Package 'SwarnSeq'

September 16, 2026

Type Package

Title Differential Expression and Differential Zero Inflation analysis

Version 0.99.5

Description
This R package performs differential expression and differential zero inflation analysis of single-cell RNA-seq (scRNA-seq) UMI counts data through adjusting cell capture efficiency.

License GPL-3

Encoding UTF-8

LazyData false

Depends R (>= 4.6.0)

Suggests knitr, rmarkdown, testthat (>= 3.0.0), BiocStyle

Imports stats, MASS, edgeR, SingleCellExperiment, SummarizedExperiment

biocViews Software, SingleCell, RNASeq, DifferentialExpression, GeneExpression, StatisticalMethod

URL https://github.com/nifmd-bbf/SwarnSeq

BugReports https://github.com/nifmd-bbf/SwarnSeq/issues

VignetteBuilder knitr

Config/testthat/edition 3

Config/roxygen2/version 8.0.0

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

git_branch devel

git_last_commit 651b42d

git_last_commit_date 2026-07-22

Repository Bioconductor 3.24

Date/Publication 2026-09-15

Author Samarendra Das [aut, cre] (ORCID: <https://orcid.org/0000-0002-0263-7027>), Satyajit Chhatoi [aut], Indian Council of Agricultural Research [fnd], Science and Engineering Research Board [fnd] (Core Research Grant (CRG004960))

Maintainer Samarendra Das <samarendra.das@icar.org.in>

Contents

`capEff` *Cell Capture Efficiency Estimating Function*

Description

This function estimates the capture efficiencies of cells from single-cell RNA-seq studies. This function takes ERCC spike-in transcript and molecular concentration data, if available. If spike-ins are not available, it uses count expression data.

Usage

```
capEff(sce, CE.range, RNAspike.use = FALSE, spike_in_sce, method = "ML")
```

Arguments

`sce`	This is a gene-by-cell raw count matrix single cell experiment object with colData contains 'clusters', 'groups' and 'auxil' as a data frame object.
`CE.range`	This is a two-element vector that sets the lower and upper limits for the estimated range of capture efficiencies.
`RNAspike.use`	This is a logical parameter. If this parameter is set to TRUE, then it requires you to provide spike counts (spike) and spike concentration (spike.conc) information.
`spike_in_sce`	This is a single cell experiment object which contains the spike-in counts data as the main assay and spike concentrations as its row data. # spike-in counts: This is the observed count matrix for spike-in transcripts, where each row is a spike-in and each column is a cell. It's only required if you set RNAspike.use to TRUE. # spike concentrations: This is a vector of theoretical counts for each spike-in transcript in a single cell, and it's only required if you set RNAspike.use to TRUE.
`method`	A string that specifies the method for computing capture efficiencies. ("ML" or "")

Value

This function returns the capture efficiencies of all the cells as a vector.

Examples

```
library(SingleCellExperiment)
# Load the test data.
data(SwarnSeqToyData); data(SpikeInData)
data <- assays(SwarnSeqToyData)[[1]][1:20, c(1:50, 350:399)]
groups <- SwarnSeqToyData$groups[c(1:50, 350:399)]
clusters <- SwarnSeqToyData$clusters[c(1:50, 350:399)]
X <- data.frame(clusters = clusters, groups = groups)
testData <- SingleCellExperiment(assays = list(counts = data), colData = X)

capeff <- capEff(sce = testData,CE.range = c(0.01, 0.05),RNAspike.use = FALSE)
```

---

dzinb *Probability Density for the Zero-Inflated Negative Binomial Distribution*

Description

This function calculates the probability density for a Zero-Inflated Negative Binomial (ZINB) distribution.

Usage

```
dzinb(x, size, mu, rho, log)
```

Arguments

x	This is a vector of non-negative counts for a gene across all the cells.
size	This is the over-dispersion parameter of the negative binomial distribution.
mu	The mean parameter of the negative binomial distribution.
rho	The zero-inflation parameter, a probability value between 0 and 1.
log	A logical value. If TRUE, this function returns the logarithm of the density. Otherwise, it returns the density itself.

