

# Package ‘iSEEd’

December 5, 2025

**Title** iSEE extension for panels related to differential expression analysis

**Version** 1.9.0

**Date** 2024-10-01

**Description** This package contains diverse functionality to extend the usage of the iSEE package, including additional classes for the panels or modes facilitating the analysis of differential expression results.  
This package does not perform differential expression. Instead, it provides methods to embed precomputed differential expression results in a SummarizedExperiment object, in a manner that is compatible with interactive visualisation in iSEE applications.

**License** Artistic-2.0

**URL** <https://github.com/iSEE/iSEEd>

**BugReports** <https://support.bioconductor.org/t/iSEEd>

**biocViews** Software, Infrastructure, DifferentialExpression

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.3

**Depends** iSEE

**Imports** DESeq2, edgeR, methods, S4Vectors, shiny, SummarizedExperiment

**Suggests** airway, BiocStyle, covr, knitr, limma, org.Hs.eg.db,  
RefManageR, rmarkdown, scuttle, sessioninfo, statmod, testthat  
(>= 3.0.0)

**Config/testthat/edition** 3

**VignetteBuilder** knitr

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

**git\_branch** devel

**git\_last\_commit** b1d08fe

**git\_last\_commit\_date** 2025-10-29

**Repository** Bioconductor 3.23  
**Date/Publication** 2025-12-04  
**Author** Kevin Rue-Albrecht [aut, cre] (ORCID:  
    <<https://orcid.org/0000-0003-3899-3872>>),  
    Thomas Sandmann [ctb] (ORCID: <<https://orcid.org/0000-0002-6601-8890>>),  
    Denali Therapeutics [fnd]  
**Maintainer** Kevin Rue-Albrecht <kevinrue67@gmail.com>

Contents

|                                      |           |
|--------------------------------------|-----------|
| contrastResultsNames . . . . .       | 2         |
| de-generics . . . . .                | 3         |
| DETable-class . . . . .              | 4         |
| iSEEd-e-pkg . . . . .                | 5         |
| iSEEDSeq2Results-class . . . . .     | 5         |
| iSEEdedgeRResults-class . . . . .    | 7         |
| iSEELimmaResults-class . . . . .     | 8         |
| LogFCLogFCPlot-class . . . . .       | 10        |
| MAPlot-class . . . . .               | 11        |
| utils-SummarizedExperiment . . . . . | 11        |
| VolcanoPlot-class . . . . .          | 12        |
| <b>Index</b>                         | <b>14</b> |

---

|                      |                                                                           |
|----------------------|---------------------------------------------------------------------------|
| contrastResultsNames | <i>Extract contrast results embedded in a SummarizedExperiment object</i> |
|----------------------|---------------------------------------------------------------------------|

---

Description

contrastResults returns either all contrasts results stored in object or a single contrast result by name.  
contrastResultsNames returns the names of contrast results embedded in object.

Usage

contrastResultsNames(object)  
  
contrastResults(object, name)

Arguments

object           A [SummarizedExperiment](#) object.  
name            (Optional) Name of a single contrast result name to extract. Use contrastResultsNames(object) to list available names.

**Value**

For `contrastResultsNames`: the names of embedded contrast results available.

For `contrastResults`: a `DataFrame` of differential expression statistics.

If `name` is missing, `contrastResults` returns a nested `DataFrame` in which each column contains the results of a single contrast. If `name` is given, `contrastResults` returns a `DataFrame` that contains the results of a single contrast.

**Examples**

```
library("iSEEde")
library("airway")
library("DESeq2")
library("iSEE")

##
# Example data ----
##

data("airway")
airway$dex <- relevel(airway$dex, "untrt")

dds <- DESeqDataSet(airway, ~ 0 + dex + cell)

dds <- DESeq(dds)
res_deseq2 <- results(dds, contrast = list("dextrt", "dexuntrt"))
airway <- embedContrastResults(res_deseq2, airway, name = "dex: trt vs untrt")

##
# List result names ---
##

contrastResultsNames(airway)

##
# Extract results ---
##

contrastResults(airway)
contrastResults(airway, "dex: trt vs untrt")
```

**Description**

An overview of the generics for accessing common pieces of information in differential expression results.

