Package 'toppgene'

March 16, 2026

Title Gene List Enrichment Analysis using the ToppGene Suite

Version 0.99.1

Description The ToppGene Suite is a one-stop portal for gene list enrichment analysis and candidate gene prioritization based on functional annotations and protein interactions network. Although the ToppCluster web application provides convenient graphical access to the ToppGene Suite, the OpenAPI 3.0 compliant interface of ToppGene is better suited for automation and reproducibility. This package includes Bioconductor class interfaces and biological examples.

BiocType Software

biocViews Clustering, GeneExpression, GeneSetEnrichment, Genetics, MotifDiscovery, Network, NetworkEnrichment, Pathways, Pharmacogenetics, Proteomics, Software, ThirdPartyClient

URL https://github.com/ImmunoSystems-lab/toppgene

BugReports https://github.com/ImmunoSystems-lab/toppgene/issues

Depends R (>= 4.6.0)

Encoding UTF-8

License GPL (>= 3)

LazyData false

Imports BiocFileCache, httr2, IRanges, jsonlite, methods, purrr, readr, S4Vectors, xml2, yaml

Suggests BiocStyle, DFplyr (>= 1.5.0), knitr, rmarkdown, testthat (>= 3.0.0)

RoxygenNote 7.3.3

VignetteBuilder knitr

Config/testthat/edition 3

Config/testthat/parallel true

git_url https://git.bioconductor.org/packages/toppgene

git_branch devel

git_last_commit fd4063d

git_last_commit_date 2026-02-16

Repository Bioconductor 3.23

Date/Publication 2026-03-15

Author Pariksheet Nanda [aut, cre] (ORCID:
<https://orcid.org/0000-0001-9726-4552>),
Jason Shoemaker [fnd] (ORCID: <https://orcid.org/0000-0003-3315-7103>)

Maintainer Pariksheet Nanda <pan79@pitt.edu>

Contents

`CategoriesDataFrame` *CategoriesDataFrame objects*

Description

Specialized [DataFrame] class with the following additional constraints to represent ToppGene parameters to run a ToppGene [enrich()] query:

- Fixed number of rows with fixed order for each category.
- Columns can only be set to the allowed numerical and set values that are described when showing the object.

The DataFrame semantics allow quickly setting multiple parameters at a time. Unlike typical DataFrame behavior, [show()] displays all 19 rows instead of using ellipses.

Usage

```
CategoriesDataFrame(...)

## S4 method for signature 'CategoriesDataFrame'
show(object)

default(x)

## S4 method for signature 'CategoriesDataFrame'
default(x)

## S4 replacement method for signature 'CategoriesDataFrame'
```

```
x[i, j, ...] <- value

## S4 replacement method for signature 'CategoriesDataFrame,ANY,ANY,ANY'
x[[i, j, ...]] <- value

## S4 replacement method for signature 'CategoriesDataFrame'
rownames(x) <- value

## S4 method for signature 'CategoriesDataFrame'
subset(x, ...)
```

Arguments

...	Arguments passed on to inherited methods.
object	'CategoriesDataFrame' object used in [validObject()] and [show()] function calls.
x	'CategoriesDataFrame' object.
i	With 'j' (x[i, j]]), the row slice(s) / row name(s) , otherwise (x[[i]]) the column slice / name.
j	The column slice / name (x[i, j]).
value	Atomic or vector assigned to 'x'.

Value

'CategoriesDataFrame' with 19 rows (categories: CoExpression, ..., ToppCell) and 5 columns (PValue, MinGenes, MaxGenes, MaxResults, Correction).

See Also

[DataFrame::DataFrame()]

Examples

```
library(DFplyr)
cats <- CategoriesDataFrame()
cats <-
    cats |>
    mutate(
        PValue = 0.001,
        MaxResults = case_when(
            grepl("Onto", rownames(cats)) ~ 25L,
            .default = MaxResults))
cats
```

enrich *Return functional enrichment of gene Entrez IDs.*

Description

The ToppGene API returns many [CATEGORIES] of gene list erichment.

Usage

```
enrich(entrez_ids, categories = CategoriesDataFrame(), max_tries = 3L)
```

Arguments

entrez_ids	Integer vector of genes.
categories	If no categories are provided, return all categories.
max_tries	Number of attempts passed on to httr2::req_retry.

Value

DataFrame with 15 columns containing the enrichment Category, ID, and associated data.

Examples

```
# Sample functional enrichment calls of the ToppGene API specification:
enrich(2L)
enrich(as.integer(c(1482, 4205, 2626, 9421, 9464, 6910, 6722)))
```

lookup *Return integer Entrez IDs from gene symbols, ensembl references, etc.*

Description

The ToppGene API returns many lookup differs from Bioconductor's identifier lookup, therefore we have to use the web API instead of the typical Bioconductor functions of [GSEABase::mapIdentfiers()], [AnnotationDbi::mapIds()], etc.

Usage

```
lookup(symbols, max_tries = 3L)
```

Arguments

symbols	Character vector of genes.
max_tries	Number of attempts passed on to httr2::req_retry.

Value

DataFrame with 4 columns: "Submitted" symbol character vector in the same order as the symbols input parameter, corresponding "Entrez" integer IDs, "OfficialSymbol" character vector, and "Description" of gene.

Examples

```
# Sample lookup call of the ToppGene API specification:
# - FLDB is an obsolete symbol for APOB.
# - APOE is the current symbol for APOE.
# - ENSG00000113196 is an ensembl gene symbol for HAND1.
# - ENSMUSG00000020287 is a mouse gene MPG.
lookup(c("FLDB", "APOE", "ENSG00000113196", "ENSMUSG00000020287"))
```

---

lookup_pubchem *Return table of drug identifiers to PubChem CIDs.*

Description

Many downstream drug analyses in Bioconductor make use of PubChem CIDs but ToppGened drug identifiers require changes prior to conversion, and the conversion itself is involved when scaling to large lists of identifiers.

Usage

```
lookup_pubchem(df)
```

Arguments

df	DataFrame with 15 columns containing the enrichment Category, ID, and associated data.

Details

Therefore this function submits queries to the PubChem Power User Gateway (PUG) in parallel for each query.

Value

DataFrame with 2 columns subset to Category == Drug containing input Source, ID, and output CID.

Examples

```
library(DFplyr)
cats <- CategoriesDataFrame()
cats <-
    cats |>
    mutate(
        PValue = case_when(
            grepl("Drug", rownames(cats)) ~ PValue,
            .default = 1e-100),
        MaxResults = case_when(
            grepl("Drug", rownames(cats)) ~ 1000L,
            .default = 1L))
# EGFR gene that has hits in all drug databases.
df_enrich <- toppgene::enrich(1956L, cats)
df_cid <- lookup_pubchem(df_enrich)
df_cid
```

`lookup_pubchem_` *Convert identifiers of a single ToppGene drug database to PubChem CIDs.*

Description

Map external Registry IDs to PubChem CIDs using the PubChem Power User Gateway (PUG) (https://pubchem.ncbi.nlm.nih.gov/docs/power-user-gateway), also specified by the NCBI identifer exchange service: https://pubchem.ncbi.nlm.nih.gov/idexchange/

Usage

```
lookup_pubchem_(ids, registry = NULL)
```

Arguments

`ids`	character vector of one or more Registry identifiers.
`registry`	optional character vector of length one with Registry name. Not specifying this argument falls back to a PubChem synonym lookup.

Value

DataFrame with PubChem CIDs or NAs guaranteed to be the length as the input when using a registry. The DataFrame may have more rows than the input when using a non-registry synonym lookup.

Index