Package ‘scTensor’

February 3, 2024

**Type**  Package

**Title**  Detection of cell-cell interaction from single-cell RNA-seq dataset by tensor decomposition

**Version**  2.12.0

**Depends**  R (>= 4.1.0)

**Imports**  methods, RSQLite, igraph, S4Vectors, plotly, reactome.db, AnnotationDbi, SummarizedExperiment, SingleCellExperiment, nnTensor (>= 1.1.5), ccTensor (>= 1.0.2), rTensor (>= 1.4.8), abind, plotrix, heatmaply, tagcloud, rmarkdown, BiocStyle, knitr, AnnotationHub, MeSHDbi (>= 1.29.2), grDevices, graphics, stats, utils, outliers, Category, meshr (>= 1.99.1), GOstats, ReactomePA, DOSE, crayon, checkmate, BiocManager, visNetwork, schex, ggplot2

**Suggests**  testthat, LRBaseDbi, Seurat, scTGIF, Homo.sapiens

**Description**  The algorithm is based on the non-negative tucker decomposition (NTD2) of nnTensor.

**License**  Artistic-2.0

**biocViews**  DimensionReduction, SingleCell, Software, GeneExpression

**VignetteBuilder**  knitr

**git_url**  https://git.bioconductor.org/packages/scTensor

**git_branch**  RELEASE_3_18

**git_last_commit**  40a43b3

**git_last_commit_date**  2023-10-24

**Repository**  Bioconductor 3.18

**Date/Publication**  2024-02-02

**Author**  Koki Tsuyuzaki [aut, cre], Kozo Nishida [aut]

**Maintainer**  Koki Tsuyuzaki <k.t.the-answer@hotmail.co.jp>
scTensor-package

Detection of cell-cell interaction from single-cell RNA-seq dataset by tensor decomposition

Description

The algorithm is based on the non-negative tucker decomposition (NTD2) of nnTensor.

Details

The DESCRIPTION file: This package was not yet installed at build time.

Index: This package was not yet installed at build time.

Author(s)

Koki Tsuyuzaki [aut, cre], Kozo Nishida [aut]

Maintainer: Koki Tsuyuzaki <k.t.the-answer@hotmail.co.jp>

See Also

GermMale,labelGermMale,tsneGermMale,cellCellSetting,cellCellDecomp,cellCellReport

Examples

ls("package:scTensor")
CCSParams-class

---

Class "CCSParams"

Description

The parameter object to be specified against cellCellSimulate function.

Objects from the Class

Objects can be created by calls of the form new("CCSParams", ...).

Slots

- **nGene**: The number of genes.
- **nCell**: The number of cells.
- **cciInfo**: The parameter to describe the CCI.
- **lambda**: The parameter for dropout simulation.
- **seed**: The seed for using random numbers.

Methods

- **newCCSParams** Generator of CCSParams object.
- **getParam** Getter function of the slot in CCSParams object.
- **setParam<-** Setter function of the slot in CCSParams object.

See Also

newCCSParams, getParam, setParam<-  

---

cellCellDecomp

Performing scTensor

---

Description

All parameters is saved to metadata slot of SingleCellExperiment object.

Usage

cellCellDecomp(sce, algorithm=c("ntd2", "ntd", "nmf", "cx", "pearson",  
"spearman", "distance", "pearson.lr", "spearman.lr", "distance.lr",  
"pcomb", "label.permutation", "cabello.aguilar", "halpern"), ranks=c(3,3), rank=3, thr1=log2(5), thr2=25, thr3=0.95, L1_A=0, L2_A=0,  
centering=TRUE, mergeas=c("mean", "sum"), outerfunc=c("*", "+"),  
comb=c("random", "all"), num.sampling=100, num.perm=1000, assayNames = "counts", decomp=TRUE)
**Arguments**

- **sce**  
  The object generated by instantiation of `SingleCellExperiment`-class.