## Definitions

- `pValue(x)` returns a named numeric vector of raw p-values.
- `log2FoldChange(x)` returns a named numeric vector of log2-fold-change values.
- `averageLog2(x)` returns a named numeric vector of average log2-expression values.

## Author(s)

Kevin Rue-Albrecht

## Examples

```
showMethods(pValue)
showMethods(log2FoldChange)
showMethods(averageLog2)
```

---

DETable-class

*The DETable class*


---

## Description

The DETable class is a [RowTable](#) subclass that is dedicated to creating a volcano plot. It retrieves the table of results for the selected differential expression contrast and creates an interactive table where each row represents a feature.

## Slot overview

The following slots control the test procedure:

- `ContrastName`, a character scalar indicating the name of the contrast to display.
- `RoundDigits`, a logical scalar indicating whether to round numeric values (see `SignifDigits`).
- `SignifDigits`, an integer scalar indicating the number of significant digits to use for rounding numbers (see `RoundDigits`).

In addition, this class inherits all slots from its parent [RowTable](#) and [Table](#) classes.

## Examples

```
x <- DETable()
x
```

iSEEd-e-pkg

*iSEE extension for panels related to differential expression analysis***Description**

iSEEd-e is a package that provides panels for **iSEE**, facilitating the interactive visualisation of differential expression results.

**Author(s)**

Kevin Rue-Albrecht <kevin.rue-albrecht@imm.ox.ac.uk>

**See Also**

Useful links:

- <https://github.com/iSEE/iSEEd-e>
- Report bugs at <https://support.bioconductor.org/t/iSEEd-e>

**Examples**

```
library("iSEEd-e")
```

iSEEd-eSeq2Results-class

*The iSEEd-eSeq2Results class***Description**

The iSEEd-eSeq2Results class is used to provide a common interface to differential expression results produced by the **DESeq2** package. It provides methods to access common differential expression statistics (e.g., log2 fold-change, p-value, log2 average abundance).

**Details**

This class inherits all its slots directly from its parent class [DataFrame](#).

**Constructor**

iSEEd-eSeq2Results(data, row.names = rownames(data)) creates an instance of a iSEEd-eSeq2Results class, with:

data A data.frame produced by DESeq2::results() or DESeq2::lfcShrink().

row.names The character vector of rownames for the [SummarizedExperiment](#) object in which the object is to be embedded. Must be a superset of rownames(data).

**Supported methods**

- `embedContrastResults(x, se, name, ...)` embeds `x` in `se` under the identifier `name`. See [embedContrastResults\(\)](#) for more details.
- `pValue(x)` returns the vector of raw p-values.
- `log2FoldChange(x)` returns the vector of log2-fold-change values.
- `averageLog2(x)` returns the vector of average log2-expression values.

**Author(s)**

Kevin Rue-Albrecht

**Examples**

```
library(DESeq2)

##
# From DESeq2::DESeq() ----
##

cnts <- matrix(rnbinom(n = 1000, mu = 100, size = 1 / 0.5), ncol = 10)
rownames(cnts) <- paste("Gene", 1:100)
cond <- factor(rep(1:2, each = 5))

# object construction
dds <- DESeqDataSetFromMatrix(cnts, DataFrame(cond), ~cond)

# standard analysis
dds <- DESeq(dds)
res <- results(dds)
head(res)

##
# iSEEDESeq2Results ----
##

# Embed the DESeq2 results in the SummarizedExperiment object
dds <- embedContrastResults(res, dds, name = "DESeq2")

##
# Access ----
##

contrastResultsNames(dds)
contrastResults(dds)
contrastResults(dds, "DESeq2")

head(pValue(contrastResults(dds, "DESeq2")))
head(log2FoldChange(contrastResults(dds, "DESeq2")))
head(averageLog2(contrastResults(dds, "DESeq2")))
```

---

`iSEEedgeRResults-class`*The iSEEedgeRResults class*

---

## Description

The `iSEEedgeRResults` class is used to provide a common interface to differential expression results produced by the **edgeR** package. It provides methods to access common differential expression statistics (e.g., log fold-change, p-value, log2 average abundance).

