Package 'affycoretools'

November 11, 2026

Date 2008-12-15

Title Functions useful for those doing repetitive analyses with Affymetrix GeneChips

Version 1.84.0

Author James W. MacDonald

Description Various wrapper functions that have been written to streamline the more common analyses that a core Biostatistician might see.

Maintainer James W. MacDonald <jmacdon@u.washington.edu>

License Artistic-2.0

Depends Biobase, methods

Imports affy, limma, GOstats, gcrma, splines, xtable, AnnotationDbi, ggplot2, gplots, oligoClasses, ReportingTools, hwriter, lattice, S4Vectors, edgeR, RSQLite, BiocGenerics, DBI, Glimma

Suggests affydata, hgfocuscdf, BiocStyle, knitr, hgu95av2.db, rgl, rmarkdown

biocViews ReportWriting, Microarray, OneChannel, GeneExpression

VignetteBuilder knitr

RoxygenNote 7.1.2

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

git_branch RELEASE_3_23

git_last_commit aeb3e29

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Contents

affycoretools 2
affycoretools-defunct 3

affystart . . . 4
annotateEset . . . 5
dataAndHeatmapPage . . . 7
doGlimma . . . 8
entrezLinks . . . 10
fixHeaderAndGo . . . 11
geneSetPage . . . 12
getCols . . . 14
getMainProbes . . . 14
gsHeatmap . . . 15
intNames . . . 16
make.cl . . . 17
makeGoGeneTable . . . 17
makeGoTable . . . 18
makeHmap . . . 19
makeImages . . . 21
makeIndices . . . 23
makeVenn . . . 24
maplot . . . 27
mirna2mrna . . . 28
outputRomer . . . 29
pca.legend . . . 32
plotDeg . . . 33
plotPCA . . . 33
runRomer . . . 35
tableFilt . . . 38
venn4Way . . . 39
vennCounts2 . . . 40
vennInLine . . . 41
vennPage . . . 42
vennSelect2 . . . 44
writeFit . . . 46

Index **48**

`affycoretools` *Make repetitive analyses of microarray and RNA-Seq data simpler with affycoretools.*

Description

The affycoretools package is primarily intended to make analyses of Affymetrix GeneChip data simpler and more straightforward. There are any number of packages designed for preprocessing or analyzing Affy data, but there are not so many that help streamline the analysis to help create useful output that can be given to collaborators.

Details

The affycoretools package is primarily intended to be used as a way to do reproducible research, where the analysis and documentation are all held in a single file, that is then processed by R to create the output data, as well as a nicely formatted pdf that documents the analysis. The affycoretools package can be used with either Sweave or knitr documents, although these days knitr is really the way to go.

In addition, affycoretools can be used with either annaffy or ReportingTools to create useful output in HTML or text format to share with your collaborators. However, ReportingTools is being actively developed and maintained, whereas annaffy is not, so the intention is to slowly convert all the functions to primarily use ReportingTools.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

---

affycoretools-defunct *Defunct functions in package* **affycoretools**

Description

Functions that have been removed from affycoretools, as they are no longer needed.

Usage

```
#These functions are defunct as of affycoretools 1.40.0
annBM
vennSelectBM
limma2biomaRt
probes2tableBM
foldFiltBM
foldFilt
vennSelectFC
hyperG2annaffy
hyperGoutput
limma2annaffy
probes2table
vennSelect
```

Details

Most of these functions have been superceded by functions intended to use ReportingTools instead of annaffy. Please see the Refactoring Affycoretools vignette for more information.

affystart *Pre-processing for Affymetrix Data*

Description

This function is designed to automatically read in all cel files in a directory, make all pre-processing QC plots and compute expression measures.

Usage

```
affystart(
  ...,
  filenames = NULL,
  groups = NULL,
  groupnames = NULL,
  plot = TRUE,
  pca = TRUE,
  squarepca = FALSE,
  plottype = "pdf",
  express = c("rma", "mas5", "gcrma"),
  addname = NULL,
  output = "txt",
  annotate = FALSE,
  ann.vec = c("SYMBOL", "GENENAME", "ENTREZID", "UNIGENE", "REFSEQ")
)
```

Arguments

...	Requires that all variables be named.
filenames	If not all cel files in a directory will be used, pass a vector of filenames.
groups	An integer vector indicating the group assignments for the PCA plot.
groupnames	A character vector with group names for PCA legend.
plot	Should density and degradation plots be made? Defaults to TRUE.
pca	Should a PCA plot be made? Defaults to TRUE.
squarepca	Should the y-axis of the PCA plot be made comparable to the x-axis? This may aid in interpretation of the PCA plot. Defaults to FALSE.
plottype	What type of plot to save. Can be "pdf","postscript", "png","jpeg", or "bmp". Defaults to "pdf". Note that "png" and "jpeg" may not be available on a given computer. See the help page for capabilities and png for more information.
express	One of either rma, mas5, gcrma. Defaults to rma. Partial matching OK.
addname	Used to append something to the name of the pca plot and the expression values output file (e.g., if function is run twice using different methods to compute expression values).

output	What format to use for the output of expression values. Currently only supports text format.
annotate	Boolean. Add annotation data to the output file?
ann.vec	A character vector of annotation data to add to the output file.

Value

Returns an ExpressionSet.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

See Also

plotHist, plotDeg, plotPCA

---

annotateEset *Method to annotate ExpressionSets automatically*

Description

This function fills the featureData slot of the ExpressionSet automatically, which is then available to downstream methods to provide annotated output. Annotating results is tedious, and can be surprisingly difficult to get right. By annotating the data automatically, we remove the tedium and add an extra layer of security since the resulting ExpressionSet will be tested for validity automatically (e.g., annotation data match up correctly with the expression data). Current choices for the annotation data are a ChipDb object (e.g., hugene10sttranscriptcluster.db) or an AffyGenePDInfo object (e.g., pd.hugene.1.0.st.v1). In the latter case, we use the parsed Affymetrix annotation csv file to get data. This is only intended for those situations where the ChipDb package is not available, and in particular is only available for those packages that contain the parsed annotation csv files (generally, Gene ST arrays, Exon ST arrays and Clariom/HTA/MTA/RTA arrays).

Usage

```
annotateEset(object, x, ...)

## S4 method for signature 'ExpressionSet,ChipDb'
annotateEset(
  object,
  x,
  columns = c("PROBEID", "ENTREZID", "SYMBOL", "GENENAME"),
  multivals = "first"
)

## S4 method for signature 'ExpressionSet,AffyGenePDInfo'
annotateEset(object, x, type = "core", ...)
```

```
## S4 method for signature 'ExpressionSet,AffyHTAPDInfo'
annotateEset(object, x, type = "core", ...)

## S4 method for signature 'ExpressionSet,AffyExonPDInfo'
annotateEset(object, x, type = "core", ...)

## S4 method for signature 'ExpressionSet,AffyExpressionPDInfo'
annotateEset(object, x, type = "core", ...)

## S4 method for signature 'ExpressionSet,character'
annotateEset(object, x, ...)

## S4 method for signature 'ExpressionSet,data.frame'
annotateEset(object, x, probecol = NULL, annocols = NULL, ...)
```

Arguments

object	An ExpressionSet to which we want to add annotation.
x	Either a ChipDb package (e.g., hugene10sttranscriptcluster.db), or a pdInfoPackage object (e.g., pd.hugene.1.0.st.v1).
...	Allow users to pass in arbitrary arguments. Particularly useful for passing in columns, multivals, and type arguments for methods.
columns	For ChipDb method; what annotation data to add. Use the columns function to see what choices you have. By default we get the ENTREZID, SYMBOL and GENENAME.
multivals	For ChipDb method; this is passed to mapIds to control how 1:many mappings are handled. The default is 'first', which takes just the first result. Other valid values are 'list' and 'CharacterList', which return all mapped results.
type	For pdInfoPackages; either 'core' or 'probeset', corresponding to the 'target' argument used in the call to rma.
probecol	Column of the data.frame that contains the probeset IDs. Can be either numeric (the column number) or character (the column header).
annocols	Column(x) of the data.frame to use for annotating. Can be a vector of numbers (which column numbers to use) or a character vector (vector of column names).

Value

An ExpressionSet that has annotation data added to the featureData slot.

Methods (by class)

- object = ExpressionSet,x = ChipDb: Annotate an ExpressionSet using a ChipDb package for annotation data.
- object = ExpressionSet,x = AffyGenePDInfo: Annotate an ExpressionSet using an AffyGenePDInfo package.

- object = ExpressionSet,x = AffyHTAPDInfo: Annotate an ExpressionSet using an AffyHTAPDInfo package.
- object = ExpressionSet,x = AffyExonPDInfo: Annotate an ExpressionSet using an AffyExonPDInfo package.
- object = ExpressionSet,x = AffyExpressionPDInfo: Annotate an ExpressionSet using an AffyExpressionPDInfo package.
- object = ExpressionSet,x = character: Method to capture character input.
- object = ExpressionSet,x = data.frame: Annotate an ExpressionSet using a user-supplied data.frame.

