Package ‘DeeDeeExperiment’

May 5, 2026

Title DeeDeeExperiment: An S4 Class for managing and exploring omics analysis results

Version 1.3.0

Description DeeDeeExperiment is an S4 class extending the SingleCellExperiment class, designed to integrate and manage omics analysis results. It introduces two dedicated slots to store Differential Expression Analysis (DEA) results and Functional Enrichment Analysis (FEA) results, providing a structured approach for downstream analysis.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Depends R (>= 4.5.0), SingleCellExperiment

Imports SummarizedExperiment, methods, S4Vectors, utils, DESeq2, edgeR, limma, writexl, cli

Suggests macrophage, knitr, BiocStyle, apeglm, mosdef, org.Hs.eg.db, topGO, clusterProfiler, DEFormats, ExperimentHub, scater, muscat, testthat (>= 3.0.0)

VignetteBuilder knitr

URL `https://github.com/imbeimainz/DeeDeeExperiment`

BugReports `https://github.com/imbeimainz/DeeDeeExperiment/issues`

biocViews Software, Infrastructure, DataRepresentation, GeneExpression, Transcription, Transcriptomics, DifferentialExpression, Pathways, GO

Config/testthat/edition 3

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

git_branch devel

git_last_commit 9bf5427

git_last_commit_date 2026-04-28

Repository Bioconductor 3.24

Date/Publication 2026-05-05

Author Najla Abassi [aut, cre] (ORCID: <https://orcid.org/0000-0001-8357-0938>), Lea Schwarz [aut] (ORCID: <https://orcid.org/0000-0003-2554-0583>), Federico Marini [aut] (ORCID: <https://orcid.org/0000-0003-3252-7758>)

Maintainer Najla Abassi <`abassi.nejla96@gmail.com`>

Contents

`clusterPro_res` `clusterPro_res`

Description

A list of FE results generated with `clusterProfiler::enrichGO()`

Format

A list of `enrichResult` objects

Details

A list of FE result for the macrophage data (salmonella vs naive and IFNg vs naive)

Value

A list

See Also

Other enrich_resus: `enrichr_res`, `fgseaRes`, `gost_res`, `gsea_res`, `topGO_results_list`

`dd_list_original` *dd_list_original*

Description

A list of `deedee_prepared` DE results.

Format

A list object

Details

documented creation in ... script

Value

A list of DE results

`DeeDeeExperiment-class` *The DeeDeeExperiment class*

Description

The `DeeDeeExperiment` class is integrate and manage omics analysis results. It inherits from the `SingleCellExperiment` class, and additionally stores DE-related/functional enrichment information via dedicated slots and `rowData`.

Usage

```
DeeDeeExperiment(
  sce = SingleCellExperiment(),
  de_results = NULL,
  enrich_results = NULL
)
```

Arguments

`sce`	A `SingleCellExperiment` object, that will be used as a scaffold to store the DE related information.
`de_results`	A named list of DE results, in any of the formats supported by the package (currently: results from `DESeq2`, `edgeR`, `limma`, `muscat`).
`enrich_results`	A named list of functional enrichment results. Each element can be either a data.frame (currently supports results from `topGO`, `enrichR`, `gProfiler`, `fgsea`, `gsea`, `DAVID`, and output of `GeneTonic` shakers), or an `enrichResult/gseaResult` objects (currently supports `clusterProfiler`)

Details

The `sce` parameter can be optionally left unspecified. If this is the case, the resulting `DeeDeeExperiment` object will contain as features the ones specified by the provided components of the object supplied via the `de_results` parameter.

The conversion of the components of the `de_results` list will be handled via conversion functions to uniform the names and set of information which will be stored in the returned `DeeDeeExperiment` object. The names of the list will be used to define the `contrasts` for the different DE analyses included, which will determine the way to access the information stored in the `dea` slot of the `DeeDeeExperiment` object.

