Package: scDD (via r-universe)

October 27, 2026

Version 1.37.0

Title Mixture modeling of single-cell RNA-seq data to identify genes with differential distributions

Description This package implements a method to analyze single-cell RNA- seq Data utilizing flexible Dirichlet Process mixture models. Genes with differential distributions of expression are classified into several interesting patterns of differences between two conditions. The package also includes functions for simulating data with these patterns from negative binomial distributions.

Depends R (>= 3.4)

NeedsCompilation yes

Imports fields, mclust, BiocParallel, outliers, ggplot2, EBSeq, arm, SingleCellExperiment, SummarizedExperiment, grDevices, graphics, stats, S4Vectors, scran

Suggests BiocStyle, knitr, gridExtra

License GPL-2

RoxygenNote 6.0.1

VignetteBuilder knitr

biocViews ImmunoOncology, Bayesian, Clustering, RNASeq, SingleCell, MultipleComparison, Visualization, DifferentialExpression

URL https://github.com/kdkorthauer/scDD

BugReports https://github.com/kdkorthauer/scDD/issues

Config/pak/sysreqs cmake libglpk-dev make libuv1-dev libxml2-dev zlib1g-dev

Repository https://bioc.r-universe.dev

Date/Publication 2026-04-28 12:44:51 UTC

RemoteUrl https://github.com/bioc/scDD

RemoteRef HEAD

RemoteSha 287a78c1e1f9ac8ea578f2781716b6df5d5b37da

Contents

calcMV . . . 2
calcRP . . . 3
classifyDD . . . 4
feDP . . . 5
findFC . . . 6
findIndex . . . 7
findOutliers . . . 8
getPosteriorParams . . . 9
jointPosterior . . . 10
lu . . . 11
luOutlier . . . 11
mclustRestricted . . . 12
permMclust . . . 13
permMclustCov . . . 14
permMclustGene . . . 15
permZero . . . 16
preprocess . . . 17
results . . . 19
scDatEx . . . 20
scDatExList . . . 21
scDatExSim . . . 21
scDD . . . 22
sideHist . . . 26
sideViolin . . . 26
simuDB . . . 28
simuDE . . . 29
simuDM . . . 30
simuDP . . . 32
simulateSet . . . 33
singleCellSimu . . . 35
testKS . . . 36
testZeroes . . . 38
validation . . . 38

Index **40**

`calcMV` *calcMV*

Description

Calculate empirical means and variances of selected genes in a given dataset.

Usage

```
calcMV(a, FC = 1, FC.thresh = NA, threshold = Inf,
  include.zeroes = FALSE)
```

Arguments

a	Numeric vector of values to calculate empirical mean and variance.
FC	Fold change for the mean and standard deviation. Default value is 1.
FC.thresh	Alternate fold change for the mean and standard deviation when the (log nonzero) mean is above the value of threshold. Default value is FC.
threshold	Mean threshold value which dictates which fold change value to use for multiplying mean and standard deviation. Default value is Inf (so FC is always used).
include.zeroes	Logical value indicating whether the zero values should be included in the calculations of the empirical means and variances.

Details

Calculate empirical means and variances of selected genes in a given dataset. Optionally, multiply the means and standard deviations by a fold change value, which can also vary by mean value. If the mean is below some specified threshold threshold, use one fold change value FC. If above the threshold, use the alternate fold change value FC.thresh. Estimates of mean and variance are robust to outliers.

Value

MV Vector of two elements, first contains the empirical mean estimate, second contains the empirical variance estimate (optionally multiplied by a fold change).

---

calcRP *calcRP*

Description

Calculate parameter R and P in NB distribution

Usage

```
calcRP(Emean, Evar)
```

Arguments

Emean	Empirical mean
Evar	Empirical variance

Value

RP Vector of two elements, first contains method of moments estimator for r and second contains method of moments estimator for p (parameters of NB distribution)

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

`classifyDD` *classifyDD*

Description

Classify significantly DD genes into the four categories (DE, DP, DM or DB) based on posterior distributions of cluster mean parameters

Usage

```
classifyDD(pe_mat, condition, sig_genes, oa, c1, c2, alpha, m0, s0, a0, b0,
  log.nonzero = TRUE, adjust.perms = FALSE, ref, min.size = 3)
```

Arguments

`pe_mat`	Matrix with genes in rows and samples in columns. Column names indicate condition.
`condition`	Vector of condition indicators (with two possible values).
`sig_genes`	Vector of the indices of significantly DD genes (indicating the row number of `pe_mat`)
`oa`	List item with one item for each gene where the first element contains the cluster membership for each nonzero sample in the overall (pooled) fit.
`c1`	List item with one item for each gene where the first element contains the cluster membership for each nonzero sample in condition 1 only fit
`c2`	List item with one item for each gene where the first element contains the cluster membership for each nonzero sample in condition 2 only fit
`alpha`	Value for the Dirichlet concentration parameter
`m0`	Prior mean value for generating distribution of cluster means
`s0`	Prior precision value for generating distribution of cluster means
`a0`	Prior shape parameter value for the generating distribution of cluster precision
`b0`	Prior scale parameter value for the generating distribution of cluster precision
`log.nonzero`	Logical indicating whether to perform log transformation of nonzero values.
`adjust.perms`	Logical indicating whether or not to adjust the permutation tests for the sample detection rate (proportion of nonzero values). If true, the residuals of a linear model adjusted for detection rate are permuted, and new fitted values are obtained using these residuals.
`ref`	one of two possible values in condition; represents the referent category.

`min.size` a positive integer that specifies the minimum size of a cluster (number of cells) for it to be used during the classification step. Any clusters containing fewer than `min.size` cells will be considered an outlier cluster and ignored in the classfication algorithm. The default value is three.

Value

cat Character vector of the same length as `sig_genes` that indicates which category of DD each significant gene belongs to (DE, DP, DM, DB, or NC (no call))

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

`feDP` *feDP*

Description

Function to identify additional DP genes, since clustering process can be consistent within each condition and still have differential proportion within each mode. The Bayes factor score also tends to be small when the correct number of clusters is not correctly detected; in that case differential proportion will manifest as a mean shift.