Author(s)

James W. MacDonald <jmacdon@u.washington>

Examples

```
## Not run:
dat <- read.celfiles(filenames = list.celfiles())
eset <- rma(dat)
## annotate using ChipDb
eset <- annotateEset(eset, hgu10sttranscriptcluster.db)
## or AffyGenePDInfo
eset <- annotateEset(eset, pd.hugene.1.0.st.v1)

## End(Not run)
```

---

dataAndHeatmapPage *A function to create an annotated HTML table for all genes in a significant gene set as well as a heatmap of these data.*

Description

This is intended to be an internal function to runRomer. It is documented here only because it may be necessary to pass alternative arguments to this function from runRomer.

Usage

```
dataAndHeatmapPage(
  object,
  fit,
  ind,
  columns = NULL,
  fname,
  heatmap,
  title,
  key = TRUE,
  fitind = NULL,
```

```
  affy = TRUE,
  ...
)
```

Arguments

object	An ExpressionSet, containing normalized, summarized gene expression data.
fit	An MArrayLM object containing the fitted data.
ind	Numeric vector indicating which rows of the data object to use.
columns	Numeric vector indicating which columns of the data object to use. If NULL, all columns will be used.
fname	The filename of the resulting output, without the 'html' file extension.
heatmap	Character. The filename of the heatmap to append to the bottom of the HTML page.
title	Title to be placed at the top of the resulting HTML page.
key	Character. The filename of the heatmap key to append to the bottom of the HTML page.
fitind	Numeric. Which column of the MArrayLM object to use for output in the HTML table.
affy	Boolean. Are these Affymetrix arrays? If TRUE, then links will be generated to netaffx for the probeset IDs.
...	Included to allow arbitrary commands to be passed to lower level functions.

Details

This function creates an annotation table using probes2table if an annotation file is used, otherwise data will be output in a simple HTML table. A heatmap showing the expression values for all the genes in the gene set is then placed below this table, along with a key that indicates the range of the expression values.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

---

doGlimma *A function to generate MA-plots from Glimma, for all contrasts.*

Description

This function is designed to parse a design and contrasts matrix in order to generate Glimma's interactive MA-plots for all contrasts. The heading for the resulting plot is based on the colnames of the contrasts matrix, so it is important to include the colnames and make them explanatory.

Usage

```
doGlimma(
  tablst,
  datobj,
  dsgn,
  cont,
  grpvec,
  padj = "BH",
  sigfilt = 0.05,
  extraname = NULL,
  ID = NULL,
  sample.cols = rep("#1f77b4", ncol(datobj)),
  ...
)
```

Arguments

tablst	An MArrayLM object or list of DGEExact or DGELRT objects
datobj	A DGEList, ExpressionSet, EList or matrix
dsgn	A design matrix
cont	A contrast matrix
grpvec	A vector of groups, usually what was used to make the design matrix
padj	Method for multiplicity adjustment. BH by default
sigfilt	Significance cutoff for selecting genes
extraname	Used to add a sub-directory to the glimma-plots directory, mainly used to disambiguate contrasts with the same name (see below).
ID	Character, default is NULL. The column name of the genes list item (from the MArrayLM, DGEExact or DGELRT objects) that will be used for labeling the individual gene expression plot at the top right of the resulting plot. The default of NULL will search for a column labeled SYMBOL to use. If the ID value doesn't match any column, and there is no SYMBOL column, then the default for glMDPlot will be used.
sample.cols	A vector of numbers or hex strings, the same length as the grpvec. This will cause the samples in the upper right plot to have different colors. Useful primarily to check for batch differences. The default is the default from glMDPlot.
...	Allows end users to pass other arguments to the Glimma glMDPlot function

Details

In addition, if there are multiple contrasts with the same name, say if the same comparisons are being made for different tissue types, the extraname argument will cause the output to be placed in glimma-plots/<extraname>, to eliminate over-writing of existing files.

Value

A character vector of the files generated, useful for using as links to the output.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

Examples

```
## Not run:
  mat <- matrix(rnorm(1e6), ncol = 20)
  grps <- factor(1:4, each=5)
  design <- model.matrix(~0 + grps)
  colnames(design) <- LETTERS[1:4]
  contrast <- matrix(c(1,-1,0,0,1,0,-1,0,1,0,0,-1,0,1,-1,0,0,1,0,-1),
  ncol = 5)
  colnames(contrast) <- paste(LETTERS[c(1,1,1,2,2)],
  LETTERS[c(2,3,4,3,4)], sep = " vs ")
  fit <- lmFit(mat, design)
  fit2 <- contrasts.fit(fit, contrast)
  fit2 <- eBayes(fit2)
  htmllinks <- doGlimma(fit2, mat, design, contrast, grps)

## End(Not run)
```

---

entrezLinks *Add links to data when using ReportingTools*

Description

These functions are intended to add links to the Affymetrix, Entrez Gene, Nuccore, and AmiGO databases when creating HTML tables using ReportingTools.

Usage

```
entrezLinks(df, ...)
```

Arguments

df	A data.frame, usually created using the `select` function of the AnnotationDbi package. For Entrez ID data, the column name must be ENTREZID. For Affy data, the column name must be PROBEID, and for GO data the column name must be Term. For Nuccore the column name can be any of "GI", "REFSEQ", "ACCNUM". Any other names will fail.
...	Allows one to pass arbitrary arguments to lower level functions. Currently unsupported.

Details

These functions are not actually intended to be called directly. Instead, they are used as targets for the .modifyDF argument of the `publish` function of ReportingTools. See the example below for more detail.

Value

A data.frame is returned, with links included.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

Examples

```
## Not run:
## say we have an ExpressionSet from HuGene 1.0 ST array
dat <- read.celfiles()
eset <- rma(dat)
## annotate the ExpressionSet
eset <- annotateEset(eset, hugene10sttranscriptcluster.db,
columns = c("ENTREZID","ACCNUM","SYMBOL"))
## and fit a model using limma
fit <- lmFit(eset, design)
fit2 <- eBayes(fit)
## and create an HTML page with links to Affy and Entrez
out <- topTable(fit2, coef=2)
htab <- HTMLReport("The title","a_short_name")
publish(out, htab, .modifyDF = list(affyLinks, entrezLinks, nuccoreLinks))
finish(htab)
## End(Not run)
```

---

fixHeaderAndGo *Fix data.frame header for use with ReportingTools*

Description

Internal function used to automatically test for columns that can be converted to links

Usage

```
fixHeaderAndGo(df, affy = TRUE, probecol = "PROBEID")
```

Arguments

df	A data.frame
affy	Boolean; does the data.frame contain Affymetrix probeset IDs?
probecol	Character. The column header containing Affymetrix probeset IDs. Defaults to "PROBEID".

Details

This is an internal function designed to test for the presence of Affymetrix Probeset IDs or Entrez Gene IDs, and if found, generate a list that can be passed to the ReportingTools publish function in order to generate hyperlinks. The underlying assumption is that the data will have been annotated using a Bioconductor annotation package, and thus Affy probeset IDs will have a column header "PROBEID", and Entrez Gene IDs will have a header "ENTREZID" (or any combination of upper and lowercase letters).

Value

Returns a list of length two (with names mdf and df). The mdf object can be passed to the publish using the .modifyDF argument, and the df object is input dat.frame with column names corrected to conform to `affyLinks` and `entrezLinks`, so links will be generated correctly.

Author(s)

Jim MacDonald

`geneSetPage` *A function to create an HTML page for each gene set, as well as the HTML pages for each significant gene set.*

Description

This is intended to be an internal function to `runRomer`, and is not intended to be called by end users. However, the ... argument to `runRomer` allows one to pass arguments to lower level functions, so the arguments are described here.

Usage

```
geneSetPage(
  rslts,
  genesets,
  object,
  fit,
  file,
  cutoff = 0.05,
  dir = ".",
  subdir = ".",
  columns = NULL,
  colnames = NULL,
  col = NULL,
  caption = NULL,
  fitind = NULL,
  bline = NULL,
  affy = TRUE,
  ...
)
```

Arguments

rslts	The results from running romer on one gene set.
genesets	Character. A vector of gene symbols for one gene set.
object	An ExpressionSet, DGEList or EList containing normalized, summarized gene expression data.
fit	An MArrayLM or DGEGLM object, containing the fitted data.
file	Filename for the resulting HTML page.
cutoff	Numeric. The cutoff for significance for a given gene set. Defaults to 0.05.
dir	The directory to write the results. Defaults to the working directory.
subdir	The subdirectory to write the individual gene set results. Defaults to the working directory.
columns	Numeric. The columns of the ExpressionSet to use for the individual gene set output pages. See dataAndHeatmapPage for more information.
colnames	Character. Alternative column names for the resulting heatmap. See dataAndHeatmapPage for more information.
col	A vector of colors for the heatmap. Defaults to bluered.
caption	Caption to put at the top of the HTML page.
fitind	Numeric. The columns of the MArrayLM object to use for the individual HTML tables.
bline	Defaults to NULL. Otherwise, a numeric vector indicating which columns of the data are the baseline samples. The data used for the heatmap will be centered by subtracting the mean of these columns from all data.
affy	Boolean; are these Affymetrix arrays? If TRUE, the Affymetrix probeset IDs will contain links to the netaffx site.
...	Allows arguments to be passed to lower-level functions. See dataAndHeatmapPage and gsHeatmap for available arguments.

