Package 'miRNApath'

February 2, 2026

Type Package

Title miRNApath: Pathway Enrichment for miRNA Expression Data

Version 1.71.0

Author James M. Ward <jmw86069@gmail.com> with contributions
from Yunling Shi, Cindy Richards, John P. Cogswell

Maintainer James M. Ward <jmw86069@gmail.com>

Description This package provides pathway enrichment techniques
for miRNA expression data. Specifically, the set of
methods handles the many-to-many relationship between miRNAs
and the multiple genes they are predicted to target (and thus
affect.) It also handles the gene-to-pathway relationships
separately. Both steps are designed to preserve the additive
effects of miRNAs on genes, many miRNAs affecting one gene,
one miRNA affecting multiple genes, or many miRNAs affecting
many genes.

License LGPL-2.1

biocViews Annotation, Pathways, DifferentialExpression,
NetworkEnrichment, miRNA

Depends methods, R(>= 2.7.0)

LazyLoad Yes

git_url https://git.bioconductor.org/packages/miRNApath

git_branch devel

git_last_commit 111d188

git_last_commit_date 2025-10-29

Repository Bioconductor 3.23

Date/Publication 2026-02-01

Contents

---

miRNApath-package   *miRNApath: Pathway Enrichment for miRNA Expression Data*

Description

This package provides methods for assessing the statistical over-representation of miRNA effects on gene sets, using supplied miRNA-to-gene associations. Because these associations are notably many-to-many (one miRNA to many genes; one gene affected by many miRNAs) the assessment is complex and warrants perhaps different approaches than are classically performed on differential gene expression datasets.

Details

Package:	miRNApath
Type:	Package
Version:	1.0
Date:	2008-04-02
License:	LGL-2.1, see COPYING.LIB

Author(s)

James M. Ward

Maintainer: James M. Ward <jmw86069@gmail.com>

References

John Cogswell (2008) Identification of miRNA changes in Alzheimer's disease brain and CSF yields putative biomarkers and insights into disease pathways, Journal of Alzheimer's Disease 14, 27-41.

See Also

loadmirnapath, filtermirnapath, loadmirnatogene, loadmirnapathways, runEnrichment

Examples

```
## Not run:
## Start with miRNA data from this package
data(mirnaobj);

## Write a file as example of required input
write.table(mirnaobj@mirnaTable, file = "mirnaTable.txt",
    quote = FALSE, row.names = FALSE, col.names = TRUE, na = "",
    sep = "\t");

## Now essentially load it back, but assign column headers
mirnaobj <- loadmirnapath( mirnafile = "mirnaTable.txt",
    pvaluecol = "P-value", groupcol = "GROUP",
    mirnacol = "miRNA Name", assayidcol = "ASSAYID" );

## Start with miRNA data from this package
data(mirnaobj);

## Write a file as example of required input
write.table(mirnaobj@mirnaGene, file = "mirnaGene.txt",
    quote = FALSE, row.names = FALSE, col.names = TRUE, na = "",
    sep = "\t");

## Load the miRNA to gene associations
mirnaobj <- loadmirnatogene( mirnafile = "mirnaGene.txt",
    mirnaobj = mirnaobj, mirnacol = "miRNA Name",
    genecol = "Entrez Gene ID",
    columns = c(assayidcol = "ASSAYID") );

## Write a file as example of required input
write.table(mirnaobj@mirnaPathways, file = "mirnaPathways.txt",
    quote = FALSE, row.names = FALSE, col.names = TRUE, na = "",
    sep = "\t");

## Load the gene to pathway associations
mirnaobj <- loadmirnapathways( mirnaobj = mirnaobj,
    pathwayfile = "mirnaPathways.txt",
    pathwaycol = "Pathway Name", genecol = "Entrez Gene ID");

## Annotate hits by filtering by P-value 0.05
mirnaobj <- filtermirnapath( mirnaobj, pvalue = 0.05,
    expression = NA, foldchange = NA );

## Now run enrichment test
mirnaobj <- runEnrichment( mirnaobj=mirnaobj, Composite=TRUE,
  groups=NULL, permutations=0 );

## Print out a summary table of significant results
finaltable <- mirnaTable( mirnaobj, groups=NULL, format="Tall",
    Significance=0.1, pvalueTypes=c("pvalues") );
finaltable[1:4,];
```

```
## Example which calls heatmap function on the resulting data
widetable <- mirnaTable( mirnaobj, groups=NULL, format="Wide",
    Significance=0.1, na.char=NA, pvalueTypes=c("pvalues") );
## Assign 1 to NA values, assuming they're all equally
## non-significant
widetable[is.na(widetable)] <- 1;

## Display a heatmap of the result across sample groups
pathwaycol <- mirnaobj@columns["pathwaycol"];
pathwayidcol <- mirnaobj@columns["pathwayidcol"];
rownames(widetable) <- apply(widetable[,c(pathwaycol,
    pathwayidcol)], 1, function(i)paste(i, collapse="-"));
wt <- as.matrix(widetable[3:dim(widetable)[2]], mode="numeric")
heatmap(wt, scale="col");

## Show results where pathways are shared in four or more
## sample groups
pathwaySubset <- apply(wt, 1, function(i)
{
    length(i[i < 1]) >= 4;
} )
heatmap(wt[pathwaySubset,], scale="row");

## End(Not run)
```

