Package 'SlimR'

December 17, 2025

Version 1.0.9

Title Machine Learning-Assisted, Marker-Based Tool for Single-Cell and Spatial Transcriptomics Annotation

Description Annotates single-cell and spatial-transcriptomic (ST) data using marker datasets. Supports unified markers list ('Markers_list') creation from built-in databases (e.g., 'Cellmarker2', 'PanglaoDB', 'scIBD', 'TCellSI', 'PCTIT', 'PCTAM'), Seurat objects, or user-supplied Excel files. SlimR can predict calculation parameters by machine learning algorithms (e.g., 'Random Forest', 'Gradient Boosting', 'Support Vector Machine', 'Ensemble Learning'), and based on Markers_list, calculate gene expression of different cell types and predict annotation information, and calculate corresponding AUC and annotate it, then verify it. At the same time, it can calculate gene expression corresponding to the cell type to generate a reference map for manual annotation (e.g., 'Heat Map', 'Feature Plots', 'Combined Plots'). For more details, see Kabacoff (2020, ISBN:9787115420572).

License MIT + file LICENSE

URL https://github.com/Zhaoqing-wang/SlimR

BugReports https://github.com/Zhaoqing-wang/SlimR/issues

Depends R (>= 3.5)

Imports cowplot, dplyr, ggplot2, patchwork, pheatmap, readxl, scales, Seurat, tidyr, tools, tibble

Suggests crayon, caret, gbm, lattice

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

Date 2025-12-17

NeedsCompilation no

Author Zhaoqing Wang [aut, cre] (ORCID: <https://orcid.org/0000-0001-8348-7245>)

Maintainer Zhaoqing Wang <zhaoqingwang@mail.sdu.edu.cn>

Repository CRAN

Date/Publication 2025-12-17 15:40:02 UTC

Contents

calculate_cluster_variability . . . 2
calculate_expression . . . 3
calculate_expression_skewness . . . 4
calculate_probability . . . 5
Cellmarker2 . . . 6
Cellmarker2_raw . . . 6
Cellmarker2_table . . . 7
Celltype_Annotation . . . 8
Celltype_annotation_Cellmarker2 . . . 9
Celltype_Annotation_Combined . . . 11
Celltype_annotation_Excel . . . 12
Celltype_Annotation_Features . . . 14
Celltype_Annotation_Heatmap . . . 16
Celltype_annotation_PanglaoDB . . . 18
Celltype_annotation_Seurat . . . 19
Celltype_Calculate . . . 21
Celltype_Verification . . . 23
estimate_batch_effect . . . 24
extract_dataset_features . . . 25
generate_training_data . . . 26
Markers_filter_Cellmarker2 . . . 26
Markers_filter_PanglaoDB . . . 28
Markers_list_PCTAM . . . 29
Markers_list_PCTIT . . . 29
Markers_list_scIBD . . . 30
Markers_list_TCellSI . . . 31
PanglaoDB . . . 31
PanglaoDB_raw . . . 32
PanglaoDB_table . . . 33
Parameter_Calculate . . . 33
postprocess_parameters . . . 35
predict_optimal_parameters . . . 36
Read_excel_markers . . . 36
Read_seurat_markers . . . 37
train_parameter_model . . . 38

Index **40**

`calculate_cluster_variability`

Calculate Cluster Variability (Use in package)

Description

Measures the degree of separation between different cell clusters based on expression patterns.

Usage

```
calculate_cluster_variability(data.features, features)
```

Arguments

data.features	Data frame containing expression data and cluster labels
features	Feature names to include in analysis

Value

Numeric value representing cluster separation strength

See Also

Other Section_1_Functions_Use_in_Package: calculate_expression(), calculate_expression_skewness(), calculate_probability(), estimate_batch_effect(), extract_dataset_features(), generate_training_data(), postprocess_parameters(), predict_optimal_parameters(), train_parameter_model()

---

calculate_expression *Counts average expression of gene set (Use in package)*

Description

Counts average expression of gene set (Use in package)

Usage

```
calculate_expression(
  object,
  features,
  assay = NULL,
  cluster_col = NULL,
  colour_low = "white",
  colour_high = "navy"
)
```

Arguments

object	Enter a Seurat object.
features	Enter one or a set of markers.
assay	Enter the assay used by the Seurat object, such as "RNA". Default parameters use "assay = NULL".
cluster_col	Enter the meta.data column in the Seurat object to be annotated, such as "seurat_cluster". Default parameters use "cluster_col = NULL".
colour_low	Color for lowest expression level. (default = "white")
colour_high	Color for highest expression level. (default = "black")

Value

Average expression genes and relatied informations in the input "Seurat" object given "cluster_col" and given "features".

See Also

Other Section_1_Functions_Use_in_Package: calculate_cluster_variability(), calculate_expression_skewness(), calculate_probability(), estimate_batch_effect(), extract_dataset_features(), generate_training_data(), postprocess_parameters(), predict_optimal_parameters(), train_parameter_model()

---

calculate_expression_skewness *Calculate Expression Distribution Skewness (Use in package)*

Description

Computes the average skewness of gene expression distributions across all features.

Usage

```
calculate_expression_skewness(expression_matrix)
```

Arguments

expression_matrix	Matrix of expression values

Value

Mean absolute skewness across all genes

See Also

Other Section_1_Functions_Use_in_Package: calculate_cluster_variability(), calculate_expression(), calculate_probability(), estimate_batch_effect(), extract_dataset_features(), generate_training_data(), postprocess_parameters(), predict_optimal_parameters(), train_parameter_model()

`calculate_probability` *Calculate gene set expression and infer probabilities with control datasets (Use in package)*

Description

Calculate gene set expression and infer probabilities with control datasets (Use in package)

Usage

```
calculate_probability(
  object,
  features,
  assay = NULL,
  cluster_col = NULL,
  min_expression = 0.1,
  specificity_weight = 3
)
```

Arguments

`object`	Enter a Seurat object.
`features`	Enter one or a set of markers.
`assay`	Enter the assay used by the Seurat object, such as "RNA". Default parameters use "assay = NULL".
`cluster_col`	Enter the meta.data column in the Seurat object to be annotated, such as "seurat_cluster". Default parameters use "cluster_col = NULL".
`min_expression`	The min_expression parameter defines a threshold value to determine whether a cell's expression of a feature is considered "expressed" or not. It is used to filter out low-expression cells that may contribute noise to the analysis. Default parameters use "min_expression = 0.1".
`specificity_weight`	The specificity_weight parameter controls how much the expression variability (standard deviation) of a feature within a cluster contributes to its "specificity score." It amplifies or suppresses the impact of variability in the final score calculation.Default parameters use "specificity_weight = 3".

Value

Average expression of genes in the input "Seurat" object given "cluster_col" and given "features".

See Also

Other Section_1_Functions_Use_in_Package: `calculate_cluster_variability()`, `calculate_expression()`, `calculate_expression_skewness()`, `estimate_batch_effect()`, `extract_dataset_features()`, `generate_training_data()`, `postprocess_parameters()`, `predict_optimal_parameters()`, `train_parameter_model()`

Cellmarker2 *Cellmarker2 dataset*

Description

A dataset containing marker genes for different cell types from Cellmarker2

Usage

```
Cellmarker2
```

Format

A data frame with 8 columns:

Details

This dataset is used to filter and create a standardized marker list. The dataset can be filtered based on species, tissue class, tissue type, cancer type, and cell type to generate a list of marker genes for specific cell types.