The content of the `enrich_results` provided by the user will be validated to ensure that it is properly formatted and correctly named. The FE tool can be automatically detected, and based on that, the appropriate shaking method is used to return a standardized format of the FEA results. The names of the list will be used to attempt to associate each enrichment result with a corresponding DE contrast stored in the `DeeDeeExperiment` object, but it also can be defined by the user.

Since a `DeeDeeExperiment` is also a `SummarizedExperiment` object, it can be seamlessly provided downstream for visualization and in-depth exploration to packages such as `iSEE` or similar.

Value

A `DeeDeeExperiment` object.

Slots

`dea` This slot is designed to hold the DE-related information. This is internally being created upon importing from the list of DE results objects, provided when instantiating the DeeDeeExperiment.

`fea` This slot is designed to hold Functional Enrichment related information.

Author(s)

Najla Abassi, Lea Schwarz, and Federico Marini

Examples

```
data("de_named_list", package = "DeeDeeExperiment")

dde_onlyde <- DeeDeeExperiment(
  de_results = de_named_list
)

# or, with a SE object as support - even without assay data available
```

```
library("SummarizedExperiment")

rd_macrophage <- DataFrame(
  gene_id = rownames(de_named_list$ifng_vs_naive)
)
rownames(rd_macrophage) <- rownames(de_named_list$ifng_vs_naive)
se_macrophage_noassays <- SummarizedExperiment(
  assays = SimpleList(),
  rowData = rd_macrophage
)

dde <- DeeDeeExperiment(
  se_macrophage_noassays,
  de_results = de_named_list
)
```

`DeeDeeExperiment-methods`

Methods for DeeDeeExperiment objects

Description

The `DeeDeeExperiment()` class provides a family of methods to get and set DE-related information and functional enrichment results in DeeDeeExperiment objects.

Usage

```
## S4 method for signature 'DeeDeeExperiment'
getDEAInfo(x)

## S4 replacement method for signature 'DeeDeeExperiment'
getDEAInfo(x) <- value

## S4 method for signature 'DeeDeeExperiment'
getDEANames(x)

## S4 method for signature 'DeeDeeExperiment'
renameDEA(x, old_name, new_name)

## S4 method for signature 'DeeDeeExperiment'
addDEA(x, dea, force = FALSE)

## S4 method for signature 'DeeDeeExperiment'
removeDEA(x, dea_name, remove_linked_fea = FALSE)

## S4 method for signature 'DeeDeeExperiment'
getDEA(
  x,
  dea_name = NULL,
  format = c("minimal", "original"),
  extra_rd = NULL,
  type = c("DFrame", "data.frame"),
```

```
  verbose = FALSE
)

## S4 method for signature 'DeeDeeExperiment'
getDEAList(x, format = c("minimal", "original"), verbose = FALSE)

## S4 method for signature 'DeeDeeExperiment'
addScenarioInfo(x, dea_name, info = NULL, force = FALSE)

## S4 method for signature 'DeeDeeExperiment'
getFEAInfo(x)

## S4 replacement method for signature 'DeeDeeExperiment'
getFEAInfo(x) <- value

## S4 method for signature 'DeeDeeExperiment'
getFEANames(x)

## S4 method for signature 'DeeDeeExperiment'
renameFEA(x, old_name, new_name)

## S4 method for signature 'DeeDeeExperiment'
addFEA(
  x,
  fea,
  de_name = NA_character_,
  fe_name = NULL,
  fea_tool = "auto",
  force = FALSE,
  verbose = FALSE
)

## S4 method for signature 'DeeDeeExperiment'
removeFEA(x, fea_name)

## S4 method for signature 'DeeDeeExperiment'
getFEA(x, fea_name = NULL, format = c("minimal", "original"), verbose = FALSE)

## S4 method for signature 'DeeDeeExperiment'
getFEAList(
  x,
  dea_name = NULL,
  format = c("minimal", "original"),
  verbose = FALSE
)

## S4 method for signature 'DeeDeeExperiment'
linkDEAandFEA(x, dea_name, fea_name, force = FALSE)
```

