

# Package ‘crossmeta’

May 9, 2023

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

**Version** 1.26.0

**Author** Alex Pickering

**Maintainer** Alex Pickering <alexvpickering@gmail.com>

**Description** Implements cross-platform and cross-species meta-analyses of Affymetrix, 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 analyses for all comparisons. After analysing each contrast separately, the user can select tissue sources for each contrast and specify any tissue sources that should be grouped for the subsequent meta-analyses.

**Depends** R (>= 4.0)

**SystemRequirements** libxml2: libxml2-dev (deb), libxml2-devel (rpm)  
libcurl: libcurl4-openssl-dev (deb), libcurl-devel (rpm)  
openssl: libssl-dev (deb), openssl-devel (rpm), libssl\_dev  
(csw), openssl@1.1 (brew)

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** TRUE

**RoxygenNote** 7.1.2

**VignetteBuilder** knitr

**URL** <https://github.com/alexvpickering/crossmeta>

**BugReports** <https://github.com/alexvpickering/crossmeta/issues>

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

**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), DT (>= 0.2), DBI (>= 1.0.0), data.table (>= 1.10.4), edgeR, fdrtool (>= 1.2.15), GEOquery (>= 2.40.0), limma (>= 3.30.13), matrixStats (>= 0.51.0), metaMA (>= 3.1.2), miniUI (>= 0.1.1), methods, oligo (>= 1.38.0), reader (>= 1.0.6), RCurl (>= 1.95.4.11), RSQLite (>=

2.1.1), stringr ( $\geq 1.2.0$ ), sva ( $\geq 3.22.0$ ), shiny ( $\geq 1.0.0$ ), shinyjs ( $\geq 2.0.0$ ), shinyBS ( $\geq 0.61$ ), shinyWidgets ( $\geq 0.5.3$ ), shinypanel ( $\geq 0.1.0$ ), tibble, XML ( $\geq 3.98.1.17$ ), readxl ( $\geq 1.3.1$ )

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

**git\_url** <https://git.bioconductor.org/packages/crossmeta>

**git\_branch** RELEASE\_3\_17

**git\_last\_commit** b7438d5

**git\_last\_commit\_date** 2023-04-25

**Date/Publication** 2023-05-09

## R topics documented:

addContrastInput	3
add_adjusted	4
add_sources	4
add_vsd	5
bulkAnnot	6
bulkAnnotInput	6
bulkForm	6
bulkFormInput	7
bulkPage	7
bulkPageUI	7
bulkTable	8
bulkTableOutput	8
ch2_subset	8
clean_y	9
delContrastsInput	9
diff_expr	10
es_meta	11
exprs.MA	13
filter_genes	13
fit_ebayes	14
fit_lm	15
fix_illum_headers	15
format_dl_annot	16
format_up_annot	16
get_ch2_mod	16
get_group_levels	17
get_palette	17
get_raw	18
get_sva_mods	18
get_top_table	19
get_vsd	20

gs.names . . . . .	20
gslist . . . . .	21
ilmn.nnum . . . . .	21
iqr_replicates . . . . .	22
is_invertible . . . . .	22
load_agil_plat . . . . .	23
load_diff . . . . .	23
load_illum_plat . . . . .	24
load_plat . . . . .	24
load_raw . . . . .	25
makeExampleCountsEset . . . . .	26
match_prev_eset . . . . .	27
open_raw_illum . . . . .	27
phenoData.ch2 . . . . .	28
prefix_illum_headers . . . . .	28
query_ref . . . . .	29
remove_autonamed . . . . .	29
run_limma . . . . .	30
run_limma_setup . . . . .	31
run_lmfit . . . . .	31
run_select_contrasts . . . . .	32
run_sva . . . . .	32
setup_prev . . . . .	33
symbol_annot . . . . .	34
to_eset . . . . .	35
to_ma . . . . .	35
validate_up_annot . . . . .	36
which_max_iqr . . . . .	36
xls_to_txt . . . . .	37

**Index****38**


---

addContrastInput	<i>Add contrast input</i>
------------------	---------------------------

---

**Description**

Add contrast input

**Usage**

addContrastInput(id)

---

add_adjusted	<i>Add expression data adjusted for pairs/surrogate variables</i>
--------------	---

---

### Description

Add expression data adjusted for pairs/surrogate variables

### Usage

```
add_adjusted(eset, svobj = list(sv = NULL), numsv = 0)
```

### Arguments

eset	ExpressionSet
svobj	surrogate variable object
numsv	Number of surrogate variables to adjust for

