Package ‘clusterProfiler’

February 27, 2024

Type Package

Title A universal enrichment tool for interpreting omics data

Version 4.10.0

Maintainer Guangchuang Yu <guangchuangyu@gmail.com>

Description This package supports functional characteristics of both coding and non-coding genomics data for thousands of species with up-to-date gene annotation. It provides a universal interface for gene functional annotation from a variety of sources and thus can be applied in diverse scenarios. It provides a tidy interface to access, manipulate, and visualize enrichment results to help users achieve efficient data interpretation. Datasets obtained from multiple treatments and time points can be analyzed and compared in a single run, easily revealing functional consensus and differences among distinct conditions.

Depends R (>= 3.5.0)

Imports AnnotationDbi, downloader, DOSE (>= 3.23.2), dplyr, enrichplot (>= 1.9.3), GO.db, GOSemSim (>= 2.27.2), gson (>= 0.0.7), httr, igraph, magrittr, methods, plyr, qvalue, rlang, stats, tidyr, utils, yulab.utils (>= 0.0.7)

Suggests AnnotationHub, knitr, jsonlite, readr, rmarkdown, org.Hs eg.db, prettydoc, BiocManager, testthat

VignetteBuilder knitr

ByteCompile true

License Artistic-2.0


BugReports https://github.com/GuangchuangYu/clusterProfiler/issues

biocViews Annotation, Clustering, GeneSetEnrichment, GO, KEGG, MultipleComparison, Pathways, Reactome, Visualization

Encoding UTF-8

RoxygenNote 7.2.3

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

git_branch RELEASE_3_18
R topics documented:

- clusterProfiler-package
- append_kegg_category
- bitr
- bitr_kegg
- browseKEGG
- compareCluster
- DataSet
- download_KEGG
- dropGO
- enrichDAVID
- enricher
- enrichGO
- enrichKEGG
- enrichMKEGG
- enrichPC
- enrichWP
- getPPI
- getTaxID
- getTaxInfo
- get_wp_organisms
- Gff2GeneTable
- go2ont
- go2term
- gofilter
- groupGO
- groupGOResult-class
- GSEA
- gseGO
- gseKEGG
- gseMKEGG
clusterProfiler-package

statistical analysis and visualization of functional profiles for genes and gene clusters

Description

The package implements methods to analyze and visualize functional profiles of gene and gene clusters.

append_kegg_category

Description

add KEGG pathway category information

Usage

append_kegg_category(x)

Arguments

x

KEGG enrichment result

Details

This function appends the KEGG pathway category information to KEGG enrichment result (either output of 'enrichKEGG' or 'gseKEGG')
**Value**

update KEGG enrichment result with category information

**Author(s)**

Guangchuang Yu

---

**Description**

Biological Id TRanslator

**Usage**

```
bitr(geneID, fromType, toType, OrgDb, drop = TRUE)
```

**Arguments**

- `geneID` : input gene id
- `fromType` : input id type
- `toType` : output id type
- `OrgDb` : annotation db
- `drop` : drop NA or not

**Value**

`data.frame`

**Author(s)**

Guangchuang Yu
**bitr_kegg**

Description

convert biological ID using KEGG API

Usage

```r
bitr_kegg(geneID, fromType, toType, organism, drop = TRUE)
```

Arguments

- `geneID`: input gene id
- `fromType`: input id type
- `toType`: output id type
- `organism`: supported organism, can be search using search_kegg_organism function
- `drop`: drop NA or not

Value

data.frame

Author(s)

Guangchuang Yu

---

**browseKEGG**

Description

open KEGG pathway with web browser

Usage

```r
browseKEGG(x, pathID)
```

Arguments

- `x`: an instance of enrichResult or gseaResult
- `pathID`: pathway ID
compareCluster

Value
url

Author(s)
Guangchuang Yu

---

**compareCluster**

*Compare gene clusters functional profile*

**Description**

Given a list of gene set, this function will compute profiles of each gene cluster.

**Usage**

```r
compareCluster(
  geneClusters,
  fun = "enrichGO",
  data = "",
  source_from = NULL,
  ...
)
```

**Arguments**

- **geneClusters**
  a list of entrez gene id. Alternatively, a formula of type `Entrez~group` or a formula of type `Entrez | logFC ~ group` for "gseGO", "gseKEGG" and "GSEA".
- **fun**
  One of "groupGO", "enrichGO", "enrichKEGG", "enrichDO" or "enrichPathway". Users can also supply their own function.
- **data**
  if geneClusters is a formula, the data from which the clusters must be extracted.
- **source_from**
  If using a custom function in "fun", provide the source package as a string here. Otherwise, the function will be obtained from the global environment.
- **...**
  Other arguments.

**Value**

A `clusterProfResult` instance.

