Package 'affycoretools'

March 25, 2013

Date 2008-12-15

Title Functions useful for those doing repetitive analyses with Affymetrix GeneChips.

Version 1.30.0

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@med.umich.edu>

License Artistic-2.0

Depends affy, Biobase, GO.db, KEGG.db

Imports biomaRt, limma, GOstats, annotate, annaffy, genefilter, gcrma,splines, xtable, AnnotationDbi, lattice, gplots

Suggests affydata, hgfocuscdf, rgl

biocViews ReportWriting, Microarray, OneChannel, GeneExpression

R topics documented:

---

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.
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.
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.
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@med.umich.edu>

See Also

plotHist, plotDeg, plotPCA

---

annBM *Select Available Annotation from a Biomart*

Description

These functions are designed to do two things that are useful for an end user. If called with no arguments, they will output a character vector of annotation sources that are typically available from a Biomart database. If called with a 'mart' connection (typically created by a call to useMart), they will return a character vector of annotation sources that exist for that particular Biomart and species. If called with a 'mart' connection and a character vector of annotation sources, they will return a list that is intended to be used by other functions for creating HTML pages. This last function doesn't have any real utility for the end user.

Usage

```
annBM(mart, annot, species)
linksBM(mart, annot, affyid = FALSE, ann.source = NULL)
```

Arguments

mart	A 'mart' connection, typically created by a call to useMart.
annot	A character vector of annotation sources. This is not typically useful for an end user to specify.
affyid	Boolean. Are the IDs being annotated Affymetrix IDs?
ann.source	Character. The filter name to use if we are annotating Affymetrix IDs.
species	A species name, of the form e.g., 'hsapiens'

Details

The purpose of these functions is to either give an example of typical annotation sources that may be available at a particular Biomart, or to output those sources that are known to exist at a Biomart.

linksBM is intended to list those annotation sources that may be turned into hyperlinks whereas annBM is intended to list those annotation sources that will not be linked.

These functions have only a few of the possible annotation sources, and currently there is no simple way to extend these sources. Additions to the list are possible, however. Please contact me if there is something in particular that should be included in either list.

Value

Normally called by an end user to output a character vector of annotation sources.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

Examples

annBM()

---

foldFilt *Output Fold Change Data*

Description

This function is designed to take an ExpressionSet and some comparisons and output either HTML tables, text files, or both.

Usage

foldFilt(object, fold = 1, groups, comps, compnames, save = FALSE, text = TRUE, html = TRUE, filterfun = NULL)

Arguments

object	An ExpressionSet object
fold	The log fold change cutoff to use. Note that this is log base two.
groups	A vector of group identifiers. Probably easiest to use a numeric vector
comps	A list containing all the comparisons to be made. Each list item should be a vector of length two. See details for more information.
compnames	A character vector of the names for each of the comparisons to be made. This will be the name of the resulting HTML or text file.
save	Boolean. If TRUE, a list will be returned. The first item in the list will be a vector showing the number of 'significant' genes for each comparison. The second item will be a matrix of -1's, 0's and 1's indicating a significant difference, and the direction of the difference. The first item is useful for creating Sweave - based reports and the second is useful for making Vennn diagrams using the vennDiagram from the limma package.
html	Boolean - if TRUE, output HTML tables
text	Boolean - if TRUE, output text tables
filterfun	A filtering function, created by genefilter to filter the data using additional criteria. See details for more information

Details

This function is useful for outputting annotated gene lists for multiple fold change comparisons. The genes will be ordered by the absolute fold change. Note that this function is essentially a wrapper to call annaffy, so is only useful for Affymetrix GeneChips for which there is an annotation package.

Without attaching a data file to this package, it is not possible to give a working example. Instead, here is a 'for instance'.

Say you have an ExpressionSet containing four Affy HG-U133Plus2 chips. There is no replication, and you simply want to output genes with a two-fold or greater difference between the first chip and each of the last three (the first chip is the control, and the other three are experimentals). The ExpressionSet is called eset.

Additionally, say we don't want any genes called significant if both of the samples have very low expression. We can set up a filter using the **genefilter** package.

f1 <- kOverA(1,6)

filt <- filterfun(f1)

foldFilt(eset, groups=1:4, comps=list(c(2, 1), c(3, 1), c(4, 1)), compnames=c("Expt1-Cont","Expt2-Cont","Expt3-Cont"), filterfun = filt)

This will output three HTML tables called 'Expt1-Cont.html', etc., each containing sorted genes that have two-fold or greater differences between the two samples.