### Value

eset with adjusted element added

---

add_sources	<i>Add sample source information for meta-analysis.</i>
-------------	---

---

### 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(), postfix = NULL)
```

### Arguments

diff_exprs	Previous result of <code>diff_expr</code> , which can be reloaded using <code>load_diff</code> .
data_dir	String specifying directory of GSE folders.
postfix	Optional string to append to saved results. Useful if need to run multiple meta-analyses on the same series but with different contrasts.

## 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`:

<code>sources</code>	Named vector specifying selected sample source for each contrast. Vector names identify the contrast.
<code>pairs</code>	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)
```

---

<code>add_vsd</code>	<i>Add VST normalized assay data element to expression set</i>
----------------------	--

---

## Description

For microarray datasets duplicates `exprs` slot into `vsd` slot.

## Usage

```
add_vsd(eset, rna_seq = TRUE)
```

## Arguments

<code>eset</code>	ExpressionSet with group column in <code>pData(eset)</code>
<code>rna_seq</code>	Is this an RNA-seq eset? Default is TRUE.

**Value**

eset with 'vsd' assayDataElement added.

---

bulkAnnot	<i>Logic for downloading and uploading bulk annotation</i>
-----------	--

---

**Description**

Logic for downloading and uploading bulk annotation

**Usage**

```
bulkAnnot(input, output, session, dataset_name, pdata)
```

---

bulkAnnotInput	<i>UI for Bulk Data annotation upload/download</i>
----------------	--

---

**Description**

UI for Bulk Data annotation upload/download

**Usage**

```
bulkAnnotInput(id)
```

---

bulkForm	<i>Logic for Bulk Data form</i>
----------	---------------------------------

---

**Description**

Logic for Bulk Data form

**Usage**

```
bulkForm(input, output, session, pdata, prev)
```

---

bulkFormInput	<i>Input form for Bulk Data page</i>
---------------	--------------------------------------

---

**Description**

Input form for Bulk Data page

**Usage**

```
bulkFormInput(id)
```

---

bulkPage	<i>Logic for Select Contrasts Interface</i>
----------	---

---

**Description**

Logic for Select Contrasts Interface

**Usage**

```
bulkPage(input, output, session, eset, gse_name, prev)
```

**Arguments**

input, output, session	shiny module boilerplate
eset	ExpressionSet
gse_name	GEO accession for the series.
prev	Previous result of diff_expr. Used to allow rechecking previous selections.

---

bulkPageUI	<i>UI for Select Contrasts Interface</i>
------------	--

---

**Description**

UI for Select Contrasts Interface

**Usage**

```
bulkPageUI(id)
```

**Arguments**

id	The id string to be namespaced.
----	---------------------------------

---

bulkTable	<i>Logic for pdata table</i>
-----------	------------------------------

---

**Description**

Logic for pdata table

**Usage**

```
bulkTable(input, output, session, eset, prev, up_annot)
```

---

bulkTableOuput	<i>Tables for datasets page</i>
----------------	---------------------------------

---

**Description**

Tables for datasets page

**Usage**

```
bulkTableOuput(id)
```

---

ch2_subset	<i>Subset for Paired Two-Channel ExpressionSet</i>
------------	--

---

**Description**

Two-channel eset use intraspotCorrelation and lmscFit so can't use duplicateCorrelation. If not using one channel in contrasts (e.g. because all reference RNA) and have paired design, better to treat as single channel so that can use duplicateCorrelation.

**Usage**

```
ch2_subset(eset, prev_anal)
```

**Arguments**

eset	Annotated ExpressionSet. Created by load_raw.
prev_anal	One item (for eset) from previous result of diff_expr.

**Value**

ExpressionSet. If two-channel, paired and one channel not used will subset to used channel.



---

clean_y	<i>Adjusts expression data for surrogate variables.</i>
---------	---

---

**Description**

Factors out effect of surrogate variables discovered during surrogate variable analysis.

**Usage**

```
clean_y(y, mod, mod.clean)
```

**Arguments**

y	Expression data of eset.
mod	Full model matrix supplied to sva.
mod.clean	Model matrix with factors to clean.

**Value**

Expression data with effects of svcs removed.