---

checkColumns   *Check data.frame for Required Columns*

Description

Internal function designed to assist checking for required columns in a data.frame.

Usage

```
checkColumns(data, mandatory=NA, numeric=NA)
```

Arguments

data   data.frame

mandatory   list of required columns

numeric   list of columns expected to be numeric

Value

boolean, or stop() with columns not meeting the criteria

Author(s)

James M. Ward <jmw86069@gmail.com>

Examples

```
## No examples currently
```

---

convertFoldChange *Converts Fold Change to Consistent Format*

Description

The script tries to determine if the fold changes are in an expected format, and if not, does its best to convert correctly. Internal function.

Usage

```
convertFoldChange(foldchanges, style)
```

Arguments

foldchanges   numerical values with fold changes

style   type of fold change to return.

Details

Allowed formats are: "log directional" (contains continuous numbers positive and negative, including numbers between -1 and 1), "directional", includes positive and negative values, but no numbers between -1 and 1, and "fractional", includes values greater than zero, especially those less than one.

Value

numerical values in the defined format

Author(s)

James M. Ward <jmw86069@gmail.com>

Examples

```
## No examples currently.
```

---

filtermirnapath   *Filter miRNApath data*

Description

This method filters the miRNApath data to denote hits versus non-hits, the required distinction for running the enrichment algorithm. Data is expected to have been loaded by the loadmirnapath method.

Usage

```
filtermirnapath(mirnaobj, pvalue=NA, expression=NA, foldchange=NA)
```

Arguments

mirnaobj	An object of type mirnapath containing data resulting from the loadmirnapath method.
pvalue	If a p-value column has been defined in the mirnapath object, this value is used to define a subset of entries within the dataset which will be denoted as hits.
expression	If an expression column has been defined in the mirnapath object, this value will be used to define entries with expression above this expression level as hits.
foldchange	If a fold change column has been defined in the mirnapath object, this value is used to require hits to have a fold change greater than or equal to this value. The fold change is evaluated in both the positive and the negative, such that a foldchange=2 will allow foldchange=2 and foldchange=-2.

Details

This method takes a mirnapath object and assigns a flag for hits and non-hits, depending upon what filter criteria was provided. If multiple criteria are provided, they will all be collectively applied such that all criteria must be fulfilled. To that end, multiple calls to this function on the same mirnapath object should successively shrink the list of hits dependent upon the given criteria.

Value

Object of type mirnapath. The state of the object will reflect that the data has been filtered.

Note

The method attempts to convert fold change columns appropriately so that filtering by 2 will properly mark entries greater than 2, less than -2, or less than 0.5, as the case may be.

Author(s)

James M. Ward <jmw86069@gmail.com>

References

John Cogswell (2008) Identification of miRNA changes in Alzheimer's disease brain and CSF yields putative biomarkers and insights into disease pathways, Journal of Alzheimer's Disease 14, 27-41.

See Also

loadmirnapath, filtermirnapath, loadmirnatogene, loadmirnapathways

Examples

```
## Load miRNA expression data from AD miRNA paper
## This data contains miRNA expression data,
data(mirnaobj);

## Display the state, which should generally be "unfiltered"
## at this point
mirnaobj@state;

## Display summary information about the object
mirnaobj;

## Annotate hits by filtering by P-value 0.05
mirnaobj <- filtermirnapath( mirnaobj, pvalue = 0.05,
    expression = NA, foldchange = NA );

## Display summary, noting the state is "filtered"
mirnaobj;
```

---

loadmirnapath   *Load miRNApath Data*

Description

This method loads data from a tab-delimited flatfile into an object of type mirnapath to be used for further miRNA analysis.