- **algorithm**  
  Algorithm for constructing cell-cell similarity matrix. "ntd2", "ntd", "nmf", "cx",  
  "pearson", "spearman", "distance", "pearson.lr", "spearman.lr", "distance.lr",  
  "pcomb" or "label.permutation" can be specified (Default: ntd2).

- **ranks**  
  The size of the core tensor decomposed by NTD. Each element means (Number  
  of Ligand-Cell Pattern, Number of Receptor-Cell Pattern, Number of LR-pairs  
  Pattern) (Default: c(3,3)).

- **rank**  
  The number of low dimension of NMF (Default: 3).

- **thr1**  
  The threshold used by pcomb (Default: log2(5)).

- **thr2**  
  The threshold used by cx (Default: 0.95).

- **L1_A**  
  The parameter to control the sparseness (Default: 0).

- **L2_A**  
  The parameter to control the outlier (Default: 0).

- **verbose**  
  The verbose parameter for `nntensor::NTD` (Default: FALSE).

- **centering**  
  When the value is TRUE, input matrix is summarized as celltype-level vectors  
  (Default: TRUE).

- **mergeas**  
  When the centering is TRUE, "sum" (celltype-level sum vector) or "mean" (celltype-  
  level average vector) is calculated (Default: "sum").

- **outerfunc**  
  When the centering is TRUE, "+" (Kronecker sum) or "*" (Kronecker product)  
  is calculated (Default: "+").

- **comb**  
  When the centering is FALSE, "random" (random cell-cell pairing) or "all" (all  
  possible cell-cell pairing) is calculated (Default: "random").

- **num.sampling**  
  The number of random sampling used (Default: 100).

- **num.perm**  
  The number of the permutation in label permutation test (Default: 1000).

- **assayNames**  
  The unit of gene expression for using scTensor (e.g. normcounts, cpm...etc)  
  (Default: "counts").

- **decomp**  
  When the value is TRUE, cell-cell interaction tensor is decomposed (Default:  
  TRUE).

**Value**

The result is saved to metadata slot of `SingleCellExperiment` object.

**Author(s)**

Koki Tsuyuzaki

**See Also**

`SingleCellExperiment`.

**Examples**

showMethods("cellCellDecomp")
cellCellRanks  

**Rank estimation of the CCI-tensor**

**Description**

SVD is performed in each mode.

**Usage**

```r
cellCellRanks(sce, centering=TRUE, 
mergeas=c("mean", "sum"), outerfunc=c("*", "+"), comb=c("random", "all"), 
um.sampling=100, num.perm=1000, assayNames = "counts", verbose=FALSE, 
um.iter1=5, num.iter2=5, num.iter3=NULL)
```

**Arguments**

- **sce**: A object generated by instantization of SingleCellExperiment-class.
- **centering**: When the value is TRUE, input matrix is summarized as celltype-level vectors (Default: TRUE).
- **mergeas**: When the centering is TRUE, "mean" (celltype-level mean vector) or "sum" (celltype-level sum vector) is calculated (Default: "mean").
- **outerfunc**: When the centering is TRUE, "+" (Kronecker product) or "*" (Kronecker sum) is calculated (Default: "+").
- **comb**: When the centering is FALSE, "random" (random cell-cell pairing) or "all" (all possible cell-cell pairing) is calculated (Default: "random").
- **num.sampling**: The number of random sampling used (Default: 100).
- **num.perm**: The number of the permutation in label permutation test (Default: 1000).
- **assayNames**: The unit of gene expression for using scTensor (e.g. normcounts, cpm...etc) (Default: "counts").
- **verbose**: The verbose parameter for nnTensor::NTD (Default: FALSE).
- **num.iter1**: The number of iteration to estimate the rank of mode-1 matricised data tensor (Default: 5).
- **num.iter2**: The number of iteration to estimate the rank of mode-2 matricised data tensor (Default: 5).
- **num.iter3**: The number of iteration to estimate the rank of mode-3 matricised data tensor (Default: NULL).

