

Package: enrichViewNet (via r-universe)

June 3, 2026

Type Package

Version 1.10.0

Date 2023-08-12

Title From functional enrichment results to biological networks

Description This package enables the visualization of functional enrichment results as network graphs. First the package enables the visualization of enrichment results, in a format corresponding to the one generated by gprofiler2, as a customizable Cytoscape network. In those networks, both gene datasets (GO terms/pathways/protein complexes) and genes associated to the datasets are represented as nodes. While the edges connect each gene to its dataset(s). The package also provides the option to create enrichment maps from functional enrichment results. Enrichment maps enable the visualization of enriched terms into a network with edges connecting overlapping genes.

Encoding UTF-8

License Artistic-2.0

Depends R (>= 4.2.0)

Imports gprofiler2, strex, RCy3, jsonlite, stringr, enrichplot, igraph, reshape2, methods

Suggests BiocStyle, knitr, rmarkdown, ggplot2, scatterpie, ggtangle, ggrepel, testthat, ggnetwork, magick

biocViews BiologicalQuestion, Software, Network, NetworkEnrichment, GO

VignetteBuilder knitr

URL <https://github.com/adeschen/enrichViewNet>,
<https://adeschen.github.io/enrichViewNet/>

BugReports <https://github.com/adeschen/enrichViewNet/issues>

RoxygenNote 7.3.3

Config/pak/sysreqs libcairo2-dev cmake libfontconfig1-dev libfreetype6-dev libglpk-dev make libicu-dev libpng-dev libuv1-dev libxml2-dev libssl-dev python3 libzmq3-dev zlib1g-dev

Repository <https://bioc-release.r-universe.dev>
Date/Publication 2026-04-28 13:01:49 UTC
RemoteUrl <https://github.com/bioc/enrichViewNet>
RemoteRef RELEASE_3_23
RemoteSha d4a98cc431bcb529318c3204b2e90dbbd7e2c528

Contents

enrichViewNet-package	2
createEnrichMap	3
createEnrichMapAsIgraph	5
createEnrichMapMultiBasic	7
createEnrichMapMultiBasicAsIgraph	9
createEnrichMapMultiComplex	11
createEnrichMapMultiComplexAsIgraph	13
createNetwork	16
demoGOST	18
parentalNapaVsDMSODEG	19
parentalNapaVsDMSOEnrichment	21
rosaNapaVsDMSODEG	23
rosaNapaVsDMSOEnrichment	25
validateCreateEnrichMapMultiComplexAsIgraphArg	26
Index	28

enrichViewNet-package *From functional enrichment results to biological networks*

Description

The enrichViewNet package enables the visualization of enrichment results, in a format corresponding to the one generated by gprofiler2 (<https://cran.r-project.org/web/packages/gprofiler2/index.html>) under the form of Cytoscape network (<https://cytoscape.org/>).

Details

In those networks, both gene datasets (GO terms/pathways/protein complexes) and genes are represented as nodes. A edge connect a gene to its datasets. In the current version, only genes present in at least one gene dataset are retained.

Author(s)

Astrid Deschênes, Pascal Belleau, Robert L Faure, Maria J Fernandes, David A Tuveson
 Maintainer: Astrid Deschênes <adeschen@hotmail.com>

See Also

- [createNetwork](#) for transforming functional enrichment results from gprofiler2 into a Cytoscape network
- [createEnrichMap](#) for transforming functional enrichment results from gprofiler2 into an enrichment map in ggplot format
- [createEnrichMapAsIgraph](#) for transforming functional enrichment results from gprofiler2 into an enrichment map in igraph format
- [createEnrichMapMultiComplex](#) for transforming functional enrichment results from multiple gprofiler2 enrichment results into an enrichment map in ggplot format
- [createEnrichMapMultiComplexAsIgraph](#) for transforming functional enrichment results from multiple gprofiler2 enrichment results into an enrichment map in igraph format

createEnrichMap	<i>Using functional enrichment results in gprofiler2 format to create an enrichment map</i>
-----------------	---

Description

User selected enrichment terms are used to create an enrichment map. The selection of the term can be by specifying by the source of the terms (GO:MF, REAC, TF, etc.) or by listing the selected term IDs. The map is only generated when there is at least one significant term to graph.

Usage

```
createEnrichMap(
  gostObject,
  query,
  source = c("TERM_ID", "GO:MF", "GO:CC", "GO:BP", "KEGG", "REAC", "TF", "MIRNA", "HPA",
    "CORUM", "HP", "WP"),
  termIDs = NULL,
  removeRoot = TRUE,
  showCategory = 30L,
  categoryLabel = 1,
  categoryNode = 1,
  line = 1,
  ...
)
```

Arguments

gostObject	a list corresponding to gprofiler2 enrichment output that contains and that contains the results from an enrichment analysis.
query	a character string representing the name of the query that is going to be used to generate the graph. The query must exist in the gostObject object.

source	a character string representing the selected source that will be used to generate the network. To hand-pick the terms to be used, "TERM_ID" should be used and the list of selected term IDs should be passed through the termIDs parameter. The possible sources are "GO:BP" for Gene Ontology Biological Process, "GO:CC" for Gene Ontology Cellular Component, "GO:MF" for Gene Ontology Molecular Function, "KEGG" for Kegg, "REAC" for Reactome, "TF" for TRANSFAC, "MIRNA" for miRTarBase, "CORUM" for CORUM database, "HP" for Human phenotype ontology and "WP" for WikiPathways. Default: "TERM_ID".
termIDs	a vector of character strings that contains the term IDs retained for the creation of the network. Default: NULL.
removeRoot	a logical that specified if the root terms of the selected source should be removed (when present). Default: TRUE.
showCategory	a positive integer representing the maximum number of terms to display. If a integer, the first n terms will be displayed. If NULL, all terms will be displayed. Default: 30L.
categoryLabel	a positive numeric representing the amount by which plotting category nodes label size should be scaled relative to the default (1). Default: 1.s
categoryNode	a positive numeric representing the amount by which plotting category nodes should be scaled relative to the default (1). Default: 1.
line	a non-negative numeric representing the scale of line width. Default: 1.
...	additional arguments that will be pass to the emapplot function.

Value

a ggplot object which is the enrichment map for enrichment results.