Usage

```
feDP(pe_mat, condition, sig_genes, oa, c1, c2, log.nonzero = TRUE,
  testZeroes = FALSE, adjust.perms = FALSE, min.size = 3)
```

Arguments

`pe_mat`	Matrix with genes in rows and samples in columns. Column names indicate condition.
`condition`	Vector of condition indicators (with two possible values).
`sig_genes`	Vector of the indices of significantly DD genes (indicating the row number of `pe_mat`)
`oa`	List item with one item for each gene where the first element contains the cluster membership for each nonzero sample in the overall (pooled) fit.
`c1`	List item with one item for each gene where the first element contains the cluster membership for each nonzero sample in condition 1 only fit
`c2`	List item with one item for each gene where the first element contains the cluster membership for each nonzero sample in condition 2 only fit
`log.nonzero`	Logical indicating whether to perform log transformation of nonzero values.

testZeroes — Logical indicating whether or not to test for a difference in the proportion of zeroes. This will only be done for genes that have at least one zero value (genes where all cells have a nonzero value will have a ‘zero.pvalue‘ of NA).

adjust.perms — Logical indicating whether or not to adjust the permutation tests for the sample detection rate (proportion of nonzero values). If true, the residuals of a linear model adjusted for detection rate are permuted, and new fitted values are obtained using these residuals.

min.size — a positive integer that specifies the minimum size of a cluster (number of cells) for it to be used during the classification step. Any clusters containing fewer than min.size cells will be considered an outlier cluster and ignored in the classfication algorithm. The default value is three.

Details

The Fisher's Exact test is used to test for independence of condition membership and clustering when the clustering is the same across conditions as it is overall (and is multimodal). When clustering within condition is not multimodal or is different across conditions (most often the case), an FDR-adjusted t-test is performed to detect overall mean shifts.

Value

cat Character vector of the same length as sig_genes that indicates which nonsignificant genes by the permutation test belong to the DP category

---

findFC *findFC*

Description

Find the appropriate Fold Change vectors for simulation that will be use in classic differential expression case.

Usage

```
findFC(SCdat, index, sd.range = c(1, 3), N = 4, overExpressionProb = 0.5,
  plot.FC = FALSE, condition = "condition")
```

Arguments

SCdat — An object of class SingleCellExperiment that contains normalized single-cell expression and metadata. The assays slot contains a named list of matrices, where the normalized counts are housed in the one named normcounts. This matrix should have one row for each gene and one sample for each column. The colData slot should contain a data.frame with one row per sample and columns that contain metadata for each sample. This data.frame should contain a variable that represents biological condition, which is in the form of numeric values (either 1 or 2) that indicates which condition each sample belongs to (in the same

order as the columns of `normcounts`). Optional additional metadata about each cell can also be contained in this data.frame, and additional information about the experiment can be contained in the `metadata` slot as a list.

`index` Reasonable set of genes for simulation

`sd.range` Numeric vector of length two which describes the interval (lower, upper) of standard deviations of fold changes to randomly select.

`N` Integer value for the number of bins to divide range of fold changes for calculating standard deviations

`overExpressionProb` Numeric value between 0 and 1 which describes the ratio of over to under expression values to sample.

`plot.FC` Logical indicating whether or not to plot the observed and simulated log2 fold changes.

`condition` A character object that contains the name of the column in `colData` that represents the biological group or condition of interest (e.g. treatment versus control). Note that this variable should only contain two possible values since `scDD` can currently only handle two-group comparisons. The default option assumes that there is a column named "condition" that contains this variable.

Details

This code is a modified version of Sam Younkin's simulate FC function. Major things that were changed are (1) standard deviations are calculated only on the nonzeroes, (2) the sampling of FCs is uniform on the log scale instead of the raw scale, and (3) the binning is done by quantiles instead of evenly spaced along the average expression values.

Value

FC.vec Return Fold Change Vectors

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

`findIndex` *findIndex*

Description

Find a reasonable set of genes (one mode and at least 25 to use for simulation.

Usage

```
findIndex(SCdat, condition = "condition")
```

Arguments

SCdat	An object of class SingleCellExperiment that contains normalized single-cell expression and metadata. The assays slot contains a named list of matrices, where the normalized counts are housed in the one named normcounts. This matrix should have one row for each gene and one sample for each column. The colData slot should contain a data.frame with one row per sample and columns that contain metadata for each sample. This data.frame should contain a variable that represents biological condition, which is in the form of numeric values (either 1 or 2) that indicates which condition each sample belongs to (in the same order as the columns of normcounts). Optional additional metadata about each cell can also be contained in this data.frame, and additional information about the experiment can be contained in the metadata slot as a list.
condition	A character object that contains the name of the column in colData that represents the biological group or condition of interest (e.g. treatment versus control). Note that this variable should only contain two possible values since scDD can currently only handle two-group comparisons. The default option assumes that there is a column named "condition" that contains this variable.

Value

Vector of indices for a reasonable set of genes that can be used for simulation.

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

findOutliers *findOutliers*

Description

Find the clusters that are considered outliers

Usage

```
findOutliers(clustering, min.size = 3)
```

Arguments

clustering	Numeric vector of cluster membership (1st item (named class) in list returned by mclustRestricted)
min.size	Numeric value for the minimum number of samples a cluster must have to be considered in the robust count. Default is 3.

Details

Function to obtain a count of the number of clusters that is robust to outliers. Requires at least `min.size` samples to be considered in the robust count.

Value

The robust count of the number of unique clusters excluding those with less than `min.size` samples.

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

`getPosteriorParams` *getPosteriorParams*

Description

Given the observations for a single gene and its clustering information, return the calculated posterior parameters

Usage

```
getPosteriorParams(y, mcobj, alpha, m0, s0, a0, b0)
```

Arguments

`y`	Numeric data vector for one gene (log-transformed non-zeroes)
`mcobj`	Object returned by `mclustRestricted`
`alpha`	Value for the Dirichlet concentration parameter
`m0`	Prior mean value for generating distribution of cluster means
`s0`	Prior precision value for generating distribution of cluster means
`a0`	Prior shape parameter value for the generating distribution of cluster precision
`b0`	Prior scale parameter value for the generating distribution of cluster precision

Value

A list of posterior parameter values under the DP mixture model framework, given the data and prior parameter values.

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

jointPosterior *jointPosterior*

Description

Function to obtain the normalized joint posterior of the data and partition.

Usage

```
jointPosterior(y, mcobj, alpha, m0, s0, a0, b0)
```

Arguments

y	Numeric data vector for one gene (log-transformed non-zeroes)
mcobj	Object returned by mclustRestricted
alpha	Value for the Dirichlet concentration parameter
m0	Prior mean value for generating distribution of cluster means
s0	Prior precision value for generating distribution of cluster means
a0	Prior shape parameter value for the generating distribution of cluster precision
b0	Prior scale parameter value for the generating distribution of cluster precision

Details

Calculates the normalized joint posterior of the data and partition under the Product Partition Model formulation of the Dirichlet Process Mixture model.

Value

log joint posterior value

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

lu *lu*

Description

Shortcut for `length(unique())`

Usage

```
lu(x)
```

Arguments

`x`	Numeric vector of cluster membership (1st item (named `class`) in list returned by `mclustRestricted`)

Details

Function to obtain a count of the number of clusters

Value

The count of the number of unique clusters.

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

luOutlier *luOutlier*

Description

Count the number of clusters with at least `min.size` samples

Usage

```
luOutlier(x, min.size = 3)
```

Arguments

`x`	Numeric vector of cluster membership (1st item (named `class`) in list returned by `mclustRestricted`)
`min.size`	Numeric value for the minimum number of samples a cluster must have to be considered in the robust count. Default is 3.

