Package: scTensor (via r-universe)

November 25, 2026

Type Package

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

Version 2.24.0

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

Config/pak/sysreqs libabsl-dev cmake libfontconfig1-dev libfreetype6-dev libgdal-dev gdal-bin libgeos-dev libglpk-dev make libmagick++-dev gsfonts libicu-dev libpng-dev libuv1-dev libxml2-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev libnode-dev zlib1g-dev

Repository Bioconductor 3.24

Date/Publication 2026-11-23 22:41:56 UTC

RemoteUrl https://github.com/bioc/scTensor

RemoteRef RELEASE_3_24

RemoteSha b3f4e8515ffdb8d1661fde62b86ac154b7763abf

Contents

scTensor-package 2

BuildSignedCCI . . . 3
CCSParams-class . . . 4
cellCellDecomp . . . 4
cellCellRanks . . . 6
cellCellReport . . . 7
cellCellSetting . . . 9
cellCellSimulate . . . 10
GermMale . . . 11
getParam . . . 12
labelGermMale . . . 12
m . . . 13
newCCSParams . . . 13
setParam . . . 14
tsneGermMale . . . 15
v . . . 15

Index **16**

`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")
```

`BuildSignedCCI` *Build Signed Cell-Cell Interaction Graphs*

Description

Constructs positive and negative cell-cell interaction matrices from ligand/receptor expression and a ligand-receptor database with sign annotations. The LR scores $X_{ijk} = L_{ik}R_{jk}$ are aggregated into A^+ (activating) and A^- (inhibitory/death-inducing) matrices based on the sign of each LR pair.

Usage

```
BuildSignedCCI(lig_expr, rec_expr, lr_sign)
```

Arguments

`lig_expr`	Matrix of ligand expression (cells x LR pairs), or (cell types x LR pairs) if pre-aggregated.
`rec_expr`	Matrix of receptor expression (cells x LR pairs), same dimensions as `lig_expr`.
`lr_sign`	Integer or numeric vector of length `ncol(lig_expr)`. +1 for activating LR pairs, -1 for inhibitory/death-inducing LR pairs.

Value

A list with components:

A_pos Positive (activating) cell-cell interaction matrix (n x n)

A_neg Negative (inhibitory) cell-cell interaction matrix (n x n)

X_pos 3D array of per-LR positive scores (sender x receiver x LR)

X_neg 3D array of per-LR negative scores (sender x receiver x LR)

edge_table Data frame with columns: sender, receiver, lr_pair, score, sign

Examples

```
# 3 cell types, 4 LR pairs (2 activating, 2 inhibitory)
lig <- matrix(c(1,0,0, 0,1,0, 1,1,0, 0,0,1), nrow = 3)
rec <- matrix(c(0,1,0, 1,0,1, 0,0,1, 1,0,0), nrow = 3)
signs <- c(1, 1, -1, -1)
result <- BuildSignedCCI(lig, rec, signs)
result$A_pos
result$A_neg
```

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), th
    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 instantization of SingleCellExperiment-class.
algorithm	Algorithm for constrcting 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 pcomb (Default: 25).
thr3	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 calculed (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

```
cellCellRanks(sce, centering=TRUE,
    mergeas=c("mean", "sum"), outerfunc=c("*", "+"), comb=c("random", "all"),
    num.sampling=100, num.perm=1000, assayNames = "counts", verbose=FALSE,
    num.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) or is calculated (Default: "+").
`comb`	When the centering is FALSE, "random" (random cell-cell pairing) or "all" (all possible cell-cell pairing) is calculed (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` *HTML report of the result of scTensor*

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=FALSE, nbins=40)
```

Arguments

`sce`	A object generated by instantization 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 pecentage 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")

p	The threshold of p-value of the enrichment analysis (Default: 1E-2)
upper	The maxium 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: FALSE). DGN enrichment is currently unavailable because `enrichDGN()` was removed from the **DOSE** package following a DisGeNET licensing change. If set to `TRUE`, a warning is issued and the parameter is treated as `FALSE`.
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"))[[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)
}
# 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, ncgenrich=TRUE, dgnenrich=FALSE, nbins=40)
   }else{
       showMethods("cellCellReport")
   }
```

`cellCellSetting` *Parameter setting for scTensor*

Description

All parameters is saved to metadata slot of SingleCellExperiment object.

