Package 'cellmarkeraccordion'

December 9, 2024

Title Annotation of normal and aberrant hematopoietic cell types in single-cell datasets

Version 0.9.5

Description cellmarkeraccordion is an R package designed to obtain a robust identification of normal and aberrant hematopoietic cell types in single-cell datasets and easy interpretation of the results. The cellmarkeraccordion package allows to automatically annotate normal and disease ritical cell populations based on the built-in Accordion gene marker database. It requires in input only the counts matrix (raw or normalized) or a Seurat object. In addition, by exploiting the built-in cell cycle labeling you can easily assign the cell cycle phase to each cell.
The users can also customize the annotation by simply providing their own genes list as input, which can be associated with cell types or even to specific pathways or signatures of interest.
Importantly, the Accordion implements novel options to explore annotation results by inspecting for each group of cells the top marker genes which mostly impacted the annotation, together with the top cell types and their relationship based on the cell ontology tree.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Depends R (>= 4.0.0)

Imports scales (>= 1.2.1),
plyr (>= 1.8.8),
data.table (>= 1.12.0),
Seurat (>= 4.0.0),
ggplot2 (>= 3.0.0),
stringr (>= 1.5.0),
ontologyIndex (>= 2.10),
igraph (>= 1.4.2),
ontologyPlot (>= 1.6),
ggraph (>= 2.1.0),
cowplot(>= 1.1.0),
ggnewscale (>= 0.4.0),
purrr (>= 1.0.0)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

R topics documented:

`accordion` *Automatically annotating and interpreting single-cell populations with the built-in Cell Marker Accordion database*

Description

This function performs cell types annotation exploiting the built-in Accordion gene marker database. It takes in input either a Seurat object or a raw or normalized count matrix and return in output the cell types assignment and the detailed informations of the annotation results (added to the Seurat object or as a list).

Usage

```
accordion(
  data,
  cluster_info = "seurat_clusters",
  assay = "RNA",
  CL_celltypes = NULL,
  species = "Human",
  tissue = NULL,
  include_descendants = FALSE,
  evidence_consistency_score_threshold = NULL,
  specificity_score_threshold = NULL,
  log2FC_threshold = NULL,
  min_n_marker = 5,
  max_n_marker = NULL,
  combined_score_quantile_threshold = NULL,
  annotation_resolution = "cluster",
  cluster_score_quantile_threshold = 0.75,
  allow_unknown = TRUE,
  annotation_name = "accordion",
  include_detailed_annotation_info = TRUE,
  condition_group_info = NULL,
  CL_celltype_group_info = NULL,
  group_markers_by = "celltype_cluster",
  top_cell_score_quantile_threshold = 0.9,
  n_top_celltypes = 5,
  n_top_markers = 5,
  top_marker_score_quantile_threshold = 0.75,
  plot = TRUE,
```

```
  color_by = "cell_type"
)
```

Arguments

`data`	Either a Seurat object (version 4 or 5) or a raw or normalized count matrix with genes on rows and cells on columns. If raw counts are provided, data are log-normalized exploiting the NormalizeData() function from the Seurat package.
`cluster_info`	in case `object` is a Seurat object, `cluster_info` should be need to be a character string specifying the name of the column in the metadata that contains cluster ids; if `object` is a count matrix, `cluster_info` should be need to be a data frame or data table containing cluster identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "cluster", which specifies the clustering id's for each cell. This parameter is necessary only when the input is a count matrix and only if the `annotation_resolution` parameter is set to "cluster". Default is "seurat_clusters".
`assay`	Character string specifying the Assay of the Seurat object. This parameter is necessary only in case `data` is a Seurat object. Default is "RNA".
`CL_celltypes`	Character string or character string vector specifying the cell types to annotate. Run the function "list_celltypes()" to obtain the available cell types. If this parameter is not specified, all cell types present in the Accordion database are used for the annotation. Default is NULL.
`species`	Character string or character string vector specifying the species. Currently, either "Human" and/or "Mouse" are supported. If multiple species are selected, marker genes are merged together. Default is "Human".
`tissue`	Character string or character string vector specifying the tissue. Run the function "list_tissues()" to obtain the available tissues. If NULL, all tissues information are aggregated together. Default is NULL.
`evidence_consistency_score_threshold`	Integer value (currently in 1,7) specifying the minimum evidence consistency (EC) score for each marker. Only markers >= this threshold are kept. If NULL, no filter is applied. Default is NULL.
`specificity_score_threshold`	numeric value in (0,1] specifying the minimum specificity score for each marker. Only markers <= this threshold are kept. If NULL, no filter is applied. Default is NULL.
`log2FC_threshold`	numeric value specifying the minimum log2FC threshold for each marker reporting this information. Only markers <= this threshold or without any log2FC are kept. If NULL, no filter is applied. Default is NULL.
`min_n_marker`	Integer value specifying the minimum number of markers to keep for each cell type. Only cell types with a number of markers >= this threshold are kept. Default is 5.
`max_n_marker`	Integer value specifying the maximum number of markers to keep for each cell type. For the selection, markers are ranked according to their combined score, obtained by multiplying evidence consistency score and specificity score. If NULL, no filter is applied. Default is NULL.
`combined_score_quantile_threshold`	numeric value in (0,1] specifying the combined score quantile threshold. For the selection, markers are ranked according to their combined score, obtained

by multiplying evidence consistency score and specificity score. Only markers > the quantile_threshold are kept. If NULL, no filter is applied. Default is NULL.

