Package ‘CIARA’

October 12, 2022

Type Package

Title Cluster Independent Algorithm for Rare Cell Types Identification

Version 0.1.0

Author Gabriele Lubatti

Maintainer Gabriele Lubatti

<gabriele.lubatti@helmholtz-muenchen.de>

Description Identification of markers of rare cell types by looking at genes whose expression is confined in small regions of the expression space <https://github.com/ScialdoneLab>.

License Artistic-2.0

Depends R (>= 4.0)

Imports Biobase, ggplot2, ggraph, magrittr

Suggests circlize, clustree, ComplexHeatmap, plotly, Seurat (>= 4.0),
testthat, knitr, rmarkdown

biocViews software

Config/testthat/edition 3

Encoding UTF-8

RoxygenNote 7.1.1

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2022-02-22 20:00:02 UTC

R topics documented:

CIARA *CIARA*

Description

It selects highly localized genes as specified in *CIARA_gene*, starting from genes in *background*

Usage

```
CIARA(
  norm_matrix,
  knn_matrix,
  background,
  cores_number = 1,
  p_value = 0.001,
  odds_ratio = 2,
  local_region = 1,
  approximation = FALSE
)
```

Arguments

`norm_matrix`	Norm count matrix (n_genes X n_cells).
`knn_matrix`	K-nearest neighbors matrix (n_cells X n_cells).
`background`	Vector of genes for which the function *CIARA_gene* is run.
`cores_number`	Integer.Number of cores to use.
`p_value`	p value returned by the function *fisher.test* with parameter alternative = "g"
`odds_ratio`	odds_ratio returned by the function *fisher.test* with parameter alternative = "g"
`local_region`	Integer. Minimum number of local regions (cell with its knn neighbours) where the binarized gene expression is enriched in 1.
`approximation`	Logical.For a given gene, the fisher test is run in the local regions of only the cells where the binarized gene expression is 1.

Value

Dataframe with n_rows equal to the length of *background* . Each row is the output from *CIARA_gene*.

Author(s)

Gabriele Lubatti <gabriele.lubatti@helmholtz-muenchen.de>

CIARA_gene *CIARA_gene*

Description

The gene expression is binarized (1/0) if the value in a given cell is above/below the median. Each of cell with its first K nearest neighbors defined a local region. If there are at least *local_region* enriched in 1 according to *fisher.test*, then the gene is defined as highly localized and a final p value is assigned to it. The final p value is the minimum of the p values from all the enriched local regions. If there are no enriched local regions, then the p value by default is set to 1

Usage

```
CIARA_gene(
  norm_matrix,
  knn_matrix,
  gene_expression,
  p_value = 0.001,
  odds_ratio = 2,
  local_region = 1,
  approximation = FALSE
)
```

Arguments

`norm_matrix`	Norm count matrix (n_genes X n_cells).
`knn_matrix`	K-nearest neighbors matrix (n_cells X n_cells).
`gene_expression`	numeric vector with the gene expression (length equal to n_cells). The gene expression is binarized (equal to 0/1 in the cells where the value is below/above the median)
`p_value`	p value returned by the function *fisher.test* with parameter alternative = "g"
`odds_ratio`	odds_ratio returned by the function *fisher.test* with parameter alternative = "g"
`local_region`	Integer. Minimum number of local regions (cell with its knn neighbours) where the binarized gene expression is enriched in 1.
`approximation`	Logical.For a given gene, the fisher test is run in the local regions of only the cells where the binarized gene expression is 1.

Value

List with one element corresponding to the p value of the gene.

Author(s)

Gabriele Lubatti <gabriele.lubatti@helmholtz-muenchen.de>

See Also

https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/fisher.test

---

cluster_analysis_integrate_rare
cluster_analysis_integrate_rare

Description

cluster_analysis_integrate_rare

Usage

```
cluster_analysis_integrate_rare(
  raw_counts,
  project_name,
  resolution,
  neighbors,
  max_dimension,
  feature_genes = NULL
)
```

Arguments

`raw_counts`	Raw count matrix (n_genes X n_cells).
`project_name`	Character name of the Seurat project.
`resolution`	Numeric value specifying the parameter *resolution* used in the Seurat function *FindClusters*.
`neighbors`	Numeric value specifying the parameter *k.param* in the Seurat function *FindNeighbors*
`max_dimension`	Numeric value specifying the maximum number of the PCA dimensions used in the parameter *dims* for the Seurat function *FindNeighbors*
`feature_genes`	vector of features specifying the argument *features* in the Seurat function *RunPCA*.

