

# Package ‘CHETAH’

December 5, 2024

**Title** Fast and accurate scRNA-seq cell type identification

**Type** Package

**Version** 1.22.0

**Date** 2021-11-20

**Description** CHETAH (CHAracterization of cEll Types Aided by Hierarchical classification) is an accurate, selective and fast scRNA-seq classifier.

Classification is guided by a reference dataset, preferentially also a scRNA-seq dataset. By hierarchical clustering of the reference data, CHETAH creates a classification tree that enables a step-wise, top-to-bottom classification. Using a novel stopping rule,

CHETAH classifies the input cells to the cell types of the references and to “intermediate types”: more general classifications that ended in an intermediate node of the tree.

**Imports** shiny, plotly, pheatmap, bioDist, dendextend, cowplot, corrplot, grDevices, stats, graphics, reshape2, S4Vectors, SummarizedExperiment

**Depends** R (>= 4.2), ggplot2, SingleCellExperiment

**License** file LICENSE

**Encoding** UTF-8

**biocViews** Classification, RNASeq, SingleCell, Clustering, GeneExpression, ImmunoOncology

**RoxygenNote** 7.2.0

**Suggests** knitr, rmarkdown, Matrix, testthat, vdiff

**VignetteBuilder** knitr

**LazyData** false

**BugReports** <https://github.com/jdekanter/CHETAH>

**URL** <https://github.com/jdekanter/CHETAH>

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

**git\_branch** RELEASE\_3\_20

**git\_last\_commit** f5484b0

**git\_last\_commit\_date** 2024-10-29

**Repository** Bioconductor 3.20

**Date/Publication** 2024-12-05

**Author** Jurrian de Kanter [aut, cre],  
Philip Lijnzaad [aut]

**Maintainer** Jurrian de Kanter <jurriandekanter@gmail.com>

## Contents

CHETAHclassifier . . . . .	2
CHETAHshiny . . . . .	5
Classify . . . . .	5
ClassifyReference . . . . .	6
CorrelateReference . . . . .	7
headneck_ref . . . . .	8
input_mel . . . . .	8
PlotCHETAH . . . . .	9
PlotTree . . . . .	10
PlotTSNE . . . . .	11
RenameBelowNode . . . . .	12
<b>Index</b>	<b>13</b>

---

CHETAHclassifier	<i>Identification of cell types aided by hierarchical clustering</i>
------------------	--

---

## Description

CHETAH classifies an input dataset by comparing it to a reference dataset in a stepwise, top-to-bottom fashion. See 'details' for a full explanation. *NOTE: We recommend to use all the default parameters*

## Usage

```
CHETAHclassifier(
  input,
  ref_cells = NULL,
  ref_profiles = NULL,
  ref_ct = "celltypes",
  input_c = NA,
  ref_c = NA,
  thresh = 0.1,
  gs_method = c("fc", "wilcox"),
  cor_method = c("spearman", "kendall", "pearson", "cosine"),
  clust_method = c("average", "single", "complete", "ward.D2", "ward.D", "mcquitty",
    "median", "centroid"),
  clust_dist = bioDist::spearman.dist,
  n_genes = 200,
  pc_thresh = 0.2,
  p_thresh = 0.05,
  fc_thresh = 1.5,
  subsample = FALSE,
```

```

    fix_ngenes = TRUE,
    plot.tree = FALSE,
    only_pos = FALSE,
    print_steps = FALSE
)

