Package 'iterbi'

April 8, 2022

Type Package

Title A iteratively bifurcated clustering strategy for single-cell sequencing data.

Version 0.4.9

Author Zhixin Li

Maintainer zxlee@connect.hku.hk

Description A iteratively bifurcated clustering strategy for single-cell sequencing data.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

R topics documented:

dataframe_to_vector . . . 2
draw_GO_barplot . . . 2
draw_iterbi_cluster_tree . . . 3
draw_marker_chain_dotplot . . . 3
draw_marker_chain_heatmap . . . 4
find_bifurcation_resolution . . . 5
get.iterbi.colors . . . 5
get_initial_bifurcation_of_two_clusters . . . 6
get_marker_GO_chain . . . 6
get_test_data . . . 7
get_unique_marker . . . 7
hello . . . 8
identify_binary_markers . . . 8
iterbi.bifucation.graph . . . 9
iterbi.bifucation.hclust . . . 9
iterbi.bifucation.kmeans . . . 10
iterbi_GO_annotation . . . 10
ora.go.kegg.clusterProfiler . . . 11
prepare_expression_matrix_heatmap . . . 11
remove_duplicated_GO . . . 12
remove_duplicate_marker_chain . . . 12
rename.iterbi . . . 13
run.iterbi . . . 13
subset_seuratObj_bycells . . . 14
write_iterbi_to_seurat . . . 14

Index 15

`dataframe_to_vector` *transform dataframe to vector*

Description

transform dataframe to vector

Usage

```
dataframe_to_vector(df)
```

Arguments

`df`	A dataframe from Seuratobj[["attr"]]

Value

A vector with names

`draw_GO_barplot` *Draw barplot for GO annotation*

Description

Draw barplot for GO annotation

Usage

```
draw_GO_barplot(barplot_df, cluster.chain)
```

Arguments

`barplot_df`	A GO annotation dataframe from clusterProfiler
`cluster.chain`	cluster.chain from iterbi

Value

A ggplot barplot

draw_iterbi_cluster_tree

Draw cluster tree/chain by clustree

Description

Draw cluster tree/chain by clustree

Usage

```
draw_iterbi_cluster_tree(
  seuratObj,
  iterbi.cellMeta,
  node_text_size = 7,
  node_size = 8
)
```

Arguments

seuratObj	A Seurat object
iterbi.cellMeta	iterbi.cellMeta from iterbi
node_text_size	see node_text_size in clustree function
node_size	see node_size in clustree function

Value

clustree object

draw_marker_chain_dotplot

Draw cluster tree/chain by clustree

Description

Draw cluster tree/chain by clustree

Usage

```
draw_marker_chain_dotplot(
  seuratObj,
  iterbi.marker.chain,
  rmDup = T,
  top_n = 3,
  rel_heights = c(0.5, 9, 1.5),
  rel_widths = c(1, 4)
)
```

Arguments

seuratObj	A Seurat object
iterbi.marker.chain	iterbi.marker.chain from iterbi
rmDup	remove duplicated genes or not
top_n	top n genes for dotplot (sort by P-value)
rel_heights	relative heights for plot_grid(), the first and last control the height of top and bottom margin, the middle one controls the height of tree
rel_widths	relative widths for plot_grid(), the first one controls the widths of tree, the second one controls the widths of dotplot

Value

A plot_grid integrated tree and dotplot

---

draw_marker_chain_heatmap

Draw heatmap by ComplexHeatmap

Description

Draw heatmap by ComplexHeatmap

Usage

```
draw_marker_chain_heatmap(
  seuratObj,
  iterbi.cellMeta,
  iterbi.marker.chain,
  compare_anno = "",
  known_markers = c()
)
```

Arguments

seuratObj	A Seurat object
iterbi.cellMeta	iterbi.cellMeta from iterbi
iterbi.marker.chain	iterbi.marker.chain from iterbi
compare_anno	select a annotation in seurat object for comparision (e.g. previous annotation)
known_markers	the genes to include in the heatmap (like known markers)

Value

ComplexHeatmap object

find_bifurcation_resolution *Find the best resolution for bifurcation in graph-based clustering*

Description

Find the best resolution for bifurcation in graph-based clustering

Usage

```
find_bifurcation_resolution(seuratObj, res_sets = 50)
```

Arguments

seuratObj	A Seurat object
res_sets	The number of resolution for searching

Value

The best resolution which can bifurcate all cells

get.iterbi.colors *Colors for iterbi*

Description

Colors for iterbi

Usage

```
get.iterbi.colors()
```

Value

A color vector

get_initial_bifurcation_of_two_clusters

Find initial bifurcation event of any two clusters in iterbi.cellMeta

Description

Find initial bifurcation event of any two clusters in iterbi.cellMeta

Usage

```
get_initial_bifurcation_of_two_clusters(
  iterbi.cellMeta,
  cluster_1,
  cluster_2,
  balance_cells = T
)
```