Usage

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

Arguments

`sce`	A object generated by instantization of SingleCellExperiment-class.
`lrbase`	Ligand-Receptor database (LRBase.XXX.eg.db-type package).
`label`	Cellular label information for distingusishing 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
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

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

---

cellCellSimulate *Parameter Simulate for scTensor*

Description

All parameters is saved to metadata slot of SingleCellExperiment object.

Usage

```
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 specity the celltype of the each column of simcount.

Author(s)

Koki Tsuyuzaki

Examples

```
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

The data matrix is downloaded from GEO Series GSE86146 (https://www.ncbi.nlm.nih.gov/geo/download/?acc=GSE86146&
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 strictlly reduced by the standard of highlly variable genes with threshold of p-value is 1E-300.

References

Li L. and Dong J. and Yan L. and Yong J. et al. (2017) Single-Cell RNA-Seq Analysis Maps Development of Human Germline Cells and Gonadal Niche Interactions. *Cell Stem Cell*, **20(6)**: 858-873

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/pi

References

Li L. and Dong J. and Yan L. and Yong J. et al. (2017) Single-Cell RNA-Seq Analysis Maps Development of Human Germline Cells and Gonadal Niche Interactions. *Cell Stem Cell*, **20(6)**: 858-873

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)
```

---

newCCSParams *New Params*

Description

Create a new CCSParams object.

Usage

```
newCCSParams()
```

Value

New Params object.

Examples

```
params <- newCCSParams()
```

---

setParam *Set a parameter*

Description

Function for setting parameter values.

Usage

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

Arguments

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

Value

Object with new parameter value.

Examples

```
params <- newCCSParams()

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

```
data(tsneGermMale)
```

Details

Rtsne is performed as follows.

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

References

Li L. and Dong J. and Yan L. and Yong J. et al. (2017) Single-Cell RNA-Seq Analysis Maps Development of Human Germline Cells and Gonadal Niche Interactions. *Cell Stem Cell*, **20(6)**: 858-873

See Also

labelGermMale, GermMale.

Examples

```
data(tsneGermMale)
```

---

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

Description

This data is internally used in cellCellSimulate function.

Usage

```
data(v)
```

Examples

```
data(v)
```

Index

∗ **classes**
CCSParams-class, 4
∗ **datasets**
GermMale, 11
labelGermMale, 12
m, 13
tsneGermMale, 15
v, 15
∗ **methods**
cellCellDecomp, 4
cellCellRanks, 6
cellCellReport, 7
cellCellSetting, 9
cellCellSimulate, 10
∗ **package**
scTensor-package, 2

BuildSignedCCI, 3

CCSParams-class, 4
cellCellDecomp, 2, 4
cellCellDecomp,SingleCellExperiment-method *(*cellCellDecomp*)*, 4
cellCellRanks, 6
cellCellRanks,SingleCellExperiment-method *(*cellCellRanks*)*, 6
cellCellReport, 2, 7
cellCellReport,SingleCellExperiment-method *(*cellCellReport*)*, 7
cellCellSetting, 2, 9
cellCellSetting,SingleCellExperiment-method *(*cellCellSetting*)*, 9
cellCellSimulate, 10
cellCellSimulate,SingleCellExperiment-method *(*cellCellSimulate*)*, 10

GermMale, 2, 11, *13*, *15*
getParam, 4, 12
getParam,CCSParams-method *(*getParam*)*, 12

labelGermMale, 2, *11*, 12, *15*

m, 13

newCCSParams, *4*, 13

scTensor *(*scTensor-package*)*, 2
scTensor-package, 2
setParam, 14
setParam,CCSParams,ANY-method *(*setParam*)*, 14
setParam,CCSParams-method *(*setParam*)*, 14
setParam<- *(*setParam*)*, 14
setParam<-,CCSParams,ANY-method *(*setParam*)*, 14
setParam<-,CCSParams-method *(*setParam*)*, 14
SingleCellExperiment, *5*, *7*, *8*, *10*

tsneGermMale, 2, *11*, *13*, 15

v, 15