Package ‘SC3’

December 2, 2025

Type Package

Title Single-Cell Consensus Clustering

Version 1.38.0

Author Vladimir Kiselev

Maintainer Vladimir Kiselev <vladimir.yu.kiselev@gmail.com>

Description A tool for unsupervised clustering and analysis of single cell RNA-Seq data.

License GPL-3

Imports graphics, stats, utils, methods, e1071, parallel, foreach, doParallel, doRNG, shiny, ggplot2, pheatmap (>= 1.0.8), ROCR, robustbase, rrcov, cluster, WriteXLS, Rcpp (>= 0.11.1), SummarizedExperiment, SingleCellExperiment, BiocGenerics, S4Vectors

Depends R(>= 3.3)

LinkingTo Rcpp, RcppArmadillo

LazyData TRUE

RoxygenNote 6.0.1

Suggests knitr, rmarkdown, mclust, scater, BiocStyle

VignetteBuilder knitr

biocViews ImmunoOncology, SingleCell, Software, Classification, Clustering, DimensionReduction, SupportVectorMachine, RNASeq, Visualization, Transcriptomics, DataRepresentation, GUI, DifferentialExpression, Transcription

NeedsCompilation no

URL https://github.com/hemberg-lab/SC3

BugReports https://support.bioconductor.org/t/sc3/

git_url https://git.bioconductor.org/packages/SC3

git_branch RELEASE_3_22

git_last_commit a9abb4a

git_last_commit_date 2025-10-29

Repository Bioconductor 3.22

Date/Publication 2025-12-01

Contents

`ann` *Cell type annotations for data extracted from a publication by Yan et al.*

Description

Cell type annotations for data extracted from a publication by Yan et al.

Usage

```
ann
```

Format

An object of class `data.frame` with 90 rows and 1 columns.

Source

http://dx.doi.org/10.1038/nsmb.2660

Each row corresponds to a single cell from ‘yan‘ dataset

`calculate_distance` *Calculate a distance matrix*

Description

Distance between the cells, i.e. columns, in the input expression matrix are calculated using the Euclidean, Pearson and Spearman metrics to construct distance matrices.

Usage

```
calculate_distance(data, method)
```

Arguments

`data`	expression matrix
`method`	one of the distance metrics: 'spearman', 'pearson', 'euclidean'

Value

distance matrix

calculate_stability — *Calculate the stability index of the obtained clusters when changing* k

Description

Stability index shows how stable each cluster is accross the selected range of k. The stability index varies between 0 and 1, where 1 means that the same cluster appears in every solution for different k.

Usage

```
calculate_stability(consensus, k)
```

Arguments

consensus	consensus item of the sc3 slot of an object of 'SingleCellExperiment' class
k	number of clusters k

Details

Imagine a given cluster is split into N clusters when k is changed (all possible values of k are provided via ks argument in the main sc3 function). In each of the new clusters there are given_cells of the given cluster and also some extra_cells from other clusters. Then we define stability as follows:

$$\frac{1}{ks * N^2} \sum_{ks} \sum_{N} \frac{given\_cells}{given\_cells + extra\_cells}$$

Where one N corrects for the number of clusters and the other N is a penalty for splitting the cluster. ks corrects for the range of k.

Value

a numeric vector containing a stability index of each cluster

consensus_matrix — *Calculate consensus matrix*

Description

Consensus matrix is calculated using the Cluster-based Similarity Partitioning Algorithm (CSPA). For each clustering solution a binary similarity matrix is constructed from the corresponding cell labels: if two cells belong to the same cluster, their similarity is 1, otherwise the similarity is 0. A consensus matrix is calculated by averaging all similarity matrices.

Usage

```
consensus_matrix(clusts)
```

Arguments

`clusts`	a matrix containing clustering solutions in columns

Value

consensus matrix

`consmx` *Consensus matrix computation*

Description

Computes consensus matrix given cluster labels

Usage

```
consmx(dat)
```

Arguments

`dat`	a matrix containing clustering solutions in columns

`ED1` *Compute Euclidean distance matrix by rows*

Description

Used in consmx function

Usage

```
ED1(x)
```

Arguments

`x`	A numeric matrix.

`ED2` *Compute Euclidean distance matrix by columns*

Description

Used in sc3-funcs.R distance matrix calculation and within the consensus clustering.

