Package 'SingleCellComplexHeatMap'

June 20, 2025

Type Package

Title Complex Heatmaps for Single Cell Expression Data with Dual Information Display

Version 0.1.1

Maintainer XueCheng Fang <xuecheng328@gmail.com>

Description Creates complex heatmaps for single cell RNA-seq data that simultaneously display gene expression levels (as color intensity) and expression percentages (as circle sizes). Supports gene grouping, cell type annotations, and time point comparisons. Built on top of 'ComplexHeatmap' and integrates with 'Seurat' objects. For more details see Gu (2022) <doi:10.1002/imt2.43> and Hao (2024) <doi:10.1038/s41587-023-01767-y>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Depends R (>= 4.0.0)

Imports ComplexHeatmap (>= 2.10.0), Seurat (>= 4.0.0), dplyr (>= 1.0.0), tidyr (>= 1.0.0), RColorBrewer, circlize (>= 0.4.0), grid, grDevices, stats, magrittr

Suggests testthat (>= 3.0.0), knitr, rmarkdown, viridis, devtools, BiocManager

URL https://github.com/FanXuRong/SingleCellComplexHeatMap

BugReports https://github.com/FanXuRong/SingleCellComplexHeatMap/issues

NeedsCompilation no

Author XueCheng Fang [aut, cre]

R topics documented:

`create_cell_annotations` *Create Cell Type and Time Point Annotations for Heatmap Columns*

Description

Parses column names to extract time points and cell types, creates annotations and reorders matrices.

Usage

```
create_cell_annotations(
  exp_mat,
  percent_mat,
  split_pattern = "_",
  time_position = 1,
  celltype_start = 2,
  time_points_order = NULL,
  cell_types_order = NULL,
  time_color_palette = "Accent",
  celltype_color_palette = "Dark2",
  show_time_annotation = TRUE,
  show_celltype_annotation = TRUE
)
```

Arguments

`exp_mat`	Expression matrix with samples as columns
`percent_mat`	Percentage matrix with samples as columns
`split_pattern`	Character string used to split column names (default: "_")
`time_position`	Integer indicating position of time point in split names (default: 1)
`celltype_start`	Integer indicating starting position of cell type in split names (default: 2)
`time_points_order`	Character vector specifying order of time points (default: NULL for automatic)
`cell_types_order`	Character vector specifying order of cell types (default: NULL for automatic)
`time_color_palette`	Character string specifying RColorBrewer palette for time points (default: "Accent")
`celltype_color_palette`	Character string specifying RColorBrewer palette for cell types (default: "Dark2")
`show_time_annotation`	Logical indicating whether to show time point annotation (default: TRUE)
`show_celltype_annotation`	Logical indicating whether to show cell type annotation (default: TRUE)

Value

A list containing exp_mat_ordered (reordered expression matrix), percent_mat_ordered (reordered percentage matrix), col_annotation (ComplexHeatmap column annotation object), col_split_factor (factor for column splitting based on time points), and annotation_df (data frame with column annotations).

See Also

create_single_cell_complex_heatmap, prepare_expression_matrices

Examples

```
## Not run:
# Prepare expression matrices first
matrices <- prepare_expression_matrices(seurat_obj, features, group_by = "timepoint_cellt

# Create cell annotations with custom ordering
col_annotations <- create_cell_annotations(
  exp_mat = matrices$exp_mat,
  percent_mat = matrices$percent_mat,
  split_pattern = "_",
  time_points_order = c("0h", "6h", "12h", "24h"),
  cell_types_order = c("T_cell", "B_cell", "Monocyte"),
  time_color_palette = "Set2",
  celltype_color_palette = "Dark2"
)

# Access results
ordered_exp_mat <- col_annotations$exp_mat_ordered
col_annotation <- col_annotations$col_annotation
col_split_factor <- col_annotations$col_split_factor

## End(Not run)
```

---

create_gene_annotations

Create Gene Group Annotations for Heatmap Rows

Description

Creates gene grouping annotations and reorders expression matrices based on gene classifications.

Usage

```
create_gene_annotations(
  exp_mat,
  percent_mat,
  gene_classification,
  color_palette = "Set1",
  sort_within_groups = TRUE
)
```

Arguments

exp_mat	Expression matrix with genes as rows
percent_mat	Percentage matrix with genes as rows
gene_classification	Named list where names are group labels and values are character vectors of gene names
color_palette	Character string specifying RColorBrewer palette name (default: "Set1")
sort_within_groups	Logical indicating whether to sort genes within each group (default: TRUE)

Value

A list containing exp_mat_ordered (reordered expression matrix), percent_mat_ordered (reordered percentage matrix), row_annotation (ComplexHeatmap row annotation object), row_split_factor (factor for row splitting), and annotation_df (data frame with gene annotations).