**Author(s)**

Guangchuang Yu [https://yulab-smu.top](https://yulab-smu.top)

**See Also**

- `compareClusterResult-class` 
- `groupGO` 
- `enrichGO`
Examples

```r
## Not run:
data(gcSample)
xx <- compareCluster(gcSample, fun="enrichKEGG",
                      organism="hsa", pvalueCutoff=0.05)
as.data.frame(xx)
# plot(xx, type="dot", caption="KEGG Enrichment Comparison")
dotplot(xx)

## formula interface
mydf <- data.frame(Entrez=c('1', '100', '1000', '100101467',
                            '100127206', '100128071'),
                    logFC = c(1.1, -0.5, 5, 2.5, -3, 3),
                    group = c('A', 'A', 'A', 'B', 'B', 'B'),
                    othergroup = c('good', 'good', 'bad', 'good', 'good', 'bad'))
xx.formula <- compareCluster(Entrez~group, data=mydf,
                            fun='groupGO', OrgDb='org.Hs.eg.db')
as.data.frame(xx.formula)

## formula interface with more than one grouping variable
xx.formula.twogroups <- compareCluster(Entrez~group+othergroup, data=mydf,
                                    fun='groupGO', OrgDb='org.Hs.eg.db')
as.data.frame(xx.formula.twogroups)
```

## End(Not run)

**DataSet**

**Datasets gcSample contains a sample of gene clusters.**

**Description**

Datasets gcSample contains a sample of gene clusters.
Datasets kegg_species contains kegg species information
Datasets kegg_category contains kegg pathway category information
Datasets DE_GSE8057 contains differential expressed genes obtained from GSE8057 dataset

**download_KEGG**

**Description**

download the latest version of KEGG pathway/module

**Usage**

download_KEGG(species, keggType = "KEGG", keyType = "kegg")
Arguments

species species
keggType one of 'KEGG' or 'MKEGG'
keyType supported keyType, see bitr_kegg

Value

list

Author(s)

Guangchuang Yu

dropGO

dropGO

description

drop GO term of specific level or specific terms (mostly too general).

Usage

dropGO(x, level = NULL, term = NULL)

Arguments

x an instance of 'enrichResult' or 'compareClusterResult'
level GO level
term GO term

Value

modified version of x

Author(s)

Guangchuang Yu
Description

enrichment analysis by DAVID

Usage

enrichDAVID(
  gene,
  idType = "ENTREZ_GENE_ID",
  universe,
  minGSSize = 10,
  maxGSSize = 500,
  annotation = "GOTERM_BP_FAT",
  pvalueCutoff = 0.05,
  pAdjustMethod = "BH",
  qvalueCutoff = 0.2,
  species = NA,
  david.user
)

Arguments

gene  input gene
idType  id type
universe  background genes. If missing, the all genes listed in the database (eg TERM2GENE table) will be used as background.
minGSSize  minimal size of genes annotated for testing
maxGSSize  maximal size of genes annotated for testing
annotation  david annotation
pvalueCutoff  adjusted pvalue cutoff on enrichment tests to report
pAdjustMethod  one of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"
qvalueCutoff  qvalue cutoff on enrichment tests to report as significant. Tests must pass i) pvalueCutoff on unadjusted pvalues, ii) pvalueCutoff on adjusted pvalues and iii) qvalueCutoff on qvalues to be reported.
species  species
david.user  david user

Value

A enrichResult instance
Author(s)

Guangchuang Yu

Description

A universal enrichment analyzer

Usage

enricher(
  gene,
  pvalueCutoff = 0.05,
  pAdjustMethod = "BH",
  universe = NULL,
  minGSSize = 10,
  maxGSSize = 500,
  qvalueCutoff = 0.2,
  gson = NULL,
  TERM2GENE,
  TERM2NAME = NA
)

Arguments

gene a vector of gene id
pvalueCutoff adjusted pvalue cutoff on enrichment tests to report
pAdjustMethod one of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"
universe background genes. If missing, the all genes listed in the database (eg TERM2GENE table) will be used as background.
minGSSize minimal size of genes annotated for testing
maxGSSize maximal size of genes annotated for testing
qvalueCutoff qvalue cutoff on enrichment tests to report as significant. Tests must pass i) pvalueCutoff on unadjusted pvalues, ii) pvalueCutoff on adjusted pvalues and iii) qvalueCutoff on qvalues to be reported.
gson a GSON object, if not NULL, use it as annotation data.
TERM2GENE user input annotation of TERM TO GENE mapping, a data.frame of 2 column with term and gene. Only used when gson is NULL.
TERM2NAME user input of TERM TO NAME mapping, a data.frame of 2 column with term and name. Only used when gson is NULL.
**enrichGO**

Value

A enrichResult instance

Author(s)

Guangchuang Yu [https://yulab-smu.top](https://yulab-smu.top)

---

enrichGO

*GO Enrichment Analysis of a gene set. Given a vector of genes, this function will return the enrichment GO categories after FDR control.*

Description

GO Enrichment Analysis of a gene set. Given a vector of genes, this function will return the enrichment GO categories after FDR control.

