Package: scToppR (via r-universe)

May 30, 2026

Title API Wrapper for ToppGene

Version 1.0.0

Description scToppR provides an easy-to-use API wrapper for the ToppGene web platform, used for gene ontology and functional enrichment research. The package also integrates visualization tools, making it a convenient tool directly connecting ToppGene to code-based workflows in R. The tool can also easily save results into different formats.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports dplyr, forcats, ggplot2, stringr, openxlsx, viridis, patchwork, utils, httr2

Depends R (>= 4.5.0)

LazyData false

Suggests airway, BiocStyle, curl, DESeq2, knitr, rmarkdown, S4Vectors, SingleCellExperiment, SummarizedExperiment, testthat (>= 3.0.0)

VignetteBuilder knitr

biocViews Pathways, SingleCell

BugReports `https://github.com/BioinformaticsMUSC/scToppR`

URL `https://github.com/BioinformaticsMUSC/scToppR`

Config/testthat/edition 3

Config/pak/sysreqs libicu-dev libssl-dev

Repository https://bioc-release.r-universe.dev

Date/Publication 2026-04-28 13:07:04 UTC

RemoteUrl https://github.com/bioc/scToppR

RemoteRef RELEASE_3_23

RemoteSha 9569ca01eb5e5c2d0753f9f2638890f6422749d3

Contents

`addToppData` *Add toppData results to SingleCellExperiment or SummarizedExperiment metadata*

Description

A convenience function to store toppData enrichment results in the metadata slot of a SingleCellExperiment or SummarizedExperiment object. Results are stored directly under the specified slot name, with optional analysis parameters stored in a separate `slot_name_params` slot.

Usage

```
addToppData(
  sce,
  toppData_results,
  slot_name = "toppData",
  include_params = TRUE
)
```

Arguments

`sce`	A SingleCellExperiment or SummarizedExperiment object
`toppData_results`	A data.frame of toppData results from toppFun()
`slot_name`	Name for the metadata slot (default: "toppData")
`include_params`	Logical, whether to include analysis parameters and timestamp in a separate `slot_name_params` slot (default: TRUE)

Value

SingleCellExperiment or SummarizedExperiment object with toppData stored in metadata

Examples

```
library(airway)
data("airway")  # example SummarizedExperiment object
data("toppdata.airway")  # example toppData results
se_with_topp <- addToppData(airway, toppdata.airway)

# Access results directly
topp_results <- S4Vectors::metadata(se_with_topp)$toppData

# Access analysis parameters (if include_params = TRUE)
topp_params <- S4Vectors::metadata(se_with_topp)$toppData_params
```

`addToppData` *Add toppData results to SingleCellExperiment or SummarizedExperiment metadata*

Description

A convenience function to store toppData enrichment results in the metadata slot of a SingleCellExperiment or SummarizedExperiment object. Results are stored directly under the specified slot name, with optional analysis parameters stored in a separate `slot_name_params` slot.

Usage

```
addToppData(
  sce,
  toppData_results,
  slot_name = "toppData",
  include_params = TRUE
)
```

Arguments

`sce`	A SingleCellExperiment or SummarizedExperiment object
`toppData_results`	A data.frame of toppData results from toppFun()
`slot_name`	Name for the metadata slot (default: "toppData")
`include_params`	Logical, whether to include analysis parameters and timestamp in a separate `slot_name_params` slot (default: TRUE)

Value

SingleCellExperiment or SummarizedExperiment object with toppData stored in metadata

Examples

```
library(airway)
data("airway")  # example SummarizedExperiment object
data("toppdata.airway")  # example toppData results
se_with_topp <- addToppData(airway, toppdata.airway)

# Access results directly
topp_results <- S4Vectors::metadata(se_with_topp)$toppData

# Access analysis parameters (if include_params = TRUE)
topp_params <- S4Vectors::metadata(se_with_topp)$toppData_params
```