Details

Function to obtain a count of the number of clusters that is robust to outliers. Requires at least `min.size` samples to be considered in the robust count.

Value

The robust count of the number of unique clusters excluding those with less than `min.size` samples.

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

`mclustRestricted` *mclustRestricted*

Description

Function to determine how many normal mixture components are present.

Usage

```
mclustRestricted(y, restrict = TRUE, min.size)
```

Arguments

`y`	Numeric vector of values to fit to a normal mixture model with Mclust.
`restrict`	Logical indicating whether or not to enforce the restriction on cluster separation based on bimodal index and ratio of largest to smallest variance (see details). If False, then Mclust results as is are returned.
`min.size`	a positive integer that specifies the minimum size of a cluster (number of cells) for it to be used during the classification step. A clustering with all clusters of size less than `min.size` is not valid and clusters will be merged if this happens.

Details

Robust to detecting multiple components that are close together by enforcing that the distance between two clusters of appreciable size (at least 4 samples), have sufficiently high bimodal index (cluster mean difference standardized by average standard deviation and multiplied by a balance factor which is one when clusters are perfectly balanced) and not have variances that differ by more than a ratio of 20. Bimodal index threshold is dependent on sample size to ensure consistent performance in power and type I error of detection of multiple components.

Value

List object with (1) vector of cluster membership, (2) cluster means, (3) cluster variances, (4) number of model parameters, (5) sample size, (6) BIC of selected model, and (6) loglikelihood of selected model.

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

`permMclust` *permMclust*

Description

Function to obtain bayes factor numerator for permutations of one gene

Usage

```
permMclust(y, nperms, condition, remove.zeroes = TRUE, log.transf = TRUE,
  restrict = FALSE, alpha, m0, s0, a0, b0, ref, min.size)
```

Arguments

`y`	Numeric data vector for one gene
`nperms`	Number of permutations of residuals to evaulate
`condition`	Vector of condition indicators for each sample
`remove.zeroes`	Logical indicating whether zeroes need to be removed from `y`
`log.transf`	Logical indicating whether the data is in the raw scale (if so, will be log-transformed)
`restrict`	Logical indicating whether to perform restricted Mclust clustering where close-together clusters are joined.
`alpha`	Value for the Dirichlet concentration parameter
`m0`	Prior mean value for generating distribution of cluster means
`s0`	Prior precision value for generating distribution of cluster means
`a0`	Prior shape parameter value for the generating distribution of cluster precision
`b0`	Prior scale parameter value for the generating distribution of cluster precision
`ref`	one of two possible values in condition; represents the referent category.
`min.size`	a positive integer that specifies the minimum size of a cluster (number of cells) for it to be used during the classification step. Any clusters containing fewer than `min.size` cells will be considered an outlier cluster and ignored in the classfication algorithm. The default value is three.

Details

Obtains bayes factor numerator for data vector y representing one gene

Value

Bayes factor numerator for the current permutation

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

`permMclustCov` *permMclustCov*

Description

Function to obtain bayes factor for permutations of one gene's residuals

Usage

```
permMclustCov(y, nperms, C, condition, remove.zeroes = TRUE,
  log.transf = TRUE, restrict = FALSE, alpha, m0, s0, a0, b0, ref, min.size)
```

Arguments

`y`	Numeric data vector for one gene
`nperms`	Number of permutations of residuals to evaulate
`C`	Matrix of confounder variables, where there is one row for each sample and one column for each covariate.
`condition`	Vector of condition indicators for each sample
`remove.zeroes`	Logical indicating whether zeroes need to be removed from y
`log.transf`	Logical indicating whether the data is in the raw scale (if so, will be log-transformed)
`restrict`	Logical indicating whether to perform restricted Mclust clustering where close-together clusters are joined.
`alpha`	Value for the Dirichlet concentration parameter
`m0`	Prior mean value for generating distribution of cluster means
`s0`	Prior precision value for generating distribution of cluster means
`a0`	Prior shape parameter value for the generating distribution of cluster precision
`b0`	Prior scale parameter value for the generating distribution of cluster precision
`ref`	one of two possible values in condition; represents the referent category.

`min.size` a positive integer that specifies the minimum size of a cluster (number of cells) for it to be used during the classification step. Any clusters containing fewer than `min.size` cells will be considered an outlier cluster and ignored in the classfication algorithm. The default value is three.

Details

Obtains bayes factor numerator for data vector y representing one gene

Value

Bayes factor numerator for the current permutation

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

`permMclustGene` *permMclustGene*

Description

Function to obtain bayes factor for all permutations of one gene (not parallelized; to be used when parallelizing over Genes)

Usage

```
permMclustGene(y, adjust.perms, nperms, condition, remove.zeroes = TRUE,
  log.transf = TRUE, restrict = TRUE, alpha, m0, s0, a0, b0, C, ref,
  min.size)
```

Arguments

`y` Numeric data vector for one gene

`adjust.perms` Logical indicating whether or not to adjust the permutation tests for the sample detection rate (proportion of nonzero values). If true, the residuals of a linear model adjusted for detection rate are permuted, and new fitted values are obtained using these residuals.

`nperms` Number of permutations of residuals to evaulate

`condition` A character object that contains the name of the column in `colData` that represents the biological group or condition of interest (e.g. treatment versus control). Note that this variable should only contain two possible values since `scDD` can currently only handle two-group comparisons. The default option assumes that there is a column named "condition" that contains this variable.

remove.zeroes	Logical indicating whether zeroes need to be removed from y
log.transf	Logical indicating whether the data is in the raw scale (if so, will be log-transformed)
restrict	Logical indicating whether to perform restricted Mclust clustering where close-together clusters are joined.
alpha	Value for the Dirichlet concentration parameter
m0	Prior mean value for generating distribution of cluster means
s0	Prior precision value for generating distribution of cluster means
a0	Prior shape parameter value for the generating distribution of cluster precision
b0	Prior scale parameter value for the generating distribution of cluster precision
C	Matrix of confounder variables, where there is one row for each sample and one column for each covariate.
ref	one of two possible values in condition; represents the referent category.
min.size	a positive integer that specifies the minimum size of a cluster (number of cells) for it to be used during the classification step. Any clusters containing fewer than min.size cells will be considered an outlier cluster and ignored in the classfication algorithm. The default value is three.

Details

Obtains bayes factor for data vector y representing one gene

Value

Bayes factor numerator for the current permutation

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

permZero *permZero*

Description

Function to generate random permutations of nonzero values.

Usage

```
permZero(m, size, zmat)
```

Arguments

m	Number of permuted sets to generate.
size	Number of samples present in the dataset
zmat	Matrix of indicators of whether the original data value is zero or not. Should contain the same number of rows and columns as original data matrix.

