Package 'NeuCA'

March 26, 2024

Type Package

Title NEUral network-based single-Cell Annotation tool

Version 1.8.1

Date 2024-03-08

Description NeuCA is is a neural-network based method for scRNA-seq data annotation. It can automatically adjust its classification strategy depending on cell type correlations, to accurately annotate cell. NeuCA can automatically utilize the structure information of the cell types through a hierarchical tree to improve the annotation accuracy. It is especially helpful when the data contain closely correlated cell types.

License GPL-2

Depends R(>= 3.5.0), keras, limma, e1071, SingleCellExperiment, kableExtra

Suggests BiocStyle, knitr, rmarkdown, networkD3

Collate functions_NN.R functions_mhNN.R utils.R

biocViews SingleCell, Software, Classification, NeuralNetwork, RNASeq, Transcriptomics, DataRepresentation, Transcription, Sequencing, Preprocessing, GeneExpression, DataImport

Encoding UTF-8

LazyData false

VignetteBuilder knitr

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

git_branch RELEASE_3_18

git_last_commit 4eb0e42

git_last_commit_date 2024-03-08

Repository Bioconductor 3.18

Date/Publication 2024-03-25

Author Ziyi Li [aut],
Hao Feng [aut, cre]

Maintainer Hao Feng <hxf155@case.edu>

R topics documented:

Baron_counts   *Single-cell RNA-seq example dataset: Baron data*

Description

Baron_counts is a matrix of scRNA-seq data. Each row represents one gene. Each column represents one cell. Baron_true_cell_label is a vector of the same length as the columns of the matrix, containing the true cell labels for each cell in the same order.

Usage

```
data(Baron_scRNA)
```

Format

Baron_counts is a matrix of gene expression values. Baron_true_cell_label is a vector of true cell labels for each cell.

Value

A matrix of gene expression values, and a vector of true cell labels for each cell.

Examples

```
data(Baron_scRNA)
dim(Baron_counts)
Baron_counts[1:5,1:5]
length(Baron_true_cell_label)
head(Baron_true_cell_label)
```

| NeuCA | *A NEUral-network based Cell Annotation (NeuCA) tool for cell type annotation using single-cell RNA-seq data.* |

Description

NeuCA is a supervised cell label assignment method that uses existing scRNA-seq data with known labels to train a neural network-based classifier, and then predict cell labels in data of interest.

Usage

```
NeuCA(train, test, model.size = "big", verbose = FALSE)
```

Arguments

train	A training scRNA-seq dataset where cell labels are already known. Must be an object of SingleCellExperiment class. Must contain cell labels as the first column in its colData.
test	A testing scRNA-seq dataset where cell labels are unknown. Must be an object of SingleCellExperiment class.
model.size	an ordinal variable indicating the complexity of the neural-network. Must be one of the following: "big", "medium" or "small"
verbose	A Boolean variable (TRUE/FALSE) indicating whether additional information about the training and testing process will be printed.

Details

When closely correlated cell types exist, NeuCA uses the cell type tree information through a hierarchical structure of neural networks to improve annotation accuracy. Feature selection is performed in hierarchical structure to further improve classification accuracy. When cell type correlations are not high, a feed-forward neural network is adopted.

Value

NeuCA returns a vector of predicted cell types. The output vector has the same length with the number of cells in the testing dataset.

Note

The input single-cell RNA-seq data, for both training and testing, should be objects of SingleCellExperiment class. The true cell type labels, for the training dataset, should be stored as the first column in its SingleCellExperiment "colData"" object.

Author(s)

Hao Feng <hxf155@case.edu>

Examples

```
#1. Load in example scRNA-seq data
#Baron_scRNA is the training scRNA-seq dataset
#Seg_scRNA is the testing scRNA-seq dataset
data("Baron_scRNA")
data("Seg_scRNA")

#2. Create SingleCellExperiment object as the input for NeuCA (if data are not already in SingleCellExperiment forma
Baron_anno = data.frame(Baron_true_cell_label, row.names = colnames(Baron_counts))
Baron_sce = SingleCellExperiment(
    assays = list(normcounts = as.matrix(Baron_counts)),
    colData = Baron_anno
    )
# use gene names as feature symbols
rowData(Baron_sce)$feature_symbol <- rownames(Baron_sce)
# remove features with duplicated names
Baron_sce <- Baron_sce[!duplicated(rownames(Baron_sce)), ]

#similarly for Seg data
Seg_anno = data.frame(Seg_true_cell_label, row.names = colnames(Seg_counts))
Seg_sce <- SingleCellExperiment(
    assays = list(normcounts = as.matrix(Seg_counts)),
    colData = Seg_anno
)
# use gene names as feature symbols
rowData(Seg_sce)$feature_symbol <- rownames(Seg_sce)
# remove features with duplicated names
Seg_sce <- Seg_sce[!duplicated(rownames(Seg_sce)), ]


#3. NeuCA training and cell type prediction
predicted.label = NeuCA(train = Baron_sce, test = Seg_sce, model.size = "big", verbose = FALSE)
head(predicted.label)
#Seg_sce have its ground true cell type stored, compare the predicted vs. the truth.
sum(predicted.label==colData(Seg_sce)[,1])/length(predicted.label)
```

Seg_counts   *Single-cell RNA-seq example dataset: Seg data*

Description

Seg_counts is a matrix of scRNA-seq data. Each row represents one gene. Each column represents one cell. Seg_true_cell_label is a vector of the same length as the columns of the matrix, containing the true cell labels for each cell in the same order.

Usage

```
data(Seg_scRNA)
```

Format

Seg_counts is a matrix of gene expression values. Seg_true_cell_label is a vector of true cell labels for each cell.

Value

A matrix of gene expression values, and a vector of true cell labels for each cell.

Examples

```
data(Seg_scRNA)
dim(Seg_counts)
Seg_counts[1:5,1:5]
length(Seg_true_cell_label)
head(Seg_true_cell_label)
```

Index