

Examples

```
data(data.obj)
boxplotSamples(data.obj, "miRNA", col.color=1)
boxplotSamples(data.obj, "mRNA", col.color=1)
```

combinePval *Combine* p *values*

Description

Combine two *p* values into one.

Usage

```
combinePval(obj, pval.1 = "pval", pval.2 = "pval.database", method="stouffer",
 w=c(1,1))
```

Arguments

`obj`	`corObject` with a `cytofile` slot defined. It must have at least two columns with p values to combine.
`pval.1`	column name (from the `cytofile` slot) of the first p value. By default: the p value of the correlation.
`pval.2`	column name (from the `cytofile` slot) of the second p value. By default: the p value from MicroCosm database.
`method`	`"stouffer"` or `"fisher"`
`w`	numeric vector of length two indicating the respective weights that will be applied to Stouffer combination. By default: no weighting.

Details

Stouffer and Weighted Stouffer (Lipták) combination is computed according to:

$$p_{comb} = 1 - \Phi\left(\frac{1}{\sqrt{w_1^2 + w_2^2}}\left(w_1\left(\Phi^{-1}\left(1 - p_1\right)\right) + w_2\left(\Phi^{-1}\left(1 - p_2\right)\right)\right)\right)$$

where,

$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{\frac{z^2}{2}}\,\mathrm{d}z$$

Fisher combination is computed according to:

$$t = -2\left(\ln p_1 + \ln p_2\right) \sim \chi_4^2$$

Value

`corObject`	a `corObject` in which a column containing the combined p values has been added to the `cytofile` slot.

References

For more information about the combination methods, see:

Zaykin D.V.. Optimally weighted Z-test is a powerful method for combining probabilities in meta analysis. Journal of Evolutionary Biology, 2011.

Gade G., Porzelius C., Fälth M., Brase J.C., Wuttig D., Kuner R., Binder H., Sültmann H., and Beissbarth T. Graph based fusion of miRNA and mRNA expression data improves clinical outcome prediction of prostate cancer. BMC Bioinformatics, 12(488), 2011.

Examples

```
data(data.obj)

## add column pval.database
data(microCosm_v5_18)
data.obj<-addDatabase(data.obj,"microCosm_v5_18_numeric")
```

```
## combine the two p-values
data.obj<-combinePval(data.obj, pval.1="pval", pval.2="pval.database")

head(data.obj@net)
```

---

corObject-class *Class* "corObject"

Description

Class object for storing all the information for a miRNA-mRNA correlation analysis.

Details

MiRNAs should be named according to miRBase 17, and mRNAs should be named according to HUGO gene symbol nomenclature.

Objects from the Class

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

Slots

`dat.miRNA`: Object of class `"matrix"`. Contains the miRNA expression (rows for miRNAs and columns for samples).

`dat.mRNA`: Object of class `"matrix"`. Contains the mRNA expression (rows for mRNAs and columns for samples).

`pheno.miRNA`: Object of class `"data.frame"`. Rows for samples and columns for phenotypical information.

`pheno.mRNA`: Object of class `"data.frame"`. Rows for samples and columns for phenotypical information.

`cor`: Object of class `"matrix"`. Rows for miRNAs and columns for mRNAs.

`pval`: Object of class `"matrix"`. Rows for miRNAs and columns for mRNAs.

`net`: Object of class `"data.frame"`. Rows for unique miRNA:mRNA pairs and columns for their corresponding information (at least: miRNA name, mRNA name, coefficient of correlation and *p* value).

`diffexp.miRNA`: Object of class `"data.frame"`. Rows for miRNAs and columns for their corresponding infomation (usually FC, logratio, mean expression, *p* value and corrected *p* value).

`diffexp.mRNA`: Object of class `"data.frame"`. Rows for mRNAs and columns for their corresponding infomation (usually FC, logratio, mean expression, *p* value and corrected *p* value).

`sig.miRNA`: Object of class `"vector"`. Vector specifying the miRNAs that are used for correlation.

`sig.mRNA`: Object of class `"vector"`. Vector specifying the mRNAs that are used for correlation.

`GO.results`: Object of class `"list"`. It contains the results of a GO analysis.

`info`: Object of class `"list"`. It contains the information of the tests and functions used.

Methods

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

Examples

```
## minimal corObject:

data(miRNA)
data(mRNA)
data(pheno.miRNA)
data(pheno.mRNA)

minimal<-new("corObject",dat.miRNA=miRNA,dat.mRNA=mRNA,
pheno.miRNA=pheno.miRNA,pheno.mRNA=pheno.mRNA)

str(minimal)

## corObject with more slots:
data(data.obj)
str(data.obj)
```

---

correctPval *Correct* p *values*

Description

Correct *p* values of one column of a `cytofile` slot.

