

# Package ‘scTGIF’

October 16, 2023

**Type** Package

**Title** Cell type annotation for unannotated single-cell RNA-Seq data

**Version** 1.14.0

**Depends** R (>= 3.6.0)

**Imports** GSEABase, Biobase, SingleCellExperiment, BiocStyle, plotly, tagcloud, rmarkdown, Rcpp, grDevices, graphics, utils, knitr, S4Vectors, SummarizedExperiment, RColorBrewer, nnTensor, methods, scales, msigdb, schex, tibble, ggplot2, igraph

**Suggests** testthat

**Description** scTGIF connects the cells and the related gene functions without cell type label.

**License** Artistic-2.0

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

**VignetteBuilder** knitr

**git\_url** <https://git.bioconductor.org/packages/scTGIF>

**git\_branch** RELEASE\_3\_17

**git\_last\_commit** 9e2b5e4

**git\_last\_commit\_date** 2023-04-25

**Date/Publication** 2023-10-15

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

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

## R topics documented:

scTGIF-package . . . . .	2
calcTGIF . . . . .	2
cellMarkerToGmt . . . . .	3
convertRowID . . . . .	4
DistalLungEpithelium . . . . .	6

label.DistalLungEpithelium . . . . .	6
pca.DistalLungEpithelium . . . . .	7
reportTGIF . . . . .	7
settingTGIF . . . . .	9

<b>Index</b>	<b>11</b>
--------------	-----------

---

scTGIF-package	<i>Cell type annotation for unannotated single-cell RNA-Seq data</i>
----------------	--

---

## Description

scTGIF connects the cells and the related gene functions without cell type label.

## Details

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

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

[calcTGIF](#) function calculates what kind of cellular patterns and functional patterns are contained in single-cell RNA-seq data and [reportTGIF](#) function generates report of analytic result. The algorithm is based on joint NMF, which is implemented in nnTensor package.

## Author(s)

NA

Maintainer: NA

## References

Dominic Grun, Anna Lyubimova, Lennart Kester, Kay Wiebrands, Onur Basak, Nobuo Sasaki, Hans Clevers, Alexander van Oudenaarden (2015) Single-cell messenger RNA sequencing reveals rare intestinal cell types. *Nature*, **525**: 251-255

---

calcTGIF	<i>Function for connecting cellular patterns and functional patterns using jNMF</i>
----------	---

---

## Description

[calcTGIF](#) function calculates what kind of cellular patterns and functional patterns are contained in single-cell RNA-seq data and [reportTGIF](#) function generates report of analytic result.

## Usage

```
calcTGIF(sce, ndim, verbose=FALSE, droplet=TRUE)
```

**Arguments**

sce	A object generated by instantiation of SingleCellExperiment-class.
ndim	The number of low-dimension of joint NMF algorithm.
verbose	The verbose parameter for nnTensor::jNMF (Default: FALSE).
droplet	Whether Droplet-based single-cell RNA-Seq or not (Default: TRUE).

**Value**

The result is saved to metadata slot of SingleCellExperiment object.

**Author(s)**

NA

**Examples**

```
showMethods("calcTGIF")
```

---

cellMarkerToGmt	<i>A function to convert the CellMarker data to GMT files.</i>
-----------------	--

---

**Description**

The GMT (Gene Matrix Transposed file format : \*.gmt) file is formatted by the Broad Institute ([https://software.broadinstitute.org/cancer/software/gsea/wiki/index.php/Data\\_formats#GMT:\\_Gene\\_Matrix\\_Transposed\\_files](https://software.broadinstitute.org/cancer/software/gsea/wiki/index.php/Data_formats#GMT:_Gene_Matrix_Transposed_files)). The data can be downloaded from the website of CellMarker (<http://biocc.hrbmu.edu.cn/CellMarker>).

**Usage**

```
cellMarkerToGmt(infile, outfile,
  uniq.column=c("tissueType", "cellName"),
  geneid.type=c("geneID", "geneSymbol"))
```

**Arguments**

infile	The input file downloaded from CellMarker website
outfile	The output GMT file converted from the CellMarker data
uniq.column	The duplicated terms in the specified column are aggregated as a row of GMT file (Default: geneID)
geneid.type	Output gene identifier. (Default: geneID)

**Value**

output A GMT file is generated.