**Value**

- **RSS**: A list with three elements, in which each element means the average reconstructed error in each rank.
- **selected**: A vector with three elements, in which each element means the estimated ranks in mode-1, 2 and 3 matricization.
Author(s)
Koki Tsuyuzaki

See Also
SingleCellExperiment.

Examples
showMethods("cellCellRanks")

cellCellReport

Description
The result is saved as HTML report which contains with multiple files.

Usage
cellCellReport(sce, reducedDimNames,
   out.dir=tempdir(), html.open=FALSE,
   title="The result of scTensor",
   author="The person who runs this script", assayNames = "counts", thr=100,
   top="full", p=0.05, upper=20,
   goenrich=TRUE, meshenrich=TRUE, reactomeenrich=TRUE,
   doenrich=TRUE, ncgenrich=TRUE, dgnenrich=TRUE, nbins=40)

Arguments

sce A object generated by instantiation of SingleCellExperiment-class.
reducedDimNames The name of two-dimentional data saved in reducedDimNames slot of SingleCellExperiment object.
out.dir The output directory for saving HTML report (out.dir: tempdir()).
html.open Whether the result of HTML report is opened when the calculation is finished (Default: FALSE).
title The title of HTML report (Default: "The result of scTensor").
author The author of HTML report (Default: "The person who runs this script").
assayNames The unit of gene expression for using scTensor (e.g. normcounts, cpm...etc) (Default: "counts").
thr The threshold for selection of top percentage of core tensor elements (Default: 100 (1 to 100)).
top top genes in each (*,* *)-pattern which are selected and summarized in the report (Default: "full")
cellCellReport

p  The threshold of p-value of the enrichment analysis (Default: 1E-2)
upper  The maximum number of HTML reports generates (Default: 20)
goenrich  Whether GO-Enrichment analysis is performed (Default: TRUE)
 meshenrich  Whether MeSH-Enrichment analysis is performed (Default: TRUE)
reactomeenrich  Whether Reactome-Enrichment analysis is performed (Default: TRUE)
doenrich  Whether DO-Enrichment analysis is performed (Default: TRUE)
ncgenrich  Whether NCG-Enrichment analysis is performed (Default: TRUE)
dgnenrich  Whether DGN-Enrichment analysis is performed (Default: TRUE)
nbins  The number of bins used for the two dimensional plot of schex (Default: 40)

Value
The result is saved as HTML report which contains with multiple files.

Author(s)
Koki Tsuyuzaki

See Also
SingleCellExperiment.

Examples
if(interactive()){
  # Package Loading
  library("SingleCellExperiment")
  library("AnnotationHub")
  if(!require(LRBaseDbi)){
    BiocManager::install("LRBaseDbi")
    library(LRBaseDbi)
  }
  ah <- AnnotationHub()
  dbfile <- query(ah, c("LRBaseDb", "Homo sapiens", "v002"))[[1]]
  LRBase.Hsa.eg.db <- LRBaseDbi::LRBaseDb(dbfile)

  # Data Loading
  data(GermMale)
  data(labelGermMale)
  data(tsneGermMale)

  # SingleCellExperiment Object
  sce <- SingleCellExperiment(assays=list(counts = GermMale))
  reducedDims(sce) <- SimpleList(TSNE=tsneGermMale$Y)