Usage

```
loadmirnapath(mirnafile="mirna_input.txt", mirnacol="miRNA Name",
    assayidcol="ASSAYID", groupcol="GROUP",
    filterflagcol="FILTERFLAG", expressioncol=NA,
    foldchangecol=NA, pvaluecol=NA)
```

Arguments

mirnafile	The tab-delimited miRNA results file to be loaded. The file is expected to be in tall-skinny format.
mirnacol	The name of the column header which contains the miRNA names being assayed.
assayidcol	The name of the column containing values which distinguish different assays for the same miRNA.
groupcol	The (optional) name of the column which contains sample group information. Enrichment is run separately for each sample group, defining a unique universe for the basis of the enrichment.
filterflagcol	The column header which does or will contain a flag distinguishing hits from non-hits. This column is typically not supplied and is created during the filtermirnapath step.
expressioncol	The (optional) column header for values containing the expression abundances of the miRNAs assayed.
foldchangecol	The (optional) column header for values containing the fold changes of the miRNAs assayed.
pvaluecol	The (pvaluecol) column header for values containing the P-values of the miRNAs assayed.

Details

This method is the primary means for loading data into the miRNApath package.

Data is not assumed to have any particular numerical values, however the basic column types are typically used: expression abundance, fold change, and P-value. Should one or more columns be specified and available, it will be available for filtering later on with filtermirnapath.

The group column assumes there is one column containing all sample group information.

The assayid column is used to distinguish multiple assays for the same miRNA, such as different vendors, or even different preparations of the same miRNA assay.

Value

The method returns an object of type mirnapath, a list with components:

mirnaTable	data.frame containing the miRNA results data
columns	list containing the names of required column headers associated to the actual column header supplied in the dataset contained in mirnaTable. Required headers: mirnacol, assayidcol. Optional headers: groupcol, pvaluecol, foldchangecol, expressioncol, filterflagcol
groupcount	the number of groups contained in mirnaTable using the groupcol, if supplied
state	the current state of the object, using the following values in order of progress through the typical workflow: unfiltered, filtered, enriched.

Author(s)

James M. Ward <jmw86069@gmail.com>

References

John Cogswell (2008) Identification of miRNA changes in Alzheimer's disease brain and CSF yields putative biomarkers and insights into disease pathways, Journal of Alzheimer's Disease 14, 27-41.

See Also

loadmirnapath, filtermirnapath, loadmirnatogene, loadmirnapathways

Examples

```
## Start with miRNA data from this package
data(mirnaobj);

## Write a file as example of required input
write.table(mirnaobj@mirnaTable, file = "mirnaobj.txt",
    quote = FALSE, row.names = FALSE, col.names = TRUE, na = "",
    sep = "\t");

## Now essentially load it back, but assign column headers
mirnaobj <- loadmirnapath( mirnafile = "mirnaobj.txt",
    pvaluecol = "P-value", groupcol = "GROUP",
    mirnacol = "miRNA Name", assayidcol = "ASSAYID" );

## Display summary information for the resulting object
mirnaobj;
```

---

loadmirnapathways   *Load gene to pathway associations for miRNApath*

Description

This method loads associations between genes and the pathways to which they belong.

Usage

```
loadmirnapathways(mirnaobj, pathwayfile, genecol="Entrez Gene ID",
pathwaycol="PATHWAY", columns=c(), pathwayidcol=NA)
```

Arguments

mirnaobj	An object of type mirnapath containing data resulting from the loadmirnapath method.
pathwayfile	The file containing the gene to pathway associations.
genecol	The name of the column header which contains the gene names associated with pathway data.
pathwaycol	The name of the column header which contains the pathway names.

columns	The names of any additional columns in the file being read which should equate with the mirnapath object.
pathwayidcol	The (optional) column header for IDs associated with the pathway names.

Details

The data loaded is expected to have gene names which exactly match those gene names loaded by `loadmirnatogene`.

Value

The method returns an object of type mirnapath, a list with components:

mirnaTable	data.frame containing the miRNA results data
columns	list containing the names of required column headers associated to the actual column header supplied in the dataset contained in mirnaTable. Required headers: mirnacol, assayidcol. Optional headers: groupcol, pvaluecol, foldchangecol, expressioncol, filterflagcol
groupcount	the number of groups contained in mirnaTable using the groupcol, if supplied
state	the current state of the object, using the following values in order of progress through the typical workflow: unfiltered, filtered, enriched.