Source

http://117.50.127.228/CellMarker/

See Also

Other Section_0_Database: Cellmarker2_raw, Cellmarker2_table, Markers_list_PCTAM, Markers_list_PCTIT, Markers_list_TCellSI, Markers_list_scIBD, PanglaoDB, PanglaoDB_raw, PanglaoDB_table

Cellmarker2_raw *Cellmarker2 raw dataset*

Description

A dataset containing marker genes for different cell types from Cellmarker2

Usage

```
Cellmarker2_raw
```

Format

A data frame with 20 columns contined in the Cellmarker2 database:

Details

This dataset is used to filter and create a standardized marker list. The dataset can be filtered based on species, tissue class, tissue type, cancer type, and cell type to generate a list of marker genes for specific cell types.

Source

http://117.50.127.228/CellMarker/

See Also

Other Section_0_Database: Cellmarker2, Cellmarker2_table, Markers_list_PCTAM, Markers_list_PCTIT, Markers_list_TCellSI, Markers_list_scIBD, PanglaoDB, PanglaoDB_raw, PanglaoDB_table

---

Cellmarker2_table *Cellmarker2 table*

Description

A dataset containing marker genes for different cell types from Cellmarker2

Usage

```
Cellmarker2_table
```

Format

A list contain different types like species, tissue_class, tissue_type, cancer_type, cell_type

Details

This list is used to choose filters for creation of standardized marker list.

Source

http://117.50.127.228/CellMarker/

See Also

Other Section_0_Database: Cellmarker2, Cellmarker2_raw, Markers_list_PCTAM, Markers_list_PCTIT, Markers_list_TCellSI, Markers_list_scIBD, PanglaoDB, PanglaoDB_raw, PanglaoDB_table

`Celltype_Annotation` *Annotate Seurat Object with SlimR Cell Type Predictions*

Description

This function assigns SlimR predicted cell types to a Seurat object based on cluster annotations, and stores the results in the meta.data slot.

Usage

```
Celltype_Annotation(
  seurat_obj,
  cluster_col,
  SlimR_anno_result,
  plot_UMAP = TRUE,
  annotation_col = "Cell_type_SlimR"
)
```

Arguments

`seurat_obj`	A Seurat object containing cluster information in meta.data.
`cluster_col`	Character string indicating the column name in meta.data that contains cluster IDs.
`SlimR_anno_result`	List generated by function Celltype_Calculate() which containing a data.frame in $Prediction_results with: 1.cluster_col (Cluster identifiers (should match cluster_col in meta.data)) 2.Predicted_cell_type (Predicted cell types for each cluster).
`plot_UMAP`	logical(1); if TRUE, plot the UMAP with cell type annotations.
`annotation_col`	The location to write in 'meta.data' that contains the predicted cell type. (default = "Cell_type_SlimR")

Value

A Seurat object with updated meta.data containing the predicted cell types.

Note

If plot_UMAP = TRUE, this function will print a UMAP plot as a side effect.

See Also

Other Section_3_Automated_Annotation: `Celltype_Calculate()`, `Celltype_Verification()`, `Parameter_Calculate()`

Examples

```
## Not run:
sce <- Celltype_Annotation(seurat_obj = sce,
    cluster_col = "seurat_clusters",
    SlimR_anno_result = SlimR_anno_result,
    plot_UMAP = TRUE,
    annotation_col = "Cell_type_SlimR"
    )

## End(Not run)
```

Celltype_annotation_Cellmarker2

Uses "marker_list" from Cellmarker2 for cell annotation

Description

Uses "marker_list" from Cellmarker2 for cell annotation

Usage

```
Celltype_annotation_Cellmarker2(
  seurat_obj,
  gene_list,
  species,
  cluster_col = "seurat_clusters",
  assay = "RNA",
  save_path = NULL,
  min_counts = 1,
  colour_low = "white",
  colour_high = "navy",
  colour_low_mertic = "white",
  colour_high_mertic = "navy"
)
```

Arguments

seurat_obj	Enter the Seurat object with annotation columns such as "seurat_cluster" in meta.data to be annotated.
gene_list	Enter the standard "Marker_list" generated by the Cellmarker2 database for the SlimR package, generated by the "Markers_filter_Cellmarker2 ()" function.
species	This parameter selects the species "Human" or "Mouse" for standard gene format correction of markers entered by "Marker_list".
cluster_col	Enter annotation columns such as "seurat_cluster" in meta.data of the Seurat object to be annotated. Default parameters use "cluster_col = 'seurat_clusters'".

assay — Enter the assay used by the Seurat object, such as "RNA". Default parameters use "assay = "RNA"".

save_path — The output path of the cell annotation picture. Example parameters use "save_path = './SlimR/Celltype_annotation_Cellmarker2/'".

min_counts — The minimum number of counts of genes in "Marker_list" entered. This number represents the number of the same gene in the same species and the same location in the Cellmarker2 database used for annotation of this cell type. Default parameters use "min_counts = 1".

colour_low — Color for lowest expression level. (default = "white")

colour_high — Color for highest expression level. (default = "navy")

colour_low_mertic — Color for lowest mertic level. (default = "white")

colour_high_mertic — Color for highest mertic level. (default = "navy")

Value

The cell annotation picture is saved in "save_path".

See Also

Other Section_5_Other_Functions_Provided: Celltype_annotation_Excel(), Celltype_annotation_PanglaoDB(), Celltype_annotation_Seurat()

Examples

```
## Not run:
Celltype_annotation_Cellmarker2(seurat_obj = sce,
    gene_list = Markers_list_Cellmarker2,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
    save_path = file.path(tempdir(),"SlimR_Celltype_annotation_Cellmarker2")
    colour_low = "white",
    colour_high = "navy",
    colour_low_mertic = "white",
    colour_high_mertic = "navy",
    )

## End(Not run)
```

`Celltype_Annotation_Combined` *Uses "marker_list" to generate combined plot for cell annotation*

Description

Uses "marker_list" to generate combined plot for cell annotation

Usage

```
Celltype_Annotation_Combined(
  seurat_obj,
  gene_list,
  species,
  cluster_col = "seurat_clusters",
  assay = "RNA",
  save_path = NULL,
  colour_low = "white",
  colour_high = "navy"
)
```

Arguments

`seurat_obj`	Enter the Seurat object with annotation columns such as "seurat_cluster" in meta.data to be annotated.
`gene_list`	A list of cells and corresponding gene controls, the name of the list is cell type, and the first column of the list corresponds to markers. Lists can be generated using functions such as "Markers_filter_Cellmarker2 ()", "Markers_filter_PanglaoDB ()", "read_excel_markers ()", "read_seurat_markers ()", etc.
`species`	This parameter selects the species "Human" or "Mouse" for standard gene format correction of markers entered by "Marker_list".
`cluster_col`	Enter annotation columns such as "seurat_cluster" in meta.data of the Seurat object to be annotated. Default parameters use "cluster_col = 'seurat_clusters'".
`assay`	Enter the assay used by the Seurat object, such as "RNA". Default parameters use "assay = 'RNA'".
`save_path`	The output path of the cell annotation picture. Example parameters use "save_path = './SlimR/Celltype_annotation_Bar/'".
`colour_low`	Color for lowest expression level. (default = "white")
`colour_high`	Color for highest expression level. (default = "navy")

Value

The cell annotation picture is saved in "save_path".