---

delContrastsInput	<i>Delete contrasts input</i>
-------------------	-------------------------------

---

**Description**

Delete contrasts input

**Usage**

```
delContrastsInput(id)
```

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,
  recheck = FALSE,
  postfix = NULL,
  port = 3838
)
```

**Arguments**

esets	List of annotated esets. Created by <a href="#">load_raw</a> .
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 <a href="#">diff_expr</a> , which can be reloaded using <a href="#">load_diff</a> . If present, previous selections, names, and pairs will be reused.
svanal	Use surrogate variable analysis? Default is TRUE.
recheck	Would you like to recheck previous group/contrast annotations? Requires <code>prev_anals</code> . Default is FALSE.
postfix	Optional string to append to saved results. Useful if need to run multiple meta-analyses on the same series but with different contrasts.
port	See <a href="#">runApp()</a> .

**Details**

Click the Download icon and fill in the *Group name* column and optionally the *Pairs* column. Then save and upload the filled in metadata csv. After doing so, select a test and control group to compare and click the + icon to add the contrast. Repeat previous step to add additional contrasts.

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 filling out the *Pairs column* before uploading the metadata.

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:

<code>pdata</code>	data.frame with phenotype data for selected samples. Columns <code>treatment</code> ('ctrl' or 'test'), <code>group</code> , and <code>pair</code> are added based on user selections.
<code>top_tables</code>	List with results of <code>topTable</code> call (one per contrast). These results account for the effects of nuisance variables discovered by surrogate variable analysis.
<code>ebayes_sv</code>	Results of call to <code>eBayes</code> with surrogate variables included in the model matrix.
<code>annot</code>	Value of <code>annot</code> 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 (opens GUI)
# anal_old <- 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 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 <code>diff_expr</code> , which can be reloaded using <code>load_diff</code> .
cutoff	Minimum fraction of contrasts that must have measured each gene. Between 0 and 1.
by_source	Should separate meta-analyses be performed for each tissue source added with <code>add_sources</code> ?

**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 = $\mu / \sqrt{\text{var}}$ .
fdr	False discovery rates calculated from column z using <code>fdrtool</code> .
pval	p-values calculated from column z using <code>fdrtool</code> .

**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 separate meta-analyses for each source (optional)
# anals <- add_sources(anals, data_dir)
```

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

---

exprs.MA	<i>Extract Log-Expression Matrix from MAList</i>
----------	--

---

### Description

Converts M and A-values to log-expression values. The output matrix will have two columns for each array, in the order all red then all green. Adapted from [plotDensities.MAList](#) instead of [exprs.MA](#) so that order is same as [phenoData.ch2](#).

### Usage

```
exprs.MA(MA)
```

### Arguments

MA                    an MAList object.

### Value

A numeric matrix with twice the columns of the input.

---

filter_genes	<i>Filter genes in RNA-seq ExpressionSet</i>
--------------	--

---

### Description

Uses [filterByExpr](#) to filter based on 'counts' assay or 'exprs' assay if 'counts' isn't available (for ARCHS4 data).

### Usage

```
filter_genes(eset)
```

### Arguments

eset                    ExpressionSet with 'counts' assayDataElement and group column in pData

### Value

filtered eset

**See Also**[filterByExpr](#)**Examples**

```
# example ExpressionSet
eset <- makeExampleCountsEset()
eset <- filter_genes(eset)
```

---

`fit_ebayes`*Fit ebayes model*

---

**Description**

Fit ebayes model

**Usage**

```
fit_ebayes(
  lm_fit,
  contrasts,
  robust = TRUE,
  trend = FALSE,
  allow.no.resid = FALSE
)
```

**Arguments**

<code>lm_fit</code>	Result of call to <a href="#">run_limma</a>
<code>contrasts</code>	Character vector of contrasts to fit.
<code>robust</code>	logical, should the estimation of <code>df.prior</code> and <code>var.prior</code> be robustified against outlier sample variances?
<code>trend</code>	logical, should an intensity-dependent trend be allowed for the prior variance? If FALSE then the prior variance is constant. Alternatively, <code>trend</code> can be a row-wise numeric vector, which will be used as the covariate for the prior variance.
<code>allow.no.resid</code>	Allow no residual degrees of freedom? if TRUE and the fit contrast matrix has no residual degrees of freedom, <a href="#">eBayes</a> fit is skipped and the result of <a href="#">contrasts.fit</a> is returned.

**Value**result of [eBayes](#)

---

fit_lm	<i>Run limma analysis.</i>
--------	----------------------------

---

### Description

Runs limma differential expression analysis on all contrasts selected by `add_contrast`. Analysis performed with and without surrogate variables discovered by `diff_setup`. Also prints MDS plot and saves results.