Author(s)

James M. Ward <jmw86069@gmail.com>

References

John Cogswell (2008) Identification of miRNA changes in Alzheimer's disease brain and CSF yields putative biomarkers and insights into disease pathways, Journal of Alzheimer's Disease 14, 27-41.

See Also

loadmirnapath, filtermirnapath, loadmirnatogene, loadmirnapathways, runEnrichment

Examples

```
## Load miRNA expression data from AD miRNA paper
## This data contains miRNA expression data,
data(mirnaobj);

## Write a file as example of required input
write.table(mirnaobj@mirnaPathways, file = "mirnaPathways.txt",
    quote = FALSE, row.names = FALSE, col.names = TRUE, na = "",
    sep = "\t");

## Load the gene to pathway associations
mirnaobj <- loadmirnapathways( mirnaobj = mirnaobj,
    pathwayfile = "mirnaPathways.txt",
```

```
      pathwaycol = "Pathway Name", genecol = "Entrez Gene ID");

## Display summary, noting the number of genes reported
mirnaobj;
```

---

loadmirnatogene   *Load miRNA to gene associations for miRNApath*

Description

This method loads associations between miRNAs to the genes they affect.

Usage

```
loadmirnatogene(mirnafile, mirnaobj, mirnacol="miRNA Name",
genecol="Entrez Gene ID", columns=NA)
```

Arguments

mirnafile	The tab-delimited miRNA results file to be loaded. The file is expected to be in tall-skinny format.
mirnaobj	An object of type mirnapath containing data resulting from the loadmirnapath method.
mirnacol	The name of the column header which contains the miRNA names being assayed. That is, the name of the column header in the file being read.
genecol	The name of the column header which contains the gene names being assayed.
columns	The names of any additional columns in the file being read which should equate with the mirnapath object.

Details

The data is expected to have miRNA names which exactly match those in the mirnaTable item of the mirnapath object. Also, the gene names are expected to match exactly with those gene names loaded by loadmirnapathways.

Value

The method returns an object of type mirnapath, a list with components:

mirnaTable	data.frame containing the miRNA results data
columns	list containing the names of required column headers associated to the actual column header supplied in the dataset contained in mirnaTable. Required headers: mirnacol, assayidcol. Optional headers: groupcol, pvaluecol, foldchangecol, expressioncol, filterflagcol
groupcount	the number of groups contained in mirnaTable using the groupcol, if supplied
state	the current state of the object, using the following values in order of progress through the typical workflow: unfiltered, filtered, enriched.

Author(s)

James M. Ward <jmw86069@gmail.com>

References

John Cogswell (2008) Identification of miRNA changes in Alzheimer's disease brain and CSF yields putative biomarkers and insights into disease pathways, Journal of Alzheimer's Disease 14, 27-41.

See Also

[loadmirnapath](#) [filtermirnapath](#) [loadmirnatogene](#) [loadmirnapathways](#)

Examples

```
## Load miRNA expression data from AD miRNA paper
## This data contains miRNA expression data,
data(mirnaobj);

## Display the state, which should generally be "unfiltered"
## at this point
mirnaobj@state;

## Display summary information about the object
mirnaobj;

## Annotate hits by filtering by P-value 0.05
mirnaobj <- filtermirnapath( mirnaobj, pvalue = 0.05,
    expression = NA, foldchange = NA );

## Write a file as example of required input
write.table(mirnaobj@mirnaGene, file = "mirnaGene.txt",
    quote = FALSE, row.names = FALSE, col.names = TRUE, na = "",
    sep = "\t");

## Load the miRNA to gene associations
mirnaobj <- loadmirnatogene( mirnafile = "mirnaGene.txt",
    mirnaobj = mirnaobj, mirnacol = "miRNA Name",
    genecol = "Entrez Gene ID",
    columns = c(assayidcol = "ASSAYID") );

## Display summary, noting the number of genes reported
mirnaobj;
```

---

mirnaobj   *mirnaobj, an S4 object of class "mirnapath"*

Description

An example miRNApath data object containing miRNA data, gene and pathway associations. The object represents the end result of the miRNApath workflow, and serves a convenient source for example data.