Details

Generates random permutations for all genes, where the zeroes are kept fixed (i.e. only permute the nonzero condition labels).

Value

a list of length 'm' (nperms) where each item is a 'ngenes' by 'size' matrix

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

preprocess *preprocess*

Description

Function to preprocess SingleCellExperiment object (1) to only keep genes with a certain number of nonzero entries, and (2) optionally apply a normalization procedure.

Usage

```
preprocess(SCdat, condition = "condition", zero.thresh = 0.9,
  scran_norm = FALSE, median_norm = FALSE)
```

Arguments

SCdat	An object of class SingleCellExperiment that contains single-cell expression and metadata. The assays slot contains a named list of matrices, where the normalized counts are housed in the one named normcounts, and unnormalized counts are stored in the one names counts. If either scran_norm or median_norm is set to TRUE, the normcounts slot will be created from the counts slot. The counts and normalized counts matrices should have one row for each gene and one sample for each column. The colData slot should contain a data.frame with one row per sample and columns that contain metadata for each sample. This data.frame should contain a variable that represents biological condition, which is in the form of numeric values (either 1 or 2) that indicates

which condition each sample belongs to (in the same order as the columns of normcounts). Optional additional metadata about each cell can also be contained in this data.frame, and additional information about the experiment can be contained in the metadata slot as a list.

condition — A character object that contains the name of the column in colData that represents the biological group or condition of interest (e.g. treatment versus control). Note that this variable should only contain two possible values since scDD can currently only handle two-group comparisons. The default option assumes that there is a column named "condition" that contains this variable.

zero.thresh — A numeric value between 0 and 1 that represents the maximum proportion of zeroes per gene allowable in the processed dataset

scran_norm — Logical indicating whether or not to normalize the data using scran Normalization from scran

median_norm — Logical indicating whether or not to normalize the data using Median Normalization from EBSeq

Value

An object of class SingleCellExperiment with genes removed if they have more than zero.thresh zeroes, and the normcounts assay added if either scran_norm or median_norm is set to TRUE and only counts is provided. If normcounts already exists and either scran_norm or median_norm is set to TRUE, then the new normalized counts are placed in the normcounts assay slot, and the original values are moved to a new slot called normcounts-orig.

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

Examples

```
# load toy example SingleCellExperiment object

data(scDatEx)

# apply the preprocess function to filter out genes if they have more than
# 75% zero

scDatEx <- preprocess(scDatEx, zero.thresh=0.75)

# apply the preprocess function again, but this time threshold on the
# proportion of zeroes and apply scran normalization
# set the zero.thresh argument to 0.75 so that genes with more than 75%
# zeroes are filtered out
# set the scran_norm argument to TRUE to return scran normalized counts

scDatEx.scran <- preprocess(scDatEx, zero.thresh=0.75, scran_norm=TRUE)
```

```
# set the median_norm argument to TRUE to return Median normalized counts

scDatEx.median <- preprocess(scDatEx, zero.thresh=0.75, median_norm=TRUE)
```

`results` *results*

Description

extract results objects after running scDD analysis

Usage

```
results(SCdat, type = c("Genes", "Zhat.c1", "Zhat.c2", "Zhat.combined"))
```

Arguments

`SCdat`	An object of class `SingleCellExperiment` that contains normalized single-cell expression and metadata, and the output of the `scDD` function.
`type`	A character variable specifying which output is desired, with possible values "Genes", "Zhat.c1", "Zhat.c2", and "Zhat.overall". The default value is "Genes", which contains a a data frame with nine columns: gene name (matches rownames of SCdat), permutation p-value for testing of independence of condition membership with clustering, Benjamini-Hochberg adjusted version of the previous column, p-value for test of difference in dropout rate (only for non-DD genes), Benjamini-Hochberg adjusted version of the previous column, name of the DD (DE, DP, DM, DB) pattern or DZ (otherwise NS = not significant), the number of clusters identified overall, the number of clusters identified in condition 1 alone, and the number of clusters identified in condition 2 alone. If `type` is "Zhat.c1", then a `matrix` is returned that contains the fitted cluster memberships (partition estimates Z) for each sample (cluster number given by 1,2,3,...) in columns and gene in rows only for condition 1. The same information is returned only for condition 2, and for the overall clustering, when `type` is set to "Zhat.c2" or "Zhat.overall", respectively. Zeroes, which are not involved in the clustering, are labeled as zero.

Details

Convenient helper function to extract the results (gene classifications, pvalues, and clustering information). Results data.frames/matrices are stored in the `metadata` slot and can also be accessed without the help of this convenience function by calling `metadata(SCdat)`.

Value

A `data.frame` which contains either the gene classification and p-value results, or cluster membership information, as detailed in the description of the `type` input parameter.

Examples

```
# load toy simulated example SingleCellExperiment object to find DD genes
data(scDatExSim)

# set arguments to pass to scDD function

prior_param=list(alpha=0.01, mu0=0, s0=0.01, a0=0.01, b0=0.01)

# call the scDD function to perform permutations and classify DD genes

scDatExSim <- scDD(scDatExSim, prior_param=prior_param, testZeroes=FALSE)

# extract main results object

RES <- results(scDatExSim)
```

---

scDatEx *Data: Toy example data*

Description

Toy example data in SingleCellExperiment format for 500 genes to illustrate how to generate simulated data from example data using simulateSet.

Usage

```
data(scDatEx)
```

Format

An object of class SingleCellExperiment containing data for 500 genes for 142 samples (78 from condition 1 and 64 from condition 2). Condition labels (1 or 2) are stored in the colData slot. The assays slot contains both normcounts and counts for illustration, but these objects are identical.

Value

An RData object, see Format section for details

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

Examples

```
# load toy example data
data(scDatEx)
```

scDatExList *Data: Toy example data list*

Description

Toy example data list (one item for each of two conditions) for 100 genes to illustrate how to use the function preprocess.

Usage

```
data(scDatExList)
```

Format

A list of two matrices (one for each of two conditions) labeled "C1" and "C2". Each matrix contains data for 100 genes and a variable number of samples (78 in C1 and 64 in C2).

Value

An RData object, see Format section for details

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

Examples

```
# load toy example data list
data(scDatExList)
```

scDatExSim *Data: Toy example of simulated data*

Description

Toy example data in SingleCellExperiment format for 500 genes to illustrate how to generate simulated data from example data using simulateSet. Contains 5 genes from each category (DE, DP, DM, DB, EE, and EP).

Usage

```
data(scDatExSim)
```

Format

An object of class `SingleCellExperiment` containing data for 30 genes for 200 samples (100 from condition 1 and 100 from condition 2). Condition labels (1 or 2) are stored in the colData slot. Row names of the assayData slot contain the two letter category label that the gene was simulated from (e.g. 'EE', 'DB', ...) along with the row number (1-30).