Details

This function creates a 'midlevel' HTML table that contains each gene set that was significant, with a link to an HTML table that shows data for each gene in that gene set (with annotation), as well as a heatmap showing the expression levels. Normally this is not run by end users, but is called as part of the runRomer function.

Value

Nothing is returned. Called only for the side effect of creating HTML tables.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

getCols *Correct Ordering of Contrasts*

Description

A function to determine the correct ordering of contrasts for `vennSelect`. This is an internal function and should not be called by the end user

Usage

```
getCols(design, contrast)
```

Arguments

`design`	A design matrix, usually produced by a call to `model.matrix`.
`contrast`	A contrasts matrix, either produced by hand or as a result of a call to `makeContrasts`

Value

This function returns the correct order for the contrasts.

Author(s)

James W. MacDonald

getMainProbes *Remove control probesets from ST arrays*

Description

This function is designed to remove all but the 'main' type of probesets from the Gene ST array types.

Usage

```
getMainProbes(input, level = "core")
```

Arguments

`input`	Either a character string (e.g., "pd.hugene.1.0.st.v1") or a FeatureSet object.
`level`	The summarization level used when calling rma.

Value

If the argument is a character string, returns a data.frame containing probeset IDs along with the probeset type, that can be used to subset e.g., an ExpressionSet of Gene ST data, or an MArrayLM object created from Gene ST data. Note that the order of the probesets is not guaranteed to match the order in your ExpressionSet or MArrayLM object, so that should be checked first. If the argument is a FeatureSet object, it returns a FeatureSet object with only main probes remaining.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

`gsHeatmap` *A function to create a simple heatmap and key.*

Description

This is an internal function called by `runRomer` and is not intended to be used directly. It is documented here only because arguments may be passed down via the `dots` argument.

Usage

```
gsHeatmap(
  object,
  ind,
  filename,
  columns = NULL,
  colnames = NULL,
  col = NULL,
  annot = NULL,
  scale.row = FALSE,
  key = TRUE,
  bline = NULL
)
```

Arguments

`object`	An `ExpressionSet` containing normalized, summarized gene expression data.
`ind`	Numeric vector indicating which rows of the `ExpressionSet` to use.
`filename`	The filename for the heatmap and associated key.
`columns`	Numeric vector indicating which columns of the `ExpressionSet` to use. If `NULL`, all columns will be used.
`colnames`	Character. Substitute column names for the heatmap. If `NULL`, the sampleNames will be used.
`col`	A vector of colors to use for the heatmap. If `NULL`, the `bluered` function will be used.

annot	A matrix or data.frame containing gene symbols to annotate the heatmap. This will normally be extracted automatically from the 'fit' object passed to geneSetPage. If there is no annotation in the fit object, then the probe IDs will be used instead.
scale.row	Boolean. Should the data be scaled by row? Defaults to FALSE.
key	Boolean. Should a key be produced that shows the numeric range for the colors of the heatmap? Defaults to TRUE.
bline	A numeric vector, usually extracted from a contrast matrix, used to 'sweep' the mean baseline sample means from the heatmap data. The end result will be a heatmap in which the colors correspond to log fold changes from the baseline samples.

Details

As noted above, this is only intended to be called indirectly by runRomer. However, certain arguments such as scale.row, or col, etc, can be passed down to this function via the dots argument, allowing the end user to have more control over the finished product.

Value

Nothing is returned. Called only for the side effect of creating heatmaps in 'png' format.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

---

intNames *Create Names for Venn Diagram Intersections*

Description

This function is designed to create the names for all the intersections of a Venn diagram based on the names of all the single comparisons. This is an internal function and is not intended to be called by the end user.

Usage

```
intNames(x)
```

Arguments

x	A TestResults object from a call to decideTests.

Value

A vector of names.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

`make.cl` *A Function to Make a Classlabel Vector for Plotting*

Description

This function takes a vector of filenames and makes a classlabel vector for plotting functions of `affystart`. This is an internal function and is not intended to be called by the end user.

Usage

```
make.cl(filenames)
```

Arguments

`filenames`	A vector of .cel filenames

Value

A vector of numbers to be used for plotting colors and line types

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

`makeGoGeneTable` *Make Gene table from GO analysis results*

Description

A function to create an HTML table showing genes that gave rise to a significant GO term

Usage

```
makeGoGeneTable(
  fit.table,
  probe.sum.table,
  go.id,
  cont.name,
  base.dir = NULL,
  extraname = NULL,
  probecol = "PROBEID",
  affy = TRUE
)
```

Arguments

fit.table	The output from topTable
probe.sum.table	The output from running probeSetSummary on a GOHyperGResults object.
go.id	The GO ID of interest
cont.name	The contrast name.
base.dir	Character. Where should the HTML tables be generated? Defaults to NULL.
extraname	Character. An extra name that can be used if the contrast name isn't descriptive enough.
probecol	The column name in the topTable object that contains probe IDs. Defaults to PROBEID.
affy	Boolean. Are the arrays from Affymetrix?

Details

This is an internal function, not intended to be called by the end user. Documentation here for clarity. After running a GO analysis, it is advantageous to output a table listing those genes that gave rise to a significant GO term. This function creates the table, along with links to Netaffx (if the data are Affymetrix) and to the NCBI Gene database (if there are Entrez Gene IDs).

Value

Returns an HTMLReportRef object.

Author(s)

Jim MacDonald

---

makeGoTable *Create HTML tables for Gene Ontology (GO) analyses*

Description

This function is used to create HTML tables to present the results from a Gene Ontology (GO) analysis.

Usage

```
makeGoTable(
  fit.table,
  go.summary,
  probe.summary,
  cont.name,
  base.dir = "GO_results",
  extraname = NULL,
```

```
  probecol = "PROBEID",
  affy = TRUE
)
```

Arguments

fit.table	The output from topTable
go.summary	The output from running summary on a GOHyperGResults object.
probe.summary	The output from running probeSetSummary on a GOHyperGResults object.
cont.name	The contrast name.
base.dir	Character. Where should the HTML tables be generated? Defaults to GO_results.
extraname	Character. An extra name that can be used if the contrast name isn't descriptive enough.
probecol	The column name in the topTable object that contains probe IDs. Defaults to PROBEID.
affy	Boolean. Are the arrays from Affymetrix?

Details

After running a GO analysis, it is often useful to first present a table showing the set of significant GO terms for a given comparison, and then have links to a sub-table for each GO term that shows the genes that were responsible for the significance of that term. The first table can be generated using the summary function, but it will not contain the links to the sub-table. The ReportingTools package has functionality to make these tables and sub-tables automatically, but the default is to include extra glyphs in the main table that are not that useful.

This function is intended to generate a more useful version of the table that one normally gets from ReportingTools.

Value

Returns an HTMLReportRef object, which can be used when creating an index page to link to the results.

Author(s)

Jim MacDonald

---

makeHmap *A function to create a heatmap-like object or matrix of correlations between miRNA and mRNA data.*

Description

This function is intended for use when both miRNA and mRNA data are available for the same samples. In this situation it may be advantageous to compute correlations between the two RNA types, in order to detect mRNA transcripts that are targeted by miRNA.

Usage

```
makeHmap(
  mRNAdat,
  miRNAdat,
  mRNAlst,
  mRNAvec = NULL,
  miRNAvec = NULL,
  chipPkg,
  header,
  plot = TRUE,
  out = TRUE,
  lhei = c(0.05, 0.95),
  margins = c(5, 8)
)
```

Arguments

mRNAdat	An ExpressionSet, data.frame or matrix of mRNA expression values. The row.names for these data should correspond to the manufacturer's probe ID. Currently, the only manufacturer supported is Affymetrix.
miRNAdat	An ExpressionSet, data.frame or matrix of mRNA expression values. The row.names for these data should correspond to the manufacturer's probe ID. Currently, the only manufacturer supported is Affymetrix.
mRNAlst	A list of mRNA probe IDs where the names of each list item are mirBase miRNA IDs. Usually this will be the output from mirna2mrna.
mRNAvec	A numeric vector used to subset or reorder the mRNA data, by column. If NULL, this will simply be 1:ncol(mRNAdat).
miRNAvec	A numeric vector used to subset or reorder the miRNA data, by column. If NULL, this will simply be 1:ncol(miRNAdat).
chipPkg	Character. The name of the chip-specific annotation package (e.g., "hgu133plus2.db").
header	Character. The plot title if a heatmap is output.
plot	Boolean. Should a heatmap be generated?
out	Boolean. Should the matrix of correlation coefficients be output?
lhei	From heatmap.2. This controls the ratio of the heatmap to the key size. If there are very few mRNAs being plotted, you may need to change to something like (0.5, 5).
margins	From heatmap.2. This controls the right and bottom margins, respectively. Increase either value if names get cut off.