Arguments

iterbi.cellMeta	iterbi.cellMeta from iterbi
cluster_1	cluster 1
cluster_2	cluster 2
balance_cells	output balanced cell number (T or F)

Value

A list. subset_seuratObj is the subset cells in cluster 1 and cluster 2 diff_marker_1 is the marker of cluster 1, diff_marker_2 is the marker of cluster 2

get_marker_GO_chain

Get GO chain corresponding to a cluster chain

Description

Get GO chain corresponding to a cluster chain

Usage

```
get_marker_GO_chain(iterbi.GO.anno, cluster.chain, top_n = 3)
```

Arguments

iterbi.GO.anno	A GO annotation dataframe from clusterProfiler
cluster.chain	cluster.chain from iterbi
top_n	top n GO terms for barplot (sort by P-value)

Value

A ggplot barplot showed GO chain

get_test_data *Download raw single-cell matrix for testing*

Description

Download raw single-cell matrix for testing

Usage

```
get_test_data()
```

Value

dowloaded single-cell dataset

get_unique_marker *filter uniquely expressed markers by expression percentage*

Description

filter uniquely expressed markers by expression percentage

Usage

```
get_unique_marker(
  seuratObj,
  iterbi.marker.chain,
  assay = "RNA",
  slot = "data",
  min_cluster_pct = 0.3,
  max_bcg_pct = 0.1,
  min_diff = 0.3
)
```

Arguments

seuratObj	A Seurat object
iterbi.marker.chain	iterbi.marker.chain dataframe contains all the markers
assay	Assay used for prediction
slot	slot used for prediction
min_cluster_pct	Minimal expression percentage of target cluster
max_bcg_pct	Maximum expression percentage of the background cells (non-target cells)
min_diff	Minimal difference (expression percentage) between target cluster and background cells

Value

Uniquely expressed iterbi.marker.chain

hello *Hello, World!*

Description

Prints 'Hello, world!'.

Usage

```
hello()
```

Examples

```
hello()
```

identify_binary_markers *Identify binary markers*

Description

Identify binary markers

Usage

```
identify_binary_markers(
  seuratObj,
  p_value = 0.01,
  min_avg_logFC = 0.1,
  min_correlation = 0.3
)
```

Arguments

seuratObj	A Seurat object
p_value	The number of resolution for searching
min_avg_logFC	Minimal threshold for logFC to pick markers
min_correlation	Minimal threshold for correlation to pick markers

Value

A list. b1 is the positive marker of cluster 0, b2 is the positive marker of cluster 1.

`iterbi.bifucation.graph` *Bifurcation based on graph-based clustering*

Description

Bifurcation based on graph-based clustering

Usage

```
iterbi.bifucation.graph(seuratObj, res_sets = 50)
```

Arguments

`seuratObj`	A Seurat object
`res_sets`	The number of resolution for searching

Value

A bifurcated Seurat object (see active.ident).

`iterbi.bifucation.hclust` *Bifurcation based on hierarchical clustering*

Description

Bifurcation based on hierarchical clustering

Usage

```
iterbi.bifucation.hclust(seuratObj, method = "euclidean")
```

Arguments

`seuratObj`	A Seurat object
`method`	The distance measurement method "euclidean or correlation"

Value

A bifurcated Seurat object (see active.ident).

`iterbi.bifucation.kmeans` *Bifurcation based on K-means clustering*

Description

Bifurcation based on K-means clustering

Usage

```
iterbi.bifucation.kmeans(seuratObj, method = "euclidean")
```

Arguments

`seuratObj`	A Seurat object
`method`	The distance measurement method "euclidean or correlation"

Value

A bifurcated Seurat object (see active.ident).

`iterbi_GO_annotation` *Perform GO annotation of iterbi.marker.chain dataframe*

Description

Perform GO annotation of iterbi.marker.chain dataframe

Usage

```
iterbi_GO_annotation(
  iterbi.marker.chain,
  organism = "hs",
  pvalueCutoff = 0.05,
  min_count = 5
)
```

Arguments

`iterbi.marker.chain`	terbi.marker.chain dataframe contains all the markers
`organism`	"hs" for Homo sapiens, or "mm" for Mus musculus
`pvalueCutoff`	cut off P-value for GO annotations
`min_count`	Minimal count of genes for GO annotations

Value

GO annotation dataframe labeled with cluster

ora.go.kegg.clusterProfiler

GO KEGG ORA analysis by clusterProfiler

Description

GO KEGG ORA analysis by clusterProfiler

Usage

```
ora.go.kegg.clusterProfiler(geneList = markerList, organism = "hs")
```