Value

An RData object, see Format section for details

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

Examples

```
# load toy example of simulated data
data(scDatExSim)
```

`scDD` *scDD*

Description

Find genes with differential distributions (DD) across two conditions

Usage

```
scDD(SCdat, prior_param = list(alpha = 0.1, mu0 = 0, s0 = 0.01, a0 = 0.01, b0
  = 0.01), permutations = 0, testZeroes = TRUE, adjust.perms = FALSE,
  param = bpparam(), parallelBy = c("Genes", "Permutations"),
  condition = "condition", min.size = 3, min.nonzero = NULL,
  level = 0.05, categorize = TRUE)
```

Arguments

`SCdat`	An object of class `SingleCellExperiment` that contains normalized single-cell expression and metadata. The `assays` slot contains a named list of matrices, where the normalized counts are housed in the one named `normcounts`. This matrix should have one row for each gene and one sample for each column. The `colData` slot should contain a data.frame with one row per sample and columns that contain metadata for each sample. This data.frame should contain a variable that represents biological condition, which is in the form of numeric values (either 1 or 2) that indicates which condition each sample belongs to (in the same order as the columns of `normcounts`). Optional additional metadata about each

cell can also be contained in this data.frame, and additional information about the experiment can be contained in the `metadata` slot as a list.

`prior_param` A list of prior parameter values to be used when modeling each gene as a mixture of DP normals. Default values are given that specify a vague prior distribution on the cluster-specific means and variances.

`permutations` The number of permutations to be used in calculating empirical p-values. If set to zero (default), the full Bayes Factor permutation test will not be performed. Instead, a fast procedure to identify the genes with significantly different expression distributions will be performed using the nonparametric Kolmogorov-Smirnov test, which tests the null hypothesis that the samples are generated from the same continuous distribution. This test will yield slightly lower power than the full permutation testing framework (this effect is more pronounced at smaller sample sizes, and is more pronounced in the DB category), but is orders of magnitude faster. This option is recommended when compute resources are limited. The remaining steps of the scDD framework will remain unchanged (namely, categorizing the significant DD genes into patterns that represent the major distributional changes, as well as the ability to visualize the results with violin plots using the `sideViolin` function).

`testZeroes` Logical indicating whether or not to test for a difference in the proportion of zeroes. This will only be done for genes that have at least one zero value (genes where all cells have a nonzero value will have a 'zero.pvalue' of NA).

`adjust.perms` Logical indicating whether or not to adjust the permutation tests for the sample detection rate (proportion of nonzero values). If true, the residuals of a linear model adjusted for detection rate are permuted, and new fitted values are obtained using these residuals.

`param` a `MulticoreParam` or `SnowParam` object of the `BiocParallel` package that defines a parallel backend. The default option is `BiocParallel::bpparam()` which will automatically creates a cluster appropriate for the operating system. Alternatively, the user can specify the number of cores they wish to use by first creating the corresponding `MulticoreParam` (for Linux-like OS) or `SnowParam` (for Windows) object, and then passing it into the `scDD` function. This could be done to specify a parallel backend on a Linux-like OS with, say 12 cores by setting `param=BiocParallel::MulticoreParam(workers=12)`

`parallelBy` For the permutation test (if invoked), the manner in which to parallelize. The default option is `"Genes"` which will spawn processes that divide up the genes across all cores defined in `param` cores, and then loop through the permutations. The alternate option is `"Permutations"` which loop through each gene and spawn processes that divide up the permutations across all cores defined in `param`. The default option is recommended when analyzing more genes than the number of permutations.

`condition` A character object that contains the name of the column in `colData` that represents the biological group or condition of interest (e.g. treatment versus control). Note that this variable should only contain two possible values since `scDD` can currently only handle two-group comparisons. The default option assumes that there is a column named "condition" that contains this variable.

`min.size` a positive integer that specifies the minimum size of a cluster (number of cells) for it to be used during the classification step. Any clusters containing fewer

than `min.size` cells will be considered an outlier cluster and ignored in the classfication algorithm. The default value is three.

`min.nonzero` a positive integer that specifies the minimum number of nonzero cells in each condition required for the test of differential distributions. If a gene has fewer nonzero cells per condition, it will still be tested for DZ (if `testZeroes` is TRUE). Default value is NULL (no minimum value is enforced).

`level` numeric value between 0 and 1 that specifies the alpha level for significance of a differential gene test (default value 0.05). This is used to decide whether to classify a gene into one of the differential patterns. If 'testZeroes' is FALSE and the adjusted p-value for a given gene is below 'level', then the gene is categorized. Alternatively, if 'testZeroes' is TRUE, then the adjusted p-value must be below 'level/2' in order to be considered significant and categorized. This is done to control for multiple testing since 'testZeroes=TRUE' means that each gene is tested for a difference in nonzeroes and zeroes separately.

`categorize` a logical indicating whether to determine which categories (DE, DP, DM, DB) each gene belongs to (default = TRUE). This can only be set to FALSE if 'permutations' is set to zero, since the full model fitting will automatically be carried out if permutations are run.

Details

Find genes with differential distributions (DD) across two conditions. Models each log-transformed gene as a Dirichlet Process Mixture of normals and uses a permutation test to determine whether condition membership is independent of sample clustering. The FDR adjusted (Benjamini-Hochberg) permutation p-value is returned along with the classification of each significant gene (with p-value less than 0.05 (or 0.025 if also testing for a difference in the proportion of zeroes)) into one of four categories (DE, DP, DM, DB). For genes that do not show significant influence, of condition on clustering, an optional test of whether the proportion of zeroes (dropout rate) is different across conditions is performed (DZ).

Value

A `SingleCellExperiment` object that contains the data and sample information from the input object, but where the results objects are now added to the `metadata` slot. The metadata slot is now a list with four items: the first (main results object) is a data.frame with the following columns:

- 'gene': gene name (matches rownames of SCdat)
- 'DDcategory': name of the DD (DE, DP, DM, DB, DZ) pattern (or NS = not significant)
- 'Clusters.combined': the number of clusters identified overall
- 'Clusters.C1': the number of clusters identified in condition 1 alone
- 'Clusters.C2': the number of clusters identified in condition 2 alone
- 'nonzero.pvalue': permutation (or KS) p-value for testing difference in nonzero expression values
- 'nonzero.pvalue.adj': Benjamini-Hochberg adjusted version of the 'nonzero.pvalue'column
- 'zero.pvalue': p-value for test of difference in dropout rate (only if 'testZeroes' is TRUE)

- 'zero.pvalue': Benjamini-Hochberg adjusted version of the previous column (only if 'testZeroes' is TRUE)
- 'combined.pvalue': Fisher's combined p-value for a difference in nonzero or zero values (only if 'testZeroes' is TRUE).
- 'combined.pvalue.adj': Benjamini-Hochberg adjusted version of the previous column (only if 'testZeroes' is TRUE)

The remaining three elements are matrices (first for condition 1 and 2 combined, then condition 1 alone, then condition 2 alone) that contains the cluster memberships for each sample (cluster 1,2,3,...) in columns and genes in rows. Zeroes, which are not involved in the clustering, are labeled as zero. See the `results` function for a convenient way to extract these results objects.