Value

Returns a list; see above for the elements of the list. This function is mainly called for the side effect of outputting HTML or text files containing annotated 'significant' gene lists.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

---

foldFiltBM *Output Fold Change Data using biomaRt*

Description

This function is designed to take an ExpressionSet and some comparisons and output HTML tables. It is very similar to foldFilt except it uses the biomaRt package to annotate genes and the annotate package to create the HTML table(s).

Usage

foldFiltBM(object, fold = 1, groups, comps, compnames, species, links = linksBM()[1:3], otherann = annBM()[1:3], filterfun = NULL, ann.source = "entrezgene", affyid = FALSE, html = TRUE, text = TRUE, save = FALSE)

Arguments

object	An ExpressionSet object
fold	The log fold change cutoff to use. Note that this is log base two.
groups	A vector of group identifiers. Probably easiest to use a numeric vector
comps	A list containing all the comparisons to be made. Each list item should be a vector of length two. See details for more information.
compnames	A character vector of the names for each of the comparisons to be made. This will be the name of the resulting HTML or text file.
species	The species name. This must be in a particular format for biomaRt. An example for human is "hsapiens" or for mouse is "mmusculus".

links	A character vector of things to annotate with hyperlinks to online databases. See linksBM for possible values.
otherann	A character vector of things to annotate with text only (i.e., no hyperlinks). See annBM for possible values.
filterfun	A filtering function created by genefilter to filter the data using additional criteria. See details for more information
ann.source	The annotation source of the IDs that will be used to annotate the genes. The default value is "entrezgene". See details for other possibilities.
affyid	Boolean. Are the IDs used to annotate these data Affymetrix IDs?
html	Boolean. Output HTML tables? Defaults to TRUE
text	Boolean. Output text tables? Defaults to TRUE
save	Boolean. If TRUE, a list will be returned. The first item in the list will be a vector showing the number of 'significant' genes for each comparison. The second item will be a matrix of -1's, 0's and 1's indicating a significant difference, and the direction of the difference. The first item is useful for creating Sweave - based reports and the second is useful for making Vennn diagrams using vennDiagram from the limma package.

Details

This function is useful for outputting annotated gene lists for multiple fold change comparisons. The genes will be ordered by the absolute fold change.

This function currently only supports Affymetrix data. It is designed for Affymetrix chips that don't have an annotation package, which includes data that have been analyzed using the 're-mapped' CDFs supplied to BioC by MBNI at University of Michigan.

The IDs that will be used to annotate the genes depend on the source of the data. If, for example, one is using an Affymetrix chip that doesn't have a BioC annotation package, then the IDs will be Affymetrix IDs. To find out the correct name to use for the ann.source argument, one can create a connection to a Biomart database using useMart and then get a list of available Affy arrays using getAffyArrays.

If one is using one of the re-mapped CDFs from MBNI at University of Michigan, then the IDs to use depend on the mapping used to create the CDF. At this time, only three types of CDFs can be used; EntrezGene, UniGene, and RefSeq. One can determine the correct ann.source argument by creating a connection to a Biomart database, and then calling linksBM(mart, linksBM())[[3]].

One can also protect against selecting probesets that have very small expression values for all samples (which likely have a large fold change due to noise, rather than signal) by using the filterfun argument. An example would be:

f <- kOverA(1, 6)

filt <- filterfun(f)

Then add filterfun = filt as an argument to the call to foldFilt.

Value

Returns a list; see above for the elements of the list. This function is mainly called for the side effect of outputting HTML or text files containing annotated 'significant' gene lists.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

getUniqueLL *Subset a Vector of Probesets*

Description

This function will take a vector of Affy IDs and return a vector of Entrez IDs that have replicated IDs removed. The resulting vector will still have the corresponding Affy IDs appended as names, which is important for some functions.

Usage

```
getUniqueLL(probes, annot)
```

Arguments

probes	A vector of probe IDs
annot	The annotation package for the chip used

Details

Subsetting a set of Affy IDs to unique Entrez Gene IDs is a common thing to do prior to doing a hypergeometric test. Functions such as hyperGTest can use un-named vectors of Entrez IDs (e.g., unique(getLL(probeIDs, annot))), but there is some functionality that requires the Entrez Gene IDs to be in a named vector, with the names being the associated Probeset IDs.

As an example, hyperGoutput will only work correctly if the input Entrez ID vector is named with the associated Probeset IDs.