**Author(s)**

NA

**Examples**

```

library("GSEABase")

tmp <- tempdir()
infile1 = paste0(tmp, "/Human_cell_markers.txt")
outfile1_1 = paste0(tmp, "/Human_cell_markers_1.gmt")
outfile1_2 = paste0(tmp, "/Human_cell_markers_2.gmt")
outfile1_3 = paste0(tmp, "/Human_cell_markers_3.gmt")
outfile1_4 = paste0(tmp, "/Human_cell_markers_4.gmt")

sink(infile1)
cat("speciesType\ttissueType\tUberonOntologyID\tcancerType\tcellType\tcellName\tCellOntologyID\tcellMarker\tgt
cat("Human\tKidney\tUBERON_0002113\tNormal\tNormal cell\tProximal tubular cell\tNA\tIntestinal Alkaline Phospha
cat("Human\tLiver\tUBERON_0002107\tNormal\tNormal cell\tIto cell (hepatic stellate cell)\tCL_0000632\tSynaptoph
cat("Human\tEndometrium\tUBERON_0001295\tNormal\tNormal cell\tTrophoblast cell\tCL_0000351\tCEACAM1\tCEACAM1\t
cat("Human\tGerm\tUBERON_0000923\tNormal\tNormal cell\tPrimordial germ cell\tCL_0000670\tVASA\tDDX4\t54514\tDD
cat("Human\tCorneal epithelium\tUBERON_0001772\tNormal\tNormal cell\tEpithelial cell\tCL_0000066\tKLF6\tKLF6\t
cat("Human\tPlacenta\tUBERON_0001987\tNormal\tNormal cell\tCytotrophoblast\tCL_0000351\tFGF10\tFGF10\t2255\tF
cat("Human\tPeriosteum\tUBERON_0002515\tNormal\tNormal cell\tPeriosteum-derived progenitor cell\tNA\tCD166, CD
cat("Human\tAmniotic membrane\tUBERON_0009742\tNormal\tNormal cell\tAmnion epithelial cell\tCL_0002536\tNANOG,
cat("Human\tPrimitive streak\tUBERON_0004341\tNormal\tNormal cell\tPrimitive streak cell\tNA\tLHX1, MIXL1\tLHX1
sink()

cellMarkerToGmt(infile1, outfile1_1, uniq.column=c("tissueType"),
  geneid.type=c("geneID"))
cellMarkerToGmt(infile1, outfile1_2, uniq.column=c("tissueType"),
  geneid.type=c("geneSymbol"))
cellMarkerToGmt(infile1, outfile1_3, uniq.column=c("cellName"),
  geneid.type=c("geneID"))
cellMarkerToGmt(infile1, outfile1_4, uniq.column=c("cellName"),
  geneid.type=c("geneSymbol"))

gmt1_1 <- getGmt(outfile1_1)
gmt1_2 <- getGmt(outfile1_2)
gmt1_3 <- getGmt(outfile1_3)
gmt1_4 <- getGmt(outfile1_4)

```

---

convertRowID

*A function to change the row names of a matrix.*


---

**Description**

To avoid to specify the duplicated row names against matrix, multiple aggregation rules are implemented.

**Usage**

```
convertRowID(input, rowID, LtoR,
             aggr.rule=c("sum", "mean", "large.mean", "large.var", "large.cv2"))
```

**Arguments**

input	A matrix filled with number (n * m).
rowID	A vector to specify the row names of input (length: n).
LtoR	A corresponding table to convert the row names of input as different type of IDs. (Left: current row names -> Right: new row names)
aggr.rule	The aggregation rule to change the row names of input and collapse/select the values, if the row names changed by LtoR are duplicated. sum: Collapses multiple row vectors by summation. mean: Collapses multiple row vectors by mean. large.mean: Select a vector having the largest mean in the duplicated vectors. large.var: Select a vector having the largest variance in the duplicated vectors. large.cv2: Select a vector having the largest CV2 in the duplicated vectors.