Arguments

x	A `DeeDeeExperiment()` object

`value`	Replacement value for replacement methods.
`old_name`	A character vector of existing DEA names to be renamed in a `DeeDeeExperiment` object
`new_name`	A character vector with new names to assign to existing DEA names in a `DeeDeeExperiment` object. It must be the same length of `old_name`, and contains unique values that don't overlap with existing DEA names.
`dea`	A named list of DE results, in any of the formats supported by the package (currently: results from DESeq2, edgeR, limma).
`force`	A logical, indicating whether to overwrite results when introducing the same results name. It defaults to FALSE.
`dea_name`	Character value, specifying the name of the DE analysis to get or remove, or match against (e.g., to fetch associated FEA results), or to which additional context and information can be attached
`remove_linked_fea`	A logical, specifying whether to remove or not the linked FEA when a DEA results is removed
`format`	A character string, specifying the DEA/FEAs output format. It takes either "minimal" to return only essential columns (e.g. log2FC, p-value, adjusted p-value for DEAs, or gs_id, gs_description, gs_pvalue, gs_genes... for FEAs), or "original" to return the full result object. It defaults to "minimal"
`extra_rd`	A character vector of additional columns from rowData(x) to include. It defaults to c("gene_id", "SYMBOL").
`type`	A character string referring to the type of object returned by `getDEA()`. It defaults to `DFrame`, but can also take the value of `data.frame`
`verbose`	Logical, whether or not to display warnings. If TRUE, warnings/messages will be displayed. If FALSE, the function runs silently
`info`	A character vector, containing contextual information about the specified DE analysis. It defaults to NULL
`fea`	A named list of Functional Enrichment results. Each element can be either a data.frame (currently supports results from `topGO`, `enrichR`, `gProfiler`, `fgsea`, `gsea`, `DAVID`, and output of `GeneTonic` shakers), or an `enrichResult`/`gseaResult` objects (currently supports `clusterProfiler`)
`de_name`	A character string to explicitly specify the name of the de result this fea should be linked to. If not provided, the function will attempt to match fea names to de results automatically.
`fe_name`	A character string giving a name to the FE results.
`fea_tool`	A character string indicating the FEA tool used. It can take any of the following values : "topGO", "clusterProfiler", "GeneTonic", "DAVID", "gsea", "fgsea", "enrichr", "gProfiler". When not specified, it defaults to "auto" and the tool is inferred automatically based on the input.
`fea_name`	Character value, specifying the name of the functional enrichment result to add or remove

Details

DEAs

- `getDEAInfo` and `getDEAInfo<-` are the methods to get and set the `dea` information as a whole. These methods return `DeeDeeExperiment` objects.

- `getDEANames` returns the names of the available DE contrasts in `DeeDeeExperiment` objects.
- `renameDEA` is the method to rename one or multiple DEAs stored in a `DeeDeeExperiment` object.
- `addDEA` and `removeDEA` are used to respectively add or remove DE-results items. These methods also return `DeeDeeExperiment` objects, with updated content in the `dea` slot.
- `getDEA` and `getDEAList` retrieve the DEA information, as well as some extra rowData information and provide this as a `DataFrame` object (for a specific analysis) or as a list, with one element for each reported analysis.
- `addScenarioInfo` is the method to add user defined contextual information for a specific DE analysis. It allows users to attach free-text notes to a specific DEA results that stored in a `DeeDeeExperiment` object. This information can include any other relevant information to help document that DEA scenario. This context is stored in the `dea` slot under the name `addScenarioInfo`, which is not a default element in `dea`.