Usage

```
data(mirnaobj)
```

Format

The format is an S4 class `"mirnapath"` with `slotNames` as follows:

mirnaTable data.frame containing the miRNA results data, expected to contain columns with miRNA name, gene name, and ideally some column(s) for filtering hits versus background, e.g. fold change, expression abundance, P-value. Once the data is filtered (see state below) there will be a column with a flag indicating which entries are hits and which are considered background. This column is found in `mirnaobj@columns["filterflagcolumn"]` and is typically `"FILTERFLAG"`.

columns Named list of column headers used throughout the analysis. The purpose of the names is partly to retain the original headers in the mirnaTable data.frame, and partly to coordinate the names with the miRNA-gene and gene-pathway tables used later in the analysis. The recognized headers: `mirnacol`, `assayidcol`, `genecol`, `pvaluecol`, `foldchangecol`, `pathwaycol`, `pathwayidcol`, `groupcol`, `mirnagene`. See the documentation for the mirnapath object type for more details about usage.

groupcount Numerical value indicating how many sample groups are available in the data, provided for convenience.

state Character value indicating the current analysis state, with values: `"unfiltered"` if results are loaded but not yet filtered; `"filtered"` if results are loaded and hits are defined with the filterflagcol column; `"enriched"` if the data is loaded, filtered, and analyzed for enrichment. One can load mirna-gene and gene-pathway data at any point which necessitates using the `mirnaobj@mirnaGene` or `mirnaobj@mirnaPathways` object elements to determine if that data has been loaded.

mirnaGene data.frame containing associations between miRNA and genes. The data should contain one miRNA-to-gene relationship per row, and should contain only those two columns. Additional columns are maintained but ignored. Note that one can use any values in the genecol column, provided they match exactly with values found in the `mirnaobj@mirnaPathways` element (see below.) Therefore, if desired one can use transcript or gene associations, or other integration methods as desired.

mirnaPathways data.frame containing gene-pathway associations. The data should contain only one gene-to-pathway association per row of data. The data can have pathway ID values, which may facilitate comparisons to pathway databases (and may allow substantial data volume reduction if necessary.) If there is no pathwayidcol column, then one will be created using a numerical assignments of the pathway names. Note that this conversion is not sensitive to pathway sources, so care should be taken to include pathway source in the pathway name if two sources share the same pathway name. The same is true for pathway ID values, should they be purely numerical and have shared values across pathway sources.

pathwaycount   Numerical value indicating how many pathways are available in the data, provided
    for convenience.

filters   List of filters applied to the data, which may include: "P-value", "Fold change",
    and/or "Expression".

enrichment   Enrichment summary data in the form of a list of elements for each sample group
    (the sample group is the name of each element.) Each list element is itself a list with en-
    richment result data for each sample group, as independently calculated: "pvalues" - list of
    P-values named by pathway ID; "Measured pathway mirnaGenes" - total number of miRNA-
    gene-pathway combinations measured, which gives some idea of the overall coverage of path-
    ways. The general point is that miRNAs have the potential to cover many genes and path-
    ways; "Total mirnaGenes" - number of miRNA-gene combinations represented in the data;
    "Enriched pathway mirnaGenes" - number of miRNA-gene values enriched in the pathway
    tested; "Enriched by miRNA" - list of miRNAs involved in the pathway tested, with the list of
    genes in parentheses per miRNA; "Enriched by Gene" - same as previous except switching
    gene and miRNA; "Total enriched mirnaGenes" - the total number of miRNA-gene values
    involved in any pathway enrichment (significant or not.) The total values are useful when
    comparing across sample groups, looking particularly for groups with few changes or those
    with a uniquely high number of changes.

pathwayList   Named list of pathways contained in the mirnaobj@mirnaPathways object, named
    by the pathway ID values found in the pathwayidcol column. This list facilitates converting
    the data in the enrichment element to pathway names, since those values are named by the
    pathway ID to conserve memory.

Source

Journal of Alzheimers Disease 14, 27-41.

References

John Cogswell (2008) Identification of miRNA changes in Alzheimer's disease brain and CSF
yields putative biomarkers and insights into disease pathways, Journal of Alzheimer's Disease 14,
27-41.

Examples

```
## Load the data
data(mirnaobj)

## Print the default summary
mirnaobj;
```

---

mirnapath-class   *S4 class mirnapath and its "show" method*

Description

miRNApath class intended to contain miRNA data, gene and pathway associations, and ultimately
the pathway enrichment results in detail.