Details

As noted above, this function is intended to generate output from simultaneous analyses of miRNA and mRNA data for the same samples, the goal being either a heatmap like plot of correlations, or the data (or both).

If creating a plot, note that if the number of significant mRNA probes is large, the resulting heatmap will have many rows and will not plot correctly on the usual graphics device within R. In order to visualize, it is almost always better to output as a pdf. In addition, the dimensions of this pdf will have to be adjusted so the row names for the heatmap will be legible. As an example, a heatmap with 10 miRNA transcripts and 100 mRNA transcripts will likely need a pdf with a width argument of 6 and a height argument of 25 or 30. It may require some experimentation to get the correct arguments to the `pdf` function.

Also please note that this function by necessity outputs rectangular data. However, there will be many instances in which a given miRNA isn't thought to target a particular mRNA. Whenever this occurs, the heatmap will have a white cell, and the output data for that combination will be NA.

Value

This function will output a numeric matrix if the 'out' argument is `TRUE`.

Author(s)

James W. MacDonald

See Also

mirna2mrna

`makeImages` *Add dotplot images*

Description

A function to add dotplot glyphs and links to HTML tables

Usage

```
makeImages(
  df,
  eset,
  grp.factor,
  design,
  contrast,
  colind,
  boxplot = FALSE,
  repdir = "./reports",
  extraname = NULL,
  weights = NULL,
  insert.after = 3,
  altnam = NULL,
  ...
)
```

Arguments

df	A data.frame from calling topTable. Note that the row.names for this data.frame must be consistent with the "eset" object. In other words, if "eset" is an ExpressionSet, then the row.names of the data.frame must consistent with the featureNames of the ExpressionSet.
eset	A matrix, data.frame, or ExpressionSet. If using RNA-Seq data, use voom from edgeR to create an EList object, and then pass in the "E" list item.
grp.factor	A factor that indicates which group ALL of the samples belong to. This will be subsetted internally, so do not subset yourself.
design	The design matrix used by limma or edgeR to fit the model.
contrast	The contrast matrix used by limma or edgeR to make comparisons.
colind	Which column of the contrast matrix are we using? In other words, for which comparison are we creating a table?
boxplot	Boolean. If TRUE, the output HTML table will have a boxplot showing differences between groups. If FALSE (default), the table will have dotplots.
repdir	A directory in which to put the HTML tables. Defaults to a "reports" directory in the working directory.
extraname	By default, the tables will go in a "reports" subdirectory, and will be named based on the column name of the contrast that is specified by the colind argument (after replacing any spaces with an underscore). If this will result in name collisions (e.g., a previous file will be over-written because the resulting names are the same), then an extraname can be appended to ensure uniqueness.
weights	Array weights, generally from arrayWeights in the limma package. These will affect the size of the plotting symbols, to reflect the relative importance of each sample.
insert.after	Which column should the image be inserted after? Defaults to 3.
altnam	Normally the output file directories are generated from the colnames of the contrast matrix. This argument can be used to over-ride the default, particularly in the case that one is computing an F-test using a set of columns from the contrast matrix.
...	Allows arbitrary arguments to be passed down to lower level functions.

Details

This function is intended to create little dotplot glyphs that can be added to an HTML table of results from e.g., a microarray or RNA-Seq experiment, showing graphically how much the different groups are changing. The glyphs have unlabeled axes to make them small enough to fit in an HTML table, and clicking on a glyph will result in a new page loading with a full sized dotplot, complete with axis labels.

This function is very similar to the stock functions in the ReportingTools package, but the standard glyphs for that package consist of a dotplot on top of a boxplot, which seems too busy to me. In addition, for most microarray analyses there are not enough replicates to make a boxplot useful.

Value

A list, two items. The first item is the input data.frame with the glyphs included, ready to be used with ReportingTools to create an HTML table. The second item is a pdf of the most differentially expressed comparison. This is useful for those who are using e.g., knitr or Sweave and want to be able to automatically insert an example dotplot in the document to show clients what to expect.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

`makeIndices` *Create Indices for Venn Diagrams*

Description

This function is used to create indices for making Venn diagrams or `vennCounts` objects. This is an internal method and is not intended to be called by the end user.

Usage

```
makeIndices(x, method = "same")
```

Arguments

`x`	A `TestResults` object from a call to `decideTests`.
`method`	One of "same", "up", "down", "both", indicating direction of differential expression. See details of `vennSelect` for more information.

Value

A list containing `TRUE` and `FALSE` values, to be used by `vennSelect`.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

makeVenn *High-level function for making Venn diagrams and outputting the results from the diagrams in HTML and CSV files.*

Description

This function is designed to output CSV and HTML tables based on an analysis using the limma or edgeR packages, with output generated using the ReportingTools package. Please note that a DGEGLM object from edgeR is simply converted to an MArrayLM object from limma and then used in the default MArrayLM method, so all arguments for the MArrayLM object pertain to the DGEGLM method as well.

Usage

```
makeVenn(object, ...)

## S4 method for signature 'MArrayLM'
makeVenn(
  object,
  contrast,
  design,
  groups = NULL,
  collist = NULL,
  p.value = 0.05,
  lfc = 0,
  method = "both",
  adj.meth = "BH",
  titleadd = NULL,
  fileadd = NULL,
  baseUrl = ".",
  reportDirectory = "./venns",
  affy = TRUE,
  probecol = "PROBEID",
  ...
)

## S4 method for signature 'DGEGLM'
makeVenn(
  object,
  contrast,
  design,
  comp.method = c("glmLRT", "glmQLFTest", "glmTreat"),
  lfc = 0,
  ...
)
```

Arguments

object	An MArrayLM or DGEGLM object.
...	Used to pass other arguments to lower level functions.
contrast	A contrasts matrix, produced either by hand, or by a call to makeContrasts
design	A design matrix.
groups	This argument is used when creating a legend for the resulting HTML pages. If NULL, the groups will be generated using the column names of the design matrix. In general it is best to leave this NULL.
collist	A list containing numeric vectors indicating which columns of the fit, contrast and design matrix to use. If NULL, all columns will be used.
p.value	A p-value to filter the results by.
lfc	A log fold change to filter the results by.
method	One of "same", "both", "up", "down", "sameup", or "samedown". See details for more information.
adj.meth	Method to use for adjusting p-values. Default is 'BH', which corresponds to 'fdr'. Ideally one would set this value to be the same as was used for decideTests.
titleadd	Additional text to add to the title of the HTML tables. Default is NULL, in which case the title of the table will be the same as the filename.
fileadd	Additional text to add to the name of the HTML and CSV tables. Default is NULL.
baseUrl	A character string giving the location of the page in terms of HTML locations. Defaults to "."
reportDirectory	A character string giving the location that the results will be written. Defaults to "./venns"
affy	Boolean. Are these Affymetrix data, and should hyperlinks to the affy website be generated in the HTML tables?
probecol	This argument is used in concert with the preceding argument. If these are Affymetrix data , then specify the column header in the MArrayLM object that contains the Affymetrix IDs. Defaults to "PROBEID", which is the expected result if the data are annotated using a BioC annotation package.
comp.method	Character. For DGEGLM objects, the DGEGLM object must first be processed using one of glmLRT, glmQLFTest, or glmTreat. Choose glmLRT if you fit a model using glmFit, glmQLFTest if you fit a model using glmQLFit, or glmTreat if you fit either of those models, but want to incorporate the log fold change into the comparison.

Details

The purpose of this function is to output HTML and text tables with lists of genes that fulfill the criteria of a call to decideTests as well as the direction of differential expression. This is a high-level function that calls vennSelect2 internally, and is intended to be used with vennPage to create a set of Venn diagrams (on an HTML page) that have clickable links in each cell of the diagram.

The links will then pass the end user to individual HTML pages that contain the genes that are represented by the counts in a given cell of the Venn diagram.

In general, the only thing that is needed to create a set of Venn diagrams is a list of numeric vectors that indicate the columns of the contrast matrix that are to be used for a given diagram. See the example below for a better explanation.