See Also

Other Section_4_Semi_Automated_Annotation: Celltype_Annotation_Features(), Celltype_Annotation_Heatmap()

Examples

```
## Not run:
Celltype_Annotation_Combined(seurat_obj = sce,
    gene_list = Markers_list,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
    save_path = file.path(tempdir(),"SlimR_Celltype_Annotation_Combined"),
    colour_low = "white",
    colour_high = "navy"
    )

## End(Not run)
```

Celltype_annotation_Excel

Uses "marker_list" from Excel input for cell annotation

Description

Uses "marker_list" from Excel input for cell annotation

Usage

```
Celltype_annotation_Excel(
  seurat_obj,
  gene_list,
  species,
  cluster_col = "seurat_clusters",
  assay = "RNA",
  save_path = NULL,
  metric_names = NULL,
  colour_low = "white",
  colour_high = "navy",
  colour_low_mertic = "white",
  colour_high_mertic = "navy"
)
```

Arguments

seurat_obj	Enter the Seurat object with annotation columns such as "seurat_cluster" in meta.data to be annotated.

gene_list	Enter the standard "Marker_list" generated by the Excel files database for the SlimR package, generated by the "read_excel_markers()" function.
species	This parameter selects the species "Human" or "Mouse" for standard gene format correction of markers entered by "Marker_list".
cluster_col	Enter annotation columns such as "seurat_cluster" in meta.data of the Seurat object to be annotated. Default parameters use "cluster_col = "seurat_clusters"".
assay	Enter the assay used by the Seurat object, such as "RNA". Default parameters use "assay = 'RNA'".
save_path	The output path of the cell annotation picture. Example parameters use "save_path = './SlimR/Celltype_annotation_Excel/'".
metric_names	Change the row name for the input mertics, not recommended unless necessary. (NULL is used as default parameter)
colour_low	Color for lowest expression level. (default = "white")
colour_high	Color for highest expression level. (default = "navy")
colour_low_mertic	Color for lowest mertic level. (default = "white")
colour_high_mertic	Color for highest mertic level. (default = "navy")

Value

The cell annotation picture is saved in "save_path".

See Also

Other Section_5_Other_Functions_Provided: Celltype_annotation_Cellmarker2(), Celltype_annotation_PanglaoDB, Celltype_annotation_Seurat()

Examples

```
## Not run:
Celltype_annotation_Excel(seurat_obj = sce,
    gene_list = Markers_list_Excel,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
    save_path = file.path(tempdir(),"SlimR_Celltype_annotation_Excel")
    colour_low = "white",
    colour_high = "navy",
    colour_low_mertic = "white",
    colour_high_mertic = "navy",
    )

## End(Not run)
```

`Celltype_Annotation_Features` *Annotate cell types using features plot with different marker databases*

Description

This function dynamically selects the appropriate annotation method based on the `gene_list_type` parameter. It supports marker databases from Cellmarker2, PanglaoDB, Seurat (via `FindAllMarkers`), or Excel files.

Usage

```
Celltype_Annotation_Features(
  seurat_obj,
  gene_list,
  gene_list_type = "Default",
  species = NULL,
  cluster_col = "seurat_clusters",
  assay = "RNA",
  save_path = NULL,
  min_counts = 1,
  metric_names = NULL,
  colour_low = "white",
  colour_high = "navy",
  colour_low_mertic = "white",
  colour_high_mertic = "navy",
  ...
)
```

Arguments

`seurat_obj`	A valid Seurat object with cluster annotations in `meta.data`.
`gene_list`	A list of data frames containing marker genes and metrics. Format depends on `gene_list_type`: - **Cellmarker2**: Generated by `Markers_filter_Cellmarker2()`. - **PanglaoDB**: Generated by `Markers_filter_PanglaoDB()`. - **Seurat**: Generated by `read_seurat_markers()`. - **Excel**: Generated by `read_excel_markers()`.
`gene_list_type`	Type of marker database to use. Be one of: `"Cellmarker2"`, `"PanglaoDB"`, `"Seurat"`, or `"Excel"`.
`species`	Species of the dataset: `"Human"` or `"Mouse"` for gene name standardization.
`cluster_col`	Column name in `meta.data` defining clusters (default: `"seurat_clusters"`).
`assay`	Assay layer in the Seurat object (default: `"RNA"`).
`save_path`	Directory to save output PNGs. Must be explicitly specified.
`min_counts`	Minimum number of counts for Cellmarker2 annotations (default: `1`).
`metric_names`	Optional. Change the row name for the input mertics, not recommended unless necessary. (NULL is used as default parameter; used in "Seurat"/"Excel").

colour_low	Color for lowest expression level. (default = "white")
colour_high	Color for highest expression level. (default = "navy")
colour_low_mertic	Color for lowest mertic level. (default = "white")
colour_high_mertic	Color for highest mertic level. (default = "navy")
...	Additional parameters passed to the specific annotation function.

Value

Saves cell type annotation PNGs in save_path. Returns invisibly.

See Also

Other Section_4_Semi_Automated_Annotation: Celltype_Annotation_Combined(), Celltype_Annotation_Heatmap()

Examples

```
## Not run:
# Example for Cellmarker2
Celltype_Annotation_Features(seurat_obj = sce,
    gene_list = Markers_list_Cellmarker2,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
    save_path = file.path(tempdir(),"SlimR_Celltype_annotation_Cellmarker2"),
    colour_low = "white",
    colour_high = "navy",
    colour_low_mertic = "white",
    colour_high_mertic = "navy",
    )

# Example for PanglaoDB
Celltype_Annotation_Features(seurat_obj = sce,
    gene_list = Markers_list_panglaoDB,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
    save_path = file.path(tempdir(),"SlimR_Celltype_annotation_PanglaoDB")
    colour_low = "white",
    colour_high = "navy",
    colour_low_mertic = "white",
    colour_high_mertic = "navy",
    )

# Example for Seurat marker list
Celltype_Annotation_Features(seurat_obj = sce,
    gene_list = Markers_list_Seurat,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
```

```
    save_path = file.path(tempdir(),"SlimR_Celltype_annotation_Seurat")
    colour_low = "white",
    colour_high = "navy",
    colour_low_mertic = "white",
    colour_high_mertic = "navy",
    )

# Example for Excel marker list
Celltype_Annotation_Features(seurat_obj = sce,
    gene_list = Markers_list_Excel,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
    save_path = file.path(tempdir(),"SlimR_Celltype_annotation_Excel")
    colour_low = "white",
    colour_high = "navy",
    colour_low_mertic = "white",
    colour_high_mertic = "navy",
    )

## End(Not run)
```