Usage

```r
enrichGO(
  gene,
  OrgDb,
  keyType = "ENTREZID",
  ont = "MF",
  pvalueCutoff = 0.05,
  pAdjustMethod = "BH",
  universe,
  qvalueCutoff = 0.2,
  minGSSize = 10,
  maxGSSize = 500,
  readable = FALSE,
  pool = FALSE
)
```

Arguments

gene a vector of entrez gene id.
OrgDb OrgDb
keyType keytype of input gene
ont One of "BP", "MF", and "CC" subontologies, or "ALL" for all three.
pvalueCutoff adjusted pvalue cutoff on enrichment tests to report
pAdjustMethod one of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"
universe background genes. If missing, the all genes listed in the database (eg TERM2GENE table) will be used as background.
qvalueCutoff  qvalue cutoff on enrichment tests to report as significant. Tests must pass i) pvalueCutoff on unadjusted pvalues, ii) pvalueCutoff on adjusted pvalues and iii) qvalueCutoff on qvalues to be reported.

minGSSize  minimal size of genes annotated by Ontology term for testing.

maxGSSize  maximal size of genes annotated for testing

readable  whether mapping gene ID to gene Name

pool  If ont='ALL', whether pool 3 GO sub-ontologies

Value

An enrichResult instance.

Author(s)

Guangchuang Yu https://yulab-smu.top

See Also

enrichResult-class, compareCluster

Examples

## Not run:
  data(geneList, package = "DOSE")
  de <- names(geneList)[1:100]
  yy <- enrichGO(de, 'org.Hs.eg.db', ont="BP", pvalueCutoff=0.01)
  head(yy)

## End(Not run)
enrichKEGG

pAdjustMethod = "BH",
universe,
minGSSize = 10,
maxGSSize = 500,
qvalueCutoff = 0.2,
use_internal_data = FALSE
)

Arguments

gene a vector of entrez gene id.
organism supported organism listed in 'https://www.genome.jp/kegg/catalog/org_list.html'
keyType one of "kegg", 'ncbi-geneid', 'ncbi-proteinid' and 'uniprot'
pvalueCutoff adjusted pvalue cutoff on enrichment tests to report
pAdjustMethod one of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"
universe background genes. If missing, the all genes listed in the database (eg TERM2GENE table) will be used as background.
minGSSize minimal size of genes annotated by Ontology term for testing.
maxGSSize maximal size of genes annotated for testing
qvalueCutoff qvalue cutoff on enrichment tests to report as significant. Tests must pass i) pvalueCutoff on unadjusted pvalues, ii) pvalueCutoff on adjusted pvalues and iii) qvalueCutoff on qvalues to be reported.

use_internal_data
logical, use KEGG.db or latest online KEGG data

Value

A enrichResult instance.

Author(s)

Guangchuang Yu https://yulab-smu.top

See Also

enrichResult-class, compareCluster

Examples

## Not run:
data(geneList, package='DOSE')
de <- names(geneList)[1:100]
yy <- enrichKEGG(de, pvalueCutoff=0.01)
head(yy)

## End(Not run)
enrichMKEGG  

**Description**

KEGG Module Enrichment Analysis of a gene set. Given a vector of genes, this function will return the enrichment KEGG Module categories with FDR control.

**Usage**

```r
enrichMKEGG(
  gene,
  organism = "hsa",
  keyType = "kegg",
  pvalueCutoff = 0.05,
  pAdjustMethod = "BH",
  universe,
  minGSSize = 10,
  maxGSSize = 500,
  qvalueCutoff = 0.2
)
```

**Arguments**

- `gene`: a vector of entrez gene id.
- `organism`: supported organism listed in 'https://www.genome.jp/kegg/catalog/org_list.html'
- `keyType`: one of "kegg", 'ncbi-geneid', 'ncbi-proteinid' and 'uniprot'
- `pvalueCutoff`: adjusted pvalue cutoff on enrichment tests to report
- `pAdjustMethod`: one of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"
- `universe`: background genes. If missing, the all genes listed in the database (eg TERM2GENE table) will be used as background.
- `minGSSize`: minimal size of genes annotated by Ontology term for testing.
- `maxGSSize`: maximal size of genes annotated for testing
- `qvalueCutoff`: qvalue cutoff on enrichment tests to report as significant. Tests must pass i) pvalueCutoff on unadjusted pvalues, ii) pvalueCutoff on adjusted pvalues and iii) qvalueCutoff on qvalues to be reported.