See Also

create_single_cell_complex_heatmap, prepare_expression_matrices

Examples

```
## Not run:
# Prepare expression matrices first
matrices <- prepare_expression_matrices(seurat_obj, features, group_by = "cell_type")

# Define gene groups
gene_groups <- list(
  "Immune_Response" = c("Gene1", "Gene2"),
  "Cell_Cycle" = c("Gene3", "Gene4"),
  "Metabolism" = c("Gene5", "Gene6")
)

# Create gene annotations
annotations <- create_gene_annotations(
  exp_mat = matrices$exp_mat,
  percent_mat = matrices$percent_mat,
  gene_classification = gene_groups,
  color_palette = "Set1"
)

# Access results
ordered_exp_mat <- annotations$exp_mat_ordered
ordered_percent_mat <- annotations$percent_mat_ordered
row_annotation <- annotations$row_annotation

## End(Not run)
```

`create_single_cell_complex_heatmap` *Create Complex Heatmap for Single Cell Expression Data*

Description

Creates a complex heatmap that displays both gene expression levels (as color intensity) and expression percentages (as circle sizes) for single cell RNA-seq data. This function provides extensive customization options while maintaining ease of use.

Usage

```
create_single_cell_complex_heatmap(
  seurat_object,
  features,
  gene_classification = NULL,
  group_by = "seurat_clusters",
  idents = NULL,
  time_points_order = NULL,
  cell_types_order = NULL,
  color_range = c(-1, 0, 2),
  color_palette = NULL,
  max_circle_size = 2,
  row_fontsize = 8,
  col_fontsize = 9,
  col_name_rotation = 90,
  row_title_fontsize = 10,
  col_title_fontsize = 10,
  show_heatmap_legend = TRUE,
  show_percentage_legend = TRUE,
  legend_side = "right",
  cell_border_color = "grey80",
  split_pattern = "_",
  gene_color_palette = "Set1",
  time_color_palette = "Accent",
  celltype_color_palette = "Dark2",
  show_gene_grouping = NULL,
  show_time_annotation = TRUE,
  show_celltype_annotation = TRUE,
  split_by = "time",
  merge_legends = TRUE,
  percentage_legend_title = "Expression %",
  percentage_legend_labels = c("0%", "25%", "50%", "75%", "100%"),
  percentage_breaks = NULL,
  return_data = FALSE,
  save_plot = NULL,
  plot_width = 10,
  plot_height = 8,
  plot_dpi = 300,
  assay = NULL,
  slot = "scale.data",
```

```
  cluster_cells = TRUE,
  cluster_features = TRUE,
  clustering_distance_rows = "euclidean",
  clustering_distance_cols = "euclidean",
  clustering_method_rows = "complete",
  clustering_method_cols = "complete",
  color_palette_main = c("blue", "white", "red"),
  annotation_colors = NULL,
  show_feature_names = TRUE,
  feature_names_gp = NULL,
  legend_title = "Expression",
  ...
)
```

Arguments

seurat_object	A Seurat object containing single cell data
features	Character vector of gene names to plot
gene_classification	Named list where names are group labels and values are character vectors of gene names (default: NULL for no gene grouping)
group_by	Character string specifying the metadata column to group by (default: "seurat_clusters")
idents	Numeric or character vector specifying which cell groups to include (default: NULL for all)
time_points_order	Character vector specifying order of time points. Only affects display order, not data filtering (default: NULL for automatic)
cell_types_order	Character vector specifying order of cell types. Only affects display order, not data filtering (default: NULL for automatic)
color_range	Numeric vector specifying color mapping break points for expression values. Its length must match color_palette if color_palette is a vector. (default: c(-1, 0, 2))
color_palette	Character vector specifying colors for expression heatmap. Its length must match color_range. If NULL, a default palette (viridis or color_palette_main) is generated to match color_range length (default: NULL)
max_circle_size	Numeric specifying maximum circle radius in mm. This applies to the highest percentage value in percentage_breaks (default: 2)
row_fontsize	Numeric specifying row name font size (default: 8)
col_fontsize	Numeric specifying column name font size (default: 9)
col_name_rotation	Numeric specifying column name rotation angle (default: 90)
row_title_fontsize	Numeric specifying row title font size (default: 10)
col_title_fontsize	Numeric specifying column title font size (default: 10)