```

### Arguments

input	<b>required:</b> an input SingleCellExperiment. (see: <a href="#">Bioconductor</a> , and the vignette <code>browseVignettes("CHETAH")</code> )
ref_cells	<b>required:</b> A reference SingleCellExperiment, with the cell types in the "cell-types" colData (or otherwise defined in ref_ct.
ref_profiles	<i>optional</i> In case of bulk-RNA seq or micro-arrays, an expression matrix with one (average) reference expression profile per cell type in the columns. ('ref_cells' must be left empty)
ref_ct	the colData of ref_cells where the cell types are stored.
input_c	the name of the assay of the input to use. NA (default) will use the first one.
ref_c	same as input_c, but for the reference.
thresh	the initial confidence threshold, which can be changed after running by <a href="#">Classify</a> )
gs_method	method for gene selection. In every node of the tree: "fc" = quick method: either a fixed number (n_genes) of genes is selected with the highest fold-change (default), or genes are selected that have a fold-change higher than fc_thresh (the latter is used when fix_ngenes = FALSE) . "wilcox": genes are selected based on fold-change (fc_thresh), percentage of expression (pc_thresh) and p-values (p_thresh), p-values are found by the wilcox test.
cor_method	the correlation measure: one of: "spearman" (default), "kendall", "pearson", "cosine"
clust_method	the method used for clustering the reference profiles. One of the methods from <a href="#">hclust</a>
clust_dist	a distance measure, default: <a href="#">spearman.dist</a>
n_genes	The number of genes used in every step. Only used if fix_ngenes = TRUE
pc_thresh	when: <i>gs_method</i> = "wilcox", only genes are selected for which more than a pc_tresh fraction of a reference group of cells express that gene
p_thresh	when: <i>gs_method</i> = "wilcox" , only genes are selected that have a p-value < p_thresh
fc_thresh	when: <i>gs_method</i> = "wilcox" or <i>gs_method</i> = "fc" AND <i>fix_ngenes</i> = FALSE, only genes are selected that have a log2 fld-change > fc_thresh between two reference groups. <b>if this mode is selected, the reference must be in the log2 space.</b>
subsample	to prevent reference types with a lot of cells to influence the gene selection, subsample types with more that subsample cells
fix_ngenes	when: <i>gs_method</i> = "fc" use a fixed number of genes for all correlations. when: <i>gs_method</i> = "wilcox" use a maximum of genes per step. When fix_ngenes = FALSE & <i>gs_method</i> = "fc" fc_thresh is used to define the fold-change cut-off for gene selection.
plot.tree	Plot the classification tree.

only_pos	<i>not recommended</i> : only use genes for a reference type that are higher expressed in that type, than the others in that node.
print_steps	whether the number of genes (positive and negative) per step per ref_cell_type should be printed

## Details

CHETAH will hierarchically cluster reference data to produce a classification tree (ct). In each node of the ct, CHETAH will assign each input cell to one of the two branches, based on gene selections, correlations and calculation of profile and confidence scores. The assignment will only be performed if the confidence score for such an assignment is higher than the Confidence Threshold. If this is not the case, classification for the cell will stop in the current node. Some input cells will reach the leaf nodes of the ct (the pre-defined cell types), these classifications are called **final types**. For other cells, assignment will stop in a node. These classifications are called **intermediate types**.

## Value

A SingleCellExperiment with added: - input\$celltype\_CHETAH a named character vector that can directly be used in any other workflow/method. - "hidden" 'int\_colData' and 'int\_metadata', not meant for direct interaction, but which can all be viewed and interacted with using: 'PlotCHETAH' and 'CHETAHshiny'. A list containing the following objects is added to input\$int\_metadata\$CHETAH

- **classification** a named vector: the classified types with the corresponding names of the input cells
- **tree** the hclust object of the classification tree
- **nodetypes** A list with the cell types under each node
- **nodecoor** the coordinates of the nodes of the classification tree
- **genes** A list per node, containing a list per reference type with the genes used for the profile scores of that type
- **parameters** The parameters used

A nested DataFrame is added to input\$int\_colData\$CHETAH. It holds 3 top-levels DataFrames

- **prof\_scores** A list with the profile scores
- **conf\_scores** A list with the confidence scores
- **correlations** A list with the correlations of the input cells to the reference profiles