`Celltype_Annotation_Heatmap` *Uses "marker_list" to generate heatmap for cell annotation*

Description

Uses "marker_list" to generate heatmap for cell annotation

Usage

```
Celltype_Annotation_Heatmap(
  seurat_obj,
  gene_list,
  species,
  cluster_col = "seurat_clusters",
  assay = "RNA",
  min_expression = 0.1,
  specificity_weight = 3,
  colour_low = "navy",
  colour_high = "firebrick3"
)
```

Arguments

`seurat_obj`	Enter the Seurat object with annotation columns such as "seurat_cluster" in meta.data to be annotated.

gene_list	A list of cells and corresponding gene controls, the name of the list is cell type, and the first column of the list corresponds to markers. Lists can be generated using functions such as "Markers_filter_Cellmarker2 ()", "Markers_filter_PanglaoDB ()", "read_excel_markers ()", "read_seurat_markers ()", etc.
species	This parameter selects the species "Human" or "Mouse" for standard gene format correction of markers entered by "Marker_list".
cluster_col	Enter annotation columns such as "seurat_cluster" in meta.data of the Seurat object to be annotated. Default parameters use "cluster_col = 'seurat_clusters'".
assay	Enter the assay used by the Seurat object, such as "RNA". Default parameters use "assay = 'RNA'".
min_expression	The min_expression parameter defines a threshold value to determine whether a cell's expression of a feature is considered "expressed" or not. It is used to filter out low-expression cells that may contribute noise to the analysis. Default parameters use "min_expression = 0.1".
specificity_weight	The specificity_weight parameter controls how much the expression variability (standard deviation) of a feature within a cluster contributes to its "specificity score." It amplifies or suppresses the impact of variability in the final score calculation.Default parameters use "specificity_weight = 3".
colour_low	Color for lowest probability level in Heatmap visualization of probability matrix. (default = "navy")
colour_high	Color for highest probability level Heatmap visualization of probability matrix. (default = "firebrick3")

Value

The heatmap of the comparison between "cluster_col" in the Seurat object and the given gene set "gene_list" needs to be annotated.

See Also

Other Section_4_Semi_Automated_Annotation: `Celltype_Annotation_Combined()`, `Celltype_Annotation_Features(`

Examples

```
## Not run:
Celltype_Annotation_Heatmap(seurat_obj = sce,
    gene_list = Markers_list,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
    min_expression = 0.1,
    specificity_weight = 3,
    colour_low = "navy",
    colour_high = "firebrick3"
    )

## End(Not run)
```

`Celltype_annotation_PanglaoDB` *Uses "marker_list" from PanglaoDB for cell annotation*

Description

Uses "marker_list" from PanglaoDB for cell annotation

Usage

```
Celltype_annotation_PanglaoDB(
  seurat_obj,
  gene_list,
  species,
  cluster_col = "seurat_clusters",
  assay = "RNA",
  save_path = NULL,
  metric_names = NULL,
  colour_low = "white",
  colour_high = "navy",
  colour_low_mertic = "white",
  colour_high_mertic = "navy"
)
```

Arguments

`seurat_obj`	Enter the Seurat object with annotation columns such as "seurat_cluster" in meta.data to be annotated.
`gene_list`	Enter the standard "Marker_list" generated by the PanglaoDB database for the SlimR package, generated by the "Markers_filter_PanglaoDB ()" function.
`species`	This parameter selects the species "Human" or "Mouse" for standard gene format correction of markers entered by "Marker_list".
`cluster_col`	Enter annotation columns such as "seurat_cluster" in meta.data of the Seurat object to be annotated. Default parameters use "cluster_col = 'seurat_clusters'".
`assay`	Enter the assay used by the Seurat object, such as "RNA". Default parameters use "assay = 'RNA'".
`save_path`	The output path of the cell annotation picture. Example parameters use "save_path = './SlimR/Celltype_annotation_PanglaoDB/'".
`metric_names`	Warning: Do not enter information. This parameter is used to check if "Marker_list" conforms to the PanglaoDB database output.
`colour_low`	Color for lowest expression level. (default = "white")
`colour_high`	Color for highest expression level. (default = "navy")
`colour_low_mertic`	Color for lowest mertic level. (default = "white")
`colour_high_mertic`	Color for highest mertic level. (default = "navy")

Value

The cell annotation picture is saved in "save_path".

See Also

Other Section_5_Other_Functions_Provided: Celltype_annotation_Cellmarker2(), Celltype_annotation_Excel(), Celltype_annotation_Seurat()

Examples

```
## Not run:
Celltype_annotation_PanglaoDB(seurat_obj = sce,
    gene_list = Markers_list_panglaoDB,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
    save_path = file.path(tempdir(),"SlimR_Celltype_annotation_PanglaoDB")
    colour_low = "white",
    colour_high = "navy",
    colour_low_mertic = "white",
    colour_high_mertic = "navy",
    )

## End(Not run)
```

Celltype_annotation_Seurat

Uses "marker_list" from Seurat object for cell annotation

Description

Uses "marker_list" from Seurat object for cell annotation

Usage

```
Celltype_annotation_Seurat(
  seurat_obj,
  gene_list,
  species,
  cluster_col = "seurat_clusters",
  assay = "RNA",
  save_path = NULL,
  metric_names = NULL,
  colour_low = "white",
  colour_high = "navy",
  colour_low_mertic = "white",
  colour_high_mertic = "navy"
)
```

Arguments

seurat_obj	Enter the Seurat object with annotation columns such as "seurat_cluster" in meta.data to be annotated.
gene_list	Enter the standard "Marker_list" generated by the Seurat object database for the SlimR package, generated by the "read_seurat_markers()" function.
species	This parameter selects the species "Human" or "Mouse" for standard gene format correction of markers entered by "Marker_list".
cluster_col	Enter annotation columns such as "seurat_cluster" in meta.data of the Seurat object to be annotated. Default parameters use "cluster_col = 'seurat_clusters'".
assay	Enter the assay used by the Seurat object, such as "RNA". Default parameters use "assay = 'RNA'".
save_path	The output path of the cell annotation picture. Example parameters use "save_path = './SlimR/Celltype_annotation_Seurat/'".
metric_names	Change the row name for the input mertics, not recommended unless necessary. (NULL is used as default parameter)
colour_low	Color for lowest expression level. (default = "white")
colour_high	Color for highest expression level. (default = "navy")
colour_low_mertic	Color for lowest mertic level. (default = "white")
colour_high_mertic	Color for highest mertic level. (default = "navy")

Value

The cell annotation picture is saved in "save_path".