**Value**

A enrichResult instance.
**enrichPC**

---

**Description**

ORA analysis for Pathway Commons

**Usage**

```r
enrichPC(gene, source, keyType = "hgnc", ...)
```

**Arguments**

- `gene`: a vector of genes (either hgnc symbols or uniprot IDs)
- `source`: Data source of Pathway Commons, e.g., 'reactome', 'kegg', 'pathbank', 'netpath', 'panther', etc.
- `keyType`: specify the type of input 'gene' (one of 'hgnc' or 'uniprot')
- `...`: additional parameters, see also the parameters supported by the enricher() function

**Details**

This function performs over-representation analysis using Pathway Commons

**Value**

A enrichResult instance

---

**enrichWP**

---

**Description**

ORA analysis for WikiPathways

**Usage**

```r
enrichWP(gene, organism, ...)
```

**Arguments**

- `gene`: a vector of entrez gene id
- `organism`: supported organisms, which can be accessed via the get_wp_organisms() function
- `...`: additional parameters, see also the parameters supported by the enricher() function
getPPI

Details

This function performs over-representation analysis using WikiPathways

Value

A enrichResult instance

Author(s)

Guangchuang Yu

Usage

getPPI(
  x,
  ID = 1,
  taxID = "auto",
  required_score = NULL,
  network_type = "functional",
  add_nodes = 0,
  show_query_node_labels = 0,
  output = "igraph"
)

Arguments

x an ‘enrichResult‘ object or a vector of proteins, e.g. ‘c("PTCH1", "TP53", "BRCA1", "BRCA2")’

ID ID or index to extract genes in the enriched term(s) if ‘x’ is an ‘enrichResult’ object

taxID NCBI taxon identifiers (e.g. Human is 9606, see: [STRING organisms](https://string-db.org/cgi/input.pl?input_page_active_form=organisms)).

required_score threshold of significance to include a interaction, a number between 0 and 1000 (default depends on the network)

network_type network type: functional (default), physical

add_nodes adds a number of proteins with to the network based on their confidence score (default:1)
getTaxID

show_query_node_labels
    when available use submitted names in the preferredName column when (0 or
    1) (default:0)

output
    one of 'data.frame' or 'igraph'

Details

[Getting the STRING network interactions](https://string-db.org/cgi/help.pl?sessionId=btsvnCeNrBk7).

Value

a `data.frame` or an `igraph` object

Author(s)

Yonghe Xia and modified by Guangchuang Yu

description

Convert species scientific name to taxonomic ID

Usage

getTaxID(species)

Arguments

species
    scientific name of a species

Value

taxonomic ID

Author(s)

Guangchuang Yu
getTaxInfo

Description
Query taxonomy information from 'stringdb' or 'ensembl' web services

Usage
getTaxInfo(species, source = "stringdb")

Arguments
- species: scientific name of a species
- source: one of 'stringdb' or 'ensembl'

Value
a 'data.frame' of query information

Author(s)
Guangchuang Yu

get_wp_organisms

Description
list supported organism of WikiPathways

Usage
get_wp_organisms()

Details
This function extracts information from 'https://wikipathways-data.wmcloud.org/current/gmt/' and lists all supported organisms

Value
supported organism list

Author(s)
Guangchuang Yu
### Gff2GeneTable

**Description**

read GFF file and build gene information table

**Usage**

```r
Gff2GeneTable(gffFile, compress = TRUE)
```

**Arguments**

- `gffFile`: GFF file
- `compress`: compress file or not

**Details**

given a GFF file, this function extracts information from it and save it in working directory

**Value**

file save.

**Author(s)**

Yu Guangchuang

---

### go2ont

**Description**

convert goid to ontology (BP, CC, MF)

**Usage**

```r
go2ont(goid)
```

**Arguments**

- `goid`: a vector of GO IDs

**Value**

data.frame
**go2term**

**Description**
convert goid to descriptive term

**Usage**
go2term(goid)

**Arguments**
goid a vector of GO IDs

**Value**
data.frame

**Author(s)**
Guangchuang Yu

---

**gofilter**

**Description**
filter GO enriched result at specific level

**Usage**
gofilter(x, level = 4)

**Arguments**
x output from enrichGO or compareCluster
level GO level

**Value**
updated object
**groupGO**

**Author(s)**

Guangchuang Yu

---

**groupGO** *Functional Profile of a gene set at specific GO level. Given a vector of genes, this function will return the GO profile at a specific level.*

**Description**

Functional Profile of a gene set at specific GO level. Given a vector of genes, this function will return the GO profile at a specific level.