FEAs

- `getFEAInfo` and `getFEAInfo<-` are the methods to get and set the `fea` information as a whole. These methods return `DeeDeeExperiment` objects.
- `getFEANames` returns the names of the available enrichment results in `DeeDeeExperiment` objects.
- `renameFEA` is the method to rename one or multiple FEAs stored in a `DeeDeeExperiment` object.
- `addFEA` and `removeFEA` are used to respectively add or remove functional enrichment results items. These methods also return `DeeDeeExperiment` objects, with updated content in the `fea` slot.
- `getFEA` is the method to retrieve FE results stored in a `DeeDeeExperiment` object for a specific contrast, as a standardized format similar to the output of `GeneTonic` shakers.
- `getFEAList` is the method that retrieves FEA results as a list. if the `dea_name` is indicated, the method will return only FEAs linked to that `dea_name`, otherwise it returns all FEAs in the `fea` slot.
- `linkDEAandFEA` is the method that allows the user to manually link a FEA result to a specific DEA result.
- `show` is the method to nicely print out the information of a `DeeDeeExperiment` object.
- `summary` is the method to print a summary of the available DE and FE results in a `DeeDeeExperiment` object.

Value

Return value varies depending on the individual methods, as described below.

Examples

```
data("de_named_list", package = "DeeDeeExperiment")
data("topGO_results_list", package = "DeeDeeExperiment")
library("SummarizedExperiment")

rd_macrophage <- DataFrame(
  gene_id = rownames(de_named_list$ifng_vs_naive)
)
rownames(rd_macrophage) <- rownames(de_named_list$ifng_vs_naive)
se_macrophage_noassays <- SummarizedExperiment(
```

```
  assays = SimpleList(),
  rowData = rd_macrophage
)

# creating a `DeeDeeExperiment`
dde <- DeeDeeExperiment(
  se_macrophage_noassays,
  de_results = de_named_list
)
dde

new_del <- list(
  ifng2 = de_named_list$ifng_vs_naive,
  ifngsalmo2 = de_named_list$ifngsalmo_vs_naive
)

# add a new (set of) DE result(s)
dde_new <- addDEA(dde, new_del)
dde_new

# removing DEAs
dde_removed <- removeDEA(dde, "ifng_vs_naive")
dde_removed

# add a new (set of) FE result(s)
dde_new <- addFEA(dde, fea = topGO_results_list)

# removing FEAs
dde_rem <- removeFEA(dde_new, "ifng_vs_naive")

# display available DEAs
getDEANames(dde)

# display available FEAs
getFEANames(dde)

# print a summary of the available DEAs and FEAs
summary(dde, FDR = 0.01)

# rename DEA
dde_new <- renameDEA(dde_new,
  old_name = "salmonella_vs_naive",
  new_name = "Salmo_vs_Naive_renamed"
)

# assign DEA to FEA

dde_new <- linkDEAandFEA(dde_new,
  dea_name = "ifngsalmo_vs_naive",
  fea_name = "ifngsalmo_vs_naive"
)
```

`DeeDeeExperiment-misc` *Miscellaneous DeeDeeExperiment methods*

Description

Miscellaneous methods for the `DeeDeeExperiment` class and its descendants that do not fit into any other documentation category such as, for example, show and summary methods.

Usage

```
## S4 method for signature 'DeeDeeExperiment'
show(object)

## S4 method for signature 'DeeDeeExperiment'
summary(object, FDR = 0.05, show_scenario_info = FALSE, ...)
```

Arguments

`object`	a `DeeDeeExperiment` object
`FDR`	Numeric, sets the significance threshold for subsetting differentially expressed genes based on adjusted p-values. Defaults to 0.05
`show_scenario_info`	Logical; if TRUE, displays the associated scenario info for each DE contrast, if available. Defaults to FALSE
`...`	additional argument passed to the summary method. Currently supports: • `FDR`: Numeric, sets the significance threshold for subsetting differentially expressed genes based on adjusted p-values. Defaults to 0.05 • `show_scenario_info`: Logical; if `TRUE`, displays the associated scenario info for each DE contrast, if available.

Value

Returns NULL

`DeeDeeExperiment-pkg` *DeeDeeExperiment*

Description

`DeeDeeExperiment` is a an S4 class that allows integrating and managing omics analysis results.