`annotation_resolution`
: Character string or character string vector specifying the resolution of the annotation. Either "cluster" and/or "cell" are supported. Default is "cluster".

`cluster_score_quantile_threshold`
: numeric value in 0,1 specifying the cluster score quantile threshold. For each cell a score specific for each cell type is computed. To annotate a cluster cl, for each cell type the `cluster_score_quantile_threshold` is computed across cells belonging to that cluster and the cell type with the maximum score is then assigned to the cluster cl. Default is 0.75.

`allow_unknown`
: Logical value indicating whether to allow cells or clusters to be labeled as "unknown". If it is set to TRUE, cells or clusters with negative scores are assigned to the "unknown" category. Default is TRUE.

`annotation_name`
: Character string specifying the name of the column in either the metadata of the input Seurat object or in the input `cluster_info` where the annotation will be stored. Per cluster and per cell annotation results will be stored in the `annotation_name_per_cluster` and `annotation_name_per_cell` columns respectively. If `include_detailed_annotation_info` parameter is set to TRUE, the detailed information the stored in a list named `annotation_name`. Default is "accordion".

`include_detailed_annotation_info`
: Logical value indicating whether to store information on the top cell types and markers in the output. If TRUE, a nested list named `annotation_name` is created. If `resolution_annotation` is set to "cluster" and/or "cell, sublists named "cluster_resolution" and/or "cell_resolution" are then added. Inside the sublist "detailed_annotation_info" the `n_top_markers` markers, group by `group_markers_by` and the `n_top_celltypes` cell types are then included. If a Seurat object is provided as input the list is stored in the misc slot of the object (`object@misc@annotation_name`). If the input is a count matrix, the list is returned in the final output. Default is TRUE.

`condition_group_info`
: in case `object` is a Seurat object, `condition_group_info` should be need to be a character string specifying the name of the column in the metadata that contains condition ids for each cell; if `object` is a count matrix, `condition_group_info` should be need to be a data frame or data table containing condition identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "condition", which specifies the condition id's for each cell. Default is NULL.

`CL_celltype_group_info`
: in case `object` is a Seurat object, `CL_celltype_group_info` should be need to be a character string specifying the name of the column in the metadata that contains cell types ids for each cell; if `object` is a count matrix, `CL_celltype_group_info` should be need to be a data frame or data table containing cell types identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "CL_celltype", which specifies the cell types for each cell. Default is NULL.

`group_markers_by`
: Character string or character string vector specifying the classification of marker genes. It possible to retrieve `n_top_markers` marker genes for each cell type

identified with cluster ("celltype_cluster") or cell ("celltype_cell") resolution; n_top_markers marker genes per cluster ("cluster") or per cell ("cell") can be also obtained. Additionally, by setting group_markers_by to "score_cell", the n_top_markers marker genes only for cells with a score greater than top_cell_score_quantile_th are retrieved. Either "celltype_cluster", "celltype_cell","cluster", "cell" or "score_cell". Default is "celltype_cluster".

top_cell_score_quantile_threshold
: numeric value in (0,1] specifying the cell score quantile threshold. For each cell type a score specific for each cell is computed. The top_cell_score_quantile_threshold is computed across cells belonging to the same cell type, and only cells with a score greater than the top_cell_score_quantile_threshold are kept. This parameter is necessary only when group_markers_by is set to "score_cell". Default is 0.90.

n_top_celltypes
: Integer value specifying the number of the top cell types to be included in the output for each cluster and cell depending on the selected annotation_resolution parameter Default is 5.

n_top_markers
: Integer value specifying the number of the top markers to be included in the output for each cell type, cluster or cell depending on the selected annotation_resolution and group_markers_by parameters. Default is 5.

top_marker_score_quantile_threshold
: numeric value in (0,1] specifying the marker score quantile threshold. For each marker a score specific for each cell is computed. To identify the n_top_markers for a cluster cl or a cell type ct, the top_marker_score_quantile_threshold is computed across cells belonging to that cluster or labeled as ct, and the n_top_markers with the maximum score are reported. Default is 0.75.

plot
: Logical value indicating whether to store plots displaying detailed annotation information. This parameter can be set to TRUE only when include_detailed_annotation_info is set to TRUE. If TRUE, lollipop plots displaying the top n_top_markers group by group_markers_by and top n_top_celltypes for each annotation_resolution together with the cell types hierarchies based on the cell ontology structure are stored in the annotation_name list. Default is TRUE.

color_by
: Character string specifying if the plot reporting the top cell types for each cluster/cell is colored based on the assigned cell type ("CL_celltype") or on cluster id ("cluster"). Default is "CL_celltype.