Author(s)

Astrid Deschênes

Examples

```
## Loading dataset containing result from an enrichment analysis done with
## gprofiler2
data(parentalNapaVsDMSOEnrichment)

## Extract query information (only one in this dataset)
query <- unique(parentalNapaVsDMSOEnrichment$result$query)

## Create graph for Gene Ontology - Cellular Component related results
createEnrichMap(gostObject=parentalNapaVsDMSOEnrichment,
  query=query, source="GO:CC", removeRoot=TRUE)
```

```
createEnrichMapAsIgraph
```

Using functional enrichment results in gprofiler2 format to create an enrichment map in an igrph format

Description

User selected enrichment terms are used to create an enrichment map. The selection of the term can be specified by the source of the terms (GO:MF, REAC, TF, etc.) or by listing the selected term IDs. The map is only generated when there is at least one significant term to graph. The output is an igrph.

Usage

```
createEnrichMapAsIgraph(
  gostObject,
  query,
  source = c("TERM_ID", "GO:MF", "GO:CC", "GO:BP", "KEGG", "REAC", "TF", "MIRNA", "HPA",
    "CORUM", "HP", "WP"),
  termIDs = NULL,
  removeRoot = TRUE,
  showCategory = 30L,
  similarityCutOff = 0.2
)
```

Arguments

<code>gostObject</code>	a list corresponding to gprofiler2 enrichment output that contains and that contains the results from an enrichment analysis.
<code>query</code>	a character string representing the name of the query that is going to be used to generate the graph. The query must exist in the <code>gostObject</code> object.
<code>source</code>	a character string representing the selected source that will be used to generate the network. To hand-pick the terms to be used, "TERM_ID" should be used and the list of selected term IDs should be passed through the <code>termIDs</code> parameter. The possible sources are "GO:BP" for Gene Ontology Biological Process, "GO:CC" for Gene Ontology Cellular Component, "GO:MF" for Gene Ontology Molecular Function, "KEGG" for Kegg, "REAC" for Reactome, "TF" for TRANSFAC, "MIRNA" for miRTarBase, "CORUM" for CORUM database, "HP" for Human phenotype ontology and "WP" for WikiPathways. Default: "TERM_ID".
<code>termIDs</code>	a vector of character strings that contains the term IDs retained for the creation of the network. Default: NULL.
<code>removeRoot</code>	a logical that specified if the root terms of the selected source should be removed (when present). Default: TRUE.

showCategory a positive integer representing the maximum number of terms to display. If a integer, the first n terms will be displayed. If NULL, all terms will be displayed. Default: 30L.

similarityCutOff a positive numeric between 0 and 1 indicating the minimum level of similarity between two terms to have an edge linking the terms. Default: 0.20.

Value

a igraph object which is the enrichment map for enrichment results. The node have 2 attributes: "name" and "size". The "name" corresponds to the term description. While the "size" corresponds to the number of genes found in the specific gene set. The edges have 3 attributes: "similarity", "width", and "weight". All those 3 attributes correspond to the Jaccard coefficient.

Author(s)

Astrid Deschênes

Examples

```
## Loading dataset containing result from an enrichment analysis done with
## gprofiler2
data(parentalNapaVsDMSOEnrichment)

## Extract query information (only one in this dataset)
query <- unique(parentalNapaVsDMSOEnrichment$result$query)

## Create graph for Gene Ontology - Cellular Component related results
mapG <- createEnrichMapAsIgraph(gostObject=parentalNapaVsDMSOEnrichment,
  query=query, source="GO:CC", removeRoot=TRUE,
  showCategory=30L, similarityCutOff=0.20)

## Required library igraph to show the graph
if(requireNamespace("igraph", quietly=TRUE)) {
  ## Using library igraph to show the graph
  library(igraph)
  plot(mapG)
}

if (requireNamespace("ggplot2", quietly=TRUE) &&
  requireNamespace("igraph", quietly=TRUE) &&
  requireNamespace("ggtangle", quietly=TRUE) &&
  requireNamespace("ggnetwork", quietly=TRUE)) {
  ## Using more complex set of libraries to display personalized graph
  library(ggplot2)
  library(igraph)
  library(ggnetwork)
  library(ggtangle)
  emapG <- ggplot(mapG, layout=layout_with_fr) +
    geom_edge(color="gray", linewidth=1) +
    geom_nodes(aes(size=size)) +
    geom_nodetext(aes(label=name), color="black", size=3) +
```

```

        theme_void()
    }
    }
}

```

createEnrichMapMultiBasic

Using functional enrichment results in gprofiler2 format to create an enrichment map with multiple groups from different enrichment analyses

Description

User selected enrichment terms are used to create an enrichment map. The selection of the term can be by specifying by the source of the terms (GO:MF, REAC, TF, etc.) or by listing the selected term IDs. The map is only generated when there is at least one significant term to graph.

Usage

```

createEnrichMapMultiBasic(
  gostObjectList,
  queryList,
  source = c("TERM_ID", "GO:MF", "GO:CC", "GO:BP", "KEGG", "REAC", "TF", "MIRNA", "HPA",
    "CORUM", "HP", "WP"),
  termIDs = NULL,
  removeRoot = TRUE,
  showCategory = 30L,
  categoryLabel = 1,
  categoryNode = 1,
  line = 1,
  ...
)

```

Arguments

- | | |
|----------------|--|
| gostObjectList | a list of gprofiler2 objects that contain the results from an enrichment analysis. The list must contain at least 2 entries. The number of entries must correspond to the number of entries for the queryList parameter. |
| queryList | a list of character strings representing the names of the queries that are going to be used to generate the graph. The query names must exist in the associated gostObjectList objects and follow the same order. The number of entries must correspond to the number of entries for the gostObjectList parameter. |
| source | a character string representing the selected source that will be used to generate the network. To hand-pick the terms to be used, "TERM_ID" should be used and the list of selected term IDs should be passed through the termIDs parameter. The possible sources are "GO:BP" for Gene Ontology Biological Process, "GO:CC" for Gene Ontology Cellular Component, "GO:MF" for Gene |

	Ontology Molecular Function, "KEGG" for Kegg, "REAC" for Reactome, "TF" for TRANSFAC, "MIRNA" for miRTarBase, "CORUM" for CORUM database, "HP" for Human phenotype ontology and "WP" for WikiPathways. Default: "TERM_ID".
termIDs	a vector of character strings that contains the term IDs retained for the creation of the network. Default: NULL.
removeRoot	a logical that specified if the root terms of the selected source should be removed (when present). Default: TRUE.
showCategory	a positive integer or a vector of characters representing terms. If a integer, the first n terms will be displayed. If vector of terms, the selected terms will be displayed. Default: 30L.
categoryLabel	a positive numeric representing the amount by which plotting category nodes label size should be scaled relative to the default (1). Default: 1.
categoryNode	a positive numeric representing the amount by which plotting category nodes should be scaled relative to the default (1). Default: 1.
line	a non-negative numeric representing the scale of line width. Default: 1.
...	additional arguments that will be pass to the emapplot function.