## Details

This class inherits all its slots directly from its parent class [DataFrame](#).

## Constructor

`iSEEedgeRResults(data, row.names = rownames(data))` creates an instance of a `iSEEedgeRResults` class, with:

`data` A `data.frame` produced by `edgeR::topTags()`.

`row.names` The character vector of rownames for the [SummarizedExperiment](#) object in which the object is to be embedded. Must be a superset of `rownames(data)`.

## Supported methods

- `embedContrastResults(x, se, name, ...)` embeds `x` in `se` under the identifier `name`. See [embedContrastResults\(\)](#) for more details.
- `pValue(x)` returns the vector of raw p-values.
- `log2FoldChange(x)` returns the vector of log2-fold-change values.
- `averageLog2(x)` returns the vector of average log2-expression values.

## Author(s)

Kevin Rue-Albrecht

## Examples

```
library(edgeR)
library(SummarizedExperiment)

##
# From edgeR::glmLRT() ----
##

nlibs <- 3
ngenes <- 100
dispersion.true <- 0.1
```

```

# Make first gene respond to covariate x
x <- 0:2
design <- model.matrix(~x)
beta.true <- cbind(Beta1=2,Beta2=c(2,rep(0,ngenes-1)))
mu.true <- 2^(beta.true %*% t(design))

# Generate count data
y <- rnbino(m(ngenes*nlibs,mu=mu.true,size=1/dispersion.true))
y <- matrix(y,ngenes,nlibs)
colnames(y) <- c("x0","x1","x2")
rownames(y) <- paste("gene",1:ngenes,sep=".")
d <- DGEList(y)

# Normalize
d <- calcNormFactors(d)

# Fit the NB GLMs
fit <- glmFit(d, design, dispersion=dispersion.true)

# Likelihood ratio tests for trend
results <- glmLRT(fit, coef=2)
tt <- topTags(results)

##
# iSEEdgeRResults ----
##

# Simulate the original SummarizedExperiment object
se <- SummarizedExperiment(assays = list(counts = d$counts))

# Embed the edgeR results in the SummarizedExperiment object
se <- embedContrastResults(tt, se, name = "edgeR")

##
# Access ----
##

contrastResultsNames(se)
contrastResults(se)
contrastResults(se, "edgeR")

head(pValue(contrastResults(se, "edgeR")))
head(log2FoldChange(contrastResults(se, "edgeR")))
head(averageLog2(contrastResults(se, "edgeR")))

```

---

iSEELimmaResults-class

*The iSEELimmaResults class*


---

## Description

The `iSEELimmaResults` class is used to provide a common interface to differential expression results produced by the **limma** package. It provides methods to access common differential expression statistics (e.g., log fold-change, p-value, log2 average abundance).

## Details

This class inherits all its slots directly from its parent class [DataFrame](#).

## Constructor

`iSEELimmaResults(data, row.names = rownames(data))` creates an instance of a `iSEELimmaResults` class, with:

`data` A data.frame produced by `limma::topTable()`.

`row.names` The character vector of rownames for the [SummarizedExperiment](#) object in which the object is to be embedded. Must be a superset of `rownames(data)`.

## Supported methods

- `embedContrastResults(x, se, name, class = "limma", ...)` embeds `x` in `se` under the identifier name. See [embedContrastResults\(\)](#) for more details.
- `pValue(x)` returns the vector of raw p-values.
- `log2FoldChange(x)` returns the vector of log2-fold-change values.
- `averageLog2(x)` returns the vector of average log2-expression values.