**Usage**

```r
groupGO(
  gene,
  OrgDb,
  keyType = "ENTREZID",
  ont = "CC",
  level = 2,
  readable = FALSE
)
```

**Arguments**

- **gene**: a vector of entrez gene id.
- **OrgDb**: OrgDb
- **keyType**: key type of input gene
- **ont**: One of "MF", "BP", and "CC" subontologies.
- **level**: Specific GO Level.
- **readable**: if readable is TRUE, the gene IDs will mapping to gene symbols.

**Value**

A groupGOResult instance.

**Author(s)**

Guangchuang Yu [https://yulab-smu.top](https://yulab-smu.top)

**See Also**

[groupGOResult-class], [compareCluster]
Examples

data(gcSample)
yy <- groupGO(gcSample[[1]], 'org.Hs.eg.db', ont="BP", level=2)
head(summary(yy))
#plot(yy)

---

**groupGOResult-class**  
Class "groupGOResult" This class represents the result of functional Profiles of a set of gene at specific GO level.

---

**Description**

Class "groupGOResult" This class represents the result of functional Profiles of a set of gene at specific GO level.

**Slots**

- **result**  GO classification result
- **ontology**  Ontology
- **level**  GO level
- **organism**  one of "human", "mouse" and "yeast"
- **gene**  Gene IDs
- **readable**  logical flag of gene ID in symbol or not.

**Author(s)**

Guangchuang Yu [https://yulab-smu.top](https://yulab-smu.top)

**See Also**

- `compareClusterResult`
- `compareCluster`
- `groupGO`

---

**GSEA**

---

**Description**

a universal gene set enrichment analysis tools
Usage

GSEA(
  geneList,
  exponent = 1,
  minGSSize = 10,
  maxGSSize = 500,
  eps = 1e-10,
  pvalueCutoff = 0.05,
  pAdjustMethod = "BH",
  gson = NULL,
  TERM2GENE,
  TERM2NAME = NA,
  verbose = TRUE,
  seed = FALSE,
  by = "fgsea",
  ...
)

Arguments

geneList order ranked geneList
exponent weight of each step
minGSSize minimal size of each geneSet for analyzing
maxGSSize maximal size of genes annotated for testing
eps This parameter sets the boundary for calculating the p value.
pvalueCutoff adjusted pvalue cutoff
pAdjustMethod one of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"
gson a GSON object, if not NULL, use it as annotation data.
TERM2GENE user input annotation of TERM TO GENE mapping, a data.frame of 2 column with term and gene. Only used when gson is NULL.
TERM2NAME user input of TERM TO NAME mapping, a data.frame of 2 column with term and name. Only used when gson is NULL.
verbose logical
seed logical
by one of 'fgsea' or 'DOSE'
...
other parameter

Value
gseaResult object

Author(s)
Guangchuang Yu https://yulab-smu.top
**Description**

Gene Set Enrichment Analysis of Gene Ontology

**Usage**

```r
gseGO(
geneList,  # order ranked geneList
ont = "BP",  # one of "BP", "MF", and "CC" subontologies, or "ALL" for all three.
OrgDb,  # OrgDb
keyType = "ENTREZID",  # keytype of gene
exponent = 1,  # weight of each step
minGSSize = 10,  # minimal size of each geneSet for analyzing
maxGSSize = 500,  # maximal size of genes annotated for testing
eps = 1e-10,  # This parameter sets the boundary for calculating the p value.
pvalueCutoff = 0.05,  # pvalue Cutoff
pAdjustMethod = "BH",  # pvalue adjustment method
verbose = TRUE,  # print message or not
seed = FALSE,  # logical
by = "fgsea",  # one of ’fgsea’ or ’DOSE’
...  # other parameter
)
```

**Arguments**

- `geneList`: order ranked geneList
- `ont`: one of "BP", "MF", and "CC" subontologies, or "ALL" for all three.
- `OrgDb`: OrgDb
- `keyType`: keytype of gene
- `exponent`: weight of each step
- `minGSSize`: minimal size of each geneSet for analyzing
- `maxGSSize`: maximal size of genes annotated for testing
- `eps`: This parameter sets the boundary for calculating the p value.
- `pvalueCutoff`: pvalue Cutoff
- `pAdjustMethod`: pvalue adjustment method
- `verbose`: print message or not
- `seed`: logical
- `by`: one of ’fgsea’ or ’DOSE’
**Value**

gseaResult object

**Author(s)**

Yu Guangchuang

---

**Description**

Gene Set Enrichment Analysis of KEGG

**Usage**

```r
gseKEGG(
  geneList,
  organism = "hsa",
  keyType = "kegg",
  exponent = 1,
  minGSSize = 10,
  maxGSSize = 500,
  eps = 1e-10,
  pvalueCutoff = 0.05,
  pAdjustMethod = "BH",
  verbose = TRUE,
  use_internal_data = FALSE,
  seed = FALSE,
  by = "fgsea",
  ...
)
```