Details

`DeeDeeExpeirment` is an S4 class extending the `SummarizedExperiment` framework to facilitate the integration and management of omics analysis results. It introduces two dedicated slots to store Differential Expression (DE) analysis results and Functional Enrichment analysis results, providing a structured approach for downstream analysis.

Author(s)

Maintainer: Najla Abassi `<abassi.nejla96@gmail.com>` (ORCID)

Authors:

- Lea Schwarz `<lea.schwarz@uni-mainz.de>` (ORCID)
- Federico Marini `<marinif@uni-mainz.de>` (ORCID)

See Also

Useful links:

- `https://github.com/imbeimainz/DeeDeeExperiment`
- Report bugs at `https://github.com/imbeimainz/DeeDeeExperiment/issues`

`de_limma` *A sample* `MArrayLM` *object*

Description

A sample `MArrayLM` object, generated with `limma`

Format

A `MArrayLM` object

Details

This `MArrayLM` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from macrophage associated with different stimulation conditions, including IFNg treatment, Salmonella infection, and their combined effects.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `MArrayLM` object, generated with `limma`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

`de_named_list` *de_named_list*

Description

A named list of DE results, in their original format (from DESeq2, edgeR or limma)

Format

A list object

Details

documented creation in the `create_dataset_example.R` script in the `scripts` package folder

Value

A named list of DE results, in their original format

`dge_exact_IFNg_both` *A sample* `DGEExact` *object*

Description

A sample `DGEExact` object, generated with `edgeR`

Format

A `DGEExact` object

Details

This `DGEExact` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from macrophage associated with IFNg to those associated with IFNg and Salmonella.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DGEExact` object, generated with `edgeR`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other edgeRresus: `dge_exact_IFNg_naive`, `dge_exact_Salm_both`, `dge_exact_Salm_naive`, `dge_lrt_IFNg_both`, `dge_lrt_IFNg_naive`, `dge_lrt_Salm_both`, `dge_lrt_Salm_naive`

`dge_exact_IFNg_naive` *A sample* `DGEExact` *object*

Description

A sample `DGEExact` object, generated with `edgeR`

Format

A `DGEExact` object

Details

This `DGEExact` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from naive macrophage to those associated with IFNg.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DGEExact` object, generated with `edgeR`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other edgeRresus: `dge_exact_IFNg_both`, `dge_exact_Salm_both`, `dge_exact_Salm_naive`, `dge_lrt_IFNg_both`, `dge_lrt_IFNg_naive`, `dge_lrt_Salm_both`, `dge_lrt_Salm_naive`

`dge_exact_Salm_both` *A sample* `DGEExact` *object*

Description

A sample `DGEExact` object, generated with `edgeR`

Format

A `DGEExact` object

Details

This `DGEExact` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from macrophage associated with Salmonella to those associated with IFNg and Salmonella.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DGEExact` object, generated with `edgeR`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other edgeRresus: `dge_exact_IFNg_both`, `dge_exact_IFNg_naive`, `dge_exact_Salm_naive`, `dge_lrt_IFNg_both`, `dge_lrt_IFNg_naive`, `dge_lrt_Salm_both`, `dge_lrt_Salm_naive`

`dge_exact_Salm_naive` *A sample* `DGEExact` *object*

Description

A sample `DGEExact` object, generated with `edgeR`

Format

A `DGEExact` object

Details

This `DGEExact` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from naive macrophage to those associated with Salmonella.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DGEExact` object, generated with `edgeR`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other edgeRresus: `dge_exact_IFNg_both`, `dge_exact_IFNg_naive`, `dge_exact_Salm_both`, `dge_lrt_IFNg_both`, `dge_lrt_IFNg_naive`, `dge_lrt_Salm_both`, `dge_lrt_Salm_naive`