Value

a ggplot object which is the enrichment map for enrichment results.

Author(s)

Astrid Deschênes

Examples

```
## Loading dataset containing results from 2 enrichment analyses done with
## gprofiler2
data(parentalNapaVsDMSOEnrichment)
data(rosaNapaVsDMSOEnrichment)

## Extract query information (only one in each dataset)
query1 <- unique(parentalNapaVsDMSOEnrichment$result$query)[1]
query2 <- unique(rosaNapaVsDMSOEnrichment$result$query)[1]

## Create graph for KEGG related results from
## 2 enrichment analyses
createEnrichMapMultiBasic(gostObjectList=list(parentalNapaVsDMSOEnrichment,
      rosaNapaVsDMSOEnrichment),
      queryList=list(query1, query2), source="KEGG", removeRoot=TRUE)
```

```
createEnrichMapMultiBasicAsIgraph
```

Using functional enrichment results in gprofiler2 format to create an enrichment map with multiple groups from different enrichment analyses as an igrph

Description

User selected enrichment terms are used to create an enrichment map. The selection of the term can be by specifying by the source of the terms (GO:MF, REAC, TF, etc.) or by listing the selected term IDs. The map is only generated when there is at least one significant term to graph. The output is an enrichment map in igrph format.

Usage

```
createEnrichMapMultiBasicAsIgraph(
  gostObjectList,
  queryList,
  source = c("TERM_ID", "GO:MF", "GO:CC", "GO:BP", "KEGG", "REAC", "TF", "MIRNA", "HPA",
    "CORUM", "HP", "WP"),
  termIDs = NULL,
  removeRoot = TRUE,
  showCategory = 30L,
  similarityCutOff = 0.2
)
```

Arguments

<code>gostObjectList</code>	a list of gprofiler2 objects that contain the results from an enrichment analysis. The list must contain at least 2 entries. The number of entries must correspond to the number of entries for the <code>queryList</code> parameter.
<code>queryList</code>	a list of character strings representing the names of the queries that are going to be used to generate the graph. The query names must exist in the associated <code>gostObjectList</code> objects and follow the same order. The number of entries must correspond to the number of entries for the <code>gostObjectList</code> parameter.
<code>source</code>	a character string representing the selected source that will be used to generate the network. To hand-pick the terms to be used, "TERM_ID" should be used and the list of selected term IDs should be passed through the <code>termIDs</code> parameter. The possible sources are "GO:BP" for Gene Ontology Biological Process, "GO:CC" for Gene Ontology Cellular Component, "GO:MF" for Gene Ontology Molecular Function, "KEGG" for Kegg, "REAC" for Reactome, "TF" for TRANSFAC, "MIRNA" for miRTarBase, "CORUM" for CORUM database, "HP" for Human phenotype ontology and "WP" for WikiPathways. Default: "TERM_ID".
<code>termIDs</code>	a vector of character strings that contains the term IDs retained for the creation of the network. Default: NULL.

removeRoot	a logical that specified if the root terms of the selected source should be removed (when present). Default: TRUE.
showCategory	a positive integer representing the maximum number of terms to display. If a integer, the first n terms will be displayed. If NULL, all terms will be displayed. Default: 30L.
similarityCutOff	a positive numeric between 0 and 1 indicating the minimum level of similarity between two terms, calculated using the Jaccard coefficient, to have an edge linking the terms. Default: 0.20.

Value

a igraph object which is the enrichment map for enrichment results. The node have 5 attributes: "name", "size", "pie", "cluster", and "pieName". The "name" corresponds to the term description. While the "size" corresponds to the number of unique genes found in the specific gene set when looking at all the experiments. The edges have 3 attributes: "similarity", "width", and "weight". All those 3 attributes correspond to the Jaccard coefficient.

Author(s)

Astrid Deschênes

Examples

```
## Loading dataset containing results from 2 enrichment analyses done with
## gprofiler2
data(parentalNapaVsDMSOEnrichment)
data(rosaNapaVsDMSOEnrichment)

## Extract query information (only one in each dataset)
query1 <- unique(parentalNapaVsDMSOEnrichment$result$query)[1]
query2 <- unique(rosaNapaVsDMSOEnrichment$result$query)[1]

## Create graph for KEGG related results from
## 2 enrichment analyses
emapG <- createEnrichMapMultiBasicAsIgraph(gostObjectList=
  list(parentalNapaVsDMSOEnrichment, rosaNapaVsDMSOEnrichment),
  queryList=list(query1, query2), source="KEGG", removeRoot=TRUE,
  similarityCutOff=0.3)

if (requireNamespace("ggplot2", quietly=TRUE) &&
    requireNamespace("igraph", quietly=TRUE) &&
    requireNamespace("scatterpie", quietly=TRUE) &&
    requireNamespace("ggtangle", quietly=TRUE) &&
    requireNamespace("ggrepel", quietly=TRUE)) {
  ## Create a visual representation of the enrichment map
  ## by default
  library(igraph)
  plot(emapG)

  ## Add see to reproduce the same graph
```

```

set.seed(12)

library(ggplot2)
library(ggtangle)
library(scatterpie)
library(ggrepel)

emapGraph <- ggplot(emapG, layout=layout_with_fr) +
  geom_edge(color="gray", linewidth=1)

pieInfo <- as.data.frame(do.call(rbind, V(emapG)$pie))
colnames(pieInfo) <- V(emapG)$pieName[[1]]

## Add information about the groups associated with each node in the
## ggplot object so that the node can be colored accordingly
for (i in seq_len(ncol(pieInfo))) {
  emapGraph$data[colnames(pieInfo)[i]] <- pieInfo[, i]
}

## Using scatterpie, ggtangle and ggrepel to generate the graph
## geom_scatterpie() allows to have scatter pie plot
## geom_text_repel() allows to have minimum overlying terms
## coord_fixed() forces the plot to have a 1:1 aspect ratio
emapGraph +
  geom_scatterpie(aes(x=x, y=y, r=size/50),
    cols=c(colnames(pieInfo)), legend_name = "Group", color=NA) +
  geom_scatterpie_legend(radius=emapGraph$data$size/50, n=4,
    x=max(emapGraph$data$x), y=min(emapGraph$data$y),
    labeller=function(x) {round(x*50)}, label_position="right") +
  geom_text_repel(aes(x=x, y=y, label=label), max.overlaps=20) +
  coord_fixed()
}

```

createEnrichMapMultiComplex

Using functional enrichment results in gprofiler2 format to create an enrichment map with multiple groups from same or different enrichment analyses

Description

User selected enrichment terms are used to create an enrichment map. The selection of the term can be specifying by the source of the terms (GO:MF, REAC, TF, etc.) or by listing the selected term IDs. The map is only generated when there is at least on significant term to graph.