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

Examples

```
# load toy simulated example SingleCellExperiment object to find DD genes

data(scDatExSim)


# check that this object is a member of the SingleCellExperiment class
# and that it contains 200 samples and 30 genes

class(scDatExSim)
show(scDatExSim)


# set arguments to pass to scDD function
# we will perform 100 permutations on each of the 30 genes

prior_param=list(alpha=0.01, mu0=0, s0=0.01, a0=0.01, b0=0.01)
nperms <- 100


# call the scDD function to perform permutations, classify DD genes,
# and return results
# we won't perform the test for a difference in the proportion of zeroes
# since none exists in this simulated toy example data
# this step will take significantly longer with more genes and/or
# more permutations

scDatExSim <- scDD(scDatExSim, prior_param=prior_param, permutations=nperms,
            testZeroes=FALSE)
```

sideHist *sideHist*

Description

Plots two histograms side by side with smoothed density overlay

Usage

```
sideHist(x, y, logT = TRUE, title.gene = "")
```

Arguments

x	First numeric vector of data to plot.
y	Second numeric vector of data to plot.
logT	Logical that indicates whether to take the log(x+1) transformation.
title.gene	Character vector that contains the gene name that you are plotting

Value

NULL (creates a baseR plot)

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

sideViolin *sideViolin*

Description

Plots two histograms side by side with smoothed density overlay

Usage

```
sideViolin(y, cond, MAP = NULL, logT = TRUE, title.gene = "",
  conditionLabels = unique(cond), axes.titles = TRUE)
```

Arguments

y	Numeric vector of data to plot.
cond	Vector of condition labels corresponding to elements of x.
MAP	List of MAP partition estimates with conditions as list items and samples as elements (integer indicating which cluster each observation belongs to; zeroes belong to cluster 1)
logT	Logical that indicates whether to take the log(x+1) transformation.
title.gene	Character vector that contains the gene name that you are plotting.
conditionLabels	Character vector containing the names of the two conditions.
axes.titles	Logical indicating whether or not to include axes labels on plots.

Value

ggplot object

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

Examples

```
# load toy simulated example ExpressionSet to find DD genes

data(scDatExSim)


# load SingleCellExperiment package to facilitate subset operations

library(SingleCellExperiment)


# plot side by side violin plots for Gene 1 (DE)

sideViolin(normcounts(scDatExSim)[1,], scDatExSim$condition,
           title.gene=rownames(scDatExSim)[1])


# plot side by side violin plots for Gene 6 (DP)

sideViolin(normcounts(scDatExSim)[6,], scDatExSim$condition,
           title.gene=rownames(scDatExSim)[6])


# plot side by side violin plots for Gene 11 (DM)
```

```
sideViolin(normcounts(scDatExSim)[11,], scDatExSim$condition,
           title.gene=rownames(scDatExSim)[11])


# plot side by side violin plots for Gene 16 (DB)

sideViolin(normcounts(scDatExSim)[16,], scDatExSim$condition,
           title.gene=rownames(scDatExSim)[16])

# plot side by side violin plots for Gene 21 (EP)

sideViolin(normcounts(scDatExSim)[21,], scDatExSim$condition,
           title.gene=rownames(scDatExSim)[21])


# plot side by side violin plots for Gene 26 (EE)

sideViolin(normcounts(scDatExSim)[26,], scDatExSim$condition,
           title.gene=rownames(scDatExSim)[26])
```

`simuDB` *simuDB*

Description

Simulation for Differential "Both" Case - both Differential Modality and Differential Mean

Usage

```
simuDB(Dataset1, Simulated_Data, DEIndex, samplename, Zeropercent_Base, f, FC,
  coeff, RP, modeFC, DP, generateZero, constantZero, varInflation)
```

Arguments

`Dataset1`	Numeric matrix of expression values with genes in rows and samples in columns.
`Simulated_Data`	Required input empty matrix to provide structure information of output matrix with simulated data
`DEIndex`	Index for DE genes
`samplename`	The name for genes that chosen for simulation
`Zeropercent_Base`	Zero percentage for corresponding gene expression values
`f`	Fold change values (number of SDs) for each gene
`FC`	Fold Change values for DE Simulation
`coeff`	Relationship coefficients for Mean and Variance
`RP`	matrix for NB parameters for genes in samplename

modeFC	Vector of values to use for fold changes between modes for DP, DM, and DB.
DP	Differetial Proportion vector
generateZero	Specification of how to generate the zero values. If "empirical" (default), the observed proportion of zeroes in each gene is used for the simuated data, and the nonzeroes are simulated from a truncated negative binomial distribution. If "simulated", all values are simulated out of a negative binomial distribution, includling the zeroes. If "constant", then each gene has a fixed proportion of zeroes equal to constantZero.
constantZero	Numeric value between 0 and 1 that indicates the fixed proportion of zeroes for every gene. Ignored if generateZero method is not equal to "constant".
varInflation	Optional numeric vector with one element for each condition that corresponds to the multiplicative variance inflation factor to use when simulating data. Useful for sensitivity studies to assess the impact of confounding effects on differential variance across conditions. Currently assumes all samples within a condition are subject to the same variance inflation factor.

Value

Simulated_Data Simulated dataset for DB

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

simuDE *simuDE*

Description

Simulation for Classic Differentially Expressed Case.

Usage

```
simuDE(Dataset1, Simulated_Data, DEIndex, samplename, Zeropercent_Base, f, FC,
  coeff, RP, modeFC, generateZero, constantZero, varInflation)
```

Arguments

Dataset1	Numeric matrix of expression values with genes in rows and samples in columns.
Simulated_Data	Required input empty matrix to provide structure information of output matrix with simulated data
DEIndex	Index for DE genes
samplename	The name for genes that chosen for simulation

`Zeropercent_Base`
: Zero percentage for corresponding gene expression values

`f`
: Fold change values (number of SDs) for each gene

`FC`
: Fold Change values for DE Simulation

`coeff`
: Relationship coefficients for Mean and Variance

`RP`
: matrix for NB parameters for genes in samplename

`modeFC`
: Vector of values to use for fold changes between modes for DP, DM, and DB.

`generateZero`
: Specification of how to generate the zero values. If "`empirical`" (default), the observed proportion of zeroes in each gene is used for the simuated data, and the nonzeroes are simulated from a truncated negative binomial distribution. If "`simulated`", all values are simulated out of a negative binomial distribution, includling the zeroes. If "`constant`", then each gene has a fixed proportion of zeroes equal to `constantZero`.

`constantZero`
: Numeric value between 0 and 1 that indicates the fixed proportion of zeroes for every gene. Ignored if `generateZero` method is not equal to "`constant`".