Usage

```
correctPval(obj, method.adj = "BH", pval= "pval")
```

Arguments

`obj`	a `corObject` with a `cytofile` slot already defined.
`method.adj`	one of `"holm"`, `"hochberg"`, `"hommel"`, `"bonferroni"`, `"BH"`, `"BY"`, `"fdr"`, `"none"`.
`pval`	name of the column with the *p* values to correct.

Value

`corObject`	a `corObject` in which a column containing the corrected *p* values has been added to the `cytofile` slot.

See Also

`combinePval`, `p.adjust`

Examples

```
data(data.obj)
data.obj <- correctPval (data.obj, method.adj="BH", pval="pval")
```

`data.obj` *Example corObject*

Description

A corObject example, from vignette.

Usage

```
data("data.obj")
```

Format

The format is: Formal class 'corObject' [package "miRComb"] with 13 slots ..@ dat.miRNA : numeric matrix ..@ dat.mRNA : numeric matrix ..@ pheno.miRNA :'data.frame' ..@ pheno.mRNA :'data.frame' ..@ cor : numeric matrix ..@ pval : numeric matrix ..@ net :'data.frame' ..@ diffexp.miRNA:'data.frame' ..@ diffexp.mRNA :'data.frame' ..@ sig.miRNA : character ..@ sig.mRNA : character ..@ GO.results :List ..@ info :List

Source

Modified from: *Affò S, Dominguez M, Lozano JJ, Sancho-Bru P et al. Transcriptome analysis identifies TNF superfamily receptors as potential therapeutic targets in alcoholic hepatitis. Gut 2013 Mar;62(3):452-60. PMID: 22637703*

References

Affò S, Dominguez M, Lozano JJ, Sancho-Bru P et al. Transcriptome analysis identifies TNF superfamily receptors as potential therapeutic targets in alcoholic hepatitis. Gut 2013 Mar;62(3):452-60. PMID: 22637703

Examples

```
data(data.obj)
str(data.obj)
```

GOanalysis *GO and KEGG enrichment analysis*

Description

GO and KEGG enrichment analysis

Usage

```
GOanalysis(obj, type, ontology, pval.cutoff = 0.05,
                pval.column = "adj.pval", dat.sum =
                obj@info[["dat.sum"]], sub.miRNA = NULL, exclude.miRNA
                = NULL, sub.mRNA = NULL, organism = "human", FC =
                NULL, up = FALSE, dw = FALSE, add.miRNA = FALSE)
```

Arguments

obj	a corObject with a cytofile slot
type	"GO", "KEGG" or "REACTOME".
ontology	If type is "GO", the ontology to be analysed: "BP" (Biological Process), "CC" (Cellular Component) or "MF" (Molecular Function). If the type is "KEGG", write "KEGG"; if it is "REACTOME", write "REACTOME".
pval.cutoff	cutoff for the pairs to enter.
pval.column	column to apply the pval.cutoff: "pval.corrected" (default) or "pval".
dat.sum	minimum concurrences for the miRNA-mRNA pairs across the database(s) used
sub.miRNA	(optional) character vector, names of the miRNAs to limit the targets.
exclude.miRNA	(optional) character vector, the mRNA targets of these miRNAs will be removed.
sub.mRNA	(optional) character, use only these targets.
FC	(optional) minimum FC for the mRNAs.
up	if TRUE, select only upregulated mRNAs.
dw	if TRUE, select only upregulated mRNAs.
organism	"human" or "mouse".
add.miRNA	if TRUE, add the miRNAs that are regulating the selected mRNAs.

Value

corObject	the same corObject, with an item of the GO.results slot added. The item is a data.frame with the name "type:ontology" and with the following columns: • Ontology: "BP" (Biological Process), "CC" (Cellular Component), "MF" (Molecular Function), "KEGG" or "REACTOME" • ID: term ID • Pvalue: *p* value

- OddsRatio: number of mRNAs found/number of expected mRNAs
- ExpCount: expected number of mRNAs
- Count: number of mRNAs in the selected category
- Size: total number of mRNAs in the selected category
- Term: term name
- fdr: corrected *p* value with BH method
- genescat: mRNAs in the category
- (optional) miRNAs: miRNAs regulating these mRNAs

References

package:GOstats, package:ReactomePA

Falcon S and Gentleman R. Using GOstats to test gene list for GO term association. Bioinformatics, 23(7):257-8, 2007.

Yu G. ReactomePA: Reactome Pathway Analysis. R package version 1.10.1.