Objects from the Class

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

Slots

`mirnaTable`: Object of class `"data.frame"`, containing the miRNA results data, expected to contain columns with miRNA name, gene name, and ideally some column(s) for filtering hits versus background, e.g. fold change, expression abundance, P-value. Once the data is filtered (see state below) there will be a column with a flag indicating which entries are hits and which are considered background. This column is found in `mirnaobj@columns["filterflagcolumn"]` and is typically `"FILTERFLAG"`.

`columns`: Object of class `"character"`, Named list of column headers used throughout the analysis. The purpose of the names is partly to retain the original headers in the mirnaTable data.frame, and partly to coordinate the names with the miRNA-gene and gene-pathway tables used later in the analysis. The recognized headers: `mirnacol`, `assayidcol`, `genecol`, `pvaluecol`, `foldchangecol`, `pathwaycol`, `pathwayidcol`, `groupcol`, `mirnagene`. See the documentation for the mirnapath object type for more details about usage.

`groupcount`: Object of class `"numeric"`, indicating how many sample groups are available in the data, provided for convenience.

`state`: Object of class `"character"`, indicating the current analysis state, with values: `"unfiltered"` if results are loaded but not yet filtered; `"filtered"` if results are loaded and hits are defined with the filterflagcol column; `"enriched"` if the data is loaded, filtered, and analyzed for enrichment. One can load mirna-gene and gene-pathway data at any point which necessitates using the `mirnaobj@mirnaGene` or `mirnaobj@mirnaPathways` object elements to determine if that data has been loaded.

`mirnaGene`: Object of class `"data.frame"`, containing associations between miRNA and genes. The data should contain one miRNA-to-gene relationship per row, and should contain only those two columns. Additional columns are maintained but ignored. Note that one can use any values in the genecol column, provided they match exactly with values found in the `mirnaobj@mirnaPathways` element (see below.) Therefore, if desired one can use transcript or gene associations, or other integration methods as desired.

`mirnaPathways`: Object of class `"data.frame"`, containing gene-pathway associations. The data should contain only one gene-to-pathway association per row of data. The data can have pathway ID values, which may facilitate comparisons to pathway databases (and may allow substantial data volume reduction if necessary.) If there is no pathwayidcol column, then one will be created using a numerical assignments of the pathway names. Note that this conversion is not sensitive to pathway sources, so care should be taken to include pathway source in the pathway name if two sources share the same pathway name. The same is true for pathway ID values, should they be purely numerical and have shared values across pathway sources.

`pathwaycount`: Object of class `"numeric"`, Numerical value indicating how many pathways are available in the data, provided for convenience.

`filters`: Object of class `"numeric"`, List of filters applied to the data, which may include: `"P-value"`, `"Fold change"`, and/or `"Expression"`.

`enrichment`: Enrichment summary data in the form of a list of elements for each sample group (the sample group is the name of each element.) Each list element is itself a list with enrichment result data for each sample group, as independently calculated: `"pvalues"` - list of

P-values named by pathway ID; "Measured pathway mirnaGenes" - total number of miRNA-gene-pathway combinations measured, which gives some idea of the overall coverage of pathways. The general point is that miRNAs have the potential to cover many genes and pathways; "Total mirnaGenes" - number of miRNA-gene combinations represented in the data; "Enriched pathway mirnaGenes" - number of miRNA-gene values enriched in the pathway tested; "Enriched by miRNA" - list of miRNAs involved in the pathway tested, with the list of genes in parentheses per miRNA; "Enriched by Gene" - same as previous except switching gene and miRNA; "Total enriched mirnaGenes" - the total number of miRNA-gene values involved in any pathway enrichment (significant or not.) The total values are useful when comparing across sample groups, looking particularly for groups with few changes or those with a uniquely high number of changes.

pathwayList: Object of class "character", pathways contained in the mirnaobj@mirnaPathways object, named by the pathway ID values found in the pathwayidcol column. This list facilitates converting the data in the enrichment element to pathway names, since those values are named by the pathway ID to conserve memory.

Methods

show signature(object = "mirnapath"): ...

Source

Journal of Alzheimers Disease 14, 27-41.

References

John Cogswell (2008) Identification of miRNA changes in Alzheimer's disease brain and CSF yields putative biomarkers and insights into disease pathways, Journal of Alzheimer's Disease 14, 27-41.

Examples

```
library(miRNApath);

data(mirnaobj);

# the slotNames and definitions are described
#showClass(mirnaobj);

# Default "show" method describes the contents of the object
mirnaobj;
```

---

mirnaTable   *Create miRNA Enrichment Summary Table as data.frame*

Description

This function takes an miRNApath object which has been evaluated by runEnrichment(), and provides a data.frame summary.