**Value**

output	A matrix, in which the row names are changed, according to the aggregation rule user specified.
ctable	The corresponding table explaining the relationship between previous row names and changed row names of input.

**Author(s)**

NA

**Examples**

```
input <- matrix(1:20, nrow=4, ncol=5)
rowID <- c("A", "B", "C", "D")
LtoR <- rbind(
  c("A", "3"),
  c("B", "2"),
  c("C", "4"),
  c("D", "7"))
(input2 <- convertRowID(input, rowID, LtoR, "sum"))
(input3 <- convertRowID(input, rowID, LtoR, "mean"))
(input4 <- convertRowID(input, rowID, LtoR, "large.mean"))
(input5 <- convertRowID(input, rowID, LtoR, "large.var"))
(input6 <- convertRowID(input, rowID, LtoR, "large.cv2"))
```

---

DistalLungEpithelium *Gene expression matrix of DistalLungEpithelium dataset containing five cluster.*

---

**Description**

A data frame with 3397 rows (genes) with following 80 columns (cells).

The data is downloaded as supplementary information of the distal lung epithelium paper (<https://www.nature.com/articles/na>)

Low-expressed genes are filtered.

All Gene ID is converted to Human Entrez Gene ID for applying the data to scTGIF.

**Usage**

```
data("DistalLungEpithelium")
```

**Source**

<http://www.nature.com/nbt/journal/v33/n2/full/nbt.3102.html>

**References**

Treutlein, B. et al. (2014) Reconstructing lineage hierarchies of the distal lung epithelium using single-cell RNA-seq. *Nature* **509**, 371-375

**Examples**

```
data("DistalLungEpithelium")
```

---

label.DistalLungEpithelium  
*Cellular label of DistalLungEpithelium dataset containing five cluster.*

---

**Description**

A vector containing 80 elements (cells).

**Usage**

```
data("label.DistalLungEpithelium")
```

**References**

Treutlein, B. et al. (2014) Reconstructing lineage hierarchies of the distal lung epithelium using single-cell RNA-seq. *Nature* **509**, 371-375

**Examples**

```
data("label.DistalLungEpithelium")
```

---

```
pca.DistalLungEpithelium
```

*The result of PCA of the DistalLungEpithelium dataset.*

---

**Description**

A matrix having 80 (cells) \* 2 (PCs) elements.

**Usage**

```
data("pca.DistalLungEpithelium")
```

**References**

Treutlein, B. et al. (2014) Reconstructing lineage hierarchies of the distal lung epithelium using single-cell RNA-seq. *Nature* **509**, 371-375

**Examples**

```
data("pca.DistalLungEpithelium")
```

---

```
reportTGIF
```

*Function for reporting the result of [calcTGIF](#) function*

---

**Description**

[calcTGIF](#) function calculates what kind of cellular patterns and functional patterns are contained in single-cell RNA-seq data and [reportTGIF](#) function generates report of analytic result.

**Usage**

```
reportTGIF(sce, out.dir=tempdir(), html.open=FALSE,
           title="The result of scTGIF",
           author="The person who runs this script",
           assayNames="counts")
```

**Arguments**

sce	A object generated by instantiation of SingleCellExperiment-class.
out.dir	Output directory user want to save the report (Default: tempdir()).
html.open	Whether html is opened when <a href="#">reportTGIF</a> is finished (Default: FALSE)
title	Title of report (Default: "The result of scTGIF")
author	The name of user name (Default: "The person who runs this script")
assayNames	The unit of gene expression for using scTGIF (e.g. normcounts, cpm...etc) (Default: "counts").

**Value**

Some file is generated to output directory user specified.