See Also

package:GOstats, package:ReactomePA

Examples

```
data(data.obj)
data.obj<-GOanalysis(data.obj,"GO","MF",pval.cutoff=0.05,dat.sum=1)
head(data.obj@GO.results[["GO:MF"]])

data.obj<-GOanalysis(data.obj,"KEGG","KEGG",pval.cutoff=0.05,dat.sum=1)
head(data.obj@GO.results[["KEGG:KEGG"]])
```

---

miRNA *miRNA data expression*

Description

miRNA data expression, in log2-intensity units

Usage

```
data(miRNA)
```

Format

The format is: num [1:1733, 1:12] 1.86 2.4 1.35 1.25 1.76 ... - attr(*, "dimnames")=List of 2 ..$: chr [1:1733] "hsa-let-7a" "hsa-let-7a*" "hsa-let-7a-2*" "hsa-let-7b" ... ..$: chr [1:12] "Control_1" "Control_2" "Control_3" "Case_1" ...

Source

Modified from Sancho-Bru P group data.

Examples

```
data(miRNA)
head(miRNA)
```

---

mkReport *Creates a pdf report*

Description

Creates a pdf report summarizing the contents of the corObject

Usage

```
mkReport(obj, file, title = "Default \\texttt{miRComb} output")
```

Arguments

obj	a corObject
file	name of the file, for example "myExampleReport"
title	Title of the report

Details

Documents myExampleReport.tex and myExampleReport.pdf will be created

Note

This function only works in Linux computers, with LaTeX and texlive already configured.

Some known problems:

- f this happens: `! ==> Fatal error occurred, no output PDF file produced!`
  Try: `sudo apt-get install texlive-recommended-fonts` or just `sudo apt-get install texlive-full`
- If you have problems with xcolor package: `xcolor.sty not found` `sudo apt-get install latex-xcolor`
- If you have problems with tikz package: `tikz.sty not found` `sudo apt-get install pgf`

Examples

```
### do not run

#data(data.obj)
#mkReport(data.obj,"myExampleReport")

### documents myExampleReport.tex and myExampleReport.pdf will be created
```

mRNA *mRNA data expression*

Description

mRNA data expression in log2-intensity units

Usage

```
data(mRNA)
```

Format

The format is: num [1:18900, 1:12] 7.06 3.38 3.23 8.41 4.63 ... - attr(*, "dimnames")=List of 2 ..$: chr [1:18900] "1/2-SBSRNA4" "A1BG" "A1BG-AS1" "A1CF" ... ..$: chr [1:12] "Control_1" "Control_2" "Control_3" "Case_1" ...

Source

Modified from: *Affò S, Dominguez M, Lozano JJ, Sancho-Bru P et al. Transcriptome analysis identifies TNF superfamily receptors as potential therapeutic targets in alcoholic hepatitis. Gut 2013 Mar;62(3):452-60. PMID: 22637703*

References

Affò S, Dominguez M, Lozano JJ, Sancho-Bru P et al. Transcriptome analysis identifies TNF superfamily receptors as potential therapeutic targets in alcoholic hepatitis. Gut 2013 Mar;62(3):452-60. PMID: 22637703

Examples

```
data(mRNA)
head(mRNA)
```

---

openCytoscape *Open cytoscape session with network loaded*

Description

Open cytoscape session with network loaded

Usage

```
openCytoscape(obj = NULL, pval.cutoff = 0.05, dat.sum =
                 obj@info[["dat.sum"]], file = NULL, cytoscape.folder =
                 "/home/mvila/Cytoscape_v2.8.3", sub.miRNA = NULL,
                 sub.mRNA = NULL, add.other = NULL, expand = FALSE)
```

Arguments

obj	corObject
pval.cutoff	minimum corrected p.value to take
dat.sum	minimum ocurrences across databases
file	file of a ".sif" network
cytoscape.folder	path where "cytoscape.jar" file is located
sub.miRNA	character vector with the restricted miRNA
sub.mRNA	character vector with the restricted mRNA
add.other	other
expand	expand the network

Details

Use a corObject with p.cutoff or file. If using directly from a corObject, a default "network_default.sif" file will be created in the working directory.

References

http://cytoscape.org/

See Also

writeSif

Examples

```
##openCytoscape(data.obj)
```

---

pearson *Pearson correlation with C++ code*

Description

The function correlates miRNA and mRNA expression from a corObject and fills the cor and pval slots.

Usage

```
pearson(obj, method = "pearson", subset.miRNA = obj@sig.miRNA, subset.mRNA = obj@sig.mRNA,
  common = NULL, d.influences = FALSE, alternative = "two.sided")
```

Arguments

obj	a corObject
method	method used for computing correlation: "pearson" or "spearman".
subset.miRNA	Optional, character vector with the names of the miRNAs to correlate.
subset.mRNA	Optional, character vector with the names of the mRNAs to correlate.
common	Optional, character vector with the names of the samples to correlate (the samples must appear in both miRNA and mRNA datasets.)
d.influences	Compute if there are any influential samples: TRUE or FALSE (default).
alternative	specification of the alternative hypothesis: "two-sided", "less" or "greater".