Value

The probability density.

---

extAdjNormData *Two-Factor Normalized Counts Matrix Splitting Function*

Description

This function adjusts the scRNA-seq data by capture efficiency for each cell, then normalizes the adjusted matrix, and finally splits the matrix into two distinct matrices based on the number of factors in the group vector.

Usage

```
extAdjNormData(
  sce,
  norm.method = c("DEseq.norm", "TMM", "log1p"),
  method = "ML",
  RNAspike.use = FALSE,
  spike_in_sce,
  CE.range = c(0.01, 0.5)
)
```

Arguments

sce	This is a gene-by-cell raw count matrix single cell experiment object with colData contains 'clusters', 'groups' and 'auxil' as a data frame object. # groups: This is a vector of factors with 2 levels that serves as a grouping variable that assigns each column (cell) of the count matrix to one of two predefined experimental conditions.
norm.method	There are three distinct statistical normalization methods for scRNA-seq count data:DESeq2, a maximum likelihood approach described by Ye et al. (2027); TMM, a trimmed mean method introduced by Robinson et al. (2010), both designed to correct for library size and compositional biases; and a third method, which is the log1p standard normalization method for scRNA-seq data.
method	A string that specifies the method for computing capture efficiencies. ("ML" or "")
RNAspike.use	This is a logical parameter. If this parameter is set to TRUE, then it requires you to provide spike counts (spike) and spike concentration (spike.conc) information.
spike_in_sce	This is a single cell experiment object which contains the spike-in counts data as the main assay and spike concentrations as its row data. # spike-in counts: This is the observed count matrix for spike-in transcripts, where each row is a spike-in and each column is a cell. It's only required if you set RNAspike.use to TRUE. # spike concentrations: This is a vector of theoretical counts for each spike-in transcript in a single cell, and it's only required if you set RNAspike.use to TRUE.
CE.range	This is a two-element vector that sets the lower and upper limits for the estimated range of capture efficiencies.

Value

This function returns a list of two distinct capture efficiency adjusted normalized counts matrices, based on the number of factors in the group vector for the scRNA-seq data.

Examples

```
library(SingleCellExperiment)
# Load the test data.
data(SwarnSeqToyData); data(SpikeInData)
data <- assays(SwarnSeqToyData)[[1]][1:20, c(1:50, 350:399)]
groups <- SwarnSeqToyData$groups[c(1:50, 350:399)]
clusters <- SwarnSeqToyData$clusters[c(1:50, 350:399)]
X <- data.frame(clusters = clusters, groups=groups)
testData <- SingleCellExperiment(assays=list(counts = data),colData=X)
```

```
ExtData <- extAdjNormData(sce=testData,norm.method="log1p",CE.range=c(0.01,0.5))
```

SpikeInData *Spike-in counts data for the SwarnSeqToyData.*

Description

A SingleCellExperiment object containing the spike-in counts and spike concentrations.

Format

SingleCellExperiment Calss.

CountData This is a gene-by-cell raw count matrix.

Value

SingleCellExperiment spike-in counts data for SwarnSeqToyData.

Source

Petropoulos, S., Edsgård, D., Reinius, B., Deng, Q., Panula, S. P., Codeluppi, S., Plaza Reyes, A., Linnarsson, S., Sandberg, R., & Lanner, F. (2016). Single-Cell RNA-Seq Reveals Lineage and X Chromosome Dynamics in Human Preimplantation Embryos. Cell, 165(4), 1012–1026. https://doi.org/10.1016/j.cell.2016.03.023

---

swarnAdjLrt *Differential gene expression analysis for single-cell data*

Description

This function performs differential gene expression (DEG) analysis on a single-cell RNA-sequencing (scRNA-seq) raw read counts matrix. The input is a non-negative integer matrix, where rows correspond to genes and columns to individual cells. The primary objective is to identify genes with statistically significant changes in expression levels between teo predefined groups of cells. This analysis is a downstream step following the initial read mapping and gene counting, which generates the input matrix. By contrasting the expression profiles of the two groups, the function identifies biomarkers or genes that characterize the biological differences between them.

