

Package ‘scTenifoldKnk’

October 14, 2022

Type Package

Title In-Silico Knockout Experiments from Single-Cell Gene Regulatory Networks

Version 1.0.1

Description A workflow based on 'scTenifoldNet' to perform in-silico knockout experiments using single-cell RNA sequencing (scRNA-seq) data from wild-type (WT) control samples as input. First, the package constructs a single-cell gene regulatory network (sc-GRN) and knocks out a target gene from the adjacency matrix of the WT scGRN by setting the gene's outdegree edges to zero. Then, it compares the knocked out sc-GRN with the WT scGRN to identify differentially regulated genes, called virtual-knockout perturbed genes, which are used to assess the impact of the gene knockout and reveal the gene's function in the analyzed cells.

URL <https://github.com/cailab-tamu/scTenifoldKnk>

BugReports <https://github.com/cailab-tamu/scTenifoldKnk/issues>

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

biocViews

Imports pbapply, RSpectra, Matrix, methods, stats, utils, MASS, scTenifoldNet

Suggests testthat (>= 2.1.0)

NeedsCompilation no

Author Daniel Osorio [aut, cre] (<<https://orcid.org/0000-0003-4424-8422>>),
Yan Zhong [aut, ctb],
Guanxun Li [aut, ctb],
Qian Xu [aut, ctb],
Andrew Hillhouse [aut, ctb],
Jingshu Chen [aut, ctb],
Laurie Davidson [aut, ctb],
Yanan Tian [aut, ctb],

Robert Chapkin [aut, ctb],
 Jianhua Huang [aut, ctb],
 James Cai [aut, ctb, ths] (<<https://orcid.org/0000-0002-8081-6725>>)

Maintainer Daniel Osorio <dcosorloh@tamu.edu>

Repository CRAN

Date/Publication 2021-01-22 20:00:02 UTC

R topics documented:

scTenifoldKnk	2
Index	4

scTenifoldKnk	<i>scTenifoldKNK</i>
---------------	----------------------

Description

Predict gene perturbations

Usage

```
scTenifoldKnk(
  countMatrix,
  gKO = NULL,
  qc_mtThreshold = 0.1,
  qc_minLSize = 1000,
  nc_lambda = 0,
  nc_nNet = 10,
  nc_nCells = 500,
  nc_nComp = 3,
  nc_scaleScores = TRUE,
  nc_symmetric = FALSE,
  nc_q = 0.9,
  td_K = 3,
  td_maxIter = 1000,
  td_maxError = 1e-05,
  td_nDecimal = 3,
  ma_nDim = 2
)
```

Arguments

countMatrix	countMatrix
gKO	gKO

qc_mtThreshold	A decimal value between 0 and 1. Defines the maximum ratio of mitochondrial reads (mitochondrial reads / library size) present in a cell to be included in the analysis. It's computed using the symbol genes starting with 'MT-' non-case sensitive.
qc_minLSize	An integer value. Defines the minimum library size required for a cell to be included in the analysis.
nc_lambda	A continuous value between 0 and 1. Defines the multiplicative value (1-lambda) to be applied over the weaker edge connecting two genes to maximize the adjacency matrix directionality.
nc_nNet	An integer value. The number of networks based on principal components regression to generate.
nc_nCells	An integer value. The number of cells to subsample each time to generate a network.
nc_nComp	An integer value. The number of principal components in PCA to generate the networks. Should be greater than 2 and lower than the total number of genes.
nc_scaleScores	A boolean value (TRUE/FALSE), if TRUE, the weights will be normalized such that the maximum absolute value is 1.
nc_symmetric	A boolean value (TRUE/FALSE), if TRUE, the weights matrix returned will be symmetric.
nc_q	A decimal value between 0 and 1. Defines the cut-off threshold of top q% relationships to be returned.
td_K	An integer value. Defines the number of rank-one tensors used to approximate the data using CANDECOMP/PARAFAC (CP) Tensor Decomposition.
td_maxIter	An integer value. Defines the maximum number of iterations if error stay above td_maxError.
td_maxError	A decimal value between 0 and 1. Defines the relative Frobenius norm error tolerance.
td_nDecimal	An integer value indicating the number of decimal places to be used.
ma_nDim	An integer value. Defines the number of dimensions of the low-dimensional feature space to be returned from the non-linear manifold alignment.

Author(s)

Daniel Osorio <dcosorih@tamu.edu>

Examples

```
# Loading single-cell data
scrNAseq <- system.file("single-cell/example.csv", package="scTenifoldKnk")
scrNAseq <- read.csv(scrNAseq, row.names = 1)

# Running scTenifoldKnk
scTenifoldKnk(countMatrix = scrNAseq, gKO = 'G100', qc_minLSize = 0)
```

Index

scTenifoldKnk, [2](#)