# Package 'crossmeta'

April 14, 2017

**Title** Cross Platform Meta-Analysis of Microarray Data

**Version** 1.0.1

**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. A user interface makes it easy to select control and treatment samples for each contrast and study. This input is used for subsequent surrogate variable analysis (models unaccounted sources of variation) and differential expression analysis. Final meta-analysis of differential expression values can include genes measured in only a subset of studies.

**Depends** R (>= 3.3)

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** TRUE

**RoxygenNote** 5.0.1

**VignetteBuilder** knitr

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

**Imports** affy (>= 1.50.0), affxparser (>= 1.44.0), AnnotationDbi (>= 1.34.4), Biobase (>= 2.32.0), BiocGenerics (>= 0.18.0), BiocInstaller (>= 1.22.3), DT (>= 0.2), data.table (>= 1.9.6), fdrtool (>= 1.2.15), GEOquery (>= 2.38.4), limma (>= 3.28.17), matrixStats (>= 0.50.2), metaMA (>= 3.1.2), miniUI (>= 0.1.1), oligo (>= 1.36.1), pander (>= 0.6.0), RColorBrewer (>= 1.1.2), rdrop2 (>= 0.7.0), stringr (>= 1.0.0), sva (>= 3.20.0), shiny (>= 0.13.2)

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

**NeedsCompilation** no

## R topics documented:

---

`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 of esets.*

## Description

User selects contrasts, then surrogate variable analysis (sva) and differential expression analysis (limma) is performed.

## Usage

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

## 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. If Present, previous selections and names will be reused. |

## Details

For each GSE, analysis results are saved in the corresponding GSE folder (in data_dir) that was created by get_raw. If analysis 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 will be reused.

## Value

List of lists (one per GSE), each containing:

| | |
|---|---|
| eset | Expression set without expression or feature data. Treatment ('ctl' or 'test') and group columns have been added to the pData slot. Only selected samples kept. |
| 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. |

## See Also

get_raw, load_raw, and load_diff.

## 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)
```

---

`es_meta` *Effect size combination meta anaylsis.*

---

## Description

Builds on GeneMeta implementation by allowing for genes that were not measured in all studies.

## Usage

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

## Arguments

| | |
|---|---|
| `diff_exprs` | Result of previous call to `diff_expr`. |
| `cutoff` | Minimum fraction of contrasts that must have measured each gene. Between 0 and 1. |

## Details

In addition to allowing for genes that were not measured in all studies, this method uses moderated unbiased effect sizes calculated by `metaMA` and determines false discovery rates using `fdrtool`.

## Value

A list with 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 by `fdrtool`. |

## See Also

zScores, effectsize, 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)

# perform meta-analysis
es <- es_meta(anals)
```

---

# 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")
```

## load_diff *Load previous differential expression analysis.*

### Description

Previous runs of `diff_expr` are loaded.

### 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`.

### See Also

`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_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(), overwrite = FALSE)
```

### Arguments

| | |
|---|---|
| `gse_names` | Character vector of GSE names. |
| `data_dir` | String specifying directory with GSE folders. |
| `overwrite` | Do you want to overwrite saved esets? |

## Value

List of annotated esets.

## See Also

get_raw to obtain raw data.

## Examples

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

---

`open_raw_illum` *Open raw Illumina microarray files.*

---

## Description

Helper function to open raw Illumina microarray files in order to check that they are formatted correctly. For details on correct format, please see 'Checking Raw Illumina Data' in vignette.

## Usage

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

## Arguments

| | |
|---|---|
| `gse_names` | Character vector of Illumina GSE names to open. |
| `data_dir` | String specifying directory with GSE folders. |

## Value

Character vector of successfully formated Illumina GSE names.

## Examples

```
library(lydata)

# Illumina GSE names
illum_names <- c("GSE50841", "GSE34817", "GSE29689")

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

# open raw data files with default text editor
# open_raw_illum(illum_names)
```

setup_prev *Setup selections when many samples.*

## Description

Function useful when number of samples makes manual (GUI) selection error prone and time-consuming. Particularly useful 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 = "")
```

## Arguments

| | |
|---|---|
| `eset` | Expression set to annotate. |
| `gse_name` | GSE name for eset. |

## 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)
```

# Index