Usage

```
swarnAdjLrt(
  sce,
  norm.method = c("DEseq.norm", "TMM", "log1p"),
  maxit = 100,
  eps = 1e-04,
  muoffset = NULL,
  phioffset = NULL,
  weights = NULL,
  method = "ML",
```

```
  p.value.adj.method = "BH",
  RNAspike.use = FALSE,
  spike_in_sce,
  CE.range = c(0.01, 0.5)
)
```

Arguments

sce	This is a gene-by-cell raw count matrix single cell experiment object with colData contains 'clusters', 'groups' and 'auxil' as a data frame object. # clusters: This is a vector of factors with the optimal number of cluster levels, with each entry assigning a cell from the count matrix's columns to its computationally inferred cluster identity. # groups: This is a vector of factors with 2 levels that serves as a grouping variable that assigns each column (cell) of the count matrix to one of two predefined experimental conditions. # 'auxil' is for CellAuxil: This is a vector of factors of cell-level metadata that provides additional information about each cell, such as its batch, donor, or quality control metrics, with each entry aligning to a column in the count matrix.
norm.method	There are three distinct statistical normalization methods for scRNA-seq count data:DESeq2, a maximum likelihood approach described by Ye et al. (2027); TMM, a trimmed mean method introduced by Robinson et al. (2010), both designed to correct for library size and compositional biases; and a third method, which is the log1p standard normalization method for scRNA-seq data.
maxit	This is the maximum number of iterations for the Expected-Maximization (EM) algorithm.
eps	This is the convergence criteria for the Expected-Maximization (EM) algorithm.
muoffset	This is the offset parameter for mean (mu), with a default value of NULL.
phioffset	This is the offset parameter for the zero inflation (phi) parameter, with a default value of NULL.
weights	This is the observation-wise weight for the cells, with a default value of NULL.
method	A string that specifies the method for computing capture efficiencies. ("ML" or "")
p.value.adj.method	A string that specifies the method for multiple hypothesis correction, which can be any of the following values: "holm," "hochberg," "hommel," "bonferroni," "BH," "BY," "fdr," or "none").
RNAspike.use	This is a logical parameter. If this parameter is set to TRUE, then it requires you to provide spike counts (spike) and spike concentration (spike.conc) information.
spike_in_sce	This is a single cell experiment object which contains the spike-in counts data as the main assay and spike concentrations as its row data. # spike-in counts: This is the observed count matrix for spike-in transcripts, where each row is a spike-in and each column is a cell. It's only required if you set RNAspike.use to TRUE. # spike concentrations: This is a vector of theoretical counts for each spike-in transcript in a single cell, and it's only required if you set RNAspike.use to TRUE.
CE.range	This is a two-element vector that sets the lower and upper limits for the estimated range of capture efficiencies.

Value

This function analyzes single-cell RNA-sequencing adjusted normalized data to find differentially expressed genes and returns a list of the results.

Examples

```
library(SingleCellExperiment)
# Load the test data.
data(SwarnSeqToyData); data(SpikeInData)
data <- assays(SwarnSeqToyData)[[1]][1:20, c(1:50, 350:399)]
groups <- SwarnSeqToyData$groups[c(1:50, 350:399)]
clusters <- SwarnSeqToyData$clusters[c(1:50, 350:399)]
X <- data.frame(clusters = clusters, groups = groups)
testData <- SingleCellExperiment(assays = list(counts = data), colData = X)
# SpikeInData <- SingleCellExperiment(assays = list(SpikeCounts),rowData=SpikeConc)
# Make the spike-in single cell experiment object like this.

res <- swarnAdjLrt(sce=testData,norm.method="log1p",RNAspike.use=FALSE)
```

---

swarnClassDe *ScRNA-seq Influential Gene Classification Function*

Description

This function is used to clasify the influential genes of single-cell RNA-seq (scRNA-seq) data.

