Package ‘scCustomize’

December 23, 2022

Type  Package
Title  Custom Visualizations & Functions for Streamlined Analyses of Single Cell Sequencing
Description  Collection of functions created and/or curated to aid in the visualization and analysis of single-cell data using ‘R’. ‘scCustomize’ aims to provide 1) Customized visualizations for aid in ease of use and to create more aesthetic and functional visuals. 2) Improve speed/reproducibility of common tasks/pieces of code in scRNA-seq analysis with a single or group of functions. For citation please use: Marsh SE (2021) “Custom Visualizations & Functions for Streamlined Analyses of Single Cell Sequencing” <doi:10.5281/zenodo.5706430>.

Version  1.1.0
Date  2022-12-22

URL  https://github.com/samuel-marsh/scCustomize,
     https://samuel-marsh.github.io/scCustomize/,
     https://doi.org/10.5281/zenodo.5706431

BugReports  https://github.com/samuel-marsh/scCustomize/issues

Depends  R (>= 4.0.0), Seurat (>= 4.3.0)
Imports  circlize, cli (>= 3.2.0), cowplot, data.table, dplyr, forcats, ggbeeswarm, ggplot2, ggprism, ggrastr, ggrepel, glue, grDevices, grid, janitor, lifecycle, magrittr, Matrix (>= 1.5.0), methods, paletteer, patchwork, pbapply, purrr, rlang (>= 1.0.1), scales, scattermore (>= 0.7), SeuratObject (>= 4.1.2), stats, stringi, stringr, tibble, tidyr, viridis

Suggests  ComplexHeatmap, dittoSeq, DropletUtils, ggpubr, hdf5r, knitr, Nebulosa, remotes, rlieger, rmarkdown, tidyselect, tidyverse, qs

License  GPL (>= 3)
Encoding  UTF-8
LazyData  true
RoxygenNote  7.2.3

NeedsCompilation  no
Author Samuel Marsh [aut, cre] (<https://orcid.org/0000-0002-3012-6945>),
Ming Tang [ctb],
Velina Kozareva [ctb],
Lucas Graybuck [ctb]
Maintainer Samuel Marsh <samuel.marsh@childrens.harvard.edu>
Repository CRAN
Date/Publication 2022-12-23 19:10:02 UTC

R topics documented:

Add_CellBender_Diff .................................................. 4
Add_Cell_Complexity_LIGER ........................................ 5
Add_Cell_Complexity_Seurat ...................................... 6
Add_Mito_Ribo_LIGER .............................................. 7
Add_Mito_Ribo_Seurat .............................................. 8
Add_Pct.Diff .......................................................... 10
Add_Sample_Meta .................................................... 11
Blank_Theme .......................................................... 12
Case_Check ............................................................ 12
CellBender_Diff_Plot .............................................. 13
CellBender_Feature_Diff ......................................... 15
Cell_Highlight_Plot ................................................. 16
Change_Delim_All .................................................. 17
Change_Delim_Prefix ............................................... 18
Change_Delim_Suffix ............................................... 19
CheckMatrix_scCustom ........................................... 19
Clustered_DotPlot ................................................ 20
Cluster_Highlight_Plot ............................................ 23
Cluster_Stats_All_Samples ...................................... 24
ColorBlind_Pal ....................................................... 25
Copy_From_GCP ...................................................... 25
Copy_To_GCP ........................................................ 26
Create_10X_H5 ....................................................... 27
Create_CellBender_Merged_Seurat .............................. 28
Create_Cluster_Annotation_File ................................ 29
Dark2_Pal ............................................................. 29
DimPlot_All_Samples ............................................... 30
DimPlot_LIGER ........................................................ 31
DimPlot_scCustom .................................................. 33
DiscretePalette_scCustomize .................................. 35
DotPlot_scCustom .................................................. 36
ensembl_mito_id ..................................................... 37
ensembl_ribo_id ..................................................... 38
Extract_Sample_Meta ............................................... 38
Extract_Top_Markers ............................................... 39
FeaturePlot_DualAssay ........................................... 41
FeaturePlot_scCustom ............................................. 42
R topics documented:

Fetch_Meta ......................................................... 44
Gene_Present ..................................................... 45
Hue_Pal ............................................................. 46
Iterate_Cluster_Highlight_Plot ................................. 46
Iterate_DimPlot_bySample ....................................... 48
Iterate_FeaturePlot_scCustom ................................. 49
Iterate_Meta_Highlight_Plot ................................. 51
Iterate_PC_Loading_Plots ....................................... 52
Iterate_Plot_Density_Custom ................................. 53
Iterate_Plot_Density_Joint ..................................... 55
Iterate_VlnPlot_scCustom ..................................... 56
JCO_Four ........................................................ 58
Liger_to_Seurat .................................................. 58
Median_Stats ..................................................... 60
Merge_Seurat_List ............................................... 61
Merge_Sparse_Data_All ........................................ 62
Meta_Highlight_Plot ............................................ 63
Meta_NUMERIC ................................................... 64
Meta_Present ..................................................... 65
Meta_Present_LIGER ........................................... 66
Meta_Remove_Seurat ........................................... 66
Move_Legend .................................................... 67
NavyAndOrange ................................................... 68
PalettePlot ....................................................... 68
PC_Plotting ........................................................ 69
Percent_Expressing ............................................. 70
plotFactors_scCustom ......................................... 71
Plot_Cells_per_Sample ......................................... 73
Plot_Density_Custom ........................................... 74
Plot_Density_Joint_Only ....................................... 75
Plot_Median_Genes ............................................. 76
Plot_Median_Mito .............................................. 77
Plot_Median_Other ............................................ 78
Plot_Median_UMIs ............................................. 79
Pull_Cluster_Annotation ....................................... 80
Pull_Directory_List ............................................. 81
QC_Plots_Combined_Vln ....................................... 82
QC_Plots_Complexity .......................................... 83
QC_Plots_Feature .............................................. 85
QC_Plots_Genes ................................................ 86
QC_Plots_Mito ................................................... 88
QC_Plots_UMIs .................................................. 89
QC_Plot_GenevsFeature ....................................... 91
QC_Plot_UMIvsFeature ....................................... 92
QC_Plot_UMIvsGene ........................................... 94
Read10X_GEO ..................................................... 96
Read10X_h5_GEO ................................................ 97
Read10X_h5_Multi_Directory .................................. 98
Add_CellBender_Diff

Description

Calculate and add differences post-cell bender analysis

Calculate the difference in features and UMIs per cell when both cell bender and raw assays are present.
**Add_Cell_Complexity_LIGER**

**Usage**

Add_CellBender_Diff(seurat_object, raw_assay_name, cell_bender_assay_name)

**Arguments**

seurat_object  object name.

raw_assay_name  name of the assay containing the raw data.

cell_bender_assay_name  name of the assay containing the Cell Bender'ed data.

**Value**

Seurat object with 2 new columns in the meta.data slot.

**Examples**

```r
## Not run:
object <- Add_CellBender_Diff(seurat_object = obj, raw_assay_name = "RAW",
cell_bender_assay_name = "RNA")
## End(Not run)
```

---

**Add_Cell_Complexity_LIGER**

*Add Cell Complexity Value*

**Description**

Add measure of cell complexity/novelty (log10PerUMI) for data QC.

**Usage**

Add_Cell_Complexity_LIGER(
    liger_object,
    meta_col_name = "log10GenesPerUMI",
    overwrite = FALSE
)

**Arguments**

liger_object  object name.

meta_col_name  name to use for new meta data column. Default is "log10GenesPerUMI".

overwrite  Logical. Whether to overwrite existing an.meta.data column. Default is FALSE meaning that function will abort if column with name provided to meta_col_name is present in meta.data slot.
Value

A LIGER Object

Examples

```r
## Not run:
object <- Add_Cell_Complexity_LIGER(liger_object = object)

## End(Not run)
```

### Description

Add measure of cell complexity/novelty (log10PerUMI) for data QC.

### Usage

```r
Add_Cell_Complexity_Seurat(
  seurat_object,
  meta_col_name = "log10GenesPerUMI",
  assay = "RNA",
  overwrite = FALSE
)
```

### Arguments

- **seurat_object**: object name.
- **meta_col_name**: name to use for new meta data column. Default is "log10GenesPerUMI".
- **assay**: assay to use in calculation. Default is "RNA". Note This should only be changed if storing corrected and uncorrected assays in same object (e.g. outputs of both Cell Ranger and Cell Bender).
- **overwrite**: Logical. Whether to overwrite existing an meta.data column. Default is FALSE meaning that function will abort if column with name provided to meta_col_name is present in meta.data slot.

### Value

A Seurat Object

### Examples

```r
library(Seurat)
pbm <- Add_Cell_Complexity_Seurat(seurat_object = pbm)
```
Add Mito, Ribo, percentages to meta.data slot of LIGER Object

Usage

Add_Mito_Ribo_LIGER(
    liger_object, 
    species, 
    mito_name = "percent_mito", 
    ribo_name = "percent_ribo", 
    mito_ribo_name = "percent_mito_ribo", 
    mito_pattern = NULL, 
    ribo_pattern = NULL, 
    mito_features = NULL, 
    ribo_features = NULL, 
    ensembl_ids = FALSE, 
    overwrite = FALSE, 
    list_species_names = FALSE 
)

Arguments

liger_object  LIGER object name.
species  Species of origin for given Seurat Object. If mouse, human, marmoset, zebrfish, rat, drosophila, or rhesus macaque (name or abbreviation) are provided the function will automatically generate mito_pattern and ribo_pattern values.
mito_name  name to use for the new meta.data column containing percent mitochondrial counts. Default is "percent_mito".
ribo_name  name to use for the new meta.data column containing percent ribosomal counts. Default is "percent_mito".
mito_ribo_name  name to use for the new meta.data column containing percent mitochondrial+ribosomal counts. Default is "percent_mito".
mito_pattern  A regex pattern to match features against for mitochondrial genes (will set automatically if species is mouse or human; marmoset features list saved separately).
ribo_pattern  A regex pattern to match features against for ribosomal genes (will set automatically if species is mouse, human, or marmoset).
mito_features  A list of mitochondrial gene names to be used instead of using regex pattern. Will override regex pattern if both are present (including default saved regex patterns).
### Description

Add Mito, Ribo, & Mito+Ribo percentages to meta.data slot of Seurat Object

### Usage

```r
Add_Mito_Ribo_Seurat(
  seurat_object,
  species,
  mito_name = "percent_mito",
  ribo_name = "percent_ribo",
  mito_ribo_name = "percent_mito_ribo",
  mito_pattern = NULL,
  ribo_pattern = NULL,
  mito_features = NULL,
  ribo_features = NULL,
  ensembl_ids = FALSE,
  assay = NULL,
  overwrite = FALSE,
  list_species_names = FALSE
)
```
Add_Mito_Ribo_Seurat

Arguments

seurat_object object name.

species Species of origin for given Seurat Object. If mouse, human, marmoset, zebrafish, rat, drosophila, or rhesus macaque (name or abbreviation) are provided the function will automatically generate mito_pattern and ribo_pattern values.

mito_name name to use for the new meta.data column containing percent mitochondrial counts. Default is "percent_mito".

ribo_name name to use for the new meta.data column containing percent ribosomal counts. Default is "percent_mito".

mito_ribo_name name to use for the new meta.data column containing percent mitochondrial+ribosomal counts. Default is "percent_mito".

mito_pattern A regex pattern to match features against for mitochondrial genes (will set automatically if species is mouse or human; marmoset features list saved separately).

ribo_pattern A regex pattern to match features against for ribosomal genes (will set automatically if species is mouse, human, or marmoset).

mito_features A list of mitochondrial gene names to be used instead of using regex pattern. Will override regex pattern if both are present (including default saved regex patterns).

ribo_features A list of ribosomal gene names to be used instead of using regex pattern. Will override regex pattern if both are present (including default saved regex patterns).

ensembl_ids logical, whether feature names in the object are gene names or ensembl IDs (default is FALSE; set TRUE if feature names are ensembl IDs).

assay Assay to use (default is the current object default assay).

overwrite Logical. Whether to overwrite existing meta.data columns. Default is FALSE meaning that function will abort if columns with any one of the names provided to mito_name ribo_name or mito_ribo_name is present in meta.data slot.

list_species_names returns list of all accepted values to use for default species names which contain internal regex/feature lists (human, mouse, marmoset, zebrafish, rat, drosophila, and rhesus macaque). Default is FALSE.

Value

A Seurat Object

Examples

library(Seurat)
pbm_small <- Add_Mito_Ribo_Seurat(seurat_object = pbmc_small, species = "human")
Add_Pct_Diff

Add percentage difference to DE results

Description

Adds new column labeled "pct_diff" to the data.frame output of FindMarkers, FindAllMarkers, or other DE test data.frames.

Usage

Add_Pct_Diff(
  marker_dataframe,
  pct.1_name = "pct.1",
  pct.2_name = "pct.2",
  overwrite = FALSE
)

Arguments

marker_dataframe
data.frame containing the results of FindMarkers, FindAllMarkers, or other DE test data.frame.
pct.1_name the name of data.frame column corresponding to percent expressed in group 1. Default is Seurat default "pct.1".
pct.2_name the name of data.frame column corresponding to percent expressed in group 2. Default is Seurat default "pct.2".
overwrite logical. If the marker_dataframe already contains column named "pct_diff" whether to overwrite or return error message. Default is FALSE.

Value

Returns input marker_dataframe with additional "pct_diff" column.

Examples

## Not run:
marker_df <- FindAllMarkers(object = obj_name)
marker_df <- Add_Pct_Diff(marker_dataframe = marker_df)
# or piped with function
marker_df <- FindAllMarkers(object = obj_name) %>%
  Add_Pct_Diff()

## End(Not run)
Add_Sample_Meta

Add Sample Level Meta Data

Description

Add meta data from ample level data.frame/tibble to cell level seurat @meta.data slot

Usage

Add_Sample_Meta(
  seurat_object,
  meta_data,
  join_by_seurat,
  join_by_meta,
  na_ok = FALSE,
  overwrite = FALSE
)

Arguments

seurat_object  object name.
meta_data      data.frame/tibble containing meta data or path to file to read. Must be formatted as either data.frame or tibble.
join_by_seurat name of the column in seurat_object@meta.data that contains matching variables to join_by_meta in meta_data.
join_by_meta   name of the column in meta_data that contains matching variables to join_by_seurat in seurat_object@meta.data.
na_ok          logical, is it ok to add NA values to seurat_object@meta.data. Default is FALSE. Be very careful if setting TRUE because if there is error in join operation it may result in all @meta.data values being replaced with NA.
overwrite      logical, if there are shared columns between seurat_object@meta.data and meta_data should the current seurat_object@meta.data columns be overwritten. Default is FALSE. This parameter excludes values provided to join_by_seurat and join_by_meta.

Value

Seurat object with new @meta.data columns

Examples

## Not run:
# meta_data present in environment
sample_level_meta <- data.frame(...)  
obj <- Add_Sample_Meta(seurat_object = obj, meta_data = sample_level_meta, join_by_seurat = "orig.ident", join_by_meta = "sample_ID")
```r
# from meta data file
obj <- Add_Sample_Meta(seurat_object = obj, meta_data = "meta_data/sample_level_meta.csv",
join_by_seurat = "orig.ident", join_by_meta = "sample_ID")
## End(Not run)
```

---

**Blank Theme**

**Description**

Shortcut for thematic modification to remove all axis labels and grid lines

**Usage**

```r
Blank_THEME(...)```

**Arguments**

```
... extra arguments passed to ggplot2::theme().
```

**Value**

Returns a list-like object of class `theme`.

**Examples**

```
# Generate a plot and customize theme
library(ggplot2)
df <- data.frame(x = rnorm(n = 100, mean = 20, sd = 2), y = rbinom(n = 100, size = 100, prob = 0.2))
p <- ggplot(data = df, mapping = aes(x = x, y = y)) + geom_point(mapping = aes(color = 'red'))
p + Blank_THEME()
```

---

## Case_Check

**Description**

Check for alternate case features Checks Seurat object for the presence of features with the same spelling but alternate case.
Usage

Case_Check(
    seurat_object,
    gene_list,
    case_check_msg = TRUE,
    return_features = TRUE
)

Arguments

seurat_object Seurat object name.
gene_list vector of genes to check.
case_check_msg logical. Whether to print message to console if alternate case features are found in addition to inclusion in returned list. Default is TRUE.
return_features logical. Whether to return vector of alternate case features. Default is TRUE.

Value

If features found returns vector of found alternate case features and prints message depending on parameters specified.

Examples

```r
## Not run:
alts <- Case_Check(seurat_object = obj_name, gene_list = DEG_list)
## End(Not run)
```

---

**CellBender_Diff_Plot**  
**Plot Number of Cells/Nuclei per Sample**

Description

Plot of total cell or nuclei number per sample grouped by another meta data variable.

Usage

```r
CellBender_Diff_Plot(
    feature_diff_df,
    pct_diff_threshold = 25,
    num_features = NULL,
    label = TRUE,
    num_labels = 20,
    repel = TRUE,
    custom_labels = NULL,
)```
plot_line = TRUE,
plot_title = "Raw Counts vs. Cell Bender Counts",
x_axis_label = "Raw Data Counts",
y_axis_label = "Cell Bender Counts",
xnudge = 0,
ynudge = 0,
max.overlaps = 100,
label_color = "dodgerblue",
fontface = "bold",
label_size = 3.88,
bg.color = "white",
bg.r = 0.15,
...)

Arguments

feature_diff_df
    name of data.frame created using CellBender_Feature_Diff.
pct_diff_threshold
    threshold to use for feature plotting. Resulting plot will only contain features which exhibit percent change >= value. Default is 25.
num_features
    Number of features to plot. Will ignore pct_diff_threshold and return plot with specified number of features. Default is NULL.
label
    logical, whether or not to label the features that have largest percent difference between raw and CellBender counts (Default is TRUE).
num_labels
    Number of features to label if label = TRUE, (default is 20).
repel
    logical, whether to use geom_text_repel to create a nicely-repelled labels; this is slow when a lot of points are being plotted. If using repel, set xnudge and ynudge to 0, (Default is TRUE).
custom_labels
    A custom set of features to label instead of the features most different between raw and CellBender counts.
plot_line
    logical, whether to plot diagonal line with slope = 1 (Default is TRUE).
plot_title
    Plot title.
x_axis_label
    Label for x axis.
y_axis_label
    Label for y axis.
xnudge
    Amount to nudge X and Y coordinates of labels by.
ynudge
    Amount to nudge X and Y coordinates of labels by.
max.overlaps
    passed to geom_text_repel, exclude text labels that overlap too many things. Defaults to 100.
label_color
    Color to use for text labels.
fontface
    font face to use for text labels ("plain", “bold”, “italic”, “bold.italic”) (Default is "bold").
label_size
    text size for feature labels (passed to geom_text_repel).
CellBender_Feature_Diff

bg.color color to use for shadow/outline of text labels (passed to `geom_text_repel`) (Default is white).

bg.r radius to use for shadow/outline of text labels (passed to `geom_text_repel`) (Default is 0.15).

Extra parameters passed to `geom_text_repel` through `LabelPoints`.

Value

A ggplot object

Examples

```r
## Not run:
# get cell bender differences data.frame
cb_stats <- CellBender_Feature_Diff(seurat_object - obj, raw_assay = "RAW",
cell_bender_assay = "RNA")

# plot
CellBender_Diff_Plot(feature_diff_df = cb_stats, pct_diff_threshold = 25)

## End(Not run)
```

---

CellBender_Feature_Diff

CellBender Feature Differences

Description

Get quick values for raw counts, CellBender counts, count differences, and percent count differences per feature.

Usage

```r
CellBender_Feature_Diff(seurat_object, raw_assay, cell_bender_assay)
```

Arguments

- `seurat_object` Seurat object name.
- `raw_assay` Name of the assay containing the raw count data.
- `cell_bender_assay` Name of the assay containing the CellBender count data.

Value

A data.frame containing summed raw counts, CellBender counts, count difference, and percent difference in counts.
Examples

## Not run:
```
cb_stats <- CellBender_Feature_Diff(seurat_object - obj, raw_assay = "RAW",
                     cell_bender_assay = "RNA")
## End(Not run)
```

---

**Cell_Highlight_Plot**

**Description**

Create Plot with meta data variable of interest highlighted

**Usage**

```
Cell_Highlight_Plot(
  seurat_object,  # Seurat object name.
  cells_highlight,  # Cell names to highlight in named list.
  highlight_color = NULL,  # Color to highlight cells.
  background_color = "lightgray",  # non-highlighted cell colors (default is "lightgray").
  pt.size = NULL,  # point size for both highlighted cluster and background.
  raster = NULL,  # Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.
  raster.dpi = c(512, 512),  # Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).
  ...  #
)
```

**Arguments**

- `seurat_object` Seurat object name.
- `cells_highlight` Cell names to highlight in named list.
- `highlight_color` Color to highlight cells.
- `background_color` non-highlighted cell colors (default is "lightgray").
- `pt.size` point size for both highlighted cluster and background.
- `raster` Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.
- `raster.dpi` Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).
label Whether to label the highlighted meta data variable(s). Default is FALSE.

split.by Variable in @meta.data to split the plot by.

split_seurat logical. Whether or not to display split plots like Seurat (shared y axis) or as individual plots in layout. Default is FALSE.

ggplot_default_colors logical. If highlight_color = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.