See Also

Other Section_5_Other_Functions_Provided: Celltype_annotation_Cellmarker2(), Celltype_annotation_Excel(), Celltype_annotation_PanglaoDB()

Examples

```
## Not run:
Celltype_annotation_Seurat(seurat_obj = sce,
    gene_list = Markers_list_Seurat,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
    save_path = file.path(tempdir(),"SlimR_Celltype_annotation_Seurat")
    colour_low = "white",
    colour_high = "navy",
    colour_low_mertic = "white",
    colour_high_mertic = "navy",
    )

## End(Not run)
```

`Celltype_Calculate` *Uses "marker_list" to calculate probability, prediction results, AUC and generate heatmap for cell annotation*

Description

Uses "marker_list" to calculate probability, prediction results, AUC and generate heatmap for cell annotation

Usage

```
Celltype_Calculate(
  seurat_obj,
  gene_list,
  species,
  cluster_col = "seurat_clusters",
  assay = "RNA",
  min_expression = 0.1,
  specificity_weight = 3,
  threshold = 0.8,
  compute_AUC = TRUE,
  plot_AUC = TRUE,
  AUC_correction = TRUE,
  colour_low = "navy",
  colour_high = "firebrick3"
)
```

Arguments

`seurat_obj`	Enter the Seurat object with annotation columns such as "seurat_cluster" in meta.data to be annotated.
`gene_list`	A list of cells and corresponding gene controls, the name of the list is cell type, and the first column of the list corresponds to markers. Lists can be generated using functions such as "Markers_filter_Cellmarker2 ()", "Markers_filter_PanglaoDB ()", "read_excel_markers ()", "read_seurat_markers ()", etc.
`species`	This parameter selects the species "Human" or "Mouse" for standard gene format correction of markers entered by "Marker_list".
`cluster_col`	Enter annotation columns such as "seurat_cluster" in meta.data of the Seurat object to be annotated. Default parameters use "cluster_col = 'seurat_clusters'".
`assay`	Enter the assay used by the Seurat object, such as "RNA". Default parameters use "assay = 'RNA'".
`min_expression`	The min_expression parameter defines a threshold value to determine whether a cell's expression of a feature is considered "expressed" or not. It is used to filter out low-expression cells that may contribute noise to the analysis. Default parameters use "min_expression = 0.1".

`specificity_weight`
: The specificity_weight parameter controls how much the expression variability (standard deviation) of a feature within a cluster contributes to its "specificity score." It amplifies or suppresses the impact of variability in the final score calculation.Default parameters use "specificity_weight = 3".

`threshold`
: This parameter refers to the normalized similarity between the "alternative cell type" and the "predicted cell type" in the returned results. (the default parameter is 0.8)

`compute_AUC`
: Logical indicating whether to calculate AUC values for predicted cell types. AUC measures how well the marker genes distinguish the cluster from others. When TRUE, adds an AUC column to the prediction results. (default: TRUE)

`plot_AUC`
: The logic indicates whether to draw an AUC curve for the predicted cell type. When TRUE, add an AUC_plot to result. (default: TRUE)

`AUC_correction`
: Logical value controlling AUC-based correction. (default = TRUE) When set to TRUE: 1.Computes AUC values for candidate cell types. (probability > threshold) 2.Selects the cell type with the highest AUC as the final predicted type. 3.Records the selected type's AUC value in the "AUC" column.

`colour_low`
: Color for lowest probability level in Heatmap visualization of probability matrix. (default = "navy")

`colour_high`
: Color for highest probability level Heatmap visualization of probability matrix. (default = "firebrick3")

Value

A list containing:

- Expression_list: List of expression matrices for each cell type
- Proportion_list: List of proportion of expression for each cell type
- Expression_scores_matrix: Matrix of expression scores
- Probability_matrix: Matrix of normalized probabilities
- Prediction_results: Data frame with cluster annotations including:
  - cluster_col: Cluster identifier
  - Predicted_cell_type: Primary predicted cell type
  - AUC: Area Under the Curve value (when compute_AUC = TRUE)
  - Alternative_cell_types: Semi-colon separated alternative cell types
- Heatmap_plot: Heatmap visualization of probability matrix
- AUC_plot: AUC visualization of Predicted cell type

See Also

Other Section_3_Automated_Annotation: `Celltype_Annotation()`, `Celltype_Verification()`, `Parameter_Calculate()`

Examples

```
## Not run:
SlimR_anno_result <- Celltype_Calculate(seurat_obj = sce,
    gene_list = Markers_list,
    species = "Human",
    cluster_col = "seurat_clusters",
    assay = "RNA",
    min_expression = 0.1,
    specificity_weight = 3,
    threshold = 0.8,
    compute_AUC = TRUE,
    plot_AUC = TRUE,
    AUC_correction = TRUE,
    colour_low = "navy",
    colour_high = "firebrick3"
    )

## End(Not run)
```

Celltype_Verification *Perform cell type verification and generate the validation dotplot*

Description

This function performs verification of predicted cell types by selecting high log2FC and high expression proportion genes and generates and generate the validation dotplot.

Usage

```
Celltype_Verification(
  seurat_obj,
  SlimR_anno_result,
  assay = "RNA",
  gene_number = 5,
  colour_low = "white",
  colour_high = "navy",
  annotation_col = "Cell_type_SlimR"
)
```

Arguments

seurat_obj	A Seurat object containing single-cell data.
SlimR_anno_result	A list containing SlimR annotation results with: Expression_list - List of expression matrices for each cell type. Prediction_results - Data frame with cluster annotations.

assay	Enter the assay used by the Seurat object, such as "RNA". Default parameters use "assay = 'RNA'".
gene_number	Integer specifying number of top genes to select per cell type.
colour_low	Color for lowest expression level. (default = "white")
colour_high	Color for highest expression level. (default = "navy")
annotation_col	Character string specifying the column in meta.data to use for grouping.

Value

A ggplot object showing expression of top variable genes.

See Also

Other Section_3_Automated_Annotation: `Celltype_Annotation()`, `Celltype_Calculate()`, `Parameter_Calculate()`

Examples

```
## Not run:
Celltype_Verification(seurat_obj = sce,
    SlimR_anno_result = SlimR_anno_result,
    assay = "RNA",
    gene_number = 5,
    colour_low = "white",
    colour_high = "navy",
    annotation_col = "Cell_type_SlimR"
    )

## End(Not run)
```

---

estimate_batch_effect *Estimate Batch Effect Strength (Use in package)*

Description

Roughly estimates the potential impact of batch effects using available metadata.

Usage

```
estimate_batch_effect(seurat_obj, assay)
```

Arguments

seurat_obj	Seurat object
assay	Assay name

Value

Batch effect score (0 indicates no detectable batch effect)

See Also

Other Section_1_Functions_Use_in_Package: calculate_cluster_variability(), calculate_expression(), calculate_expression_skewness(), calculate_probability(), extract_dataset_features(), generate_training_data(), postprocess_parameters(), predict_optimal_parameters(), train_parameter_model()

---

extract_dataset_features

Extract Dataset Characteristics for Machine Learning (Use in package)

Description

Computes various statistical features from single-cell data that are used as input for the parameter prediction model.

Usage

```
extract_dataset_features(
  seurat_obj,
  features,
  assay = NULL,
  cluster_col = NULL
)
```

Arguments

seurat_obj	Seurat object
features	Features to analyze
assay	Assay name
cluster_col	Cluster column name

Value

List of dataset characteristics including expression statistics, variability measures, and cluster properties

See Also

Other Section_1_Functions_Use_in_Package: calculate_cluster_variability(), calculate_expression(), calculate_expression_skewness(), calculate_probability(), estimate_batch_effect(), generate_training_data(), postprocess_parameters(), predict_optimal_parameters(), train_parameter_model()

`generate_training_data` *Generate Training Data for Machine Learning Model (Use in package)*

Description

Creates synthetic training data based on empirical rules about optimal parameter relationships with dataset characteristics.