  # User's Original Normalization Function
  CPMED <- function(input){
    libsize <- colSums(input)
    median(libsize) * t(t(input) / libsize)
```r
# Normalization
normcounts(sce) <- log10(CPMED(counts(sce)) + 1)

# Registration of required information into metadata(sce)
cellCellSetting(sce, LRBase.Hsa.eg.db, names(labelGermMale))

# Rank Estimation
rks <- cellCellRanks(sce, assayNames="normcounts")

# CCI Tensor Decomposition
set.seed(1234)
cellCellDecomp(sce, ranks=rks$selected, assayNames="normcounts")

# HTML Report
options(device.ask.default = FALSE)
cellCellReport(sce, reducedDimNames="TSNE", out.dir=tempdir(), html.open=FALSE,
title="The result of scTensor", author="The person who runs this script",
assayNames="counts", thr=100,
top="full", p=0.05, upper=20,
goenrich=TRUE, meshenrich=TRUE, reactomeenrich=TRUE,
doenrich=TRUE, nogenrich=TRUE, dgenrich=TRUE, nbins=40)

else{
  showMethods("cellCellReport")
}
```

---

**cellCellSetting**  
*Parameter setting for scTensor*

**Description**

All parameters is saved to metadata slot of SingleCellExperiment object.

**Usage**

```r
cellCellSetting(sce, lrbase, label, lr.evidence="known", color=NULL)
```

**Arguments**

- **sce**: A object generated by instantiation of SingleCellExperiment-class.
- **lrbase**: Ligand-Receptor database (LRBase.XXX.eg.db-type package).
- **label**: Cellular label information for distinguishing which cells belong to common celltypes.
- **lr.evidence**: The evidence code for L-R pair list (Default: "known"). When you specify "known", DLRP, IUPHAR, HPMR, CELLPHONEDB, SINGLECELLSIGNALR are searched, and other databases are searched, when you specify "putative". You can also specify multiple databases at once (e.g. c("SWISSPROT_STRING", "TREMBL_STRING")). cf. https://github.com/rikenbit/lrbase-workflow
**cellCellSimulate**

- **color**  
  Color scheme for adding color against the cells (Default: NULL). If the value is not specified, automatically the color vector is generated.

**Value**

The result is saved to metadata slot of SingleCellExperiment object.

**Author(s)**

Koki Tsuyuzaki

**See Also**

*SingleCellExperiment.*

**Examples**

```r
showMethods("cellCellSetting")
```

---

**cellCellSimulate**  
*Parameter Simulate for scTensor*

**Description**

All parameters is saved to metadata slot of SingleCellExperiment object.

**Usage**

```r
cellCellSimulate(params = newCCSParams(), verbose = TRUE)
```

**Arguments**

- **params**  
  A parameter object generated by newCCSParams().
- **verbose**  
  Whether the message is outputted or not (Default: TRUE).

**Value**

A list object containing simcount, LR, and celltype. simcount is the synthetic count matrix, LR is the synthetic ligand-receptor pair list, and celltype is the vector to specify the celltype of each column of simcount.

**Author(s)**

Koki Tsuyuzaki

**Examples**

```r
showMethods("cellCellSimulate")
```
GermMale

The matrix which is used as test data of scTensor.

Description

A matrix with 242 rows (genes) * 852 columns (cells).

Usage

data(GermMale)

Details


Only male data is extracted and then the gene symbol is converted to NCBI Gene ID by Homo.sapiens package.

For saving the package size, the number of genes are strictly reduced by the standard of highly variable genes with threshold of p-value is 1E-300.

References


See Also

labelGermMale, tsneGermMale.

Examples

data(GermMale)

getParam

Get a parameter

Description

Accessor function for getting parameter values.

Usage

getParam(object, name)

## S4 method for signature 'CCSParams'
getParam(object, name)
Arguments

    object object to get parameter from.
    name   name of the parameter to get.

Value

    The extracted parameter value

Examples

    params <- newCCSParams()
    getParam(params, "nGene")
    getParam(params, "nCell")
    getParam(params, "CCIInfo")
    getParam(params, "lambda")
    getParam(params, "seed")

---

labelGermMale    The vector contains the celltype information and color scheme of GermMale

Description

    A vector with 852 length (cells).

Usage

    data(labelGermMale)

Details

    The Cluster label is downloaded from original paper page of Cell Stem Cell (https://www.sciencedirect.com/science/article/pii/S1934590917300784)

References


See Also

    GermMale, tsneGermMale.

