Package 'richCluster'

December 18, 2025

Type Package

Title Fast, Robust Clustering Algorithms for Gene Enrichment Data

Version 1.0.2

Date 2025-12-12

Maintainer Junguk Hur <hurlabshared@gmail.com>

Description Fast 'C++' agglomerative hierarchical clustering
algorithm packaged into easily callable R functions, designed
to help cluster biological terms based on how
similar of genes are expressed in their activation.

License GPL-3

Depends R (>= 3.5.0)

Imports dplyr, fields, heatmaply, igraph, iheatmapr, magrittr,
networkD3, plotly, Rcpp (>= 1.0.14), stats, tidyr, viridis

Suggests devtools, knitr, rmarkdown, roxygen2, testthat

LinkingTo Rcpp

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.3.3

URL https://github.com/hurlab/richCluster

BugReports https://github.com/hurlab/richCluster/issues

NeedsCompilation yes

Author Junguk Hur [aut, cre] (ORCID: <https://orcid.org/0000-0002-0736-2149>),
Sarah Hong [aut],
Jane Kim [aut]

Repository CRAN

Date/Publication 2025-12-18 14:30:02 UTC

Contents

`cluster` *Cluster Terms from Enrichment Results*

Description

This function performs clustering on enrichment results by integrating gene similarity scores and various clustering strategies.

Usage

```
cluster(
  enrichment_results,
  df_names = NULL,
  min_terms = 5,
  min_value = 0.1,
  distance_metric = "kappa",
  distance_cutoff = 0.5,
  linkage_method = "average",
  linkage_cutoff = 0.5
)
```

Arguments

enrichment_results	A list of dataframes, each containing enrichment results. Each dataframe should include at least the columns 'Term', 'GeneID', and 'Padj'.
df_names	Optional, a character vector of names for the enrichment result dataframes. Must match the length of 'enrichment_results'. Default is 'NULL'.
min_terms	Minimum number of terms each final cluster must include
min_value	Minimum 'Pvalue' a term must have in order to be counted in final clustering
distance_metric	A string specifying the distance metric to use (e.g., "kappa").
distance_cutoff	A numeric value for the distance cutoff (0 < cutoff <= 1).
linkage_method	A string specifying the linkage method to use (e.g., "average"). Supported options are "single", "complete", "average", and "ward".
linkage_cutoff	A numeric value between 0 and 1 for the membership cutoff.

Value

A named list containing: - 'distance_matrix': The distance matrix used in clustering. - 'clusters': The final clusters. - 'df_list': The original list of enrichment result dataframes. - 'merged_df': The merged dataframe containing combined results. - 'cluster_options': A list of clustering parameters used in the analysis. - 'df_names' (optional): The names of the input dataframes if provided.

---

cluster_bar *Cluster-level Bar Plot of Enrichment Significance*

Description

Generates a horizontal bar plot showing average enrichment significance for each cluster, across one or more enrichment datasets.

Usage

```
cluster_bar(cluster_result, clusters = NULL, value_type = "Padj", title = NULL)
```

Arguments

cluster_result	A result list returned by cluster.
clusters	Optional numeric vector of cluster IDs to include. Defaults to all clusters.
value_type	The column name to use for enrichment significance ("Padj" or "Pvalue").
title	Optional plot title. If NULL, a default will be generated.

Value

A plotly object representing the bar plot.

Examples

```
# Load example data
cluster_result <- readRDS(system.file("extdata", "cluster_result.rds",
                                      package = "richCluster"))
cbar <- cluster_bar(cluster_result)
cbar
```

`cluster_correlation_hmap` *Create a Correlation Heatmap for a Specific Cluster*

Description

This function generates a correlation heatmap for a specific cluster based on the provided distance matrix.