Usage

```

createEnrichMapMultiComplex(
  gostObjectList,

```

```

queryInfo,
showCategory = 30L,
categoryLabel = 1,
categoryNode = 1,
line = 1,
...
)

```

Arguments

- gostObjectList** a list of `gprofiler2` objects that contain the results from an enrichment analysis. The list must contain at least 2 entries. The number of entries must correspond to the number of entries for the `queryList` parameter.
- queryInfo** a data.frame contains one row per group being displayed. The number of rows must correspond to the number of entries for the `gostObjectList` parameter. The mandatory columns are:
- **queryName**: a character string representing the name of the query retained for this group). The query names must exist in the associated `gostObjectList` objects and follow the same order.
 - **source**: a character string representing the selected source that will be used to generate the network. To hand-pick the terms to be used, "TERM_ID" should be used and the list of selected term IDs should be passed through the `termIDs` parameter. The possible sources are "GO:BP" for Gene Ontology Biological Process, "GO:CC" for Gene Ontology Cellular Component, "GO:MF" for Gene Ontology Molecular Function, "KEGG" for Kegg, "REAC" for Reactome, "TF" for TRANSFAC, "MIRNA" for miRTarBase, "CORUM" for CORUM database, "HP" for Human phenotype ontology and "WP" for WikiPathways. Default: "TERM_ID".
 - **removeRoot**: a logical that specified if the root terms of the selected source should be removed (when present).
 - **termIDs**: a character strings that contains the term IDS retained for the creation of the network separated by a comma ',' when the "TERM_ID" source is selected. Otherwise, it should be a empty string ("").
 - **groupName**: a character strings that contains the name of the group to be shown in the legend. Each group has to have a unique name.
- showCategory** a positive integer or a vector of characters representing terms. If a integer, the first n terms will be displayed. If vector of terms, the selected terms will be displayed. Default: 30L.
- categoryLabel** a positive numeric representing the amount by which plotting category nodes label size should be scaled relative to the default (1). Default: 1.
- categoryNode** a positive numeric representing the amount by which plotting category nodes should be scaled relative to the default (1). Default: 1.
- line** a non-negative numeric representing the scale of line width. Default: 1.
- ... additional arguments that will be pass to the `emapplot` function.

Value

a ggplot object which is the enrichment map for enrichment results.

Author(s)

Astrid Deschênes

Examples

```
## Loading dataset containing results from 2 enrichment analyses done with
## gprofiler2
data(parentalNapaVsDMSOEnrichment)
data(rosaNapaVsDMSOEnrichment)

## Create list of enrichment objects required to generate the enrichment map
gostObjectList=list(parentalNapaVsDMSOEnrichment,
  parentalNapaVsDMSOEnrichment, rosaNapaVsDMSOEnrichment,
  rosaNapaVsDMSOEnrichment)

## Create data frame containing required information enabling the
## selection of the retained enriched terms for each enrichment analysis.
## One line per enrichment analyses present in the gostObjectList parameter
## With this data frame, the enrichment results will be split in 4 groups:
## 1) KEGG significant terms from parental napa vs DMSO (no root term)
## 2) REACTOME significant terms from parental napa vs DMSO (no root term)
## 3) KEGG significant terms from rosa napa vs DMSO (no root term)
## 4) REACTOME significant terms from rosa napa vs DMSO (no root term)
queryDataFrame <- data.frame(queryName=c("parental_napa_vs_DMSO",
  "parental_napa_vs_DMSO", "rosa_napa_vs_DMSO", "rosa_napa_vs_DMSO"),
  source=c("KEGG", "REAC", "KEGG", "REAC"),
  removeRoot=c(TRUE, TRUE, TRUE, TRUE), termIDs=c("", "", "", ""),
  groupName=c("parental - KEGG", "parental - Reactome",
  "rosa - KEGG", "rosa - Reactome"), stringsAsFactors=FALSE)

## Create graph for KEGG and REACTOME significant results from
## 2 enrichment analyses
createEnrichMapMultiComplex(gostObjectList=gostObjectList,
  queryInfo=queryDataFrame, line=1.5)
```

createEnrichMapMultiComplexAsIgraph

Using functional enrichment results in gprofiler2 format to create an enrichment map with multiple groups from same or different enrichment analyses in an igraph format

Description

User selected enrichment terms are used to create an enrichment map. The selection of the term can be by specifying by the source of the terms (GO:MF, REAC, TF, etc.) or by listing the selected term IDs. The map is only generated when there is at least one significant term to graph.

Usage

```
createEnrichMapMultiComplexAsIgraph(
  gostObjectList,
  queryInfo,
  showCategory = 30L,
  similarityCutOff = 0.2
)
```

Arguments

- | | |
|-------------------------------|---|
| <code>gostObjectList</code> | a list of <code>gprofiler2</code> objects that contain the results from an enrichment analysis. The list must contain at least 2 entries. The number of entries must correspond to the number of entries for the <code>queryList</code> parameter. |
| <code>queryInfo</code> | a <code>data.frame</code> contains one row per group being displayed. The number of rows must correspond to the number of entries for the <code>gostObjectList</code> parameter. The mandatory columns are: <ul style="list-style-type: none"> • <code>queryName</code>: a character string representing the name of the query retained for this group). The query names must exist in the associated <code>gostObjectList</code> objects and follow the same order. • <code>source</code>: a character string representing the selected source that will be used to generate the network. To hand-pick the terms to be used, "TERM_ID" should be used and the list of selected term IDs should be passed through the <code>termIDs</code> parameter. The possible sources are "GO:BP" for Gene Ontology Biological Process, "GO:CC" for Gene Ontology Cellular Component, "GO:MF" for Gene Ontology Molecular Function, "KEGG" for Kegg, "REAC" for Reactome, "TF" for TRANSFAC, "MIRNA" for miRTarBase, "CORUM" for CORUM database, "HP" for Human phenotype ontology and "WP" for WikiPathways. Default: "TERM_ID". • <code>removeRoot</code>: a logical that specified if the root terms of the selected source should be removed (when present). • <code>termIDs</code>: a character strings that contains the term IDs retained for the creation of the network separated by a comma ',' when the "TERM_ID" source is selected. Otherwise, it should be an empty string (""). • <code>groupName</code>: a character strings that contains the name of the group to be shown in the legend. Each group has to have a unique name. |
| <code>showCategory</code> | a positive integer or NULL. If a integer, the first n terms will be displayed. If NULL, all terms will be displayed. Default: 30L. |
| <code>similarityCutOff</code> | a positive numeric, larger than zero and small than 1 that represent the minimum similarity level between two nodes (terms) to be linked by an edge. Default: 0.2. |

Value

a igraph object which is the enrichment map for enrichment results. The node have 5 attributes: "name", "size", "pie", "cluster", and "pieName". The "name" corresponds to the term description. While the "size" corresponds to the number of unique genes found in the specific gene set when looking at all the experiments. The edges have 3 attributes: "similarity", "width", and "weight". All those 3 attributes correspond to the Jaccard coefficient.