Some important things to note: First, the names of the HTML and text tables are extracted from the `colnames` of the `TestResults` object, which come from the contrasts matrix, so it is important to use something descriptive. Second, the method argument is analogous to the `include` argument from `vennCounts` or `vennDiagram`. Choosing "both" will select genes that are differentially expressed in one or more comparisons, regardless of direction. Choosing "up" or "down" will select genes that are only differentially expressed in one direction. Choosing "same" will select genes that are differentially expressed in the same direction. Choosing "sameup" or "samedown" will select genes that are differentially expressed in the same direction as well as 'up' or 'down'.

Note that this is different than sequentially choosing "up" and then "down". For instance, a gene that is upregulated in one comparison and downregulated in another comparison will be listed in the intersection of those two comparisons if "both" is chosen, it will be listed in only one comparison for both the "up" and "down" methods, and it will be listed in the union (e.g., not selected) if "same" is chosen.

Unlike `vennSelect`, this function automatically creates both HTML and CSV output files.

Also please note that this function relys on annotation information contained in the "genes" slot of the "fit" object. If there are no annotation data, then just statistics will be output in the resulting HTML tables.

Value

A list containing the output from calling `vennSelect2` on the columns specified by the collist argument. This is intended as input to `vennPage`, which will use those data to create the HTML page with Venn diagrams with clickable links.

Methods (by class)

- `MArrayLM`: Make a Venn diagram using an MArrayLM object.
- `DGEGLM`: Make a Venn diagram using a DGEGLM object.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

Examples

```
## Not run:
  mat <- matrix(rnorm(1e6), ncol = 20)
  design <- model.matrix(~factor(1:4, each=5))
  colnames(design) <- LETTERS[1:4]
  contrast <- matrix(c(1,-1,0,0,1,0,-1,0,1,0,0,-1,0,1,-1,0,0,1,0,-1),
  ncol = 5)
  colnames(contrast) <- paste(LETTERS[c(1,1,1,2,2)],
  LETTERS[c(2,3,4,3,4)], sep = " vs ")
```

```
    fit <- lmFit(mat, design)
    fit2 <- contrasts.fit(fit, contrast)
    fit2 <- eBayes(fit2)
    ## two Venn diagrams - a 3-way Venn with the first three contrasts
    ## and a 2-way Venn with the last two contrasts
    collist <- list(1:3,4:5)
    venn <- makeVenn(fit2, contrast, design, collist = collist)
    vennPage(venn, "index.html", "Venn diagrams")

## End(Not run)
```

`maplot` *A Function to make MA plots from all arrays.*

Description

This function creates an MA plot for all arrays in either an ExpressionSet or a matrix. A 'baseline' array is created using the median expression for each gene, and each array is then compared to the baseline array.

Usage

```
maplot(object)
```

Arguments

`object`	An ExpressionSet or matrix containing log-transformed array data.

Value

No output. Used only for the side effect of creating MA plots.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

`mirna2mrna` *A function to map miRNA to mRNA.*

Description

This function is intended use when there are miRNA and mRNA data for the same subjects, and the goal is to detect mRNAs that appear to be targeted by the miRNA.

Usage

```
mirna2mrna(
  miRNAids,
  miRNAannot,
  mRNAids,
  orgPkg,
  chipPkg,
  sanger = TRUE,
  miRNAcol = NULL,
  mRNAcol = NULL,
  transType = "ensembl"
)
```

Arguments

`miRNAids`	A character vector of miRNA IDs. Currently only supports Affymetrix platform.
`miRNAannot`	Character. The filename (including path if not in working directory) for the file containing miRNA to mRNA mappings.
`mRNAids`	A character vector of mRNA IDs. Currently only supports Affymetrix platform.
`orgPkg`	Character. The Bioconductor organism package (e.g., org.Hs.eg.db) to be used for mapping.
`chipPkg`	Character. The Bioconductor chip-specific package (e.g., hgu133plus2.db) to be used for mapping.
`sanger`	Boolean. Is the miRNAannot file a Sanger miRBase targets file? These can be downloaded from http://www.ebi.ac.uk/enright-srv/microcosm/cgi-bin/targets/v5/download.pl
`miRNAcol`	Numeric. If using a Sanger miRBase targets file, leave `NULL`. Otherwise, use this to indicate which column of the miRNAannot file contains miRNA IDs.
`mRNAcol`	Numeric. If using Sanger miRBase targets file, leave `NULL`. Otherwise, use this to indicate which column of the miRNAannot file contains mRNA IDs.
`transType`	Character. Designates the type of transcript ID for mRNA supplied by the miRNAannot file. If using the Sanger miRBase files, this is ensembl. Other choices include refseq and accnum.

Details

This function is intended to take a vector of miRNA IDs that are significantly differentially expressed in a given experiment and then map those IDs to putative mRNA transcripts that the miRNAs are supposed to target. The mRNA transcript IDs are then mapped to chip-specific probeset IDs, which are then subsetted to only include those probesets that were also significantly differentially expressed.

The output from this function is intended as input for `makeHmap`.

Value

A list with names that correspond to each significant miRNA, and the mRNA probeset IDs that are targeted by that miRNA.

Author(s)

James W. MacDonald

See Also

makeHmap

---

outputRomer *A function to create HTML output from the results of running romer on a set of contrasts.*

Description

This function is actually intended to be a sub-function of `runRomer`, but can hypothetically run by itself if the `romer` step has already been done.

Usage

```
outputRomer(object, fit, ...)

## S4 method for signature 'ExpressionSet,MArrayLM'
outputRomer(
  object,
  fit,
  rsltlst,
  genesetlst,
  design = NULL,
  contrast = NULL,
  changenames = TRUE,
  dir = "genesets",
  explanation = NULL,
  baseline.hmap = TRUE,
  file = "indexRomer.html",
```

```
  affy = TRUE,
  ...
)

## S4 method for signature 'DGEList,DGEGLM'
outputRomer(
  object,
  fit,
  rsltlst,
  genesetlst,
  design = NULL,
  contrast = NULL,
  changenames = TRUE,
  dir = "genesets",
  explanation = NULL,
  baseline.hmap = TRUE,
  file = "indexRomer.html",
  ...
)

## S4 method for signature 'EList,MArrayLM'
outputRomer(
  object,
  fit,
  rsltlst,
  genesetlst,
  design = NULL,
  contrast = NULL,
  changenames = TRUE,
  dir = "genesets",
  explanation = NULL,
  baseline.hmap = TRUE,
  file = "indexRomer.html",
  ...
)
```

Arguments

object	An ExpressionSet, DGEList or EList object containing normalized, summarized gene expression data.
fit	An MArrayLM or DGEGLM object, containing the fitted data.
...	Arguments to be passed to lower-level functions. See geneSetPage, dataAndHeatmapPage and gsHeatmap for available arguments.
rsltlst	A list of results, generated by the romer function. See discussion for more information.
genesetlst	A list of genesets, usually created by loading in the RData files that can be downloaded from http://bioinf.wehi.edu.au/software/MSigDB/. See details for more information.

design	A design matrix describing the model.
contrast	A contrast matrix describing the contrasts that were fit. This matrix should have colnames, which will be used to name subdirectories containing results.
changenames	Boolean. When creating heatmaps of the gene sets, should the columns be appended with the colnames from the design matrix? If FALSE, the sampleNames will be used.
dir	Character. The subdirectory to use for the output data. Defaults to 'genesets'.
explanation	If NULL, a generic paragraph will be placed at the top of the indexRomer.html page, giving a brief explanation of the analysis. Alternatively, this can be replaced with other text. Please note that this text should conform to HTML standards (e.g., will be pasted into the HTML document as-is, so should contain any required HTML markup).
baseline.hmap	Boolean. If TRUE, then the resulting heatmaps will be centered by subtracting the mean of the baseline sample. As an example, in a contrast of treatment A - treatment B, the mean of the treatment B samples will be subtracted. The heatmap colors then represent the fold change between the A and B samples.
file	Character. The filename to output. Defaults to indexRomer.html.
affy	Boolean. Are these Affymetrix arrays? if TRUE, then thre will be links generated in the HTML table to the netaffx site.

Details

This function is intended to be an internal function for runRomer. However, it is possible that runRomer errored out after saving the results from running romer on a set of contrasts, and all that remains is to create the output HTML.

Please note that the first two arguments to this function have certain expectations. The rsltlst should be the output from running romer. If using the saved output from runRomer, one should first load the 'romer.Rdata' file, which will introduce a list object with the name 'romerlst' into the working directory, so the first argument should be rsltlst = romerlst.

Second, see the code for runRomer, specifically the line that creates the 'sets' object, which will show how to create the correct genesetlst object.

Value

Nothing is returned. The function is run only for the side effect of creating HTML tables with output for each significant gene set.