**Arguments**

- `geneList`: order ranked geneList
- `organism`: supported organism listed in [https://www.genome.jp/kegg/catalog/org_list.html](https://www.genome.jp/kegg/catalog/org_list.html)
- `keyType`: one of "kegg", 'ncbi-geneid', 'ncib-proteinid' and 'uniprot'
- `exponent`: weight of each step
- `minGSSize`: minimal size of each geneSet for analyzing
- `maxGSSize`: maximal size of genes annotated for testing
- `eps`: This parameter sets the boundary for calculating the p value.
- `pvalueCutoff`: pvalue Cutoff
- `pAdjustMethod`: pvalue adjustment method
**gseMKEGG**

verbose: print message or not  
use_internal_data: logical, use KEGG.db or latest online KEGG data  
seed: logical  
by: one of 'fgsea' or 'DOSE'  
...

**Value**

gseaResult object

**Author(s)**

Yu Guangchuang

---

**gseMKEGG**

**gseMKEGG**

---

**Description**

Gene Set Enrichment Analysis of KEGG Module

**Usage**

gseMKEGG(  
geneList,  
organism = "hsa",  
keyType = "kegg",  
exponent = 1,  
minGSSize = 10,  
maxGSSize = 500,  
eps = 1e-10,  
pvalueCutoff = 0.05,  
pAdjustMethod = "BH",  
verbose = TRUE,  
seed = FALSE,  
by = "fgsea",  
...  
)

**Arguments**

geneList: order ranked geneList  
organism: supported organism listed in 'https://www.genome.jp/kegg/catalog/org_list.html'  
keyType: one of "kegg", 'ncbi-geneid', 'ncib-proteinid' and 'uniprot'  
exponent: weight of each step
### gsePC

**Arguments**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>geneList</td>
<td>a ranked gene list</td>
</tr>
<tr>
<td>source</td>
<td>Data source of Pathway Commons, e.g., 'reactome', 'kegg', 'pathbank', 'net-path', 'panther', etc.</td>
</tr>
<tr>
<td>keyType</td>
<td>specify the type of input 'gene' (one of 'hgnc' or 'uniprot')</td>
</tr>
<tr>
<td>...</td>
<td>additional parameters, see also the parameters supported by the GSEA() function</td>
</tr>
</tbody>
</table>

**Description**

GSEA analysis for Pathway Commons

**Usage**

```r
gsePC(geneList, source, keyType, ...)
```

**Details**

This function performs GSEA using Pathway Commons

**Value**

A gseaResult instance
**gseWP**

**Description**

GSEA analysis for WikiPathways

**Usage**

\[\text{gseWP(geneList, organism, ...)}\]

**Arguments**

- **geneList**: ranked gene list
- **organism**: supported organisms, which can be accessed via the get_wp_organisms() function
- **...**: additional parameters, see also the parameters supported by the GSEA() function

**Details**

This function performs GSEA using WikiPathways

**Value**

A gseaResult instance

**Author(s)**

Guangchuang Yu

---

**gson_GO**

**Description**

download the latest version of KEGG pathway and stored in a 'GSON' object

**Usage**

\[\text{gson_GO(OrgDb, keytype = "ENTREZID", ont = "BP")}\]
**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrgDb</td>
<td>OrgDb</td>
</tr>
<tr>
<td>keytype</td>
<td>keytype of genes.</td>
</tr>
<tr>
<td>ont</td>
<td>one of &quot;BP&quot;, &quot;MF&quot;, &quot;CC&quot;, and &quot;ALL&quot;</td>
</tr>
</tbody>
</table>