Usage

```
mirnaTable(mirnaobj, groups=NULL, format="Tall",
    Significance=0.2, na.char=NA, pvalueTypes=c("pvalues",
    "permpvalues"), maxStringLength=NA)
```

Arguments

mirnaobj   An object of type mirnapath containing data resulting from the `loadmirnapath` method.

groups   List of groups to include in the data.frame, or NULL to include all groups in the miRNApath object.

format   This parameter tells the method to return "Tall", "SuperTall", or "Wide" data. See details below for a description of each format.

Significance A numerical value specifying the P-value cutoff to use to subset the data returned in the data.frame. To avoid subsetting the data, provide a value of 1.

na.char   Value to use for NA instead of leaving NA as-is, potentially useful for text output.

pvalueTypes   Defines which P-value columns should be returned, more useful for the Wide format which could otherwise have two sets of P-value columns if permutation adjustment were used.

maxStringLength

              Defines the maximum length per character string, after being determined by `nchar`. Strings and column headers are both truncated to this length.

Details

This function simply combines the various results from the runEnrichment method into one data.frame suitable for plotting or printing in a table. Due to potentially large data volume, the subset feature even when used liberally can substantially reduce the returned dataset size.

The `maxStringLength` value is particularly useful, often critical, for displaying a summary table in text format, since pathway names sample group names can be quite long. Although there is no default, a recommended value of 50 seems to fit the appropriate balance of being short enough to fit within a table, and yet be long enough to describe the pathway. The Wide format will contain sample group names as column headers, and a value of 50 should not in theory affect the name, except where it wouldn't be readable in a table anyway.

Value

data.frame

For Tall data, the columns contain P-values and other values useful for discriminating potential hits, the rows contain each miRNA-group combination tested which meets the P-value cutoff. The miRNAs and genes contributing to the enrichment results are concatenated to be summarized in one row and can be rather large.

For SuperTall data, the Tall table as described above is returned, except that the concatenated miRNA-gene values are separated to one row each. Every individual miRNA and gene value is

represented on its own row, which can facilitate some summary views or data filtering techniques (e.g. Excel or Spotfire.)

For Wide data, the columns contain the group names, the rows contain the pathway name, and the cells contain the P-value. Note that the column names will have the P-value column header prepended to the column name, e.g. "pvalue.GroupName".

An important note when supplying string na.char values, be sure to convert the data to a numeric matrix before calling functions such as heatmap, taking care to remove string values or convert strings to 1.0 beforehand.

Author(s)

James M. Ward <jmw86069@gmail.com>

References

John Cogswell (2008) Identification of miRNA changes in Alzheimer's disease brain and CSF yields putative biomarkers and insights into disease pathways, Journal of Alzheimer's Disease 14, 27-41.

See Also

loadmirnapath, filtermirnapath, loadmirnatogene, loadmirnapathways, runEnrichment,

Examples

```
## Start with miRNA data from this package
data(mirnaobj);

## Now run enrichment test
mirnaobj <- runEnrichment( mirnaobj=mirnaobj, Composite=TRUE,
    groups=NULL, permutations=0 );

## Print out a summary table of significant results
finaltable <- mirnaTable( mirnaobj, groups=NULL, format="Tall",
     Significance=0.1, pvalueTypes=c("pvalues") );
finaltable[1:20,];

## Example which calls heatmap function on the resulting data
widetable <- mirnaTable( mirnaobj, groups=NULL, format="Wide",
     Significance=0.1, na.char=NA, pvalueTypes=c("pvalues") );
## Assign 1 to NA values, assuming they're all equally
## non-significant
widetable[is.na(widetable)] <- 1;

## Display a heatmap of the result across sample groups
pathwaycol <- mirnaobj@columns["pathwaycol"];
pathwayidcol <- mirnaobj@columns["pathwayidcol"];
rownames(widetable) <- apply(widetable[,c(pathwaycol,
    pathwayidcol)], 1, function(i)paste(i, collapse="-"));
wt <- as.matrix(widetable[3:dim(widetable)[2]], mode="numeric")
heatmap(wt, scale="col");
```

```
## Show results where pathways are shared in four or more
## sample groups
pathwaySubset <- apply(wt, 1, function(i)
{
   length(i[i < 1]) >= 4;
} )
heatmap(wt[pathwaySubset,], scale="row");
```

---

runEnrichment   *Perform gene set enrichment analysis on a miRNApath object*

Description

This method performs a hypergeometric enrichment analysis on a miRNApath object, which should already contain miRNA data, miRNA-gene associations, gene-pathway associations, and some criteria for filtering miRNA hits from the full tested set.