### Usage

```
fit_lm(eset, svobj = list(sv = NULL), numsv = 0, rna_seq = TRUE)
```

### Arguments

eset	Annotated eset created by <code>load_raw</code> . Replicate features and non-selected samples removed by <code>iqr_replicates</code> .
svobj	Surrogate variable analysis results. Returned from <code>run_sva</code> .
numsv	Number of surrogate variables to model.
rna_seq	Is this an RNA-seq eset? Default is TRUE.

### Value

list with slots: \* fit Result of `lmFit`. \* mod model matrix used for fit.

---

fix_illum_headers	<i>Attempts to fix Illumina raw data header</i>
-------------------	---

---

### Description

Reads raw data files and tries to fix them up so that they can be loaded by `read.ilmn`.

### Usage

```
fix_illum_headers(elist_paths, eset = NULL)
```

### Arguments

elist_paths	Path to Illumina raw data files. Usually contain patterns: <code>non_normalized.txt</code> , <code>raw.txt</code> , or <code>_supplementary_.txt</code>
eset	ExpressionSet from <code>getGEO</code> .

### Value

Character vector for annotation argument to `read.ilmn`. Fixed raw data files are saved with file-name ending in `_fixed.txt`

---

format_dl_annot	<i>Format downloaded annotation</i>
-----------------	-------------------------------------

---

**Description**

Format downloaded annotation

**Usage**

```
format_dl_annot(annot)
```

---

format_up_annot	<i>Format uploaded annotation</i>
-----------------	-----------------------------------

---

**Description**

Format uploaded annotation

**Usage**

```
format_up_annot(up, ref)
```

---

get_ch2_mod	<i>Get design matrix for two-channel array</i>
-------------	--

---

**Description**

Get design matrix for two-channel array

**Usage**

```
get_ch2_mod(eset)
```

**Arguments**

eset	ExpressionSet with colnames that end in '_red' and '_green' indicating channel and eset\$group indicating group membership.
------	---

**Value**

model matrix for use by [inraspotCorrelation](#) and [lmscFit](#)



---

<code>get_group_levels</code>	<i>Get group levels for bulk data plots</i>
-------------------------------	---

---

**Description**

Get group levels for bulk data plots

**Usage**

```
get_group_levels(pdata)
```

**Arguments**

<code>pdata</code>	Data.frame of phenotype data
--------------------	------------------------------

---

<code>get_palette</code>	<i>Get a Pallete to Distinguish Groups</i>
--------------------------	--

---

**Description**

Get a Pallete to Distinguish Groups

**Usage**

```
get_palette(levs, dark = FALSE, with_all = FALSE)
```

**Arguments**

<code>levs</code>	Character vector of levels to get colour pallete for.
-------------------	---

**Value**

Character vector with colour codes of `length(levs)`.

---

get_raw	<i>Download and unpack microarray supplementary files from GEO.</i>
---------	---

---

**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")
```

---

get_sva_mods	<i>Get model matrices for surrogate variable analysis</i>
--------------	---

---

**Description**

Used by `add_adjusted` to create model matrix with surrogate variables.

**Usage**

```
get_sva_mods(pdata)
```

**Arguments**

pdata	data.frame of phenotype data with column 'group' and 'pair' (optional).
-------	---

**Value**

List with model matrix(mod) and null model matrix (mod0) used for sva.

---

get_top_table	<i>Get top table</i>
---------------	----------------------

---

### Description

Get top table

### Usage

```
get_top_table(  
  lm_fit,  
  groups = c("test", "ctrl"),  
  with.es = TRUE,  
  robust = FALSE,  
  trend = FALSE,  
  allow.no.resid = FALSE  
)
```

### Arguments

lm_fit	Result of <a href="#">run_limma</a>
groups	Test and Control group as strings.
with.es	Add 'dprime' and 'vardprime' from <a href="#">effectsize</a> ? Default is TRUE.
robust	logical, should the estimation of <code>df.prior</code> and <code>var.prior</code> be robustified against outlier sample variances?
trend	logical, should an intensity-dependent trend be allowed for the prior variance? If FALSE then the prior variance is constant. Alternatively, <code>trend</code> can be a row-wise numeric vector, which will be used as the covariate for the prior variance.
allow.no.resid	Allow no residual degrees of freedom? if TRUE and the fit contrast matrix has no residual degrees of freedom, <a href="#">eBayes</a> fit is skipped and the result of <a href="#">contrasts.fit</a> is returned.

### Value

result of [toptable](#)

---

`get_vsd`*Get variance stabilized data for exploratory data analysis*

---

**Description**

Get variance stabilized data for exploratory data analysis

**Usage**