Usage

```
ED2(x)
```

Arguments

`x`	A numeric matrix.

estkTW — *Estimate the optimal k for k-means clustering*

Description

The function finds the eigenvalues of the sample covariance matrix. It will then return the number of significant eigenvalues according to the Tracy-Widom test.

Usage

```
estkTW(dataset)
```

Arguments

`dataset`	processed input expression matrix.

Value

an estimated number of clusters k

get_auroc — *Calculate the area under the ROC curve for a given gene.*

Description

For a given gene a binary classifier is constructed based on the mean cluster expression values (these are calculated using the cell labels). The classifier prediction is then calculated using the gene expression ranks. The area under the receiver operating characteristic (ROC) curve is used to quantify the accuracy of the prediction. A `p-value` is assigned to each gene by using the Wilcoxon signed rank test.

Usage

```
get_auroc(gene, labels)
```

Arguments

`gene`	expression data of a given gene
`labels`	cell labels correspodning to the expression values of the gene

get_biolgy — *Wrapper for calculating biological properties*

Description

Wrapper for calculating biological properties

Usage

```
get_biolgy(dataset, labels, regime)
```

Arguments

`dataset`	expression matrix
`labels`	cell labels corresponding clusters
`regime`	defines what biological analysis to perform. "marker" for marker genes, "de" for differentiall expressed genes and "outl" for outlier cells

Value

results of either

get_de_genes — *Find differentially expressed genes*

Description

Differential expression is calculated using the non-parametric Kruskal-Wallis test. A significant `p-value` indicates that gene expression in at least one cluster stochastically dominates one other cluster. Note that the calculation of differential expression after clustering can introduce a bias in the distribution of `p-values`, and thus we advise to use the `p-values` for ranking the genes only.

Usage

```
get_de_genes(dataset, labels)
```

Arguments

`dataset`	expression matrix
`labels`	cell labels corresponding to the columns of the expression matrix

Value

a numeric vector containing the differentially expressed genes and correspoding p-values

Examples

```
d <- get_de_genes(yan[1:10,], as.numeric(ann[,1]))
head(d)
```

`get_marker_genes` *Calculate marker genes*

Description

Find marker genes in the dataset. The `get_auroc` is used to calculate marker values for each gene.

Usage

```
get_marker_genes(dataset, labels)
```

Arguments

`dataset`	expression matrix
`labels`	cell labels corresponding clusters

Value

data.frame containing the marker genes, corresponding cluster indexes and adjusted `p-values`

Examples

```
d <- get_marker_genes(yan[1:10,], as.numeric(ann[,1]))
d
```

`get_outl_cells` *Find cell outliers in each cluster.*

Description

Outlier cells in each cluster are detected using robust distances, calculated using the minimum covariance determinant (MCD), namely using `covMcd`. The outlier score shows how different a cell is from all other cells in the cluster and it is defined as the differences between the square root of the robust distance and the square root of the 99.99

Usage

```
get_outl_cells(dataset, labels)
```

Arguments

`dataset`	expression matrix
`labels`	cell labels corresponding to the columns of the expression matrix

Value

a numeric vector containing the cell labels and correspoding outlier scores ordered by the labels

Examples

```
d <- get_outl_cells(yan[1:10,], as.numeric(ann[,1]))
head(d)
```

---

get_processed_dataset *Get processed dataset used by* SC3 *clustering*

Description

Takes data from the `logcounts` slot, removes spike-ins and applies the gene filter.

Usage

```
get_processed_dataset(object)
```

Arguments

`object`	an object of `SingleCellExperiment` class

---

markers_for_heatmap *Reorder and subset gene markers for plotting on a heatmap*

Description

Reorders the rows of the input data.frame based on the `sc3_k_markers_clusts` column and also keeps only the top 10 genes for each value of `sc3_k_markers_clusts`.

Usage

```
markers_for_heatmap(markers)
```

Arguments

`markers`	a `data.frame` object with the following colnames: `sc3_k_markers_clusts`, `sc3_k_markers_auroc`, `sc3_k_markers_padj`.