Value

A named vector of unique Entrez IDs

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

---

hyperG2annaffy *HTML tables from GOIDs*

Description

Output HTML tables containing the 'enriched' genes for each GO term resulting from a call to hyperGtable.

Usage

```
hyperG2annaffy(probids, lib, eset, fit = NULL, subset = NULL, comp = 1, type = "MF",
pvalue = 0.05, min.count = 10)
```

Arguments

probids	A vector of Affymetrix probe IDs
lib	An annotation package (e.g., hgu95av2)
eset	An ExpressionSet
fit	An lmFit object. Only necessary if statistics are desired in the resulting table. Defaults to NULL.
subset	A numeric vector used to select GO terms to output (see description for more information). Defaults to NULL
comp	Which contrast/parameter estimate should be used to extract the relevant statistics? Only used if fit is not NULL. See description for more information.
type	One of "MF", "CC", "BP", indicating molecular function, cellular component, or biological process, respectively.
pvalue	The significance level used to choose GO terms
min.count	The minimum number of a given GO term that must be on the chip in order to choose that GO term. This protects against very low p-values that result from the situation where there are very few genes with a given GO term on the chip, but one or two are found in the set of significant genes.

Details

This function is used to create HTML tables based on the output of hyperGtable. The basic idea is as follows; as part of an analysis, say hyperGtable was used to create a table of 'enriched' GO terms. Unfortunately, the table only lists GO terms and the number of probesets that are annotated to those GO terms, and the client may be interested in knowing what probesets are enriched for each (or some) GO term.

The default behaviour is to output an HTML table for each GO term, containing the probesets that are annotated at that term (and that are in the set of significant genes). If only some of the GO terms are of interest, one may use the subset argument to select only particular rows. In addition, if the relevant t-statistics, p-values and fold changes are of interest, one can also use the fit argument to point to an lmFit object that contains these data, as well as the comp argument to indicate which parameter or contrast to use. Note that the comp argument defaults to 1, so the first parameter or contrast will be extracted by default.

Value

This function is used only for the side effect of creating HTML tables.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

hyperGoutput *Output Tables Based on Hypergeometric Test*

Description

This function will output various tables containing probesets that are annotated to a particular GO, KEGG, or PFAM term. The tables are based on the results from a call to hyperGtest.

Usage

hyperGoutput(hyptObj, eset, pvalue, categorySize, sigProbesets, fit = NULL, subset = NULL,comp = 1, output = c("significant", "all", "split"), statistics = c("tstat", "pval", "FC"), html = TRUE, text = TRUE, ...)

Arguments

hyptObj	A HyperGResult object, usually produced by a call to hyperGTest
eset	An ExpressionSet object
pvalue	The p-value cutoff used for selecting significant GO terms. If not specified, it will be extracted from the HyperGResult object
categorySize	Number of terms in the universe required for a term to be significant. See details for more information
sigProbesets	Vector of probeset IDs that were significant in the original analysis.
fit	An MArrayLM object, produced from a call to eBayes
subset	Numeric vector used to select particular tables to output. The default is to output tables for all terms. See details for more information
comp	Numeric vector of length one, used to indicate which comparison in the MArrayLM object to use for extracting relevant statistics. See details for more information
output	One of 'selected', 'all', or 'split'. See details for more information
statistics	Which statistics to output in the resulting tables. Choices include 'tstat', 'pval', or 'FC', corresponding to t-statistics, p-values, and fold change, respectively
html	Boolean. Output HTML tables? Defaults to TRUE
text	Boolean. Output text tables? Defaults to TRUE
...	Allows end user to pass further arguments. The most notable would be an anncols argument, passed to probes2table to control the hyperlinked annotation columns. See aaf.handler for more information

Details

This function is designed to be used to output the results from a hypergeometric test for over-represented terms. This function would be used at the end of an analysis such as:

1.) Compute expression values 2.) Fit a model using limma 3.) Output significant probesets using limma2annaffy 4.) Perform hypergeometric test using hyperGTest

At step 4, one can output a list of the over-represented terms using htmlReport. One might then be interested in knowing which probesets contributed to the significance of a particular term, which is what this function is designed to do.

One argument that can be passed to htmlReport (and also to hyperGoutput) is categorySize, which gives a lower bound for the number of probesets with a particular term in the universe. In other words, assume that a particular GO term is annotated to three probesets on a given chip. If, after doing a t-test to detect differentially expressed probesets, one of those probesets were found to be significantly differentially expressed and was then used to do a hypergeometric test, that GO term would be significant, with a small p-value. However, this is probably not very strong evidence that the GO term is actually over-represented, since there were only three to begin with. By setting categorySize to a sensible value (such as 10), this situation can be avoided.