Details

If a Seurat object was provided in input, the function returns the Seurat object with markers-based scaled data in the scale.data slot and cell types annotation results in the metadata. If include_detailed_annotation_in and plot were set to TRUE, a list containing cell types and markers information, together with ggplot objects, is stored in the "misc@annotation_name" slot. If a count matrix was provided in input, the function returns a list containing the following elements:

- "scaled_matrix":normalized and scaled expression matrix;

If annotation_resolution is set to "cell":

- "cell_annotation":data table containing cell types annotation results for each cell;

If annotation_resolution is set to "cluster":

- "cluster_annotation":data table containing cell types annotation results for each cell;

If include_detailed_annotation_info is set to TRUE:

- "annotation_name":list containing detailed information of cell types annotation.

Value

A Seurat object or a list

---

accordion_cell_cycle *Automatically identify and interpreting cell cycle state of single-cell populations*

Description

This function identifies cell cycle states exploiting the collection of marker genes associated to each phase, including G0. It takes in input either a Seurat object or a raw or normalized count matrix and return in output the cell cycle assignment and the detailed informations of the annotation results (added to the Seurat object or as a list).

Usage

```
accordion_cell_cycle(
  data,
  cluster_info = "seurat_clusters",
  assay = "RNA",
  species = "Human",
  annotation_resolution = "cell",
  annotation_name = "accordion_cell_cycle",
  cluster_score_quantile_threshold = 0.75,
  allow_unknown = FALSE,
  include_detailed_annotation_info = FALSE,
  condition_group_info = NULL,
  cell_type_group_info = NULL,
  group_markers_by = "celltype_cell",
  n_top_celltypes = 5,
  n_top_markers = 5,
  top_marker_score_quantile_threshold = 0.75,
  plot = FALSE
)
```

Arguments

data	Either a Seurat object (version 4 or 5) or a raw or normalized count matrix with genes on rows and cells on columns. If raw counts are provided, data are log-normalized exploiting the NormalizeData() function from the Seurat package.
cluster_info	in case object is a Seurat object, cluster_info should be need to be a character string specifying the name of the column in the metadata that contains cluster ids; if object is a count matrix, cluster_info should be need to be a data frame or data table containing cluster identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "cluster", which specifies the clustering id's for

each cell. This parameter is necessary only when the input is a count matrix and only if the annotation_resolution parameter is set to "cluster". Default is "seurat_clusters".

assay Character string specifying the Assay of the Seurat object. This parameter is necessary only in case data is a Seurat object. Default is "RNA".

species Character string or character string vector specifying the species. Currently, either "Human" and/or "Mouse" are supported. If multiple species are selected, marker genes are merged together. Default is "Human".

annotation_resolution
Character string or character string vector specifying the resolution of the annotation. Either "cluster" and/or "cell" are supported. Default is "cell".

annotation_name
Character string specifying the name of the column in either the metadata of the input Seurat object or in the input cluster_info where the annotation will be stored. Per cluster and per cell annotation results will be stored in the annotation_name_per_cluster and annotation_name_per_cell columns respectively. If include_detailed_annotation_info parameter is set to TRUE, the detailed information the stored in a list named annotation_name. Default is "accordion_cell_cycle".

cluster_score_quantile_threshold
numeric value in 0,1 specifying the cluster score quantile threshold. For each cell a score specific for each cell type is computed. To annotate a cluster cl, for each cell type the cluster_score_quantile_threshold is computed across cells belonging to that cluster and the cell type with the maximum score is then assigned to the cluster cl. Default is 0.75.

allow_unknown Logical value indicating whether to allow cells or clusters to be labeled as "unknown". If it is set to TRUE, cells or clusters with negative scores are assigned to the "unknown" category. Default is TRUE.

include_detailed_annotation_info
Logical value indicating whether to store information on the top cell types and markers in the output. If TRUE, a nested list named annotation_name is created. If resolution_annotation is set to "cluster" and/or "cell, sublists named "cluster_resolution" and/or "cell_resolution" are then added. Inside the sublist "detailed_annotation_info" the n_top_markers markers, group by group_markers_by and the n_top_celltypes cell types are then included. If a Seurat object is provided as input the list is stored in the misc slot of the object (object@misc@annotation_name). If the input is a count matrix, the list is returned in the final output. Default is FALSE. @param condition_group_info in case object is a Seurat object, condition_group_info should be need to be a character string specifying the name of the column in the metadata that contains condition ids for each cell; if object is a count matrix, condition_group_info should be need to be a data frame or data table containing condition identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "condition", which specifies the condition id's for each cell. Default is NULL. @param cell_type_group_info in case object is a Seurat object, cell_type_group_info should be need to be a character string specifying the name of the column in the metadata that contains cell types ids for each cell; if object is a count matrix, cell_type_group_info should be need to be a data frame or data table containing cell types identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "cell_type", which specifies the cell types for each cell. Default is NULL.

group_markers_by
: Character string or character string vector specifying the classification of marker genes. It possible to retrieve n_top_markers marker genes for each cell type identified with cluster ("celltype_cluster") or cell ("celltype_cell") resolution; n_top_markers marker genes per cluster ("cluster") or per cell ("cell") can be also obtained. Either "celltype_cluster", "celltype_cell", "cluster" and/or "cell". Default is "celltype_cell".

n_top_celltypes
: Integer value specifying the number of the top cell types to be included in the output for each cluster and cell depending on the selected annotation_resolution parameter Default is 5.