Author(s)

Astrid Deschênes

Examples

```
## Loading dataset containing results from 2 enrichment analyses done with
## gprofiler2 queries
data(parentalNapaVsDMSOEnrichment)
data(rosaNapaVsDMSOEnrichment)

## The graph will be split in 4 groups
## Groups 1 and 2 are using the parental Napa vs DMSO dataset
## Group 3 and 4 are using the rosa Napa vs DMSO dataset
gostObjectList=list(parentalNapaVsDMSOEnrichment,
                    parentalNapaVsDMSOEnrichment, rosaNapaVsDMSOEnrichment,
                    rosaNapaVsDMSOEnrichment)

## Create data frame containing required information enabling the
## selection of the retained enriched terms for each enrichment analysis.
## One line per enrichment analyses present in the gostObjectList parameter
## With this data frame, the enrichment results will be split in 4 groups:
## 1) KEGG significant terms from parental napa vs DMSO (no root term)
## 2) REACTOME significant terms from parental napa vs DMSO (no root term)
## 3) KEGG significant terms from rosa napa vs DMSO (no root term)
## 4) REACTOME significant terms from rosa napa vs DMSO (no root term)
queryDataFrame <- data.frame(queryName=c("parental_napa_vs_DMSO",
    "parental_napa_vs_DMSO", "rosa_napa_vs_DMSO", "rosa_napa_vs_DMSO"),
    source=c("KEGG", "REAC", "KEGG", "REAC"),
    removeRoot=c(TRUE, TRUE, TRUE, TRUE), termIDs=c("", "", "", "")),
    groupName=c("parental - KEGG", "parental - Reactome",
    "rosa - KEGG", "rosa - Reactome"), stringsAsFactors=FALSE)

## Create graph for KEGG and REACTOME significant results from
## 2 enrichment analyses in an igraph format
emap <- createEnrichMapMultiComplexAsIgraph(gostObjectList=gostObjectList,
    queryInfo=queryDataFrame, showCategory=5)

if (requireNamespace("ggplot2", quietly=TRUE) &&
    requireNamespace("igraph", quietly=TRUE) &&
    requireNamespace("scatterpie", quietly=TRUE) &&
    requireNamespace("ggtangle", quietly=TRUE) &&
    requireNamespace("ggrepel", quietly=TRUE)) {
  ## Create a visual representation of the enrichment map
  ## by default
```

```

library(igraph)
plot(emap)

## Add see to reproduce the same graph
set.seed(12)

library(ggplot2)
library(ggtangle)
library(scatterpie)
library(ggrepel)

emapG <- ggplot(emap, layout=layout_with_fr) +
  geom_edge(color="gray", linewidth=1)

pieInfo <- as.data.frame(do.call(rbind, V(emap)$pie))
colnames(pieInfo) <- V(emap)$pieName[[1]]

## Add information about the groups associated with each node in the
## ggplot object so that the node can be colored accordingly
for (i in seq_len(ncol(pieInfo))) {
  emapG$data[colnames(pieInfo)[i]] <- pieInfo[, i]
}

## Using scatterpie, ggtangle and ggrepel to generate the graph
## geom_scatterpie() allows to have scatter pie plot
## geom_text_repel() allows to have minimum overlying terms
## coord_fixed() forces the plot to have a 1:1 aspect ratio
emapG +
  geom_scatterpie(aes(x=x, y=y, r=size/50),
    cols=c(colnames(pieInfo)), legend_name = "Group", color=NA) +
  geom_scatterpie_legend(radius=emapG$data$size/50, n=4,
    x=max(emapG$data$x), y=max(emapG$data$y),
    labeller=function(x) {round(x*50)}, label_position="right") +
  geom_text_repel(aes(x=x, y=y, label=label), max.overlaps=20) +
  coord_fixed()
}

```

createNetwork

Using functional enrichment results from gprofiler2 to create a Cytoscape network

Description

User selected enrichment terms are used to create a Cytoscape network where the selected terms and the genes that were part of the enrichment analysis are all represented as nodes. Edges are linking the genes to their terms. The selection of the term can be by specifying the source of the terms (GO:MF, REAC, TF, etc.) or by listing the selected term IDs. The network is only generated when there is at least one significant term to graph. When the enrichment analysis contains more than one query, only one query can be selected to generate the network.

Usage

```

createNetwork(
  gostObject,
  source = c("TERM_ID", "GO:MF", "GO:CC", "GO:BP", "KEGG", "REAC", "TF", "MIRNA", "HPA",
    "CORUM", "HP", "WP"),
  termIDs = NULL,
  removeRoot = TRUE,
  query = NULL,
  title = "gprofiler network",
  collection = "enrichment results",
  fileName = "gprofilerNetwork.cx"
)

```

Arguments

<code>gostObject</code>	a list created by gprofiler2 that contains the results from an enrichment analysis.
<code>source</code>	a character string representing the selected source that will be used to generate the network. To hand-pick the terms to be used, "TERM_ID" should be used and the list of selected term IDs should be passed through the termIDs parameter. The possible sources are "GO:BP" for Gene Ontology Biological Process, "GO:CC" for Gene Ontology Cellular Component, "GO:MF" for Gene Ontology Molecular Function, "KEGG" for Kegg, "REAC" for Reactome, "TF" for TRANSFAC, "MIRNA" for miRTarBase, "CORUM" for CORUM database, "HP" for Human phenotype ontology and "WP" for WikiPathways. Default: "TERM_ID".
<code>termIDs</code>	a vector of character strings that contains the term IDs retained for the creation of the network. Default: NULL.
<code>removeRoot</code>	a logical that specified if the root terms of the selected source should be removed (when present). Default: TRUE.
<code>query</code>	a character string that specified the retained query to generate the network. When NULL, the query present in the result is retained; NULL cannot be used when more than one query is present. Default: NULL.
<code>title</code>	a character string representing the name assigned to the network. Default: "gprofiler network".
<code>collection</code>	a character string representing the collection name assigned to the network. Default: "enrichment results".
<code>fileName</code>	a character string representing the name of the CX JSON file that is created when Cytoscape is not running. The name must have a '.cx' extension. Default: "gprofilerNetwork_01.cx".