`show_heatmap_legend`
: Logical indicating whether to show heatmap legend (default: TRUE)

`show_percentage_legend`
: Logical indicating whether to show percentage legend (default: TRUE)

`legend_side`
: Character string specifying legend position (default: "right")

`cell_border_color`
: Character string specifying cell border color (default: "grey80")

`split_pattern`
: Character string used to split column names for parsing (default: "_")

`gene_color_palette`
: Character string specifying RColorBrewer palette for gene groups (default: "Set1")

`time_color_palette`
: Character string specifying RColorBrewer palette for time points (default: "Accent")

`celltype_color_palette`
: Character string specifying RColorBrewer palette for cell types (default: "Dark2")

`show_gene_grouping`
: Logical indicating whether to show gene grouping (default: TRUE if gene_classification provided)

`show_time_annotation`
: Logical indicating whether to show time point annotation (default: TRUE)

`show_celltype_annotation`
: Logical indicating whether to show cell type annotation (default: TRUE)

`split_by`
: Character string specifying how to split columns: "time", "celltype", or "none" (default: "time")

`merge_legends`
: Logical indicating whether to merge legends (default: TRUE)

`percentage_legend_title`
: Character string for percentage legend title (default: "Expression %")

`percentage_legend_labels`
: Character vector for percentage legend labels

`percentage_breaks`
: Numeric vector specifying actual percentage values corresponding to labels

`return_data`
: Logical; if TRUE, return underlying data instead of drawing only

`save_plot`
: File path to save the drawn heatmap (PNG)

`plot_width`
: Numeric; width in inches for saving

`plot_height`
: Numeric; height in inches for saving

`plot_dpi`
: Numeric; resolution (DPI) for saved plot

`assay`
: Seurat assay name to extract data from

`slot`
: Seurat slot name within assay (e.g., "scale.data", "data")

`cluster_cells`
: Logical; whether to cluster columns (cells)

`cluster_features`
: Logical; whether to cluster rows (features)

`clustering_distance_rows`
: Distance metric for row clustering

`clustering_distance_cols`
: Distance metric for column clustering

`clustering_method_rows`
: Clustering method for rows

`clustering_method_cols`
: Clustering method for columns

`color_palette_main`
: Fallback color palette when viridis unavailable

`annotation_colors`
: Named list of custom annotation colors

`show_feature_names`
: Logical; whether to show feature (row) names

`feature_names_gp`
: gpar object controlling feature name appearance

`legend_title` Character; title for main heatmap legend

`...` Additional arguments passed to ComplexHeatmap::Heatmap()

Value

A ComplexHeatmap object. If return_data is TRUE, returns a list containing the heatmap object and underlying data matrices.

`prepare_expression_matrices`
Prepare Expression and Percentage Matrices from Seurat DotPlot

---

Description

Extracts and reshapes expression data from a Seurat DotPlot object into matrices suitable for complex heatmap visualization.

Usage

```
prepare_expression_matrices(
  seurat_object,
  features,
  group_by = "seurat_clusters",
  idents = NULL,
  split_pattern = "_",
  time_position = 1,
  celltype_start = 2
)
```

Arguments

`seurat_object`
: A Seurat object containing single cell data

`features` Character vector of gene names to plot

`group_by` Character string specifying the metadata column to group by (default: "seurat_clusters")

`idents` Numeric or character vector specifying which cell groups to include (default: NULL for all)

`split_pattern` Character string used to split column names for parsing (default: "_")

`time_position` Integer indicating position of time point in split names (default: 1)

`celltype_start` Integer indicating starting position of cell type in split names (default: 2)

Value

A list containing exp_mat (matrix of scaled expression values), percent_mat (matrix of expression percentages), and dotplot_data (original DotPlot data frame).

See Also

`create_single_cell_complex_heatmap`

Examples

```
## Not run:
# Basic usage
matrices <- prepare_expression_matrices(
  seurat_object = my_seurat,
  features = c("Gene1", "Gene2", "Gene3"),
  group_by = "cell_type"
)

# Advanced usage with specific cell groups
matrices <- prepare_expression_matrices(
  seurat_object = my_seurat,
  features = gene_list,
  group_by = "timepoint_celltype",
  idents = c("0h_T_cell", "6h_T_cell", "12h_T_cell"),
  split_pattern = "_"
)

# Access the results
expression_matrix <- matrices$exp_mat
percentage_matrix <- matrices$percent_mat
original_data <- matrices$dotplot_data

## End(Not run)
```