`varInflation`
: Optional numeric vector with one element for each condition that corresponds to the multiplicative variance inflation factor to use when simulating data. Useful for sensitivity studies to assess the impact of confounding effects on differential variance across conditions. Currently assumes all samples within a condition are subject to the same variance inflation factor.

Details

Method called by main function `singleCellSimu` to simulate genes that have different means in each condition. Not intended to be called directly by user.

Value

Simulated_Data Simulated dataset for DE

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

`simuDM` *simuDM*

Description

Simulation for Differential Modalities Case

Usage

```
simuDM(Dataset1, Simulated_Data, DEIndex, samplename, Zeropercent_Base, f, FC,
  coeff, RP, modeFC, generateZero, constantZero, varInflation)
```

Arguments

Dataset1	Numeric matrix of expression values with genes in rows and samples in columns.
Simulated_Data	Required input empty matrix to provide structure information of output matrix with simulated data
DEIndex	Index for DE genes
samplename	The name for genes that chosen for simulation
Zeropercent_Base	Zero percentage for corresponding gene expression values
f	Fold change values (number of SDs) for each gene
FC	Fold Change values for DE Simulation
coeff	Relationship coefficients for Mean and Variance
RP	matrix for NB parameters for genes in samplename
modeFC	Vector of values to use for fold changes between modes for DP, DM, and DB.
generateZero	Specification of how to generate the zero values. If "empirical" (default), the observed proportion of zeroes in each gene is used for the simuated data, and the nonzeroes are simulated from a truncated negative binomial distribution. If "simulated", all values are simulated out of a negative binomial distribution, includling the zeroes. If "constant", then each gene has a fixed proportion of zeroes equal to constantZero.
constantZero	Numeric value between 0 and 1 that indicates the fixed proportion of zeroes for every gene. Ignored if generateZero method is not equal to "constant".
varInflation	Optional numeric vector with one element for each condition that corresponds to the multiplicative variance inflation factor to use when simulating data. Useful for sensitivity studies to assess the impact of confounding effects on differential variance across conditions. Currently assumes all samples within a condition are subject to the same variance inflation factor.

Value

Simulated_Data Simulated dataset for DM

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

simuDP *simuDP*

Description

Simulation for Differential Proportion Case

Usage

```
simuDP(Dataset1, Simulated_Data, DEIndex, samplename, Zeropercent_Base, f, FC,
  coeff, RP, modeFC, DP, generateZero, constantZero, varInflation)
```

Arguments

Dataset1	Numeric matrix of expression values with genes in rows and samples in columns.
Simulated_Data	Required input empty matrix to provide structure information of output matrix with simulated data
DEIndex	Index for DE genes
samplename	The name for genes that chosen for simulation
Zeropercent_Base	Zero percentage for corresponding gene expression values
f	Fold change values (number of SDs) for each gene
FC	Fold Change values for DE Simulation
coeff	Relationship coefficients for Mean and Variance
RP	matrix for NB parameters for genes in samplename
modeFC	Vector of values to use for fold changes between modes for DP, DM, and DB.
DP	Differetial Proportion vector
generateZero	Specification of how to generate the zero values. If "empirical" (default), the observed proportion of zeroes in each gene is used for the simuated data, and the nonzeroes are simulated from a truncated negative binomial distribution. If "simulated", all values are simulated out of a negative binomial distribution, includling the zeroes. If "constant", then each gene has a fixed proportion of zeroes equal to constantZero.
constantZero	Numeric value between 0 and 1 that indicates the fixed proportion of zeroes for every gene. Ignored if generateZero method is not equal to "constant".
varInflation	Optional numeric vector with one element for each condition that corresponds to the multiplicative variance inflation factor to use when simulating data. Useful for sensitivity studies to assess the impact of confounding effects on differential variance across conditions. Currently assumes all samples within a condition are subject to the same variance inflation factor.

Value

Simulated_Data Simulated dataset for DP

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

---

`simulateSet` *simulateSet*

Description

Simulation of a complete dataset, where the number of each type of differential distributions and equivalent distributions is specified.

Usage

```
simulateSet(SCdat, numSamples = 100, nDE = 250, nDP = 250, nDM = 250,
  nDB = 250, nEE = 5000, nEP = 4000, sd.range = c(1, 3), modeFC = c(2,
  3, 4), plots = TRUE, plot.file = NULL, random.seed = 284,
  varInflation = NULL, condition = "condition", param = bpparam())
```

Arguments

`SCdat`	An object of class `SingleCellExperiment` that contains normalized single-cell expression and metadata. The `assays` slot contains a named list of matrices, where the normalized counts are housed in the one named `normcounts`. This matrix should have one row for each gene and one sample for each column. The `colData` slot should contain a data.frame with one row per sample and columns that contain metadata for each sample. This data.frame should contain a variable that represents biological condition, which is in the form of numeric values (either 1 or 2) that indicates which condition each sample belongs to (in the same order as the columns of `normcounts`). Optional additional metadata about each cell can also be contained in this data.frame, and additional information about the experiment can be contained in the `metadata` slot as a list.
`numSamples`	numeric value for the number of samples in each condition to simulate
`nDE`	Number of DE genes to simulate
`nDP`	Number of DP genes to simulate
`nDM`	Number of DM genes to simulate
`nDB`	Number of DB genes to simulate
`nEE`	Number of EE genes to simulate
`nEP`	Number of EP genes to simulate
`sd.range`	Numeric vector of length two which describes the interval (lower, upper) of standard deviations of fold changes to randomly select.
`modeFC`	Vector of values to use for fold changes between modes for DP, DM, and DB.

plots	Logical indicating whether or not to generate fold change and validation plots
plot.file	Character containing the file string if the plots are to be sent to a pdf instead of to the standard output.
random.seed	Numeric value for a call to set.seed for reproducibility.
varInflation	Optional numeric vector with one element for each condition that corresponds to the multiplicative variance inflation factor to use when simulating data. Useful for sensitivity studies to assess the impact of confounding effects on differential variance across conditions. Currently assumes all samples within a condition are subject to the same variance inflation factor.
condition	A character object that contains the name of the column in colData that represents the biological group or condition of interest (e.g. treatment versus control). Note that this variable should only contain two possible values since scDD can currently only handle two-group comparisons. The default option assumes that there is a column named "condition" that contains this variable.
param	a MulticoreParam or SnowParam object of the BiocParallel package that defines a parallel backend. The default option is BiocParallel::bpparam() which will automatically creates a cluster appropriate for the operating system. Alternatively, the user can specify the number of cores they wish to use by first creating the corresponding MulticoreParam (for Linux-like OS) or SnowParam (for Windows) object, and then passing it into the scDD function. This could be done to specify a parallel backend on a Linux-like OS with, say 12 cores by setting param=BiocParallel::MulticoreParam(workers=12)

Value

An object of class SingleCellExperiment that contains simulated single-cell expression and metadata. The assays slot contains a named list of matrices, where the simulated counts are housed in the one named normcounts. This matrix should have one row for each gene (nDE + nDP + nDM + nDB + nEE + nEP rows) and one sample for each column (numSamples columns). The colData slot contains a data.frame with one row per sample and a column that represents biological condition, which is in the form of numeric values (either 1 or 2) that indicates which condition each sample belongs to (in the same order as the columns of normcounts). The rowData slot contains information about the category of the gene (EE, EP, DE, DM, DP, or DB), as well as the simulated foldchange value.