Details

Slow version of the correlation function. Works on large datasets but is much more slow. Moreover, Kendall correlation is specially slow. Use always correlation function if it is possible.

If TRUE, a 3-dimension matrix is added to the info slot, labeled "influenting.sample". First dimension: miRNA names; second dimension: mRNA names; third dimension: sample names; fill: Cook's Distance for a specific sample in a specific miRNA-mRNA linear regression (defined by the dimension label-names).

Value

corObject the slots "cor" and "pval" filled. Optionally, a matrix named "influenting.sample" is added to the info slot.

Note

This function can take a long time to complete when is applied to large datasets.

See Also

corObject-class,cor, correlation.alternative, cooks.distance

Examples

```
#data.obj<-correlation(data.obj, method = "pearson", alternative="less")
```

---

pheno.miRNA *Phenotypical miRNA information*

Description

Phenotypical miRNA information

Usage

```
data(pheno.miRNA)
```

Format

A data frame with 12 observations on the following 2 variables.

`group` a factor with levels `H` (Healthy) `D` (Disease)

`DvH` a numeric vector

Source

Modified from: *Affò S, Dominguez M, Lozano JJ, Sancho-Bru P et al. Transcriptome analysis identifies TNF superfamily receptors as potential therapeutic targets in alcoholic hepatitis. Gut 2013 Mar;62(3):452-60. PMID: 22637703*

References

Affò S, Dominguez M, Lozano JJ, Sancho-Bru P et al. Transcriptome analysis identifies TNF superfamily receptors as potential therapeutic targets in alcoholic hepatitis. Gut 2013 Mar;62(3):452-60. PMID: 22637703

Examples

```
data(pheno.miRNA)
pheno.miRNA
```

---

pheno.mRNA *Phenotypical mRNA information*

Description

Phenotypical mRNA information

Usage

```
data(pheno.mRNA)
```

Format

A data frame with 12 observations on the following 2 variables.

`group` a factor with levels `H` (Healthy) `D` (Disease)

`DvH` a numeric vector

Source

Modified from: *Affò S, Dominguez M, Lozano JJ, Sancho-Bru P et al. Transcriptome analysis identifies TNF superfamily receptors as potential therapeutic targets in alcoholic hepatitis. Gut 2013 Mar;62(3):452-60. PMID: 22637703*

References

Affò S, Dominguez M, Lozano JJ, Sancho-Bru P et al. Transcriptome analysis identifies TNF superfamily receptors as potential therapeutic targets in alcoholic hepatitis. Gut 2013 Mar;62(3):452-60. PMID: 22637703

Examples

```
data(pheno.mRNA)
pheno.mRNA
```

`plot3d` *PCA plot in 3D*

Description

PCA plot in 3D

Usage

```
plot3d(obj, subset, col.color = 1, angle = 45, colors = c("violet", "turquoise"), ...)
```

Arguments

`obj`	a `corObject`
`subset`	"miRNA" or "mRNA"
`col.color`	number or name of the column in the pheno slot that will be used
`angle`	angle orientation
`colors`	colors
`...`	further arguments to be passed

Value

A 3d pca plot.

Examples

```
data(data.obj)
plot3d(data.obj,"mRNA")
```

plotCircos *Circos plot*

Description

Do a standard circos plot

Usage

```
plotCircos(obj, pval.cutoff = 0.05, dat.sum = obj@info[["dat.sum"]],
 n = NULL, sub.miRNA = NULL, sub.mRNA = NULL)
```

Arguments

obj	a corObject whith a cytofile slot
pval.cutoff	maximum corrected *p* value allowed
dat.sum	numeric, minimum concordance across databases
n	numeric, limit to the first "n" pairs (sorted by corrected *p* value)
sub.miRNA	(optional) character vector, include only pairs containing these miRNAs.
sub.mRNA	(optional) character vector, include only pairs containing these mRNAs.

Value

a plot

References

www.circos.ca

http://cran.r-project.org/web/packages/circlize/index.html

See Also

plot

Examples

```
data(data.obj)
plotCircos(data.obj, n="50")
```

plotCordist *Plot distances/correlation between miRNA or mRNA samples*

Description

Plot distances/correlation between miRNA or mRNA samples

Usage

```
plotCordist(obj, subset, type = "cor", method.cor = "pearson",
 method.dist = "euclidean", ...)
```

Arguments

obj	a corObject.
subset	"miRNA" or "mRNA"
type	"cor" (correlation) or "dist" (distance).
method.cor	method used for computing correlation: "pearson" or "spearman".
method.dist	method used for computing distance from dist function: This must be one of '"euclidean"', '"maximum"', '"manhattan"', '"canberra"', '"binary"' or '"minkowski"'. Any unambiguous substring can be given.
...	further arguments.