This function will output HTML and/or text tables containing annotation information about each probeset as well as the expression values. In addition, if limma were used to fit the model, the relevant statistics (t-statistic, p-value, fold change) can also be output in the table by passing the MArrayLM object that resulted from a

call to eBayes. The statistics argument can

be used to control which statistics are output.

By default hyperGoutput will output tables for all significant terms, which may end up being quite a few tables. Usually only a few terms are of interest, so there is a subset argument that can be used to select only those terms. This argument follows directly from the order of the table output by htmlReport or summary. For instance, if the first, third and fifth terms in the HTML table output by htmlReport were of interest, one would use subset=c(1,3,5).

One critical step prior to the hypergeometric test is to subset the probesets to unique Entrez Gene IDs. It should be noted however, that the functions used by hypergOutput will output all the probesets annotated to a particular term. The output argument is used to control this behavior. If output = "significant" (the default), then only those probesets that correspond to the original subsetting will be output. If output = "all", then all probesets will be output (grouped by Entrez ID), with the 'significant' probeset first. If output = "split", then all the probesets will be output, with all the 'significant' probesets first, followed by the other probesets, grouped by Entrez ID.

Note that the 'significant' probesets come from one of two sources. First, one can pass a character vector of probeset IDs corresponding to those that were significant in the original analysis (recommended). Second, if the geneIds slot of the GOHyperGParams object containes a named vector of Entrez Gene IDs, then the names from that vector will be used. This can be accomplished by using either findLargest or getUniqueLL.

Since the geneIds are by definition a unique set of Entrez Gene IDs, any duplicate probeset IDs will have been removed, so the first method is to be preferred for accuracy.

Value

This function returns no value, and is called solely for the side effect of outputting HTML and/or text tables.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

See Also

hyperGTest, htmlReport, probeSetSummary

limma2annaffy *Function to Create HTML Tables from limma Objects*

Description

This function is designed to take an ExpressionSet and an lmFit, model.matrix, and contrast object from limma and convert into HTML tables using annaffy. The alternate function limma2annaffy.na is designed to be run without user intervention.

Usage

```
limma2annaffy(eset, fit, design, contrast, lib, adjust = "fdr",
 anncols = aaf.handler()[c(1:3, 6:7, 9:12)], number = 30, pfilt = NULL,
 fldfilt= NULL,tstat = TRUE, pval = TRUE, FC = TRUE,
 expression = TRUE, html = TRUE, text = FALSE, save = FALSE, addname =
NULL, addtitle = NULL, interactive = TRUE)
```

Arguments

eset	An ExpressionSet containing affymetrix expression values.
fit	An lmFit object.
design	A model.matrix object.
contrast	A contrasts matrix from limma.
lib	An annotation package for the Affy chips used.
adjust	Multiplicity adjustment. Choices are "fdr","holm","hommel","bonferroni", or "none". Partial matching allowed.
anncols	A vector of things to annotate, produced by a call to aaf.handler().
number	Number of genes to output to table. See details for more information.
pfilt	A p-value to filter output. See details for more information.
fldfilt	A fold change to filter output. See details for more information.
tstat	Boolean: Output t-statistics in table? Defaults to FALSE.
pval	Boolean: Output (adjusted) p-values in table? Defaults to FALSE.
FC	Boolean: Output fold changes in table? Defaults to FALSE.
expression	Boolean: Output expression values in table? Defaults to TRUE.
html	Boolean: Output data in HTML tables? Defaults to TRUE.
text	Boolean: Output data in text tables? Defaults to TRUE.
save	Boolean: Save tables as R objects for further processing? Defaults to FALSE.
addname	A character vector to add to the end of the automatically generated output file names. Useful for multiple calls to eliminate over-writing of existing HTML or text tables.
addtitle	A character vector to add to the title for the HTML table. By default the title will be the same as the filename. If the addname argument is not NULL, then that will be appended to the filename (and will be used as the HTML title). If addtitle is not NULL, it will be appended to the filename and that will then be used as the HTML table title.
interactive	Boolean: Is this an interactive call, or run as part of a script (e.g., in an Sweave document)? Defaults to TRUE

Details

This function is designed to automatically output HTML or text tables, with filenames taken from the column names of the contrast matrix. The number of genes output can be controlled several different ways. First, if pfilt and fldfilt are both NULL, the top genes will be output based on the number variable. Otherwise, the genes are filtered based on p-value, fold change, or both. If the genes are filtered this way, the number of genes to be output will be listed and the filter(s) can then be adjusted if necessary.

