Package ‘enrichplot’

May 29, 2024

Title Visualization of Functional Enrichment Result

Version 1.24.0

Description The ‘enrichplot’ package implements several visualization methods for interpreting functional enrichment results obtained from ORA or GSEA analysis. It is mainly designed to work with the ‘clusterProfiler’ package suite. All the visualization methods are developed based on ‘ggplot2’ graphics.

Depends R (>= 3.5.0)

Imports aplot (>= 0.2.1), DOSE (>= 3.16.0), ggfun (>= 0.1.3), ggnewscale, ggplot2, ggraph, graphics, grid, igraph, methods, plyr, purrr, RColorBrewer, reshape2, rlang, stats, utils, scatterpie, shadowtext, GOSemSim, magrittr, ggtree, yulab.utils (>= 0.0.8)

Suggests clusterProfiler, dplyr, europepmc, ggupset, knitr, rmarkdown, org.Hs.eg.db, prettydoc, tibble, tidyr, ggforce, AnnotationDbi, ggplotify, ggridges, grDevices, gridExtra, ggrepel (>= 0.9.0), ggstar, scales, ggtreeExtra, tidydr

Remotes YuLab-SMU/tidydr

VignetteBuilder knitr

License Artistic-2.0


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

biocViews Annotation, GeneSetEnrichment, GO, KEGG, Pathways, Software, Visualization

Encoding UTF-8

RoxygenNote 7.3.1

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

git_branch RELEASE_3_19

git_last_commit c3b957c

git_last_commit_date 2024-04-30

Repository Bioconductor 3.19
Date/Publication  2024-05-29

Author   Guangchuang Yu [aut, cre] (<https://orcid.org/0000-0002-6485-8781>),  
         Erqiang Hu [ctb] (<https://orcid.org/0000-0002-1798-7513>),  
         Chun-Hui Gao [ctb] (<https://orcid.org/0000-0002-1445-7939>)

Maintainer  Guangchuang Yu <guangchuangyu@gmail.com>

Contents

<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>enrichplot-package</td>
<td>2</td>
</tr>
<tr>
<td>autofacet</td>
<td>3</td>
</tr>
<tr>
<td>barplot.enrichResult</td>
<td>4</td>
</tr>
<tr>
<td>cnetplot</td>
<td>5</td>
</tr>
<tr>
<td>color_palette</td>
<td>8</td>
</tr>
<tr>
<td>dotplot</td>
<td>8</td>
</tr>
<tr>
<td>drag_network</td>
<td>12</td>
</tr>
<tr>
<td>emapplot</td>
<td>13</td>
</tr>
<tr>
<td>emapplot_cluster</td>
<td>18</td>
</tr>
<tr>
<td>fortify.compareClusterResult</td>
<td>19</td>
</tr>
<tr>
<td>geom_gsea_gene</td>
<td>20</td>
</tr>
<tr>
<td>ggtable</td>
<td>21</td>
</tr>
<tr>
<td>goplot</td>
<td>21</td>
</tr>
<tr>
<td>gseadist</td>
<td>23</td>
</tr>
<tr>
<td>gseaplot</td>
<td>24</td>
</tr>
<tr>
<td>gseaplot2</td>
<td>25</td>
</tr>
<tr>
<td>gsearank</td>
<td>26</td>
</tr>
<tr>
<td>gsInfo</td>
<td>27</td>
</tr>
<tr>
<td>heatmap</td>
<td>27</td>
</tr>
<tr>
<td>pairwise_termsim</td>
<td>28</td>
</tr>
<tr>
<td>plotting.clusterProfile</td>
<td>30</td>
</tr>
<tr>
<td>pmcplot</td>
<td>31</td>
</tr>
<tr>
<td>reexports</td>
<td>31</td>
</tr>
<tr>
<td>ridgeplot</td>
<td>32</td>
</tr>
<tr>
<td>set_enrichplot_color</td>
<td>33</td>
</tr>
<tr>
<td>ssplot</td>
<td>34</td>
</tr>
<tr>
<td>treeplot</td>
<td>36</td>
</tr>
<tr>
<td>upsetplot</td>
<td>41</td>
</tr>
</tbody>
</table>

Index  43

---

enrichplot-package  enrichplot: Visualization of Functional Enrichment Result
autofacet

Description

The `enrichplot` package implements several visualization methods for interpreting functional enrichment results obtained from ORA or GSEA analysis. It is mainly designed to work with the `clusterProfiler` package suite. All the visualization methods are developed based on `ggplot2` graphics.