`dge_lrt_IFNg_both` *A sample* `DGELRT` *object*

Description

A sample `DGELRT` object, generated with `edgeR`

Format

A `DGELRT` object

Details

This `DGELRT` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from macrophage associated with IFNg to those associated with IFNg and Salmonella.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DGELRT` object, generated with `edgeR`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other edgeRresus: `dge_exact_IFNg_both`, `dge_exact_IFNg_naive`, `dge_exact_Salm_both`, `dge_exact_Salm_naive`
`dge_lrt_IFNg_naive`, `dge_lrt_Salm_both`, `dge_lrt_Salm_naive`

`dge_lrt_IFNg_naive` *A sample* `DGELRT` *object*

Description

A sample `DGELRT` object, generated with `edgeR`

Format

A `DGELRT` object

Details

This `DGELRT` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from macrophage associated with IFNg to those associated with IFNg and naive.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DGELRT` object, generated with `edgeR`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other edgeRresus: `dge_exact_IFNg_both`, `dge_exact_IFNg_naive`, `dge_exact_Salm_both`, `dge_exact_Salm_naive`
`dge_lrt_IFNg_both`, `dge_lrt_Salm_both`, `dge_lrt_Salm_naive`

`dge_lrt_Salm_both` *A sample* `DGELRT` *object*

Description

A sample `DGELRT` object, generated with `edgeR`

Format

A `DGELRT` object

Details

This `DGELRT` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from macrophage associated with Salmonella to those associated with IFNg and Salmonella.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DGELRT` object, generated with `edgeR`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other edgeRresus: `dge_exact_IFNg_both`, `dge_exact_IFNg_naive`, `dge_exact_Salm_both`, `dge_exact_Salm_naive`
`dge_lrt_IFNg_both`, `dge_lrt_IFNg_naive`, `dge_lrt_Salm_naive`

`dge_lrt_Salm_naive` *A sample* `DGELRT` *object*

Description

A sample `DGELRT` object, generated with `edgeR`

Format

A `DGELRT` object

Details

This `DGELRT` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from naive macrophage to those associated with Salmonella.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DGELRT` object, generated with `edgeR`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other edgeRresus: `dge_exact_IFNg_both`, `dge_exact_IFNg_naive`, `dge_exact_Salm_both`, `dge_exact_Salm_naive`
`dge_lrt_IFNg_both`, `dge_lrt_IFNg_naive`, `dge_lrt_Salm_both`

`enrichr_res` `enrichr_res`

Description

A list of FE results generated with enrichr::enrichr()

Format

A list of data.frame objects

Details

A list of FE result for the macrophage data (salmonella vs naive)

Value

A list

See Also

Other enrich_resus: `clusterPro_res`, `fgseaRes`, `gost_res`, `gsea_res`, `topGO_results_list`

`export_result_for_dde` *Export DEA/FEA/ASSAY results from a* `DeeDeeExperiment` *to excel files*

Description

Extracts DEA and/or FEA results stored in a `DeeDeeExperiment` and writes them to excel files. Each contrast/result is written to a separate sheet. DEA and FEA are written to separate files.

Usage

```
export_result_for_dde(
  x,
  res_type = c("assay", "dea", "fea", "all"),
  res_format = c("minimal", "original"),
  output_dir = getwd(),
  file_name = NULL,
  assay_type = c("counts"),
  force = FALSE
)
```

Arguments

`x`	A `DeeDeeExperiment` object containing DEA/FEA results to be extracted
`res_type`	A character vector indicating the result to extract (i.e. `"dea"`, `"fea"`, `"assay"`) from the corresponding slot If set to `"all"`, DEAs, FEAs, as well as the specified assays are extracted into separate excel files
`res_format`	A character string specifying the DEA/FEAs output format to be exported (i.e. `"minimal"` or `"original"`)
`output_dir`	A character string specifying the directory where the excel file will be written
`file_name`	Optional character string specifying a file name. If `NULL`, a default name is generated
`assay_type`	Optional character vector specifying the assays to export. It defaults to `"counts"`
`force`	Logical. If `TRUE`, an existing file with the same name will be overwritten