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

Examples

```
# Load toy example ExpressionSet to simulate from

data(scDatEx)
```

```
# check that this object is a member of the ExpressionSet class
# and that it contains 142 samples and 500 genes

class(scDatEx)
show(scDatEx)


# set arguments to pass to simulateSet function
# we will simuate 30 genes total; 5 genes of each type;
# and 100 samples in each of two conditions

nDE <- 5
nDP <- 5
nDM <- 5
nDB <- 5
nEE <- 5
nEP <- 5
numSamples <- 100
seed <- 816


# create simulated set with specified numbers of DE, DP, DM, DM, EE, and
# EP genes,
# specified number of samples, DE genes are 2 standard deviations apart, and
# multimodal genes have modal distance of 4 standard deviations

SD <- simulateSet(scDatEx, numSamples=numSamples, nDE=nDE, nDP=nDP, nDM=nDM,
                  nDB=nDB, nEE=nEE, nEP=nEP, sd.range=c(2,2), modeFC=4,
                  plots=FALSE,
                  random.seed=seed)
```

`singleCellSimu` *singleCellSimu*

Description

Called by `simulateSet` to simulate a specified number of genes from one DD category at a time.

Usage

```
singleCellSimu(Dataset1, Method, index, FC, modeFC, DP, Validation = FALSE,
  numGenes = 1000, numDE = 100, numSamples = 100,
  generateZero = c("empirical", "simulated", "constant"),
  constantZero = NULL, varInflation = NULL)
```

Arguments

`Dataset1`	Numeric matrix of expression values with genes in rows and samples in columns.

Method	Type of simulation should choose from "DE" "DP" "DM" "DB"
index	Reasonable set of genes for simulation
FC	Fold Change values for DE Simulation
modeFC	Vector of values to use for fold changes between modes for DP, DM, and DB.
DP	Differetial Proportion vector
Validation	Show Validation plots or not
numGenes	numeric value for the number of genes to simulate
numDE	numeric value for the number of genes that will differ between two conditions
numSamples	numeric value for the number of samples in each condition to simulate
generateZero	Specification of how to generate the zero values. If "empirical" (default), the observed proportion of zeroes in each gene is used for the simuated data, and the nonzeroes are simulated from a truncated negative binomial distribution. If "simulated", all values are simulated out of a negative binomial distribution, includling the zeroes. If "constant", then each gene has a fixed proportion of zeroes equal to constantZero.
constantZero	Numeric value between 0 and 1 that indicates the fixed proportion of zeroes for every gene. Ignored if generateZero method is not equal to "constant".
varInflation	Optional numeric vector with one element for each condition that corresponds to the multiplicative variance inflation factor to use when simulating data. Useful for sensitivity studies to assess the impact of confounding effects on differential variance across conditions. Currently assumes all samples within a condition are subject to the same variance inflation factor.

Value

Simulated_Data A list object where the first element contains a matrix of the simulated dataset, the second element contains the DEIndex, and the third element contains the fold change (between two conditions for DE, between two modes for DP, DM, and DB).

---

testKS *testKS*

Description

Function to perform KS test

Usage

```
testKS(dat, condition, inclZero = TRUE, numDE = NULL, DEIndex)
```

Arguments

dat	Matrix of single-cell RNA-seq data with genes in rows and samples in columns.
condition	Vector containing the indicator of which condition each sample (in the columns of dat) belongs to.
inclZero	Logical indicating whether to include zero in the test of different distributions
numDE	numeric value for the number of genes that will differ between two conditions
DEIndex	Vector containing the row numbers of the DE genes

Value

List object containing the significant gene indices, their adjusted p-values, and (if DE genes are supplied) the power and fdr.

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

Examples

```
# load toy simulated example ExpressionSet to find KS genes

data(scDatExSim)


# load SingleCellExperiment package to facilitate subset operations

library(SingleCellExperiment)


# check that this object is a member of the ExpressionSet class
# and that it contains 200 samples and 30 genes

class(scDatExSim)
show(scDatExSim)

# perform KS test and obtain adjusted p-values
RES_KS <- testKS(normcounts(scDatExSim), scDatExSim$condition, inclZero=FALSE,
                 numDE=20, DEIndex=1:20)
```

testZeroes *testZeroes*

Description

Test for a difference in the proportion of zeroes between conditions for a specified set of genes

Usage

```
testZeroes(dat, cond, these = 1:nrow(dat))
```

Arguments

dat	Matrix of single cell expression data with genes in rows and samples in columns.
cond	Vector of condition labels
these	vector of row numbers (gene numbers) to test for a difference in the proportion of zeroes.

Details

Test for a difference in the proportion of zeroes between conditions that is not explained by the detection rate. Utilizes Bayesian logistic regression.

Value

Vector of FDR adjusted p-values

validation *validation*

Description

Draw validation plots to show that the simulated dataset emulates characteristics of observed dataset.

Usage

```
validation(MV, DEIndex, Zeropercent_Base, Simulated_Data, numGenes)
```

Arguments

MV	Mean and Variance matrix for observed data
DEIndex	Index for genes chosen to be DE (can be NULL)
Zeropercent_Base	Zero percentage for corresponding gene expression values
Simulated_Data	Simulated dataset
numGenes	numeric value for the number of genes to simulate

Value

Validation plots

References

Korthauer KD, Chu LF, Newton MA, Li Y, Thomson J, Stewart R, Kendziorski C. A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. Genome Biology. 2016 Oct 25;17(1):222. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1077-y

Index

∗ **datasets**
    scDatEx, 20
    scDatExList, 21
    scDatExSim, 21

calcMV, 2
calcRP, 3
classifyDD, 4

feDP, 5
findFC, 6
findIndex, 7
findOutliers, 8

getPosteriorParams, 9

jointPosterior, 10

lu, 11
luOutlier, 11

mclustRestricted, *8–11*, 12

permMclust, 13
permMclustCov, 14
permMclustGene, 15
permZero, 16
preprocess, 17, *21*

results, 19

scDatEx, 20
scDatExList, 21
scDatExSim, 21
scDD, 22
sideHist, 26
sideViolin, 26
simuDB, 28
simuDE, 29
simuDM, 30
simuDP, 32
simulateSet, *20*, *21*, 33, *35*
singleCellSimu, *30*, 35

testKS, 36
testZeroes, 38

validation, 38