Author(s)

Maintainer: Guangchuang Yu <guangchuangyu@gmail.com> (ORCID)

Other contributors:

- Erqiang Hu <13766876214@163.com> (ORCID) [contributor]
- Chun-Hui Gao <gaospecial@gmail.com> (ORCID) [contributor]

See Also

Useful links:

- Report bugs at https://github.com/GuangchuangYu/enrichplot/issues

autofacet  automatically split barplot or dotplot into several facets

Description

automatically split barplot or dotplot into several facets

Usage

autofacet(by = "row", scales = "free", levels = NULL)

Arguments

by  one of 'row' or 'column'

scales  wether 'fixed' or 'free'

levels  set facet levels

Value

a ggplot object
Description

barplot of enrichResult

Usage

```r
## S3 method for class 'enrichResult'
barplot(
  height,
  x = "Count",
  color = "p.adjust",
  showCategory = 8,
  font.size = 12,
  title = "",
  label_format = 30,
  ...
)
```

Arguments

- **height**: enrichResult object
- **x**: one of 'Count' and 'GeneRatio'
- **color**: one of 'pvalue', 'p.adjust' and 'qvalue'
- **showCategory**: number of categories to show
- **font.size**: font size
- **title**: plot title
- **label_format**: a numeric value sets wrap length, alternatively a custom function to format axis labels. by default wraps names longer that 30 characters
- **...**: other parameter, ignored

Value

ggplot object

Examples

```r
library(DOSE)
data(geneList)
de <- names(geneList)[1:100]
x <- enrichDO(de)
barplot(x)
# use `showCategory` to select the displayed terms. It can be a number of a vector of terms.
barplot(x, showCategory = 10)
```
categorys <- c("pre-malignant neoplasm", "intestinal disease", "breast ductal carcinoma", "non-small cell lung carcinoma")
barplot(x, showCategory = categorys)

cnetplot

description
Gene-Concept Network

Usage

cnetplot(x, ...)

## S4 method for signature 'enrichResult'
cnetplot(x, ...)

## S4 method for signature 'list'
cnetplot(x, ...)

## S4 method for signature 'gseaResult'
cnetplot(x, ...)

## S4 method for signature 'compareClusterResult'
cnetplot(x, ...)

cnetplot.enrichResult(
  x,
  showCategory = 5,
  foldChange = NULL,
  layout = "kk",
  colorEdge = FALSE,
  circular = FALSE,
  node_label = "all",
  cex_category = 1,
  cex_gene = 1,
  cex_label_category = 1,
  cex_label_gene = 1,
  color_category = "#E5C494",
  color_gene = "#B3B3B3",
  shadowtext = "all",
  color.params = list(foldChange = NULL, edge = FALSE, category = "#E5C494", gene = "#B3B3B3"),
  cex.params = list(category_node = 1, gene_node = 1, category_label = 1, gene_label = 1),
  hilight.params = list(category = NULL, alpha_hilight = 1, alpha_no_hilight = 0.3),
  ...
)
Arguments

x  Enrichment result.