Value

TRUE

Author(s)

Astrid Deschênes

Examples

```
## Loading dataset containing result from an enrichment analysis done with
## gprofiler2
data(parentalNapaVsDMSOEnrichment)

## Some of the enrichment results present in the dataset
head(parentalNapaVsDMSOEnrichment$result)

## Create network for Gene Ontology - Molecular Function related results
## in Cytoscape (when the application is opened)
## Otherwise, create a CX file in the temporary directory
## The file can be opened in Cytoscape
createNetwork(gostObject=parentalNapaVsDMSOEnrichment, source="KEGG",
              removeRoot=FALSE, title="KEGG Graph",
              fileName=file.path(tempdir(), "KEGG_demo.cx"))
```

demoGOST

The result of a functional enrichment analysis done with gprofiler2
(<https://cran.r-project.org/web/packages/gprofiler2/vignettes/gprofiler2.html>).

Description

The object is a list with 2 entries. It contains the results of the enrichment analysis as well as the metadata related to the analysis.

Usage

```
data(demoGOST)
```

Format

The list contains two entries. The result entry contains a data.frame with the significant results obtained by an enrichment analysis done with gprofiler2. The meta entry contains a named list with all the metadata for the query.

Details

This dataset can be used to test the `createNetwork` function.

Value

A list containing two entries. The result entry contains a `data.frame` with the significant results obtained by an enrichment analysis done with `gprofiler2`. The meta entry contains a named list with all the metadata for the query.

See Also

- [createNetwork](#) for transforming functional enrichment results from `gprofiler2` into a Cytoscape network
- [createEnrichMap](#) for transforming functional enrichment results from `gprofiler2` into an enrichment map

Examples

```
## Loading dataset containing result from an enrichment analysis done with
## gprofiler2
data(demoGOST)

## Create network for WikiPathways results
## in Cytoscape (if the application is open)
## Otherwise, create a CX file in the temporary directory
## The file can be opened in Cytoscape
createNetwork(gostObject=demoGOST, source="WP", title="Wikipathways",
              fileName=file.path(tempdir(), "Wikipathways_Demo.cx"))
```

parentalNapaVsDMSODEG *The result of a differential expression analysis done between napabucasin treated and DMSO control parental MiaPaCa2 cells. The cells were treated for 2 hours with 0.5 uM napabucasin. The protocol to generate the RNA-seq is described in Froeling F.E.M. et al 2019.*

Description

The object is a `data.frame` with 24184 rows and 4 columns. Each row correspond to a tested gene.

Usage

```
data(parentalNapaVsDMSODEG)
```

Format

a `data.frame` containing the results of a differential expression analysis between napabucasin treated and DMSO control parental MiaPaCa2 cells for all 24184 genes tested. The 4 columns are:

- "EnsemblID": a character string representing the unique Ensembl identifier for the tested gene
- "log2FoldChange": a numeric representing the expression difference (in log2FoldChange) between the napabucasin treatment and the DMSO control for the tested gene
- "padj": a numeric representing the adjusted p-value associated to the difference in expression for the tested gene
- "GeneName": a character string representing the name of the tested gene

Details

The differentially expressed genes between napabucasin-treated cells (0.5 μ M) and DMSO as vehicle control are reprinted from Clinical Cancer Research, 2019, 25 (23), 7162–7174, Fieke E.M. Froeling, Manojit Mosur Swamynathan, Astrid Deschênes, Iok In Christine Chio, Erin Brosnan, Melissa A. Yao, Priya Alagesan, Matthew Lucito, Juying Li, An-Yun Chang, Lloyd C. Trotman, Pascal Belleau, Youngkyu Park, Harry A. Rogoff, James D. Watson, David A. Tuveson, Bioactivation of napabucasin triggers reactive oxygen species-mediated cancer cell death, with permission from AACR.

Value

a `data.frame` containing the results of a differential expression analysis between napabucasin treated and DMSO control parental MiaPaCa2 cells for all 24184 genes tested. The 4 columns are:

- "EnsemblID": a character string representing the unique Ensembl identifier for the tested gene
- "log2FoldChange": a numeric representing the expression difference (in log2FoldChange) between the napabucasin treatment and the DMSO control for the tested gene
- "padj": a numeric representing the adjusted p-value associated to the difference in expression for the tested gene
- "GeneName": a character string representing the name of the tested gene

Source

The original RNA-sequencing data is available at the Gene Expression Omnibus (GEO) under the accession number GSE135352.

See Also

- [createNetwork](#) for transforming functional enrichment results from gprofiler2 into a Cytoscape network
- [createEnrichMap](#) for transforming functional enrichment results from gprofiler2 into an enrichment map

Examples

```
## Required library
library(gprofiler2)

## Loading data set containing the results of a differentially expressed
## analysis between 2-hour treatment with 0.5 uM napabucasin and
## DMSO vehicle control parental MiaPaCa2 cells
data(parentalNapaVsDMSODEG)

allGenes <- unique(parentalNapaVsDMSODEG$EnsemblID)

## Select the significantly differentially expressed genes
selection <- which(abs(parentalNapaVsDMSODEG$log2FoldChange) > 1 &
  parentalNapaVsDMSODEG$padj < 0.05)

selectedGenes <- unique(parentalNapaVsDMSODEG$EnsemblID[selection])

## Run an enrichment analysis using WikiPathways dataset
gostres <- gost(query = list(parental_napa_vs_DMSO=selectedGenes),
  organism="hsapiens",
  correction_method = "g_SCS",
  sources=c("WP"), significant=TRUE, evcodes=TRUE,
  custom_bg=allGenes, exclude_iea=TRUE)
```

```
parentalNapaVsDMSOEnrichment
```

The result of an enrichment analysis has been done using the significantly differentially expressed genes between napabucasin treated and DMSO control parental MiaPaCa2 cells. The cells were treated for 2 hour with 0.5 uM napabucasin. The protocol to generate the RNA-seq is described in Froeling F.E.M. et al 2019.