... Extra parameters passed to DimPlot.

Value

A ggplot object

Examples

library(Seurat)

# Creating example non-overlapping vectors of cells
MS4A1 <- WhichCells(object = pbmc_small, expression = MS4A1 > 4)
GZMB <- WhichCells(object = pbmc_small, expression = GZMB > 4)

# Format as named list
cells <- list("MS4A1" = MS4A1,
             "GZMB" = GZMB)

Cell_Highlight_Plot(seurat_object = pbmc_small, cells_highlight = cells)

Change_Delim_All (Change all delimiters in cell name)

Description

Change all instances of delimiter in cell names from list of data.frames/matrices or single data.frame/matrix

Usage

Change_Delim_All(data, current_delim, new_delim)

Arguments

data Either matrix/data.frame or list of matrices/data.frames with the cell barcodes in the column names.
current_delim a single value of current delimiter.
new_delim a single value of new delimiter desired.
Value
matrix or data.frame with new column names.

Examples
```r
## Not run:
dge_matrix <- Change_Delim_All(data = dge_matrix, current_delim = ".", new_delim = "-")
## End(Not run)
```

---

**Change_Delim_Prefix** *Change barcode prefix delimiter*

Description
Change barcode prefix delimiter from list of data.frames/matrices or single data.frame/matrix

Usage
`Change_Delim_Prefix(data, current_delim, new_delim)`

Arguments
- `data` Either matrix/data.frame or list of matrices/data.frames with the cell barcodes in the column names.
- `current_delim` a single value of current delimiter.
- `new_delim` a single value of new delimiter desired.

Value
matrix or data.frame with new column names.

Examples
```r
## Not run:
dge_matrix <- Change_Delim_Prefix(data = dge_matrix, current_delim = ".", new_delim = "-")
## End(Not run)
```
**Change_Delim_Suffix**  
*Change barcode suffix delimiter*

**Description**

Change barcode suffix delimiter from list of data.frames/matrices or single data.frame/matrix

**Usage**

Change_Delim_Suffix(data, current_delim, new_delim)

**Arguments**

- **data**: Either matrix/data.frame or list of matrices/data.frames with the cell barcodes in the column names.
- **current_delim**: a single value of current delimiter.
- **new_delim**: a single value of new delimiter desired.

**Value**

matrix or data.frame with new column names.

**Examples**

```r
## Not run:
dge_matrix <- Change_Delim_Suffix(data = dge_matrix, current_delim = ".", new_delim = "-")
## End(Not run)
```

---

**CheckMatrix_scCustom**  
*Check Matrix Validity*

**Description**

Native implementation of SeuratObjects CheckMatrix but with modified warning messages.

**Usage**

CheckMatrix_scCustom(
  object,
  checks = c("infinite", "logical", "integer", "na")
)

Clustered DotPlot

Arguments

- **object**: A matrix
- **checks**: Type of checks to perform, choose one or more from:
  - "infinite": Emit a warning if any value is infinite
  - "logical": Emit a warning if any value is a logical
  - "integer": Emit a warning if any value is not an integer
  - "na": Emit a warning if any value is an NA or NaN

Value

Emits warnings for each test and invisibly returns NULL

References

Re-implementing CheckMatrix only for sparse matrices with modified warning messages. Original function from SeuratObject [https://github.com/mojaveazure/seurat-object/blob/9c0eda946e162d8595696e5280a1f7070bf021a9/R/utils.R#L625-L650](https://github.com/mojaveazure/seurat-object/blob/9c0eda946e162d8595696e5280a1f7070bf021a9/R/utils.R#L625-L650) (License: MIT).

Examples

```r
## Not run:
mat <- Read10X(...)
CheckMatrix_scCustom(object = mat)
## End(Not run)
```

Clustered DotPlot

Clustered DotPlots using ComplexHeatmap

Usage

```r
Clustered DotPlot(
  seurat_object,
  features,
  colors_use_exp = viridis_plasma_dark_high,
  exp_color_min = -2,
  exp_color_middle = NULL,
  exp_color_max = 2,
  print_exp_quantiles = FALSE,
  colors_use_idents = NULL,
  x_lab_rotate = TRUE,
  flip = FALSE,
)```
k = 1,
feature_km_repeats = 1000,
ident_km_repeats = 1000,
row_km_repeats = deprecated(),
column_km_repeats = deprecated(),
row_label_size = 8,
raster = FALSE,
plot_km_elbow = TRUE,
elbow_kmax = NULL,
assay = NULL,
group.by = NULL,
idents = NULL,
show_parent_dend_line = TRUE,
ggplot_default_colors = FALSE,
color_seed = 123,
seed = 123
)

Arguments

seurat_object  Seurat object name.
features  Features to plot.
colors_use_exp  Color palette to use for plotting expression scale. Default is viridis::plasma(n = 20, direction = -1).
exp_color_min  Minimum scaled average expression threshold (everything smaller will be set to this). Default is -2.
exp_color_middle  What scaled expression value to use for the middle of the provided colors_use_exp. By default will be set to value in middle of exp_color_min and exp_color_max.
exp_color_max  Minimum scaled average expression threshold (everything smaller will be set to this). Default is 2.
print_exp_quantiles  Whether to print the quantiles of expression data in addition to plots. Default is FALSE. NOTE: These values will be altered by choices of exp_color_min and exp_color_min if there are values below or above those cutoffs, respectively.
colors_use_idents  specify color palette to used for identity labels. By default if number of levels plotted is less than or equal to 36 it will use "polychrome" and if greater than 36 will use "varibow" with shuffle = TRUE both from DiscretePalette_scCustomize.
x_lab_rotate  How to rotate column labels. By default set to TRUE which rotates labels 45 degrees. If set FALSE rotation is set to 0 degrees. Users can also supply custom angle for text rotation.
flip  logical, whether to flip the axes of final plot. Default is FALSE; rows = features and columns = idents.
k  Value to use for k-means clustering on features Sets (km) parameter in ComplexHeatmap::Heatmap(). From ComplexHeatmap::Heatmap(): Apply k-means clustering on rows. If the
value is larger than 1, the heatmap will be split by rows according to the k-means clustering. For each row slice, hierarchical clustering is still applied with parameters above.

feature_km_repeats
Number of k-means runs to get a consensus k-means clustering for features. Note if feature_km_repeats is set to value greater than one, the final number of groups might be smaller than row_km, but this might mean the original row_km is not a good choice. Default is 1000.

ident_km_repeats
Number of k-means runs to get a consensus k-means clustering. Similar to feature_km_repeats. Default is 1000.

row_km_repeats [Deprecated] soft-deprecated. See feature_km_repeats

column_km_repeats [Deprecated] soft-deprecated. See ident_km_repeats

row_label_size
Size of the feature labels. Provided to row_names_gp in Heatmap call.

raster
Logical, whether to render in raster format (faster plotting, smaller files). Default is FALSE.

plot_km_elbow
Logical, whether or not to return the Sum Squared Error Elbow Plot for k-means clustering. Estimating elbow of this plot is one way to determine "optimal" value for k. Based on: https://stackoverflow.com/a/15376462/15568251.

elbow_kmax
The maximum value of k to use for plot_km_elbow. Suggest setting larger value so the true shape of plot can be observed. Value must be 1 less than number of features provided. If NULL parameter will be set dependent on length of feature list up to elbow_kmax = 20.

assay
Name of assay to use, defaults to the active assay.

group.by
Group (color) cells in different ways (for example, orig.ident).

idents
Which classes to include in the plot (default is all).

show_parent_dend_line
Logical, Sets parameter of same name in ComplexHeatmap::Heatmap(). When heatmap is split, whether to add a dashed line to mark parent dendrogram and children dendrograms. Default is TRUE.

ggplot_default_colors
logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.

color_seed
random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.

seed
Sets seed for reproducible plotting (ComplexHeatmap plot).

Value
A ComplexHeatmap or if plot_km_elbow = TRUE a list containing ggplot2 object and ComplexHeatmap.

Author(s)
Ming Tang (Original Code), Sam Marsh (Wrap single function, added/modified functionality)
Cluster_Highlight_Plot

References

https://divingintogeneticsandgenomics.rbind.io/post/clustered-dotplot-for-single-cell-rnaseq/

See Also

https://twitter.com/tangming2005

Examples

library(Seurat)
Clustered_DotPlot(seurat_object = pbmc_small, features = c("CD3E", "CD8", "GZMB", "MS4A1"))

Description

Create Plot with cluster of interest highlighted

Usage

Cluster_Highlight_Plot(
  seurat_object,
  cluster_name,
  highlight_color = "navy",
  background_color = "lightgray",
  pt.size = NULL,
  raster = NULL,
  raster.dpi = c(512, 512),
  label = FALSE,
  split.by = NULL,
  split_seurat = FALSE,
  ...
)

Arguments

seurat_object  Seurat object name.
cluster_name   Name(s) (or number(s)) identity of cluster to be highlighted.
highlight_color Color(s) to highlight cells (default "navy").
background_color non-highlighted cell colors.
pt.size  point size for both highlighted cluster and background.

raster   Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.

raster.dpi  Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).

label   Whether to label the highlighted cluster(s). Default is FALSE.

split.by  Feature to split plots by (i.e. "orig.ident").

split_seurat  logical. Whether or not to display split plots like Seurat (shared y axis) or as individual plots in layout. Default is FALSE.

Value
   A ggplot object

Examples
   Cluster_Highlight_Plot(seurat_object = pbmc_small, cluster_name = "1", highlight_color = "gold", background_color = "lightgray", pt.size = 2)

Cluster_Stats_All_Samples
   Calculate Cluster Stats

Description
   Calculates both overall and per sample cell number and percentages per cluster based on orig.ident

Usage
   Cluster_Stats_All_Samples(seurat_object, group_by_var = "orig.ident")

Arguments
   seurat_object  Seurat object name.
   group_by_var   meta data column to classify samples (default = "orig.ident").

Value
   A data.frame

Examples
   ## Not run:
   stats <- Cluster_Stats_All_Samples(seurat_object = object, group_by_var = "orig.ident")

   ## End(Not run)
ColorBlind_Pal

Description
Shortcut to a modified 8 color palette based on Color Universal Design (CUD) colorblindness friendly palette.

Usage
ColorBlind_Pal()

Value
modified/reordered color palette (8 colors) based on ditto-seq

References
palette is slightly modified version of the Color Universal Design (CUD) colorblindness friendly palette [https://jfly.uni-koeln.de/color/](https://jfly.uni-koeln.de/color/).

Examples
cols <- ColorBlind_Pal()
PalettePlot(palette = cols)

Copy_From_GCP

Copy folder from GCP bucket from R Console

Description
Run command from R console without moving to terminal to copy folder from GCP bucket to local storage

Usage
Copy_From_GCP(folder_file_path, gcp_bucket_path)

Arguments
folder_file_path
folder to be copied to GCP bucket.

gcp_bucket_path
GCP bucket path to copy to files.
Copy_To_GCP

Value

No return value. Performs system copy from GCP bucket.

Examples

```r
## Not run:
Copy_From_GCP(folder_file_path = "plots/", gcp_bucket_path = "gs://bucket_name_and_folder_path")

## End(Not run)
```

Copy_To_GCP

Copy folder to GCP bucket from R Console

Description

Run command from R console without moving to terminal to copy folder to GCP bucket

Usage

```r
Copy_To_GCP(folder_file_path, gcp_bucket_path)
```

Arguments

- `folder_file_path` folder to be copied to GCP bucket.
- `gcp_bucket_path` GCP bucket path to copy to files.

Value

No return value. Performs system copy to GCP bucket.

Examples

```r
## Not run:
Copy_To_GCP(folder_file_path = "plots/", gcp_bucket_path = "gs://bucket_name_and_folder_path")

## End(Not run)
```
Create_10X_H5

Create H5 from 10X Outputs

Description

Creates HDF5 formatted output analogous to the outputs created by Cell Ranger and can be read into Seurat, LIGER, or SCE class object. Requires DropletUtils package from Bioconductor.

Usage

Create_10X_H5(
  raw_data_file_path,
  source_type = "10X",
  save_file_path,
  save_name
)

Arguments

- raw_data_file_path: file path to raw data file(s).
- source_type: type of source data (Default is "10X"). Alternatively can provide "Matrix" or "data.frame".
- save_file_path: file path to directory to save file.
- save_name: name prefix for output H5 file.

Value

A HDF5 format file that will be recognized as 10X Cell Ranger formatted file by Seurat or LIGER.

Examples

```r
## Not run:
Create_10X_H5(raw_data_file_path = "file_path", save_file_path = "file_path2", save_name = "NAME")

## End(Not run)
```
Create_CellBender_Merged_Seurat

Create Seurat Object with Cell Bender and Raw data

Description

Enables easy creation of Seurat object which contains both cell bender data and raw count data as separate assays within the object.

Usage

Create_CellBender_Merged_Seurat(
  raw_cell_bender_matrix = NULL,
  raw_counts_matrix = NULL,
  raw_assay_name = "RAW",
  min_cells = 5,
  min_features = 200,
  ...
)

Arguments

raw_cell_bender_matrix
  matrix file containing the cell bender correct counts.

raw_counts_matrix
  matrix file contain the uncorrected Cell Ranger (or other) counts.

raw_assay_name
  a key value to use specifying the name of assay. Default is "RAW".

min_cells
  value to supply to min.cells parameter of CreateSeuratObject. Default is 5.

min_features
  value to supply to min.features parameter of CreateSeuratObject. Default is 200.

...
  Extra parameters passed to CreateSeuratObject.

Value

A Seurat Object contain both the Cell Bender corrected counts ("RNA" assay) and uncorrected counts ("RAW" assay; or other name specified to raw_assay_name).

Examples

## Not run:
seurat_obj <- Create_CellBender_Merged_Seurat(raw_cell_bender_matrix = cb_matrix,
  raw_counts_matrix = cr_matrix)

## End(Not run)
Create_Cluster_Annotation_File

Create cluster annotation csv file

Description

create annotation file

Usage

Create_Cluster_Annotation_File(
  file_path = NULL,
  file_name = "cluster_annotation"
)

Arguments

  file_path      path to directory to save file. Default is current working directory.
  file_name      name to use for annotation file. Function automatically adds file type ".csv" suffix. Default is "cluster_annotation".

Value

No value returned. Creates .csv file.

Examples

## Not run:
Create_Cluster_Annotation_File(file_path = "cluster_annotation_folder_name")

## End(Not run)

Dark2_Pal

Dark2 Palette

Description

Shortcut to Dark2 color palette from RColorBrewer (8 Colors)

Usage

Dark2_Pal()

Value

"Dark2" color palette (8 colors)
References

Dark2 palette from RColorBrewer being called through paletteer. See RColorBrewer for more info on palettes https://CRAN.R-project.org/package=RColorBrewer

Examples

cols <- Dark2_Pal()
PalettePlot(palette = cols)

---

DimPlot_All_Samples  DimPlot by Meta Data Column

Description

Creates DimPlot layout containing all samples within Seurat Object from orig.ident column

Usage

DimPlot_All_Samples(
  seurat_object,
  meta_data_column = "orig.ident",
  colors_use = "black",
  pt.size = NULL,
  title_size = 15,
  num_columns = NULL,
  reduction = NULL,
  dims = c(1, 2),
  raster = NULL,
  raster.dpi = c(512, 512),
  ...
)

Arguments

seurat_object  Seurat object name.
meta_data_column  Meta data column to split plots by.
colors_use  single color to use for all plots or a vector of colors equal to the number of plots.
pt.size  Adjust point size for plotting.
title_size  size for plot title labels.
um_columns  number of columns in final layout plot.
reduction  Dimensionality Reduction to use (if NULL then defaults to Object default).
dims  Which dimensions to plot. Defaults to c(1,2) if not specified.
**raster**  
Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.

**raster.dpi**  
Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).

...  
Extra parameters passed to DimPlot.

**Value**

A ggplot object

**Examples**

```r
library(Seurat)

pbmc_small$sample_id <- sample(c("sample1", "sample2"), size = ncol(pbmc_small), replace = TRUE)

DimPlot.All.Samples(seurat_object = pbmc_small, meta_data_column = "sample_id", color = "black", num_columns = 2)
```

---

**DimPlot_LIGER**  

**DimPlot LIGER Version**

**Description**

Standard and modified version of LIGER’s plotByDatasetAndCluster

**Usage**

```r
DimPlot_LIGER(
  liger_object,
  group_by = NULL,
  split_by = NULL,
  colors_use_cluster = NULL,
  colors_use_meta = NULL,
  pt_size = NULL,
  shuffle = TRUE,
  shuffle_seed = 1,
  reduction_label = "UMAP",
  label = TRUE,
  label_size = NA,
  label_repel = FALSE,
  label_box = FALSE,
  label_color = "black",
  combination = FALSE,
  raster = NULL,
  raster.dpi = c(512, 512),
)```

num_columns = NULL,
  ggplot_default_colors = FALSE,
  color_seed = 123
)

Arguments

liger_object  liger liger_object. Need to perform clustering before calling this function

group_by  Variable to be plotted. If NULL will plot clusters from liger@clusters slot. If combination = TRUE will plot both clusters and meta data variable.

split_by  Variable to split plots by.

colors_use_cluster  colors to use for plotting by clusters. By default if number of levels plotted is less than or equal to 36 will use "polychrome" and if greater than 36 will use "varibow" with shuffle = TRUE both from DiscretePalette_scCustomize.

colors_use_meta  colors to use for plotting by meta data (cell.data) variable. By default if number of levels plotted is less than or equal to 36 it will use "polychrome" and if greater than 36 will use "varibow" with shuffle = TRUE both from DiscretePalette_scCustomize.

pt_size  Adjust point size for plotting.

shuffle  logical. Whether to randomly shuffle the order of points. This can be useful for crowded plots if points of interest are being buried. (Default is TRUE).

shuffle_seed  Sets the seed if randomly shuffling the order of points.

reduction_label  What to label the x and y axes of resulting plots. LIGER does not store name of technique and therefore needs to be set manually. Default is "UMAP".

label  logical. Whether or not to label the clusters. ONLY applies to plotting by cluster. Default is TRUE.

label_size  size of cluster labels.

label_repel  logical. Whether to repel cluster labels from each other if plotting by cluster (if group_by = NULL or group_by = "cluster"). Default is FALSE.

label_box  logical. Whether to put a box around the label text (uses geom_text vs geom_label). Default is FALSE.

label_color  Color to use for cluster labels. Default is "black".

combination  logical, whether to return patchwork displaying both plots side by side. (Default is FALSE).

raster  Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.

raster.dpi  Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).

num_columns  Number of columns in plot layout. Only valid if split.by != NULL.

ggplot_default_colors  logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.
DimPlot_scCustom

color_seed random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.

Value

A ggplot/patchwork object

Examples

## Not run:
DimPlot_LIGER(lier_object = obj_name, reduction_label = "UMAP")

## End(Not run)

DimPlot_scCustom DimPlot with modified default settings

Description

Creates DimPlot with some of the settings modified from their Seurat defaults (colors_use, shuffle, label).

Usage

DimPlot_scCustom(
  seurat_object,
  colors_use = NULL,
  pt.size = NULL,
  reduction = NULL,
  group.by = NULL,
  split.by = NULL,
  split_seurat = FALSE,
  figure_plot = FALSE,
  shuffle = TRUE,
  seed = 1,
  label = NULL,
  label.size = 4,
  label.color = "black",
  label.box = FALSE,
  dims = c(1, 2),
  repel = FALSE,
  raster = NULL,
  raster.dpi = c(512, 512),
  num_columns = NULL,
  ggplot_default_colors = FALSE,
  color_seed = 123,
  ...
)
Arguments

**seurat_object**
Seurat object name.

**colors_use**
Color palette to use for plotting. By default if number of levels plotted is less than or equal to 36 it will use "polychrome" and if greater than 36 will use "varibow" with shuffle = TRUE both from DiscretePalette_scCustomize.

**pt.size**
Adjust point size for plotting.

**reduction**
Dimensionality Reduction to use (if NULL then defaults to Object default).

**group.by**
Name of one or more metadata columns to group (color) cells by (for example, orig.ident); default is the current active.ident of the object.

**split.by**
Feature to split plots by (i.e. "orig.ident").

**split_seurat**
Logical. Whether or not to display split plots like Seurat (shared y axis) or as individual plots in layout. Default is FALSE.

**figure_plot**
Logical. Whether to remove the axes and plot with legend on left of plot denoting axes labels. (Default is FALSE). Requires split_seurat = TRUE.