Value

An (invisible) character vector of file paths to the exported excel files. The vector may contain one path `res_type = "dea"` or `res_type = "fea"` or three paths `res_type = "all"`

Examples

```
data("de_named_list", package = "DeeDeeExperiment")
data("topGO_results_list", package = "DeeDeeExperiment")
dde <- DeeDeeExperiment(de_results = de_named_list, enrich_results = topGO_results_list)

export_result_for_dde(dde,
res_type = "dea",
res_format = "minimal", output_dir = tempdir(), force = TRUE)

export_result_for_dde(dde,
res_type = c("dea", "fea"),
res_format = "original", output_dir = tempdir(), force = TRUE)
```

`fgseaRes` `fgseaRes`

Description

A data frame of FE results generated with `fgsea::fgsea()`

Format

A data.table/data.frame object

Details

Tabular representation of FE result for the macrophage data (IFNg vs naive)

Value

A data.table/data.frame

See Also

Other enrich_resus: `clusterPro_res`, `enrichr_res`, `gost_res`, `gsea_res`, `topGO_results_list`

`gost_res` `gost_res`

Description

A list of FE results generated with `gprofiler2::gost()`

Format

A list, as returned by `gprofiler2`

Details

A list of FE result for the macrophage data (salmonella vs naive)

Value

A list

See Also

Other enrich_resus: `clusterPro_res`, `enrichr_res`, `fgseaRes`, `gsea_res`, `topGO_results_list`

`gsea_res` `gsea_res`

Description

An individual set of FE results generated with `clusterProfiler::gseGO()`

Format

A `gseaResult` object

Details

A set of FE results for the macrophage data (IFNg vs naive)

Value

A `gseaResult` object

See Also

Other enrich_resus: `clusterPro_res`, `enrichr_res`, `fgseaRes`, `gost_res`, `topGO_results_list`

`IFNg_both` *A sample* `DESeqResults` *object*

Description

A sample `DESeqResults` object, generated with `DESeq2`

Format

A `DESeqResults` object

Details

This `DESeqResults` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from macrophage associated with IFNg to those associated with IFNg and Salmonella.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DESeqResults` object, generated with `DESeq2`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other DEresus: `IFNg_naive`, `Salm_both`, `Salm_naive`

`IFNg_naive` *A sample* `DESeqResults` *object*

Description

A sample `DESeqResults` object, generated with `DESeq2`

Format

A `DESeqResults` object

Details

This `DESeqResults` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from naive macrophage to those associated with IFNg.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DESeqResults` object, generated with `DESeq2`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other DEresus: `IFNg_both`, `Salm_both`, `Salm_naive`

`limma_list_for_dde` *Convert a* `MArrayLM` *object with multiple contrasts into a list of DE results tables compatible with DeeDeeExperiment*

Description

This helper function extracts DE results for each contrast contained in a `limma::MArrayLM` object and reformats them into a list of standardized data frames suitable for integration in a `DeeDeeExperiment` object. Each resulting data frame includes renamed columns: `logFC` to `log2FoldChange` ,`P.Value` to `pvalue`, and `adj.P.Val` to `padj`

Usage

```
limma_list_for_dde(fit, number = nrow(fit), sort.by = "none")
```

Arguments

`fit`	A `MArrayLM` object, as produced by the `limma` workflow
`number`	An integer specifying the maximum number of genes to extract per contrast
`sort.by`	A character string specifying which statistic to rank the genes by. It must be one of the values accepted by the `sort.by` argument in the `limma::topTable()` function. Defaults to "none".

Details

The function assumes that each column in `fit$coefficients` corresponds to a contrast of interest. The names of the resulting list elements are taken directly from `colnames(fit$coefficients)`. The names in the input object must therefore accurately reflect the intended contrast names.

Value

A named list of DE results tables, one per contrast. Each table contains standardized columns and the list is annotated with metadata indicating its package origin (`limma`).