Value

A plot.

References

http://www.phaget4.org/R/image_matrix.html

See Also

plot3d, plotPca, dist

Examples

```
data(data.obj)
plotCordist(data.obj, "miRNA", type="cor")
plotCordist(data.obj, "mRNA", type="dist")
```

`plotCorrelation` *Plot correlations*

Description

Plot the correlation of a miRNA and mRNA and/or the diagnostic plots of the linear regression.

Usage

```
plotCorrelation(obj, miRNA, mRNA, type = "cor", samples = "all",
  col.color = 1, i.legend = !is.na(col.color), pos.legend = "topright",
  sample.names = FALSE, pos.sample.names = 1, cex.main = 1.35,
  alternative = "two.sided", colors = c("turquoise", "violet"))
```

Arguments

`obj`	corObject
`miRNA`	character, miRNA selected
`mRNA`	character, mRNA selected
`type`	"cor" or "residuals"
`samples`	colunm name of the pheno.miRNA where to look for the grouping factor. The column must contain character names.
`col.color`	name of the column in the pheno slot used for coloring the samples.
`i.legend`	TRUE or FALSE. Plot legend
`pos.legend`	set the legend position from `legend` '"bottomright"', '"bottom"', '"bottomleft"', '"left"', '"topleft"', '"top"', '"topright"', '"right"' and '"center"'.
`sample.names`	TRUE or FALSE. Plot text of sample names.
`pos.sample.names`	Position of the text for sample names from `text`. Values of '1', '2', '3' and '4', respectively indicate positions below, to the left of, above and to the right of the specified coordinates.
`cex.main`	`cex` parameter for the main title.
`alternative`	alternative hypotheses used for computing the *p* value of the correlation. One of `"two.sided"`, `"less"`,`"greater"`.
`colors`	In case of a two-level factor grouping, colors to be used.

Value

A correlation plot.

Examples

```
data(data.obj)
plotCorrelation(data.obj, miRNA="hsa-miR-4423-3p", mRNA="PRR15L",
 type="cor", col.color="group", sample.names=TRUE)
plotCorrelation(data.obj, miRNA="hsa-miR-4423-3p", mRNA="PRR15L",
 type="residuals")
```

`plotDensity` *Plot miRNA or mRNA density*

Description

Plot miRNA or mRNA density, groping as you want

Usage

```
plotDensity(obj, subset, col.color=1, colors=c("turquoise", "violet"))
```

Arguments

`obj`	a `corObject`.
`subset`	`"miRNA"` or `"mRNA"`.
`col.color`	name of the column in the pheno slot used for coloring the samples.
`colors`	In case of a two-level factor grouping, colors to be used.

Value

A density plot

Examples

```
data(data.obj)
plotDensity(data.obj, "miRNA")
plotDensity(data.obj, "mRNA")
```

plotGO *Plot GO enriched terms*

Description

From RamiGO package, plots the hierarchy of the significant GO terms

Usage

```
plotGO(obj, type, ontology, fdr = 0.05, filename = "GO_tree_default")
```

Arguments

obj	a corObject.
type	"GO" or "KEGG".
ontology	For GO terms, one of "BP", "CC" or "MF". For KEGG terms, use "KEGG".
fdr	FDR cutoff.
filename	name of the TIFF figure.

Value

A file with the GO terms plotted.

Examples

```
### do not run
# data(data.obj)
# plotGO(data.obj,"GO","BP", fdr = 0.05, filename= "GO_example")

### this will create the file "GO_example.png"
```

plotHeatmap *Plot heatmaps*

Description

Plot heatmaps of the top "n" miRNA, mRNA o both miRNA-mRNA pairs.

Usage

```
plotHeatmap(obj, class, n = 50, col.color = 1, min.exp = NULL)
```

Arguments

obj	a corObject
class	"miRNA", "mRNA" or "both".
n	number of items to plot.
col.color	phenotype column name or number which contains that will be used to label the samples between groups.
min.exp	(optional) minimum mean expression of the items to be plotted

Value

A plot.