```
get_vsd(eset)
```

**Arguments**

`eset` ExpressionSet loaded with `load_raw`.

**Value**

matrix with variance stabilized expression data.

---

`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.

---

gslis	<i>KEGG human pathway genes.</i>
-------	----------------------------------

---

**Description**

Genes for human KEGG pathways. Updated Feb 2017.

**Usage**

```
data(gslis)
```

**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.

---

ilmn.nnum	<i>Count numeric columns in raw Illumina data files</i>
-----------	---

---

**Description**

Excludes probe ID cols

**Usage**

```
ilmn.nnum(elist_paths)
```

**Arguments**

`elist_paths` Paths to raw illumina data files

**Value**

Number of numeric columns in `elist_paths` excluding probe ID columns.

---

iqr_replicates	<i>Removes features with replicated annotation.</i>
----------------	---

---

### Description

For rows with duplicated annot, highested IQR retained.

### Usage

```
iqr_replicates(eset, annot = "SYMBOL", rm.dup = FALSE)
```

### Arguments

eset	Annotated eset created by load_raw.
annot	feature to use to remove replicates.
rm.dup	remove duplicates (same measure, multiple ids)? Used for Pathway analysis so that doesn't treat probes that map to multiple genes as distinct measures.

### Value

Expression set with unique features at probe or gene level.

---

is_invertible	<i>Check uploaded bulk pdata to make sure the study design is invertible</i>
---------------	--

---

### Description

Check uploaded bulk pdata to make sure the study design is invertible

### Usage

```
is_invertible(pdata)
```

---

load_agil_plat	<i>Load Agilent raw data</i>
----------------	------------------------------

---

**Description**

Load Agilent raw data

**Usage**

```
load_agil_plat(eset, gse_name, gse_dir, ensql)
```

**Arguments**

eset	ExpressionSet from <a href="#">getGEO</a> .
gse_name	Accession name for eset.
gse_dir	Direction with Agilent raw data.
ensql	For development. Path to sqlite file with ENTREZID and SYMBOL columns created in data-raw/entrezdt.

**Value**

ExpressionSet

---

load_diff	<i>Load previous differential expression analyses.</i>
-----------	--

---

**Description**

Loads previous differential expression analyses.

**Usage**

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

**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").
postfix	Optional string to append to saved results. Useful if need to run multiple meta-analyses on the same series but with different contrasts.

**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_illum_plat	<i>Illumina loader utility for load_plat.</i>
-----------------	---

---

**Description**

Used by load\_plat to load an eset.

**Usage**

```
load_illum_plat(eset, gse_name, gse_dir, ensql)
```

**Arguments**

eset	Expression set obtained by getGEO.
gse_name	String specifying GSE name.
gse_dir	String specifying path to GSE folder.

**Value**

Annotated eset.

**See Also**

[load\\_plat](#).

---

load_plat	<i>Load and pre-process raw Affymetrix, Illumina, and Agilent microarrays.</i>
-----------	--

---

**Description**

Load raw files previously downloaded with get\_raw. Used by load\_raw.

**Usage**

```
load_plat(gse_name, data_dir, gpl_dir, ensql)
```



**Arguments**

gse_name	GSE names.
data_dir	String specifying directory with GSE folder.
gpl_dir	String specifying parent directory to search for previously downloaded GPL.soft files.
ensql	For development. Path to sqlite file with ENTREZID and SYMBOL columns created in data-raw/entrezdt.

**Details**

Data is normalized, SYMBOL and PROBE annotation are added to fData slot.

**Value**

List of annotated esets, one for each platform in gse\_name.

**See Also**

[get\\_raw](#) to obtain raw data.

---

load_raw	<i>Load and annotate raw data downloaded from GEO.</i>
----------	--

---

**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)
```

---

makeExampleCountsEset *Make example ExpressionSet*

---

**Description**

adapted from DESeq2::makeExampleDESeqDataSet

**Usage**

```
makeExampleCountsEset(
  n = 1000,
  m = 12,
  betaSD = 0,
  interceptMean = 4,
  interceptSD = 2,
  dispMeanRel = function(x) 4/x + 0.1,
  sizeFactors = rep(1, m)
)
```

**Arguments**

n	number of rows
m	number of columns
betaSD	the standard deviation for non-intercept betas, i.e. $\beta \sim N(0, \text{betaSD})$
interceptMean	the mean of the intercept betas (log <sub>2</sub> scale)
interceptSD	the standard deviation of the intercept betas (log <sub>2</sub> scale)
dispMeanRel	a function specifying the relationship of the dispersions on $2^{\text{trueIntercept}}$
sizeFactors	multiplicative factors for each sample

**Examples**

```
eset <- makeExampleCountsEset()
```

---

match_prev_eset	<i>Reuse contrast selections from previous analysis.</i>
-----------------	--

---

**Description**

Transfers user-supplied selections from previous call of `diff_expr`.