Description

The enrichment analysis was done with gprofile2 package (Kolberg L et al 2020) with database version 'e109_eg56_p17_1d3191d' and g:SCS multiple testing correction method applying significance threshold of 0.05 (Raudvere U et al 2019). All tested genes were used as background.

Usage

```
data(parentalNapaVsDMSOEnrichment)
```

Format

a list created by gprofiler2 that contains the results from the enrichment analysis:

- "result": a data.frame with the significantly enriched terms
- "meta": a list with the meta-data information

Details

The object is a named list with 2 entries. The 'result' entry contains a data.frame with the enrichment analysis results and the 'meta' entry contains metadata information.

The dataset used for the enrichment analysis is associated to this publication:

Froeling F.E.M. et al. Bioactivation of Napabucasin Triggers Reactive Oxygen Species-Mediated Cancer Cell Death. Clin Cancer Res 1 December 2019; 25 (23): 7162–7174

The enrichment analysis has been done with gprofile2 package (Kolberg L et al 2020) with database version 'e109_eg56_p17_1d3191d' and g:SCS multiple testing correction method applying significance threshold of 0.05 (Raudvere U et al 2019). All tested genes were used as background.

Value

a list containing 2 entries:

- "result": a data.frame with the significantly enriched terms
- "meta": a list with the meta-data information

Source

The original RNA-sequencing data is available at the Gene Expression Omnibus (GEO) under the accession number GSE135352.

See Also

- [createNetwork](#) for transforming functional enrichment results from gprofiler2 into a Cytoscape network
- [createEnrichMap](#) for transforming functional enrichment results from gprofiler2 into an enrichment map
- [createEnrichMapAsIgraph](#) for transforming functional enrichment results from gprofiler2 into an enrichment map in a igraph format

Examples

```
## Loading dataset containing the results of the enrichment analysis
## done on a differentially expressed
## analysis between 2-hour treatment with 0.5 uM napabucasin and
## DMSO vehicle control parental MiaPaCa2 cells
data(parentalNapaVsDMSOEnrichment)

## Create an enrichment map for the GO:CC terms
createEnrichMap(gostObject=parentalNapaVsDMSOEnrichment,
  query="parental_napa_vs_DMSO", source="GO:CC")
```

rosaNapaVsDMSODEG	<i>The result of a differential expression analysis done between napabucasin treated and DMSO control MiaPaCa2 cells stably expressing the Rosa26 control vector. The cells were treated for 2 hours with 0.5 uM napabucasin. The protocol to generate the RNA-seq is described in Froeling F.E.M. et al 2019.</i>
-------------------	--

Description

The object is a data.frame with 23542 rows and 4 columns. Each row correspond to a tested gene.

Usage

```
data(rosaNapaVsDMSODEG)
```

Format

a data.frame containing the results of a differential expression analysis between napabucasin treated and DMSO control MiaPaCa2 cells stably expressing the Rosa26 control vector for all 23542 genes tested. The 4 columns are:

- "EnsemblID": a character string representing the unique Ensembl identifier for the tested gene
- "log2FoldChange": a numeric representing the expression difference (in log2FoldChange) between the napabucasin treatment and the DMSO control for the tested gene
- "padj": a numeric representing the adjusted p-value associated to the difference in expression for the tested gene
- "GeneName": a character string representing the name of the tested gene

Details

The differentially expressed genes between napabucasin-treated cells (0.5 uM) and DMSO as vehicle control are reprinted from Clinical Cancer Research, 2019, 25 (23), 7162–7174, Fieke E.M. Froeling, Manojit Mosur Swamynathan, Astrid Deschênes, Iok In Christine Chio, Erin Brosnan, Melissa A. Yao, Priya Alagesan, Matthew Lucito, Juying Li, An-Yun Chang, Lloyd C. Trotman, Pascal Belleau, Youngkyu Park, Harry A. Rogoff, James D. Watson, David A. Tuveson, Bioactivation of napabucasin triggers reactive oxygen species-mediated cancer cell death, with permission from AACR.

Value

a data.frame containing the results of a differential expression analysis between napabucasin treated and DMSO control MiaPaCa2 cells stably expressing the Rosa26 control vector for all 23542 genes tested. The 4 columns are:

- "EnsemblID": a character string representing the unique Ensembl identifier for the tested gene

- "log2FoldChange": a numeric representing the expression difference (in log2FoldChange) between the napabucasin treatment and the DMSO control for the tested gene
- "padj": a numeric representing the adjusted p-value associated to the difference in expression for the tested gene
- "GeneName": a character string representing the name of the tested gene

Source

The original RNA-sequencing data is available at the Gene Expression Omnibus (GEO) under the accession number GSE135352.

See Also

- [createNetwork](#) for transforming functional enrichment results from gprofiler2 into a Cytoscape network
- [createEnrichMap](#) for transforming functional enrichment results from gprofiler2 into an enrichment map

Examples

```
## Required library
library(gprofiler2)

## Loading dataset containing the results of a differentially expressed
## analysis between 2-hour treatment with 0.5 uM napabucasin and
## DMSO vehicle control MiaPaCa2 cells stably expressing the
## Rosa26 control vector
data(rosaNapaVsDMSODEG)

allGenes <- unique(rosaNapaVsDMSODEG$EnsemblID)

## Select the significantly differentially expressed genes
selection <- which(abs(rosaNapaVsDMSODEG$log2FoldChange) > 1 &
                  rosaNapaVsDMSODEG$padj < 0.05)

selectedGenes <- unique(rosaNapaVsDMSODEG$EnsemblID[selection])

## Run an enrichment analysis using Transfac dataset (transcription factor)
gostres <- gost(query = list(rosa_napa_vs_DMSO=selectedGenes),
               organism="hsapiens",
               correction_method = "g_SCS",
               sources=c("TF"), significant=TRUE, evcodes=TRUE,
               custom_bg=allGenes, exclude_iea=TRUE)
```

rosaNapaVsDMSOEnrichment

The result of an enrichment analysis that has been done using the significantly differentially expressed genes between napabucasin treated and DMSO control MiaPaCa2 cells stably expressing the Rosa26 control vector. The cells were treated for 2 hour with 0.5 uM napabucasin. The protocol to generate the RNA-seq is described in Froeling F.E.M. et al 2019.

Description

The enrichment analysis was done with gprofile2 package (Kolberg L et al 2020) with database version 'e109_eg56_p17_1d3191d' and g:SCS multiple testing correction method applying significance threshold of 0.05 (Raudvere U et al 2019). All tested genes were used as background.