n_top_markers
: Integer value specifying the number of the top markers to be included in the output for each cell type, cluster or cell depending on the selected annotation_resolution and group_markers_by parameters. Default is 5.

top_marker_score_quantile_threshold
: numeric value in (0,1] specifying the marker score quantile threshold. For each marker a score specific for each cell is computed. To identify the n_top_markers for a cluster cl or a cell type ct, the top_marker_score_quantile_threshold is computed across cells belonging to that cluster or labeled as ct, and the n_top_markers with the maximum score are reported. Default is 0.75.

plot
: Logical value indicating whether to store plots displaying detailed annotation information. This parameter can be set to TRUE only when include_detailed_annotation_info is set to TRUE. If TRUE, lollipop plots displaying the top n_top_markers group by group_markers_by and n_top_celltypes for each annotation_resolution together with the cell types hierarchies based on the cell ontology structure are stored in the annotation_name list. Default is FALSE.

Details

If a Seurat object was provided in input, the function returns the Seurat object with markers-based scaled data in the scale.data slot and cell types annotation results in the metadata. If include_detailed_annotation_in and plot were set to TRUE, a list named annotation_name containing cell types and markers information, together with ggplot objects, is stored in the "misc" slot. If a count matrix was provided in input, the function returns a list containing the following elements:

- "scaled_matrix":normalized and scaled expression matrix;

If annotation_resolution is set to "cell":

- "cell_annotation":data table containing cell types annotation results for each cell;

If annotation_resolution is set to "cluster":

- "cluster_annotation":data table containing cell types annotation results for each cell;

If include_detailed_annotation_info is set to TRUE:

- "annotation_name":list containing detailed information of cell types annotation.

Value

A Seurat object or a list

accordion_custom *Automatically annotating and interpreting single-cell populations with custom marker genes sets*

Description

This function performs cell types or signatures/pathways annotation based on cusom marker genes set. It takes in input either a Seurat object or a raw or normalized count matrix and a table of marker genes associated to cell types or even to pathways and return in output the cell types/pathways assignment and the detailed informations of the annotation results (added to the Seurat object or as a list).

Usage

```
accordion_custom(
  data,
  marker_table,
  category_column = "cell_type",
  marker_column = "marker",
  marker_type_column = "marker_type",
  weight_column = "weight",
  cluster_info = "seurat_clusters",
  assay = "RNA",
  min_n_marker = 5,
  max_n_marker = NULL,
  combined_score_quantile_threshold = NULL,
  annotation_resolution = "cluster",
  cluster_score_quantile_threshold = 0.75,
  allow_unknown = TRUE,
  annotation_name = "accordion_custom",
  include_detailed_annotation_info = TRUE,
  condition_group_info = NULL,
  cell_type_group_info = NULL,
  group_markers_by = "celltype_cluster",
  top_cell_score_quantile_threshold = 0.9,
  n_top_celltypes = 5,
  n_top_markers = 5,
  top_marker_score_quantile_threshold = 0.75,
  plot = TRUE
)
```

Arguments

data	Either a Seurat object (version 4 or 5) or a raw or normalized count matrix with genes on rows and cells on columns. If raw counts are provided, data are log-normalized exploiting the NormalizeData() function from the Seurat package.
marker_table	Data table or data frame containing cell type markers. The table needs to have at least two columns, the category_column, which specifies cell types or categories, and the marker_column, which specifies the corresponding markers on each row. Columns indicating the marker type (either positive or negative), and the marker weight can be optionally included.

`category_column`
: String characters specifying the name of the `marker_table` column containing cell types or categories. Default is "cell_type".

`marker_column`
: String characters specifying the name of the `marker_table` column containing markers. Default is "marker".

`marker_type_column`
: Optional string characters specifying the name of the `marker_table` column containing string characters indicating the type of markers, either "positive" or "negative". If no `marker_type_column` is found in the `marker_table` all markers are considered "positive". Default is "marker_type".

`weight_column`
: Optional string characters specifying the name of the `marker_table` column containing numeric value indicating the weight for each marker. If no `weight_column` is found in the `marker_table` all markers are equally weighted as 1. Default is "weight".

`cluster_info`
: in case `object` is a Seurat object, `cluster_info` should be need to be a character string specifying the name of the column in the metadata that contains cluster ids; if `object` is a count matrix, `cluster_info` should be need to be a data frame or data table containing cluster identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "cluster", which specifies the clustering id's for each cell. This parameter is necessary only when the input is a count matrix and only if the `annotation_resolution` parameter is set to "cluster". Default is "seurat_clusters".

`assay`
: Character string specifying the Assay of the Seurat object. This parameter is necessary only in case `data` is a Seurat object. Default is "RNA".

`min_n_marker`
: Integer value specifying the minimum number of markers to keep for each cell type. Only cell types with a number of markers >= this threshold are kept. Default is 5.

`max_n_marker`
: Integer value specifying the maximum number of markers to keep for each cell type. For the selection, markers are ranked according to their combined score, obtained by multiplying evidence consistency score and specificity score. If NULL, no filter is applied. Default is NULL.

`combined_score_quantile_threshold`
: numeric value in (0,1] specifying the combined score quantile threshold. For the selection, markers are ranked according to their combined score, obtained by multiplying weight and specificity score. Only markers > the quantile_threshold are kept. If NULL, no filter is applied. Default is NULL.