**shuffle**
Logical. Whether to randomly shuffle the order of points. This can be useful for crowded plots if points of interest are being buried. (Default is TRUE).

**seed**
Sets the seed if randomly shuffling the order of points.

**label**
Whether to label the clusters. By default if group.by = NULL label = TRUE, and otherwise it is FALSE.

**label.size**
Sets size of labels.

**label.color**
Sets the color of the label text.

**label.box**
Whether to put a box around the label text (geom_text vs geom_label).

**dins**
Which dimensions to plot. Defaults to c(1,2) if not specified.

**repel**
Repel labels.

**raster**
Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.

**raster.dpi**
Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).

**num_columns**
Number of columns in plot layout. Only valid if split.by != NULL.

**ggplot_default_colors**
Logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.

**color_seed**
Random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.

**...**
Extra parameters passed to DimPlot.

Value

A ggplot object
DiscretePalette_scCustomize

References

Many of the param names and descriptions are from Seurat to facilitate ease of use as this is simply a wrapper to alter some of the default parameters https://github.com/satijalab/seurat/blob/master/R/visualization.R (License: GPL-3). figure_plot parameter/code modified from code by Tim Stuart via twitter: https://twitter.com/timoast/status/1526237116035891200? s=20&t=foJOF81aPSjr1t7pk1cUPg.

Examples

library(Seurat)
DimPlot_scCustom(seurat_object = pbmc_small)

DiscretePalette_scCustomize

Discrete color palettes

Description

Helper function to return a number of discrete color palettes.

Usage

DiscretePalette_scCustomize(
  num_colors,
  palette = NULL,
  shuffle_pal = FALSE,
  seed = 123
)

Arguments

num_colors Number of colors to be generated.

palette Options are "alphabet", "alphabet2", "glasbey", "polychrome", "stepped", "ditto_seq", 
"varibow".

shuffle_pal randomly shuffle the outputted palette. Most useful for varibow palette as that 
is normally an ordered palette.

seed random seed for the palette shuffle. Default = 123.

Value

A vector of colors
References

This function uses the paletteer package https://github.com/EmilHvitfeldt/paletteer to provide simplified access to color palettes from many different R package sources while minimizing scCustomize current and future dependencies.

The following packages & palettes are called by this function (see individual packages for palette references/citations):

1. pals (via paletteer) https://CRAN.R-project.org/package=pals
   - alphabet, alphabet2, glasbey, polychrome, and stepped.
   - dittoColors.
3. colorway https://github.com/hypercompetent/colorway
   - varibow

Function name and implementation modified from Seurat (License: GPL-3). https://github.com/satijalab/seurat

Examples

```r
pal <- DiscretePalette_scCustomize(num_colors = 36, palette = "varibow")
PalettePlot(palette = pal)
```

---

**DotPlot_scCustom**  
**Customized DotPlot**

**Description**

Code for creating customized DotPlot

**Usage**

```r
DotPlot_scCustom(
  seurat_object,
  features,
  colors_use = viridis_plasma_dark_high,
  remove_axis_titles = TRUE,
  x_lab_rotate = FALSE,
  y_lab_rotate = FALSE,
  facet_label_rotate = FALSE,
  flip_axes = FALSE,
  ...
)
```
**ensembl_mito_id**

### Arguments

- **seurat_object**: Seurat object name.
- **features**: Features to plot.
- **colors_use**: Specify color palette to used. Default is `viridis_plasma_dark_high`.
- **remove_axis_titles**: Logical. Whether to remove the x and y axis titles. Default = `TRUE`.
- **x_lab_rotate**: Rotate x-axis labels 45 degrees (Default is `FALSE`).
- **y_lab_rotate**: Rotate x-axis labels 45 degrees (Default is `FALSE`).
- **facet_label_rotate**: Rotate facet labels on grouped DotPlots by 45 degrees (Default is `FALSE`).
- **flip_axes**: Whether or not to flip and X and Y axes (Default is `FALSE`).
- ... Extra parameters passed to `DotPlot`.

### Value

A ggplot object

### Examples

```r
library(Seurat)
DotPlot_scCustom(seurat_object = pbmc_small, features = c("CD3E", "CD8", "GZMB", "MS4A1"))
```

---

### Description

A list of ensembl ids for mitochondrial genes (Ensembl version 105)

### Usage

`ensembl_mito_id`

### Format

A list of six vectors

- **Mus_musculus_mito_ensembl**: Ensembl IDs for mouse mitochondrial genes
- **Homo_sapiens_mito_ensembl**: Ensembl IDs for human mitochondrial genes
- **Danio_rerio_mito_ensembl**: Ensembl IDs for zebrafish mitochondrial genes
- **Rattus_norvegicus_mito_ensembl**: Ensembl IDs for rat mitochondrial genes
- **Drosophila_melanogaster_mito_ensembl**: Ensembl IDs for fly mitochondrial genes
- **Macaca_mulatta_mito_ensembl**: Ensembl IDs for macaque mitochondrial genes
ensembl_ribo_id  Ensembl Ribo IDs

Description
A list of ensembl ids for ribosomal genes (Ensembl version 105)

Usage
ensemb_ribo_id

Format
A list of seven vectors

Mus_musculus_ribo_ensembl  Ensembl IDs for mouse ribosomal genes
Homo_sapiens_ribo_ensembl  Ensembl IDs for human ribosomal genes
Callithrix_jacchus_ribo_ensembl  Ensembl IDs for marmoset ribosomal genes
Danio rerio_ribo_ensembl  Ensembl IDs for zebrafish ribosomal genes
Rattus norvegicus_ribo_ensembl  Ensembl IDs for rat ribosomal genes
Drosophila melanogaster_ribo_ensembl  Ensembl IDs for fly ribosomal genes
Macaca mulatta_ribo_ensembl  Ensembl IDs for macaque ribosomal genes

Extract_Sample_Meta  Extract sample level meta.data

Description
Returns a by identity meta.data data.frame with one row per sample. Useful for downstream quick view of sample breakdown, meta data table creation, and/or use in pseudobulk analysis

Usage
Extract_Sample_Meta(
  object,
  sample_name = "orig.ident",
  variables_include = NULL,
  variables_exclude = NULL,
  include_all = FALSE
)

Extract_Top_Markers

Arguments

object Seurat object
sample_name meta.data column to use as sample. Output data.frame will contain one row per level or unique value in this variable.
variables_include @meta.data columns to keep in final data.frame. All other columns will be discarded. Default is NULL.
variables_exclude columns to discard in final data.frame. Many cell level columns are irrelevant at the sample level (e.g., nFeature_RNA, percent_mito).
  • Default parameter value is NULL but internally will set to discard nFeature_ASSAY(s), nCount_ASSAY(s), percent_mito, percent_ribo, percent_mito_ribo, and log10GenesPerUMI.
  • If sample level median values are desired for these type of variables the output of this function can be joined with output of Median_Stats.
  • Set parameter to include_all = TRUE to prevent any columns from being excluded.
include_all logical, whether or not to include all object meta data columns in output data.frame. Default is FALSE.

Value

Returns a data.frame with one row per sample_name.

Examples

library(Seurat)
pbmc_small$batch <- sample(c("batch1", "batch2"), size = ncol(pbmc_small), replace = TRUE)
sample_meta <- Extract_Sample_Meta(object = pbmc_small, sample_name = "orig.ident")

# Only return specific columns from meta data (orig.ident and batch)
sample_meta <- Extract_Sample_Meta(object = pbmc_small, sample_name = "orig.ident",
variables_include = "batch")

# Return all columns from meta data
sample_meta <- Extract_Sample_Meta(object = pbmc_small, sample_name = "orig.ident",
include_all = TRUE)

Extract_Top_Markers Extract Top N Marker Genes

Description

Extract vector gene list (or named gene vector) from data.frame results of FindAllMarkers or similar analysis.
Usage

Extract_Top_Markers(
  marker_dataframe,
  num_genes = 10,
  group_by = "cluster",
  rank_by = "avg_log2FC",
  gene_column = "gene",
  gene_rownames_to_column = FALSE,
  data_frame = FALSE,
  named_vector = FALSE,
  make_unique = FALSE
)

Arguments

marker_dataframe
  data.frame output from FindAllMarkers or similar analysis.
num_genes
  number of genes per group (e.g., cluster) to include in output list.
group_by
  column name of marker_dataframe to group data by. Default is "cluster" based on FindAllMarkers.
rank_by
  column name of marker_dataframe to rank data by when selecting num_genes per group_by. Default is "avg_log2FC" based on FindAllMarkers.
gene_column
  column name of marker_dataframe that contains the gene IDs. Default is "gene" based on FindAllMarkers.
gene_rownames_to_column
  logical. Whether gene IDs are stored in rownames and should be moved to column. Default is FALSE.
data_frame
  Logical, whether or not to return filtered data.frame of the original markers_dataframe or to return a vector of gene IDs. Default is FALSE.
named_vector
  Logical, whether or not to name the vector of gene names that is returned by the function. If TRUE will name the vector using the column provided to group_by. Default is TRUE.
make_unique
  Logical, whether an unnamed vector should return only unique values. Default is FALSE. Not applicable when data_frame = TRUE or named_vector = TRUE.

Value

filtered data.frame, vector, or named vector containing gene IDs.

Examples

## Not run:
top10_genes <- Extract_Top_Markers(marker_dataframe = markers_results, num_genes = 10,
group_by = "cluster", rank_by = "avg_log2FC")

## End(Not run)
FeaturePlot_DualAssay Customize FeaturePlot of two assays

Description
Create Custom FeaturePlots and preserve scale (no binning) from same features in two assays simultaneously. Intended for plotting same modality present in two assays.

Usage
FeaturePlot_DualAssay(
  seurat_object,
  features,
  assay1 = "RAW",
  assay2 = "RNA",
  colors_use = viridis_plasma_dark_high,
  na_color = "lightgray",
  order = TRUE,
  pt.size = NULL,
  reduction = NULL,
  na_cutoff = 1e-09,
  raster = NULL,
  raster.dpi = c(512, 512),
  slot = "data",
  num_columns = NULL,
  alpha_exp = NULL,
  alpha_na_exp = NULL,
...
)

Arguments
- seurat_object: Seurat object name.
- features: Feature(s) to plot.
- assay1: name of assay one. Default is "RAW" as featured in Create_CellBender_Merged_Seurat
- assay2: name of assay two. Default is "RNA" as featured in Create_CellBender_Merged_Seurat
- colors_use: list of colors or color palette to use.
- na_color: color to use for points below lower limit.
- order: whether to move positive cells to the top (default = TRUE).
- pt.size: Adjust point size for plotting.
- reduction: Dimensionality Reduction to use (if NULL then defaults to Object default).
- na_cutoff: Value to use as minimum expression cutoff. To set no cutoff set to NA.
- raster: Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.
FeaturePlot_scCustom

Description
Create Custom FeaturePlots and preserve scale (no binning)

Usage

```
FeaturePlot_scCustom(
  seurat_object, features, colors_use = viridis_plasma_dark_high, na_color = "lightgray",
  order = TRUE,
  pt.size = NULL, reduction = NULL, na_cutoff = 1e-09,
  raster = NULL, raster.dpi = c(512, 512), split.by = NULL,
```

Value
A ggplot object

Examples
```r
## Not run:
FeaturePlot_DualAssay(seurat_object = object, features = "Cx3cr1", assay1 = "RAW", assay2 = "RNA",
  colors_use = viridis_plasma_dark_high, na_color = "lightgray")
```

```r
## End(Not run)
```
num_columns = NULL,
slot = "data",
alpha_exp = NULL,
alpha_na_exp = NULL,
label = FALSE,
label_feature_yaxis = FALSE,
combine = TRUE,
...)

**Arguments**

- **seurat_object**  Seurat object name.
- **features**  Feature(s) to plot.
- **colors_use**  list of colors or color palette to use.
- **na_color** color to use for points below lower limit.
- **order** whether to move positive cells to the top (default = TRUE).
- **pt.size** Adjust point size for plotting.
- **reduction** Dimensionality Reduction to use (if NULL then defaults to Object default).
- **na_cutoff** Value to use as minimum expression cutoff. This will be lowest value plotted use palette provided to colors_use. Leave as default value to plot only positive non-zero values using color scale and zero/negative values as NA. To plot all values using color palette set to NA.
- **raster** Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.
- **raster.dpi** Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).
- **split.by** Variable in @meta.data to split the plot by.
- **num_columns** Number of columns in plot layout.
- **slot** Which slot to pull expression data from? Default is "data".
- **alpha_exp** new alpha level to apply to expressing cell color palette (colors_use). Must be value between 0-1.
- **alpha_na_exp** new alpha level to apply to non-expressing cell color palette (na_color). Must be value between 0-1.
- **label** logical, whether to label the clusters. Default is FALSE.
- **label_feature_yaxis** logical, whether to place feature labels on secondary y-axis as opposed to above legend key. Default is FALSE. When setting label_feature_yaxis = TRUE the number of columns in plot output will automatically be set to the number of levels in split.by'
- **combine** Combine plots into a single patchworked ggplot object. If FALSE, return a list of ggplot objects.
- **...** Extra parameters passed to FeaturePlot.
Fetch_Meta

Value

A ggplot object

Examples

library(Seurat)
FeaturePlot_scCustom(seurat_object = pbmc_small, features = "CD3E",
  colors_use = viridis_plasma_dark_high, na_color = "lightgray")

Fetch_Meta

Get meta data from object

Description

Quick function to properly pull meta.data from objects.

Usage

Fetch_Meta(object)

Arguments

object list of matrices to merge.

Value

A data.frame

Examples

## Not run:
meta_data <- Fetch_Meta(object = pbmc)

## End(Not run)
Gene_Present

Check if genes/features are present

Description

Check if genes are present in object and return vector of found genes. Return warning messages for genes not found.

Usage

Gene_Present(
  data,
  gene_list,
  case_check = TRUE,
  case_check_msg = TRUE,
  print_msg = TRUE,
  omit_warn = TRUE,
  return_none = FALSE,
  seurat_assay = NULL
)

Arguments

data  Name of input data. Currently only data of classes: Seurat, liger, data.frame, 
dgCMatrix, dgTMatrix, tibble are accepted. Gene_IDs must be present in row-

names of the data.

gene_list vector of genes to check.

case_check logical. Whether or not to check if features are found if the case is changed from 
the input list (Sentence case to Upper and vice versa). Default is TRUE.

case_check_msg logical. Whether to print message to console if alternate case features are found 
in addition to inclusion in returned list. Default is TRUE.

print_msg logical. Whether message should be printed if all features are found. Default is 
TRUE.

omit_warn logical. Whether to print message about features that are not found in current 
object. Default is TRUE.

return_none logical. Whether list of found vs. bad features should still be returned if no 
features are found. Default is FALSE.

seurat_assay Name of assay to pull feature names from if data is Seurat Object. Defaults to 
DefaultAssay(OBJ) if NULL.

Value

A list of length 3 containing 1) found features, 2) not found features, 3) features found if case was modified.
Examples

```r
## Not run:
features <- Gene_Present(data = obj_name, gene_list = DEG_list, print_msg = TRUE, case_check = TRUE)
found_features <- features[[1]]

## End(Not run)
```

---

Hue_Pal

**Hue_Pal**

Description

Shortcut to hue_pal to return to ggplot2 defaults if user desires, from scales package.

Usage

```r
Hue_Pal(num_colors)
```

Arguments

- **num_colors**
  - number of colors to return in palette.

Value

hue color palette (as many colors as desired)

Examples

```r
cols <- Hue_Pal(num_colors = 8)
PalettePlot(palette = cols)
```

---

Iterate_Cluster_Highlight_Plot

**Iterate Cluster Highlight Plot**

Description

Iterate the create plots with cluster of interest highlighted across all cluster (active.idents) in given Seurat Object
Iterate_Cluster_Highlight_Plot

Usage

Iterate_Cluster_Highlight_Plot(
  seurat_object,
  highlight_color = "navy",
  background_color = "lightgray",
  pt.size = NULL,
  reduction = NULL,
  file_path = NULL,
  file_name = NULL,
  file_type = NULL,
  single_pdf = FALSE,
  dpi = 600,
  raster = NULL,
  ...
)

Arguments

  seurat_object  Seurat object name.
  highlight_color Color to highlight cells (default "navy"). Can provide either single color to use
                   for all clusters/plots or a vector of colors equal to the number of clusters to use
                   (in order) for the clusters/plots.
  background_color non-highlighted cell colors.
  pt.size        point size for both highlighted cluster and background.
  reduction      Dimensionality Reduction to use (if NULL then defaults to Object default).
  file_path      directory file path and/or file name prefix. Defaults to current wd.
  file_name      name suffix to append after sample name.
  file_type      File type to save output as. Must be one of following: ".pdf", ".png", ".tiff",
                  ".jpeg", or ".svg".
  single_pdf     saves all plots to single PDF file (default = FALSE). ‘file_type’ must be .pdf
  dpi            dpi for image saving.
  raster         Convert points to raster format. Default is NULL which will rasterize by default
                  if greater than 200,000 cells.

Value

  Saved plots

Examples

  ## Not run:
  Iterate_Cluster_Highlight_Plot(seurat_object = object, highlight_color = "navy",
                                 background_color = "lightgray", file_path = "path/", file_name = "name", file_type = "pdf",
                                 ...
Iterate DimPlot by Sample

Iterate DimPlot By Sample

Description

Iterate DimPlot by orig.ident column from Seurat object metadata

Usage

Iterate_DimPlot_bySample(
  seurat_object,
  file_path = NULL,
  file_name = NULL,
  file_type = NULL,
  single_pdf = FALSE,
  dpi = 600,
  color = "black",
  reduction = NULL,
  dims = c(1, 2),
  pt.size = NULL,
  ...
)

Arguments

- **seurat_object**: Seurat object name.
- **file_path**: directory file path and/or file name prefix. Defaults to current wd.
- **file_name**: name suffix to append after sample name.
- **file_type**: File type to save output as. Must be one of following: ".pdf", ".png", ".tiff", ".jpeg", or ".svg".
- **single_pdf**: saves all plots to single PDF file (default = FALSE). ‘file_type’ must be .pdf
- **dpi**: dpi for image saving.
- **color**: color scheme to use.
- **reduction**: Dimensionality Reduction to use (default is object default).
- **dims**: Dimensions to plot.
- **pt.size**: Adjust point size for plotting.
- **...**: Extra parameters passed to DimPlot.
Iterate_FeaturePlot_scCustom

Iterative Plotting of Gene Lists using Custom FeaturePlots

Description

Create and Save plots for Gene list with Single Command

Usage

Iterate_FeaturePlot_scCustom(
  seurat_object,
  gene_list,
  colors_use = viridis_plasma_dark_high,
  na_color = "lightgray",
  na_cutoff = 1e-09,
  split.by = NULL,
  order = TRUE,
  return_plots = FALSE,
  file_path = NULL,
  file_name = NULL,
  file_type = NULL,
  single_pdf = FALSE,
  dpi = 600,
  pt.size = NULL,
  reduction = NULL,
  raster = NULL,
  alpha_exp = NULL,
  alpha_na_exp = NULL,
  ...
)

Value

A ggplot object

Examples

## Not run:
Iterate_DimPlot_bySample(seurat_object = object, file_path = "plots/", file_name = "tsne",
  file_type = ".jpg", dpi = 600, color = "black")

## End(Not run)
Arguments

- **seurat_object**: Seurat object name.
- **gene_list**: vector of genes to plot. If a named vector is provided then the names for each gene will be incorporated into plot title if `single_pdf = TRUE` or into file name if FALSE.
- **colors_use**: color scheme to use.
- **na_color**: color for non-expressed cells.
- **na_cutoff**: Value to use as minimum expression cutoff. To set no cutoff set to `NA`.
- **split.by**: Variable in `@meta.data` to split the plot by.
- **order**: whether to move positive cells to the top (default = TRUE).
- **return_plots**: logical. Whether to return plots to list instead of saving them to file(s). Default is FALSE.
- **file_path**: directory file path and/or file name prefix. Defaults to current wd.
- **file_name**: name suffix and file extension.
- **file_type**: File type to save output as. Must be one of following: ".pdf", ".png", ".tiff", ".jpeg", or ".svg".
- **single_pdf**: saves all plots to single PDF file (default = FALSE). `file_type` must be .pdf.
- **dpi**: dpi for image saving.
- **pt.size**: Adjust point size for plotting.
- **reduction**: Dimensionality Reduction to use (if NULL then defaults to Object default).
- **raster**: Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.
- **alpha_exp**: new alpha level to apply to expressing cell color palette (colors_use). Must be value between 0-1.
- **alpha_na_exp**: new alpha level to apply to non-expressing cell color palette (na_color). Must be value between 0-1.
- **...**: Extra parameters passed to FeaturePlot.