**Usage**

```
match_prev_eset(eset, prev_anal)
```

**Arguments**

<code>eset</code>	Annotated eset. Created by <code>load_raw</code> .
<code>prev_anal</code>	One item (for <code>eset</code> ) from previous result of <code>diff_expr</code> . If present, previous selections and names will be reused.

**Value**

Expression set with samples and `pData` as in `prev_anal`.

**See Also**

[diff\\_expr](#)

---

<code>open_raw_illum</code>	<i>Open raw Illumina microarray files.</i>
-----------------------------	--

---

**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**

<code>gse_names</code>	Character vector of Illumina GSE names to open.
<code>data_dir</code>	String specifying directory with GSE folders.

**Value**

Character vector of successfully formatted 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)
```

---

phenoData.ch2

*Construct AnnotatedDataFrame from Two-Channel ExpressionSet*

---

**Description**

Construct AnnotatedDataFrame from Two-Channel ExpressionSet

**Usage**

```
phenoData.ch2(eset)
```

**Arguments**

eset                    ExpressionSet with pData for two-channel Agilent array.

**Value**

AnnotatedDataFrame with twice as many rows as eset, one for each channel of each array in order all red then all green.

---

prefix\_illum\_headers    *Run prefix on Illumina raw data files*

---

**Description**

Run prefix on Illumina raw data files

**Usage**

```
prefix_illum_headers(elist_paths)
```

**Arguments**

elist\_paths            Paths to raw Illumina data files

**Value**

Paths to fixed versions of `elist_paths`

---

query_ref	<i>Get correlation between query and reference signatures.</i>
-----------	--

---

**Description**

Determines the pearson correlation between the query and each reference signature.

**Usage**

```
query_ref(query, ref, sorted = TRUE, ngenes = 200)
```

**Arguments**

query	Named numeric vector of differential expression values for query genes. Usually 'meta' slot of <code>get_dprimes</code> result.
ref	A matrix of differential expression to query against (rows are genes, columns are samples).
sorted	Would you like the results sorted by decreasing similarity? Default is TRUE.
ngenes	The number of top differentially-regulated (up and down) query genes to use.

**Value**

Vector of pearson correlations between query and reference signatures.

---

remove_automated	<i>Remove columns that are automated by data.table</i>
------------------	--

---

**Description**

Auto-named columns start with 'V' followed by the column number.

**Usage**

```
remove_automated(ex)
```

**Arguments**

ex	data.frame loaded with <a href="#">fread</a>
----	--

**Value**

ex with auto-named columns removed.

---

run_limma	<i>Linear model fitting of eset with limma.</i>
-----------	---

---

### Description

After selecting control and test samples for a contrast, surrogate variable analysis ([sva](#)) and linear model fitting with [lmFit](#) is performed.

### Usage

```
run_limma(  
  eset,  
  annot = "SYMBOL",  
  svobj = list(sv = NULL),  
  numsv = 0,  
  filter = TRUE  
)
```

### Arguments

eset	Annotated eset created by <a href="#">load_raw</a> .
annot	String, column name in fData. For duplicated values in this column, the row with the highest interquartile range across selected samples will be kept. Appropriate values are "SYMBOL" (default - for gene level analysis) or "ENTREZID_HS" (for probe level analysis).
svobj	Surrogate variable analysis results. Returned from <a href="#">run_sva</a> .
numsv	Number of surrogate variables to model.
filter	For RNA-seq. Should genes with low counts be filtered? <a href="#">dseqr shiny app</a> performs this step separately. Should be TRUE (default) if used outside of <a href="#">dseqr shiny app</a> .

### Details

If analyses need to be repeated, previous results can be reloaded with [readRDS](#) and supplied to the `prev_anal` parameter. In this case, previous selections will be reused.

### Value

List with:

fit	result of <a href="#">lmFit</a> .
mod	<code>model.matrix</code> used for fit

---

run_limma_setup	<i>Setup ExpressionSet for running limma analysis</i>
-----------------	---

---

**Description**

Setup ExpressionSet for running limma analysis

**Usage**