See Also

plotCorrelation

Examples

```
data(data.obj)
plotHeatmap(data.obj, "miRNA", n=100)
```

---

plotNetwork *Plot a network*

Description

Plot the network of the selected miRNA-mRNa interactions, with selected features

Usage

```
plotNetwork(obj, pval.cutoff = 0.05, sub.miRNA = NULL,
 sub.mRNA = NULL, names = TRUE, dat.sum = obj@info[["dat.sum"]],
 add.other = NULL, vertex.cex = NULL, n = NULL, node.size = 1.5)
```

Arguments

obj	a corObject.
pval.cutoff	*p* value cutoff.
sub.miRNA	Optional, character vector with the names of the miRNAs to correlate.
sub.mRNA	Optional, character vector with the names of the miRNAs to correlate.
names	Plot the names of the miRNAs and mRNAs. TRUE or FALSE.
dat.sum	minimum occurrences between databases.
add.other	Optional, character vector: name of the dataframe containing additional interactions (usually mRNA-mRNA interactions) that will also be displayed.

vertex.cex	Optional, character vector: name of the dataframe containing the relative size for each node in the network.
n	maximum number of interactions.
node.size	Size of the node

Details

The colurs are representative of the interactions.

Value

A network Plot

Examples

```
data(data.obj)
data(interact.table)
plotNetwork(data.obj, pval.cutoff=0.01, dat.sum=1,
 vertex.cex="interact.table", names=FALSE)
```

---

plotPca *PCA with miRNA or mRNA data*

Description

PCA with miRNA or mRNA data

Usage

```
plotPca(obj, subset, col.color = 1, colors = c("turquoise", "violet"), pos.leg="topleft", ...)
```

Arguments

obj	a corObject.
subset	"miRNA" or "mRNA".
col.color	name of the column in the pheno slot used for coloring the samples.
colors	In case of a two-level factor grouping, colors to be used.
pos.leg	legend position.
...	further arguments.

Value

A pca plot

Examples

```
data(data.obj)
plotPca(data.obj,"miRNA")
plotPca(data.obj,"mRNA")
```

`plotVolcano` *Volcano plot*

Description

Plots a Volcano plot from the differential expression analysis.

Usage

```
plotVolcano(obj, subset, FC1 = 1.5, FC2 = 2, FDR = 0.05, cex = 1)
```

Arguments

`obj`	a "corObject"
`subset`	"miRNA" or "mRNA"
`FC1`	first FoldChange cutoff
`FC2`	second FoldChange cutoff
`FDR`	significance cutoff (FDR)
`cex`	cex value

Value

A volcano plot.

Examples

```
data(data.obj)
plotVolcano(data.obj,"miRNA")
plotVolcano(data.obj,"mRNA")
```

removeSamp *Remove samples or miRNA/mRNA*

Description

Remove samples or miRNA/mRNA from a corObject.

Usage

```
removeSamp(obj, dataset, samples = NA, genes = NA, keep=FALSE)
```

Arguments

obj	corObject
dataset	"miRNA" or "mRNA"
samples	colnames of the samples to be removed.
genes	rownames of the genes (mRNA or miRNA) to be removed.
keep	TRUE (keep given colnames/rownames) or FALSE. By default, FALSE.

Details

Genes are removed from miRNAdat/mRNAdat slots. Samples are removed from both pheno.miRNA/pheno.mRNA and miRNAdat/mRNAdat slots.

Value

corObject	with the selected samples removed

Examples

```
data(data.obj)

dim(data.obj@dat.miRNA)
data.obj<-removeSamp(data.obj,"miRNA",samples="D_3",genes="hsa-miR-200c")
dim(data.obj@dat.miRNA)

colnames(data.obj@dat.mRNA)
data.obj<-removeSamp(data.obj,"mRNA",samples=c("D_1","D_2"),keep=TRUE)
colnames(data.obj@dat.mRNA)
```

selSubsetCor *Show relevant miRNA-mRNA interactions*

Description

Show differentially expressed miRNAs or mRNAs, according to your criteria

Usage

```
selSubsetCor(obj, pval.cutoff = 1, dat.sum = 0, sub.miRNA = NULL,
 sub.mRNA = NULL)
```

Arguments

obj	a corObject
pval.cutoff	minimum adj.pval to consider
dat.sum	minimum number of concurrences across databases
sub.miRNA	optional character vector, limit to these miRNAs
sub.mRNA	optional character vector, limit to these mRNAs

Value

A data.frame with the selected miRNA-mRNA pairs.

Examples

```
data(data.obj)
selSubsetCor(data.obj, pval.cutoff=0.05, dat.sum=2)
```

selSubsetExprs *Show differentially expressed miRNAs or mRNAs*

Description

Show differentially expressed miRNAs or mRNAs, according to your criteria

Usage

```
selSubsetExprs(obj, dataset, FC = NA, logratio = foldchange2logratio(FC),
 pval = NA, adj.pval = NA, min.meanExp = NA, up = FALSE, dw = FALSE)
```

Arguments

obj	a corObject
dataset	"miRNA" or "mRNA".
FC	minimum absolute FoldChange cutoff
logratio	minimum absolute logratio cutoff
pval	*p* value cutoff
adj.pval	adjusted *p* value cutoff
min.meanExp	minimum mean expression cutoff
up	TRUE or FALSE. Select only upregulated miRNAs or mRNAs.
dw	TRUE or FALSE. Select only upregulated miRNAs or mRNAs.