Value

Saved plots

Examples

```r
## Not run:
Iterate_FeaturePlot_scCustom(seurat_object = object, gene_list = DEG_list, 
colors_use = viridis_plasma_dark_high, na_color = "lightgray", file_path = "plots/", 
file_name = "tsne", file_type = ".jpg", dpi = 600)
## End(Not run)
```
Iterate_Meta_Highlight_Plot

Description

Iterate the create plots with meta data variable of interest highlighted.

Usage

Iterate_Meta_Highlight_Plot(
  seurat_object,
  meta_data_column,
  new_meta_order = NULL,
  meta_data_sort = TRUE,
  highlight_color = "navy",
  background_color = "lightgray",
  pt.size = NULL,
  reduction = NULL,
  file_path = NULL,
  file_name = NULL,
  file_type = NULL,
  single_pdf = FALSE,
  dpi = 600,
  raster = NULL,
  ...
)

Arguments

  seurat_object       Seurat object name.
  meta_data_column   Name of the column in seurat_object@meta.data slot to pull value from for highlighting.
  new_meta_order      The order in which to plot each level within meta_data_column if single_PDF is TRUE.
  meta_data_sort      logical. Whether or not to sort and relevel the levels in meta_data_column if single_PDF is TRUE. Default is TRUE.
  highlight_color     Color to highlight cells (default "navy"). Can provide either single color to use for all clusters/plots or a vector of colors equal to the number of clusters to use (in order) for the clusters/plots.
  background_color    non-highlighted cell colors.
  pt.size             point size for both highlighted cluster and background.
Iterate_PC_Loading_Plots

Description

Plot PC Heatmaps and Dim Loadings for exploratory analysis

Usage

Iterate_PC_Loading_Plots(
  seurat_object,
  dims_plot = NULL,
  file_path = NULL,
  name_prefix = NULL,
  file_name = "PC_Loading_Plots",
  return_plots = FALSE
)
Iterate_Plot_Density_Custom

Arguments

- seurat_object: Seurat object name.
- dims_plot: number of PCs to plot (integer). Default is all dims present in PCA.
- file_path: directory file path to save file.
- name_prefix: prefix for file name (optional).
- file_name: suffix for file name. Default is "PC_Loading_Plots".
- return_plots: Whether to return the plot list (Default is FALSE). Must assign to environment to save plot list.

Value

A list of plots outputted as pdf

See Also

PCHeatmap and VizDimLoadings

Examples

```r
## Not run:
Iterate_PC_Loading_Plots(seurat_object = seurat, dims_plot = 25, file_path = "plots/"
```

Iterate_Plot_Density_Custom

Iterative Plotting of Gene Lists using Custom Density Plots

Description

Create and save plots for gene list with single command. Requires Nebulosa package from Bioconductor.

Usage

```r
Iterate_Plot_Density_Custom(
  seurat_object,
  gene_list,
  viridis_palette = "magma",
  custom_palette = NULL,
  pt.size = 1,
  file_path = NULL,
  file_name = NULL,
  file_type = NULL,
  single_pdf = FALSE,
)```

Iterate_Plot_Density_Custom

dpi = 600,
reduction = NULL,
combine = TRUE,
joint = FALSE,
...
)

Arguments

seurat_object Seurat object name.
gene_list vector of genes to plot. If a named vector is provided then the names for each gene will be incorporated into plot title if single_pdf = TRUE or into file name if FALSE.
viridis_palette color scheme to use.
custom_palette color for non-expressed cells.
pt.size Adjust point size for plotting.
file_path directory file path and/or file name prefix. Defaults to current wd.
file_name name suffix and file extension.
file_type File type to save output as. Must be one of following: ".pdf", ".png", ".tiff", ".jpeg", or ".svg".
single_pdf saves all plots to single PDF file (default = FALSE). file_type must be .pdf.
dpi dpi for image saving.
reduction Dimensionality Reduction to use (if NULL then defaults to Object default)
combine Create a single plot? If FALSE, a list with ggplot objects is returned.
joint NULL. This function only supports joint = FALSE. Leave as NULL to generate plots. To iterate joint plots see function: Iterate_Plot_Density_Joint.
...
Extra parameters passed to plot_density.

Value

Saved plots

Examples

## Not run:
Iterate_Plot_Density_Custom(seurat_object = object, gene_list = DEG_list, viridis_palette = "magma", file_path = "plots/", file_name = ".density_plots", file_type = ".jpg", dpi = 600)

## End(Not run)
Iterate_Plot_Density_Joint

Iterative Plotting of Gene Lists using Custom Joint Density Plots

Description

Create and save plots for gene list with single command. Requires Nebulosa package from Bioconductor.

Usage

Iterate_Plot_Density_Joint(
  seurat_object,
  gene_list,
  viridis_palette = "magma",
  custom_palette = NULL,
  pt.size = 1,
  file_path = NULL,
  file_name = NULL,
  file_type = NULL,
  single_pdf = FALSE,
  dpi = 600,
  reduction = NULL,
  combine = TRUE,
  joint = NULL,
  ...
)

Arguments

seurat_object Seurat object name.
gene_list a list of vectors of genes to plot jointly. Each entry in the list will be plotted for the joint density. All entries in list must be greater than 2 features. If a named list is provided then the names for each list entry will be incorporated into plot title if single_pdf = TRUE or into file name if FALSE.
viridis_palette color scheme to use.
custom_palette color for non-expressed cells.
pt.size Adjust point size for plotting.
file_path directory file path and/or file name prefix. Defaults to current wd.
file_name name suffix and file extension.
file_type File type to save output as. Must be one of following: ".pdf", ".png", ".tiff", ".jpeg", or ".svg".
single_pdf saves all plots to single PDF file (default = FALSE). ‘file_type‘ must be .pdf.
Iterate_VlnPlot_scCustom

Iterative Plotting of Gene Lists using VlnPlot_scCustom

Description

Create and Save plots for Gene list with Single Command

Usage

Iterate_VlnPlot_scCustom(
  seurat_object,  
gene_list,  
colors_use = NULL,  
pt.size = NULL,  
group.by = NULL,  
split.by = NULL,  
file_path = NULL,  
file.name = NULL,  
file.type = NULL,  
single.pdf = FALSE,  
raster = NULL,  
dpi = 600,  
ggplot.default.colors = FALSE,  
color.seed = 123,
  ...
)

Value

Saved plots

Examples

## Not run:
Iterate_Plot_Density_Joint(seurat_object = object, gene_list = DEG_list, viridis_palette = "magma", 
file_path = "plots/", file.name = "joint_plots", file.type = ".jpg", dpi = 600)

## End(Not run)
## Iterate_VlnPlot_scCustom

### Arguments

- `seurat_object`: Seurat object name.
- `gene_list`: list of genes to plot.
- `colors_use`: color palette to use for plotting. By default if number of levels plotted is less than or equal to 36 it will use "polychrome" and if greater than 36 will use "varibow" with shuffle = TRUE both from DiscretePalette_scCustomize.
- `pt.size`: point size for plotting.
- `group.by`: Name of one or more metadata columns to group (color) plot by (for example, orig.ident); default is the current active.ident of the object.
- `split.by`: Feature to split plots by (i.e. "orig.ident").
- `file_path`: directory file path and/or file name prefix. Defaults to current wd.
- `file_name`: name suffix and file extension.
- `file_type`: File type to save output as. Must be one of following: ".pdf", ".png", ".tiff", ".jpeg", or ".svg".
- `single_pdf`: saves all plots to single PDF file (default = FALSE). ‘file_type" must be .pdf.
- `raster`: Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 total points plotted (# Cells x # of features).
- `dpi`: dpi for image saving.
- `ggplot_default_colors`: logical. If `colors_use` = NULL, Whether or not to return plot using default gg-plot2 "hue" palette instead of default "polychrome" or "varibow" palettes.
- `color_seed`: random seed for the "varibow" palette shuffle if `colors_use` = NULL and number of groups plotted is greater than 36. Default = 123.
- `...`: Extra parameters passed to VlnPlot.

### Value

- Saved plots

### Examples

```r
## Not run:
Iterate_VlnPlot_scCustom(seurat_object = object, gene_list = DEG_list, colors = color_list, file_path = "plots/", file_name = ".vln", file_type = ".jpg", dpi = 600)

## End(Not run)
```
JCO_Four

Description

Shortcut to a specific JCO 4 color palette from ggsci package.

Usage

JCO_Four()

Value

4 color palette from the JCO ggsci palette

References

Selection of colors from the JCO palette from ggsci being called through paletteer. See ggsci for more info on palettes https://CRAN.R-project.org/package=ggsci

Examples

cols <- JCO_Four()
PalettePlot(palette = cols)

Liger_to_Seurat

Create a Seurat object containing the data from a liger object

Description

Merges raw.data and scale.data of object, and creates Seurat object with these values along with tsne.coords, iNMF factorization, and cluster assignments. Supports Seurat V2 and V3.

Usage

Liger_to_Seurat(
  liger_object,
  nms = names(liger_object@H),
  renormalize = TRUE,
  use.liger.genes = TRUE,
  by.dataset = FALSE,
  keep_meta = TRUE,
  reduction_label = NULL,
  seurat_assay = "RNA"
)
Arguments

- **liger_object**: liger object.
- **nms**: By default, labels cell names with dataset of origin (this is to account for cells in different datasets which may have same name). Other names can be passed here as vector, must have same length as the number of datasets. (default names(H)).
- **renormalize**: Whether to log-normalize raw data using Seurat defaults (default TRUE).
- **use.liger.genes**: Whether to carry over variable genes (default TRUE).
- **by.dataset**: Include dataset of origin in cluster identity in Seurat object (default FALSE).
- **keep_meta**: logical. Whether to transfer additional metadata (nGene/nUMI/dataset already transferred) to new Seurat Object. Default is TRUE.
- **reduction_label**: Name of dimensionality reduction technique used. Enables accurate transfer or name to Seurat object instead of defaulting to "tSNE".
- **seurat_assay**: Name to set for assay in Seurat Object. Default is "RNA".

Details

Stores original dataset identity by default in new object metadata if dataset names are passed in nms. iNMF factorization is stored in dim.reduction object with key "iNMF".

Value

Seurat object with raw.data, scale.data, reduction_label, iNMF, and ident slots set.

References

Original function is part of LIGER package [https://github.com/welch-lab/liger](https://github.com/welch-lab/liger) (Licence: GPL-3). Function was slightly modified for use in scCustomize with keep.meta parameter. Also posted as PR to liger GitHub.

Examples

```r
## Not run:
seurat_object <- Liger_to_Seurat(liger_object = LIGER_OBJ, reduction_label = "UMAP")
## End(Not run)
```
Median_Stats

Description

Get quick values for median Genes, UMIs, %mito per cell grouped by meta.data variable.

Usage

Median_Stats(
  seurat_object, 
  group_by_var = "orig.ident", 
  default_var = TRUE, 
  median_var = NULL
)

Arguments

seurat_object Seurat object name.

group_by_var Column in meta.data slot to group results by (default = "orig.ident").

default_var logical. Whether to include the default meta.data variables of: "nCount_RNA", "nFeature_RNA", "percent_mito", "percent_ribo", "percent_mito_ribo" in addition to variables supplied to median_var.

median_var Column(s) in @meta.data to calculate medians for in addition to defaults. Must be of class() integer or numeric.

Value

A data.frame.

Examples

## Not run:
med_stats <- Median_Stats(seurat_object - obj, group_by_var = "orig.ident")

## End(Not run)
Merge_Seurat_List

Merge a list of Seurat Objects

Description

Enables easy merge of a list of Seurat Objects. See See `merge` for more information,

Usage

```r
Merge_Seurat_List(
  list_seurat,
  add.cell.ids = NULL,
  merge.data = TRUE,
  project = "SeuratProject"
)
```

Arguments

- **list_seurat**: list composed of multiple Seurat Objects.
- **add.cell.ids**: A character vector of length(x = c(x, y)). Appends the corresponding values to the start of each objects’ cell names. See `merge`.
- **merge.data**: Merge the data slots instead of just merging the counts (which requires renormalization). This is recommended if the same normalization approach was applied to all objects. See `merge`.
- **project**: Project name for the Seurat object. See `merge`.

Value

A Seurat Object

Examples

```r
## Not run:
object_list <- list(obj1, obj2, obj3, ...)
merged_object <- Merge_Seurat_List(list_seurat = object_list)

## End(Not run)
```
Merge_Sparse_Data_All  Merge a list of Sparse Matrices

Description

Enables easy merge of a list of sparse matrices

Usage

```
Merge_Sparse_Data_All(
  matrix_list,
  add_cell_ids = NULL,
  prefix = TRUE,
  cell_id_delimiter = "_"
)
```

Arguments

- `matrix_list`: list of matrices to merge.
- `add_cell_ids`: a vector of sample ids to add as prefix to cell barcode during merge.
- `prefix`: logical. Whether `add_cell_ids` should be added as prefix to current cell barcodes/names or as suffix to current cell barcodes/names. Default is TRUE, add as prefix.
- `cell_id_delimiter`: The delimiter to use when adding cell id prefix/suffix. Default is "_".

Value

A sparse Matrix

References

Original function is part of LIGER package as non-exported function `https://github.com/welch-lab/liger/blob/master/R/utilities.R` (License: GPL-3). Function was modified for use in scCustomize (add progress bar, prefix vs. suffix, and delimiter options).

Examples

```
## Not run:
data_list <- Read10X_GEO(...)merged <- Merge_Sparse_Data_All(matrix_list = data_list, add_cell_ids = names(data_list),prefix = TRUE, cell_id_delimiter = "_")
## End(Not run)
```
Description

Create Plot with meta data variable of interest highlighted

Usage

Meta_Highlight_Plot(
  seurat_object,
  meta_data_column,
  meta_data_highlight,
  highlight_color = "navy",
  background_color = "lightgray",
  pt.size = NULL,
  raster = NULL,
  raster.dpi = c(512, 512),
  label = FALSE,
  split.by = NULL,
  split_seurat = FALSE,
  ...
)

Arguments

seurat_object  Seurat object name.
meta_data_column Name of the column in seurat_object@meta.data slot to pull value from for highlighting.
meta_data_highlight Name of variable(s) within meta_data_name to highlight in the plot.
highlight_color  Color to highlight cells (default "navy").
background_color non-highlighted cell colors.
pt.size  point size for both highlighted cluster and background.
raster Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.
raster.dpi  Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).
label  Whether to label the highlighted meta data variable(s). Default is FALSE.
split.by  Variable in @meta.data to split the plot by.
split_seurat logical. Whether or not to display split plots like Seurat (shared y axis) or as individual plots in layout. Default is FALSE.
...  Extra parameters passed to DimPlot.
Examples

library(Seurat)
pbmc_small$sample_id <- sample(c("sample1", "sample2"), size = ncol(pbmc_small), replace = TRUE)

Meta_Highlight_Plot(seurat_object = pbmc_small, meta_data_column = "sample_id",
meta_data_highlight = "sample1", highlight_color = "gold", background_color = "lightgray",
pt.size = 2)

Value

A ggplot object

Description

Check if any present meta data columns are numeric and returns vector of valid numeric columns.
Issues warning message if any columns not in numeric form.

Usage

Meta_Numeric(data)

Arguments

data a data.frame contain meta.data.

Value

vector of meta data columns that are numeric.

Examples

## Not run:
numeric_meta_columns <- Meta_Numeric(data = meta_data)

## End(Not run)
Check if meta data are present

Description

Check if meta data columns are present in object and return vector of found columns. Return warning messages for meta data columns not found.

Usage

```
Meta_Present(
  seurat_object,
  meta_col_names,
  print_msg = TRUE,
  omit_warn = TRUE,
  abort = TRUE
)
```

Arguments

- `seurat_object` object name.
- `meta_col_names` vector of column names to check.
- `print_msg` logical. Whether message should be printed if all features are found. Default is TRUE.
- `omit_warn` logical. Whether to print message about features that are not found in current object. Default is TRUE.
- `abort` logical. Whether or not to stop function and print stop message if no input meta_col_names are found. Default is TRUE.

Value

vector of meta data columns that are present

Examples

```
## Not run:
meta_variables <- Meta_Present(seurat_object = obj_name, gene_list = DEG_list, print_msg = TRUE)

## End(Not run)
```
Meta_Present_LIGER  

Check if meta data are present

Description

Check if meta data columns are present in object and return vector of found columns. Return warning messages for meta data columns not found.

Usage

Meta_Present_LIGER(liger_object, meta_col_names, print_msg = TRUE)

Arguments

- liger_object: object name.
- meta_col_names: vector of column names to check.
- print_msg: logical. Whether message should be printed if all features are found. Default is TRUE.

Value

vector of meta data columns that are present

Examples

## Not run:
meta_variables <- Meta_Present_LIGER(liger_object = obj, gene_list = DEG_list, print_msg = TRUE)
## End(Not run)

Meta_Remove_Seurat  

Remove meta data columns containing Seurat Defaults

Description

Remove any columns from new meta_data data.frame in preparation for adding back to Seurat Object

Usage

Meta_Remove_Seurat(
  meta_data,
  seurat_object,  
  barcodes_to_rownames = FALSE,
  barcodes_colname = "barcodes"
)
Move_Legend

Arguments

- `meta_data` : data.frame containing meta data.
- `seurat_object` : object name.
- `barcodes_to_rownames` : logical, are barcodes present as column and should they be moved to rownames (to be compatible with `Seurat::AddMetaData`). Default is FALSE.
- `barcodes_colname` : name of barcodes column in `meta_data`. Required if `barcodes_to_rownames = TRUE`.

Value

data.frame with only new columns.

Examples

```r
## Not run:
new_meta <- Meta_Remove_Seurat(meta_data = meta_data_df, seurat_object = object)
object <- AddMetaData(object = object, metadata = new_meta)
## End(Not run)
```
# Generate a plot and customize theme
library(ggplot2)
df <- data.frame(x = rnorm(n = 100, mean = 20, sd = 2), 
y = rbinom(n = 100, size = 100, prob = 0.2))
p <- ggplot(data = df, mapping = aes(x = x, y = y)) + geom_point(mapping = aes(color = 'red'))
p + Move_Legend("left")

---

**NavyAndOrange**  
*Navy and Orange Dual Color Palette*

**Description**
Shortcut to navy orange color plot

**Usage**
NavyAndOrange(flip_order = FALSE)

**Arguments**
- **flip_order**  
  Whether to flip the order of colors.

**Value**
Navy orange palette

**Examples**
cols <- NavyAndOrange()
PalettePlot(palette = cols)

---

**PalettePlot**  
*Plot color palette in viewer*

**Description**
Plots given color vector/palette in viewer to evaluate palette before plotting on data.

**Usage**
PalettePlot(palette = NULL)

**Arguments**
- **palette**  
  a vector of colors (either named colors of hex codes).
Value

Plot of all colors in supplied palette/vector

References


Examples

```r
pal <- DiscretePalette_scCustomize(num_colors = 36, palette = "varibow")
PalettePlot(palette = pal)
```

---

**Description**

Plot PC Heatmaps and Dim Loadings for exploratory analysis. Plots a single Heatmap and Gene Loading Plot. Used for PC_Loading_Plots function.

**Usage**

```r
PC_Plotting(seurat_object, dim_number)
```

**Arguments**

- `seurat_object` Seurat Object.
- `dim_number` A single dim to plot (integer).

**Value**

A plot of PC heatmap and gene loadings for single

**See Also**

PCHeatmap and VizDimLoadings

**Examples**

```r
library(Seurat)
PC_Plotting(seurat_object = pbmc_small, dim_number = 1)
```
Percent_Expressing  

*Calculate percent of expressing cells*

**Description**

Calculates the percent of cells that express a given set of features by various grouping factors.

**Usage**

```r
Percent_Expressing(
  seurat_object,
  features,
  threshold = 0,
  group_by = NULL,
  split_by = NULL,
  entire_object = FALSE,
  slot = "data",
  assay = NULL
)
```

**Arguments**

- `seurat_object`: Seurat object name.
- `features`: Feature(s) to plot.
- `threshold`: Expression threshold to use for calculation of percent expressing (default is 0).
- `group_by`: Factor to group the cells by.
- `split_by`: Factor to split the groups by.
- `entire_object`: logical (default = FALSE). Whether to calculate percent of expressing cells across the entire object as opposed to by cluster or by `group_by` variable.
- `slot`: Slot to pull feature data for. Default is "data".
- `assay`: Assay to pull feature data from. Default is active assay.

**Value**

A data.frame

**References**

Part of code is modified from Seurat package as used by DotPlot to generate values to use for plotting. Source code can be found here: [https://github.com/satijalab/seurat/blob/4e868fcede49dc0a3df47f94f5fb54b/R/visualization.R#L339](https://github.com/satijalab/seurat/blob/4e868fcede49dc0a3df47f94f5fb54b/R/visualization.R#L339) (License: GPL-3).
Examples

```r
## Not run:
percent_stats <- Percent_Expressing(seurat_object = object, features = "Cx3cr1", threshold = 0)

## End(Not run)
```

---

#### plotFactors_scCustom

**Customized version of plotFactors**

**Description**

Modified and optimized version of `plotFactors` function from LIGER package.

**Usage**