---

norm_laplacian *Graph Laplacian calculation*

Description

Calculate graph Laplacian of a symmetrix matrix

Usage

```
norm_laplacian(A)
```

Arguments

`A`	symmetric matrix

organise_de_genes *Get differentiall expressed genes from an object of* `SingleCellExperiment` *class*

Description

This functions returns all marker gene columns from the `phenoData` slot of the input object corresponding to the number of clusters `k`. Additionally, it rearranges genes by the cluster index and order them by the area under the ROC curve value inside of each cluster.

Usage

```
organise_de_genes(object, k, p_val)
```

Arguments

`object`	an object of `SingleCellExperiment` class
`k`	number of cluster
`p_val`	p-value threshold

organise_marker_genes *Get marker genes from an object of* `SingleCellExperiment` *class*

Description

This functions returns all marker gene columns from the `phenoData` slot of the input object corresponding to the number of clusters `k`. Additionally, it rearranges genes by the cluster index and order them by the area under the ROC curve value inside of each cluster.

Usage

```
organise_marker_genes(object, k, p_val, auroc)
```

Arguments

`object`	an object of `SingleCellExperiment` class
`k`	number of cluster
`p_val`	p-value threshold
`auroc`	area under the ROC curve threshold

prepare_for_svm *A helper function for the SVM analysis*

Description

Defines train and study cell indeces based on the svm_num_cells and svm_train_inds input parameters

Usage

```
prepare_for_svm(N, svm_num_cells = NULL, svm_train_inds = NULL, svm_max)
```

Arguments

`N`	number of cells in the input dataset
`svm_num_cells`	number of random cells to be used for training
`svm_train_inds`	indeces of cells to be used for training
`svm_max`	define the maximum number of cells below which SVM is not run

Value

A list of indeces of the train and the study cells

reindex_clusters *Reindex cluster labels in ascending order*

Description

Given an `hclust` object and the number of clusters `k` this function reindex the clusters inferred by `cutree(hc, k)[hc$order]`, so that they appear in ascending order. This is particularly useful when plotting heatmaps in which the clusters should be numbered from left to right.

Usage

```
reindex_clusters(hc, k)
```

Arguments

`hc`	an object of class hclust
`k`	number of cluster to be inferred from hc

Examples

```
hc <- hclust(dist(USArrests), 'ave')
cutree(hc, 10)[hc$order]
reindex_clusters(hc, 10)[hc$order]
```

sc3 *Run all steps of* SC3 *in one go*

Description

This function is a wrapper that executes all steps of SC3 analysis in one go.

Usage

```
sc3.SingleCellExperiment(object, ks, gene_filter, pct_dropout_min,
  pct_dropout_max, d_region_min, d_region_max, svm_num_cells, svm_train_inds,
  svm_max, n_cores, kmeans_nstart, kmeans_iter_max, k_estimator, biology,
  rand_seed)

## S4 method for signature 'SingleCellExperiment'
sc3(object, ks = NULL, gene_filter = TRUE,
  pct_dropout_min = 10, pct_dropout_max = 90, d_region_min = 0.04,
  d_region_max = 0.07, svm_num_cells = NULL, svm_train_inds = NULL,
  svm_max = 5000, n_cores = NULL, kmeans_nstart = NULL,
  kmeans_iter_max = 1e+09, k_estimator = FALSE, biology = FALSE,
  rand_seed = 1)
```

Arguments

object	an object of SingleCellExperiment class.
ks	a range of the number of clusters k used for SC3 clustering. Can also be a single integer.
gene_filter	a boolen variable which defines whether to perform gene filtering before SC3 clustering.
pct_dropout_min	if gene_filter = TRUE, then genes with percent of dropouts smaller than pct_dropout_min are filtered out before clustering.
pct_dropout_max	if gene_filter = TRUE, then genes with percent of dropouts larger than pct_dropout_max are filtered out before clustering.
d_region_min	defines the minimum number of eigenvectors used for kmeans clustering as a fraction of the total number of cells. Default is 0.04. See SC3 paper for more details.
d_region_max	defines the maximum number of eigenvectors used for kmeans clustering as a fraction of the total number of cells. Default is 0.07. See SC3 paper for more details.
svm_num_cells	number of randomly selected training cells to be used for SVM prediction. The default is NULL.
svm_train_inds	a numeric vector defining indeces of training cells that should be used for SVM training. The default is NULL.
svm_max	define the maximum number of cells below which SVM is not run.
n_cores	defines the number of cores to be used on the user's machine. If not set, 'SC3' will use all but one cores of your machine.