Arguments

geneList	a gene list
organism	"hs" for Homo sapiens, or "mm" for Mus musculus

Value

a list with GO and KEGG annotation result (clusterProfiler format)

prepare_expression_matrix_heatmap

Prepare expression matrix for heatmap visualization, estimate the overall expression of marker modules

Description

Prepare expression matrix for heatmap visualization, estimate the overall expression of marker modules

Usage

```
prepare_expression_matrix_heatmap(
  seuratObj,
  iterbi.marker.chain,
  assay = "RNA",
  slot = "scale.data",
  known_markers = c()
)
```

Arguments

seuratObj	A Seurat object
iterbi.marker.chain	iterbi.marker.chain from iterbi
assay	Assay used for prediction
slot	slot used for prediction
known_markers	the genes to include in the heatmap (like known markers)

Value

A expression matrix contain the average expression of marker modules and known markers

---

`remove_duplicated_GO` *remove duplicated GO terms based on overlaps of genes*

Description

remove duplicated GO terms based on overlaps of genes

Usage

```
remove_duplicated_GO(tmp.GO.df, max.overlap = 0.6)
```

Arguments

`tmp.GO.df`	GO annotation dataframe from clusterProfiler package
`max.overlap`	Maximum overlapped percentage of genes

Value

GO annotation dataframe without duplications

`remove_duplicate_marker_chain`
Remove duplicated markers from last level

Description

Remove duplicated markers from last level

Usage

```
remove_duplicate_marker_chain(iterbi.marker.chain)
```

Arguments

`iterbi.marker.chain`	iterbi.marker.chain from iterbi

Value

iterbi.marker.chain without duplicated genes

rename.iterbi *rename the clusters inside the iterbi result*

Description

rename the clusters inside the iterbi result

Usage

```
rename.iterbi(iterbi.result)
```

Arguments

iterbi.result	A result file from run.iterbi() function

Value

A renamed list. cellMeta contains the final bifurcation for each level marker_chain contains all the significant markers for each cluster bifucation contains the bifurcation details (parent, child1, child2)

run.iterbi *The main function to perform iteratively bifurcation clustering*

Description

The main function to perform iteratively bifurcation clustering

Usage

```
run.iterbi(
  seuratObj,
  method = "graph",
  min.marker.num = 100,
  max.level.num = 20,
  min.cell.count = 50,
  res_sets = 30,
  verbose = T
)
```

Arguments

seuratObj	A Seurat object
method	The method to perform bifurcation clustering "graph (default), hclust or kmeans"
min.marker.num	Minimal number of markers to confirm a bifurcation
max.level.num	Maximum number of level for bifurcation
min.cell.count	Minimal number of cells to perform bifurcation (must bigger than PC number: 50)
res_sets	The number of resolution for searching
verbose	Print detail proccessing messages

Value

A list. cellMeta contains the preliminary bifurcation for each level marker_chain contains all the significant markers for each cluster bifucation contains the bifurcation details (parent, child1, child2)

`subset_seuratObj_bycells`

Subset seurat object by cell names

Description

Subset seurat object by cell names

Usage

```
subset_seuratObj_bycells(seuratObj, tmp.cells)
```

Arguments

`seuratObj`	A Seurat object
`tmp.cells`	cells for subseting

Value

A subset of seurat object

`write_iterbi_to_seurat`

Write iterbi result to Seurat object

Description

Write iterbi result to Seurat object

Usage

```
write_iterbi_to_seurat(seuratObj, iterbi.result)
```

Arguments

`seuratObj`	A Seurat object
`iterbi.result`	iterbi.result

Value

A Seurat object contains iterbi.result. iterbi will be stored in assay data region of Seurat object ("seuratObj@assays$iterbi")

Index

dataframe_to_vector, 2
draw_GO_barplot, 2
draw_iterbi_cluster_tree, 3
draw_marker_chain_dotplot, 3
draw_marker_chain_heatmap, 4

find_bifurcation_resolution, 5

get.iterbi.colors, 5
get_initial_bifurcation_of_two_clusters, 6
get_marker_GO_chain, 6
get_test_data, 7
get_unique_marker, 7

hello, 8

identify_binary_markers, 8
iterbi.bifucation.graph, 9
iterbi.bifucation.hclust, 9
iterbi.bifucation.kmeans, 10
iterbi_GO_annotation, 10

ora.go.kegg.clusterProfiler, 11

prepare_expression_matrix_heatmap, 11

remove_duplicate_marker_chain, 12
remove_duplicated_GO, 12
rename.iterbi, 13
run.iterbi, 13

subset_seuratObj_bycells, 14

write_iterbi_to_seurat, 14