Value

A data.frame with the selected differentially expressed miRNAs or mRNAs and their characteristics.

Examples

```
data(data.obj)
selSubsetExprs(data.obj, "miRNA", adj.pval=0.05, FC=1.5)
selSubsetExprs(data.obj, "mRNA", adj.pval=0.05, up=TRUE)
```

---

summary.corObject *Brief report of a* corObject

Description

Tests if a corObject is valid and prints the information about the tests performed.

Usage

```
## S3 method for class corObject
summary(object, ...)
```

Arguments

object	a corObject.
...	other.

Value

Simple version of printing of the report.

See Also

mkReport

Examples

```
data(data.obj)
summary(data.obj)
```

---

topTable *Print or plot the top connected miRNA/mRNA*

Description

Print or plot the top connected miRNA/mRNA

Usage

```
topTable(obj, class, pval.cutoff = 0.05, dat.sum =
                 obj@info[["dat.sum"]], plot = FALSE, names = FALSE, n = NULL,
remove.names = FALSE)
```

Arguments

obj	corObject
class	"miRNA" or "mRNA".
pval.cutoff	*p* value to cutoff.
dat.sum	number of minimum ocurrences across databases.
plot	FALSE or TRUE.
names	FALSE or TRUE. Apart from the frequence, add the names of the hits. (In this case the object returned is a data.frame).
n	maximum number of pairs to consider.
remove.names	in case of a plot, omit x-axis labels.

Details

If plot=FALSE then the table is displayed, if plot=TRUE the table is not displayed and a barplot is plotted. If names=FALSE a vector is returned, if names=TRUE, a data.frame is returned.

Value

A data.frame

See Also

plotHeatmap

Examples

```
data(data.obj)

# get the names
topTable(data.obj, "miRNA", names=TRUE, plot=FALSE)
topTable(data.obj, "mRNA", names=TRUE, plot=FALSE)

# plot
topTable(data.obj, "miRNA", names=TRUE, plot=TRUE)
```

`writeCsv` *Write a csv file*

Description

Export the `cytofile` slot to a a csv file.

Usage

```
writeCsv(obj, name, pval.cutoff = 1, dat.sum =
                 obj@info[["dat.sum"]], pval = "adj.pval")
```

Arguments

`obj`	a `corObject`.
`name`	the name of the file to write.
`pval.cutoff`	minimum corrected *p* value to take.
`dat.sum`	number of minimum ocurrences across databases.
`pval`	name of the p.value column to select.

Value

A csv file.

See Also

`writeExcel`

Examples

```
## do not run
#data(data.obj)
#writeCsv(data.obj,"results_csv.csv")
```

writeExcel *Write an Excel file*

Description

Export the cytofile slot to an Excel 2003 file.

Usage

```
writeExcel(obj, name, pval.cutoff = 0.05, pval =
                  "adj.pval")
```

Arguments

obj	a corObject.
name	the name of the file to write.
pval.cutoff	minimum corrected *p* value to take.
pval	name of the column where to take the *p* values.

Details

It writes an excel document with the selected pairs.

Value

An excel file.

See Also

writeCsv

Examples

```
## do not run
#data(data.obj)
#writeCsv(data.obj,"results_csv.csv")
```

`writeSif` *Write a SIF file*

Description

Export the network (from `cytofile` slot) to a SIF file.

Usage

```
writeSif(obj, file, pval.cutoff = 0.05, dat.sum =
                 obj@info[["dat.sum"]], add.other = NULL, sub.miRNA =
                 NULL, sub.mRNA = NULL, expand = FALSE, vertex.cex =
                 "interact.table")
```

Arguments

`obj`	a `corObject`.
`file`	file to write.
`pval.cutoff`	minimum corrected *p* value to take.
`dat.sum`	number of minimum ocurrences across databases
`add.other`	a character vector. Name of the dataframe that contains extra interactions (usually mRNA-mRNA interactions) that will be added to the network.
`sub.miRNA`	character vector. Restrict to these miRNAs.
`sub.mRNA`	character vector. Restrict to these mRNAs.
`expand`	expand with another table. TRUE or FALSE.
`vertex.cex`	table to use to expand

Value

A sif file, plus a node attributes file (*the node attributes still in preparation)

References

www.cytoscape.org

See Also

`writeCsv`, `writeExcel`, `openCytoscape`

Examples

```
## do not run
#data(data.obj)
#writeSif(data.obj,"results_sif")
```