kmeans_nstart
: nstart parameter passed to kmeans function. Can be set manually. By default it is 1000 for up to 2000 cells and 50 for more than 2000 cells.

kmeans_iter_max
: iter.max parameter passed to kmeans function.

k_estimator
: boolean parameter, defines whether to estimate an optimal number of clusters k. If user has already defined the ks parameter the estimation does not affect the user's paramater.

biology
: boolean parameter, defines whether to compute differentially expressed genes, marker genes and cell outliers.

rand_seed
: sets the seed of the random number generator. SC3 is a stochastic method, so setting the rand_seed to a fixed values can be used for reproducibility purposes.

Value

an object of SingleCellExperiment class

---

sc3_calc_biology — *Calculate DE genes, marker genes and cell outliers.*

Description

This function calculates differentially expressed (DE) genes, marker genes and cell outliers based on the consensus SC3 clusterings.

Usage

```
sc3_calc_biology.SingleCellExperiment(object, ks, regime)

## S4 method for signature 'SingleCellExperiment'
sc3_calc_biology(object, ks = NULL,
  regime = NULL)
```

Arguments

object
: an object of SingleCellExperiment class

ks
: a continuous range of integers - the number of clusters k to be used for SC3 clustering. Can also be a single integer.

regime
: defines what biological analysis to perform. "marker" for marker genes, "de" for differentiall expressed genes and "outl" for outlier cells

Details

DE genes are calculated using get_de_genes. Results of the DE analysis are saved as new columns in the featureData slot of the input object. The column names correspond to the adjusted p-values of the genes and have the following format: sc3_k_de_padj, where k is the number of clusters.

Marker genes are calculated using get_marker_genes. Results of the marker gene analysis are saved as three new columns (for each k) to the featureData slot of the input object. The column names correspond to the SC3 cluster labels, to the adjusted p-values of the genes and to the area under the ROC curve and have the following format: sc3_k_markers_clusts, sc3_k_markers_padj and sc3_k_markers_auroc, where k is the number of clusters.

Outlier cells are calculated using get_outl_cells. Results of the cell outlier analysis are saved as new columns in the phenoData slot of the input object. The column names correspond to the log2(outlier_score) and have the following format: sc3_k_log2_outlier_score, where k is the number of clusters.

Additionally, biology item is added to the sc3 slot and is set to TRUE indicating that the biological analysis of the dataset has been performed.

Value

an object of SingleCellExperiment class

---

sc3_calc_consens *Calculate consensus matrix.*

Description

This function calculates consensus matrices based on the clustering solutions contained in the kmeans item of the sc3 slot of the metadata(object). It then creates and populates the consensus item of the sc3 slot with consensus matrices, their hierarchical clusterings in hclust objects, and Silhouette indeces of the clusters. It also removes the previously calculated kmeans clusterings from the sc3 slot, as they are not needed for further analysis.

Usage

```
sc3_calc_consens.SingleCellExperiment(object)

## S4 method for signature 'SingleCellExperiment'
sc3_calc_consens(object)
```

Arguments

object	an object of SingleCellExperiment class

Details

Additionally, it also adds new columns to the colData slot of the input object. The column names correspond to the consensus cell labels and have the following format: sc3_k_clusters, where k is the number of clusters.

Value

an object of SingleCellExperiment class

`sc3_calc_dists` *Calculate distances between the cells.*

Description

This function calculates distances between the cells. It creates and populates the following items of the `sc3` slot of the `metadata(object)`:

- `distances` - contains a list of distance matrices corresponding to Euclidean, Pearson and Spearman distances.

Usage

```
sc3_calc_dists.SingleCellExperiment(object)

## S4 method for signature 'SingleCellExperiment'
sc3_calc_dists(object)
```

Arguments

`object`	an object of `SingleCellExperiment` class

Value

an object of `SingleCellExperiment` class

`sc3_calc_transfs` *Calculate transformations of the distance matrices.*

Description

This function transforms all `distances` items of the `sc3` slot of the `metadata(object)` using either principal component analysis (PCA) or by calculating the eigenvectors of the associated graph Laplacian. The columns of the resulting matrices are then sorted in descending order by their corresponding eigenvalues. The first `d` columns (where `d = max(metadata(object)$sc3$n_dim)`) of each transformation are then written to the `transformations` item of the `sc3` slot. Additionally, this function also removes the previously calculated `distances` from the `sc3` slot, as they are not needed for further analysis.