```r
plotFactors_scCustom(
  liger_object, 
  num_genes = 8, 
  colors_use_factors = NULL, 
  colors_use_dimreduc = c("lemonchiffon", "red"), 
  pt.size_factors = 1, 
  pt.size_dimreduc = 1, 
  reduction_label = "UMAP", 
  raster = TRUE, 
  raster.dpi = c(512, 512), 
  order = FALSE, 
  plot_dimreduc = TRUE, 
  save_plots = TRUE, 
  file_path = NULL, 
  file_name = NULL, 
  return_plots = FALSE, 
  cells.highlight = NULL, 
  reorder_datasets = NULL, 
  ggplot_default_colors = FALSE, 
  color_seed = 123
)
```

**Arguments**

- `liger_object` *liger* `liger_object`. Need to perform clustering and factorization before calling this function.
- `num_genes` Number of genes to display for each factor (Default 8).
- `colors_use_factors` Colors to use for plotting factor loadings. By default datasets will be plotted using "varibow" with shuffle = TRUE from both from `DiscretePalette_scCustomize`. 

colors_use_dimreduc
colors to use for plotting factor loadings on dimensionality reduction coordinates (tSNE/UMAP). Default is c('lemonchiffon', 'red'),

pt.size_factors
Adjust point size for plotting in the factor plots.

pt.size_dimreduc
Adjust point size for plotting in dimensionality reduction plots.

reduction_label
What to label the x and y axes of resulting plots. LIGER does not store name of technique and therefore needs to be set manually. Default is 'UMAP'.

raster
Convert points to raster format. Default is NULL which will rasterize by default if greater than 200,000 cells.

raster.dpi
Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).

order
logical. Whether to plot higher loading cells on top of cells with lower loading values in the dimensionality reduction plots (Default = FALSE).

plot_dimreduc
logical. Whether to plot factor loadings on dimensionality reduction coordinates. Default is TRUE.

save_plots
logical. Whether to save plots. Default is TRUE

file_path
directory file path and/or file name prefix. Defaults to current wd.

file_name
name suffix to append after sample name.

return_plots
logical. Whether or not to return plots to the environment. (Default is FALSE)

cells.highlight
Names of specific cells to highlight in plot (black) (default NULL).

reorder_datasets
New order to plot datasets in for the factor plots if different from current factor level order in cell.data slot.

ggplot_default_colors
logical. If colors_use_factors = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "varibow" palette.

color_seed
random seed for the palette shuffle if colors_use_factors = NULL. Default = 123.

Value
A list of ggplot/patchwork objects and/or PDF file.

Author(s)
Velina Kozareva (Original code for modified function), Sam Marsh (Added/modified functionality)

References
Based on plotFactors functionality from original LIGER package.
Plot_Cells_per_Sample

Examples

## Not run:
plotFactors_scCustom(liger_object = liger_obj, return_plots = FALSE, plot_dimreduc = TRUE, raster = FALSE, save_plots = TRUE)

## End(Not run)

---

Plot_Cells_per_Sample  Plot Number of Cells/Nuclei per Sample

Description

Plot of total cell or nuclei number per sample grouped by another meta data variable.

Usage

Plot_Cells_per_Sample(
    seurat_object,
    sample_col = "orig.ident",
    group_by = NULL,
    colors_use = NULL,
    plot_title = "Cells/Nuclei per Sample",
    y_axis_label = "Number of Cells",
    x_axis_label = NULL,
    legend_title = NULL,
    x_lab_rotate = TRUE,
    color_seed = 123
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>seurat_object</td>
<td>Seurat object name.</td>
</tr>
<tr>
<td>sample_col</td>
<td>Specify which column in meta.data specifies sample ID (i.e. orig.ident).</td>
</tr>
<tr>
<td>group_by</td>
<td>Column in meta.data slot to group results by (i.e. &quot;Treatment&quot;).</td>
</tr>
<tr>
<td>colors_use</td>
<td>List of colors or color palette to use.</td>
</tr>
<tr>
<td>plot_title</td>
<td>Plot title.</td>
</tr>
<tr>
<td>y_axis_label</td>
<td>Label for y axis.</td>
</tr>
<tr>
<td>x_axis_label</td>
<td>Label for x axis.</td>
</tr>
<tr>
<td>legend_title</td>
<td>Label for plot legend.</td>
</tr>
<tr>
<td>x_lab_rotate</td>
<td>logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.</td>
</tr>
<tr>
<td>color_seed</td>
<td>random seed for the &quot;varibow&quot; palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.</td>
</tr>
</tbody>
</table>
**Plot_Density_Custom**

**Value**
A ggplot object

**Examples**
```r
## Not run:
Plot_Cells_per_Sample(seurat_object = obj, sample_col = "orig.ident", group_by = "Treatment")
## End(Not run)
```

---

**Plot_Density_Custom**  
*Nebulosa Density Plot*

**Description**
Allow for customization of Nebulosa plot_density. Requires Nebulosa package from Bioconductor.

**Usage**
```r
Plot_Density_Custom(
  seurat_object, 
  features, 
  joint = FALSE, 
  viridis_palette = "magma", 
  custom_palette = NULL, 
  pt.size = 1, 
  reduction = NULL, 
  combine = TRUE, 
  ...
)
```

**Arguments**
- `seurat_object`: Seurat object name.
- `features`: Features to plot.
- `joint`: logical. Whether to return joint density plot. Default is FALSE.
- `viridis_palette`: default viridis palette to use (must be one of: "viridis", "magma", "cividis", "inferno", "plasma"). Default is "magma".
- `custom_palette`: non-default color palette to be used in place of default viridis options.
- `pt.size`: Adjust point size for plotting.
- `reduction`: Dimensionality Reduction to use (if NULL then defaults to Object default).
- `combine`: Create a single plot? If FALSE, a list with ggplot objects is returned.
- `...`: Extra parameters passed to `plot_density`.
Value

A ggplot object

Examples

```r
library(Seurat)
Plot_Density_Custom(seurat_object = pbmc_small, features = "CD3E")
```

Description

Return only the joint density plot from Nebulosa plot_density function. Requires Nebulosa package from Bioconductor.

Usage

```r
Plot_Density_Joint_Only(
  seurat_object,
  features,
  viridis_palette = "magma",
  custom_palette = NULL,
  pt.size = 1,
  reduction = NULL,
  ...
)
```

Arguments

- `seurat_object`: Seurat object name.
- `features`: Features to plot.
- `viridis_palette`: default viridis palette to use (must be one of: "viridis", "magma", "cividis", "inferno", "plasma"). Default is "magma".
- `custom_palette`: non-default color palette to be used in place of default viridis options.
- `pt.size`: Adjust point size for plotting.
- `reduction`: Dimensionality Reduction to use (if NULL then defaults to Object default).
- `...`: Extra parameters passed to `plot_density`.
Value
A ggplot object

Examples

library(Seurat)
Plot_Density_Joint_Only(seurat_object = pbmc_small, features = c("CD8A", "CD3E"))

Plot_Median_Genes
Plot Median Genes per Cell per Sample

Description
Plot of median genes per cell per sample grouped by desired meta data variable.

Usage

Plot_Median_Genes(
  seurat_object,
  sample_col = "orig.ident",
  group_by = NULL,
  colors_use = NULL,
  plot_title = "Median Genes/Cell per Sample",
  y_axis_label = "Median Genes",
  x_axis_label = NULL,
  legend_title = NULL,
  x_lab_rotate = TRUE,
  color_seed = 123
)

Arguments

seurat_object Seurat object name.
sample_col Specify which column in meta.data specifies sample ID (i.e. orig.ident).
group_by Column in meta.data slot to group results by (i.e. "Treatment").
colors_use List of colors or color palette to use. Only applicable if group_by is not NULL.
plot_title Plot title.
y_axis_label Label for y axis.
x_axis_label Label for x axis.
legend_title Label for plot legend.
x_lab_rotate logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.
color_seed random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.
Value
A ggplot object

Examples

```r
library(Seurat)
# Create example groups
pbmc_small$sample_id <- sample(c("sample1", "sample2"), size = ncol(pbmc_small), replace = TRUE)

# Plot
Plot_Median_Genes(seurat_object = pbmc_small, sample_col = "orig.ident", group_by = "sample_id")
```

Description
Plot of median percent mito per cell per sample grouped by desired meta data variable.

Usage

```r
Plot_Median_Mito(
  seurat_object,
  sample_col = "orig.ident",
  group_by = NULL,
  colors_use = NULL,
  plot_title = "Median % Mito per Sample",
  y_axis_label = "Percent Mitochondrial Reads",
  x_axis_label = NULL,
  legend_title = NULL,
  x_lab_rotate = TRUE,
  color_seed = 123
)
```

Arguments

- `seurat_object`: Seurat object name.
- `sample_col`: Specify which column in meta.data specifies sample ID (i.e. orig.ident).
- `group_by`: Column in meta.data slot to group results by (i.e. "Treatment").
- `colors_use`: List of colors or color palette to use. Only applicable if `group_by` is not NULL.
- `plot_title`: Plot title.
- `y_axis_label`: Label for y axis.
- `x_axis_label`: Label for x axis.
- `legend_title`: Label for plot legend.
x_lab_rotate logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.
color_seed random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.

Value
A ggplot object

Examples
## Not run:
# Add mito
obj <- Add_Mito_Ribo_Seurat(seurat_object = obj, species = "human")

# Plot
Plot_Median_Mito(seurat_object = obj, sample_col = "orig.ident", group_by = "sample_id")
## End(Not run)

Plot_Median_Other Plot Median other variable per Cell per Sample

Description
Plot of median other variable per cell per sample grouped by desired meta data variable.

Usage
Plot_Median_Other(
  seurat_object,
  median_var,
  sample_col = "orig.ident",
  group_by = NULL,
  colors_use = NULL,
  plot_title = NULL,
  y_axis_label = NULL,
  x_axis_label = NULL,
  legend_title = NULL,
  x_lab_rotate = TRUE,
  color_seed = 123
)

Arguments
seurat_object Seurat object name.
median_var Variable in meta.data slot to calculate and plot median values for.
Plot_Median_UMIs

Plot Median UMIs per Cell per Sample

Description

Plot of median UMIs per cell per sample grouped by desired meta data variable.

Usage

Plot_Median_UMIs(
  seurat_object,  
  sample_col = "orig.ident",  
  group_by = NULL,  
  colors_use = NULL,  
  plot_title = "Median UMIs/Cell per Sample",  
)

sample_col Specify which column in meta.data specifies sample ID (i.e. orig.ident).
group_by Column in meta.data slot to group results by (i.e. "Treatment").
colors_use List of colors or color palette to use. Only applicable if group_by is not NULL.
plot_title Plot title.
y_axis_label Label for y axis.
x_axis_label Label for x axis.
legend_title Label for plot legend.
x_lab_rotate logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.
color_seed random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.

Value

A ggplot object

Examples

## Not run:
library(Seurat)

###

cd_features <- list(c('CD79B', 'CD79A', 'CD19', 'CD200', 'CD3D', 'CD2', 'CD3E', 'CD7', 'CD8A', 'CD14', 'CD1C', 'CD68', 'CD9', 'CD247'))

pbmc_small <- AddModuleScore(object = pbmc_small, features = cd_features, ctrl = 5, name = 'CD_Features')

Plot_Median_Other(seurat_object = pbmc_small, median_var = "CD_Features1", sample_col = "orig.ident", group_by = "Treatment")

## End(Not run)
```r
y_axis_label = "Median UMIs",
x_axis_label = NULL,
legend_title = NULL,
x_lab_rotate = TRUE,
color_seed = 123
)

Arguments

  seurat_object  Seurat object name.
  sample_col     Specify which column in meta.data specifies sample ID (i.e. orig.ident).
  group_by       Column in meta.data slot to group results by (i.e. "Treatment").
  colors_use     List of colors or color palette to use. Only applicable if group_by is not NULL.
  plot_title     Plot title.
  y_axis_label   Label for y axis.
  x_axis_label   Label for x axis.
  legend_title   Label for plot legend.
  x_lab_rotate   logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.
  color_seed     random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.

Value

  A ggplot object

Examples

  library(Seurat)
  # Create example groups
  pbmc_small$sample_id <- sample(c("sample1", "sample2"), size = ncol(pbmc_small), replace = TRUE)
  # Plot
  Plot_Median_UMIs(seurat_object = pbmc_small, sample_col = "orig.ident", group_by = "sample_id")
```

---

**Pull_Cluster_Annotation**

*Pull cluster information from annotation csv file.*

**Description**

short cut filter and pull function compatible with annotation files created by `Create_Cluster_Annotation_File` by default but also any other csv file.
Usage

Pull_Cluster_Annotation(
    annotation = NULL,
    cluster_name_col = "cluster",
    cell_type_col = "cell_type"
)

Arguments

annotation name of the data.frame/tibble or path to CSV file containing cluster annotation.
cluster_name_col name of column containing cluster names/numbers (default is "cluster").
cell_type_col name of column contain the cell type annotation (default is "cell_type").

Value

a list of named vectors for every cell type in the cell_type_col column of the annotation table and vectors new cluster names (for use with Rename_Clusters function or manual identity renaming).

Examples

## Not run:
# If pulling from a data.frame/tibble
cluster_annotation <- Pull_Cluster_Annotation(annotation = annotation_df,
    cluster_name_col = "cluster", cell_type_col = "cell_type")

# If pulling from csv file
cluster_annotation <- Pull_Cluster_Annotation(annotation = "file_path/file_name.csv",
    cluster_name_col = "cluster", cell_type_col = "cell_type")

## End(Not run)

---

Pull_Directory_List

Description

Enables easy listing of all sub-directories for use as input library lists in Read10X multi functions.

Usage

Pull_Directory_List(base_path)

Arguments

base_path path to the parent directory which contains all of the subdirectories of interest.
Value

A vector of sub-directories within base_path.

Examples

```r
## Not run:
data_dir <- 'path/to/data/directory'
library_list <- Pull_Directory_List(base_path = data_dir)
## End(Not run)
```

Description

Custom VlnPlot for initial QC checks including lines for thresholding

Usage

```r
QC_Plots_Combined_Vln(
  seurat_object,
  group.by = NULL,
  feature_cutoffs = NULL,
  UMI_cutoffs = NULL,
  mito_cutoffs = NULL,
  mito_name = "percent_mito",
  pt.size = NULL,
  colors_use = NULL,
  x_lab_rotate = TRUE,
  y_axis_log = FALSE,
  raster = NULL,
  ggplot_default_colors = FALSE,
  color_seed = 123,
  ...
)
```

Arguments

- **seurat_object**: Seurat object name.
- **group.by**: Name of one or more metadata columns to group (color) cells by (for example, orig.ident); default is the current active.ident of the object.
- **feature_cutoffs**: Numeric vector of length 1 or 2 to plot lines for potential low/high threshold for filtering.
** QC_Plots_Complexity **

** QC Plots Cell "Complexity"**

**Description**

Custom VlnPlot for initial QC checks including lines for thresholding.

**Value**

A ggplot object

**Examples**

```r
## Not run:
QC_Plots_Combined_Vln(seurat_object = object)
## End(Not run)
```
Usage

QC_Plots_Complexity(
  seurat_object,
  feature = "log10GenesPerUMI",
  group.by = NULL,
  x_axis_label = NULL,
  y_axis_label = "log10(Genes) / log10(UMIs)",
  plot_title = "Cell Complexity",
  low_cutoff = NULL,
  high_cutoff = NULL,
  pt.size = NULL,
  colors_use = NULL,
  x_lab_rotate = TRUE,
  y_axis_log = FALSE,
  raster = NULL,
  ggplot_default_colors = FALSE,
  color_seed = 123,
  ...
)

Arguments

  seurat_object  Seurat object name.
  feature        Feature from Meta Data to plot.
  group.by       Name of one or more metadata columns to group (color) cells by (for example, orig.ident); default is the current active.ident of the object.
  x_axis_label   Label for x axis.
  y_axis_label   Label for y axis.
  plot_title     Plot Title.
  low_cutoff     Plot line a potential low threshold for filtering.
  high_cutoff    Plot line a potential high threshold for filtering.
  pt.size        Point size for plotting
  colors_use     vector of colors to use for plot.
  x_lab_rotate   Rotate x-axis labels 45 degrees (Default is TRUE).
  y_axis_log     logical. Whether to change y axis to log10 scale (Default is FALSE).
  raster         Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 total points plotted (# Cells x # of features).
  ggplot_default_colors
                  logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.
  color_seed     random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.
  ...            Extra parameters passed to VlnPlot.
**QC_Plots_Feature**

**Value**
A ggplot object

**Examples**
```
library(Seurat)
pbmc_small <- Add_Cell_Complexity_Seurat(pbmc_small)

