Package 'crossmeta'

October 17, 2020

Title Cross Platform Meta-Analysis of Microarray Data

Version 1.14.0

Author Alex Pickering

Maintainer Alex Pickering <alexvpickering@gmail.com>

Description Implements cross-platform and cross-species meta-analyses of Affymentrix, Illumina, and Agilent microarray data. This package automates common tasks such as downloading, normalizing, and annotating raw GEO data. The user then selects control and treatment samples in order to perform differential expression/pathway analyses for all comparisons. After analysing each contrast seperately, the user can select tissue sources for each contrast and specify any tissue sources that should be grouped for the subsequent meta-analyses. Finally, effect size and pathway meta-analyses can proceed and the results graphically explored.

Depends R (>= 3.5)

License MIT + file LICENSE

Encoding UTF-8

LazyData TRUE

RoxygenNote 6.1.1

VignetteBuilder knitr

Suggests knitr, rmarkdown, lydata, org.Hs.eg.db, testthat, ccdata

Imports affy (>= 1.52.0), affxparser (>= 1.46.0), AnnotationDbi (>= 1.36.2), Biobase (>= 2.34.0), BiocGenerics (>= 0.20.0), BiocManager (>= 1.30.4), ccmap, DT (>= 0.2), DBI (>= 1.0.0), data.table (>= 1.10.4), doParallel (>= 1.0.10), doRNG (>= 1.6), foreach (>= 1.4.3), fdrtool (>= 1.2.15), ggplot2 (>= 2.2.1), GEOquery (>= 2.40.0), limma (>= 3.30.13), matrixStats (>= 0.51.0), metaMA (>= 3.1.2), metap (>= 0.8), miniUI (>= 0.1.1), oligo (>= 1.38.0), plotly(>= 4.5.6), reshape (>= 0.8.6), reader(>= 1.0.6), RColorBrewer (>= 1.1.2), RCurl (>= 1.95.4.11), RSQLite (>= 2.1.1), rdrop2 (>= 0.7.0), stringr (>= 1.2.0), sva (>= 3.22.0), shiny (>= 1.0.0), stats (>= 3.3.3), XML (>= 3.98.1.17),

biocViews GeneExpression, Transcription, DifferentialExpression, Microarray, TissueMicroarray, OneChannel, Annotation, BatchEffect, Preprocessing, GUI

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

git_branch RELEASE_3_11

git_last_commit e954119

git_last_commit_date 2020-04-27

Date/Publication 2020-10-16

R topics documented:

`add_sources` *Add sample source information for meta-analysis.*

Description

User selects a tissue source for each contrast and indicates any sources that should be paired. This step is required if you would like to perform source-specific effect-size/pathway meta-analyses.

Usage

```
add_sources(diff_exprs, data_dir = getwd())
```

Arguments

`diff_exprs`	Previous result of `diff_expr`, which can be reloaded using `load_diff`.
`data_dir`	String specifying directory of GSE folders.

Details

The **Sources** tab is used to add a source for each contrast. To do so: click the relevant contrast rows, search for a source in the *Sample source* dropdown box, and then click the *Add* button.

The **Pairs** tab is used to indicate sources that should be paired (treated as the same source for subsequent effect-size and pathway meta-analyses). To do so: select at least two sources from the *Paired sources* dropdown box, and then click the *Add* button.

For each GSE, analysis results with added sources/pairs are saved in the corresponding GSE folder (in `data_dir`) that was created by `get_raw`.

Value

Same as `diff_expr` with added slots for each GSE in `diff_exprs`:

`sources`	Named vector specifying selected sample source for each contrast. Vector names identify the contrast.
`pairs`	List of character vectors indicating tissue sources that should be treated as the same source for subsequent effect-size and pathway meta-analyses.

Examples

```
library(lydata)

# load result of previous call to diff_expr:
data_dir  <- system.file("extdata", package = "lydata")
gse_names <- c("GSE9601", "GSE34817")
anals     <- load_diff(gse_names, data_dir)

# run shiny GUI to add tissue sources
# anals <- add_sources(anals, data_dir)
```

`contribute` *Contribute results of meta-analysis to public database.*

Description

Contributed results will be used to build a freely searchable database of gene expression meta-analyses.

Usage

```
contribute(diff_exprs, subject)
```

Arguments

`diff_exprs`	Result of call to `diff_expr`.
`subject`	String identifying meta-analysis subject (e.g. "rapamycin" or "prostate_cancer").

