Package 'scKWARN'

September 14, 2020

Type Package

Title Single-cell RNA Sequencing Normalization Using A Local Average Technique

Version 1.0.2

Date 2020-09-14

Author Hsu, Chih-Yuan

Maintainer Chih-Yuan Hsu <chih-yuan.hsu@vumc.org>

Description A normalization method for single-cell RNA sequencing data.

License GPL-2 | GPL-3

Imports Rcpp (>= 1.0.1), Matrix, stats, methods

LinkingTo Rcpp

LazyData true

RoxygenNote 6.1.1

Archs x64

R topics documented:

`LocASN` *Single-cell RNA sequencing normalization using a local average technique*

Description

A function of normalizing single cell RNA-seq gene expression.

Usage

```
LocASN(countmatrix, conditions = NULL, gene_num_gezero = 3,
  cell_num_gezero = 10, numforEst = 10)
```

Arguments

countmatrix	Input. Unnormalized count matrix (genes by cells).
conditions	Input (Optional). Indicate which cells are sampled from the same conditions. The default value, NULL, denotes all the cells are sampled from the same condition.
gene_num_gezero	Input (Optional). A threshold (interger) to determine the inclusion of a gene. The gene included needs to be expressed in at least *gene_num_gezero* cells. The default value is *gene_num_gezero* = 3.
cell_num_gezero	Input (Optional). A threshold (interger) to determine the inclusion of a cell. The cell included needs to contain at least *cell_num_gezero* expressed genes. The default value is *cell_num_gezero* = 10.
numforEst	Input (Optional). Use genes which expressed in at least *numforEst* (integer) cells to calculate the similarity between cells. The default value is *numforEst* = 10.

Value

NormalizedData	Matrix (genes by cells). Data matrix after normalization.
scalingFactor	Vector. Cell-specific scaling factors.
delete_genes	Vector. Indeice of the genes deleted.
delete_cells	Vector. Indeice of the cells deleted.

Examples

```
set.seed(12345)
G <- 2000; n <- 600 # G: number of genes, n: number of cells
NB_cell <- function(j) rnbinom(G, size = 0.1, mu = rgamma(G, shape = 2, rate = 2))
countsimdata <- sapply(1:n, NB_cell)
colnames(countsimdata) <- paste("cell", 1:n, sep = "_")
rownames(countsimdata) <- paste("gene", 1:G, sep = "_")
Result <- LocASN(countmatrix = countsimdata)
Result$NormalizedData[1:10,1:10]; Result$scalingFactor[1:10]

#conditions <- c(rep(1,n/2), rep(2,n/2))
#Result2 <- LocASN(countmatrix = countsimdata, conditions = conditions)
#Result2$NormalizedData[1:10,1:10]; Result2$scalingFactor[1:10]
```

Index