**Value**

a `GSON` object

---

**Description**

download the latest version of KEGG pathway and stored in a `GSON` object

**Usage**

```r
gson_KEGG(species, KEGG_Type = "KEGG", keyType = "kegg")
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>species</td>
<td>species</td>
</tr>
<tr>
<td>KEGG_Type</td>
<td>one of &quot;KEGG&quot; and &quot;MKEGG&quot;</td>
</tr>
<tr>
<td>keyType</td>
<td>one of &quot;kegg&quot;, 'ncbi-geneid', 'ncib-proteinid' and 'uniprot'.</td>
</tr>
</tbody>
</table>

**Value**

a `GSON` object

**Author(s)**

Guangchuang Yu
gson_KEGG_mapper  
Build KEGG annotation for novel species using KEGG Mapper

Description
KEGG Mapper service can annotate protein sequences for novel species with KO database, and KO annotation need to be converted into Pathway or Module annotation, which can then be used in 'clusterProfiler'

Usage
```r
gson_KEGG_mapper(
  file,
  format = c("BLAST", "Ghost", "Kofam"),
  type = c("pathway", "module"),
  species = NULL,
  ...
)
```

Arguments
- **file**: the name of the file which comes from the KEGG Mapper service, see Details for file format
- **format**: string indicate format of KEGG Mapper result
- **type**: string indicate annotation database
- **species**: your species, NULL if ignored
- **...**: pass to gson::gson()

Details
File is a two-column dataset with K numbers in the second column, optionally preceded by the user’s identifiers in the first column. This is consistent with the output files of automatic annotation servers, BlastKOALA, GhostKOALA, and KofamKOALA. KOALA (KEGG Orthology And Links Annotation) is KEGG’s internal annotation tool for K number assignment of KEGG GENES using SSEARCH computation. BlastKOALA and GhostKOALA assign K numbers to the user’s sequence data by BLAST and GHOSTX searches, respectively, against a nonredundant set of KEGG GENES. KofamKOALA is a new member of the KOALA family available at GenomeNet using the HMM profile search, rather than the sequence similarity search, for K number assignment. [see](https://www.kegg.jp/blastkoala/), [https://www.kegg.jp/ghostkoala/](https://www.kegg.jp/ghostkoala/) and [https://www.genome.jp/tools/kofamkoala/](https://www.genome.jp/tools/kofamkoala/) for more information.

Value
- a gson instance
Examples

```r
## Not run:
file = system.file('extdata', "kegg_mapper_blast.txt", package='clusterProfiler')
gson_KEGG_mapper(file, format = "BLAST", type = "pathway")
## End(Not run)
```

Description

Download the latest version of WikiPathways data and stored in a 'GSON' object

Usage

```r
gson_WP(organism)
```

Arguments

- `organism`: supported organism, which can be accessed via the `get_wp_organisms()` function.

Description

list ID types supported by annoDb

Usage

```r
idType(OrgDb = "org.Hs.eg.db")
```

Arguments

- `OrgDb`: annotation db

Value

character vector

Author(s)

Guangchuang Yu
## ko2name

**Description**
convert ko ID to descriptive name

**Usage**
ko2name(ko)

**Arguments**
- `ko` ko ID

**Value**
data.frame

**Author(s)**
guangchuang yu

## merge_result

**Description**
merge a list of enrichResult objects to compareClusterResult

**Usage**
merge_result(enrichResultList)

**Arguments**
- `enrichResultList` a list of enrichResult objects

**Value**
a compareClusterResult instance

**Author(s)**
Guangchuang Yu
Description

plot GO graph

Usage

plotGOgraph(
x,  
firstSigNodes = 10,  
useInfo = "all",  
sigForAll = TRUE,  
useFullNames = TRUE,  
...  
)

Arguments

x output of enrichGO or gseGO
firstSigNodes number of significant nodes (rectangle nodes in the graph)
useInfo additional info
sigForAll if TRUE the score/p-value of all nodes in the DAG is shown, otherwise only score will be shown
useFullNames logical
... additional parameter of showSigOfNodes, please refer to topGO

Value

GO DAG graph

Author(s)

Guangchuang Yu
read.gmt.pc

**Description**

Parse gmt file from Pathway Common

**Usage**

```r
read.gmt.pc(gmtfile, output = "data.frame")
```

**Arguments**

- `gmtfile` A gmt file
- `output` one of `"data.frame"` or `"GSON"`

**Details**

This function parse gmt file downloaded from Pathway common

**Value**

A data.frame or A GSON object depends on the value of `"output"`

---

**reexports**

*Objects exported from other packages*

---

**Description**

These objects are imported from other packages. Follow the links below to see their documentation.

- **DOSE** `geneID, geneInCategory, gsfilter, setReadable`
- **dplyr** `arrange, filter, group_by, mutate, n, rename, select, slice, summarise`
- **enrichplot** `cnetplot, dotplot, emapplot, gplot, gseaplot, heatplot, ridgeplot`
- **GOSemSim** `buildGOmap, read.blast2go, read.gaf`
- **gson** `read.gmt, read.gmt.wp`
- **magrittr** `%<>%`, `%>%`
search_kegg_organism

Description

search kegg organism, listed in https://www.genome.jp/kegg/catalog/org_list.html

Usage

search_kegg_organism(str, by = "scientific_name", ignore.case = FALSE, use_internal_data = TRUE)

Arguments

str string
by one of 'kegg.code', 'scientific_name' and 'common_name'
ignore.case TRUE or FALSE
use_internal_data logical, use kegg_species.rda or latest online KEGG data

Value

data.frame

Author(s)

Guangchuang Yu

simplify

simplify method

Description

simplify output from enrichGO and gseGO by removing redundancy of enriched GO terms
simplify output from compareCluster by removing redundancy of enriched GO terms
Usage