Value

Seurat object including raw and normalized counts matrices, UMAP coordinates and cluster result.

Author(s)

Gabriele Lubatti <gabriele.lubatti@helmholtz-muenchen.de>

See Also

https://www.rdocumentation.org/packages/Seurat/versions/4.0.1/topics/FindClusters
https://www.rdocumentation.org/packages/Seurat/versions/4.0.1/topics/FindNeighbors
https://www.rdocumentation.org/packages/Seurat/versions/4.0.1/topics/RunPCA

---

cluster_analysis_sub *cluster_analysis_sub*

Description

cluster_analysis_sub

Usage

```
cluster_analysis_sub(
  raw_counts,
  resolution,
  neighbors,
  max_dimension,
  name_cluster
)
```

Arguments

raw_counts	Raw count matrix (n_genes X n_cells).
resolution	Numeric value specifying the parameter *resolution* used in the Seurat function *FindClusters*.
neighbors	Numeric value specifying the parameter *k.param* in the Seurat function *FindNeighbors*
max_dimension	Numeric value specifying the maximum number of the PCA dimensions used in the parameter *dims* for the Seurat function *FindNeighbors*
name_cluster	Character.Name of the original cluster for which the sub clustering is done.

Value

Seurat object including raw and normalized counts matrices and cluster result.

Author(s)

Gabriele Lubatti <gabriele.lubatti@helmholtz-muenchen.de>

See Also

https://www.rdocumentation.org/packages/Seurat/versions/4.0.1/topics/RunPCA https://www.rdocumentation.org/packages/Seurat/versions/4.0.1/topics/FindVariableFeatures

---

find_resolution *find_resolution*

Description

find_resolution

Usage

```
find_resolution(seurat_object, resolution_vector)
```

Arguments

seurat_object	Seurat object as returned by *cluster_analysis_integrate_rare*
resolution_vector	vector with all values of resolution for which the Seurat function *FindClusters* is run

Value

Clustree object showing the connection between clusters obtained at different level of resolution as specified in *resolution_vector*.

Author(s)

Gabriele Lubatti <gabriele.lubatti@helmholtz-muenchen.de>

See Also

https://CRAN.R-project.org/package=clustree

get_background_full *get_background_full*

Description

get_background_full

Usage

```
get_background_full(
  norm_matrix,
  threshold = 1,
  n_cells_low = 3,
  n_cells_high = 20
)
```

Arguments

`norm_matrix`	Norm count matrix (n_genes X n_cells).
`threshold`	threshold in expression for a given gene
`n_cells_low`	minimum number of cells where a gene is expressed at a level above threshold
`n_cells_high`	maximum number of cells where a gene is expressed at a level above threshold

Value

Character vector with all genes expressed at a level higher than *threshold* in a number of cells between *n_cells* and *n_cells_high*.

Author(s)

Gabriele Lubatti <`gabriele.lubatti@helmholtz-muenchen.de`>

markers_cluster_seurat *markers_cluster_seurat*

Description

The Seurat function *FindMarkers* is used to identify general marker for each cluster (specific cluster vs all other cluster). This list of markers is then filtered keeping only the genes that appear as markers in a unique cluster.

Usage

```
markers_cluster_seurat(seurat_object, cluster, cell_names, number_top)
```

Arguments

`seurat_object`	Seurat object as returned by *cluster_analysis_sub* or by *cluster_analysis_integrate_rare*.
`cluster`	Vector of length equal to the number of cells, with cluster assignment.
`cell_names`	Vector of length equal to the number of cells, with cell names.
`number_top`	Integer. Number of top marker genes to keep for each cluster.

Value

List of three elements. The first is a vector with *number_top* marker genes for each cluster. The second is a vector with *number_top* marker genes and corresponding cluster. The third element is a vector with all marker genes for each cluster.