`annotation_resolution`
: Character string or character string vector specifying the resolution of the annotation. Either "cluster" and/or "cell" are supported. Default is "cluster".

`cluster_score_quantile_threshold`
: numeric value in 0,1 specifying the cluster score quantile threshold. For each cell a score specific for each cell type is computed. To annotate a cluster cl, for each cell type the `cluster_score_quantile_threshold` is computed across cells belonging to that cluster and the cell type with the maximum score is then assigned to the cluster cl. Default is 0.75.

`allow_unknown`
: Logical value indicating whether to allow cells or clusters to be labeled as "unknown". If it is set to TRUE, cells or clusters with negative scores are assigned to the "unknown" category. Default is TRUE.

annotation_name
: Character string specifying the name of the column in either the metadata of the input Seurat object or in the input cluster_info where the annotation will be stored. Per cluster and per cell annotation results will be stored in the annotation_name_per_cluster and annotation_name_per_cell columns respectively. If include_detailed_annotation_info parameter is set to TRUE, the detailed information the stored in a list named annotation_name. Default is "accordion_custom".

include_detailed_annotation_info
: Logical value indicating whether to store information on the top cell types and markers in the output. If TRUE, a list named annotation_name is created. If resolution_annotation is set to "cluster" and/or "cell, sublists named "cluster_resolution" and/or "cell_resolution" are then added. Inside the sublist "detailed_annotation_info" the n_top_markers markers, group by group_markers_by and the n_top_celltypes cell types are then included. If a Seurat object is provided as input the list is stored in the misc slot of the object (object@misc@annotation_name). If the input is a count matrix, the list is returned in the final output. Default is TRUE.

condition_group_info
: in case object is a Seurat object, condition_group_info should be need to be a character string specifying the name of the column in the metadata that contains condition ids for each cell; if object is a count matrix, condition_group_info should be need to be a data frame or data table containing condition identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "condition", which specifies the condition id's for each cell. Default is NULL.

cell_type_group_info
: in case object is a Seurat object, cell_type_group_info should be need to be a character string specifying the name of the column in the metadata that contains cell types ids for each cell; if object is a count matrix, cell_type_group_info should be need to be a data frame or data table containing cell types identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "cell_type", which specifies the cell types for each cell. Default is NULL.

group_markers_by
: Character string or character string vector specifying the classification of marker genes. It possible to retrieve n_top_markers marker genes for each cell type identified with cluster ("celltype_cluster") or cell ("celltype_cell") resolution; n_top_markers marker genes per cluster ("cluster") or per cell ("cell") can be also obtained. Additionally, by setting group_markers_by to "score_cell", the n_top_markers marker genes only for cells with a score greater than top_cell_score_quantile_th are retrieved. Either "celltype_cluster", "celltype_cell", "cluster", "cell" or "score_cell". Default is "celltype_cluster".

top_cell_score_quantile_threshold
: numeric value in (0,1] specifying the cell score quantile threshold. For each cell type/signature a score specific for each cell is computed. The top_cell_score_quantile_threshol is computed across cells belonging to the same cell type/signature, and only cells with a score greater than the top_cell_score_quantile_threshold are kept. This parameter is necessary only when group_markers_by is set to "score_cell". Default is 0.90.

n_top_celltypes
: Integer value specifying the number of the top cell types to be included in the

output for each cluster and cell depending on the selected annotation_resolution parameter Default is 5.

n_top_markers Integer value specifying the number of the top markers to be included in the output for each cell type, cluster or cell depending on the selected annotation_resolution and group_markers_by parameters. Default is 5.

top_marker_score_quantile_threshold
numeric value in (0,1] specifying the marker score quantile threshold. For each marker a score specific for each cell is computed. To identify the n_top_markers for a cluster cl or a cell type ct, the top_marker_score_quantile_threshold is computed across cells belonging to that cluster or labeled as ct, and the n_top_markers with the maximum score are reported. Default is 0.75.

plot Logical value indicating whether to store plots displaying detailed annotation information. This parameter can be set to TRUE only when include_detailed_annotation_info is set to TRUE. If TRUE, lollipop plots displaying the top n_top_markers group by group_markers_by and op n_top_celltypes for each annotation_resolution together with the cell types hierarchies based on the cell ontology structure are stored in the annotation_name list. Default is TRUE.

Details

If a Seurat object was provided in input, the function returns the Seurat object with markers-based scaled data in the scale.data slot and cell types annotation results in the metadata. If include_detailed_annotation_in and plot were set to TRUE, a list containing cell types and markers information, together with ggplot objects, is stored in the "misc@annotation_name" slot. If a count matrix was provided in input, the function returns a list containing the following elements:

- "scaled_matrix":normalized and scaled expression matrix;

If annotation_resolution is set to "cell":

- "cell_annotation":data table containing cell types annotation results for each cell;

If annotation_resolution is set to "cluster":

- "cluster_annotation":data table containing cell types annotation results for each cell;

If include_detailed_annotation_info is set to TRUE:

- "annotation_name":list containing detailed information of cell types annotation.