Usage

```
cluster_correlation_hmap(
  final_clusters,
  distance_matrix,
  cluster_number,
  merged_df
)
```

Arguments

`final_clusters`	A dataframe containing the final cluster data.
`distance_matrix`	A matrix representing the distances between terms.
`cluster_number`	An integer specifying the cluster number to visualize.
`merged_df`	A dataframe with all terms used to map term indices to names.

Value

An interactive heatmaply heatmap.

cluster_dot *Cluster-level Dot Plot of Enrichment Significance*

Description

Creates a dot plot summarizing cluster-level enrichment across datasets. Each point represents a cluster, with its size proportional to the number of terms and its x-position reflecting average significance (e.g., Padj or Pvalue).

Usage

```
cluster_dot(cluster_result, clusters = NULL, value_type = "Padj", title = NULL)
```

Arguments

cluster_result	A result list returned from cluster.
clusters	Optional numeric vector of cluster IDs to include. Defaults to all clusters.
value_type	The name of the value column to visualize (e.g., "Padj" or "Pvalue").
title	Optional title for the plot. If NULL, a default title is generated.

Value

A plotly object representing the dot plot.

Examples

```
# Load example data
cluster_result <- readRDS(system.file("extdata", "cluster_result.rds",
                                      package = "richCluster"))
cdot <- cluster_dot(cluster_result)
cdot
```

cluster_hmap *Create a Heatmap of Clustered Enrichment Results*

Description

Generates an interactive heatmap from the given clustering results, visualizing -log10(Padj) values for each cluster. The function aggregates values per cluster and assigns representative terms as row names.

Usage

```
cluster_hmap(
  cluster_result,
  clusters = NULL,
  value_type = "Padj",
  aggr_type = mean
)
```

Arguments

cluster_result	A list containing a data frame (‘cluster_df‘) with clustering results. The data frame must contain at least the columns ‘Cluster‘, ‘Term‘, and ‘value_type_*‘ values.
clusters	Optional. A numeric or character vector specifying the clusters to include. If NULL (default), all clusters are included.
value_type	A character string specifying the column name prefix for values to display in hmap cells. Defaults to ‘"Padj"‘.
aggr_type	A function used to aggregate values across clusters (e.g., ‘mean‘ or ‘median‘). Defaults to ‘mean‘.

Details

The function processes the given cluster data frame (‘cluster_df‘), aggregating the ‘value_type_*‘ values per cluster using the specified ‘aggr_type‘ function. The -log10 transformation is applied, and infinite values are replaced with 0.

Representative terms are selected by choosing the term with the lowest ‘value_type‘ in each cluster.

The final heatmap is generated using ‘heatmaply::heatmaply()‘, with an interactive ‘plotly‘ visualization.

Value

An interactive heatmap object (‘plotly‘), displaying the -log10(Padj) values across clusters, with representative terms as row labels.

---

cluster_network *Create a Network Graph for a Specific Cluster*

Description

This function generates a network graph for a specific cluster based on the provided distance matrix. The opacity and length of the edges correspond to the given distance_metric (eg, kappa) score similarity between terms, which is based on shared gene content.

Usage

```
cluster_network(final_clusters, distance_matrix, cluster_number, merged_df)
```

Arguments

final_clusters	A dataframe containing the final cluster data.
distance_matrix	A matrix representing the distances between terms.
cluster_number	An integer specifying the cluster number to visualize.
merged_df	A dataframe with all terms used to map term indices to names.

Value

An interactive networkD3 network graph.

---

compare_network_graphs_plotly
Compare Network Graphs using Plotly

Description

This function creates a side-by-side comparison of network graphs for a single cluster using different p-value types.

Usage

```
compare_network_graphs_plotly(cluster_result, cluster_num, pval_names)
```

Arguments

cluster_result	The result from the clustering function.
cluster_num	The cluster number to plot.
pval_names	A list of p-value names to compare.

Value

A plotly object.

david_cluster *Cluster Terms using DAVID's method*

Description

This function performs clustering on enrichment results using an algorithm inspired by DAVID's functional clustering method.