Details

Performs meta-analysis on `diff_exprs` using `es_meta`. Sends overall mean effect size values and minimal information needed to reproduce meta-analysis.

Value

NULL (used to contribute meta-analysis).

Examples

```
library(lydata)

# location of data
data_dir <- system.file("extdata", package = "lydata")

# gather GSE names
gse_names  <- c("GSE9601", "GSE15069", "GSE50841", "GSE34817", "GSE29689")

# load differential expression analyses
anals <- load_diff(gse_names, data_dir)

# contribute results of meta-analysis
# contribute(anals, subject = "LY294002")
```

`diff_expr` *Differential expression analysis of esets.*

Description

After selecting control and test samples for each contrast, surrogate variable analysis (sva) and differential expression analysis is performed.

Usage

```
diff_expr(esets, data_dir = getwd(), annot = "SYMBOL",
  prev_anals = list(NULL), svanal = TRUE)
```

Arguments

`esets`	List of annotated esets. Created by `load_raw`.
`data_dir`	String specifying directory of GSE folders.
`annot`	String, column name in fData common to all esets. For duplicated values in this column, the row with the highest interquartile range across selected samples will be kept. If meta-analysis will follow, appropriate values are "SYMBOL" (default - for gene level analysis) or, if all esets are from the same platform, "PROBE" (for probe level analysis).
`prev_anals`	Previous result of `diff_expr`, which can be reloaded using `load_diff`. If present, previous selections, names, and pairs will be reused.
`svanal`	Use surrogate variable analysis? Default is `TRUE`.

Details

The **Samples** tab is used to select control and test samples for each contrast. To do so: select rows for control samples, type a group name in the *Control group name* text input box and click the *Add Group* button. Repeat for test samples. While adding additional contrasts, a previous control group can be quickly reselected from the *Previous selections* dropdown box. After control and test samples have been added for all contrasts that you wish to include, click the *Done* button. Repeat for all GSEs.

Paired samples (e.g. the same subject before and after treatment) can be specified by selecting sample rows to pair and then clicking *Pair Samples*. The author does not usually specify paired samples and instead allows surrogate variable analysis to discover these inter-sample relationships from the data itself.

The **Contrasts** tab is used to view and delete contrasts that have already been added.

For each GSE, analysis results are saved in the corresponding GSE folder in `data_dir` that was created by `get_raw`. If analyses needs to be repeated, previous results can be reloaded with `load_diff` and supplied to the `prev_anals` parameter. In this case, previous selections, names, and pairs will be reused.

Value

List of named lists, one for each GSE. Each named list contains:

`pdata`	data.frame with phenotype data for selected samples. Columns `treatment` ('ctrl' or 'test'), `group`, and `pairs` are added based on user selections.
`top_tables`	List with results of `topTable` call (one per contrast). These results account for the effects of nuissance variables discovered by surrogate variable analysis.
`ebayes_sv`	Results of call to `eBayes` with surrogate variables included in the model matrix.
`annot`	Value of `annot` variable.

Examples

```
library(lydata)

# location of raw data
data_dir <- system.file("extdata", package = "lydata")

# gather GSE names
gse_names  <- c("GSE9601", "GSE15069", "GSE50841", "GSE34817", "GSE29689")

# load first eset
esets <- load_raw(gse_names[1], data_dir)

# run analysis
# anals <- diff_expr(esets, data_dir)

# re-run analysis on first eset
prev <- load_diff(gse_names[1], data_dir)
# anals <- diff_expr(esets[1], data_dir, prev_anals = prev)
```

`diff_path` *Differential expression of KEGG pathways.*

Description

Performs PADOG pathway analysis using KEGG database (downloaded Feb 2017).

Usage

```
diff_path(esets, prev_anals, data_dir = getwd())
```

Arguments

`esets`	List of annotated esets. Created by `load_raw`.
`prev_anals`	Previous result of `diff_expr`, which can be reloaded using `load_diff`.
`data_dir`	String specifying directory for GSE folders.

Details

If you wish to perform source-specific pathway meta-analyses, `add_sources` must be used before `diff_paths`.

For each GSE, analysis results are saved in the corresponding GSE folder in `data_dir` that was created by `get_raw`. PADOG outperforms other pathway analysis algorithms at prioritizing expected pathways (see references).

Value

List of named lists, one for each GSE. Each named list contains:

`padog_tables`	data.frames containing `padog` pathway analysis results for each contrast.