## Examples

```
data('input_mel')
data('headneck_ref')
## Melanoma data from Tirosh et al. (2016) Science
input_mel
## Head-Neck data from Puram et al. (2017) Cancer Cell
headneck_ref
input_mel <- CHETAHclassifier(input = input_mel, ref_cells = headneck_ref)
```

---

 CHETAHshiny

*Launch a web page to interactively go through the classification*


---

**Description**

Launch a web page to interactively go through the classification

**Usage**

```
CHETAHshiny(input, redD = NA, input_c = NA)
```

**Arguments**

input	a SingleCellExperiment on which <a href="#">CHETAHclassifier</a> has been run
redD	the name of the reducedDim of the input to use for plotting
input_c	the name of the assay of the input to use. NA (default) will use the first one.

**Value**

Opens a web page in your default browser

---

 Classify

*(Re)classify after running [CHETAHclassifier](#) using a confidence threshold*
*NOTE: In case of bulk reference profiles: only the correlations will be used, as the data does not allow for profile or confidence scores to be calculated.*


---

**Description**

(Re)classify after running [CHETAHclassifier](#) using a confidence threshold

NOTE: In case of bulk reference profiles: only the correlations will be used, as the data does not allow for profile or confidence scores to be calculated.

**Usage**

```
Classify(input, thresh = 0.1, return_clas = FALSE)
```

**Arguments**

input	a SingleCellExperiment on which <a href="#">CHETAHclassifier</a> has been run
thresh	a confidence threshold between -0 and 2. Selecting 0 will classify all cells, whereas 2 will result in (almost) no cells to be classified. <i>recommended: between 0.1 (fairly confident) and 1 (very confident)</i>
return_clas	Instead of returning the SingleCellExperiment, only return the classification vector

**Value**

a character vector of the cell types with the names of the cells

**Examples**

```
data('input_mel')
data('headneck_ref')
## Classify all cells
input_mel <- Classify(input_mel, 0)

## Classify only cells with a very high confidence
input_mel <- Classify(input_mel, 1)

## Back to the default
input_mel <- Classify(input_mel)

## Return only the classification vector
celltypes <- Classify(input_mel, 1, return_clas = TRUE)
```

---

ClassifyReference	<i>Use a reference dataset to classify itself. A good reference should have almost no mixture between reference cells.</i>
-------------------	--

---

**Description**

Use a reference dataset to classify itself. A good reference should have almost no mixture between reference cells.

**Usage**

```
ClassifyReference(
  ref_cells,
  ref_ct = "celltypes",
  ref_c = "counts",
  return = FALSE,
  ...
)
```

**Arguments**

ref_cells	the reference, similar to <a href="#">CHETAHclassifier</a> 's ref_cells
ref_ct	the colData of ref_cells where the cell types are stored.
ref_c	same as input_c, but for the reference.
return	return the matrix that was used to produce the plot
...	Other variables to pass to <a href="#">CHETAHclassifier</a>

**Value**

A square plot. The rows are the original cell types, the columns the classification labels. The colors and sizes of the squares indicate which part of the cells of the rowname type are classified to the type of the column name. On the left of the plot, the percentage of cells that is classified to an intermediate type is plotted. A good reference would classify nearly 100

**Examples**

```
data('headneck_ref')
ClassifyReference(ref_cells = headneck_ref)
```

---

CorrelateReference	<i>Correlate all reference profiles to each other using differentially expressed genes.</i>
--------------------	---

---

**Description**

Correlate all reference profiles to each other using differentially expressed genes.

**Usage**

```
CorrelateReference(
  ref_cells = NULL,
  ref_profiles = NULL,
  ref_ct = "celltypes",
  ref_c = NA,
  return = FALSE,
  n_genes = 200,
  fix_ngenes = TRUE,
  print_steps = FALSE,
  only_pos = FALSE
)
```

**Arguments**

ref_cells	the reference, similar to <a href="#">CHETAHclassifier</a> 's ref_cells
ref_profiles	similar to <a href="#">CHETAHclassifier</a> 's ref_profiles
ref_ct	the colData of ref_cells where the cell types are stored.
ref_c	the assay of ref_cells to use
return	return the matrix that was used to produce the plot
n_genes	as in <a href="#">CHETAHclassifier</a>
fix_ngenes	as in <a href="#">CHETAHclassifier</a>
print_steps	as in <a href="#">CHETAHclassifier</a>
only_pos	as in <a href="#">CHETAHclassifier</a>

**Value**

A square plot. The values show how much two reference profiles correlate, when using the genes with the highest fold-change.