This function currently only supports Affymetrix data.

Value

If save is TRUE, a list of tables from topTable will be output.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

See Also

topTable, aafTableAnn

---

limma2biomaRt *Function to Create HTML Tables from limma Objects using biomaRt for Annotation*

Description

This function is designed to take an ExpressionSet and an lmFit, model.matrix, and contrast object from limma and convert into HTML and text tables using biomaRt. The alternate function limma2biomaRt.na is designed to be run without user intervention.

Usage

```
limma2biomaRt(eset, fit, design, contrast, species, links = linksBM()[1:3],
otherdata = annBM()[1:3], ann.source = "entrezgene", adjust = "fdr",
number = 30, pfilt = NULL, fldfilt = NULL, tstat = TRUE,
pval = TRUE, FC = TRUE, expression = TRUE, html = TRUE, text = TRUE,
save = FALSE, addname = NULL, interactive = TRUE, affyid = FALSE)
```

Arguments

eset	An ExpressionSet containing affymetrix expression values.
fit	An lmFit object.
design	A model.matrix object.
contrast	A contrasts matrix from limma.
species	The species name. This must be in a particular format for biomaRt. An example for human is "hsapiens", or for mouse "mmusculus".
links	A character vector of things to annotate with hyperlinks to online databases. See linksBM for possible values.

otherdata	A character vector of things to annotate with text only (i.e., no hyperlinks). See annBM for possible values.
ann.source	The annotation source of the IDs that will be used to annotate the genes. The default value is "entrezgene". See details for other possibilities.
adjust	Multiplicity adjustment. Choices are "fdr","holm","hommel","bonferroni", or "none". Partial matching allowed.
number	Number of genes to output to table. See details for more information.
pfilt	A p-value to filter output. See details for more information.
fldfilt	A fold change to filter output. See details for more information.
tstat	Boolean: Output t-statistics in table? Defaults to FALSE.
pval	Boolean: Output (adjusted) p-values in table? Defaults to FALSE.
FC	Boolean: Output fold changes in table? Defaults to FALSE.
expression	Boolean: Output expression values in table? Defaults to TRUE.
html	Boolean: Output data in HTML tables? Defaults to TRUE.
text	Boolean: Output data in text tables? Defaults to TRUE
save	Boolean: Save tables as R objects for further processing? Defaults to FALSE.
addname	A character vector to add to the end of the automatically generated output file names. Useful for multiple calls to eliminate over-writing of existing HTML or text tables.
interactive	Boolean: Is this an interactive call, or run as part of a script (e.g., in an Sweave document)? Defaults to TRUE
affyid	Boolean. Are the IDs used to annotate these data Affymetrix IDs?

Details

This function is designed to automatically output HTML tables, with filenames taken from the column names of the contrast matrix. The number of genes output can be controlled several different ways. First, if pfilt and fldfilt are both NULL, the top genes will be output based on the number variable. Otherwise, the genes are filtered based on p-value, fold change, or both. If the genes are filtered this way, the number of genes to be output will be listed and the filter(s) can then be adjusted if necessary.

This function currently only supports Affymetrix data. It is designed for Affymetrix chips that don't have an annotation package, which includes data that have been analyzed using the 're-mapped' CDFs supplied to BioC by MBNI at University of Michigan.

The IDs that will be used to annotate the genes depend on the source of the data. If, for example, one is using an Affymetrix chip that doesn't have a BioC annotation package, then the IDs will be Affymetrix IDs. To find out the correct name to use for the ann.source argument, one can create a connection to a Biomart database using useMart and then get a list of available Affy arrays using listFilters.

If one is using one of the re-mapped CDFs from MBNI at University of Michigan, then the IDs to use depend on the mapping used to create the CDF. At this time, only three types of CDFs can be used; EntrezGene, UniGene, and RefSeq. One can determine the correct ann.source argument by creating a connection to a Biomart database, and then calling linksBM(mart, linksBM())[[3]].

Value

If save is TRUE, a list of tables from topTable will be output.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

See Also

topTable, aafTableAnn

---

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 =

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?

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

---

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, layout)

Arguments

object	An ExpressionSet or matrix containing log-transformed array data.
layout	A numeric vector, length two. Best results will be obtained if both values are the same, and between 2 and 5 (e.g., c(3,3))

Value

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

Author(s)

James W. MacDonald <jmacdon@med.umich.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

---

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)
plotHist(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@med.umich.edu>

Examples

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

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

Description

This function makes a PCA plot from an ExpressionSet or matrix

Usage

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, ...)