If `add_sources` is used first:

`sources`	Named vector specifying selected sample source for each contrast. Vector names identify the contrast.
`pairs`	List of character vectors indicating tissue sources that should be treated as the same source for subsequent pathway meta-analysis.

References

Tarca AL, Bhatti G, Romero R. A Comparison of Gene Set Analysis Methods in Terms of Sensitivity, Prioritization and Specificity. Chen L, ed. PLoS ONE. 2013;8(11):e79217. doi:10.1371/journal.pone.0079217.

Dong X, Hao Y, Wang X, Tian W. LEGO: a novel method for gene set over-representation analysis by incorporating network-based gene weights. Scientific Reports. 2016;6:18871. doi:10.1038/srep18871.

Examples

```
library(lydata)

# location of data
data_dir <- system.file("extdata", package = "lydata")

# gather GSE names
gse_names  <- c("GSE9601", "GSE15069", "GSE50841", "GSE34817", "GSE29689")

# load esets
esets <- load_raw(gse_names, data_dir)

# load previous differential expression analysis
anals <- load_diff(gse_names, data_dir)

# add tissue sources to perform seperate meta-analyses for each source (recommended)
# anals <- add_sources(anals)

# perform pathway analysis for each contrast
# path_anals <- diff_path(esets, anals, data_dir)
```

---

es_meta *Effect size combination meta analysis.*

Description

Performs effect-size meta-analyses across all studies and seperately for each tissue source.

Usage

```
es_meta(diff_exprs, cutoff = 0.3, by_source = FALSE)
```

Arguments

diff_exprs	Previous result of diff_expr, which can be reloaded using load_diff.
cutoff	Minimum fraction of contrasts that must have measured each gene. Between 0 and 1.
by_source	Should seperate meta-analyses be performed for each tissue source added with add_sources?

Details

Builds on zScores function from GeneMeta by allowing for genes that were not measured in all studies. This implementation also uses moderated unbiased effect sizes calculated by effectsize from metaMA and determines false discovery rates using fdrtool.

Value

A list of named lists, one for each tissue source. Each list contains two named data.frames. The first, `filt`, has all the columns below for genes present in cutoff or more fraction of contrasts. The second, `raw`, has only `dprime` and `vardprime` columns, but for all genes (NAs for genes not measured by a given contrast).

`dprime`	Unbiased effect sizes (one column per contrast).
`vardprime`	Variances of unbiased effect sizes (one column per contrast).
`mu`	Overall mean effect sizes.
`var`	Variances of overall mean effect sizes.
`z`	Overall z score = `mu / sqrt(var)`.
`fdr`	False discovery rates calculated from column `z` using fdrtool.
`pval`	p-values calculated from column `z` using fdrtool.

Examples

```
library(lydata)

# location of data
data_dir <- system.file("extdata", package = "lydata")

# gather GSE names
gse_names  <- c("GSE9601", "GSE15069", "GSE50841", "GSE34817", "GSE29689")

# load previous analysis
anals <- load_diff(gse_names, data_dir)

# add tissue sources to perform seperate meta-analyses for each source (optional)
# anals <- add_sources(anals, data_dir)

# perform meta-analysis
es <- es_meta(anals, by_source = TRUE)
```

`explore_paths` *Explore pathway meta analyses.*

Description

Shiny app for interactively exploring the results of effect-size and pathway meta-analyses. The app also interfaces with the ccmap package in order to explore drugs that are predicted to reverse or mimic your signature.

Usage

```
explore_paths(es_res, path_res, drug_info = NULL, type = c("both",
  "mimic", "reverse"))
```

Arguments

es_res	Result of call to es_meta.
path_res	Result of call to path_meta.
drug_info	Matrix of differential expression values for drugs (rows are genes, columns are drugs). If NULL (default), cmap_es is used.
type	Desired direction of drug action on query signature (see details). One of either 'both' (Default), 'mimic', or 'reverse'.

Details

For a given tissue source (top left dropdown box) and KEGG pathway (bottom left dropdown box, ordered by increasing false discovery rate), effect-sizes (y-axis) are plotted for each gene in the pathway (x-axis, ordered by decreasing asbsolute effect size).

For each gene, open circles give the effect-sizes for each contrast. The transparency of the open circles is proportional to the standard deviation of the effect-size for each contrast. For each gene, error bars give one standard deviation above and below the the overall meta-analysis effect-size.