Usage

```
sc3_calc_transfs.SingleCellExperiment(object)

## S4 method for signature 'SingleCellExperiment'
sc3_calc_transfs(object)
```

Arguments

`object`	an object of `SingleCellExperiment` class

Value

an object of `SingleCellExperiment` class

`sc3_estimate_k` *Estimate the optimal number of cluster* `k` *for a scRNA-Seq expression matrix*

Description

Uses Tracy-Widom theory on random matrices to estimate the optimal number of clusters `k`. It creates and populates the `k_estimation` item of the `sc3` slot of the `metadata(object)`.

Usage

```
sc3_estimate_k.SingleCellExperiment(object)

## S4 method for signature 'SingleCellExperiment'
sc3_estimate_k(object)
```

Arguments

`object`	an object of `SingleCellExperiment` class

Value

an estimated value of k

`sc3_export_results_xls` *Write* `SC3` *results to Excel file*

Description

This function writes all `SC3` results to an excel file.

Usage

```
sc3_export_results_xls.SingleCellExperiment(object, filename)

## S4 method for signature 'SingleCellExperiment'
sc3_export_results_xls(object,
  filename = "sc3_results.xls")
```

Arguments

`object`	an object of `SingleCellExperiment` class
`filename`	name of the excel file, to which the results will be written

sc3_interactive — *Opens* SC3 *results in an interactive session in a web browser.*

Description

Runs interactive shiny session of SC3 based on precomputed clusterings.

Usage

```
sc3_interactive.SingleCellExperiment(object)

## S4 method for signature 'SingleCellExperiment'
sc3_interactive(object)
```

Arguments

object	an object of SingleCellExperiment class

Value

Opens a browser window with an interactive shiny app and visualize all precomputed clusterings.

sc3_kmeans — kmeans *clustering of cells.*

Description

This function performs kmeans clustering of the matrices contained in the transformations item of the sc3 slot of the metadata(object). It then creates and populates the following items of the sc3 slot:

- kmeans - contains a list of kmeans clusterings.

Usage

```
sc3_kmeans.SingleCellExperiment(object, ks)

## S4 method for signature 'SingleCellExperiment'
sc3_kmeans(object, ks = NULL)
```

Arguments

object	an object of SingleCellExperiment class
ks	a continuous range of integers - the number of clusters k to be used for SC3 clustering. Can also be a single integer.

Value

an object of SingleCellExperiment class

sc3_plot_cluster_stability

Plot stability of the clusters

Description

Stability index shows how stable each cluster is accross the selected range of ks. The stability index varies between 0 and 1, where 1 means that the same cluster appears in every solution for different k.

Usage

```
sc3_plot_cluster_stability.SingleCellExperiment(object, k)

## S4 method for signature 'SingleCellExperiment'
sc3_plot_cluster_stability(object, k)
```

Arguments

`object`	an object of 'SingleCellExperiment' class
`k`	number of clusters

sc3_plot_consensus

Plot consensus matrix as a heatmap

Description

The consensus matrix is a NxN matrix, where N is the number of cells. It represents similarity between the cells based on the averaging of clustering results from all combinations of clustering parameters. Similarity 0 (blue) means that the two cells are always assigned to different clusters. In contrast, similarity 1 (red) means that the two cells are always assigned to the same cluster. The consensus matrix is clustered by hierarchical clustering and has a diagonal-block structure. Intuitively, the perfect clustering is achieved when all diagonal blocks are completely red and all off-diagonal elements are completely blue.

Usage

```
sc3_plot_consensus.SingleCellExperiment(object, k, show_pdata)

## S4 method for signature 'SingleCellExperiment'
sc3_plot_consensus(object, k,
  show_pdata = NULL)
```

Arguments

`object`	an object of 'SingleCellExperiment' class
`k`	number of clusters
`show_pdata`	a vector of colnames of the pData(object) table. Default is NULL. If not NULL will add pData annotations to the columns of the output matrix

---

`sc3_plot_de_genes` *Plot expression of DE genes of the clusters identified by* `SC3` *as a heatmap*

Description

`SC3` plots gene expression profiles of the 50 genes with the lowest p-values.