QC_Plots_Complexity(seurat_object = pbmc_small)
```

**Description**
Custom VlnPlot for initial QC checks including lines for thresholding

**Usage**
```
QC_Plots_Feature(
  seurat_object,
  feature,
  group.by = NULL,
  x_axis_label = NULL,
  y_axis_label = NULL,
  plot_title = NULL,
  low_cutoff = NULL,
  high_cutoff = NULL,
  pt.size = NULL,
  colors_use = NULL,
  x_lab_rotate = TRUE,
  y_axis_log = FALSE,
  raster = NULL,
  ggplot_default_colors = FALSE,
  color_seed = 123,
  ...
)
```

**Arguments**
- `seurat_object`  Seurat object name.
- `feature` Feature from Meta Data to plot.
- `group.by` Name of one or more metadata columns to group (color) cells by (for example, orig.ident); default is the current active.ident of the object.
- `x_axis_label` Label for x axis.
y_axis_label   Label for y axis.
plot_title     Plot Title.
low_cutoff     Plot line a potential low threshold for filtering.
high_cutoff    Plot line a potential high threshold for filtering.
pt.size       Point size for plotting
colors_use     vector of colors to use for plot.
x_lab_rotate   Rotate x-axis labels 45 degrees (Default is TRUE).
y_axis_log     logical. Whether to change y axis to log10 scale (Default is FALSE).
raster         Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 total points plotted (# Cells x # of features).
ggplot_default_colors
               logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.
color_seed     random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.
...            Extra parameters passed to VlnPlot.

Value

A ggplot object

Examples

## Not run:
QC_Plots_Feature(seurat_object = object, feature = "FEATURE_NAME",
y_axis_label = "FEATURE per Cell", plot_title = "FEATURE per Cell", high_cutoff = 10,
low_cutoff = 2)

## End(Not run)
QC_Plots_Genes

Usage

QC_Plots_Genes(
  seurat_object,
  plot_title = "Genes Per Cell/Nucleus",
  group.by = NULL,
  x_axis_label = NULL,
  y_axis_label = "Features",
  low_cutoff = NULL,
  high_cutoff = NULL,
  pt.size = NULL,
  colors_use = NULL,
  x_lab_rotate = TRUE,
  y_axis_log = FALSE,
  raster = NULL,
  ggplot_default_colors = FALSE,
  color_seed = 123,
  ...
)

Arguments

  seurat_object  Seurat object name.
  plot_title     Plot Title.
  group.by       Name of one or more metadata columns to group (color) cells by (for example, orig.ident); default is the current active.ident of the object.
  x_axis_label   Label for x axis.
  y_axis_label   Label for y axis.
  low_cutoff     Plot line a potential low threshold for filtering.
  high_cutoff    Plot line a potential high threshold for filtering.
  pt.size        Point size for plotting
  colors_use     vector of colors to use for plot.
  x_lab_rotate   Rotate x-axis labels 45 degrees (Default is TRUE).
  y_axis_log     logical. Whether to change y axis to log10 scale (Default is FALSE).
  raster         Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 total points plotted (# Cells x # of features).
  ggplot_default_colors
                   logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.
  color_seed      random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.
  ...             Extra parameters passed to VlnPlot.

Value

A ggplot object
Examples

library(Seurat)
QC_Plots_Genes(seurat_object = pbmc_small, plot_title = "Genes per Cell", low_cutoff = 40,
high_cutoff = 85)

QC_Plots_Mito

Description

#' Custom VlnPlot for initial QC checks including lines for thresholding

Usage

QC_Plots_Mito(
  seurat_object,
  mito_name = "percent_mito",
  plot_title = "Mito Gene % per Cell/Nucleus",
  group.by = NULL,
  x_axis_label = NULL,
  y_axis_label = "% Mitochondrial Gene Counts",
  low_cutoff = NULL,
  high_cutoff = NULL,
  pt.size = NULL,
  colors_use = NULL,
  x_lab_rotate = TRUE,
  y_axis_log = FALSE,
  raster = NULL,
  ggplot_default_colors = FALSE,
  color_seed = 123,
  ...
)

Arguments

seurat_object Seurat object name.
mito_name The column name containing percent mitochondrial counts information. Default value is "percent_mito" which is default value created when using Add_Mito_Ribo_Seurat().
plot_title Plot Title.
group.by Name of one or more metadata columns to group (color) cells by (for example, orig.ident); default is the current active.ident of the object.
x_axis_label Label for x axis.
y_axis_label Label for y axis.
low_cutoff Plot line a potential low threshold for filtering.
QC_Plots_UMIs

high_cutoff  Plot line a potential high threshold for filtering.
pt.size     Point size for plotting
colors_use  vector of colors to use for plot.
x_lab_rotate Rotate x-axis labels 45 degrees (Default is TRUE).
y_axis_log  logical. Whether to change y axis to log10 scale (Default is FALSE).
raster      Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 total points plotted (# Cells x # of features).
ggplot_default_colors logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.
color_seed  random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.
...         Extra parameters passed to VlnPlot.

Value
A ggplot object

Examples

## Not run:
QC_Plots_Mito(seurat_object = object, plot_title = "Percent Mito per Cell", high_cutoff = 10)

## End(Not run)

---

QC_Plots_UMIs

QC Plots UMIs

Description

#* Custom VlnPlot for initial QC checks including lines for thresholding

Usage

QC_Plots_UMIs(
  seurat_object,
  plot_title = "UMIs per Cell/Nucleus",
  group.by = NULL,
  x_axis_label = NULL,
  y_axis_label = "UMIs",
  low_cutoff = NULL,
  high_cutoff = NULL,
  pt.size = NULL,
  colors_use = NULL,
  ...
x_lab_rotate = TRUE,
y_axis_log = FALSE,
raster = NULL,
ggplot_default_colors = FALSE,
color_seed = 123,
...}

Arguments

seurat_object Seurat object name.
plot_title Plot Title.
group.by Name of one or more metadata columns to group (color) cells by (for example, orig.ident); default is the current active.ident of the object.
x_axis_label Label for x axis.
y_axis_label Label for y axis.
low_cutoff Plot line a potential low threshold for filtering.
high_cutoff Plot line a potential high threshold for filtering.
pt.size Point size for plotting
colors_use vector of colors to use for plot.
x_lab_rotate Rotate x-axis labels 45 degrees (Default is TRUE).
y_axis_log logical. Whether to change y axis to log10 scale (Default is FALSE).
raster Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 total points plotted (# Cells x # of features).
ggplot_default_colors logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.
color_seed random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.
...

Value

A ggplot object

Examples

library(Seurat)
QC_Plots_UMIs(seurat_object = pbmc_small, plot_title = "UMIs per Cell", low_cutoff = 75, high_cutoff = 600)
QC Plots Genes vs Misc

Description

Custom FeatureScatter for initial QC checks including lines for thresholding

Usage

QC_Plot_GenevsFeature(
    seurat_object,
    feature1,
    x_axis_label = NULL,
    y_axis_label = "Genes per Cell/Nucleus",
    low_cutoff_gene = NULL,
    high_cutoff_gene = NULL,
    low_cutoff_feature = NULL,
    high_cutoff_feature = NULL,
    colors_use = NULL,
    pt.size = 1,
    group.by = NULL,
    raster = NULL,
    raster.dpi = c(512, 512),
    ggplot_default_colors = FALSE,
    color_seed = 123,
    shuffle_seed = 1,
    ...
)

Arguments

seurat_object  Seurat object name.
feature1        First feature to plot.
x_axis_label   Label for x axis.
y_axis_label   Label for y axis.
low_cutoff_gene Plot line a potential low threshold for filtering genes per cell.
high_cutoff_gene Plot line a potential high threshold for filtering genes per cell.
low_cutoff_feature Plot line a potential low threshold for filtering feature1 per cell.
high_cutoff_feature Plot line a potential high threshold for filtering feature1 per cell.
colors_use    vector of colors to use for plotting by identity.
pt.size       Adjust point size for plotting.
group.by Name of one or more metadata columns to group (color) cells by (for example, orig.ident). Default is @active.ident.

raster Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 cells.

raster.dpi Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).

ggplot_default_colors logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.

color_seed random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.

shuffle_seed Sets the seed if randomly shuffling the order of points (Default is 1).

... Extra parameters passed to FeatureScatter.

Value A ggplot object

Examples

## Not run:
QC_Plot_GenevsFeature(seurat_object = obj, y_axis_label = "Feature per Cell")

## End(Not run)
group.by = NULL,
raster = NULL,
raster.dpi = c(512, 512),
ggplot.default.colors = FALSE,
color.seed = 123,
shuffle.seed = 1,
...
)

Arguments

seurat.object Seurat object name.
feature1 First feature to plot.
x.axis.label Label for x axis.
y.axis.label Label for y axis.
low_cutoff.UMI Plot line a potential low threshold for filtering UMI per cell.
high_cutoff.UMI Plot line a potential high threshold for filtering UMI per cell.
low_cutoff.feature Plot line a potential low threshold for filtering feature1 per cell.
high_cutoff.feature Plot line a potential high threshold for filtering feature1 per cell.
colors.use vector of colors to use for plotting by identity.
pt.size Adjust point size for plotting.
group.by Name of one or more metadata columns to group (color) cells by (for example, orig.ident). Default is @active.ident.
raster Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 cells.
raster.dpi Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).
ggplot.default.colors logical. If colors.use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.
color.seed random seed for the "varibow" palette shuffle if colors.use = NULL and number of groups plotted is greater than 36. Default = 123.
shuffle.seed Sets the seed if randomly shuffling the order of points (Default is 1).
...

Value

A ggplot object
Examples

## Not run:
QC_Plot_UMIvsFeature(seurat_object = obj, y_axis_label = "Feature per Cell")

## End(Not run)

QC_Plot_UMIvsGene  

**QC Plots Genes vs UMI**

**Description**

Custom FeatureScatter for initial QC checks including lines for thresholding

**Usage**

```r
QC_Plot_UMIvsGene(
    seurat_object,
    x_axis_label = "UMIs per Cell/Nucleus",
    y_axis_label = "Genes per Cell/Nucleus",
    low_cutoff_gene = -Inf,
    high_cutoff_gene = Inf,
    low_cutoff_UMI = -Inf,
    high_cutoff_UMI = Inf,
    colors_use = NULL,
    meta_gradient_name = NULL,
    meta_gradient_color = viridis_plasma_dark_high,
    meta_gradient_na_color = "lightgray",
    meta_gradient_low_cutoff = NULL,
    cells = NULL,
    combination = FALSE,
    pt.size = 1,
    group.by = NULL,
    raster = NULL,
    raster.dpi = c(512, 512),
    ggplot_default_colors = FALSE,
    color_seed = 123,
    shuffle_seed = 1,
    ...
)
```

**Arguments**

- `seurat_object`: Seurat object name.
- `x_axis_label`: Label for x axis.
- `y_axis_label`: Label for y axis.
low_cutoff_gene
Plot line a potential low threshold for filtering genes per cell.

high_cutoff_gene
Plot line a potential high threshold for filtering genes per cell.

low_cutoff_UMI
Plot line a potential low threshold for filtering UMIs per cell.

high_cutoff_UMI
Plot line a potential high threshold for filtering UMIs per cell.

colors_use
vector of colors to use for plotting by identity.

meta_gradient_name
Name of continuous meta data variable to color points in plot by. (MUST be continuous variable i.e. "percent_mito").

meta_gradient_color
The gradient color palette to use for plotting of meta variable (default is viridis "Plasma" palette with dark colors high).

meta_gradient_na_color
Color to use for plotting values when a meta_gradient_low_cutoff is set (default is "lightgray").

meta_gradient_low_cutoff
Value to use as threshold for plotting. meta_gradient_name values below this value will be plotted using meta_gradient_na_color.

cells
Cells to include on the scatter plot (default is all cells).

combination
logical (default FALSE). Whether or not to return a plot layout with both the plot colored by identity and the meta data gradient plot.

pt.size
Passes size of points to both FeatureScatter and geom_point.

group.by
Name of one or more metadata columns to group (color) cells by (for example, orig.ident). Default is @active.ident.

raster
Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 cells.

raster.dpi
Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).

ggplot_default_colors
logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.

color_seed
Random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.

shuffle_seed
Sets the seed if randomly shuffling the order of points (Default is 1).

...
Extra parameters passed to FeatureScatter.

Value
A ggplot object
Examples

library(Seurat)
QC_Plot_UMIvsGene(seurat_object = pbmc_small, x_axis_label = "UMIs per Cell/Nucleus",
y_axis_label = "Genes per Cell/Nucleus")

---

Description

Enables easy loading of sparse data matrices provided by 10X genomics. That have file prefixes added to them by NCBI GEO or other repos.

Usage

Read10X_GEO(
  data_dir = NULL,
  sample_list = NULL,
  sample_names = NULL,
  gene.column = 2,
  cell.column = 1,
  unique.features = TRUE,
  strip.suffix = FALSE,
  parallel = FALSE,
  num_cores = NULL,
  merge = FALSE
)

Arguments

data_dir Directory containing the matrix.mtx, genes.tsv (or features.tsv), and barcodes.tsv files provided by 10X.
sample_list A vector of file prefixes/names if specific samples are desired. Default is NULL and will load all samples in given directory.
sample_names a set of sample names to use for each sample entry in returned list. If NULL will set names to the file name of each sample.
gene.column Specify which column of genes.tsv or features.tsv to use for gene names; default is 2.
cell.column Specify which column of barcodes.tsv to use for cell names; default is 1.
unique.features Make feature names unique (default TRUE).
strip.suffix Remove trailing "-1" if present in all cell barcodes.
parallel logical (default FALSE). Whether to use multiple cores when reading in data. Only possible on Linux based systems.
num_cores if parallel = TRUE indicates the number of cores to use for multicore processing.
merge logical (default FALSE) whether or not to merge samples into a single matrix or return list of matrices. If TRUE each sample entry in list will have cell barcode prefix added. The prefix will be taken from sample_names.

Value

If features.csv indicates the data has multiple data types, a list containing a sparse matrix of the data from each type will be returned. Otherwise a sparse matrix containing the expression data will be returned.

References

Code used in function has been slightly modified from Seurat::Read10X function of Seurat package https://github.com/satijalab/seurat (License: GPL-3). Function was modified to support file prefixes and altered loop by Samuel Marsh for scCustomize (also previously posted as potential PR to Seurat GitHub).

Examples

```r
## Not run:
data_dir <- 'path/to/data/directory'
expression_matrices <- Read10X_GEO(data_dir = data_dir)
# To create object from single file
seurat_object = CreateSeuratObject(counts = expression_matrices[[1]])
## End(Not run)
```

Read10X_h5_GEO **Load in NCBI GEO data from 10X in HDF5 file format**

Description

Enables easy loading of HDF5 data matrices provided by 10X genomics. That have file prefixes added to them by NCBI GEO or other repos or programs (i.e. Cell Bender)

Usage

```r
Read10X_h5_GEO(
  data_dir = NULL,
  sample_list = NULL,
  sample_names = NULL,
  shared_suffix = NULL,
  parallel = FALSE,
  num_cores = NULL,
  merge = FALSE,
  ...)
```
Arguments

data_dir  Directory containing the .h5 files provided by 10X.
sample_list  A vector of file prefixes/names if specific samples are desired. Default is NULL and will load all samples in given directory.
sample_names  a set of sample names to use for each sample entry in returned list. If NULL will set names to the file name of each sample.
shared_suffix a suffix and file extension shared by all samples.
parallel  logical (default FALSE). Whether to use multiple cores when reading in data. Only possible on Linux based systems.
num_cores if parallel = TRUE indicates the number of cores to use for multicore processing.
merge  logical (default FALSE) whether or not to merge samples into a single matrix or return list of matrices. If TRUE each sample entry in list will have cell barcode prefix added. The prefix will be taken from sample_names.

...  Additional arguments passed to Read10X_h5

Value

If the data has multiple data types, a list containing a sparse matrix of the data from each type will be returned. Otherwise a sparse matrix containing the expression data will be returned.

Examples

```r
# Not run:
data_dir <- 'path/to/data/directory'
expression_matrices <- Read10X_h5_GEO(data_dir = data_dir)
# To create object from single file
seurat_object = CreateSeuratObject(counts = expression_matrices[[1]])
```

Description

Enables easy loading of sparse data matrices provided by 10X genomics that are present in multiple subdirectories. Can function with either default output directory structure of Cell Ranger or custom directory structure.
Read10X_h5_Multi_Directory

Usage

Read10X_h5_Multi_Directory(
  base_path,
  secondary_path = NULL,
  default_10X_path = TRUE,
  h5_filename = "filtered_feature_bc_matrix.h5",
  cell_bender = FALSE,
  sample_list = NULL,
  sample_names = NULL,
  replace_suffix = FALSE,
  new_suffix_list = NULL,
  parallel = FALSE,
  num_cores = NULL,
  merge = FALSE,
  ...
)

Arguments

base_path path to the parent directory which contains all of the subdirectories of interest.
secondary_path path from the parent directory to count matrix files for each sample.
default_10X_path logical (default TRUE) sets the secondary path variable to the default 10X directory structure.
h5_filename name of h5 file (including .h5 suffix). If all h5 files have same name (i.e. Cell Ranger output) then use full file name. By default function uses Cell Ranger name: "filtered_feature_bc_matrix.h5". If h5 files have sample specific prefixes (i.e. from Cell Bender) then use only the shared part of file name (e.g., "_filtered_out.h5").
cell_bender logical (default FALSE). Is the h5 file from cell bender output, needed to set correct file names.
sample_list a vector of sample directory names if only specific samples are desired. If NULL will read in subdirectories in parent directory.
sample_names a set of sample names to use for each sample entry in returned list. If NULL will set names to the subdirectory name of each sample.
replace_suffix logical (default FALSE). Whether or not to replace the barcode suffixes of matrices using Replace_Suffix.
new_suffix_list a vector of new suffixes to replace existing suffixes if replace_suffix = TRUE. See Replace_Suffix for more information. To remove all suffixes set new_suffix_list = ""
parallel logical (default FALSE) whether or not to use multi core processing to read in matrices.
um_cores how many cores to use for parallel processing.
merge logical (default FALSE) whether or not to merge samples into a single matrix or return list of matrices. If TRUE each sample entry in list will have cell barcode prefix added. The prefix will be taken from sample_names.

... Extra parameters passed to Read10X_h5.

Value

a list of sparse matrices (merge = FALSE) or a single sparse matrix (merge = TRUE).

Examples

## Not run:
base_path <- 'path/to/data/directory'
expression_matrices <- Read10X_h5_Multi_Directory(base_path = base_path)

## End(Not run)
### Read_CellBender_h5_Mat

Load CellBender h5 matrices (corrected)

#### Description

Extract sparse matrix with corrected counts from CellBender h5 output file.

#### Usage

```
Read_CellBender_h5_Mat(file_name, use.names = TRUE, unique.features = TRUE)
```

#### Arguments

- **file_name**: Path to h5 file.
- **use.names**: Label row names with feature names rather than ID numbers (default TRUE).
- **unique.features**: Make feature names unique (default TRUE).

---

**default_10X_path**

logical (default TRUE) sets the secondary path variable to the default 10X directory structure.

**sample_list**

a vector of sample directory names if only specific samples are desired. If NULL will read in subdirectories in parent directory.

**sample_names**

a set of sample names to use for each sample entry in returned list. If NULL will set names to the subdirectory name of each sample.

**parallel**

logical (default FALSE) whether or not to use multi core processing to read in matrices.

**num_cores**

how many cores to use for parallel processing.

**merge**

logical (default FALSE) whether or not to merge samples into a single matrix or return list of matrices. If TRUE each sample entry in list will have cell barcode prefix added. The prefix will be taken from `sample_names`.

Extra parameters passed to `Read10X`.

#### Value

a list of sparse matrices (merge = FALSE) or a single sparse matrix (merge = TRUE).

#### Examples

```r
## Not run:
base_path <- '/path/to/data/directory'
expression_matrices <- Read10X_Multi_Directory(base_path = base_path)
## End(Not run)
```

---

**Read_CellBender_h5_Mat**

Load CellBender h5 matrices (corrected)

#### Description

Extract sparse matrix with corrected counts from CellBender h5 output file.

#### Usage

```
Read_CellBender_h5_Mat(file_name, use.names = TRUE, unique.features = TRUE)
```

#### Arguments

- **file_name**: Path to h5 file.
- **use.names**: Label row names with feature names rather than ID numbers (default TRUE).
- **unique.features**: Make feature names unique (default TRUE).
Value

sparse matrix

References

Code used in function has been modified from Seurat::Read10X_h5 function of Seurat package https://github.com/satijalab/seurat (License: GPL-3).

Examples

## Not run:
mat <- Read_CellBender_h5_Mat(file_name = "/SampleA_out_filtered.h5")

## End(Not run)
filtered_h5  logical (default TRUE). Will set the shared file name suffix custom_name is NULL.

custom_name  if file name was customized in CellBender then this parameter should contain the portion of file name that is shared across all samples. Must included the ".h5" extension as well.

sample_list  a vector of sample directory names if only specific samples are desired. If NULL will read in subdirectories in parent directory.

sample_names  a set of sample names to use for each sample entry in returned list. If NULL will set names to the subdirectory name of each sample.

replace_suffix  logical (default FALSE). Whether or not to replace the barcode suffixes of matrices using Replace_Suffix.

new_suffix_list  a vector of new suffixes to replace existing suffixes if replace_suffix = TRUE. See Replace_Suffix for more information. To remove all suffixes set new_suffix_list = "".

parallel  logical (default FALSE) whether or not to use multi core processing to read in matrices.

num_cores  how many cores to use for parallel processing.

merge  logical (default FALSE) whether or not to merge samples into a single matrix or return list of matrices. If TRUE each sample entry in list will have cell barcode prefix added. The prefix will be taken from sample_names.

...  Extra parameters passed to Read_CellBender_h5_Mat.

Value
list of sparse matrices

Examples

## Not run:
base_path <- 'path/to/data/directory'
mat_list <- Read_CellBender_h5_Multi_Directory(base_path = base_path)

## End(Not run)
Usage

Read_CellBender_h5_Multi_File(
  data_dir = NULL,
  filtered_h5 = TRUE,
  custom_name = NULL,
  sample_list = NULL,
  sample_names = NULL,
  parallel = FALSE,
  num_cores = NULL,
  merge = FALSE,
  ...
)

Arguments

data_dir Directory containing the .h5 files output by CellBender.
filtered_h5 logical (default TRUE). Will set the shared file name suffix if custom_name is NULL.
custom_name if file name was customized in CellBender then this parameter should contain the portion of file name that is shared across all samples. Must included the ".h5" extension as well.
sample_list a vector of sample names if only specific samples are desired. If NULL will read in all files within data_dir directory.
sample_names a set of sample names to use for each sample entry in returned list. If NULL will set names to the subdirectory name of each sample.
parallel logical (default FALSE) whether or not to use multi core processing to read in matrices.
num_cores how many cores to use for parallel processing.
merge logical (default FALSE) whether or not to merge samples into a single matrix or return list of matrices. If TRUE each sample entry in list will have cell barcode prefix added. The prefix will be taken from sample_names.
...
Extra parameters passed to Read_CellBender_h5_Mat.

Value

list of sparse matrices

Examples

```r
## Not run:
base_path <- 'path/to/data/directory'
mat_list <- Read_CellBender_h5_Multi_File(data_dir = base_path)

## End(Not run)
```
Description

Can read delimited file types (i.e. csv, tsv, txt)

Usage

```r
Read_GEO_Delim(
  data_dir,
  file_suffix,
  move_genes_rownames = TRUE,
  sample_list = NULL,
  full_names = FALSE,
  sample_names = NULL,
  barcode_suffix_period = FALSE,
  parallel = FALSE,
  num_cores = NULL,
  merge = FALSE
)
```

Arguments

data_dir          Directory containing the files.
file_suffix       The file suffix of the individual files. Must be the same across all files being imported. This is used to detect files to import and their GEO IDs.
move_genes_rownames logical. Whether gene IDs are present in first column or in row names of delimited file. If TRUE will move the first column to row names before creating final matrix. Default is TRUE.
sample_list       a vector of samples within directory to read in (can be either with or without file_suffix see full_names). If NULL will read in all subdirectories.
full_names        logical (default FALSE). Whether or not the sample_list vector includes the file suffix. If FALSE the function will add suffix based on file_suffix parameter.
sample_names      a set of sample names to use for each sample entry in returned list. If NULL will set names to the directory name of each sample.
barcode_suffix_period Is the barcode suffix a period and should it be changed to ".". Default (FALSE; barcodes will be left identical to their format in input files.). If TRUE "." in barcode suffix will be changed to ".".
parallel          logical (default FALSE). Whether to use multiple cores when reading in data. Only possible on Linux based systems.
num_cores if parallel = TRUE indicates the number of cores to use for multicore processing.
merge logical (default FALSE) whether or not to merge samples into a single matrix or return list of matrices. If TRUE each sample entry in list will have cell barcode prefix added. The prefix will be taken from sample_names.

Value
List of gene x cell matrices in list format named by sample name.

Examples
## Not run:
data_dir <- 'path/to/data/directory'
expression_matrices <- Read_GEO_Delim(data_dir = data_dir)
## End(Not run)

Read_Metrics_10X

Get data.frame with all metrics from the Cell Ranger count analysis (present in web_summary.html)

Usage
Read_Metrics_10X(
  base_path,
  secondary_path = NULL,
  default_10X = TRUE,
  lib_list = NULL,
  lib_names = NULL
)

Arguments
base_path path to the parent directory which contains all of the subdirectories of interest.
secondary_path path from the parent directory to count "outs/" folder which contains the "metrics_summary.csv" file.
default_10X logical (default TRUE) sets the secondary path variable to the default 10X directory structure.
lib_list a list of sample names (matching directory names) to import. If NULL will read in all samples in parent directory.
lib_names a set of sample names to use for each sample. If NULL will set names to the directory name of each sample.
**Value**

A data frame with sample metrics from cell ranger.

**Examples**