The top drugs for the full signature in a given tissue (top right dropdown box, red points) and just the pathway genes (bottom right dropdown box, blue points) are orderered by decreasing (if type is 'both' or 'mimic') or increasing (if type is 'reverse') similarity. Positive and negative pearson correlations correspond to drugs that, respectively, mimic and reverse the query signature.

Drug effect sizes can be made visible by either clicking the legend entries (top left of plot) or selecting a new drug in the dropdown boxes.

When a new tissue source or pathway is selected, the top drug and pathway dropdown boxes are approriately updated.

Value

None

Examples

```
library(lydata)

data_dir  <- system.file("extdata", package = "lydata")
gse_names  <- c("GSE9601", "GSE15069", "GSE50841", "GSE34817", "GSE29689")

# load result of previous call to diff_expr:
es_anals <- load_diff(gse_names, data_dir)

# run shiny GUI to add tissue sources
# es_anals <- add_sources(es_anals, data_dir)

# perform effect-size meta-analyses for each tissue source
es_res <- es_meta(es_anals, by_source = TRUE)

# load result of previous call to diff_path:
# path_anals <- load_path(gse_names, data_dir)

# perform pathway meta-analyses for each tissue source
# path_res <- path_meta(path_anals, ncores = 1, nperm = 100, by_source = TRUE)

# explore pathway meta-analyses
```

```
# explore_paths(es_res, path_res)
```

`get_raw` *Download and unpack microarray supplementary files from GEO.*

Description

Downloads and unpacks microarray supplementary files from GEO. Files are stored in the supplied data directory under the GSE name.

Usage

```
get_raw(gse_names, data_dir = getwd())
```

Arguments

`gse_names`	Character vector of GSE names to download.
`data_dir`	String specifying directory for GSE folders.

Value

NULL (for download/unpack only).

See Also

`load_raw`.

Examples

```
get_raw("GSE41845")
```

`gs.names` *Map between KEGG pathway numbers and names.*

Description

Used to map human KEGG pathway numbers to names. Updated Feb 2017.

Usage

```
data(gs.names)
```

Format

An object of class `character` of length 310.

Value

A named character vector of human KEGG pathway names. Names of vector are KEGG pathway numbers.

gslist *KEGG human pathway genes.*

Description

Genes for human KEGG pathways. Updated Feb 2017.

Usage

```
data(gslist)
```

Format

An object of class `list` of length 310.

Value

A named list with entrez ids of genes for human KEGG pathways. List names are KEGG pathway numbers.

load_diff *Load previous differential expression analyses.*

Description

Loads previous differential expression analyses.

Usage

```
load_diff(gse_names, data_dir = getwd(), annot = "SYMBOL")
```

Arguments

`gse_names`	Character vector specifying GSE names to be loaded.
`data_dir`	String specifying directory of GSE folders.
`annot`	Level of previous analysis (e.g. "SYMBOL" or "PROBE").

Value

Result of previous call to `diff_expr`.

Examples

```
library(lydata)

data_dir <- system.file("extdata", package = "lydata")
gse_names <- c("GSE9601", "GSE34817")
prev <- load_diff(gse_names, data_dir)
```

load_path *Load previous pathway analyses.*

Description

Load previous pathway analyses.

Usage

```
load_path(gse_names, data_dir = getwd())
```

Arguments

gse_names	Character vector of GSE names.
data_dir	String specifying directory for GSE folders.

Value

Result of previous call to diff_path.

Examples

```
library(lydata)

# location of data
data_dir <- system.file("extdata", package = "lydata")

# gather GSE names
gse_names  <- c("GSE9601", "GSE15069", "GSE50841", "GSE34817", "GSE29689")

# load previous pathway analyses
# path_anals <- load_path(gse_names, data_dir)
```

load_raw *Load and annotate raw data downloaded from GEO.*

Description

Loads and annotates raw data previously downloaded with get_raw. Supported platforms include Affymetrix, Agilent, and Illumina.

Usage

```
load_raw(gse_names, data_dir = getwd(), gpl_dir = "..",
  overwrite = FALSE, ensql = NULL)
```

Arguments

gse_names	Character vector of GSE names.
data_dir	String specifying directory with GSE folders.
gpl_dir	String specifying parent directory to search for previously downloaded GPL.soft files.
overwrite	Do you want to overwrite saved esets from previous load_raw?
ensql	For development. Path to sqlite file with ENTREZID and SYMBOL columns created in data-raw/entrezdt.

Value

List of annotated esets.