Methods (by class)

- object = ExpressionSet,fit = MArrayLM: Output romer results using microarray data
- object = DGEList,fit = DGEGLM: Output romer results using RNA-Seq data processed using edgeR
- object = EList,fit = MArrayLM: Output romer results using RNA-Seq data processed using voom.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

`pca.legend` *A Function to Automagically Place a Legend in a PCA Plot*

Description

This function places a legend in a PCA plot depending on which corner is available. If there are no available corners, user intervention will be required. This is an internal function and not intended to be called by the end user.

Usage

```
pca.legend(
  pca,
  groups,
  groupnames,
  pch.df,
  x.coord = NULL,
  y.coord = NULL,
  saveup = FALSE
)
```

Arguments

`pca`	A 'pca' object
`groups`	A vector of numbers indicating group membership
`groupnames`	A vector of names describing the different groups
`pch.df`	A numeric data.frame delineating the plotting symbols (column 1) and colors (column 2) to use
`x.coord`	X-coordinate for legend. Used if automatic placement will fail
`y.coord`	Y-coordinate for legend. Used if automatic placement will fail
`saveup`	Boolean: Save a small value for plotting additional text?

Value

This function returns nothing. It is used only for the side effect of placing a legend in a plot.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

plotDeg *Functions to Plot Density and RNA Degradation Plots*

Description

These functions make density and RNA degradation plots with automatic placement of legends.

Usage

```
plotDeg(dat, filenames = NULL)
```

Arguments

dat	An AffyBatch object, or in the case of plotHist, a matrix (e.g., from a call to read.probematrix. Note that plotDeg requires an AffyBatch object to work correctly.
filenames	Filenames that will be used in the legend of the resulting plot. If NULL (the default), these names will be extracted from the sampleNames slot of the AffyBatch object.

Value

These functions are called only for the side effect of making the plots. Nothing else is returned.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

Examples

```
library("affydata")
data(Dilution)
plotDeg(Dilution)
plotHist(Dilution)
```

plotPCA *A Function to Make a PCA Plot from an ExpressionSet or matrix*

Description

This function makes a PCA plot from an ExpressionSet or matrix

Usage

```
## S4 method for signature 'matrix'
plotPCA(
  object,
  groups = NULL,
  groupnames = NULL,
  addtext = NULL,
  x.coord = NULL,
  y.coord = NULL,
  screeplot = FALSE,
  squarepca = FALSE,
  pch = NULL,
  col = NULL,
  pcs = c(1, 2),
  legend = TRUE,
  main = "Principal Components Plot",
  plot3d = FALSE,
  outside = FALSE,
  ...
)

## S4 method for signature 'ExpressionSet'
plotPCA(object, ...)
```

Arguments

object	An ExpressionSet object or matrix.
groups	A numeric vector delineating group membership for samples. Default is NULL, in which case default plotting symbols and colors will be used.
groupnames	A character vector describing the different groups. Default is NULL, in which case the sample names will be used.
addtext	A character vector of additional text to be placed just above the plotting symbol for each sample. This is helpful if there are a lot of samples for identifying e.g., outliers.
x.coord	Pass an x-coordinate if automatic legend placement fails
y.coord	Pass a y-coordinate if automatic legend placement fails.
screeplot	Boolean: Plot a screeplot instead of a PCA plot? Defaults to FALSE.
squarepca	Should the y-axis of the PCA plot be made comparable to the x-axis? This may aid in interpretation of the PCA plot. Defaults to FALSE.
pch	A numeric vector indicating what plotting symbols to use. Default is NULL, in which case default plotting symbols will be used. Note that this argument will override the 'groups' argument.
col	A numeric or character vector indicating what color(s) to use for the plotting symbols. Default is NULL in which case default colors will be used. Note that this argument will override the 'groups' argument.

pcs	A character vector of length two (or three if plot3d is TRUE), indicating which principal components to plot. Defaults to the first two principal components.
legend	Boolean. Should a legend be added to the plot? Defaults to TRUE.
main	A character vector for the plot title.
plot3d	Boolean. If TRUE, then the PCA plot will be rendered in 3D using the rgl package. Defaults to FALSE. Note that the pcs argument should have a length of three in this case.
outside	Boolean. If TRUE the legend will be placed outside the plotting region, at the top right of the plot.
...	Further arguments to be passed to plot. See the help page for plot for further information.

Value

This function returns nothing. It is called only for the side effect of producing a PCA plot or screeplot.

Functions

- plotPCA,ExpressionSet-method:

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

Examples

```
library("affy")
data(sample.ExpressionSet)
plotPCA(sample.ExpressionSet, groups =
 as.numeric(pData(sample.ExpressionSet)[,2]), groupnames =
 levels(pData(sample.ExpressionSet)[,2]))
```

---

runRomer *A function to run the romer function on a set of contrasts.*

Description

This function automates both running romer on a set of contrasts as well as the creation of output HTML tables that can be used to explore the results. The basic idea here is that one might have used limma to fit a model and compute some contrasts, and then want to do a GSEA using romer.

Usage

```
runRomer(object, ...)

## S4 method for signature 'ExpressionSet'
runRomer(
  object,
  fit,
  setloc,
  annot = NULL,
  design = NULL,
  contrast = NULL,
  wts = NULL,
  save = TRUE,
  baseline.hmap = TRUE,
  affy = TRUE,
  ...
)

## S4 method for signature 'DGEList'
runRomer(
  object,
  fit,
  setloc,
  design = NULL,
  contrast = NULL,
  save = TRUE,
  baseline.hmap = TRUE,
  ...
)

## S4 method for signature 'EList'
runRomer(
  object,
  fit,
  setloc,
  design = NULL,
  contrast = NULL,
  save = TRUE,
  baseline.hmap = TRUE,
  ...
)
```

Arguments

object	An ExpressionSet, DGEList, or EList object
...	Used to pass arguments to lower-level functions. See outputRomer geneSetPage, dataAndHeatmapPage and gsHeatmap for available arguments.
fit	A fitted model from either limma (e.g., MArrayLM) or edgeR (e.g., DGEGLM)

setloc	A character vector giving the path for gene set RData files (see description for more information), or a named list (or list of lists), where the top-level names consist of gene set grouping names (like KeGG or GO), the next level names consist of gene set names (like NAKAMURA_CANCER_MICROENVIRONMENT_UP), and the list items themselves are gene symbols, matching the expected capitalization for the species being used (e.g., for human, they are ALL CAPS. For most other species only the First Letter Is Capitalized).
annot	Character. The name of the array annotation package. If NULL, the annotation data will be extracted from the fData slot (for ExpressionSets) or the genes list (for DGEList or EList objects).
design	A design matrix describing the model fit to the data. Ideally this should be a cell-means model (e.g., no intercept term), as the design and contrast matrices are used to infer which data to include in the output heatmaps. There is no guarantee that this will work correctly with a treatment-contrasts parameterization (e.g., a model with an intercept).
contrast	A contrast matrix describing the contrasts that were computed from the data. This contrast should have colnames, which will be used to create parts of the resulting directory structure.
wts	Optional weights vector - if array weights were used to fit the model, they should be supplied here as well.
save	Boolean. If true, after running the romer step, the results will be saved in a file 'romer.Rdata', which can be used as input for outputRomer to create HTML tables. Since romer can take a long time to run, it is advantageous to keep the default.
baseline.hmap	Boolean. If TRUE, then the resulting heatmaps will be centered by subtracting the mean of the baseline sample. As an example, in a contrast of treatment A - treatment B, the mean of the treatment B samples will be subtracted. The heatmap colors then represent the fold change between the A and B samples.
affy	Boolean; are these Affymetrix arrays? If TRUE, the output tables will contain links to the netaffx site.

Details

The romer expects as input a list or lists of gene symbols that represent individual gene sets. One example is the various gene sets from the Broad Institute that are available at http://bioinf.wehi.edu.au/software/MSigDB/, which are distributed as RData files. The default assumption for this function is that the end user will have downloaded these files, and the setloc argument simply tells runRomer where to find them.

Alternatively, user-based gene sets could be created (these should consist of lists of character vectors of gene symbols - see one of the Broad gene sets for an example).

This function will run romer using all the gene sets in the referenced directory, on all the contrasts supplied, and then output the results in a (default) 'genesets' subdirectory. There will be an HTML file in the working directory with a (default) filename of 'indexRomer.html' that will point to individual HTML files in the genesets subdirectory, which will point to individual files in subdirectories within the genesets subdirectory (named after the colnames of the contrast matrix).

Value

If save is TRUE, return a list that can be re-processed using `outputRomer`. this is useful in cases where you might need to re-run multiple times.

Nothing is returned. This function is called only for the side-effects of creating output HTML files in the working and sub-directories.