Usage

```
generate_training_data(dataset_features, n_samples = 1000)
```

Arguments

`dataset_features`	List of actual dataset characteristics
`n_samples`	Number of synthetic samples to generate

Value

Data frame with synthetic features and optimal parameter targets

See Also

Other Section_1_Functions_Use_in_Package: `calculate_cluster_variability()`, `calculate_expression()`, `calculate_expression_skewness()`, `calculate_probability()`, `estimate_batch_effect()`, `extract_dataset_features()`, `postprocess_parameters()`, `predict_optimal_parameters()`, `train_parameter_model()`

`Markers_filter_Cellmarker2` *Create Marker_list from the Cellmarkers2 database*

Description

Create Marker_list from the Cellmarkers2 database

Usage

```
Markers_filter_Cellmarker2(
  df,
  species = NULL,
  tissue_class = NULL,
  tissue_type = NULL,
  cancer_type = NULL,
  cell_type = NULL
)
```

Arguments

df	Standardized Cellmarkers2 database. It is read as data(Cellmarkers2) in the SlimR library.
species	Species information in Cellmarkers2 database. The default input is "Human" or "Mouse".The input can be retrieved by "Cellmarkers2_table". For more information,please refer to http://117.50.127.228/CellMarker/ on Cellmarkers2's official website.
tissue_class	Tissue_class information in Cellmarkers2 database. The input can be retrieved by "Cellmarkers2_table". For more information, please refer to http://117.50.127.228/CellMarker/ on Cellmarkers2's official website.
tissue_type	Tissue_type information in Cellmarkers2 database. The input can be retrieved by "Cellmarkers2_table". For more information, please refer to http://117.50.127.228/CellMarker/ on Cellmarkers2's official website.
cancer_type	Cancer_type information in Cellmarkers2 database. The input can be retrieved by "Cellmarkers2_table". For more information, please refer to http://117.50.127.228/CellMarker/ on Cellmarkers2's official website.
cell_type	Cell_type information in Cellmarkers2 database. The input can be retrieved by "Cellmarkers2_table". For more information, please refer to http://117.50.127.228/CellMarker/ on Cellmarkers2's official website.

Value

The standardized "Marker_list" in the SlimR package

See Also

Other Section_2_Standardized_Markers_List: `Markers_filter_PanglaoDB()`, `Read_excel_markers()`, `Read_seurat_markers()`

Examples

```
Cellmarker2 <- SlimR::Cellmarker2
Markers_list_Cellmarker2 <- Markers_filter_Cellmarker2(
    Cellmarker2,
    species = "Human",
    tissue_class = "Intestine",
    tissue_type = NULL,
```

```
    cancer_type = NULL,
    cell_type = NULL
    )
```

Markers_filter_PanglaoDB *Create Marker_list from the PanglaoDB database*

Description

Create Marker_list from the PanglaoDB database

Usage

```
Markers_filter_PanglaoDB(df, species_input, organ_input)
```

Arguments

df	Standardized PanglaoDB database. It is read as data(PanglaoDB) in the SlimR library.
species_input	Species information in PanglaoDB database. The default input is "Human" or "Mouse".The input can be retrieved by "PanglaoDB_table". For more information,please refer to https://panglaodb.se/ on PanglaoDB's official website.
organ_input	Organ type information in the PanglaoDB database. The input can be retrieved by "PanglaoDB_table".For more information, please refer to https://panglaodb.se/ on PanglaoDB's official website.

Value

The standardized "Marker_list" in the SlimR package

See Also

Other Section_2_Standardized_Markers_List: `Markers_filter_Cellmarker2()`, `Read_excel_markers()`, `Read_seurat_markers()`

Examples

```
PanglaoDB <- SlimR::PanglaoDB
Markers_list_panglaoDB <- Markers_filter_PanglaoDB(
    PanglaoDB,
    species_input = 'Human',
    organ_input = 'GI tract'
    )
```

`Markers_list_PCTAM` *List of Macrophage subtype markers in the article "Macrophage diversity in cancer revisited in the era of single-cell omics"*

Description

A dataset containing marker genes for different Macrophage subtypes from the article "Macrophage diversity in cancer revisited in the era of single-cell omics"

Usage

```
Markers_list_PCTAM
```

Format

A list with 7 tables.

Details

This list is a table of 7 types of Tumor-associated macrophages (TAMs) markers obtained from the article "Macrophage diversity in cancer revisited in the era of single-cell omics". The data source is "https://doi.org/10.1016/j.it.2022.04.008", and the reference literature is: Ruo-Yu Ma et al. (2022) doi:10.1016/j.it.2022.04.008.

Source

doi:10.1016/j.it.2022.04.008

See Also

Other Section_0_Database: `Cellmarker2`, `Cellmarker2_raw`, `Cellmarker2_table`, `Markers_list_PCTIT`, `Markers_list_TCellSI`, `Markers_list_scIBD`, `PanglaoDB`, `PanglaoDB_raw`, `PanglaoDB_table`

`Markers_list_PCTIT` *List of T cell subtype markers in the article "Pan-cancer single cell landscape of tumor-infiltrating T cells"*

Description

A dataset containing marker genes for different T cell types from the article "Pan-cancer single cell landscape of tumor-infiltrating T cells"

Usage

```
Markers_list_PCTIT
```

Format

A list with 40 tables.

Details

This list is a table of 40 types of pan-cancer tumor-infiltrating T cell (PCTIT) markers obtained from the article "Pan-cancer single cell landscapeof tumor-infiltrating T cells". The data source is "https://doi.org/10.1126/science.abe6474", and the reference literature is: L. Zheng et al. (2021) doi:10.1126/science.abe6474.

Source

doi:10.1126/science.abe6474

See Also

Other Section_0_Database: Cellmarker2, Cellmarker2_raw, Cellmarker2_table, Markers_list_PCTAM, Markers_list_TCellSI, Markers_list_scIBD, PanglaoDB, PanglaoDB_raw, PanglaoDB_table

---

Markers_list_scIBD *List of cell type markers in the article scIBD*

Description

A dataset containing marker genes for different human intestine cell types from scIBD

Usage

```
Markers_list_scIBD
```

Format

A list with one hundred and one tables.

Details

This list is a table of 101 types of human intestine cell types markers obtained from scIBD. The article doi source is "https://doi.org/10.1038/s43588-023-00464-9", and the reference literature is: Nie et al. (2023) doi:10.1038/s43588-023-00464-9. Note: The 'Markers_list_scIBD' was generated using section 2.5.2 and the parameters 'sort_by = "logFC"' and 'gene_filter = 20' were set.

Source

doi:10.1038/s43588023004649

See Also

Other Section_0_Database: Cellmarker2, Cellmarker2_raw, Cellmarker2_table, Markers_list_PCTAM, Markers_list_PCTIT, Markers_list_TCellSI, PanglaoDB, PanglaoDB_raw, PanglaoDB_table

Markers_list_TCellSI *List of T cell subtype markers in the article TCellSI*

Description

A dataset containing marker genes for different T cell subtypes from TCellSI

Usage

```
Markers_list_TCellSI
```

Format

A list with ten tables.