```r
## S4 method for signature 'enrichResult'
method <- simplify(
  x,
  cutoff = 0.7,
  by = "p.adjust",
  select_fun = min,
  measure = "Wang",
  semData = NULL
)

## S4 method for signature 'gseaResult'
simplify(x,
  cutoff = 0.7,
  by = "p.adjust",
  select_fun = min,
  measure = "Wang",
  semData = NULL
)

## S4 method for signature 'compareClusterResult'
simplify(x,
  cutoff = 0.7,
  by = "p.adjust",
  select_fun = min,
  measure = "Wang",
  semData = NULL
)
```

Arguments

- `x` output of enrichGO
- `cutoff` similarity cutoff
- `by` feature to select representative term, selected by `select_fun` function
- `select_fun` function to select feature passed by `by` parameter
- `measure` method to measure similarity
- `semData` GOSemSimDATA object

Value

updated enrichResult object
updated compareClusterResult object
**Author(s)**

Guangchuang Yu  
Gwang-Jin Kim and Guangchuang Yu

**References**

issue #28 https://github.com/GuangchuangYu/clusterProfiler/issues/28  
issue #162 https://github.com/GuangchuangYu/clusterProfiler/issues/162

---

**Description**

retrieves annotation data from uniprot

**Usage**

uniprot_get(taxID)

**Arguments**

taxID taxonomy ID

**Value**

gene table data frame

**Author(s)**

guangchuang yu
Index

* classes
  groupGOResult-class, 22
* datasets
  DataSet, 7
* internal
  reexports, 34
* manip
  compareCluster, 6
  enrichGO, 11
  enrichKEGG, 12
  groupGO, 21
  %<>%(reexports), 34
  %%(reexports), 34
  %<>%, 34
  %%, 34
  append_kegg_category, 3
  arrange, 34
  arrange(reexports), 34
  bitr, 4
  bitr_kegg, 5
  browseKEGG, 5
  buildGOmap, 34
  buildGOmap(reexports), 34
  clusterProfiler
    (clusterProfiler-package), 3
  clusterProfiler-package, 3
  cnetplot, 34
  cnetplot(reexports), 34
  compareCluster, 6, 12, 13, 21, 22
  compareClusterResult, 22
  DataSet, 7
  DE_GSE8057(DataSet), 7
  dotplot, 34
  dotplot(reexports), 34
  download_KEGG, 7
  dropGO, 8
  emapplot, 34
  emapplot(reexports), 34
  enrichDAVID, 9
  enricher, 10
  enrichGO, 6, 11
  enrichKEGG, 12
  enrichMKEGG, 14
  enrichPC, 15
  enrichWP, 15
  filter, 34
  filter(reexports), 34
  gcSample(DataSet), 7
  geneID, 34
  geneID(reexports), 34
  geneInCategory, 34
  geneInCategory(reexports), 34
  get_wp_organisms, 18
  getPPI, 16
  getTaxID, 17
  getTaxInfo, 18
  Gff2GeneTable, 19
  go2ont, 19
  go2term, 20
  gofilter, 20
  goplot, 34
  goplot(reexports), 34
  group_by, 34
  group_by(reexports), 34
  groupGO, 6, 21, 22
  groupGOResult-class, 22
  GSEA, 22
  gseaplot, 34
  gseaplot(reexports), 34
  gseGO, 24
  gseKEGG, 25
  gseMKEGG, 26
  gsePC, 27
  gseWP, 28
gsfilter, 34
gsfilter (reexports), 34
gson.GO, 28
gson.KEGG, 29
gson.KEGG_mapper, 30
gson.WP, 31
heatplot, 34
heatplot (reexports), 34
idType, 31
kegg_category (DataSet), 7
kegg_species (DataSet), 7
ko2name, 32
merge_result, 32
mutate, 34
mutate (reexports), 34
n, 34
n (reexports), 34
plotGOgraph, 33
read.blast2go, 34
read.blast2go (reexports), 34
read.gaf, 34
read.gaf (reexports), 34
read.gmt, 34
read.gmt (reexports), 34
read.gmt.pc, 34
read.gmt.wp, 34
reexports, 34
rename, 34
rename (reexports), 34
ridgeplot, 34
ridgeplot (reexports), 34
search_kegg_organism, 35
select, 34
select (reexports), 34
setReadable, 34
setReadable (reexports), 34
show, groupGOResult-method (groupGOResult-class), 22
simplify, 35
simplify, compareClusterResult-method (simplify), 35
simplify, enrichResult-method (simplify), 35
simplify, gseaResult-method (simplify), 35
slice, 34
slice (reexports), 34
summarise, 34
summarise (reexports), 34
uniprot_get, 37