**Examples**

```
data('headneck_ref')
CorrelateReference(ref_cells = headneck_ref)
```

---

headneck_ref	<i>A SingleCellExperiment with celltypes in the "celltypes" colData. A subset of the Head-Neck data from Puram et al. (2017) Cancer Cell.</i>
--------------	---

---

**Description**

A SingleCellExperiment with celltypes in the "celltypes" colData. A subset of the Head-Neck data from Puram et al. (2017) Cancer Cell.

**Usage**

```
data('headneck_ref')
```

**Format**

A list of expression matrices. Each object is named as the cell type of the cells in that matrix. Each matrix has the cell (names) in the columns and the genes in the rows.

**Source**

for the original data: [GEO](#)

**References**

Puram et al. (2017) Cancer Cell 171:1611-1624

---

input_mel	<i>A SingleCellExperiment on which CHEATHclassifier is run using the <a href="#">headneck_ref</a> It holds subset of the Melanoma data, from Tirosh et al. (2016), Science.</i>
-----------	---

---

**Description**

A SingleCellExperiment on which CHEATHclassifier is run using the [headneck\\_ref](#) It holds subset of the Melanoma data, from Tirosh et al. (2016), Science.

**Usage**

```
data('input_mel')
```

**Format**

This is a SingleCellExperiment

**Source**

for the original data: [GEO](#)

**References**

Tirosh et al. (2016) Science 6282:189-196



---

**PlotCHETAH***Plot the CHETAH classification on 2D visualization like t-SNE + the corresponding classification tree, colored with the same colors*

---

**Description**

Plot the CHETAH classification on 2D visualization like t-SNE + the corresponding classification tree, colored with the same colors

**Usage**

```
PlotCHETAH(  
  input,  
  redD = NA,  
  interm = FALSE,  
  return = FALSE,  
  tree = TRUE,  
  pt.size = 1,  
  return_col = FALSE,  
  col = NULL  
)
```

**Arguments**

<code>input</code>	a <code>SingleCellExperiment</code> on which <code>CHETAHclassifier</code> has been run
<code>redD</code>	the name of the reducedDim of the input to use for plotting
<code>interm</code>	color the intermediate instead of the final types
<code>return</code>	return the plot instead of printing it
<code>tree</code>	plot the tree, along with the classification
<code>pt.size</code>	the point-size of the classification plot
<code>return_col</code>	whether the colors that are used for the classification plot should be returned
<code>col</code>	custom colors for the cell types. <i>the colors should be named with the corresponding cell types</i>

**Value**

a ggplot object

**Examples**

```
data('input_mel')  
#' ## Standard plot (final types colored)  
PlotCHETAH(input = input_mel)  
  
## Intermediate types colored  
PlotCHETAH(input = input_mel, interm = TRUE)  
  
## Plot only the t-SNE plot  
PlotCHETAH(input = input_mel, tree = FALSE)
```

---

PlotTree

*Plots the chetah classification tree with nodes numbered*

---

## Description

Plots the chetah classification tree with nodes numbered

## Usage

```
PlotTree(  
  input,  
  col = NULL,  
  col_nodes = NULL,  
  return = FALSE,  
  no_bgc = FALSE,  
  plot_limits = c(-0.4, 0.1),  
  labelsize = 6  
)
```