Arguments

object	An ExpressionSet, matrix or prcomp object.
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.
...	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.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

Examples

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

---

probes2table *Convert Affy Probe ids to Annotated HTML Table*

Description

A function to convert a vector of Affy ids to an annotated HTML table.

Usage

```
probes2table(eset, probids, lib, otherdata = NULL,
 anncols = aaf.handler()[c(1:3, 6:7, 9:12)], html = TRUE, text = FALSE,
 express = TRUE, save = FALSE, filename, title = NULL)
```

Arguments

eset	An ExpressionSet containing Affy expression values.
probids	A vector of probe ids.
lib	An annotation package for the Affy chips used.
otherdata	A *named* list of additional information to include in the resulting table. Examples would be t-statistics, p-values, fold change, etc. Each list item should be a vector the same length as the probids vector. The name associated with each list item will be used as the column name in the resulting table.
anncols	A vector of things to annotate, produced by a call to aaf.handler().
html	Output data in HTML tables? Defaults to TRUE.
text	Output data in text tables? Defaults to TRUE.
express	Output expression values in table? Defaults to TRUE.
save	Should tables be saved as R objects for further processing? Defaults to FALSE.
filename	Filename of the resulting HTML table.
title	Title for HTML table. If NULL, the filename will be used.

Value

If save is TRUE, a data.frame is saved containing the data.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

See Also

topTable, aafTableAnn

---

probes2tableBM *Convert Affy Probe ids to Annotated HTML Table using biomaRt*

Description

A function to convert a vector of Affy ids to an annotated HTML or text table. This function is very similar to probes2table, except it uses the biomaRt package to annotate genes, and the annotate package to create the HTML table.

Usage

```
probes2tableBM(eset, probids, species, filename, otherdata = NULL,
 links = linksBM()[1:3], otherann = annBM()[1:3], ann.source = "entrezgene",
 express = TRUE, html = TRUE, text = TRUE, affyid = FALSE)
```

Arguments

eset	An ExpressionSet containing Affy expression values.
probids	A vector of probe ids.
species	The species name. This must be in a particular format for biomaRt. An example for human is "hsapiens" or for mouse is "mmusculus".
filename	File name of the resulting HTML table.
otherdata	A *named* list of additional information to include in the resulting table. Examples would be t-statistics, p-values, fold change, etc. Each list item should be a vector the same length as the probids vector. The name associated with each list item will be used as the column name in the resulting table.
links	A character vector of things to annotate with hyperlinks to online databases. See linksBM for possible values.
otherann	A character vector of things to annotate with text only (i.e., no hyperlinks). See annBM for possible values.
ann.source	The annotation source of the IDs that will be used to annotate the genes. The default value is "entrezgene". See details for other possibilities.
express	Output expression values in table? Defaults to TRUE.
html	Boolean. Output HTML table? Defaults to TRUE
text	Boolean. Output text table? Defaults to TRUE
affyid	Boolean. Are the IDs used to annotate these data Affymetrix IDs?

Details

This function is designed to output HTML tables based on a set of IDs. This function currently only supports Affymetrix data. It is designed for Affymetrix chips that don't have an annotation package, which includes data that have been analyzed using the 're-mapped' CDFs supplied to BioC by MBNI at University of Michigan.

The IDs that will be used to annotate the genes depend on the source of the data. If, for example, one is using an Affymetrix chip that doesn't have a BioC annotation package, then the IDs will be Affymetrix IDs. To find out the correct name to use for the ann.source argument, one can create a connection to a Biomart database using useMart and then get a list of available Affy arrays using listFilters.

If one is using one of the re-mapped CDFs from MBNI at University of Michigan, then the IDs to use depend on the mapping used to create the CDF. At this time, only three types of CDFs can be used; EntrezGene, UniGene, and RefSeq. One can determine the correct ann.source argument by creating a connection to a Biomart database, and then calling linksBM(mart, linksBM())[[3]].