Details

This list is a table of 10 types of T cell markers obtained from TCellSI. The data source is "https://github.com/GuoBioinfoLab/ and the reference literature is: Yang et al. (2024) doi:10.1002/imt2.231.

Source

https://github.com/GuoBioinfoLab/TCellSI/

See Also

Other Section_0_Database: Cellmarker2, Cellmarker2_raw, Cellmarker2_table, Markers_list_PCTAM, Markers_list_PCTIT, Markers_list_scIBD, PanglaoDB, PanglaoDB_raw, PanglaoDB_table

PanglaoDB *PanglaoDB dataset*

Description

A dataset containing marker genes for different cell types from PanglaoDB

Usage

```
PanglaoDB
```

Format

A data frame with 9 columns:

Details

This dataset is used to filter and create a standardized marker list.'

Source

https://panglaodb.se/

See Also

Other Section_0_Database: Cellmarker2, Cellmarker2_raw, Cellmarker2_table, Markers_list_PCTAM, Markers_list_PCTIT, Markers_list_TCellSI, Markers_list_scIBD, PanglaoDB_raw, PanglaoDB_table

---

PanglaoDB_raw *PanglaoDB raw dataset*

Description

A dataset containing marker genes for different cell types from PanglaoDB

Usage

```
PanglaoDB_raw
```

Format

A data frame with 14 columns contined in the PanglaoDB database:

Details

This dataset is used to filter and create a standardized marker list.'

Source

https://panglaodb.se/

See Also

Other Section_0_Database: Cellmarker2, Cellmarker2_raw, Cellmarker2_table, Markers_list_PCTAM, Markers_list_PCTIT, Markers_list_TCellSI, Markers_list_scIBD, PanglaoDB, PanglaoDB_table

PanglaoDB_table *PanglaoDB table*

Description

A dataset containing marker genes for different cell types from PanglaoDB

Usage

```
PanglaoDB_table
```

Format

A list contain different types like species, organ, cell type.

Details

This list is used to choose filters for creation of standardized marker list.

Source

https://panglaodb.se/

See Also

Other Section_0_Database: Cellmarker2, Cellmarker2_raw, Cellmarker2_table, Markers_list_PCTAM, Markers_list_PCTIT, Markers_list_TCellSI, Markers_list_scIBD, PanglaoDB, PanglaoDB_raw

Parameter_Calculate *Adaptive Parameter Tuning for Single-Cell Data Annotation in SlimR*

Description

This function uses machine learning to automatically determine optimal min_expression and specificity_weight parameters for single-cell data analysis based on dataset characteristics.

Usage

```
Parameter_Calculate(
  seurat_obj,
  features,
  assay = NULL,
  cluster_col = NULL,
  method = "ensemble",
  n_models = 3,
  return_model = FALSE,
  verbose = TRUE
)
```

Arguments

seurat_obj	A Seurat object containing single-cell data
features	Character vector of feature names (genes) to analyze
assay	Name of assay to use (default: default assay)
cluster_col	Column name in metadata containing cluster information
method	Machine learning method: "rf" (random forest), "gbm" (gradient boosting), "svm" (support vector machine), or "ensemble" (default)
n_models	Number of models for ensemble learning (default: 3)
return_model	Whether to return trained model (default: FALSE)
verbose	Whether to print progress messages (default: TRUE)

Value

A list containing:

- min_expression: Recommended expression threshold
- specificity_weight: Recommended specificity weight
- performance: Model performance metric (R-squared)
- dataset_features: Extracted dataset characteristics
- model: Trained model (if return_model = TRUE)

See Also

Other Section_3_Automated_Annotation: Celltype_Annotation(), Celltype_Calculate(), Celltype_Verification()

Examples

```
## Not run:
# Basic usage
SlimR_params <- Parameter_Calculate(
  seurat_obj = sce,
  features = c("CD3E", "CD4", "CD8A"),
  assay = "RNA",
  cluster_col = "seurat_clusters",
  method = "ensemble",
  n_models = 3,
  return_model = FALSE,
  verbose = TRUE
  )

# Use with custom method
SlimR_params <- Parameter_Calculate(
  seurat_obj = sce,
  features = unique(Markers_list_Cellmarker2$`B cell`$marker),
  assay = "RNA",
  cluster_col = "seurat_clusters",
  method = "rf",
```

```
  return_model = FALSE,
  verbose = TRUE
  )

## End(Not run)
```

`postprocess_parameters` *Post-process Predicted Parameters (Use in package)*

Description

Applies constraints and dataset-specific adjustments to ensure predicted parameters are within reasonable ranges.

Usage

```
postprocess_parameters(predicted_params, dataset_features)
```

Arguments

`predicted_params`	List of raw predicted parameters
`dataset_features`	Characteristics of current dataset

Value

List of finalized parameters after post-processing

See Also

Other Section_1_Functions_Use_in_Package: `calculate_cluster_variability()`, `calculate_expression()`, `calculate_expression_skewness()`, `calculate_probability()`, `estimate_batch_effect()`, `extract_dataset_features()`, `generate_training_data()`, `predict_optimal_parameters()`, `train_parameter_model()`

predict_optimal_parameters *Predict Optimal Parameters Using Trained Model (Use in package)*

Description

Applies the trained machine learning model to predict optimal parameters for the current dataset.

Usage

```
predict_optimal_parameters(model, dataset_features)
```

Arguments

model	Trained machine learning model (now a list with two models)
dataset_features	Extracted characteristics of current dataset

Value

List containing predicted min_expression and specificity_weight

See Also

Other Section_1_Functions_Use_in_Package: calculate_cluster_variability(), calculate_expression(), calculate_expression_skewness(), calculate_probability(), estimate_batch_effect(), extract_dataset_features(), generate_training_data(), postprocess_parameters(), train_parameter_model()

Read_excel_markers *Create "Marker_list" from Excel files ".xlsx"*

Description

Create "Marker_list" from Excel files ".xlsx"

Usage

```
Read_excel_markers(path, has_colnames = TRUE)
```

Arguments

path	The path information of Marker files stored in ".xlsx" format. The Sheet name in the file is filled with cell type. The first line of each Sheet is the table head, the first column is filled with markers information, and the following column is filled with mertic information.
has_colnames	Logical value indicating whether the first row contains column names. If FALSE, the first column will be named "Markers" and subsequent columns will be named "Col1", "Col2", etc.

Value

The standardized "Marker_list" in the SlimR package.