## Author(s)

Kevin Rue-Albrecht

## Examples

```
library(limma)
library(SummarizedExperiment)

##
# From limma::lmFit() ----
##

sd <- 0.3 * sqrt(4 / rchisq(100, df = 4))
y <- matrix(rnorm(100 * 6, sd = sd), 100, 6)
rownames(y) <- paste("Gene", 1:100)
y[1:2, 4:6] <- y[1:2, 4:6] + 2
design <- cbind(Grp1 = 1, Grp2vs1 = c(0, 0, 0, 1, 1, 1))

fit <- lmFit(y, design)
fit <- eBayes(fit)
tt <- topTable(fit, coef = 2)
head(tt)
```

```
##
# iSEELimmaResults ----
##

# Simulate the original SummarizedExperiment object
se <- SummarizedExperiment(assays = list(counts = y))

# Embed the Limma-Voom results in the SummarizedExperiment object
se <- embedContrastResults(tt, se, name = "Limma-Voom", class = "limma")

##
# Access ----
##

contrastResultsNames(se)
contrastResults(se)
contrastResults(se, "Limma-Voom")

head(pValue(contrastResults(se, "Limma-Voom")))
head(log2FoldChange(contrastResults(se, "Limma-Voom")))
head(averageLog2(contrastResults(se, "Limma-Voom")))
```

---

LogFCLogFCPlot-class    *The LogFCLogFCPlot class*

---

## Description

The LogFCLogFCPlot class is a [RowDataPlot](#) subclass that is dedicated to comparing the log-fold-change value of two contrasts. It retrieves the log-fold change of the two selected contrasts and creates a row-based plot where each point represents a feature.

## Slot overview

The following slots control the test procedure:

- ContrastNameX, a character scalar indicating the name of the contrast to display on the x-axis.
- ContrastNameY, a character scalar indicating the name of the contrast to display on the y-axis.

In addition, this class inherits all slots from its parent [RowDotPlot](#), [DotPlot](#), and [Panel](#) classes.

## Examples

```
x <- LogFCLogFCPlot()
x
```

MAPlot-class

*The MAPlot class***Description**

The MAPlot is a [RowDataPlot](#) subclass that is dedicated to creating an MA plot. It retrieves the log-fold change (M) and mean average (A) values and creates a row-based plot where each point represents a feature.

**Slot overview**

The following slots control the test procedure:

- ContrastName, a character scalar indicating the name of the contrast to display.

In addition, this class inherits all slots from its parent [RowDotPlot](#), [DotPlot](#), and [Panel](#) classes.

**Examples**

```
x <- MAPlot()
x
```

utils-SummarizedExperiment

*Generics for Embedding Results into a SummarizedExperiment Object*

**Description**

An overview of the generics for embedding results into a [SummarizedExperiment](#) object, in a format compatible with [iSEEde](#).

**Usage**

```
embedContrastResults(x, se, name, ...)
```

```
embedContrastResultsMethods
```

```
## S4 method for signature 'ANY'
embedContrastResults(x, se, name, ...)
```

```
## S4 method for signature 'data.frame'
embedContrastResults(x, se, name, class, ...)
```

**Arguments**

|       |                                                                                                                         |
|-------|-------------------------------------------------------------------------------------------------------------------------|
| x     | Object to be embedded.                                                                                                  |
| se    | A <a href="#">SummarizedExperiment</a> object.                                                                          |
| name  | Identifier for the embedded object.                                                                                     |
| ...   | Arguments passed to and from other methods.                                                                             |
| class | Class to use for embedding x. Only used when class(x) does not uniquely identify the package that generated the object. |

**Format**

embedContrastResultsMethods: Named character vector mapping keywords to class names designed to store differential expression results.