Usage

```
david_cluster(
  enrichment_results,
  df_names = NULL,
  similarity_threshold = 0.5,
  initial_group_membership = 3,
  final_group_membership = 3,
  multiple_linkage_threshold = 0.5
)
```

Arguments

enrichment_results	A list of dataframes, each containing enrichment results. Each dataframe should include at least the columns 'Term', 'GeneID', and 'Padj'.
df_names	Optional, a character vector of names for the enrichment result dataframes. Must match the length of 'enrichment_results'. Default is 'NULL'.
similarity_threshold	A numeric value for the kappa score cutoff (0 < cutoff <= 1).
initial_group_membership	Minimum number of terms to form an initial seed group.
final_group_membership	Minimum number of terms for a final cluster.
multiple_linkage_threshold	A numeric value for the merging threshold.

Value

A named list containing the clustering results.

export_df *Export Cluster Result as Dataframe*

Description

Returns a comprehensive dataframe containing all the different terms in all clusters.

Usage

```
export_df(cluster_result)
```

Arguments

cluster_result	The cluster_result object from cluster()

Value

A data.frame view of the clustering

filter_clusters *Filter Clusters by Number of Terms*

Description

Filters the full list of clusters by keeping only those with greater than or equal to min_terms # of terms.

Usage

```
filter_clusters(all_clusters, min_terms)
```

Arguments

all_clusters	A dataframe containing the merged seeds with column named ‘ClusterIndices‘.
min_terms	An integer specifying the minimum number of terms required in a cluster.

Value

The filtered data frame with clusters filtered to include only those with at least ‘min_terms‘ terms.

format_colnames *Format Column Names for Merging*

Description

This function maps a vector of column names to standardized names for "GeneID", "Pvalue", and "Padj" based on known variations.

Usage

```
format_colnames(colnames)
```

Arguments

colnames	A character vector of column names to be standardized.

Value

A character vector of standardized column names.

full_network *Create a Network Graph for the Entire Distance Matrix*

Description

This function generates a network graph for the entire distance matrix.

Usage

```
full_network(cluster_result)
```

Arguments

cluster_result	Cluster result named list from richCluster::cluster()

Value

An interactive networkD3 network graph.

merge_enrichment_results *Merge List of Enrichment Results*

Description

This function merges multiple enrichment results ('enrichment_results') into a single dataframe by combining unique GeneID elements across each geneset, and averaging Pvalue / Padj values for each term across all enrichment_results.

Usage

```
merge_enrichment_results(enrichment_results)
```

Arguments

enrichment_results	A list of geneset dataframes containing columns c('Term', 'GeneID', 'Pvalue', 'Padj')

Value

A single merged geneset dataframe with all original columns suffixed with the index of the geneset, with new columns 'GeneID', 'Pvalue', 'Padj' containing the merged values.

plot_network_graph *Plot Network Graph for a Cluster*

Description

This function visualizes a single cluster as a network graph.

Usage

```
plot_network_graph(
  cluster_result,
  cluster_num,
  distance_matrix,
  valuetype_list
)
```

Arguments

cluster_result	The result from the clustering function.
cluster_num	The cluster number to plot.
distance_matrix	The distance matrix used for clustering.
valuetype_list	A list of value types (e.g., "Pvalue_1", "Padj_1") to use for node coloring.

Value

A plot object.

---

runRichCluster *Run clustering in C++ backend*

Description

Run clustering in C++ backend

Usage

```
runRichCluster(
  terms,
  geneIDs,
  distanceMetric,
  distanceCutoff,
  linkageMethod,
  linkageCutoff
)
```

Arguments

terms	Character vector of term names
geneIDs	Character vector of geneIDs
distanceMetric	e.g. "kappa"
distanceCutoff	numeric between 0 and 1
linkageMethod	e.g. "average"
linkageCutoff	numeric between 0 and 1

Value

A list containing the clustering results with the following components:

distance_matrix A numeric matrix containing pairwise distances between terms based on gene similarity

all_clusters A data frame with columns 'Cluster' (cluster ID) and 'TermIndices' (comma-separated indices of terms in each cluster)

linkage_tree The hierarchical clustering dendrogram structure from the agglomerative clustering process

---

`term_bar` *Term-level Bar Plot for a Specific Cluster*

Description

Creates a horizontal bar plot showing enrichment values for individual terms in a selected cluster.

