Package 'scCAN'

October 14, 2022

Type Package

Title Single-Cell Clustering using Autoencoder and Network Fusion

Version 1.0.4

Maintainer Bang Tran <bang.t.s@nevada.unr.edu>

Description A single-cell Clustering method using 'Autoencoder' and Network fusion ('scCAN') for segregating the cells from the high-dimensional 'scRNA-Seq' data. The software automatically determines the optimal number of clusters and then partitions the cells in a way such that the results are robust to noise and dropouts. 'scCAN' is fast and it supports Windows, Linux, and Mac OS.

License LGPL

Encoding UTF-8

LazyData true

LazyDataCompression xz

Depends R (>= 3.5.0), scDHA, FNN, purrr

Imports stats

RoxygenNote 7.1.2

Suggests knitr

VignetteBuilder knitr

NeedsCompilation no

Author Bang Tran [aut, cre],
Duc Tran [aut],
Hung Nguyen [aut],
Tin Nguyen [fnd]

Repository CRAN

Date/Publication 2022-04-06 00:10:02 UTC

R topics documented:

`adjustedRandIndex` *adjustedRandIndex*

Description

The function to calculate adjusted Rand index value with the inputs of true clusters and predicted clusters

Usage

```
adjustedRandIndex(x, y)
```

Arguments

`x`	A vector that contain predicted cluster assignment.
`y`	A vector that contain true cluster assignment.

`calculate_celltype_prob` *calculate_celltype_prob*

Description

Calculate clusters and cell types similarity based on the markers.

Usage

```
calculate_celltype_prob(clt_marker_list, marker_database_list, type = "jacc")
```

Arguments

`clt_marker_list`	A list of markers for all cluster.
`marker_database_list`	A list of markers of all reference cell types.
`type`	A parameter to select the method to measure cluster and cell type similarity • jacc - Jaccard index. • ac - Accuracy. • f1 - F1 score. .

Value

A confusion matrix between clusters and cell types. Each cell represents a probabilty of a cluster belongs to a cell type.

`curate_markers` *curate_markers*

Description

Filter genes that have low p-value and fold-change.

Usage

```
curate_markers(
  whole_list,
  gene_names,
  wilcox_threshold = 0.001,
  logfc_threshold = 1.5
)
```

Arguments

`whole_list` A list of markers for all clusters.

`gene_names` All the gene names of the expression matrix.

`wilcox_threshold`
A threshold for p-value `wilcox_threshold = 0.001` by default.

`logfc_threshold`
A threshold for fold-change `logfc_threshold = 1.5` by default.

Value

A list of markers that are strong expressed for discovered clusters.

`find_markers` *find_markers*

Description

Perform cluster-wise Wilcox test and fold-change for each gene.

Usage

```
find_markers(input_data_matrix, cluster_labels, identity = 1, threads = 8)
```

Arguments

`input_data_matrix`
An expression matrix in which rows are genes and columns are cells.

`cluster_labels` A vector of cluster labels obtained from clustering methods.

`identity` A parameter to select specific cluster `identity = 1` by default.

`threads` A parameter to control number of cores used for analysis `threads = 1` by default.

Value

A list that contains p-value and fold-change ratio for all genes of each cluster.

`find_specific_marker` *find_specific_marker*

Description

Calculate cluster and cell type similarity based on the markers.

Usage

```
find_specific_marker(gene_name, f_list, type = "jacc")
```

Arguments

`gene_name` A list of markers belong to the cluster.

`f_list` A list of markers belongs to a reference cell type.

`type` A parameter to select the method to measure cluster and cell type similarity

- jacc - Jaccard index.
- ac - Accuracy.
- f1 - F1 score.

.

Value

A vector of probabilties of a cluster belongs to cell types.

`get_cluster_markers` *get_cluster_markers*

Description

Find markers for each cluster

Usage

```
get_cluster_markers(input_data_matrix, labels_vector, threads = 1)
```

Arguments

`input_data_matrix`
An expression matrix in which rows are genes and columns are cells.

`labels_vector` A vector of cluster labels ontained from clustering methods.

`threads` A parameter to control number of cores used for analysis `threads = 1` by default.

Value

A list that contains markers for each cluster.

`scCAN` *scCAN*

Description

This is the main function to perform sc-RNA seq data clustering clustering. scCAN is fully unsupervised scRNA-seq clustering framework that uses deep neural network and network fusion-based clustering algorithm. First, scCAN applies a non-negative autoencoder to filter scRNA-seq data. Second, the filtered data is passed to stacked Bayesian autoencoder to get multiple low-dimensional representations of input data. Subsequently, scCAN converts these compressed data into networks and unify those networks to a single graph. Then, scCAN uses a spectral clustering algorithm to obtain final clusters assignment.

Usage

```
scCAN(
  data,
  sparse = FALSE,
  n.neighbors = 30,
  alpha = 0.5,
  n.iters = 10,
  ncores = 10,
  r.seed = 1,
  subsamp = T,
  k = 2:15,
  samp.size = 5000
)
```

Arguments

`data` Gene expression matrix, with rows represent samples and columns represent genes.

`sparse` Boolen variable indicating whether data is a sparse matrix. The input must be a non negative sparse matrix.

`n.neighbors` Number of neighboring cells that are used to caculate the edge's weight. The number of neighbors are set `n.neighbors = 30` by default.

`alpha` A hyper parameter that control the weight of graph. This values is set to `alpha = 0.5` by default.

`n.iters` A hyper-parameter to set the number of network fusion iterations. It is set to `n.iters = 10` by default.

`ncores` Number of processor cores to use.

`r.seed` A parameter to set a seed for reproducibility. This values is set to `r.seed = 1` by default.

`subsamp` Enable subsampling process for big data. This values is set to `subsamp = T` by default.

`k` A vector to search for optimal number of cluster.

`samp.size` A parameter to control number of sub-sampled cells.

Value

List with the following keys:

- cluster - A numeric vector containing cluster assignment for each sample.
- k - The optimal number of cluster.
- latent - The latent data generated from autoencoders.

References

1. Duc Tran, Hung Nguyen, Bang Tran, Carlo La Vecchia, Hung N. Luu, Tin Nguyen (2021). Fast and precise single-cell data analysis using a hierarchical autoencoder. Nature Communications, 12, 1029. doi: 10.1038/s41467-021-21312-2

Examples

```
## Not run:
# Load the package and the example data (SCE dataset)
library(scCAN)
#Load example data
data("SCE")
```

```
#Get data matrix and label
data <- t(SCE$data); label <- as.character(SCE$cell_type1)

#Generate clustering result, the input matrix has rows as samples and columns as genes
result <- scCAN(data, r.seed = 1)

#Get the clustering result
cluster <- result$cluster

#Calculate adjusted Rand Index
ari <- round(scCAN::adjustedRandIndex(cluster,label), 2)
print(paste0("ARI = ", ari))


## End(Not run)
```

SCE *SCE*

Description

SCE dataset includes scRNA-seq data and cell type information.

Usage

```
SCE
```

Format

An object of class `list` of length 2.

Index