```r
## Not run:
metrics <- Read_Metrics_10X(base_path = "/path/to/directories", default_10X = TRUE)
## End(Not run)
```

---

**Description**

Wrapper function to rename active identities in Seurat Object with new idents.

**Usage**

```r
Rename_Clusters(seurat_object, new_idents, meta_col_name = NULL, ...)
```

**Arguments**

- `seurat_object` object name.
- `new_idents` vector of new cluster names. Must be equal to the length of current active.ident in Seurat Object. Will accept named vector (with old idents as names) or will name the new.idents vector internally.
- `meta_col_name` (Optional). Whether or not to create new named column in Object@meta.data to store the old identities.
- `...` Extra parameters passed to RenameIdents.

**Value**

Seurat Object with new identities placed in active.ident slot.

**Examples**

```r
## Not run:
obj <- Rename_Clusters(seurat.object = obj_name, new.idents = new.idents_vec, 
meta.col.name = "Round01_Res0.6_Idents")
## End(Not run)
```
Replace_Suffix

Description

Replace barcode suffixes in matrix, data.frame, or list of matrices/data.frames

Usage

Replace_Suffix(data, current_suffix, new_suffix)

Arguments

data: Either matrix/data.frame or list of matrices/data.frames with the cell barcodes in the column names.
current_suffix: a single value or vector of values representing current barcode suffix. If suffix is the same for all matrices/data.frames in list only single value is required.
new_suffix: a single value or vector of values representing new barcode suffix to be added. If desired suffix is the same for all matrices/data.frames in list only single value is required. If no suffix is desired set new_suffix = "".

Value

matrix or data.frame with new column names.

Examples

## Not run:
dge_matrix <- Replace_Suffix(data = dge_matrix, current_suffix = "-1", new_suffix = "-2")
## End(Not run)

scCustomize_Palette

Color Palette Selection for scCustomize

Description

Function to return package default discrete palettes depending on number of groups plotted.

Usage

scCustomize_Palette(num_groups, ggplot_default_colors, color_seed = 123)
Arguments

num_groups  number of groups to be plotted. If ggplot_default_colors = FALSE then by default:
  • If number of levels plotted equal to 2 then colors will be NavyAndOrange().
  • If number of levels plotted greater than 2 but less than or equal to 36 it will use "polychrome" from DiscretePalette_scCustomize.
  • If greater than 36 will use "varibow" with shuffle = TRUE from DiscretePalette_scCustomize.

ggplot_default_colors  logical. Whether to use default ggplot hue palette or not.

color_seed  random seed to use for shuffling the "varibow" palette.

Value

vector of colors to use for plotting.

Examples

cols <- scCustomize_Palette(num_groups = 24, ggplot_default_colors = FALSE)
PalettePlot(palette = cols)

Description

Plot a combined plot of the Alignment QC metrics from sequencing output.

Usage

Seq_QC_Plot_Alignment_Combined(
  metrics_dataframe,
  plot_by = "sample_id",
  colors_use = NULL,
  x_lab_rotate = FALSE,
  patchwork_title = "Sequencing QC Plots: Read Alignment Metrics",
  significance = FALSE,
  ...
)


Seq_QC_Plot_Antisense

Arguments

metrics_dataframe
  data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by
  Grouping factor for the plot. Default is to plot as single group with single point
  per sample.

colors_use
  colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2
  palette if less than 8 groups and DiscretePalette_scCustomize(palette =
  "polychrome") if more than 8.

x_lab_rotate
  logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

patchwork_title
  Title to use for the patchworked plot output.

significance
  logical. Whether to calculate and plot p-value comparisons when plotting by
  grouping factor. Default is FALSE.

...
  Other variables to pass to ggpubr::stat_compare_means when doing signifi-
  cance testing.

Value

A ggplot object

Examples

## Not run:
Seq_QC_Plot_Antisense_Combined(metrics_dataframe = metrics)

## End(Not run)

---

Description

Plot the fraction of reads mapped Antisense to Gene

Usages

Seq_QC_Plot_Antisense(
  metrics_dataframe,
  plot_by = "sample_id",
  colors_use = NULL,
  x_lab_rotate = FALSE,
  significance = FALSE,
  ...
)
Arguments

metrics_dataframe: data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by: Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use: colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.

x_lab_rotate: logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

significance: logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

...: Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value

A ggplot object

Examples

```r
## Not run:
Seq_QC_Plot_Antisense(metrics_dataframe = metrics)

## End(Not run)
```

Description

Plot a combined plot of the basic QC metrics from sequencing output.

Usage

```r
Seq_QC_Plot_Basic_Combined(
  metrics_dataframe,  
  plot_by = "sample_id",  
  colors_use = NULL,  
  x_lab_rotate = FALSE,  
  patchwork_title = "Sequencing QC Plots: Basic Cell Metrics",  
  significance = FALSE,  
  ...
)
```
Arguments

metrics_dataframe
  data.frame containing Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by
  Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use
  Colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.

x_lab_rotate
  Logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

patchwork_title
  Title to use for the patchworked plot output.

significance
  Logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

...
  Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value

A ggplot object

Examples

## Not run:
Seq_QC_Plot_Basic_Combined(metrics_dataframe = metrics)

## End(Not run)
Arguments

metrics_dataframe
  data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by
  Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use
  colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.

x_lab_rotate
  logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

significance
  logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

...
  Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value

A ggplot object

Examples

## Not run:
Seq_QC_Plot_Genes(metrics_dataframe = metrics)

## End(Not run)
Seq_QC_Plot_Genome

Arguments

metrics_dataframe: data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by: Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use: colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.

x_lab_rotate: logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

significance: logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

...: Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value

A ggplot object

Examples

## Not run:
Seq_QC_Plot_Genes(metrics_dataframe = metrics)

## End(Not run)

Seq_QC_Plot_Genome  QC Plots Sequencing metrics (Alignment)

Description

Plot the fraction of reads confidently mapped to genome

Usage

Seq_QC_Plot_Genome(
  metrics_dataframe,
  plot_by = "sample_id",
  colors_use = NULL,
  x_lab_rotate = FALSE,
  significance = FALSE,
  ...
)
Arguments

metrics_dataframe  
data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by  
Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use  
colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and  DiscretePalettescCustomize(palette = "polychrome") if more than 8.

x_lab_rotate  
logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

significance  
logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

...  
Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value

A ggplot object

Examples

```r
## Not run:
Seq_QC_Plot_Genome(metrics_dataframe = metrics)
```

```r
## End(Not run)
```

Seq_QC_Plot_Intergenic

QC Plots Sequencing metrics (Alignment)

Description

Plot the fraction of reads confidently mapped to intergenic regions

Usage

```r
Seq_QC_Plot_Intergenic(
  metrics_dataframe,  
  plot_by = "sample_id",  
  colors_use = NULL,  
  x_lab_rotate = FALSE,  
  significance = FALSE,  
  ...
)
```
Arguments

metrics_dataframe  data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by  Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use  colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.

x_lab_rotate  logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

significance  logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

...  Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value

A ggplot object

Examples

## Not run:
Seq_QC_Plot_Intronic(metrics_dataframe = metrics)

## End(Not run)


**Arguments**

- `metrics_dataframe` : data.frame contain Cell Ranger QC Metrics (see `Read_Metrics_10X`).
- `plot_by` : Grouping factor for the plot. Default is to plot as single group with single point per sample.
- `colors.use` : colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.
- `x_lab_rotate` : logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.
- `significance` : logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.
- `...` : Other variables to pass to `ggpubr::stat_compare_means` when doing significance testing.

**Value**

A ggplot object

**Examples**

```r
## Not run:
Seq_QC_Plot_Intronic(metrics.dataframe = metrics)

## End(Not run)
```

---

**Description**

Plot the number of cells per sample

**Usage**

```r
Seq_QC_Plot_Number_Cells(  
metrics.dataframe,  
plot_by = "sample_id",  
colors.use = NULL,  
ox_lab_rotate = FALSE,  
significance = FALSE,  
...  
)
```
Arguments

metrics_dataframe
data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by
Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use
colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.

x_lab_rotate
logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

significance
logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

... Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value

A ggplot object

Examples

## Not run:
Seq_QC_Plot_Number_Cells(metrics_dataframe = metrics)

## End(Not run)

Seq_QC_Plot_Reads_in_Cells

QC Plots Sequencing metrics

Description

Plot the fraction of reads in cells per sample

Usage

Seq_QC_Plot_Reads_in_Cells(
metrics_dataframe,
plot_by = "sample_id",
colors_use = NULL,
x_lab_rotate = FALSE,
significance = FALSE,
...
)
**Arguments**

- `metrics_dataframe`: data.frame contain Cell Ranger QC Metrics (see `Read_Metrics_10X`).
- `plot_by`: Grouping factor for the plot. Default is to plot as single group with single point per sample.
- `colors_use`: colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.
- `x_lab_rotate`: logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.
- `significance`: logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.
- `...`: Other variables to pass to `ggpubr::stat_compare_means` when doing significance testing.

**Value**

A ggplot object

**Examples**

```r
## Not run:
Seq_QC_Plot_Reads_in_Cells(metrics_dataframe = metrics)

## End(Not run)
```

---

**Description**

Plot the mean number of reads per cell

**Usage**

```r
Seq_QC_Plot_Reads_per_Cell(
  metrics_dataframe,
  plot_by = "sample_id",
  colors_use = NULL,
  x_lab_rotate = FALSE,
  significance = FALSE,
  ...
)
```
Seq_QC_Plot_Saturation

QC Plots Sequencing metrics

Description

Plot the sequencing saturation percentage per sample

Usage

Seq_QC_Plot_Saturation(
  metrics_dataframe, 
  plot_by = "sample_id", 
  colors_use = NULL, 
  x_lab_rotate = FALSE, 
  significance = FALSE, 
  ... 
)
Arguments

metrics_dataframe
data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.

x_lab_rotate logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

significance logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

... Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value
A ggplot object

Examples

## Not run:
Seq_QC_Plot_Saturation(metrics_dataframe = metrics)

## End(Not run)
Arguments

metrics_dataframe: data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by: Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use: colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.

x_lab_rotate: logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

significance: logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

...: Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value

A ggplot object

Examples

## Not run:
Seq_QC_Plot_Total_Genes(metrics_dataframe = metrics)

## End(Not run)

Description

QC Plots Sequencing metrics (Alignment)

Plot the fraction of reads confidently mapped to transcriptome

Usage

Seq_QC_Plot_Transcriptome(
  metrics_dataframe,
  plot_by = "sample_id",
  colors_use = NULL,
  x_lab_rotate = FALSE,
  significance = FALSE,
  ...
)
Arguments

metrics_dataframe: data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by: Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use: colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.

x_lab_rotate: logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

significance: logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

...: Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value

A ggplot object

Examples

## Not run:
Seq_QC_Plot_Transcriptome(metrics_dataframe = metrics)

## End(Not run)
Arguments

metrics_dataframe
data.frame contain Cell Ranger QC Metrics (see Read_Metrics_10X).

plot_by Grouping factor for the plot. Default is to plot as single group with single point per sample.

colors_use colors to use for plot if plotting by group. Defaults to RColorBrewer Dark2 palette if less than 8 groups and DiscretePalette_scCustomize(palette = "polychrome") if more than 8.

x_lab_rotate logical. Whether to rotate the axes labels on the x-axis. Default is FALSE.

significance logical. Whether to calculate and plot p-value comparisons when plotting by grouping factor. Default is FALSE.

... Other variables to pass to ggpubr::stat_compare_means when doing significance testing.

Value

A ggplot object

Examples

## Not run:
Seq_QC_Plot_UMIs(metrics_dataframe = metrics)

## End(Not run)

Setup_scRNAseq_Project

Setup project directory structure

Description

Create reproducible project directory organization when initiating a new analysis.

Usage

Setup_scRNAseq_Project(
  custom_dir_file = NULL,
  cluster_annotation_path = NULL,
  cluster_annotation_file_name = "cluster_annotation.csv"
)
**Arguments**

- `custom_dir_file`  
  file to file containing desired directory structure. Default is NULL and will provide generic built-in directory structure.

- `cluster_annotation_path`  
  path to place cluster annotation file using `Create_Cluster_Annotation_File`.

- `cluster_annotation_file_name`  
  name to use for annotation file if created (optional).

**Value**

No return value. Creates system folders.

**Examples**

```r
## Not run:
# If using built-in directory structure.
Setup_scRNAseq_Project()
## End(Not run)
```

---

**Single_Color_Palette**  
*Single Color Palettes for Plotting*

**Description**

Selects colors from modified versions of RColorBrewer single colors palettes

**Usage**

```r
Single_Color_Palette(pal_color, num_colors = NULL, seed_use = 123)
```

**Arguments**

- `pal_color`  
  color palette to select (Options are: 'reds', 'blues', 'greens', 'purples', 'oranges', 'grays').

- `num_colors`  
  set number of colors (max = 7).

- `seed_use`  
  set seed for reproducibility (default: 123).

**Value**

A vector of colors

**References**

See RColorBrewer for more info on palettes [https://CRAN.R-project.org/package=RColorBrewer](https://CRAN.R-project.org/package=RColorBrewer)
Examples

```r
pal <- Single_Color_Palette(pal_color = "reds", num_colors = 7)
PalettePlot(palette = pal)
```

Description

Create FeatureScatter using split.by

Usage

```r
Split_FeatureScatter(
  seurat_object,
  feature1 = NULL,
  feature2 = NULL,
  split.by = NULL,
  group.by = NULL,
  colors_use = NULL,
  pt.size = NULL,
  title_size = 15,
  num_columns = NULL,
  raster = NULL,
  raster.dpi = c(512, 512),
  ggplot_default_colors = FALSE,
  color_seed = 123,
  ...
)
```

Arguments

- `seurat_object` Seurat object name.
- `feature1` First feature to plot.
- `feature2` Second feature to plot.
- `split.by` Feature to split plots by (i.e. "orig.ident").
- `group.by` Name of one or more metadata columns to group (color) cells by (for example, orig.ident). Use "ident" to group.by active.ident class.
- `colors_use` color for the points on plot.
- `pt.size` Adjust point size for plotting.
- `title_size` size for plot title labels.
- `num_columns` number of columns in final layout plot.
Stacked_VlnPlot

raster
- Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 cells.

raster.dpi
- Pixel resolution for rasterized plots, passed to geom_scattermore(). Default is c(512, 512).

ggplot_default_colors
- logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.

color_seed
- random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.

... Extra parameters passed to FeatureScatter.

Value
A ggplot object

Examples

library(Seurat)
pbmc_small$sample_id <- sample(c("sample1", "sample2"), size = ncol(pbmc_small), replace = TRUE)

Split_FeatureScatter(seurat_object = pbmc_small, feature1 = "nCount_RNA", feature2 = "nFeature_RNA", split.by = "sample_id")

Stacked_VlnPlot  Stacked Violin Plot

Description
Code for creating stacked violin plot gene expression.

Usage

Stacked_VlnPlot(
  seurat_object,
  features,
  group.by = NULL,
  split.by = NULL,
  idents = NULL,
  x_lab_rotate = FALSE,
  plot_legend = FALSE,
  colors_use = NULL,
  color_seed = 123,
  ggplot_default_colors = FALSE,
  plot_spacing = 0.15,
  spacing_unit = "cm",
  vln_linewidth = NULL,
)
Arguments

- **seurat_object**  
  Seurat object name.
- **features**  
  Features to plot.
- **group.by**  
  Group (color) cells in different ways (for example, orig.ident).
- **split.by**  
  A variable to split the violin plots by.
- **idents**  
  Which classes to include in the plot (default is all).
- **x_lab_rotate**  
  Rotate x-axis labels 45 degrees (Default is FALSE).
- **plot_legend**  
  logical. Adds plot legend containing *idents* to the returned plot.
- **colors_use**  
  specify color palette to used in `VlnPlot`. By default if number of levels plotted is less than or equal to 36 it will use "polychrome" and if greater than 36 will use "varibow" with shuffle = TRUE both from `DiscretePalette_scCustomize`.
- **color_seed**  
  random seed for the "varibow" palette shuffle if `colors_use = NULL` and number of groups plotted is greater than 36. Default = 123.
- **ggplot_default_colors**  
  logical. If `colors_use = NULL`, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.
- **plot_spacing**  
  Numerical value specifying the vertical spacing between each plot in the stack. Default is 0.15 ("cm"). Spacing dependent on unit provided to `spacing_unit`.
- **spacing_unit**  
  Unit to use in specifying vertical spacing between plots. Default is "cm".
- **vln_linewidth**  
  Adjust the linewidth of violin outline. Must be numeric.
- **pt.size**  
  Adjust point size for plotting. Default for StackedVlnPlot is 0 to avoid issues with rendering so many points in vector form. Alternatively, see `raster` parameter.
- **raster**  
  Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 total points plotted (# Cells x # of features).
- **add.noise**  
  logical, determine if adding a small noise for plotting (Default is TRUE).
- **...**  
  Extra parameters passed to `VlnPlot`.

Value

A ggplot object

Author(s)

Ming Tang (Original Code), Sam Marsh (Wrap single function, added/modified functionality)
References


See Also

https://twitter.com/tangming2005

Examples

```r
library(Seurat)
Stacked_VlnPlot(seurat_object = pbmc_small, features = c("CD3E", "CD8", "GZMB", "MS4A1"),
                 x_lab_rotate = TRUE)
```

---

**Store Misc Info Seurat**

*Store misc data in Seurat object*

**Description**

Wrapper function save variety of data types to the `@misc` slot of Seurat object.

**Usage**

```r
Store_Misc_Info_Seurat(
  seurat_object,  # object name.
  data_to_store,  # data to be stored in @misc slot. Can be single piece of data or list. If list of data see list_as_list parameter for control over data storage.
  data_name,      # name to give the entry in @misc slot. Must be of equal length of the number of data items being stored.
  list_as_list = FALSE,  # logical. If data_to_store is a list, this dictates whether to store in @misc slot as list (TRUE) or whether to store each entry in the list separately (FALSE). Default is FALSE.
  overwrite = FALSE)  # Logical. Whether to overwrite existing items with the same name. Default is FALSE, meaning that function will abort if item with data_name is present in misc slot.
```

**Arguments**

- `seurat_object`: object name.
- `data_to_store`: data to be stored in @misc slot. Can be single piece of data or list. If list of data see list_as_list parameter for control over data storage.
- `data_name`: name to give the entry in @misc slot. Must be of equal length of the number of data items being stored.
- `list_as_list`: logical. If data_to_store is a list, this dictates whether to store in @misc slot as list (TRUE) or whether to store each entry in the list separately (FALSE). Default is FALSE.
- `overwrite`: Logical. Whether to overwrite existing items with the same name. Default is FALSE, meaning that function will abort if item with data_name is present in misc slot.
Value

Seurat Object with new entries in the @misc slot.

Examples

```r
library(Seurat)
clu_pal <- c("red", "green", "blue")

pbmc_small <- Store_Misc_Info_Seurat(seurat_object = pbmc_small, data_to_store = clu_pal, data_name = "rd1_colors")
```

Description

Wrapper function around Store_Misc_Info_Seurat to store color palettes.

Usage