See Also

Other Section_2_Standardized_Markers_List: `Markers_filter_Cellmarker2()`, `Markers_filter_PanglaoDB()`, `Read_seurat_markers()`

Examples

```
## Not run:
Markers_list_Excel <- Read_excel_markers(
    "D:/Laboratory/Marker_load.xlsx"
    )

## End(Not run)
```

Read_seurat_markers *Create "Marker_list" from Seurat object*

Description

Create "Marker_list" from Seurat object

Usage

```
Read_seurat_markers(
  df,
  sources = c("Seurat", "presto"),
  sort_by = "FSS",
  gene_filter = 20
)
```

Arguments

df	Dataframe generated by "FindAllMarkers" function, recommend to use parameter "group.by = "Cell_type"" and "only.pos = TRUE".
sources	Type of markers sources to use. Be one of: "Seurat" or "presto".
sort_by	Marker sorting parameter, for Seurat sources, select "avg_log2FC" or "p_val_adj" or "FSS" (Feature Significance Score, FSS, product value of log2FC and Expression ratio). Default parameters use "sort_by = 'FSS'".for presto sources, select "logFC" or "padj" or "FSS". Default parameters use "sort_by = 'FSS'".
gene_filter	The number of markers left for each cell type based on the "sort_by" parameter's level of difference. Default parameters use "gene_fliter = 20"

Value

The standardized "Marker_list" in the SlimR package.

See Also

Other Section_2_Standardized_Markers_List: `Markers_filter_Cellmarker2()`, `Markers_filter_PanglaoDB()`, `Read_excel_markers()`

Examples

```
## Not run:
# Example for Seurat sources markers
seurat_markers <- Seurat::FindAllMarkers(
    object = sce,
    group.by = "Cell_type",
    only.pos = TRUE)

Markers_list_Seurat <- Read_seurat_markers(seurat_markers,
    sources = "Seurat",
    sort_by = "avg_log2FC",
    gene_filter = 20
    )

# Example for presto sources markers
seurat_markers <- dplyr::filter(
    presto::wilcoxauc(
      X = sce,
      group_by = "Cell_type",
      seurat_assay = "RNA"
      ),
    padj < 0.05, logFC > 0.5
    )

Markers_list_Seurat <- Read_seurat_markers(seurat_markers,
    sources = "presto",
    sort_by = "logFC",
    gene_filter = 20
    )

## End(Not run)
```

---

train_parameter_model *Train Parameter Prediction Model (Use in package)*

Description

Trains machine learning models to predict optimal parameters based on dataset characteristics.

Usage

```
train_parameter_model(
  training_data,
  method = "ensemble",
  n_models = 3,
  verbose = TRUE
)
```

Arguments

training_data	Data frame with features and target parameters
method	Machine learning method to use
n_models	Number of models for ensemble learning
verbose	Whether to print training progress

Value

List containing trained model and performance metrics

See Also

Other Section_1_Functions_Use_in_Package: calculate_cluster_variability(), calculate_expression(), calculate_expression_skewness(), calculate_probability(), estimate_batch_effect(), extract_dataset_features(), generate_training_data(), postprocess_parameters(), predict_optimal_paramet

Index

∗ **Section_0_Database**
  Cellmarker2, 6
  Cellmarker2_raw, 6
  Cellmarker2_table, 7
  Markers_list_PCTAM, 29
  Markers_list_PCTIT, 29
  Markers_list_scIBD, 30
  Markers_list_TCellSI, 31
  PanglaoDB, 31
  PanglaoDB_raw, 32
  PanglaoDB_table, 33
∗ **Section_1_Functions_Use_in_Package**
  calculate_cluster_variability, 2
  calculate_expression, 3
  calculate_expression_skewness, 4
  calculate_probability, 5
  estimate_batch_effect, 24
  extract_dataset_features, 25
  generate_training_data, 26
  postprocess_parameters, 35
  predict_optimal_parameters, 36
  train_parameter_model, 38
∗ **Section_2_Standardized_Markers_List**
  Markers_filter_Cellmarker2, 26
  Markers_filter_PanglaoDB, 28
  Read_excel_markers, 36
  Read_seurat_markers, 37
∗ **Section_3_Automated_Annotation**
  Celltype_Annotation, 8
  Celltype_Calculate, 21
  Celltype_Verification, 23
  Parameter_Calculate, 33
∗ **Section_4_Semi_Automated_Annotation**
  Celltype_Annotation_Combined, 11
  Celltype_Annotation_Features, 14
  Celltype_Annotation_Heatmap, 16
∗ **Section_5_Other_Functions_Provided**
  Celltype_annotation_Cellmarker2, 9
  Celltype_annotation_Excel, 12
  Celltype_annotation_PanglaoDB, 18
  Celltype_annotation_Seurat, 19
∗ **datasets**
  Cellmarker2, 6
  Cellmarker2_raw, 6
  Cellmarker2_table, 7
  Markers_list_PCTAM, 29
  Markers_list_PCTIT, 29
  Markers_list_scIBD, 30
  Markers_list_TCellSI, 31
  PanglaoDB, 31
  PanglaoDB_raw, 32
  PanglaoDB_table, 33

calculate_cluster_variability, 2, *4, 5*, *25, 26*, *35, 36*, *39*
calculate_expression, *3*, 3, *4, 5*, *25, 26*, *35, 36*, *39*
calculate_expression_skewness, *3, 4*, 4, *5*, *25, 26*, *35, 36*, *39*
calculate_probability, *3, 4*, 5, *25, 26*, *35, 36*, *39*
Cellmarker2, 6, *7*, *29–33*
Cellmarker2_raw, *6*, 6, *7*, *29–33*
Cellmarker2_table, *6, 7*, 7, *29–33*
Celltype_Annotation, 8, *22*, *24*, *34*
Celltype_annotation_Cellmarker2, 9, *13*, *19, 20*
Celltype_Annotation_Combined, 11, *15*, *17*
Celltype_annotation_Excel, *10*, 12, *19, 20*
Celltype_Annotation_Features, *12*, 14, *17*
Celltype_Annotation_Heatmap, *12*, *15*, 16
Celltype_annotation_PanglaoDB, *10*, *13*, 18, *20*
Celltype_annotation_Seurat, *10*, *13*, *19*, 19
Celltype_Calculate, *8*, 21, *24*, *34*
Celltype_Verification, *8*, *22*, 23, *34*

estimate_batch_effect, *3–5*, 24, *25, 26*, *35*,

36, *39*
extract_dataset_features, *3–5*, *25*, 25, *26*, *35*, *36*, *39*

generate_training_data, *3–5*, *25*, 26, *35*, *36*, *39*

Markers_filter_Cellmarker2, 26, *28*, *37*, *38*
Markers_filter_PanglaoDB, *27*, 28, *37*, *38*
Markers_list_PCTAM, *6*, *7*, 29, *30–33*
Markers_list_PCTIT, *6*, *7*, *29*, 29, *30–33*
Markers_list_scIBD, *6*, *7*, *29*, *30*, 30, *31–33*
Markers_list_TCellSI, *6*, *7*, *29*, *30*, 31, *32*, *33*

PanglaoDB, *6*, *7*, *29–31*, 31, *32*, *33*
PanglaoDB_raw, *6*, *7*, *29–32*, 32, *33*
PanglaoDB_table, *6*, *7*, *29–32*, 33
Parameter_Calculate, *8*, *22*, *24*, 33
postprocess_parameters, *3–5*, *25*, *26*, 35, *36*, *39*
predict_optimal_parameters, *3–5*, *25*, *26*, *35*, 36, *39*

Read_excel_markers, *27*, *28*, 36, *38*
Read_seurat_markers, *27*, *28*, *37*, 37

train_parameter_model, *3–5*, *25*, *26*, *35*, *36*, 38