Author(s)

Gabriele Lubatti <`gabriele.lubatti@helmholtz-muenchen.de`>

See Also

https://www.rdocumentation.org/packages/Seurat/versions/4.0.1/topics/FindMarkers

---

`merge_cluster` *merge_cluster*

Description

merge_cluster

Usage

```
merge_cluster(old_cluster, new_cluster, max_number = NULL)
```

Arguments

`old_cluster`	original cluster assignment that need to be updated
`new_cluster`	new cluster assignment that need to be integrated with *old_cluster*.
`max_number`	Threshold in size for clusters in *new_cluster*. Only cluster with number of cells smaller than *max_number* will be integrated in *old cluster*. If *max_number* is NULL, then all the clusters in *new_cluster* are integrated in *old cluster*.

Value

Numeric vector of length equal to *old_cluster* showing the merged cluster assignment between *old cluster* and *new_cluster*.

Author(s)

Gabriele Lubatti <`gabriele.lubatti@helmholtz-muenchen.de`>

plot_balloon_marker *plot_balloon_marker*

Description

plot_balloon_marker

Usage

```
plot_balloon_marker(
  norm_counts,
  cluster,
  marker_complete,
  max_number,
  max_size = 5,
  text_size = 7
)
```

Arguments

`norm_counts`	Norm count matrix (genes X cells).
`cluster`	Vector of length equal to the number of cells, with cluster assignment.
`marker_complete`	Third element of the output list as returned by the function *markers_cluster_seurat*
`max_number`	Integer. Maximum number of markers for each cluster for which we want to plot the expression.
`max_size`	Integer. Size of the dots to be plotted.
`text_size`	Size of the text in the heatmap plot.

Value

ggplot2 object showing balloon plot.

Author(s)

Gabriele Lubatti <gabriele.lubatti@helmholtz-muenchen.de>

plot_gene *plot_gene*

Description

Cells are coloured according to the expression of *gene_id* and plotted according to *coordinate_umap*.

Usage

```
plot_gene(norm_counts, coordinate_umap, gene_id, title_name)
```

Arguments

`norm_counts`	Norm count matrix (genes X cells).
`coordinate_umap`	Data frame with dimensionality reduction coordinates. Number of rows must be equal to the number of cells
`gene_id`	Character name of the gene.
`title_name`	Character name.

Value

ggplot2 object.

Author(s)

Gabriele Lubatti <`gabriele.lubatti@helmholtz-muenchen.de`>

See Also

https://CRAN.R-project.org/package=ggplot2

plot_genes_sum *plot_genes_sum*

Description

The sum of each gene in *genes_relevant* across all cells is first normalized to 1. Then for each cell, the sum from the (normalized) genes expression is computed and shown in the output plot.

Usage

```
plot_genes_sum(coordinate_umap, norm_counts, genes_relevant, name_title)
```

Arguments

`coordinate_umap`
: Data frame with dimensionality reduction coordinates. Number of rows must be equal to the number of cells

`norm_counts` Norm count matrix (genes X cells).

`genes_relevant` Vector with gene names for which we want to visualize the sum in each cell.

`name_title` Character value.

Value

ggplot2 object.

Author(s)

Gabriele Lubatti <`gabriele.lubatti@helmholtz-muenchen.de`>

See Also

`https://CRAN.R-project.org/package=ggplot2`

`plot_heatmap_marker` *plot_heatmap_marker*

Description

plot_heatmap_marker

Usage

```
plot_heatmap_marker(
  marker_top,
  marker_all_cluster,
  cluster,
  condition,
  norm_counts,
  text_size
)
```

Arguments

`marker_top` First element returned by *markers_cluster_seurat*

`marker_all_cluster`
: Second element returned by *markers_cluster_seurat*

`cluster` Vector of length equal to the number of cells, with cluster assignment.

`condition` Vector or length equal to the number of cells, specifying the condition of the cells (i.e. batch, dataset of origin..)

`norm_counts` Norm count matrix (genes X cells).

`text_size` Size of the text in the heatmap plot.

Value

Heatmap class object.

Author(s)

Gabriele Lubatti <gabriele.lubatti@helmholtz-muenchen.de>

See Also

https://www.rdocumentation.org/packages/ComplexHeatmap/versions/1.10.2/topics/Heatmap

---

plot_interactive *plot_interactive*

Description

It shows in an interactive plot which are the highly localized genes in each cell. It is based on plotly library

Usage

```
plot_interactive(
  coordinate_umap,
  color,
  text,
  min_x = NULL,
  max_x = NULL,
  min_y = NULL,
  max_y = NULL
)
```

Arguments

`coordinate_umap`
: Data frame with dimensionality reduction coordinates. Number of rows must be equal to the number of cells

`color`
: vector of length equal to n_rows in coordinate_umap.Each cell will be coloured following a gradient according to the corresponding value of this vector.