Value

A Seurat object or a list

`accordion_disease` *Automatically annotating and interpreting aberrant single-cell populations with the built-in Cell Marker Accordion disease database*

Description

This function identified aberrant populations exploiting the built-in Accordion gene marker disease database. It takes in input either a Seurat object or a raw or normalized count matrix and return in output the cell types assignment and the detailed informations of the annotation results (added to the Seurat object or as a list).

Usage

```
accordion_disease(
  data,
  disease = NULL,
  tissue = NULL,
  cluster_info = "seurat_clusters",
  assay = "RNA",
  NCIT_celltypes = NULL,
  species = "Human",
  include_descendants = FALSE,
  evidence_consistency_score_threshold = NULL,
  specificity_score_threshold = NULL,
  log2FC_threshold = NULL,
  malignant_quantile_threshold = 0.95,
  min_n_marker = 5,
  max_n_marker = NULL,
  combined_score_quantile_threshold = NULL,
  disease_vs_healthy = TRUE,
  annotation_resolution = "cluster",
  cluster_score_quantile_threshold = 0.75,
  allow_unknown = TRUE,
  annotation_name = "accordion_disease",
  include_detailed_annotation_info = TRUE,
  condition_group_info = NULL,
  NCIT_celltype_group_info = NULL,
  group_markers_by = "celltype_cluster",
  top_cell_score_quantile_threshold = 0.9,
  n_top_celltypes = 5,
  n_top_markers = 5,
  top_marker_score_quantile_threshold = 0.75,
  plot = TRUE,
  color_by = "cell_type"
)
```

Arguments

`data`	Either a Seurat object (version 4 or 5) or a raw or normalized count matrix with genes on rows and cells on columns. If raw counts are provided, data are log-normalized exploiting the NormalizeData() function from the Seurat package.

`disease` Character string or character string vector specifying diseases to consider. Run the function "list_diseases()" to obtain the available diseases. If NULL, all diseases are considered. Default is NULL.

`tissue` Character string or character string vector specifying the tissue. Run the function "list_disease_tissues()" to obtain the available tissues. If NULL, all tissues information are aggregated together. Default is NULL.

`cluster_info` in case `object` is a Seurat object, `cluster_info` should be need to be a character string specifying the name of the column in the metadata that contains cluster ids; if `object` is a count matrix, `cluster_info` should be need to be a data frame or data table containing cluster identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "cluster", which specifies the clustering id's for each cell. This parameter is necessary only when the input is a count matrix and only if the `annotation_resolution` parameter is set to "cluster". Default is "seurat_clusters".

`assay` Character string specifying the Assay of the Seurat object. This parameter is necessary only in case `data` is a Seurat object. Default is "RNA".

`NCIT_celltypes` Character string or character string vector specifying the cell types to annotate. Run the function "list_aberrant_celltypes()" to obtain the available aberrant cell types. If this parameter is not specified, all aberrant cell types are used for the annotation. Default is NULL.

`species` Character string or character string vector specifying the species. Currently, either "Human" and/or "Mouse" are supported. If multiple species are selected, marker genes are merged together. Default is "Human".

`evidence_consistency_score_threshold`
Integer value (currently in 1,7) specifying the minimum evidence consistency (EC) score for each marker. Only markers >= this threshold are kept. If NULL, no filter is applied. Default is NULL.

`specificity_score_threshold`
numeric value in (0,1] specifying the minimum specificity score for each marker. Only markers <= this threshold are kept. If NULL, no filter is applied. Default is NULL.

`min_n_marker` Integer value specifying the minimum number of markers to keep for each cell type. Only cell types with a number of markers >= this threshold are kept. Default is 5.

`max_n_marker` Integer value specifying the maximum number of markers to keep for each cell type. For the selection, markers are ranked according to their combined score, obtained by multiplying evidence consistency score and specificity score. If NULL, no filter is applied. Default is NULL.

`combined_score_quantile_threshold`
numeric value in (0,1] specifying the combined score quantile threshold. For the selection, markers are ranked according to their combined score, obtained by multiplying evidence consistency score and specificity score. Only markers > the quantile_threshold are kept. If NULL, no filter is applied. Default is NULL.

`disease_vs_healthy`
Logical value indicating whether to compare the markers associated with disease cell types with respect to the markers associated with the corresponding healthy cell types. If TRUE the specificity score is calculated considering if a gene is also a marker for the corresponding healthy cell type. For each cell, a specific

score for disease cell types and another score for the healthy types (named as the cell types label) are returned. Default is TRUE.

`annotation_resolution`
: Character string or character string vector specifying the resolution of the annotation. Either "cluster" and/or "cell" are supported. Default is "cluster".

`cluster_score_quantile_threshold`
: numeric value in 0,1 specifying the cluster score quantile threshold. For each cell a score specific for each cell type is computed. To annotate a cluster cl, for each cell type the `cluster_score_quantile_threshold` is computed across cells belonging to that cluster and the cell type with the maximum score is then assigned to the cluster cl. Default is 0.75.

`allow_unknown`
: Logical value indicating whether to allow cells or clusters to be labeled as "unknown". If it is set to TRUE, cells or clusters with negative scores are assigned to the "unknown" category. Default is TRUE.