Usage

```
swarnClassDe(results, alpha)
```

Arguments

results	This is obtained from 'SwarnUnadjLRT' or 'SwarnAdjLRT' function.
alpha	Level of Significance

Value

It returns the list with one more element, SwarnClassDE, that represents the SwarnClass.

Examples

```
library(SingleCellExperiment)
# Load the test data.
data(SwarnSeqToyData); data(SpikeInData)
data <- assays(SwarnSeqToyData)[[1]][1:20, c(1:50, 350:399)]
groups <- SwarnSeqToyData$groups[c(1:50, 350:399)]
clusters <- SwarnSeqToyData$clusters[c(1:50, 350:399)]

X <- data.frame(clusters = clusters, groups = groups)
testData <- SingleCellExperiment(assays = list(counts = data), colData = X)
# SpikeInData <- SingleCellExperiment(assays=list(SpikeCounts),rowData=SpikeConc)
# Make the spike-in single cell experiment object like this.
```

```
res <- swarnAdjLrt(sce=testData,norm.method="log1p",RNAspike.use=TRUE,spike_in_sce=SpikeInData)
SwarnClass <- swarnClassDe(results = res, alpha = 0.01)
```

`SwarnSeqToyData` *A test scRNA-seq Real Dataset for SwarnSeq*

Description

A test dataset containing a single-cell RNA-seq (scRNA-seq) read counts matrix and column metadata.

Format

SingleCellExperiment Calss.

CountData This is a gene-by-cell raw count matrix.

Value

SingleCellExperiment object of raw scRNA-seq data.

Source

Petropoulos, S., Edsgård, D., Reinius, B., Deng, Q., Panula, S. P., Codeluppi, S., Plaza Reyes, A., Linnarsson, S., Sandberg, R., & Lanner, F. (2016). Single-Cell RNA-Seq Reveals Lineage and X Chromosome Dynamics in Human Preimplantation Embryos. Cell, 165(4), 1012–1026. https://doi.org/10.1016/j.cell.2016.03.023

`swarnTopTags` *Most Statistically Significant Gene Filtering Function*

Description

This function selects the most statistically significant genes in your scRNA-seq data.

Usage

```
swarnTopTags(results, m)
```

Arguments

`results`	This is obtained from 'SwarnUnadjLRT' or 'SwarnAdjLRT' function.
`m`	This is an integer that represents the number of statistically significant genes to be selected from the downstream analysis.

Value

A list of the top genes along with their statistics.

Examples

```
library(SingleCellExperiment)
# Load the test data.
data(SwarnSeqToyData); data(SpikeInData)
data <- assays(SwarnSeqToyData)[[1]][1:20, c(1:50, 350:399)]
groups <- SwarnSeqToyData$groups[c(1:50, 350:399)]
clusters <- SwarnSeqToyData$clusters[c(1:50, 350:399)]

X <- data.frame(clusters = clusters, groups = groups)
testData <- SingleCellExperiment(assays = list(counts = data), colData = X)
# SpikeInData <- SingleCellExperiment(assays = list(SpikeCounts), rowData = SpikeConc)
# Make the spike-in single cell experiment object like this.

res <- swarnAdjLrt(sce=testData,norm.method="DEseq.norm", RNAspike.use = FALSE)
top_genes <- swarnTopTags(results = res, m = 50)
```

`swarnUnadjLrt` *Differential gene expression analysis for single-cell data without Capture Efficiency Correction*

Description

This function performs differential gene expression (DEG) analysis on a single-cell RNA-sequencing (scRNA-seq) raw read counts matrix. The input is a non-negative integer matrix, where rows correspond to genes and columns to individual cells. The primary objective is to identify genes with statistically significant changes in expression levels between teo predefined groups of cells. This analysis is a downstream step following the initial read mapping and gene counting, which generates the input matrix. By contrasting the expression profiles of the two groups, the function identifies biomarkers or genes that characterize the biological differences between them.