`text`
: Character vector specifying the highly localized genes in each cell. It is the output from *selection_localized_genes*.

`min_x`
: Set the min limit on the x axis.

`max_x`
: Set the max limit on the x axis.

`min_y`
: Set the min limit on the y axis.

`max_y`
: Set the min limit on the y axis.

Value

plotly object given by *plot_ly function* (from library *plotly*).

Author(s)

Gabriele Lubatti <`gabriele.lubatti@helmholtz-muenchen.de`>

See Also

`https://plotly.com/r/`

`plot_umap` *plot_umap*

Description

plot_umap

Usage

```
plot_umap(coordinate_umap, cluster)
```

Arguments

`coordinate_umap`	Data frame with dimensionality reduction coordinates. Number of rows must be equal to the number of cells
`cluster`	Vector of length equal to the number of cells, with cluster assignment.

Value

ggplot2 object.

Author(s)

Gabriele Lubatti <`gabriele.lubatti@helmholtz-muenchen.de`>

See Also

`https://CRAN.R-project.org/package=ggplot2`

selection_localized_genes *selection_localized_genes*

Description

selection_localized_genes

Usage

```
selection_localized_genes(
  norm_counts,
  localized_genes,
  min_number_cells = 4,
  max_number_genes = 10
)
```

Arguments

`norm_counts` Norm count matrix (genes X cells).

`localized_genes` vector of highly localized genes as provided by the last element of the list given as output from *CIARA_mixing_final*.

`min_number_cells` Minimum number of cells where a genes must be expressed (> 0).

`max_number_genes` Maximum number of genes to show for each cell in the interactive plot from *plot_interactive*.

Value

Character vector where each entry contains the name of the top *max_number_genes* for the corresponding cell.

Author(s)

Gabriele Lubatti <gabriele.lubatti@helmholtz-muenchen.de>

`test_hvg` *test_hvg*

Description

For each cluster in *cluster*, HVGs are defined with Seurat function *FindVariableFeatures*. A Fisher test is performed to see if there is a statistically significant enrichment between the top *number_hvg* and the *localized_genes*

Usage

```
test_hvg(
  raw_counts,
  cluster,
  localized_genes,
  background,
  number_hvg,
  min_p_value
)
```

Arguments

`raw_counts`	Raw count matrix (n_genes X n_cells).
`cluster`	Vector of length equal to the number of cells, with cluster assignment.
`localized_genes`	Character vector with localized genes detected by CIARA.
`background`	Character vector with all the genes names to use as background for the Fisher test.
`number_hvg`	Integer value. Number of top HVGs provided by the Seurat function *FindVariableFeatures*.
`min_p_value`	Threshold on p values provided by Fisher test.

Value

A list with two elements.

`first` element	The first one is a list with length equal to the number of clusters. Each entry is list of three elements. The first two elements contain the p value and the odds ration given by the Fisher test The third is a vector with genes names that are present both in *localized_genes* and in top *number_hvg* HVGs .
`second` element	a character vector with the name of the cluster that have a p value smaller than *min_p_value*.

Author(s)

Gabriele Lubatti <`gabriele.lubatti@helmholtz-muenchen.de`>

See Also

https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/fisher.test

---

`white_black_markers` *white_black_markers*

Description

A white-marker is a gene whose median expression across cells belong to *single_cluster* is greater than *threshold* and in all the other clusters is equal to zero.

Usage

```
white_black_markers(
  cluster,
  single_cluster,
  norm_counts,
  marker_list,
  threshold = 0
)
```

Arguments

`cluster`	Vector of length equal to the number of cells, with cluster assignment.
`single_cluster`	Character. Label of one specify cluster
`norm_counts`	Norm count matrix (genes X cells).
`marker_list`	Third element of the output list as returned by the function *markers_cluster_seurat*
`threshold`	Numeric. The median of the genes across cells belong to *single_cluster* has to be greater than *threshold* in order to be consider as a white-black marker for *single_cluster*

Value

Logical vector of length equal to *marker_list*, with TRUE/FALSE if the gene is/is not a white-black marker for *single_cluster*.

Author(s)

Gabriele Lubatti <gabriele.lubatti@helmholtz-muenchen.de>

Index