Usage

```
data(rosaNapaVsDMSOEnrichment)
```

Format

a list created by gprofiler2 that contains the results from the enrichment analysis:

- "result": a data.frame with the significantly enriched terms
- "meta": a list with the meta-data information

Details

The object is a named list with 2 entries. The 'result' entry contains a data.frame with the enrichment analysis results and the 'meta' entry contains metadata information.

The dataset used for the enrichment analysis is associated to this publication:

Froeling F.E.M. et al. Bioactivation of Napabucasin Triggers Reactive Oxygen Species-Mediated Cancer Cell Death. Clin Cancer Res 1 December 2019; 25 (23): 7162–7174

The enrichment analysis has been done with gprofile2 package (Kolberg L et al 2020) with database version 'e109_eg56_p17_1d3191d' and g:SCS multiple testing correction method applying significance threshold of 0.05 (Raudvere U et al 2019). All tested genes were used as background.

Value

a list created by gprofiler2 that contains the results from the enrichment analysis:

- "result": a data.frame with the significantly enriched terms
- "meta": a list with the meta-data information

Source

The original RNA-sequencing data is available at the Gene Expression Omnibus (GEO) under the accession number GSE135352.

See Also

- [createNetwork](#) for transforming functional enrichment results from gprofiler2 into a Cytoscape network
- [createEnrichMap](#) for transforming functional enrichment results from gprofiler2 into an enrichment map

Examples

```
## Loading dataset containing the enrichment analysis done on the
## differentially expressed analysis between 2-hour treatment with 0.5 uM
## napabucasin and DMSO control MiaPaCa2 cells stably expressing
## the Rosa26 control vector
data(rosaNapaVsDMSOEnrichment)

## Create an enrichment map for the KEGG terms
createEnrichMap(gostObject=rosaNapaVsDMSOEnrichment,
  query="rosa_napa_vs_DMSO", source="KEGG")
```

```
validateCreateEnrichMapMultiComplexAsIgraphArg
```

Validate arguments passed to validateCreateEnrichMapMultiComplex() function

Description

Validate the arguments passed to validateCreateEnrichMapMultiComplex() function. First, the object containing the enrichment results must correspond to an object created by gprofiler2 software. Second, the selected source must at least have one enriched term in the results. Then, if the source is 'TERM_ID', the listed terms must be present in the enrichment results.

Usage

```
validateCreateEnrichMapMultiComplexAsIgraphArg(
  gostObjectList,
  queryInfo,
  showCategory,
  similarityCutOff = similarityCutOff
)
```

Arguments

gostObjectList a list of gprofiler2 objects that contain the results from an enrichment analysis. The list must contain at least 2 entries. The number of entries must correspond to the number of entries for the queryList parameter.

queryInfo a data.frame contains one row per group being displayed. The number of rows must correspond to the number of entries for the gostObjectList parameter. The mandatory columns are:

- queryName: a character string representing the name of the query retained for this group). The query names must exist in the associated gostObjectList objects and follow the same order.
 - source: a character string representing the selected source that will be used to generate the network. To hand-pick the terms to be used, "TERM_ID" should be used and the list of selected term IDs should be passed through the termIDs parameter. The possible sources are "GO:BP" for Gene Ontology Biological Process, "GO:CC" for Gene Ontology Cellular Component, "GO:MF" for Gene Ontology Molecular Function, "KEGG" for Kegg, "REAC" for Reactome, "TF" for TRANSFAC, "MIRNA" for miRTarBase, "CORUM" for CORUM database, "HP" for Human phenotype ontology and "WP" for WikiPathways. Default: "TERM_ID".
 - removeRoot: a logical that specified if the root terms of the selected source should be removed (when present).
 - termIDs: a character strings that contains the term IDS retained for the creation of the network separated by a comma ',' when the "TERM_ID" source is selected. Otherwise, it should be a empty string ("").
 - groupName: a character strings that contains the name of the group to be shown in the legend. Each group has to have a unique name.
- showCategory a positive integer or NULL. If a integer, the first n terms will be displayed. If NULL, all terms will be displayed.
- similarityCutOff a positive numeric, larger than zero and small than 1 that represent the minimum similarity level between two nodes (terms) to be linked by an edge.

Value

TRUE when all arguments are valid

Author(s)

Astrid Deschênes

Examples

```
## Load the result of an enrichment analysis done with gprofiler2
data(parentalNapaVsDMSOEnrichment)
data(rosaNapaVsDMSOEnrichment)

queryDataFrame <- data.frame(queryName=c("parental_napa_vs_DMSO",
  "rosa_napa_vs_DMSO"), source=c("KEGG", "WP"), removeRoot=c(TRUE, TRUE),
  termIDs=c("", ""), groupName=c("parental - KEGG", "rosa - WP"),
  stringsAsFactors=FALSE)

## Check that all arguments are valid
enrichViewNet:::validateCreateEnrichMapMultiComplexAsIgraphArg(
  gostObjectList=list(parentalNapaVsDMSOEnrichment,
    rosaNapaVsDMSOEnrichment),
  queryInfo=queryDataFrame, showCategory=20, similarityCutOff=0.2)
```

Index

* datasets

- demoGOST, [18](#)
- parentalNapaVsDMSODEG, [19](#)
- parentalNapaVsDMSOEnrichment, [21](#)
- rosaNapaVsDMSODEG, [23](#)
- rosaNapaVsDMSOEnrichment, [25](#)

* package

- enrichViewNet-package, [2](#)

- createEnrichMap, [3](#), [3](#), [19](#), [20](#), [22](#), [24](#), [26](#)
- createEnrichMapAsIgraph, [3](#), [5](#), [22](#)
- createEnrichMapMultiBasic, [7](#)
- createEnrichMapMultiBasicAsIgraph, [9](#)
- createEnrichMapMultiComplex, [3](#), [11](#)
- createEnrichMapMultiComplexAsIgraph, [3](#),
[13](#)
- createNetwork, [3](#), [16](#), [18–20](#), [22](#), [24](#), [26](#)

- demoGOST, [18](#)

- emapplot, [4](#), [8](#), [12](#)
- enrichViewNet (enrichViewNet-package), [2](#)
- enrichViewNet-package, [2](#)

- parentalNapaVsDMSODEG, [19](#)
- parentalNapaVsDMSOEnrichment, [21](#)

- rosaNapaVsDMSODEG, [23](#)
- rosaNapaVsDMSOEnrichment, [25](#)

- validateCreateEnrichMapMultiComplexAsIgraphArg,
[26](#)