```
run_limma_setup(eset, prev)
```

**Arguments**

eset	ExpressionSet
prev	previous result of call to diff_expr

**Value**

eset ready for run\_limma

---

run_lmfit	<i>Perform lmFit analysis from limma.</i>
-----------	---

---

**Description**

If paired samples, runs [duplicateCorrelation](#) to estimate intra-patient variance.

**Usage**

```
run_lmfit(eset, mod, rna_seq = TRUE)
```

**Arguments**

eset	Annotated eset created by load_raw. Non-selected samples and duplicate features removed by add_contrasts and iqr_replicates.
mod	Model matrix generated by diff_setup. With or without surrogate variables.
rna_seq	Is this an RNA-seq eset? Default is TRUE.

**Value**

result from call to limma lmFit.

---

run\_select\_contrasts *Shiny gadget to upload groups and select contrasts*

---

### Description

Shiny gadget to upload groups and select contrasts

### Usage

```
run_select_contrasts(
  eset,
  gse_name,
  prev = NULL,
  app_dir = system.file("select_contrasts", package = "crossmeta", mustWork = TRUE),
  port = 3838
)
```

### Arguments

eset	ExpressionSet
gse_name	GEO accession for the series.
prev	Previous result of diff_expr. Used to allow rechecking previous selections.
app_dir	Directory to shiny app. For local development use 'inst/select_contrasts'. Default is in 'select_contrasts' sub directory of crossmeta package.
port	See <a href="#">runApp()</a> .

### Value

result of [setup\\_prev](#). Used to specify sample groups and contrasts for differential expression analysis.

---

run\_sva *Run surrogate variable analysis*

---

### Description

Run surrogate variable analysis

### Usage

```
run_sva(mods, eset, svanal = TRUE)
```



**Arguments**

mods	result of <a href="#">get_sva_mods</a>
eset	ExpressionSet
svanal	Should surrogate variable analysis be run? If FALSE, returns dummy result.

---

setup_prev	<i>Setup selections when many samples.</i>
------------	--

---

**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 'pair' (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	<i>Add hgnc symbol to expression set.</i>
--------------	---

---

## 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 homogene 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)
```

---

to_eset	<i>Convert limma object to ExpressionSet</i>
---------	--

---

**Description**

Convert limma object to ExpressionSet

**Usage**

```
to_eset(object, eset)
```

**Arguments**

object	an EList of MAlist object containing expression data.
eset	ExpressionSet from <a href="#">getGEO</a> . Used for annotation.

**Value**

ExpressionSet using expression data from object and annotation from eset.

---

to_ma	<i>Covert expression values to MAlist</i>
-------	---

---

**Description**

Covert expression values to MAlist

**Usage**

```
to_ma(y)
```

**Arguments**

y	Expression values from two-channel agilent array in order all red then all green.
---	---

**Value**

MAList

**Examples**

```
A <- matrix(rnorm(100), ncol = 5)
M <- matrix(rnorm(100), ncol = 5)
MA <- new('MAList', list(M=M, A=A))
colnames(MA) <- letters[1:5]
```

```
y <- exprs.MA(MA)
MA2 <- crossmeta:::to_ma(y)
all.equal(MA, MA2)
```

---

validate_up_annot	<i>Validate uploaded bulk annotation</i>
-------------------	--

---

**Description**

Validate uploaded bulk annotation

**Usage**

```
validate_up_annot(up, ref)
```

---

which_max_iqr	<i>Get row indices of maximum IQR within annotation groups</i>
---------------	--

---

**Description**

Groups by group\_by and determines row with maximum IQR.

**Usage**

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

**Arguments**

eset	ExpressionSet
group_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

---

xls_to_txt	<i>Covert .xls files to .txt</i>
------------	----------------------------------

---

**Description**

For converting Illumina \_Supplementary\_\*.xls files to .txt for load\_illum\_plat.