**Value**

An updated [SummarizedExperiment](#) object that contains the embedded object.

**Definitions**

- `embedContrastResults(x, se, name, ...)` embeds the results x in the [SummarizedExperiment](#) se.

**Author(s)**

Kevin Rue-Albrecht

**Examples**

```
embedContrastResultsMethods
showMethods(embedContrastResults)
```

---

VolcanoPlot-class      *The VolcanoPlot class*

---

**Description**

The VolcanoPlot is a [RowDataPlot](#) subclass that is dedicated to creating a volcano plot. It retrieves the log-fold change and p-value from and creates a row-based plot where each point represents a feature.

**Slot overview**

The following slots control the test procedure:

- ContrastName, a character scalar indicating the name of the contrast to display.

In addition, this class inherits all slots from its parent [RowDotPlot](#), [DotPlot](#), and [Panel](#) classes.

**Examples**

```
x <- VolcanoPlot()  
x
```

# Index

- \* **datasets**
  - utils-SummarizedExperiment, [11](#)
- \* **internal**
  - iSEEde-pkg, [5](#)
  - .cacheCommonInfo, DETable-method (DETable-class), [4](#)
  - .cacheCommonInfo, LogFCLogFCPlot-method (LogFCLogFCPlot-class), [10](#)
  - .cacheCommonInfo, MAPlot-method (MAPlot-class), [11](#)
  - .cacheCommonInfo, VolcanoPlot-method (VolcanoPlot-class), [12](#)
  - .createObservers, DETable-method (DETable-class), [4](#)
  - .createObservers, LogFCLogFCPlot-method (LogFCLogFCPlot-class), [10](#)
  - .createObservers, MAPlot-method (MAPlot-class), [11](#)
  - .createObservers, VolcanoPlot-method (VolcanoPlot-class), [12](#)
  - .defineDataInterface, DETable-method (DETable-class), [4](#)
  - .defineDataInterface, LogFCLogFCPlot-method (LogFCLogFCPlot-class), [10](#)
  - .defineDataInterface, MAPlot-method (MAPlot-class), [11](#)
  - .defineDataInterface, VolcanoPlot-method (VolcanoPlot-class), [12](#)
  - .fullName, DETable-method (DETable-class), [4](#)
  - .fullName, LogFCLogFCPlot-method (LogFCLogFCPlot-class), [10](#)
  - .fullName, MAPlot-method (MAPlot-class), [11](#)
  - .fullName, VolcanoPlot-method (VolcanoPlot-class), [12](#)
  - .generateDotPlotData, LogFCLogFCPlot-method (LogFCLogFCPlot-class), [10](#)
  - .generateDotPlotData, MAPlot-method (MAPlot-class), [11](#)
  - .generateDotPlotData, VolcanoPlot-method (VolcanoPlot-class), [12](#)
  - .generateTable, DETable-method (DETable-class), [4](#)
  - .panelColor, DETable-method (DETable-class), [4](#)
  - .panelColor, LogFCLogFCPlot-method (LogFCLogFCPlot-class), [10](#)
  - .panelColor, MAPlot-method (MAPlot-class), [11](#)
  - .panelColor, VolcanoPlot-method (VolcanoPlot-class), [12](#)
  - .refineParameters, DETable-method (DETable-class), [4](#)
  - .refineParameters, LogFCLogFCPlot-method (LogFCLogFCPlot-class), [10](#)
  - .refineParameters, MAPlot-method (MAPlot-class), [11](#)
  - .refineParameters, VolcanoPlot-method (VolcanoPlot-class), [12](#)
- averageLog2 (de-generics), [3](#)
- averageLog2, iSEEDESeq2Results-method (iSEEDESeq2Results-class), [5](#)
- averageLog2, iSEEedgeRResults-method (iSEEedgeRResults-class), [7](#)
- averageLog2, iSEELimmaResults-method (iSEELimmaResults-class), [8](#)
- contrastResults (contrastResultsNames), [2](#)
- contrastResultsNames, [2](#)
- DataFrame, [3](#), [5](#), [7](#), [9](#)
- de-generics, [3](#)
- DETable (DETable-class), [4](#)
- DETable-class, [4](#)
- DotPlot, [10–12](#)