Value

This function is only used for the side effect of outputting an HTML table.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

See Also

topTable

---

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@med.umich.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)
```

---

vennSelect *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.

Usage

```
vennSelect(eset, design, x, contrast, fit, method = "same", adj.meth =
"BH", stat = "fstat", otherstats = c("pval","FC"), order.by = "pval",
foldFilt = NULL, save = FALSE, titleadd = NULL, ...)
```

Arguments

eset	A ExpressionSet object.
design	A design matrix, usually from a call to model.matrix. See details for more information.
x	A TestResults object, usually from a call to decideTests.
contrast	A contrasts matrix, produced either by hand, or by a call to makeContrasts
fit	An MArrayLM object, from a call to eBayes.
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.
stat	The statistic to report in the resulting HTML tables. Choices are 'fstat', 'tstat', and NULL. Ideally, the statistic chosen would correspond to the method used in decideTests. In other words, if one used methods such as 'separate' or 'hierarchical', which are based on a t-statistic, one should choose 'tstat', however, if one used 'nestedF', the logical choice would be 'fstat'.
otherstats	Other statistics to be included in the HTML tables. Choices include 'pval' and 'FC'.
order.by	Which statistic should be used to order the probesets? Choices include 'fstat', 'tstat', 'pval', and 'FC'. Note that if 'FC' is chosen and there are more than one set of fold changes, the first will be used.
foldFilt	A fold change to use for filtering. Default is NULL, meaning no filtering will be done.
save	Boolean - If TRUE, output a count of genes that fulfill the criteria. Useful for e.g., Sweave-type reports.
...	Used to pass other arguments to probes2table, in particular, to change the argument to anncols which controls the columns of hyperlinks to online databases (e.g., Entrez Gene, etc.). See aaf.handler for more information.
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.

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.

Calling the function normally will result in the output of HTML and text tables:

vennSelect(eset, fit, design, x)

Calling the function with save set to TRUE will output both HTML and text tables as well as a vector of counts for each comparison. This is useful when using the function programmatically (e.g., when making reports using Sweave).

out <- vennSelect(eset, fit, design, x, save = TRUE)

An alternative would be to use vennCounts2 and vennDiagram to output a Venn diagram, which is probably more reasonable since the tables being output are supposed to be based on a Venn diagram.

Value

Normally called only for the side effect of producing HTML and text tables. However, setting save to TRUE will output a vector of counts that can be used for making Sweave-style reports.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

---

vennSelectBM *Select and Output Genelists Based on Venn Diagrams using biomaRt*

Description

This function is designed to output HTML and text tables based on the results of a call to decideTests. This function is very similar to vennSelect, except it uses the biomaRt package to annotate genes, and the annotate package to create the HTML table.

Usage