**Author(s)**

NA

**Examples**

```

if(interactive()){
  # Package loading
  library("SingleCellExperiment")
  library("GSEABase")
  library("msigbr")

  # Test data
  data("DistalLungEpithelium")
  data("pca.DistalLungEpithelium")
  data("label.DistalLungEpithelium")

  # Test data
  par(ask=FALSE)
  plot(pca.DistalLungEpithelium, col=label.DistalLungEpithelium, pch=16,
       main="Distal lung epithelium dataset", xlab="PCA1",
       ylab="PCA2", bty="n")
  text(0.1, 0.05, "AT1", col="#FF7F00", cex=2)
  text(0.07, -0.15, "AT2", col="#E41A1C", cex=2)
  text(0.13, -0.04, "BP", col="#A65628", cex=2)
  text(0.125, -0.15, "Clara", col="#377EB8", cex=2)
  text(0.09, -0.2, "Ciliated", col="#4DAF4A", cex=2)

  # Load the gmt file from MSigDB
  # Only "Entrez Gene ID" can be used in scTGIF
  # e.g. gmt <- GSEABase::getGmt(
  #   "/PATH/YOU/SAVED/THE/GMTFILES/h.all.v6.0.entrez.gmt")
  # Here we use msigbr to retrieve mouse gene sets

  # Mouse gene set (NCBI Gene ID)
  m_df <- msigbr(species = "Mus musculus", category = "H"),
  c("gs_name", "entrez_gene")]

  # Convert to GeneSetCollection
  hallmark = unique(m_df$gs_name)
  gsc <- lapply(hallmark, function(h){
    target = which(m_df$gs_name == h)
    geneIds = unique(as.character(m_df$entrez_gene[target]))
    GeneSet(setName=h, geneIds)
  })
  gmt <- GeneSetCollection(gsc)

  # SingleCellExperiment-class
  sce <- SingleCellExperiment(

```



```

    assays = list(counts = DistalLungEpithelium))
  reducedDims(sce) <- SimpleList(PCA=pca.DistalLungEpithelium)

  # 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)
  settingTGIF(sce, gmt, reducedDimNames="PCA",
    assayNames="normcounts")

  # Functional Annotation based on jNMF
  calcTGIF(sce, ndim=7)

  # HTML Reprt
  reportTGIF(sce,
    html.open=TRUE,
    title="scTGIF Report for DistalLungEpithelium dataset",
    author="Koki Tsuyuzaki")
}

```

---

 settingTGIF

*Parameter setting for scTGIF*


---

## Description

All parameters is saved to metadata slot of SingleCellExperiment object.

## Usage

```
settingTGIF(sce, gmt, reducedDimNames, assayNames="counts", nbins=40)
```

## Arguments

sce	A object generated by instantiation of SingleCellExperiment-class.
gmt	Object generated from GSEABase::getGmt function. GMT file can be downloaded from MSigDB web (site <a href="http://software.broadinstitute.org/gsea/login.jsp#msigdb">http://software.broadinstitute.org/gsea/login.jsp#msigdb</a> ). Please confirm that the gmt file contains Human Entrez Gene ID, not gene symbol. Also confirm that the DataMatrix has Human Entrez Gene ID.
reducedDimNames	The names of reducedDim(sce) that user want use in scTGIF.
assayNames	The unit of gene expression for using scTGIF (e.g. normcounts, cpm...etc) (Default: "counts").
nbins	The number of bins of schex (Default: 40).

**Value**

The result is saved to metadata slot of SingleCellExperiment object.

**Author(s)**

NA

**Examples**

```
showMethods("settingTGIF")
```

# Index

## \* datasets

- DistalLungEpithelium, 6
- label.DistalLungEpithelium, 6
- pca.DistalLungEpithelium, 7

## \* methods

- calcTGIF, 2
- cellMarkerToGmt, 3
- convertRowID, 4
- reportTGIF, 7
- settingTGIF, 9

## \* package

- scTGIF-package, 2

calcTGIF, 2, 2, 7

calcTGIF, SingleCellExperiment-method  
(calcTGIF), 2

cellMarkerToGmt, 3

convertRowID, 4

DistalLungEpithelium, 6

label.DistalLungEpithelium, 6

pca.DistalLungEpithelium, 7

reportTGIF, 2, 7, 7

reportTGIF, SingleCellExperiment-method  
(reportTGIF), 7

scTGIF (scTGIF-package), 2

scTGIF-package, 2

settingTGIF, 9

settingTGIF, SingleCellExperiment-method  
(settingTGIF), 9