Usage

```
runEnrichment(mirnaobj, Composite=TRUE, groups=NULL,
permutations=0)
```

Arguments

mirnaobj   An object of type mirnapath containing data resulting from the loadmirnapath
              method.

Composite   Defines whether the enrichment treats miRNA-gene as the enriched entity, or
              uses only the gene.

groups   List of groups to include in the analysis, although each group is analyzed inde-
              pendent from the other groups.

permutations The number of permutations to use in calculating an adusted P-value. Value of
              0 will perform no permutations.

Details

The composite flag indicates whether to treat the fully expanded miRNA-gene combinations as separate enrichment events (TRUE), or whether to treat all effects on one gene as one collective event. The latter case reverts to the classic un-ordered hypergeometric enrichment technique.

However the expansion of combinations is the current method chosen to represent the multiple predicted effects of miRNAs to one gene, and the predicted effect of one miRNA to multiple genes. The algorithm will identify statistically significantly enriched results when the combination of these effects is greater than would be anticipated by random chance.

The adjusted P-value is calculated using the rank of unadjusted P-values divided by the number of permutations minus one (such that the best rank from 1,000 permutations yields an adjusted P-value of 0.001.) The default value 0 was put in place to save time, since most adjustments resulted in stronger "hits" and weaker "non-hits" in terms of pathways enriched. Thus the results are not substantially changed, and permutation adjustment is saved for the final result set.

Value

The method returns an object of type mirnapath, a list with components:

`mirnaTable`	data.frame containing the miRNA results data
`columns`	list containing the names of required column headers associated to the actual column header supplied in the dataset contained in mirnaTable. Required headers: mirnacol, assayidcol, groupcol, filterflagcol.
`groupcount`	The number of groups contained in mirnaTable using the groupcol, if supplied
`state`	The current state of the object, in this case "enriched".
`mirnaGene`	data.frame containing associations between miRNAs and genes.
`mirnaPathways`	data.frame containing gene-pathway associations.
`pathwaycount`	Numerical value indicating how many pathways are available in the data, provided for convenience.
`filters`	List of filters applied to the data, which may include: "P-value", "Fold change", and/or "Expression".
`enrichment`	Enrichment summary data in the form of a list of elements for each sample group (the sample group is the name of each element.) Each list element is itself a list with enrichment result data for each sample group, as independently calculated: "pvalues" - list of P-values named by pathway ID. "Measured pathway mirnaGenes" - total number of miRNA-gene-pathway combinations measured, which gives some idea of the overall coverage of pathways. The general point is that miRNAs have the potential to cover many genes and pathways. "Total mirnaGenes" - number of miRNA-gene combinations represented in the data. "Enriched pathway mirnaGenes" - number of miRNA-gene values enriched in the pathway tested. "Enriched by miRNA" - list of miRNAs involved in the pathway tested, with the list of genes in parentheses per miRNA. "Enriched by Gene" - same as previous except switching gene and miRNA. "Total enriched mirnaGenes" - the total number of miRNA-gene values involved in any pathway enrichment (significant or not.) The total values are useful when comparing across sample groups, looking particularly for groups with few changes or those with a uniquely high number of changes. Lastly, with permutations > 0 "Permutation P-value" will contain the rank-adjusted P-value as described in the details section.
`pathwayList`	Named list of pathways contained in the mirnaobj\$mirnaPathways object, named by the pathway ID values found in the pathwayidcol column. This list facilitates converting the data in the enrichment element to pathway names, since those values are named by the pathway ID to conserve memory.

Author(s)

James M. Ward <jmw86069@gmail.com>

References

John Cogswell (2008) Identification of miRNA changes in Alzheimer's disease brain and CSF yields putative biomarkers and insights into disease pathways, Journal of Alzheimer's Disease 14, 27-41.

See Also

loadmirnapath, filtermirnapath, loadmirnatogene, loadmirnapathways

Examples

```
## Not run:
## Start with miRNA data from this package
data(mirnaobj);

## Now run enrichment test
mirnaobj <- runEnrichment( mirnaobj=mirnaobj, Composite=TRUE,
    groups=NULL, permutations=0 );

## End(Not run)
```

Index