Usage

```
sc3_plot_de_genes.SingleCellExperiment(object, k, p.val, show_pdata)

## S4 method for signature 'SingleCellExperiment'
sc3_plot_de_genes(object, k, p.val = 0.01,
  show_pdata = NULL)
```

Arguments

`object`	an object of 'SingleCellExperiment' class
`k`	number of clusters
`p.val`	significance threshold used for the DE genes
`show_pdata`	a vector of colnames of the pData(object) table. Default is NULL. If not NULL will add pData annotations to the columns of the output matrix

---

`sc3_plot_expression` *Plot expression matrix used for SC3 clustering as a heatmap*

Description

The expression panel represents the original input expression matrix (cells in columns and genes in rows) after the gene filter. Genes are clustered by kmeans with k = 100 (dendrogram on the left) and the heatmap represents the expression levels of the gene cluster centers after log2-scaling.

Usage

```
sc3_plot_expression.SingleCellExperiment(object, k, show_pdata)

## S4 method for signature 'SingleCellExperiment'
sc3_plot_expression(object, k,
  show_pdata = NULL)
```

Arguments

`object`	an object of 'SingleCellExperiment' class
`k`	number of clusters
`show_pdata`	a vector of colnames of the pData(object) table. Default is NULL. If not NULL will add pData annotations to the columns of the output matrix

---

`sc3_plot_markers` *Plot expression of marker genes identified by* `SC3` *as a heatmap.*

Description

By default the genes with the area under the ROC curve (AUROC) > 0.85 and with the p-value < 0.01 are selected and the top 10 marker genes of each cluster are visualized in this heatmap.

Usage

```
sc3_plot_markers.SingleCellExperiment(object, k, auroc, p.val, show_pdata)

## S4 method for signature 'SingleCellExperiment'
sc3_plot_markers(object, k, auroc = 0.85,
  p.val = 0.01, show_pdata = NULL)
```

Arguments

`object`	an object of 'SingleCellExperiment' class
`k`	number of clusters
`auroc`	area under the ROC curve
`p.val`	significance threshold used for the DE genes
`show_pdata`	a vector of colnames of the pData(object) table. Default is NULL. If not NULL will add pData annotations to the columns of the output matrix

---

`sc3_plot_silhouette` *Plot silhouette indexes of the cells*

Description

A silhouette is a quantitative measure of the diagonality of the consensus matrix. An average silhouette width (shown at the bottom left of the silhouette plot) varies from 0 to 1, where 1 represents a perfectly block-diagonal consensus matrix and 0 represents a situation where there is no block-diagonal structure. The best clustering is achieved when the average silhouette width is close to 1.

Usage

```
sc3_plot_silhouette.SingleCellExperiment(object, k)

## S4 method for signature 'SingleCellExperiment'
sc3_plot_silhouette(object, k)
```

Arguments

`object`	an object of 'SingleCellExperiment' class
`k`	number of clusters

sc3_prepare *Prepare the* `SingleCellExperiment` *object for* `SC3` *clustering.*

Description

This function prepares an object of `SingleCellExperiment` class for `SC3` clustering. It creates and populates the following items of the `sc3` slot of the `metadata(object)`:

- `kmeans_iter_max` - the same as the `kmeans_iter_max` argument.
- `kmeans_nstart` - the same as the `kmeans_nstart` argument.
- `n_dim` - contains numbers of the number of eigenvectors to be used in `kmeans` clustering.
- `rand_seed` - the same as the `rand_seed` argument.
- `svm_train_inds` - if SVM is used this item contains indexes of the training cells to be used for SC3 clustering and further SVM prediction.
- `svm_study_inds` - if SVM is used this item contains indexes of the cells to be predicted by SVM.
- `n_cores` - the same as the `n_cores` argument.