Usage

```
term_bar(cluster_result, cluster = 1, value_type = "Padj", title = NULL)
```

Arguments

`cluster_result`	A result list returned by `cluster`.
`cluster`	Cluster ID (numeric) or term name (character) to visualize.
`value_type`	The column name to use for enrichment significance ("Padj" or "Pvalue").
`title`	Optional plot title. If NULL, a default will be generated.

Value

A `plotly` object representing the bar plot.

Examples

```
# Load example data
cluster_result <- readRDS(system.file("extdata", "cluster_result.rds",
                                      package = "richCluster"))
tbar <- term_bar(cluster_result, cluster = 1)
tbar
```

term_dot *Term-level Dot Plot for a Specific Cluster*

Description

Creates a dot plot of individual terms within a specified cluster, showing their significance and number of genes.

Usage

```
term_dot(cluster_result, cluster = 1, value_type = "Padj", title = NULL)
```

Arguments

cluster_result	A result list returned from cluster.
cluster	Cluster ID (numeric) or term name (character) to plot.
value_type	The name of the value column to visualize (e.g., "Padj" or "Pvalue").
title	Optional title for the plot. If NULL, a default title is generated using the representative term.

Value

A plotly object representing the dot plot of terms.

Examples

```
# Load example data
cluster_result <- readRDS(system.file("extdata", "cluster_result.rds",
                                      package = "richCluster"))
tdot <- term_dot(cluster_result, cluster = 1)
tdot
```

term_hmap *Generate a Heatmap of Enrichment Results for Specific Clusters and Terms*

Description

Creates an interactive heatmap displaying -log10(Padj) values for selected clusters and terms. Users can specify clusters numerically or select them by providing term names. The function ensures that the final heatmap includes all terms from the selected clusters as well as any explicitly provided terms.

Usage

```
term_hmap(cluster_result, clusters, terms, value_type, aggr_type, title = NULL)
```

Arguments

cluster_result	A list containing a data frame (‘cluster_df‘) with clustering results. The data frame must include at least the columns ‘Cluster‘, ‘Term‘, and ‘Padj_*‘ values.
clusters	Optional. A numeric vector specifying the cluster numbers to display, or a character vector specifying terms whose clusters should be included. Defaults to ‘NULL‘, which includes all clusters.
terms	Optional. A character vector specifying additional terms to include in the heatmap. Defaults to ‘NULL‘.
value_type	A character string specifying the column name prefix for adjusted p-values. Defaults to ‘"Padj"‘.
aggr_type	A function used to aggregate values across clusters (e.g., ‘mean‘ or ‘median‘). Defaults to ‘mean‘.
title	An optional parameter to title the plot something else.

Details

The function processes the given ‘cluster_df‘, identifying the clusters and terms to be visualized. If ‘clusters‘ is specified as a numeric vector, the function directly filters based on cluster numbers. If ‘clusters‘ is given as a character vector, it identifies the clusters associated with those terms and retrieves all terms from the selected clusters.

The ‘Padj_*‘ values are transformed using ‘-log10()‘, and infinite values are replaced with ‘0‘. The resulting heatmap is generated using ‘heatmaply::heatmaply()‘ with fixed row ordering (no hierarchical clustering).

Value

An interactive heatmap object (‘plotly‘), displaying the -log10(Padj) values across clusters, with representative terms as row labels and color-coded cluster annotations.

Index