Methods (by class)

- `ExpressionSet`: Perform gene set analysis using microarray data.
- `DGEList`: Perform gene set analysis using RNA-Seq data processed using edgeR.
- `EList`: Perform gene set analysis using RNA-Seq data processed using voom.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

James W. MacDonald <jmacdon@u.washington.edu>

`tableFilt` *Filter a topTable object*

Description

This function is designed to filter genes from a `topTable` object based on p-value and/or fold change. This is an internal function and is not intended to be called by thte end user.

Usage

```
tableFilt(
  fit,
  coef = 1,
  number = 30,
  fldfilt = NULL,
  pfilt = NULL,
  adjust = "fdr"
)
```

Arguments

`fit`	An `MArrayLM` object, resulting from a call to `eBayes`
`coef`	The contrast to be extracted into the topTable. See ?topTable for more information.
`number`	The number of genes to output. Only used if both foldfilt and pfilt are NULL.
`fldfilt`	The absolute value of fold difference to filter on. This assumes the data are log transformed.
`pfilt`	The p-value to filter on.

adjust	The multiplicity adjustment to use. Options are '"bonferroni"', '"holm"', '"hochberg"', '"hommel"', '"fdr"' and '"none"'. If '"none"' then the p-values are not adjusted. A 'NULL' value will result in the default adjustment method, which is '"fdr"'.

Value

Returns a data.frame containing the selected genes.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

---

venn4Way *4-way Venn Diagrams*

Description

A function to create a 4-way Venn diagram

Usage

```
venn4Way(
  fit,
  contrast,
  p.value,
  lfc,
  adj.meth,
  baseUrl = ".",
  reportDirectory = "./venns",
  affy = TRUE,
  probecol = "PROBEID",
  ...
)
```

Arguments

fit	An MArrayLM object, created by the limma package.
contrast	A contrasts matrix, used by limma to generate the comparisons made.
p.value	A p-value cutoff for significance
lfc	A log fold change cutoff
adj.meth	The method used to adjust for multiple comparisons.
baseUrl	The base directory for the tables generated. Defaults to ".", meaning the current directory.
reportDirectory	The directory in which to put the results. Defaults to a "venns" subdirectory.
affy	Boolean. Set to TRUE if using Affymetrix microarrays.

probecol	The column containing either the Affymetrix probeset IDs (if the affy argument is set to TRUE) or the name of a column in the output tables that contains uinque identifiers (Entrez Gene IDs, gene symbols, etc).
...	Allows arbitrary arguments to be passed to lower level functions

Details

This function is an internal function and not really intended to be called by the end user. It is generally called by the vennPage function. The goal is to create a 4-way Venn diagram in an HTML page with clickable links to tables of the genes found in a given cell. In addition, the numbers in each cell are underlined with colored bars that help end users tell what contrasts are captured by that cell.

Value

Returns a list. The first item is a (list of) HTMLReportRef objects that can be used by ReportingTools to create HTML links. The second item is the output from the venn function in gtools, and the third item is the name of the contrasts used to generate the Venn diagram.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

---

vennCounts2 *Compute Counts for Venn Diagram*

Description

This function is designed to compute counts for a Venn diagram. It is slightly different from vennCounts in the additional ability to compute counts for genes that are differentially expressed in the same direction.

Usage

```
vennCounts2(x, method = "same", fit = NULL, foldFilt = NULL)
```

Arguments

x	A TestResults object, produced by a call to decideTests or foldFilt.
method	One of "same", "both", "up", "down". See details for more information.
fit	An MArrayLM object, produced by a call to lmFit and eBayes. Only necessary if 'foldFilt' = TRUE.
foldFilt	A fold change to filter samples. This is primarily here for consistency with the corresponding argument in vennSelect.

Details

The function vennCounts will return identical results except for the "same" method. This will only select those genes that both pass the criteria of decideTests as well as being differentially expressed in the same direction. Note that this is different from the "both" method, which simply requires that a given gene be differentially expressed in e.g., two different comparisons without any requirement that the direction be the same.

Value

A VennCounts object.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

Examples

```
library("limma")
tstat <- matrix(rt(300,df=10),100,3)
tstat[1:33,] <- tstat[1:33,]+2
clas <- classifyTestsF(tstat,df=10,p.value=0.05)
a <- vennCounts2(clas)
print(a)
vennDiagram(a)
```

---

vennInLine *Generate Venn diagrams with links for Rmarkdown documents*

Description

A function to generate Venn diagrams for use within Rmarkdown documents, particularly for those using the Bioconductor BiocStyle package for formatting.

Usage

```
vennInLine(
  vennlst,
  caplst,
  cex.venn = 1,
  shift.title = FALSE,
  reportDirectory = NULL,
  ...
)
```

Arguments

vennlst	The output from makeVenn.
caplst	A list of captions to accompany each Venn diagram.
cex.venn	Adjustment parameter for the numbers in the Venn diagram. The default is usually OK.
shift.title	Boolean. Should the titles for the Venn diagram be shifted to accommodate long contrast names?
reportDirectory	Directory containing the Venn diagram. This is usually set by makeVenn and for most people, the default NULL argument should be used.
...	Allows users to pass arbitrary arguments to lower level functions.

Details

This function is intended for those who use Rmarkdown documents to present results and who would like to include Venn diagrams showing the overlap between two to four contrasts. The Venn diagrams that are generated include links for each cell of the diagram that will open HTML pages that contain results for the genes that are found within the cell of the Venn diagram.

Please note that this function is tailored specifically for use within Rmarkdown documents, particularly those that use the Bioconductor BiocStyle package. The function call should be present in a code block using the argument results = "asis", because we are directly generating HTML rather than placing a figure.

Value

This function returns the required HTML text to generate the Venn diagram

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

See Also

vennPage particularly for the example.

---

vennPage *High-level function for making Venn diagrams with clickable links to HTML pages with the underlying genes.*

Description

This function is designed to be used in conjunction with the makeVenn function, to first create a set of HTML pages containing the genes that are represented by the cells of a Venn diagram, and then create an HTML page with the same Venn diagrams, with clickable links that will point the end user to the HTML pages.

Usage

```
vennPage(
  vennlst,
  pagename,
  pagetitle,
  cex.venn = 1,
  shift.title = FALSE,
  baseUrl = ".",
  reportDirectory = NULL,
  ...
)
```

Arguments

vennlst	The output from makeVenn.
pagename	Character. The file name for the resulting HTML page. Something like 'venns' is reasonable. Note that the .html will automatically be appended.
pagetitle	Character. The heading for the HTML page.
cex.venn	Numeric. Adjusts the size of the font in the Venn diagram. Usually the default is OK.
shift.title	Boolean. Should the right contrast name of the Venn diagram be shifted down? Useful for long contrast names. If a two-way Venn diagram, this will shift the right name down so they don't overlap. If a three-way Venn diagram, this will shift the top right name down.
baseUrl	Character. The base URL for the resulting HTML page. The default of "." is usually optimal.
reportDirectory	If NULL, the reportDirectory will be extracted from the vennlst. This is usually what one should do.
...	To allow passing other arguments to lower level functions. Currently not used.

Details

This function is intended to be used as part of a pipeline, by first calling makeVenn and then using the output from that function as input to this function to create the HTML page with clickable links.

Value

An HTMLReport object. If used as input to the ReportingTools publish function, this will create a link on an index page to the Venn diagram HTML page. See e.g., the microarray analysis vignette for ReportingTools for more information.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

Examples

```
## Not run:
  mat <- matrix(rnorm(1e6), ncol = 20)
  design <- model.matrix(~factor(1:4, each=5))
  colnames(design) <- LETTERS[1:4]
  contrast <- matrix(c(1,-1,0,0,1,0,-1,0,1,0,0,-1,0,1,-1,0,0,1,0,-1),
  ncol = 5)
  colnames(contrast) <- paste(LETTERS[c(1,1,1,2,2)],
  LETTERS[c(2,3,4,3,4)], sep = " vs ")
  fit <- lmFit(mat, design)
  fit2 <- contrasts.fit(fit, contrast)
  fit2 <- eBayes(fit2)
  ## two Venn diagrams - a 3-way Venn with the first three contrasts
  ## and a 2-way Venn with the last two contrasts
  collist <- list(1:3,4:5)
  venn <- makeVenn(fit2, contrast, design, eset, collist = collist)
  vennreport <- vennPage(venn, "index.html", "Venn diagrams")
  indexPage <- HTMLReport("index", "My results", reportDirectory =
  ".", baseUrl = ".")
  publish(vennreport)
  finish(indexPage)

## End(Not run)
```

---

vennSelect2 *Select and Output Genelists Based on Venn Diagrams*

Description

This function is designed to output text and/or HTML tables based on the results of a call to decideTests, using the ReportingTools package.