**Usage**

```
xls_to_txt(xls_paths)
```

**Arguments**

xls_paths	Paths to .xls files
-----------	---------------------

# Index

- \* <-
  - run\_select\_contrasts, 32
- \* =
  - run\_select\_contrasts, 32
- \* **GSE**
  - run\_select\_contrasts, 32
- \* #
  - run\_select\_contrasts, 32
- \* **analysis**
  - run\_select\_contrasts, 32
- \* **data\_dir**[[1]]
  - run\_select\_contrasts, 32
- \* **data\_dir**
  - run\_select\_contrasts, 32
- \* **datasets**
  - gs.names, 20
  - gslist, 21
- \* **data**
  - run\_select\_contrasts, 32
- \* **eset**
  - run\_select\_contrasts, 32
- \* **gather**
  - run\_select\_contrasts, 32
- \* **gse\_name**
  - run\_select\_contrasts, 32
- \* **gse\_name**
  - run\_select\_contrasts, 32
- \* **internal**
  - addContrastInput, 3
  - bulkAnnot, 6
  - bulkAnnotInput, 6
  - bulkForm, 6
  - bulkFormInput, 7
  - bulkTable, 8
  - bulkTableOuput, 8
  - ch2\_subset, 8
  - clean\_y, 9
  - delContrastsInput, 9
  - fit\_lm, 15
  - format\_dl\_annot, 16
  - format\_up\_annot, 16
  - get\_ch2\_mod, 16
  - get\_group\_levels, 17
  - get\_palette, 17
  - is\_invertible, 22
  - load\_illum\_plat, 24
  - load\_plat, 24
  - match\_prev\_eset, 27
  - query\_ref, 29
  - run\_lmfit, 31
  - run\_select\_contrasts, 32
  - to\_eset, 35
  - to\_ma, 35
  - validate\_up\_annot, 36
  - which\_max\_iqr, 36
  - xls\_to\_txt, 37
- \* **library(lydata)**
  - run\_select\_contrasts, 32
- \* **load\_raw(gse\_name,**
  - run\_select\_contrasts, 32
- \* **load**
  - run\_select\_contrasts, 32
- \* **location**
  - run\_select\_contrasts, 32
- \* **lydata)**
  - run\_select\_contrasts, 32
- \* **names**
  - run\_select\_contrasts, 32
- \* **of**
  - run\_select\_contrasts, 32
- \* **package**
  - run\_select\_contrasts, 32
- \* **previous**
  - run\_select\_contrasts, 32
- \* **run\_select\_contrasts(eset,**
  - run\_select\_contrasts, 32
- \* **system.file(extdata,**
  - run\_select\_contrasts, 32

\* **z**  
    run\_select\_contrasts, 32

add\_adjusted, 4  
add\_sources, 4, 12  
add\_vsd, 5  
addContrastInput, 3

bulkAnnot, 6  
bulkAnnotInput, 6  
bulkForm, 6  
bulkFormInput, 7  
bulkPage, 7  
bulkPageUI, 7  
bulkTable, 8  
bulkTableOutput, 8

ch2\_subset, 8  
clean\_y, 9  
contrasts.fit, 14, 19

delContrastsInput, 9  
diff\_expr, 4, 5, 10, 10, 12, 23, 27, 33  
duplicateCorrelation, 31

eBayes, 11, 14, 19  
effectsize, 12, 19  
es\_meta, 11  
exprs.MA, 13, 13

fdrtool, 12  
filter\_genes, 13  
filterByExpr, 13, 14  
fit\_ebayes, 14  
fit\_lm, 15  
fix\_illum\_headers, 15  
format\_dl\_annot, 16  
format\_up\_annot, 16  
fread, 29

get\_ch2\_mod, 16  
get\_group\_levels, 17  
get\_palette, 17  
get\_raw, 5, 11, 18, 25  
get\_sva\_mods, 18, 33  
get\_top\_table, 19  
get\_vsd, 20  
getGEO, 15, 23, 35  
gs.names, 20  
gslist, 21

ilmn.nnum, 21  
intraspotCorrelation, 16  
iqr\_replicates, 22  
is\_invertible, 22

lmFit, 15, 30  
lmscFit, 16  
load\_agil\_plat, 23  
load\_diff, 4, 10–12, 23  
load\_illum\_plat, 24  
load\_plat, 24, 24  
load\_raw, 10, 18, 20, 25, 34

makeExampleCountsEset, 26  
match\_prev\_eset, 27

open\_raw\_illum, 27

phenoData.ch2, 13, 28  
plotDensities.MAList, 13  
prefix\_illum\_headers, 28

query\_ref, 29

read.ilmn, 15  
readRDS, 30  
remove\_autonamed, 29  
run\_limma, 14, 19, 30  
run\_limma\_setup, 31  
run\_lmfit, 31  
run\_select\_contrasts, 32  
run\_sva, 15, 30, 32  
runApp(), 10, 32

setup\_prev, 32, 33  
sva, 10, 30  
symbol\_annot, 34

to\_eset, 35  
to\_ma, 35  
topTable, 11  
toptable, 19

validate\_up\_annot, 36

which\_max\_iqr, 36

xls\_to\_txt, 37

zScores, 12