Examples

    data(labelGermMale)
m

The gene-wise mean vector of Quartz-Seq data.

Description
This data is internally used in cellCellSimulate function.

Usage
data(m)

Examples
data(m)

newCCSPrams

New Params

Description
Create a new CCSPrams object.

Usage
newCCSPrams()

Arguments
Nothing.

Value
New Params object.

Examples
params <- newCCSPrams()
setParam

Set a parameter

Description
Function for setting parameter values.

Usage

setParam(object, name) <- value
## S4 method for signature 'CCSPrams'
setParam(object, name, value)

Arguments

object    object to set parameter in.
name      name of the parameter to set.
value     value to set the parameter to.

Value
Object with new parameter value.

Examples

params <- newCCSPrams()

setParam(params, "nGene") <- 20000
setParam(params, "nCell") <- c(12, 43, 323)
setParam(params, "cciInfo") <- list(nPair=2000,
   CCI1=list(
      LPattern=c(1,0,0),
      RPATTERN=c(0,1,1),
      nGene=100,
      fc="E10"),
   CCI2=list(
      LPATTERN=c(0,0,1),
      RPATTERN=c(1,1,1),
      nGene=200,
      fc="E10"),
   CCI3=list(
      LPATTERN=c(1,1,1),
      RPATTERN=c(1,0,1),
      nGene=300,
      fc="E10")
   )
setParam(params, "lambda") <- 0.1
setParam(params, "seed") <- 111
tsneGermMale  
*The result of Rtsne against GermMale*

**Description**

A List contains some parameters and the result of Rtsne function.

**Usage**

```r
data(tsneGermMale)
```

**Details**

Rtsne is performed as follows.
```
library(Rtsne) set.seed(123) tsneGermMale <- Rtsne(dist(t(GermMale)), is_distance=TRUE, perplexity=40)
```

**References**


**See Also**

`labelGermMale`, `GermMale`.

**Examples**

```r
data(tsneGermMale)
```

---

v  
*The gene-wise variance vector of Quartz-Seq data.*

**Description**

This data is internally used in cellCellSimulate function.

**Usage**

```r
data(v)
```

**Examples**

```r
data(v)
```
Index

* classes
  CCSParams-class, 3

* datasets
  GermMale, 10
  labelGermMale, 11
  m, 12
  tsneGermMale, 14
  v, 14

* methods
  cellCellDecomp, 3
  cellCellRanks, 5
  cellCellReport, 6
  cellCellSetting, 8
  cellCellSimulate, 9

* package
  scTensor-package, 2
  CCSParams-class, 3
  cellCellDecomp, 2, 3
  cellCellDecomp,SingleCellExperiment-method
    (cellCellDecomp), 3
  cellCellRanks, 5
  cellCellRanks,SingleCellExperiment-method
    (cellCellRanks), 5
  cellCellReport, 2, 6
  cellCellReport,SingleCellExperiment-method
    (cellCellReport), 6
  cellCellSetting, 2, 8
  cellCellSetting,SingleCellExperiment-method
    (cellCellSetting), 8
  cellCellSimulate, 9
  cellCellSimulate,SingleCellExperiment-method
    (cellCellSimulate), 9

  GermMale, 2, 10, 11, 14
  getParam, 3, 10
  getParam,CCSParams-method (getParam), 10

  labelGermMale, 2, 10, 11, 14
  m, 12
  newCCSParams, 3, 12
  scTensor (scTensor-package), 2
  scTensor-package, 2
  setParam, 13
  setParam,CCSParams,ANY-method
    (setParam), 13
  setParam,CCSParams-method (setParam), 13
  setParam<-,CCSParams,ANY-method
    (setParam), 13
  setParam<-,CCSParams-method (setParam),
    13
  SingleCellExperiment, 4, 6, 7, 9
  tsneGermMale, 2, 10, 11, 14
  v, 14