`annotation_name`
: Character string specifying the name of the column in either the metadata of the input Seurat object or in the input `cluster_info` where the annotation will be stored. Per cluster and per cell annotation results will be stored in the `annotation_name_per_cluster` and `annotation_name_per_cell` columns respectively. If `include_detailed_annotation_info` parameter is set to TRUE, the detailed information the stored in a list named `annotation_name`. Default is "accordion_disease".

`include_detailed_annotation_info`
: Logical value indicating whether to store information on the top cell types and markers in the output. If TRUE, a nested list named `annotation_name` is created. If `resolution_annotation` is set to "cluster" and/or "cell, sublists named "cluster_resolution" and/or "cell_resolution" are then added. Inside the sublist "detailed_annotation_info" the `n_top_markers` markers, group by `group_markers_by` and the `n_top_celltypes` cell types are then included. If a Seurat object is provided as input the list is stored in the misc slot of the object (object@misc@`annotation_name`). If the input is a count matrix, the list is returned in the final output. Default is TRUE.

`condition_group_info`
: in case `object` is a Seurat object, `condition_group_info` should be need to be a character string specifying the name of the column in the metadata that contains condition ids for each cell; if `object` is a count matrix, `condition_group_info` should be need to be a data frame or data table containing condition identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "condition", which specifies the condition id's for each cell. Default is NULL.

`NCIT_celltype_group_info`
: in case `object` is a Seurat object, `NCIT_celltype_group_info` should be need to be a character string specifying the name of the column in the metadata that contains cell types ids for each cell; if `object` is a count matrix, `NCIT_celltype_group_info` should be need to be a data frame or data table containing cell types identity for each cell. The data frame or data table should contain at least two columns, one named "cell", which specifies cell id's, and one named "NCIT_celltype", which specifies the cell types for each cell. Default is NULL.

`group_markers_by`
: Character string or character string vector specifying the classification of marker genes. It possible to retrieve `n_top_markers` marker genes for each cell type

identified with cluster ("celltype_cluster") or cell ("celltype_cell") resolution; n_top_markers marker genes per cluster ("cluster") or per cell ("cell") can be also obtained. Additionally, by setting group_markers_by to "score_cell", the n_top_markers marker genes only for cells with a score greater than top_cell_score_quantile_th are retrieved. Either "celltype_cluster", "celltype_cell", "cluster", "cell" or "score_cell". Default is "celltype_cluster".

top_cell_score_quantile_threshold
: numeric value in (0,1] specifying the cell score quantile threshold. For each cell type a score specific for each cell is computed. The top_cell_score_quantile_threshold is computed across cells belonging to the same cell type, and only cells with a score greater than the top_cell_score_quantile_threshold are kept. This parameter is necessary only when group_markers_by is set to "score_cell". Default is 0.90.

n_top_celltypes
: Integer value specifying the number of the top cell types to be included in the output for each cluster and cell depending on the selected annotation_resolution parameter Default is 5.

n_top_markers
: Integer value specifying the number of the top markers to be included in the output for each cell type, cluster or cell depending on the selected annotation_resolution and group_markers_by parameters. Default is 5.

top_marker_score_quantile_threshold
: numeric value in (0,1] specifying the marker score quantile threshold. For each marker a score specific for each cell is computed. To identify the n_top_markers for a cluster cl or a cell type ct, the top_marker_score_quantile_threshold is computed across cells belonging to that cluster or labeled as ct, and the n_top_markers with the maximum score are reported. Default is 0.75.

plot
: Logical value indicating whether to store plots displaying detailed annotation information. This parameter can be set to TRUE only when include_detailed_annotation_info is set to TRUE. If TRUE, lollipop plots displaying the top n_top_markers group by group_markers_by and op n_top_celltypes for each annotation_resolution together with the cell types hierarchies based on the cell ontology structure are stored in the "accordion" list. Default is TRUE.

Details

If a Seurat object was provided in input, the function returns the Seurat object with markers-based scaled data in the scale.data slot and cell types annotation results in the metadata. If include_detailed_annotation_in and plot were set to TRUE, a list containing cell types and markers information, together with ggplot objects, is stored in the "misc@annotation_name" slot. If a count matrix was provided in input, the function returns a list containing the following elements:

- "scaled_matrix":normalized and scaled expression matrix;

If annotation_resolution is set to "cell":

- "cell_annotation":data table containing cell types annotation results for each cell;

If annotation_resolution is set to "cluster":

- "cluster_annotation":data table containing cell types annotation results for each cell;

If include_detailed_annotation_info is set to TRUE:

- "annotation_name":list containing detailed information of cell types annotation.

Value

A Seurat object or a list

`accordion_plot` *Interpreting annotation results*

Description

This function generates lollipop plots displaying the detailed annotation results obtained with the `accordion_annotation`, `accordion_disease_annotation` and `accordion_custom_annotation` functions.