`get_Entrez` *Convert genes into Entrez format*

Description

Convert genes into Entrez format

Usage

```
get_Entrez(genes)
```

Arguments

`genes`	A list of genes

Value

a vector of genes in Entrez format

Examples

```
get_Entrez(genes = c("IFNG", "FOXP3"))
```

`get_ToppCats` *Get a vector of ToppFun categories*

Description

Get a vector of ToppFun categories

Usage

```
get_ToppCats()
```

Value

a vector

Examples

```
get_ToppCats()
```

`ifnb.de` *IFNB DE results*

Description

A dataframe of differentially expressed genes generated using the FindMarkers function for each cluster from the Kang 2018 IFNB dataset Created using the IFNB dataset from the SeuratData package

Usage

```
data("ifnb.de")
```

Format

A dataframe with 92,860 rows and 7 columns

p_val P values

avg_log2FC avg log 2 fc values

pct.1 percentage of cells expressing gene in group 1

pct.2 percentage of cells expressing gene in group 2

p_val_adj adjusted p-value (FDR)

cluster cell group name

gene gene name

Source

`https://www.nature.com/articles/nbt.4042`

`ifnb.markers.df` *IFNB Marker DF*

Description

A dataframe of 100 top markers for each class in 'seurat_annotations' column using presto::wilcoxauc() and presto::top_markers() Created using the IFNB dataset from the SeuratData package

Usage

```
data("ifnb.markers.df")
```

Format

A dataframe with 100 rows and 14 columns

rank rank of marker

B cell group name

B Activated cell group name

CD14 Mono cell group name

CD16 Mono cell group name

CD4 Memory T cell group name

CD4 Naive T cell group name

CD8 T cell group name

DC cell group name

Eryth cell group name

Mk cell group name

CNK cell group name

pDC cell group name

T activated cell group name

Source

`https://www.nature.com/articles/nbt.4042`

Kang HM, Subramaniam M, Targ S, et al. Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. Nat Biotechnol. 2018;36(1):89-94. doi:10.1038/nbt.4042

`ifnb.markers.list.CD8T` *IFNB Marker DF*

Description

A list of the 100 top markers for CD8 T cells in ifnb dataset using presto::wilcoxauc() and presto::top_markers() Created using the IFNB dataset from the SeuratData package

Usage

```
data("ifnb.markers.list.CD8T")
```

Format

A character vector with 100 genes

ifnb.markers.list.CD8T rank of marker

Source

https://www.nature.com/articles/nbt.4042

Kang HM, Subramaniam M, Targ S, et al. Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. Nat Biotechnol. 2018;36(1):89-94. doi:10.1038/nbt.4042

`pbmc.markers` *PBMC markers*

Description

A dataframe of marker genes generated using the FindMarkers function for each cluster from the PBMC 3k dataset

Usage

```
data("pbmc.markers")
```

Format

A dataframe with 11,629 rows and 7 columns

p_val P values

avg_log2FC avg log 2 fc values

pct.1 percentage of cells expressing gene in group 1

pct.2 percentage of cells expressing gene in group 2

p_val_adj adjusted p-value (FDR)

cluster cell group name

gene gene name

Source

10X Genomics PBMC 3k dataset. Available from https://www.10xgenomics.com/resources/datasets/. Analysis following Seurat PBMC tutorial: https://satijalab.org/seurat/articles/pbmc3k_tutorial.html

---

`toppBalloon` *Create a balloon plot from toppdata results*

Description

This function creates balloon plots from ToppGene enrichment results. It accepts either a data.frame with toppData results, or a SummarizedExperiment/SingleCellExperiment object with toppData stored in the metadata.