Index

∗Topic **Cook distance**
addCorrelation.R, 4
∗Topic **GO**
GOanalysis, 18
plotGO, 30
∗Topic **KEGG**
GOanalysis, 18
∗Topic **LaTeX**
mkReport, 20
∗Topic **Reactome**
GOanalysis, 18
∗Topic **Type II error**
combinePval, 13
correctPval, 16
∗Topic **add**
addDatabase, 6
addFoldchanges, 8
addNet, 9
∗Topic **barplot**
topTable, 37
∗Topic **boxplot**
boxplotCorrelation, 12
boxplotSamples, 13
∗Topic **circos**
plotCircos, 26
∗Topic **classes**
corObject-class, 15
∗Topic **correlation**
addCorrelation, 3
addCorrelation.R, 4
boxplotCorrelation, 12
pearson, 22
plotCorrelation, 28
∗Topic **csv**
writeCsv, 38
∗Topic **cytoscape**
openCytoscape, 21
∗Topic **database**
addDatabase, 6
∗Topic **datasets**
data.obj, 17
miRNA, 19
mRNA, 21
pheno.miRNA, 23
pheno.mRNA, 24
∗Topic **density**
plotDensity, 29
∗Topic **differential expression**
addFoldchanges, 8
addScore, 10
∗Topic **differential**
addDiffexp, 7
∗Topic **distance**
plotCordist, 27
∗Topic **enrichment**
GOanalysis, 18
∗Topic **excel**
writeExcel, 39
∗Topic **expression**
selSubsetCor, 35
selSubsetExprs, 35
∗Topic **external file**
mkReport, 20
plotGO, 30
writeCsv, 38
writeExcel, 39
writeSif, 40
∗Topic **heatmap**
plotHeatmap, 30
∗Topic **meta-analysis**
combinePval, 13
∗Topic **miRNA**
addDatabase, 6
∗Topic **network**
plotNetwork, 31
∗Topic **net**
addDatabase, 6
addFoldchanges, 8

addNet, 9
∗Topic **output**
writeCsv, 38
writeExcel, 39
∗Topic **p.value**
combinePval, 13
∗Topic **package**
information-package, 2
∗Topic **pca**
plot3d, 25
plotPca, 32
∗Topic **pearson**
addCorrelation.R, 4
∗Topic **plot**
boxplotCorrelation, 12
boxplotSamples, 13
plot3d, 25
plotCircos, 26
plotCordist, 27
plotCorrelation, 28
plotDensity, 29
plotGO, 30
plotHeatmap, 30
plotNetwork, 31
plotPca, 32
plotVolcano, 33
topTable, 37
∗Topic **regression**
boxplotCorrelation, 12
plotCorrelation, 28
∗Topic **remove**
removeSamp, 34
∗Topic **report**
mkReport, 20
∗Topic **residuals**
plotCorrelation, 28
∗Topic **samples**
plotCordist, 27
∗Topic **score**
addScore, 10
∗Topic **selection**
selSubsetCor, 35
selSubsetExprs, 35
∗Topic **subset**
selSubsetCor, 35
selSubsetExprs, 35
∗Topic **volcano plot**
plotVolcano, 33
∗Topic **write**
openCytoscape, 21
writeCsv, 38
writeExcel, 39
writeSif, 40

addCorrelation, 3, *5*
addCorrelation.R, 4, *4*
addDatabase, 6
addDiffexp, 7, *10, 11*
addFoldchanges, 8, *10*
addNet, 9
addScore, 10
addSig, 11

boxplotCorrelation, 12
boxplotSamples, 13

circlize, *3*
combinePval, 13, *16*
cor, *4, 5*, *23*
corObject-class, 15
correctPval, 16

data.obj, 17
DESeq, *3*

GOanalysis, 18
GOstats, *3*
gplots, *3*
gtools, *3*

Hmisc, *3*

information *(*information-package*)*, 2
information-package, 2

limma, *3*

methods, *3*
miRData, *3*
miRNA, 19
mkReport, 20, *36*
mRNA, 21

network, *3*

openCytoscape, 21, *40*

p.adjust, *16*
pearson, 22

pheno.miRNA, 23
pheno.mRNA, 24
plot3d, 25, *27*
plotCircos, 26
plotCordist, 27
plotCorrelation, *12*, 28, *31*
plotDensity, 29
plotGO, 30
plotHeatmap, 30, *37*
plotNetwork, 31
plotPca, *27*, 32
plotVolcano, 33

RamiGO, *3*
RankProd, *3*
ReactomePA, *3*
removeSamp, 34

scatterplot3d, *3*
selSubsetCor, 35
selSubsetExprs, 35
summary *(*summary.corObject*)*, 36
summary.corObject, 36

text, *28*
topTable, 37

VennDiagram, *3*

writeCsv, 38, *39, 40*
writeExcel, *38*, 39, *40*
writeSif, *22*, 40
WriteXLS, *3*

xtable, *3*