Usage

```
swarnUnadjLrt(
  sce,
  norm.method = c("DEseq.norm", "TMM", "log1p"),
  maxit = 100,
  eps = 1e-04,
  muoffset = NULL,
  phioffset = NULL,
  weights = NULL,
  method = NA,
  p.value.adj.method = "BH"
)
```

Arguments

`sce`	This is a gene-by-cell raw count matrix single cell experiment object with colData contains 'clusters', 'groups' and 'auxil' as a data frame object. # clusters: This is a vector of factors with the optimal number of cluster levels, with each entry assigning a cell from the count matrix's columns to its computationally inferred cluster identity. # groups: This is a vector of factors with 2 levels that serves as a grouping variable that assigns each column (cell) of the count matrix

to one of two predefined experimental conditions. # 'auxil' is for CellAuxil: This is a vector of factors of cell-level metadata that provides additional information about each cell, such as its batch, donor, or quality control metrics, with each entry aligning to a column in the count matrix.

`norm.method` There are three distinct statistical normalization methods for scRNA-seq count data:DESeq2, a maximum likelihood approach described by Ye et al. (2027); TMM, a trimmed mean method introduced by Robinson et al. (2010), both designed to correct for library size and compositional biases; and a third method, which is the log1p standard normalization method for scRNA-seq data.

`maxit` This is the maximum number of iterations for the Expected-Maximization (EM) algorithm.

`eps` This is the convergence criteria for the Expected-Maximization (EM) algorithm.

`muoffset` This is the offset parameter for mean (mu), with a default value of NULL.

`phioffset` This is the offset parameter for the zero inflation (phi) parameter, with a default value of NULL.

`weights` This is the observation-wise weight for the cells, with a default value of NULL.

`method` A string that specifies the method for computing capture efficiencies. ("ML" or "")

`p.value.adj.method`
A string that specifies the method for multiple hypothesis correction, which can be any of the following values: "holm," "hochberg," "hommel," "bonferroni," "BH," "BY," "fdr," or "none").

Value

This function analyzes single-cell RNA-sequencing unadjusted normalized data to find differentially expressed genes and returns a list of the results.

Examples

```
library(SingleCellExperiment)
# Load the test data.
data(SwarnSeqToyData)
data <- assays(SwarnSeqToyData)[[1]][1:20, c(1:50, 350:399)]
groups <- SwarnSeqToyData$groups[c(1:50, 350:399)]
clusters <- SwarnSeqToyData$clusters[c(1:50, 350:399)]

X <- data.frame(clusters = clusters, groups = groups)
testData <- SingleCellExperiment(assays = list(counts = data), colData = X)

res <- swarnUnadjLrt(sce = testData, norm.method = "DEseq.norm")
```

ZINBEM *Estimating Parameters via the Expected-Maximization algorithm.*

Description

This function uses the Expected-Maximization (EM) algorithm to estimate the maximum likelihood estimates (MLE) of the parameters.

Usage

```
ZINBEM(
  counts_data,
  group,
  CellCluster,
  CellAuxil = NULL,
  weights = NULL,
  muoffset = NULL,
  phioffset = NULL,
  maxit = NULL,
  eps = NULL
)
```

Arguments

counts_data	This is a gene-by-cell raw count matrix.
group	This is a vector of factors with 2 levels that serves as a grouping variable that assigns each column (cell) of the count matrix to one of two predefined experimental conditions.
CellCluster	This is a vector of factors with the optimal number of cluster levels, with each entry assigning a cell from the count matrix's columns to its computationally inferred cluster identity.
CellAuxil	This is a vector of factors of cell-level metadata that provides additional information about each cell, such as its batch, donor, or quality control metrics, with each entry aligning to a column in the count matrix.
weights	This is the observation-wise weight for the cells, with a default value of NULL.
muoffset	This is the offset parameter for mean (mu), with a default value of NULL.
phioffset	This is the offset parameter for the zero inflation (phi) parameter, with a default value of NULL.
maxit	This is the maximum number of iterations for the Expected-Maximization (EM) algorithm.
eps	This is the convergence criteria for the Expected-Maximization (EM) algorithm.

Value

This function returns a list of results.

Index