Usage

```
vennSelect2(
  fit,
  contrast,
  design,
  groups = NULL,
  cols = NULL,
  p.value = 0.05,
  lfc = 0,
  method = "same",
  adj.meth = "BH",
  titleadd = NULL,
  fileadd = NULL,
  baseUrl = ".",
```

```
  reportDirectory = "./venns",
  affy = TRUE,
  probecol = "PROBEID",
  ...
)
```

Arguments

fit	An MArrayLM object, from a call to eBayes.
contrast	A contrasts matrix, produced either by hand, or by a call to makeContrasts
design	A design matrix.
groups	This argument is used when creating a legend for the resulting HTML pages. If NULL, the groups will be generated using the column names of the design matrix.
cols	A numeric vector indicating which columns of the fit, contrast and design matrix to use. If NULL, all columns will be used.
p.value	A p-value to filter the results by.
lfc	A log fold change to filter the results by.
method	One of "same", "both", "up", "down", "sameup", or "samedown". See details for more information.
adj.meth	Method to use for adjusting p-values. Default is 'BH', which corresponds to 'fdr'. Ideally one would set this value to be the same as was used for decideTests.
titleadd	Additional text to add to the title of the HTML tables. Default is NULL, in which case the title of the table will be the same as the filename.
fileadd	Additional text to add to the name of the HTML and CSV tables. Default is NULL.
baseUrl	A character string giving the location of the page in terms of HTML locations. Defaults to "."
reportDirectory	A character string giving the location that the results will be written. Defaults to "./venns"
affy	Boolean; are these Affymetrix arrays, and do you want hyperlinks for each probeset to the Affy website to be generated for the resulting HTML tables?
probecol	If the "affy" argument is TRUE, what is the column header for the Affymetrix probeset IDs? Defaults to "PROBEID", which is the default if the data are annotated using a Bioconductor annotation package.
...	Used to pass arguments to lower level functions.

Details

The purpose of this function is to output HTML and text tables with lists of genes that fulfill the criteria of a call to decideTests as well as the direction of differential expression.

Some important things to note: First, the names of the HTML and text tables are extracted from the colnames of the TestResults object, which come from the contrasts matrix, so it is important

to use something descriptive. Second, the method argument is analogous to the include argument from vennCounts or vennDiagram. Choosing "both" will select genes that are differentially expressed in one or more comparisons, regardless of direction. Choosing "up" or "down" will select genes that are only differentially expressed in one direction. Choosing "same" will select genes that are differentially expressed in the same direction. Choosing "sameup" or "samedown" will select genes that are differentially expressed in the same direction as well as 'up' or 'down'.

Note that this is different than sequentially choosing "up" and then "down". For instance, a gene that is upregulated in one comparison and downregulated in another comparison will be listed in the intersection of those two comparisons if "both" is chosen, it will be listed in only one comparison for both the "up" and "down" methods, and it will be listed in the union (e.g., not selected) if "same" is chosen.

Unlike vennSelect, this function automatically creates both HTML and CSV output files.

Value

A list with two items. First, a list of HTMLReport objects from the ReportingTools package, which can be used to create an index page with links to the HTML pages created by this function. See the help page for HTMLReport in ReportingTools as well as the vignettes for more information. The second item is a vennCounts object from limma, which can be used to create a Venn diagram, e.g., in a report if this function is called within a Sweave or knitR pipeline.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

---

writeFit *Function to output annotated fit data from limma*

Description

This function is designed to take an ExpressionSet an annotation package and an lmFit object, and output an annotated text file containing t-statistics, p-values, and fold change data for all contrasts.

Usage

```
writeFit(
  fit,
  annotation = NULL,
  eset,
  touse = c("symbol", "genename", "accnum", "entrezid", "unigene")
)
```

Arguments

fit	A lmFit object, created by the limma package.
annotation	An annotation package, specific for the chip used in the analysis.
eset	An ExpressionSet object containing expression values.
touse	Character vector of BiMaps from annotation package. As an example, if the annotation package is the hgu133plus2.db package, then 'symbol' refers to the hgu133plus2SYMBOL BiMap.

Details

This function is designed to output annotation data as well as statistics (p-values, fold change, t-statistics) for all probes on a chip.

Value

A data.frame is returned.

Author(s)

James W. MacDonald <jmacdon@u.washington.edu>

See Also

write.fit

Index

∗ **hplot**
affystart, 4
maplot, 27
plotDeg, 33
plotPCA, 33
vennCounts2, 40
∗ **internal**
affycoretools-defunct, 3
getCols, 14
intNames, 16
make.cl, 17
makeIndices, 23
pca.legend, 32
tableFilt, 38
∗ **manip**
affystart, 4
doGlimma, 8
entrezLinks, 10
getMainProbes, 14
makeHmap, 19
makeVenn, 24
mirna2mrna, 28
vennCounts2, 40
vennInLine, 41
vennPage, 42
vennSelect2, 44
writeFit, 46

affycoretools, 2
affycoretools-defunct, 3
affyLinks *(*entrezLinks*)*, 10
affystart, 4
annBM *(*affycoretools-defunct*)*, 3
annotateEset, 5
annotateEset,ExpressionSet,AffyExonPDInfo-method *(*annotateEset*)*, 5
annotateEset,ExpressionSet,AffyExpressionPDInfo-method *(*annotateEset*)*, 5
annotateEset,ExpressionSet,AffyGenePDInfo-method *(*annotateEset*)*, 5
annotateEset,ExpressionSet,AffyHTAPDInfo-method *(*annotateEset*)*, 5
annotateEset,ExpressionSet,character-method *(*annotateEset*)*, 5
annotateEset,ExpressionSet,ChipDb-method *(*annotateEset*)*, 5
annotateEset,ExpressionSet,data.frame-method *(*annotateEset*)*, 5

bluered, *13*, *15*

columns, *6*

dataAndHeatmapPage, 7
decideTests, *25*, *40, 41*, *44, 45*
DGEGLM, *25*
doGlimma, 8

eBayes, *40*, *45*
ensemblLinks *(*entrezLinks*)*, 10
entrezLinks, 10
ExpressionSet, *15*

fixHeaderAndGo, 11
foldFilt *(*affycoretools-defunct*)*, 3
foldFiltBM *(*affycoretools-defunct*)*, 3

geneSetPage, 12
getCols, 14
getMainProbes, 14
glmFit, *25*
glmLRT, *25*
glmQLFit, *25*
glmQLFTest, *25*
glmTreat, *25*
GOHyperGResults, *18, 19*
goLinks *(*entrezLinks*)*, 10
gsHeatmap, 15

heatmap.2, *20*
HTMLReportRef, *18, 19*

hyperG2annaffy *(affycoretools-defunct)*, 3
hyperGoutput *(affycoretools-defunct)*, 3

intNames, 16

limma2annaffy *(affycoretools-defunct)*, 3
limma2biomaRt *(affycoretools-defunct)*, 3
lmFit, *40*

make.cl, 17
makeContrasts, *14*, *25*, *45*
makeGoGeneTable, 17
makeGoTable, 18
makeHmap, 19, *29*
makeImages, 21
makeIndices, 23
makeVenn, 24
makeVenn,DGEGLM-method *(makeVenn)*, 24
makeVenn,MArrayLM-method *(makeVenn)*, 24
mapIds, *6*
maplot, 27
MArrayLM, *25*, *40*, *45*
mirna2mrna, *20*, 28

nuccoreLinks *(entrezLinks)*, 10

outputRomer, 29
outputRomer,DGEList,DGEGLM-method *(outputRomer)*, 29
outputRomer,EList,MArrayLM-method *(outputRomer)*, 29
outputRomer,ExpressionSet,MArrayLM-method *(outputRomer)*, 29

pca.legend, 32
plotDeg, 33
plotHist *(plotDeg)*, 33
plotPCA, 33
plotPCA,ExpressionSet-method *(plotPCA)*, 33
plotPCA,matrix-method *(plotPCA)*, 33
probes2table *(affycoretools-defunct)*, 3
probes2tableBM *(affycoretools-defunct)*, 3
probeSetSummary, *18*, *19*
publish, *12*

rma, *6*
romer, *13*, *29–31*, *35*, *37*
runRomer, 35
runRomer,DGEList-method *(runRomer)*, 35
runRomer,EList-method *(runRomer)*, 35
runRomer,ExpressionSet-method *(runRomer)*, 35

screeplot, *34*
summary, *19*

tableFilt, 38
TestResults, *40*
topTable, *18*, *19*

venn4Way, 39
VennCounts, *41*
vennCounts, *26*, *40*, *41*, *46*
vennCounts2, 40
vennDiagram, *26*, *46*
vennInLine, 41
vennPage, 42
vennSelect *(affycoretools-defunct)*, 3
vennSelect2, 44
vennSelectBM *(affycoretools-defunct)*, 3
vennSelectFC *(affycoretools-defunct)*, 3

write.fit, *47*
writeFit, 46