Usage

```
toppBalloon(
  toppData,
  categories = NULL,
  balloons = 3,
  x_axis_text_size = 6,
  cluster_col = "Cluster",
  filename = "toppBalloon",
  save = FALSE,
  save_dir = tempdir(),
  height = 6,
  width = 8,
  slot_name = "toppData",
  ...
)
```

Arguments

`toppData`	A toppData results dataframe, SummarizedExperiment, or SingleCellExperiment object
`categories`	The topp categories to plot
`balloons`	Number of balloons per group to plot
`x_axis_text_size`	Size of the text on the x axis
`cluster_col`	The column name for clusters (default: "Cluster")
`filename`	Filename of the saved balloon plot
`save`	Save the balloon plot if TRUE
`save_dir`	Directory to save the balloon plot
`height`	Height of the saved balloon plot

`width`	Width of the saved balloon plot
`slot_name`	For SE/SCE objects, the metadata slot name containing toppData (default: "toppData")
`...`	Additional parameters for future use

Value

ggplot object or list of ggplot objects

Examples

```
data("toppdata.pbmc")

# With data.frame
toppBalloon(toppdata.pbmc, balloons = 3, save = FALSE)

# With SummarizedExperiment (if toppData stored in metadata)
# toppBalloon(se_object, categories = "GeneOntologyMolecularFunction")
```

`toppdata.airway` *toppData example using the airway dataset results*

Description

A dataframe of of sample toppData results created from the ifnb.de dataset using the toppFun() function

Usage

```
data("toppdata.airway")
```

Format

A dataframe with 902 rows and 14 columns

Category ToppGene category

ID ToppGene Term ID

Name ToppGene Term Name

PValue P value

QValueFDRBH adjusted p-value (FDR)

QValueFDRBY adjusted p-value (BY)

QValueBonferroni adjusted p-value (Bonferroni)

TotalGenes Total genes in background

GenesInTerm Genes in ToppGene Term

GenesInQuery Genes in submitted query

GenesInTermQuery Intersection of genes in Term and in Query

Source ToppGene result source

URL ToppGene associated URL

Cluster cell group name

Source

`https://toppgene.cchmc.org`

Generated using ToppGene API (`https://toppgene.cchmc.org/`). Chen J, Bardes EE, Aronow BJ, Jegga AG. ToppGene Suite for gene list enrichment analysis and candidate gene prioritization. Nucleic Acids Res. 2009;37(Web Server issue):W305-11. doi: 10.1093/nar/gkp427.

Himes, E. B, Jiang, X., Wagner, P., Hu, R., Wang, Q., Klanderman, B., Whitaker, M. R, Duan, Q., Lasky-Su, J., Nikolos, C., Jester, W., Johnson, M., Panettieri, A. R, Tantisira, G. K, Weiss, T. S, Lu, Q. (2014). "RNA-Seq Transcriptome Profiling Identifies CRISPLD2 as a Glucocorticoid Responsive Gene that Modulates Cytokine Function in Airway Smooth Muscle Cells." PLoS ONE, 9(6), e99625. http://www.ncbi.nlm.nih.gov/pubmed/24926665.

`https://www.bioconductor.org/packages/release/data/experiment/html/airway.html`

`toppdata.ifnb` *toppData example for ifnb.de*

Description

A dataframe of of sample toppData results created from the ifnb.de dataset using the toppFun() function

Usage

```
data("toppdata.ifnb")
```

Format

A dataframe with 12,227 rows and 14 columns

Category ToppGene category

ID ToppGene Term ID

Name ToppGene Term Name

PValue P value

QValueFDRBH adjusted p-value (FDR)

QValueFDRBY adjusted p-value (BY)

QValueBonferroni adjusted p-value (Bonferroni)

TotalGenes Total genes in background

GenesInTerm Genes in ToppGene Term

GenesInQuery Genes in submitted query

GenesInTermQuery Intersection of genes in Term and in Query

Source ToppGene result source

URL ToppGene associated URL

Cluster cell group name

Source

Generated using ToppGene API (`https://toppgene.cchmc.org/`). Chen J, Bardes EE, Aronow BJ, Jegga AG. ToppGene Suite for gene list enrichment analysis and candidate gene prioritization. Nucleic Acids Res. 2009;37(Web Server issue):W305-11. doi: 10.1093/nar/gkp427.