See Also

`limma::topTable()`

Examples

```
data("de_limma", package = "DeeDeeExperiment")
new_limma_list <- limma_list_for_dde(de_limma)
```

`muscat_list_for_dde` *Convert* `muscat::pbDS()` *results into a flat list of data frames compatible with* `DeeDeeExperiment`

Description

This helper function extracts and flattens the nested structure returned by `muscat::pbDS()`, returning one table per contrast–cluster combination. Each resulting data frame will have standardized column names (`log2FoldChange`, `pvalue`, `padj`).

Usage

```
muscat_list_for_dde(res, padj_col = c("p_adj.loc", "p_adj.glb"))
```

Arguments

`res`	A list, typically the output of `muscat::pbDS()` function, containing one or more contrasts
`padj_col`	A character string specifying which adjusted p-value column to extract. It can be either "p_adj.loc" or "p_adj.glb".

Details

The function is intended to simplify the integration of muscat results into a `DeeDeeExperiment` object. It automatically renames columns (`logFC` to `log2FoldChange`, `p_val` to `pvalue`, and the selected adjusted p-value column to `padj`) and annotates the resulting list with metadata about the originating package.

The function checks that each contrast entry contains a valid `table` component as expected from `pbDS()` output. Invalid or empty contrasts are skipped with a warning message. The names of the list elements in `res` must match contrast names found in the `table` slot of each entry.

Value

A named list of data frames

Examples

```
data("muscat_pbDS_res", package = "DeeDeeExperiment")
new_muscat_list <- muscat_list_for_dde(list(`stim-ctrl` = muscat_res))
```

`muscat_res` muscat_res

Description

A small example object generated with `muscat::pbDS()` ona subset Kang et al. dataset. The results represent DE between stimulated vs control samples.

Format

A `list` structured as returned by `muscat::pbDS()`

Details

The original data were obtained from Kang et al. dataset in `ExperimentHub` and processed using `muscat` workflows. Only a small subset of genes and clusters was retained to reduce object size.

Value

A named `list` of DE results tables compatible with `muscat_list_for_dde`

`Salm_both` *A sample* `DESeqResults` *object*

Description

A sample `DESeqResults` object, generated with `DESeq2`

Format

A `DESeqResults` object

Details

This `DESeqResults` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from macrophage associated with Salmonella to those associated with IFNg and Salmonella.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DESeqResults` object, generated with `DESeq2`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other DEresus: `IFNg_both`, `IFNg_naive`, `Salm_naive`

`Salm_naive` *A sample* `DESeqResults` *object*

Description

A sample `DESeqResults` object, generated with `DESeq2`

Format

A `DESeqResults` object

Details

This `DESeqResults` object contains the results of a Differential Expression Analysis performed on data from the `macrophage` package, more precisely contrasting the counts from naive macrophage to those associated with Salmonella.

The code to create said object can be found in the folder `/inst/scripts` in the DeeDeeExperiment package, the file is called `generate_data.R`.

Value

A sample `DESeqResults` object, generated with `DESeq2`

References

Alasoo, et al. "Shared genetic effects on chromatin and gene expression indicate a role for enhancer priming in immune response", Nature Genetics, January 2018 doi: 10.1038/s41588-018-0046-7.

See Also

Other DEresus: `IFNg_both`, `IFNg_naive`, `Salm_both`

`supported_fea_formats` *Display available FEA formats*

Description

Display available FEA formats

Usage

```
supported_fea_formats()
```

Value

a data.frame of possible FEA input formats

Examples

```
supported_fea_formats()
```

`topGO_results_list` `topGO_results_list`

Description

A list of FE results generated with `mosdef::topGOtable()`

Format

A list of data.frame objects

Details

A list of FE results for the macrophage data

Value

A list

See Also

Other enrich_resus: `clusterPro_res`, `enrichr_res`, `fgseaRes`, `gost_res`, `gsea_res`

Index