Examples

```
library(lydata)
data_dir <- system.file("extdata", package = "lydata")
eset <- load_raw("GSE9601", data_dir = data_dir)
```

---

path_meta *Pathway p-value meta analysis.*

Description

Uses Fisher's method to combine p-values from PADOG pathway analyses.

Usage

```
path_meta(path_anals, ncores = parallel::detectCores(), nperm = ncores
  * 10000, by_source = FALSE)
```

Arguments

path_anals	Previous result of diff_path, which can be reloaded using load_path.
ncores	Number of cores to use. Default is all available.
nperm	Number of permutation to perform to calculate p-values.
by_source	Should seperate meta-analyses be performed for each tissue source added with add_sources?

Details

Permutation p-values are determined by shuffling pathway names associated with PADOG p-values prior to meta-analysis. Permutation p-values are then adjusted using the Benjamini & Hochberg method to obtain false discovery rates.

Value

A list of matrices, one for each tissue source. Each matrix contains a column of PADOG p-values for each contrast and permutation p- and fdr-values for the meta analysis.

See Also

sumlog, padog.

Examples

```
library(lydata)

# location of data
data_dir <- system.file("extdata", package = "lydata")

# gather GSE names
gse_names  <- c("GSE9601", "GSE15069", "GSE50841", "GSE34817", "GSE29689")

# load previous pathway analyses
# path_anals <- load_path(gse_names, data_dir)

# perform pathway meta analysis
# path_res <- path_meta(path_anals, ncores = 1, nperm = 100)
```

---

setup_prev *Setup selections when many samples.*

Description

Function is useful when number of samples makes manual selection with diff_expr error prone and time-consuming. This is often true for large clinical data sets.

Usage

```
setup_prev(eset, contrasts)
```

Arguments

eset	List containing one expression set with pData 'group' and 'pairs' (optional) columns. Name of eset should be the GSE name.
contrasts	Character vector specifying contrasts to analyse. Each contrast must take the form "B-A" where both "B" and "A" are present in eset pData 'group' column. "B" is the treatment group and "A" is the control group.

Value

List containing necessary information for prev_anal parameter of diff_expr.

Examples

```
library(lydata)
library(Biobase)

# location of raw data
data_dir <- system.file("extdata", package = "lydata")
```

```
# load eset
gse_name  <- c("GSE34817")
eset <- load_raw(gse_name, data_dir)

# inspect pData of eset
# View(pData(eset$GSE34817))  # if using RStudio
head(pData(eset$GSE34817))    # otherwise

# get group info from pData (differs based on eset)
group <- pData(eset$GSE34817)$characteristics_ch1.1

# make group names concise and valid
group <- gsub("treatment: ", "", group)
group <- make.names(group)

# add group to eset pData
pData(eset$GSE34817)$group <- group

# setup selections
sel <- setup_prev(eset, contrasts = "LY-DMSO")

# run differential expression analysis
# anal <- diff_expr(eset, data_dir, prev_anal = sel)
```

`symbol_annot` *Add hgnc symbol to expression set.*

Description

Function first maps entrez gene ids to homologous human entrez gene ids and then to hgnc symbols.

Usage

```
symbol_annot(eset, gse_name = "", ensql = NULL)
```

Arguments

`eset`	Expression set to annotate.
`gse_name`	GSE name for eset.
`ensql`	For development. Path to sqlite file with ENTREZID and SYMBOL columns created in data-raw/entrezdt.

Details

Initial entrez gene ids are obtained from bioconductor annotation data packages or from feature data of supplied expression set. Homologous human entrez ids are obtained from homologene and then mapped to hgnc symbols using org.Hs.eg.db. Expression set is expanded if 1:many mappings occur.

Value

Expression set with hgnc symbols ("SYMBOL") and row names ("PROBE") added to fData slot.

See Also

load_raw.

Examples

```
library(lydata)

# location of raw data
data_dir <- system.file("extdata", package = "lydata")

# load eset
eset <- load_raw("GSE9601", data_dir)[[1]]

# annotate eset (need if load_raw failed to annotate)
eset <- symbol_annot(eset)
```

---

which_max_iqr *Get row indices of maximum IQR within annotation groups*

Description

Groups by group_by and determines row with maximum IQR.

Usage

```
which_max_iqr(eset, groub_by, x = exprs(eset))
```

Arguments

eset	ExpressionSet
groub_by	Column in fData(eset) to group by
x	matrix of expression values to use for IQR

Value

Integer vector of row numbers representing rows with the maximum IQR after grouping by group_by

Index