`https://toppgene.cchmc.org`

Kang HM, Subramaniam M, Targ S, et al. Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. Nat Biotechnol. 2018;36(1):89-94. doi:10.1038/nbt.4042

`toppdata.pbmc` *toppData example*

Description

A dataframe of of sample toppData results created from the pbmc.markers dataset using the toppFun() function

Usage

```
data("toppdata.pbmc")
```

Format

A dataframe with 8,550 rows and 14 columns

Category ToppGene category

ID ToppGene Term ID

Name ToppGene Term Name

PValue P value

QValueFDRBH adjusted p-value (FDR)

QValueFDRBY adjusted p-value (BY)

QValueBonferroni adjusted p-value (Bonferroni)

TotalGenes Total genes in background

GenesInTerm Genes in ToppGene Term

GenesInQuery Genes in submitted query

GenesInTermQuery Intersection of genes in Term and in Query

Source ToppGene result source

URL ToppGene associated URL

Cluster cell group name

Source

https://toppgene.cchmc.org

Generated using ToppGene API (https://toppgene.cchmc.org/). Chen J, Bardes EE, Aronow BJ, Jegga AG. ToppGene Suite for gene list enrichment analysis and candidate gene prioritization. Nucleic Acids Res. 2009;37(Web Server issue):W305-11. doi: 10.1093/nar/gkp427.

10X Genomics PBMC 3k dataset. Available from https://www.10xgenomics.com/resources/datasets/. Analysis following Seurat PBMC tutorial: https://satijalab.org/seurat/articles/pbmc3k_tutorial.html

---

`toppFun` *Get results from ToppFun*

Description

The toppFun() function takes a data.frame or other tabular data structure and selects genes to use in querying ToppGene.

Usage

```
toppFun(
  input_data,
  type = "degs",
  topp_categories = NULL,
  cluster_col = "cluster",
  gene_col = "gene",
  p_val_col = "adj_p_val_col",
  logFC_col = "avg_logFC",
  direction_mode = "all",
  num_genes = 1000,
  pval_cutoff = 0.5,
  fc_cutoff = 0,
  fc_filter = "ALL",
  clusters = NULL,
  correction = "FDR",
  key_type = "SYMBOL",
  min_genes = 2,
  max_genes = 1500,
  max_results = 50,
  verbose = TRUE
)
```

Arguments

`input_data`	A vector of markers or dataframe with columns as cluster labels

`type`	One of c("degs", "marker_list", or "marker_df). If "degs" is selected, the input_data is assumed to be a data.frame with logfoldchange, pvalue, and gene name columns. If "marker_list" is selected, input_data is assumed to be a list of genes with no other stats, and any thresholds pertaining to "degs" will be ignored. If "marker_df" is selected, the input_data is assumed to be a data.frame with columns as clusters/celltypes, and entries are lists of markers.
`topp_categories`	A string or vector with specific toppfun categories for the query
`cluster_col`	Column name for the groups of cells (e.g. cluster or celltype)
`gene_col`	Column name for genes (e.g. gene or feature)
`p_val_col`	Column name for the p-value or adjusted p-value (preferred)
`logFC_col`	Column name for the avg log FC column
`direction_mode`	One of c("all", "split"). Whether to use all genes in the pathway analysis, or to split by up and down regulated genes
`num_genes`	Number of genes per group to use for toppGene query
`pval_cutoff`	(adjusted) P-value cutoff for filtering differentially expressed genes
`fc_cutoff`	Avg log fold change cutoff for filtering differentially expressed genes
`fc_filter`	Include "ALL" genes, or only "UPREG" or "DOWNREG" for each cluster
`clusters`	Which clusters to include in toppGene query
`correction`	P-value correction method ("FDR" is "BH")
`key_type`	Gene name format
`min_genes`	Minimum number of genes to match in a query
`max_genes`	Maximum number of genes to match in a query
`max_results`	Maximum number of results per cluster
`verbose`	Verbosity setting, TRUE or FALSE