## Arguments

input	a SingleCellExperiment on which <a href="#">CHETAHclassifier</a> has been run
col	a vector of colors, with the names of the reference cell types
col_nodes	a vector of colors, ordered for node 1 till the last node
return	instead of printing, return the ggplot object
no_bgc	remove the background color from the node numbers
plot_limits	define the Decreasing the former further is usefull when the labels are cut of the plot (default = c(-0,25, 01)).
labelsize	the size of the intermediate and leaf node labels (default = 6)

## Value

A ggplot object of the classification tree

## Examples

```
data('input_mel')  
PlotTree(input = input_mel)
```

---

**PlotTSNE***Plots a variable on a t-SNE*

---

**Description**

Plots a variable on a t-SNE

**Usage**

```
PlotTSNE(  
  toplot,  
  input,  
  redD = NA,  
  col = NULL,  
  return = FALSE,  
  limits = NULL,  
  pt.size = 1,  
  shiny = NULL,  
  y_limits = NULL,  
  x_limits = NULL,  
  legend_label = ""  
)
```

**Arguments**

<code>toplot</code>	the variable that should be plotted. Either a character vector or a factor, or a (continuous) numeric. If <code>toplot</code> is not named with the rownames of <code>redD</code> , it is assumed that the order of the two is the same.
<code>input</code>	a <code>SingleCellExperiment</code> on which <code>CHETAHclassifier</code> has been run
<code>redD</code>	the name of the reducedDim of the input to use for plotting
<code>col</code>	a vector of colors. If <code>toplot</code> is a numeric, this will become a continuous scale. <i>If <code>toplot</code> is a character vector, the colors should be named with the unique values (/levels) of <code>toplot</code></i>
<code>return</code>	instead of printing, return the ggplot object
<code>limits</code>	the limits of the continuous variable to plot. When not provided the minimal and maximal value will be used
<code>pt.size</code>	the point-size
<code>shiny</code>	Needed for the shiny application: should always be NULL
<code>y_limits</code>	the y-axis limits
<code>x_limits</code>	the x-axis limits, if NULL
<code>legend_label</code>	the label of the legend

**Value**

A ggplot object

**Examples**

```
data('input_mel')
CD8 <- assay(input_mel)['CD8A', ]
PlotTSNE(topplot = CD8, input = input_mel)
```

---

RenameBelowNode	<i>In the CHETAH classification, replace the name of a Node and all the names of the final and intermediate types under that Node.</i>
-----------------	--

---

**Description**

In the CHETAH classification, replace the name of a Node and all the names of the final and intermediate types under that Node.

**Usage**

```
RenameBelowNode(
  input,
  whichnode,
  replacement,
  nodes_exclude = NULL,
  types_exclude = NULL,
  node_only = FALSE,
  return_clas = FALSE
)
```

**Arguments**

input	a SingleCellExperiment on which <a href="#">CHETAHclassifier</a> has been run
whichnode	the number of the Node
replacement	a character vector that replaces the names under the selected Node
nodes_exclude	<i>optional</i> the names of the types that should <b>NOT</b> be replaced
types_exclude	<i>optional</i> numbers of the Nodes under the selected Node, that should <b>NOT</b> be replaced
node_only	only rename the Node itself, without affecting the types under that Node
return_clas	Instead of returning the SingleCellExperiment, only return the classification vector

**Value**

The SingleCellExperiment with the new classification or if 'return\_clas = TRUE' the classification vector.

**Examples**

```
## In the example data replace all T-cell subtypes by "T cell"
data('input_mel')
#' input_mel <- RenameBelowNode(input = input_mel, whichnode = 7, replacement = "T cell")
```

# Index

## \* datasets

headneck\_ref, 8  
input\_mel, 8

CHETAHclassifier, 2, 5–7, 9–12

CHETAHshiny, 5

Classify, 3, 5

ClassifyReference, 6

CorrelateReference, 7

hclust, 3

headneck\_ref, 8, 8

input\_mel, 8

PlotCHETAH, 9

PlotTree, 10

PlotTSNE, 11

RenameBelowNode, 12

spearman.dist, 3