Usage

```
accordion_plot(
  data,
  info_to_plot = "accordion",
  resolution = "cluster",
  group_markers_by = "celltype_cluster",
  color_by = "cell_type"
)
```

Arguments

`data`	A Seurat object or a list containing "detailed_annotation_info", from either accordion(), accordion_disease() or accordion_custom() functions, run with include_detailed_annotation_info parameter set to TRUE.
`info_to_plot`	Character string or character string vector specifying the list from which extract the detailed annotation information, either "accordion", "accordion_disease" or "accordion_custom", for which returns the plot, either "accordion", "accordion_disease" or "accordion_custom".
`resolution`	Character string or character string vector specifying the annotation resolution for which provided the plots. Either "cluster" and/or "cell" are supported. Default is "cluster".
`group_markers_by`	Character string or character string vector specifying the classification of marker genes. It is possible to retrieve top marker genes for each cell type identified with cluster ("celltype_cluster") or cell ("celltype_cell") resolution; top marker genes per cluster ("cluster") or per cell ("cell") can be also obtained. Additionally, by setting `group_markers_by` to "score_cell", the `n_top_markers` marker genes only for cells with a score greater than `top_cell_score_quantile_threshold` are retrieved. Either "celltype_cluster", "celltype_cell", "cluster", "cell" or "score_cell". Default is "celltype_cluster".
`color_by`	Character string specifying if the plot reporting the top cell types for each cluster/cell is colored based on the assigned cell type ("cell_type") or on cluster id ("cluster"). Default is "cell_type.

Details

top cell types (or pathways) and top markers It takes in input either a Seurat object or a raw or normalized count matrix and a table of marker genes associated to cell types or even to pathways and return in output the cell types/pathways assignment and the detailed informations of the annotation results (added to the Seurat object or as a list).

If a Seurat object was provided in input, the function returns the Seurat object with a list of ggplot objects added to the "misc" slot in the `info_to_plot` list. If a list was provided in input, the function returns the same list with the addition of the ggplot objects.

Value

A Seurat object or a list.

`list_aberrant_celltypes` *List aberrant cell types available in the Cell Marker Accordion disease database*

Description

List aberrant cell types available in the Cell Marker Accordion disease database

Usage

```
list_aberrant_celltypes(
  species = c("Human", "Mouse"),
  disease = NULL,
  tissue = NULL
)
```

Arguments

`species`	Character string or character string vector specifying the species for which to extract the associate list of available cell types. Currently, either “Human” and/or “Mouse” are supported. Default is c("Mouse",“Human”). @param disease Character string or character string vector specifying diseases to consider. If NULL, information from all diseases are considered. Default is NULL. @param tissue Character string or character string vector specifying the tissue for which to extract the associate list of available cell types. If NULL, information from all tissues are retrieved.

Value

List of aberrant cell types available in the Cell Marker Accordion disease database

list_celltypes *List cell types available in the Cell Marker Accordion database*

Description

List cell types available in the Cell Marker Accordion database

Usage

```
list_celltypes(species = c("Human", "Mouse"), tissue = NULL)
```

Arguments

species	Character string or character string vector specifying the species for which to extract the associate list of available cell types. Currently, either "Human" and/or "Mouse" are supported. Default is c("Mouse","Human"). @param tissue Character string or character string vector specifying the tissue for which to extract the associate list of available cell types. If NULL, information from all tissues are retrieved.

Value

List of cell types available in the Cell Marker Accordion database

list_diseases *List diseases available in the Cell Marker Accordion disease database*

Description

List diseases available in the Cell Marker Accordion disease database

Usage

```
list_diseases(
  species = c("Human", "Mouse"),
  tissue = NULL,
  aberrant_celltype = NULL
)
```

Arguments

species	Character string or character string vector specifying the species for which to extract the associate list of available diseases Currently, either "Human" and/or "Mouse" are supported. Default is c("Mouse","Human"). @param tissue Character string or character string vector specifying the tissue for which to extract the associate list of available diseases. If NULL, information from all tissues are retrieved. @param aberrant_celltype Character string or character string vector specifying the aberrant celltype for which to extract the associate list of available tissues. If NULL, information from all aberrant cell types are retrieved. @return List of diseases available in the Cell Marker Accordion disease database

list_disease_tissues *List tissues available in the Cell Marker Accordion disease database*

Description

List tissues available in the Cell Marker Accordion disease database

Usage

```
list_disease_tissues(
  species = c("Human", "Mouse"),
  disease = NULL,
  aberrant_celltype = NULL
)
```

Arguments

species	Character string or character string vector specifying the species for which to extract the associate list of available tissues. Currently, either "Human" and/or "Mouse" are supported. Default is c("Mouse","Human"). @param disease Character string or character string vector specifying diseases to consider. If NULL, information from all diseases are considered. Default is NULL. @param celltype Character string or character string vector specifying the celltype for which to extract the associate list of available tissues. If NULL, information from all cell types are retrieved.

Value

List of tissues available in the Cell Marker Accordion disease database

list_tissues *List tissues available in the Cell Marker Accordion database*

Description

List tissues available in the Cell Marker Accordion database

Usage

```
list_tissues(species = c("Human", "Mouse"), celltype = NULL)
```

Arguments

species	Character string or character string vector specifying the species for which to extract the associate list of available tissues. Currently, either "Human" and/or "Mouse" are supported. Default is c("Mouse","Human"). @param celltype Character string or character string vector specifying the celltype for which to extract the associate list of available tissues. If NULL, information from all cell types are retrieved.

Value

List of tissues available in the Cell Marker Accordion database

Index