Details

The use of data from ToppGene is governed by their Terms of Use: https://toppgene.cchmc.org/navigation/termsofuse.jsp

Value

data.frame

Examples

```
data("ifnb.de")
toppData <- toppFun(ifnb.de,
    topp_categories = NULL,
    cluster_col = "celltype",
    gene_col = "gene",
    p_val_col = "p_val_adj",
    logFC_col = "avg_log2FC"
)
```

`toppPlot` *Create a dotplot from toppdata results*

Description

This function creates dotplots from ToppGene enrichment results. It accepts either a data.frame with toppData results, or a SummarizedExperiment/SingleCellExperiment object with toppData stored in the metadata.

Usage

```
toppPlot(
  toppData,
  category = NULL,
  clusters = NULL,
  cluster_col = "Cluster",
  p_val_adj = "QValueFDRBH",
  p_val_display = "FDR_BH",
  num_terms = 10,
  save = FALSE,
  save_dir = tempdir(),
  width = 8,
  height = 6,
  file_prefix = "toppPlot",
  combine = FALSE,
  ncols = 2,
  y_axis_text_size = 10,
  slot_name = "toppData",
  ...
)
```

Arguments

`toppData`	A toppData results dataframe, SummarizedExperiment, or SingleCellExperiment object
`category`	The topp categories to plot
`clusters`	The cluster(s) to plot
`cluster_col`	The column name for clusters (default: "Cluster")
`p_val_adj`	The P-value correction method: "BH", "Bonferroni", "BY", or "none"
`p_val_display`	If "log", display the p-value in terms of -log10(p_value)
`num_terms`	The number of terms from the toppData results to be plotted, per cluster
`save`	Whether to save the file automatically
`save_dir`	Directory to save file
`width`	width of the saved file (inches)

`height`	height of the saved file (inches)
`file_prefix`	file prefix if saving the plot - the cluster name is also added automatically
`combine`	If TRUE and multiple clusters selected, return a patchwork object of all plots; if FALSE return list of plots
`ncols`	If patchwork element returned, number of columns for subplots
`y_axis_text_size`	Size of the Y axis text - for certain categories, it's helpful to decrease this
`slot_name`	For SE/SCE objects, the metadata slot name containing toppData (default: "toppData")
`...`	Additional parameters for future use

Value

ggplot object or list of ggplot objects

Examples

```
data("toppdata.pbmc")

# With data.frame
toppPlot(toppdata.pbmc,
    category = "GeneOntologyMolecularFunction",
    clusters = 0,
    save = FALSE
)

# With SummarizedExperiment (if toppData stored in metadata)
# toppPlot(se_object, category = "GeneOntologyMolecularFunction")
```

`toppSave` *Save toppData results (optionally) split by celltype/cluster*

Description

Save toppData results (optionally) split by celltype/cluster

Usage

```
toppSave(
  toppData,
  filename = "toppData_results",
  save_dir = NULL,
  split = TRUE,
  format = "xlsx",
  cluster_col = "Cluster",
  verbose = TRUE
)
```

Arguments

`toppData`	Results from toppFun as a dataframe
`filename`	filename prefix for each split file
`save_dir`	the directory to save files
`split`	Boolean, whether to split the dataframe by celltype/cluster
`format`	Saved file format, one of c("xlsx", "csv", "tsv")
`cluster_col`	Column name for the groups of cells (e.g. cluster or celltype), usually "Cluster"
`verbose`	Verbosity setting, TRUE or FALSE

Value

A saved file

Examples

```
data("toppdata.ifnb")
toppSave(toppdata.ifnb,
    filename = "toppFun_results",
    save_dir = tempdir(),
    split = TRUE,
    format = "xlsx")
```

Index