Usage

```
sc3_prepare.SingleCellExperiment(object, gene_filter, pct_dropout_min,
  pct_dropout_max, d_region_min, d_region_max, svm_num_cells, svm_train_inds,
  svm_max, n_cores, kmeans_nstart, kmeans_iter_max, rand_seed)

## S4 method for signature 'SingleCellExperiment'
sc3_prepare(object, gene_filter = TRUE,
  pct_dropout_min = 10, pct_dropout_max = 90, d_region_min = 0.04,
  d_region_max = 0.07, svm_num_cells = NULL, svm_train_inds = NULL,
  svm_max = 5000, n_cores = NULL, kmeans_nstart = NULL,
  kmeans_iter_max = 1e+09, rand_seed = 1)
```

Arguments

`object`	an object of `SingleCellExperiment` class.
`gene_filter`	a boolen variable which defines whether to perform gene filtering before SC3 clustering.
`pct_dropout_min`	if `gene_filter` = `TRUE`, then genes with percent of dropouts smaller than `pct_dropout_min` are filtered out before clustering.
`pct_dropout_max`	if `gene_filter` = `TRUE`, then genes with percent of dropouts larger than `pct_dropout_max` are filtered out before clustering.
`d_region_min`	defines the minimum number of eigenvectors used for kmeans clustering as a fraction of the total number of cells. Default is `0.04`. See `SC3` paper for more details.
`d_region_max`	defines the maximum number of eigenvectors used for kmeans clustering as a fraction of the total number of cells. Default is `0.07`. See `SC3` paper for more details.

svm_num_cells	number of randomly selected training cells to be used for SVM prediction. The default is NULL.
svm_train_inds	a numeric vector defining indeces of training cells that should be used for SVM training. The default is NULL.
svm_max	define the maximum number of cells below which SVM is not run.
n_cores	defines the number of cores to be used on the user's machine. If not set, 'SC3' will use all but one cores of your machine.
kmeans_nstart	nstart parameter passed to kmeans function. Default is 1000 for up to 2000 cells and 50 for more than 2000 cells.
kmeans_iter_max	iter.max parameter passed to kmeans function. Default is 1e+09.
rand_seed	sets the seed of the random number generator. SC3 is a stochastic method, so setting the rand_seed to a fixed values can be used for reproducibility purposes.

Value

an object of SingleCellExperiment class

---

sc3_run_svm *Run the hybrid* SVM *approach.*

Description

This method parallelize SVM prediction for each k (the number of clusters). Namely, for each k, support_vector_machines function is utilized to predict the labels of study cells. Training cells are selected using svm_train_inds item of the sc3 slot of the metadata(object).

Usage

```
sc3_run_svm.SingleCellExperiment(object, ks)

## S4 method for signature 'SingleCellExperiment'
sc3_run_svm(object, ks = NULL)
```

Arguments

object	an object of SingleCellExperiment class
ks	a continuous range of integers - the number of clusters k to be used for SC3 clustering. Can also be a single integer.

Details

Results are written to the sc3_k_clusters columns to the colData slot of the input object, where k is the number of clusters.

Value

an object of SingleCellExperiment class

support_vector_machines *Run support vector machines (SVM) prediction*

Description

Train an SVM classifier on a training dataset (train) and then classify a study dataset (study) using the classifier.

Usage

```
support_vector_machines(train, study, kern)
```

Arguments

train	training dataset with colnames, corresponding to training labels
study	study dataset
kern	kernel to be used with SVM

Value

classification of the study dataset

tmult *Matrix left-multiplied by its transpose*

Description

Given matrix A, the procedure returns A'A.

Usage

```
tmult(x)
```

Arguments

x	Numeric matrix.

`transformation` *Distance matrix transformation*

Description

All distance matrices are transformed using either principal component analysis (PCA) or by calculating the eigenvectors of the graph Laplacian (Spectral). The columns of the resulting matrices are then sorted in descending order by their corresponding eigenvalues.

Usage

```
transformation(dists, method)
```

Arguments

`dists`	distance matrix
`method`	transformation method: either 'pca' or 'laplacian'

Value

transformed distance matrix

`yan` *Single cell RNA-Seq data extracted from a publication by Yan et al.*

Description

Single cell RNA-Seq data extracted from a publication by Yan et al.

Usage

```
yan
```

Format

An object of class `data.frame` with 20214 rows and 90 columns.

Source

http://dx.doi.org/10.1038/nsmb.2660

Columns represent cells, rows represent genes expression values.

Index