- embedContrastResults  
(utils-SummarizedExperiment),  
[11](#)
- embedContrastResults(), [6](#), [7](#), [9](#)
- embedContrastResults, ANY-method  
(utils-SummarizedExperiment),  
[11](#)
- embedContrastResults, data.frame-method  
(utils-SummarizedExperiment),  
[11](#)
- embedContrastResults, DESeqResults-method  
(iSEEDESeq2Results-class), [5](#)
- embedContrastResults, iSEEDESeq2Results-method  
(iSEEDESeq2Results-class), [5](#)
- embedContrastResults, iSEEedgeRResults-method  
(iSEEedgeRResults-class), [7](#)
- embedContrastResults, iSEELimmaResults-method  
(iSEELimmaResults-class), [8](#)
- embedContrastResults, TopTags-method  
(iSEEedgeRResults-class), [7](#)
- embedContrastResultsMethods  
(utils-SummarizedExperiment),  
[11](#)
- initialize, DETable-method  
(DETable-class), [4](#)
- initialize, LogFCLogFCPlot-method  
(LogFCLogFCPlot-class), [10](#)
- initialize, MAPlot-method  
(MAPlot-class), [11](#)
- initialize, VolcanoPlot-method  
(VolcanoPlot-class), [12](#)
- iSEEdede (iSEEdede-pkg), [5](#)
- iSEEdede-package (iSEEdede-pkg), [5](#)
- iSEEdede-pkg, [5](#)
- iSEEDESeq2Results  
(iSEEDESeq2Results-class), [5](#)
- iSEEDESeq2Results-class, [5](#)
- iSEEedgeRResults  
(iSEEedgeRResults-class), [7](#)
- iSEEedgeRResults-class, [7](#)
- iSEELimmaResults  
(iSEELimmaResults-class), [8](#)
- iSEELimmaResults-class, [8](#)
- log2FoldChange (de-generics), [3](#)
- log2FoldChange, iSEEDESeq2Results-method  
(iSEEDESeq2Results-class), [5](#)
- log2FoldChange, iSEEedgeRResults-method  
(iSEEedgeRResults-class), [7](#)
- log2FoldChange, iSEELimmaResults-method  
(iSEELimmaResults-class), [8](#)
- LogFCLogFCPlot (LogFCLogFCPlot-class),  
[10](#)
- LogFCLogFCPlot-class, [10](#)
- MAPlot (MAPlot-class), [11](#)
- MAPlot-class, [11](#)
- Panel, [10–12](#)
- pValue (de-generics), [3](#)
- pValue, iSEEDESeq2Results-method  
(iSEEDESeq2Results-class), [5](#)
- pValue, iSEEedgeRResults-method  
(iSEEedgeRResults-class), [7](#)
- pValue, iSEELimmaResults-method  
(iSEELimmaResults-class), [8](#)
- RowDataPlot, [10–12](#)
- RowDotPlot, [10–12](#)
- RowTable, [4](#)
- showAsCell, iSEEDESeq2Results-method  
(iSEEDESeq2Results-class), [5](#)
- showAsCell, iSEEedgeRResults-method  
(iSEEedgeRResults-class), [7](#)
- showAsCell, iSEELimmaResults-method  
(iSEELimmaResults-class), [8](#)
- SummarizedExperiment, [2](#), [5](#), [7](#), [9](#), [11](#), [12](#)
- Table, [4](#)
- utils-SummarizedExperiment, [11](#)
- VolcanoPlot (VolcanoPlot-class), [12](#)
- VolcanoPlot-class, [12](#)