```
vennSelectBM(eset, design, x, contrast, fit, method = "same", adj.meth = "BH",
stat = "fstat", otherstats = c("pval", "FC"), order.by = "pval",
foldFilt = NULL, save = FALSE, species, links = linksBM()[1:3],
otherdata = annBM()[1:3], ann.source = "entrezgene",
html = TRUE, text = TRUE, affyid = FALSE, ...)
```

Arguments

eset	A ExpressionSet object.
design	A design matrix, usually from a call to model.matrix. See details for more information.
x	A TestResults object, usually from a call to decideTests.
contrast	A contrasts matrix, produced either by hand, or by a call to makeContrasts
fit	An MArrayLM object, from a call to eBayes.
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.
stat	The statistic to report in the resulting HTML tables. Choices are 'fstat', 'tstat', and NULL. Ideally, the statistic chosen would correspond to the method used in decideTests. In other words, if one used methods such as 'separate' or 'hierarchical', which are based on a t-statistic, one should choose 'tstat', however, if one used 'nestedF', the logical choice would be 'fstat'.
otherstats	Other statistics to be included in the HTML tables. Choices include 'pval' and 'FC'.
order.by	Which statistic should be used to order the probesets? Choices include 'fstat', 'tstat', 'pval', and 'FC'. Note that if 'FC' is chosen and there are more than one set of fold changes, the first will be used.

foldFilt	A fold change to use for filtering. Default is NULL, meaning no filtering will be done.
save	Boolean - If TRUE, output a count of genes that fulfill the criteria. Useful for e.g., Sweave-type reports.
species	The species name. This must be in a particular format for biomaRt. An example for human is "hsapiens", or for mouse "mmusculus".
links	A character vector of things to annotate with hyperlinks to online databases. See linksBM for possible values.
otherdata	A character vector of things to annotate with text only (i.e., no hyperlinks). See annBM for possible values.
ann.source	The annotation source of the IDs that will be used to annotate the genes. The default value is "entrezgene". See details for other possibilities.
html	Boolean. Output HTML tables? Defaults to TRUE
text	Boolean. Output text tables? Defaulst to TRUE
affyid	Boolean. Are the IDs used to annotate these data Affymetrix IDs?
...	Used to pass other variables to e.g., htmlpage.

Details

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

The IDs that will be used to annotate the genes depend on the source of the data. If, for example, one is using an Affymetrix chip that doesn't have a BioC annotation package, then the IDs will be Affymetrix IDs. To find out the correct name to use for the ann.source argument, one can create a connection to a Biomart database using useMart and then deduce the correct argument by the output from listFilters(mart). It will usually be something starting with 'affy', and contain the name of the chip.

If one is using one of the re-mapped CDFs from MBNI at University of Michigan, then the IDs to use depend on the mapping used to create the CDF. At this time, only three types of CDFs can be used; EntrezGene, UniGene, and RefSeq. One can determine the correct ann.source argument by creating a connection to a Biomart database, and then calling linksBM(mart, linksBM())[[3]].

Some important things to note: First, the names of the HTML 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.

Calling the function normally will result in the output of HTML tables.

Calling the function with save set to TRUE will output HTML tables as well as a vector of counts for each comparison. This is useful when using the function programmatically (e.g., when making reports using Sweave).

out <- vennSelectBM(eset, fit, design, x, <other arguments>, save = TRUE)

An alternative would be to use vennCounts2 and vennDiagram to output a Venn diagram, which is probably more reasonable since the tables being output are supposed to be based on a Venn diagram.

Value

Normally called only for the side effect of producing HTML tables. However, setting save to TRUE will output a vector of counts that can be used for making Sweave-style reports.

Author(s)

James W. MacDonald <jmacdon@med.umich.edu>

---

vennSelectFC *Select and Output Gene Lists Based on Venn Diagrams*

Description

This function is designed to output text and/or HTML tables based on the results of a call to foldFilt. The general idea being that one might want to create a Venn diagram showing probesets that are unique to particular comparisons, or consistent between comparisons, and then might want to output the probesets that are contained in each cell of the Venn diagram.

Usage

vennSelectFC(eset, x, comps, order.by = "sum", method = "same", text = TRUE, html = TRUE, ...)

Arguments

eset	A ExpressionSet object.
x	An indicator matrix showing up or down regulation based on fold change, usually from a call to foldFilt. See details for more information.
comps	A list containing all the comparisons to be made. Each list item should be a vector of length two. This should be identical to the 'comps' argument used in the call to foldFilt See details for more information.
order.by	One of 'sum', 'max', 'median', or 'mean'. This orders the output for those tables that have multiple fold change values based on the summary statistic chosen. Defaults to 'sum'.
method	One of "same", "both", "up", "down", "sameup", or "samedown". See details for more information.
text	Boolean. Output text tables? Defaults to TRUE
html	Boolean. Output HTML tables? Defaults to TRUE
...	Used to pass other arguments to probes2table, in particular, to change the argument to anncols which controls the columns of hyperlinks to online databases (e.g., Entrez Gene, etc.). See aaf.handler for more information.

Details

The purpose of this function is to output the probesets listed in a Venn diagram that has been produced by a call to foldFilt. A small example would be as follows:

Assume an ExpressionSet exists that contains expression values for three Affymetrix chips, say a control, and two experimentals. One might want to know what probesets are different between each of the experimentals and the control, and those that are different between both of the experimentals and the control. We first make the comparisons, based on a fold change of 2 (or a difference of 1 on the log scale).

comps <- list(c(1,2), c(1,3))

This list indicates what comparisons we want. In this case 1vs2 and 1vs3.

out <- foldFilt(eset, fold = 1, groups = 1:3, comps = comps, compnames=c("Control vs experimental1", "Control vs experimental2"), save = TRUE)

By setting save = TRUE, we are saving a list, the first item being a vector of the number of probesets in each comparison, the second item being an indicator matrix showing up or down regulation based on a two-fold difference. We could make a Venn diagram using this matrix with vennCounts2 and vennDiagram. If we then wanted to output the probesets in each cell of that Venn diagram, we could use vennSelectFC as follows:

vennSelectFC(eset, out[[2]], comps)

One thing to note here is that the names of the resulting tables as well as the columns containing the fold change values will be extracted from the column names of the indicator matrix. This matrix will get its column names from the 'compnames' argument to foldFilt, so it is best to use reasonable names here. Also note that any character used in the 'compnames' argument that is not a valid character for a file name will be stripped out.

Value

Called only for the side effect of outputting HTML and/or text tables.

Author(s)

James W. MacDonald <jmacdon@med.umich.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, eset)

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.

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@med.umich.edu>

See Also

write.fit

Index