```r
Store_Palette_Seurat(
  seurat_object, 
  palette, 
  palette_name, 
  list_as_list = FALSE, 
  overwrite = FALSE
)
```

Arguments

- **seurat_object**: object name.
- **palette**: vector or list of vectors containing color palettes to store. If list of palettes see list_as_list parameter for control over data storage.
- **palette_name**: name to give the palette(s) in @misc slot. Must be of equal length to the number of data items being stored.
- **list_as_list**: logical. If data_to_store is a list, this dictates whether to store in @misc slot as list (TRUE) or whether to store each entry in the list separately (FALSE). Default is FALSE.
- **overwrite**: Logical. Whether to overwrite existing items with the same name. Default is FALSE, meaning that function will abort if item with data_name is present in misc slot.

Value

Seurat Object with new entries in the @misc slot.
Examples

library(Seurat)
clu_pal <- c("red", "green", "blue")

pbmc_small <- Store_Misc_Info_Seurat(seurat_object = pbmc_small, data_to_store = clu_pal, data_name = "rd1_colors")

theme_ggprism_mod

Modified ggprism theme

Description

Modified ggprism theme which restores the legend title.

Usage

theme_ggprism_mod(
  palette = "black_and_white",
  base_size = 14,
  base_family = "sans",
  base_fontface = "bold",
  base_line_size = base_size/20,
  base_rect_size = base_size/20,
  axis_text_angle = 0,
  border = FALSE
)

Arguments

palette string. Palette name, use names(ggprism_data$themes) to show all valid palette names.
base_size numeric. Base font size, given in "pt".
base_family string. Base font family, default is "sans".
base_fontface string. Base font face, default is "bold".
base_line_size numeric. Base linewidth for line elements
base_rect_size numeric. Base linewidth for rect elements
axis_text_angle integer. Angle of axis text in degrees. One of: 0, 45, 90, 270.
border logical. Should a border be drawn around the plot? Clipping will occur unless e.g. coord_cartesian(clip = "off") is used.

Value

Returns a list-like object of class theme.
References

theme is a modified version of theme_prism from ggprism package https://github.com/csdaw/ggprism (License: GPL-3). Param text is from ggprism::theme_prism() documentation theme_prism. Theme adaptation based on ggprism vignette https://csdaw.github.io/ggprism/articles/themes.html#make-your-own-ggprism-theme-1.

Examples

```r
# Generate a plot and customize theme
library(ggplot2)
df <- data.frame(x = rnorm(n = 100, mean = 20, sd = 2), y = rbinom(n = 100, size = 100, prob = 0.2))
p <- ggplot(data = df, mapping = aes(x = x, y = y)) + geom_point(mapping = aes(color = "red"))
p + theme_ggprism_mod()
```

---

**Top_Genes_Factor**

*Extract top loading genes for LIGER factor*

**Description**

Extract vector to the top loading genes for specified LIGER iNMF factor

**Usage**

```r
Top_Genes_Factor(liger_object, liger_factor, num_genes = 10)
```

**Arguments**

- `liger_object`: LIGER object name.
- `liger_factor`: LIGER factor number to pull genes from.
- `num_genes`: number of top loading genes to return as vector.

**Value**

A LIGER Object

**Examples**

```r
## Not run:
top_genes_factor10 <- Top_Genes_Factor(liger_object = object, num_genes = 10)
## End(Not run)
```
UnRotate_X

Unrotate x axis on VlnPlot

Description

Shortcut for thematic modification to unrotate the x axis (e.g., for Seurat VlnPlot is rotated by default).

Usage

UnRotate_X(...)

Arguments

... extra arguments passed to ggplot2::theme().

Value

Returns a list-like object of class theme.

Examples

library(Seurat)
p <- VlnPlot(object = pbmc_small, features = "CD3E")
p + UnRotate_X()

VariableFeaturePlot_scCustom

Custom Labeled Variable Features Plot

Description

Creates variable features plot with N number of features already labeled by default.

Usage

VariableFeaturePlot_scCustom(
  seurat_object,
  num_features = 10,
  label = TRUE,
  pt.size = 1,
  colors_use = c("black", "red"),
  repel = TRUE,
  y_axis_log = FALSE,
  assay = NULL,
)
variable_features_ALL_LIGER

```
selection.method = NULL,
...
```

### Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>seurat_object</td>
<td>Seurat object name.</td>
</tr>
<tr>
<td>num_features</td>
<td>Number of top variable features to highlight by color/label.</td>
</tr>
<tr>
<td>label</td>
<td>logical. Whether to label the top features. Default is TRUE.</td>
</tr>
<tr>
<td>pt.size</td>
<td>Adjust point size for plotting.</td>
</tr>
<tr>
<td>colors_use</td>
<td>colors to use for plotting. Default is &quot;black&quot; and &quot;red&quot;.</td>
</tr>
<tr>
<td>repel</td>
<td>logical (default TRUE). Whether or not to repel the feature labels on plot.</td>
</tr>
<tr>
<td>y_axis_log</td>
<td>logical. Whether to change y axis to log10 scale (Default is FALSE).</td>
</tr>
<tr>
<td>assay</td>
<td>Assay to pull variable features from.</td>
</tr>
<tr>
<td>selection.method</td>
<td>If more than one method use to calculate variable features specify which method to use for plotting. See selection.method parameter in <code>VariableFeaturePlot</code> for list of options.</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>Extra parameters passed to <code>VariableFeaturePlot</code>.</td>
</tr>
</tbody>
</table>

### Value

A ggplot object

### Examples

```
library(Seurat)
VariableFeaturePlot_scCustom(seurat_object = pbmc_small, num_features = 10)
```

---

**Variable_Features_ALL_LIGER**

*Perform variable gene selection over whole dataset*

**Description**

Performs variable gene selection for LIGER object across the entire object instead of by dataset and then taking union.
Usage

Variable_Features_ALL_LIGER(
  liger_object,
  num_genes = NULL,
  var.thresh = 0.3,
  alpha.thresh = 0.99,
  tol = 1e-04,
  do.plot = FALSE,
  pt.size = 0.3,
  chunk = 1000
)

Arguments

  liger_object LIGER object name.
  num_genes Number of genes to find. Optimizes the value of var.thresh to get this number of genes. (Default is NULL).
  var.thresh Variance threshold. Main threshold used to identify variable genes. Genes with expression variance greater than threshold (relative to mean) are selected. (higher threshold -> fewer selected genes).
  alpha.thresh Alpha threshold. Controls upper bound for expected mean gene expression (lower threshold -> higher upper bound). (default 0.99)
  tol Tolerance to use for optimization if num.genes values passed in (default 0.0001).
  do.plot Display log plot of gene variance vs. gene expression. Selected genes are plotted in green. (Default FALSE)
  pt.size Point size for plot.
  chunk size of chunks in hdf5 file. (Default 1000)

Value

A LIGER Object with variable genes in correct slot.

References

Matching function parameter text descriptions are taken from rliger::selectGenes which is called by this function after creating new temporary object/dataset. https://github.com/welch-lab/liger. (Licence: GPL-3).

Examples

```r
## Not run:
lier_obj <- Variable_Features_ALL_LIGER(liger_object = liger_obj, num_genes = 2000)
## End(Not run)
```
Viridis Shortcuts

Description

Quick shortcuts to access viridis palettes

Usage

viridis_plasma_dark_high
viridis_plasma_light_high
viridis_inferno_dark_high
viridis_inferno_light_high
viridis_magma_dark_high
viridis_magma_light_high
viridis_dark_high
viridis_light_high

Format

An object of class character of length 30.
An object of class character of length 30.
An object of class character of length 30.
An object of class character of length 30.
An object of class character of length 30.
An object of class character of length 30.
An object of class character of length 30.
An object of class character of length 30.

Value

A color palette for plotting
Examples

```r
## Not run:
FeaturePlot_scCustom(object = seurat_object, features = "Cx3cr1",
                     colors_use = viridis_plasma_dark_high, na_color = "lightgray")

## End(Not run)
```

VlnPlot_scCustom  VlnPlot with modified default settings

Description

Creates DimPlot with some of the settings modified from their Seurat defaults (colors_use, shuffle, label).

Usage

```r
VlnPlot_scCustom(
  seurat_object,
  features,
  colors_use = NULL,
  pt.size = NULL,
  group.by = NULL,
  split.by = NULL,
  idents = NULL,
  num_columns = NULL,
  raster = NULL,
  add.noise = TRUE,
  ggplot_default_colors = FALSE,
  color_seed = 123,
  ...
)
```

Arguments

- `seurat_object`: Seurat object name.
- `features`: Feature(s) to plot.
- `colors_use`: Color palette to use for plotting. By default if number of levels plotted is less than or equal to 36 it will use "polychrome" and if greater than 36 will use "varibow" with shuffle = TRUE both from DiscretePalette_scCustomize.
- `pt.size`: Adjust point size for plotting.
- `group.by`: Name of one or more metadata columns to group (color) cells by (for example, orig.ident); default is the current active.ident of the object.
- `split.by`: Feature to split plots by (i.e. "orig.ident").
idents Which classes to include in the plot (default is all).
num_columns Number of columns in plot layout. Only valid if split.by != NULL.
raster Convert points to raster format. Default is NULL which will rasterize by default if greater than 100,000 total points plotted (# Cells x # of features).
add.noise logical, determine if adding a small noise for plotting (Default is TRUE).
ggplot_default_colors logical. If colors_use = NULL, Whether or not to return plot using default ggplot2 "hue" palette instead of default "polychrome" or "varibow" palettes.
color_seed random seed for the "varibow" palette shuffle if colors_use = NULL and number of groups plotted is greater than 36. Default = 123.
... Extra parameters passed to VlnPlot.

Value
A ggplot object

References
Many of the param names and descriptions are from Seurat to facilitate ease of use as this is simply a wrapper to alter some of the default parameters https://github.com/satijalab/seurat/blob/master/R/visualization.R (License: GPL-3).

Examples
library(Seurat)
VlnPlot_scCustom(seurat_object = pbmc_small, features = "CD3E")
Index

* datasets
  * ensembl_mito_id, 37
  * ensembl_ribo_id, 38
  * viridis_plasma_dark_high, 136

* data
  * ensembl_mito_id, 37
  * ensembl_ribo_id, 38

* helper_util
  * Case_Check, 12
  * Change_Delim_All, 17
  * Change_Delim_Prefix, 18
  * Change_Delim_Suffix, 19
  * CheckMatrix_scCustom, 19
  * Fetch_Meta, 44
  * Gene_Present, 45
  * Merge_Sparse_Data_All, 62
  * Meta_Numeric, 64
  * Meta_Present, 65
  * Replace_Suffix, 108

* iterative_plotting
  * Iterate_Cluster_Highlight_Plot, 46
  * Iterate_DimPlot_bySample, 48
  * Iterate_FeaturePlot_scCustom, 49
  * Iterate_Meta_Highlight_Plot, 51
  * Iterate_PC_Loading_Plots, 52
  * Iterate_Plot_Density_Custom, 53
  * Iterate_Plot_Density_Joint, 55
  * Iterate_VlnPlot_scCustom, 56

* liger_object_util
  * Add_Cell_Complexity_LIGER, 5
  * Add_Mito_Ribo_LIGER, 7
  * Meta_Present_LIGER, 66
  * Top_Genes_Factor, 132
  * Variable_Features_ALL_LIGER, 134

* liger_plotting
  * DimPlot_LIGER, 31
  * plotFactors_scCustom, 71

* marker_annotation_util
  * Add_Pct_Diff, 10
  * Create_Cluster_Annotation_File, 29
  * Extract_Top_Markers, 39
  * Pull_Cluster_Annotation, 80

* object_qc_plotting
  * QC_Plot_GenevsFeature, 91
  * QC_Plot_UMIvsFeature, 92
  * QC_Plot_UMIvsGene, 94
  * QC_Plots_Combined_Vln, 82
  * QC_Plots_Complexity, 83
  * QC_Plots_Feature, 85
  * QC_Plots_Genes, 86
  * QC_Plots_Mito, 88
  * QC_Plots_UMIs, 89

* object_util
  * Add_Cell_Complexity_Seurat, 6
  * Add_CellBender_Diff, 4
  * Add_Mito_Ribo_Seurat, 8
  * Add_Sample_Meta, 11
  * Extract_Sample_Meta, 38
  * Liger_to_Seurat, 58
  * Merge_Seurat_List, 61
  * Meta_Remove_Seurat, 66
  * Rename_Clusters, 107
  * Store_Misc_Info_Seurat, 129
  * Store_Palette_Seurat, 130

* organization_util
  * Copy_From_GCP, 25
  * Copy_To_GCP, 26
  * Setup_scRNAseq_Project, 124

* other_seurat_plotting
  * Plot_Density_Custom, 74
  * Plot_Density_Joint_Only, 75

* palettes
  * ColorBlind_Pal, 25
  * Dark2_Pal, 29
  * DiscretePalette_scCustomize, 35
  * Hue_Pal, 46
  * JCO_Four, 58
  * NavyAndOrange, 68
PalettePlot, 68
scCustomize_Palette, 108
Single_Color_Palette, 125
viridis_plasma_dark_high, 136

* read_&_write
Create_10X_H5, 27
Create_CellBender_Merged_Seurat, 28
Pull_Directory_List, 81
Read10X_GEO, 96
Read10X_h5_GEO, 97
Read10X_h5_Multi_Directory, 98
Read10X_Multi_Directory, 100
Read_CellBender_h5_Mat, 101
Read_CellBender_h5_Multi_Directory, 102
Read_CellBender_h5_Multi_File, 103
Read_GEO_Delim, 105
Read_Metrics_10X, 106

* seqqc_plotting_alignment
Seq_QC_Plot_Antisense, 110
Seq_QC_Plot_Exonic, 112
Seq_QC_Plot_Genome, 114
Seq_QC_Plot_Intergenic, 115
Seq_QC_Plot_Intronic, 116
Seq_QC_Plot_Transcriptome, 122

* seqqc_plotting_basic
Seq_QC_Plot_Genes, 113
Seq_QC_Plot_Number_Cells, 117
Seq_QC_Plot_Reads_in_Cells, 118
Seq_QC_Plot_Reads_per_Cell, 119
Seq_QC_Plot_Saturation, 120
Seq_QC_Plot_Total_Genes, 121
Seq_QC_Plot_UMIs, 123

* seqqc_plotting_layout
Seq_QC_Plot_Alignment_Combined, 109
Seq_QC_Plot_Basic_Combined, 111

* seurat_plotting
Cell_Highlight_Plot, 16
Cluster_Highlight_Plot, 23
Clustered_DotPlot, 20
DimPlot_All_Samples, 30
DimPlot_scCustom, 33
DotPlot_scCustom, 36
FeaturePlot_DualAssay, 41
FeaturePlot_scCustom, 42
Meta_Highlight_Plot, 63
PC_Plotting, 69
Split_FeatureScatter, 126
Stacked_VlnPlot, 127
VariableFeaturePlot_scCustom, 133
VlnPlot_scCustom, 137

* stats_plotting
CellBender_Diff_Plot, 13
Plot_Cells_per_Sample, 73
Plot_Median_Genes, 76
Plot_Median_Mito, 77
Plot_Median_Other, 78
Plot_Median_UMIs, 79

* stats
CellBender_Feature_Diff, 15
Cluster_Stats_All_Samples, 24
Median_Stats, 60
Percent_Expressing, 70

* themes
Blank_Theme, 12
Move_Legend, 67
theme_ggprism_mod, 131
UnRotate_X, 133

Add_Cell_Complexity_LIGER, 5
Add_Cell_Complexity_Seurat, 6
Add_CellBender_Diff, 4
Add_Mito_Ribo_LIGER, 7
Add_Mito_Ribo_Seurat, 8
Add_Pct_Diff, 10
Add_Sample_Meta, 11

Case_Check, 12
Cell_Highlight_Plot, 16
CellBender_Diff_Plot, 13
CellBender_Feature_Diff, 14, 15
Change_Delim_All, 17
Change_Delim_Prefix, 18
Change_Delim_Suffix, 19
CheckMatrix_scCustom, 19
Cluster_Highlight_Plot, 23
Cluster_Stats_All_Samples, 24
Clustered_DotPlot, 20
ColorBlind_Pal, 25
Copy_from_GCP, 25
Copy_to_GCP, 26
Create_10X_H5, 27
Create_CellBender_Merged_Seurat, 28, 41
INDEX

Create_Cluster_Annotation_File, 29, 125
CreateSeuratObject, 28

Dark2_Pal, 29
DimPlot, 17, 24, 31, 34, 47, 48, 52, 63
DimPlot_All_Samples, 30
DimPlot_LIGER, 31
DimPlot_scCustom, 33
DiscretePalette_scCustomize, 32, 35, 71
DotPlot, 37, 70
DotPlot_scCustom, 36

ensembl_mito_id, 37
ensembl_ribo_id, 38
Extract_Sample_Meta, 38
Extract_Top_Markers, 39

FeaturePlot, 42, 43, 50
FeaturePlot_DualAssay, 41
FeaturePlot_scCustom, 42
FeatureScatter, 92, 93, 95, 127
Fetch_Meta, 44
FindAllMarkers, 10, 39, 40
FindMarkers, 10

Gene_Present, 45
gem_text_repel, 14, 15

Hue_Pal, 46

Iterate_Cluster_Highlight_Plot, 46
Iterate_DimPlot_bySample, 48
Iterate_FeaturePlot_scCustom, 49
Iterate_Meta_Highlight_Plot, 51
Iterate_PC_Loading_Plots, 52
Iterate_Plot_Density_Custom, 53
Iterate_Plot_Density_Joint, 55
Iterate_VlnPlot_scCustom, 56

JCO_Four, 58

LabelPoints, 15
Liger_to_Seurat, 58

Median_Stats, 39, 60
merge, 61
Merge_Seurat_List, 61
Merge_Sparse_Data_All, 62
Meta_Highlight_Plot, 63
Meta_Numeric, 64

Meta_Present, 65
Meta_Present_LIGER, 66
Meta_Remove_Seurat, 66
Move_Legend, 67

NavyAndOrange, 68

PalettePlot, 68
patchwork, 43
PC_Plotting, 69
PCHeatmap, 53, 69
Percent_Expressing, 70
Plot_Cells_per_Sample, 73
plot_density, 54, 56, 74, 75
Plot_Density_Custom, 74
Plot_Density_Joint_Only, 75
Plot_Median_Genes, 76
Plot_Median_Mito, 77
Plot_Median_Other, 78
Plot_Median_UMIs, 79
plotFactors_scCustom, 71
Pull_Cluster_Annotation, 80
Pull_Directory_List, 81

QC_Plot_GenevsFeature, 91
QC_Plot_UMIvsFeature, 92
QC_Plot_UMIvsGene, 94
QC_Plots_Combined_Vln, 82
QC_Plots_Complexity, 83
QC_Plots_Feature, 85
QC_Plots_Genes, 86
QC_Plots_Mito, 88
QC_Plots_UMIs, 89

Read10X, 101
Read10X_GEO, 96
Read10X_h5, 98, 100
Read10X_h5_GEO, 97
Read10X_h5 Multi_Directory, 98
Read10X_Multi_Directory, 100
Read_CellBender_h5_Mat, 101, 103, 104
Read_CellBender_h5 Multi_Directory, 102
Read_CellBender_h5 Multi_File, 103
Read_GEO_Delim, 105
Read_Metrics_10X, 106, 110–112
Rename_Clusters, 107
Rename_Idents, 107
Replace_Suffix, 99, 103, 108
| scCustomize_Palette | 108 |
| Seq_QC_Plot_Alignment_Combined | 109 |
| Seq_QC_Plot_Antisense | 110 |
| Seq_QC_Plot_Basic_Combined | 111 |
| Seq_QC_Plot_Exonic | 112 |
| Seq_QC_Plot_Genes | 113 |
| Seq_QC_Plot_Genome | 114 |
| Seq_QC_Plot_Intergenic | 115 |
| Seq_QC_Plot_Intronic | 116 |
| Seq_QC_Plot_Number_Cells | 117 |
| Seq_QC_Plot_Reads_in_Cells | 118 |
| Seq_QC_Plot_Reads_per_Cell | 119 |
| Seq_QC_Plot_Saturation | 120 |
| Seq_QC_Plot_Total_Genes | 121 |
| Seq_QC_Plot_Transcriptome | 122 |
| Seq_QC_Plot_UMIs | 123 |
| Setup_scRNAseq_Project | 124 |
| Single_Color_Palette | 125 |
| Split_FeatureScatter | 126 |
| Stacked_VlnPlot | 127 |
| Store_Misc_Info_Seurat | 129 |
| Store_Palette_Seurat | 130 |
| theme_ggprism_mod | 131 |
| theme_prism | 132 |
| Top_Genes_Factor | 132 |
| UnRotate_X | 133 |
| Variable_Features_ALL_LIGER | 134 |
| VariableFeaturePlot | 134 |
| VariableFeaturePlot_scCustom | 133 |
| viridis_dark_high | 136 |
| (viridis_plasma_dark_high) | 136 |
| viridis_inferno_dark_high | 136 |
| (viridis_plasma_dark_high) | 136 |
| viridis_inferno_light_high | 136 |
| (viridis_plasma_dark_high) | 136 |
| viridis_light_high | 136 |
| (viridis_plasma_dark_high) | 136 |
| viridis_magma_dark_high | 136 |
| (viridis_plasma_dark_high) | 136 |
| viridis_magma_light_high | 136 |
| (viridis_plasma_dark_high) | 136 |
| viridis_plasma_dark_high | 136 |
| (viridis_plasma_dark_high) | 136 |
| viridis_plasma_light_high | 136 |
| (viridis_plasma_dark_high) | 136 |
| VizDimLoadings | 53, 69 |
| VlnPlot | 57, 83, 84, 86, 87, 89, 90, 128, 138 |
| VlnPlot_scCustom | 137 |