...  Additional parameters

showCategory  A number or a vector of terms. If it is a number, the first n terms will be displayed. If it is a vector of terms, the selected terms will be displayed.

foldChange  Fold Change of nodes, the default value is NULL. If the user provides the Fold Change value of the nodes, it can be used to set the color of the gene node. Will be removed in the next version.

layout  Layout of the map, e.g. 'star', 'circle', 'gem', 'dh', 'graphopt', 'grid', 'mds', 'randomly', 'fr', 'kk', 'drl' or 'lgl'.

colorEdge  Logical, whether coloring edge by enriched terms, the default value is FALSE. Will be removed in the next version.

circular  Logical, whether using circular layout, the default value is FALSE. Will be removed in the next version.

node_label  Select which labels to be displayed. one of 'category', 'gene', 'all'(the default) and 'none'.

cex_category  Number indicating the amount by which plotting category nodes should be scaled relative to the default, the default value is 1. Will be removed in the next version.

cex_gene  Number indicating the amount by which plotting gene nodes should be scaled relative to the default, the default value is 1. Will be removed in the next version.

cex_label_category  Scale of category node label size, the default value is 1. Will be removed in the next version.

cex_label_gene  Scale of gene node label size, the default value is 1. Will be removed in the next version.

color_category  Color of category node. Will be removed in the next version.

color_gene  Color of gene node. Will be removed in the next version.

shadowtext  select which node labels to use shadow font, one of 'category', 'gene', 'all' and 'none', default is 'all'.

color.params  list, the parameters to control the attributes of highlighted nodes and edges. see the color.params in the following. color.params control the attributes of highlight, it can be referred to the following parameters:

  • foldChange Fold Change of nodes for enrichResult, or size of nodes for compareClusterResult, the default value is NULL.
  • edge Logical, whether coloring edge by enriched terms, the default value is FALSE.
  • category Color of category node.
  • gene Color of gene node.

cex.params  list, the parameters to control the size of nodes and lables. see the cex.params in the following. cex.params control the attributes of highlight, it can be referred to the following parameters:
cnetplot

- foldChange only used in compareClusterResult object, fold Change of nodes, the default value is NULL. If the user provides the Fold Change value of the nodes, it can be used to set the size of the gene node.
- category_node Number indicating the amount by which plotting category nodes should be scaled relative to the default, the default value is 1.
- gene_node Number indicating the amount by which plotting gene nodes should be scaled relative to the default, the default value is 1.
- category_label Scale of category node label size, the default value is 1.
- gene_label Scale of gene node label size, the default value is 1.

hilight.params list, the parameters to control the attributes of highlighted nodes and edges. see the hilight.params in the following. hilight.params control the attributes of highlight, it can be referred to the following parameters:
- category category nodes to be highlight.
- alpha_hilight alpha of highlighted nodes.
- alpha_no_hilight alpha of unhighlighted nodes.

Details

plot linkages of genes and enriched concepts (e.g. GO categories, KEGG pathways)

Value

ggplot object

Author(s)

Guangchuang Yu

Examples

```r
## Not run:
library(DOSE)
data(geneList)
de <- names(geneList)[1:100]
x <- enrichDO(de)
x2 <- pairwise_termsim(x)
cnetplot(x2)
# use `layout` to change the layout of map
cnetplot(x2, layout = "star")
# use `showCategory` to select the displayed terms. It can be a number of a vector of terms.
cnetplot(x2, showCategory = 10)
categories <- c("pre-malignant neoplasm", "intestinal disease", "breast ductal carcinoma", "non-small cell lung carcinoma")
cnetplot(x2, showCategory = categories)
# 'compareClusterResult' object is also supported.
library(clusterProfiler)
library(DOSE)
library(org.Hs.eg.db)
data(gcSample)
xx <- compareCluster(gcSample, fun="enrichGO", OrgDb="org.Hs.eg.db")
```
```r
xx2 <- pairwise_termsim(xx)
cnetplot(xx2)
## End(Not run)
```

**color_palette**

Description
create color palette for continuous data

Usage
`color_palette(colors)`

Arguments
- `colors` colors of length \( \geq 2 \)

Value
color vector

Author(s)
guangchuang yu

Examples
`color_palette(c("red", "yellow", "green"))`

**dotplot**

Description
dotplot for enrichment result
dotplot

Usage

dotplot(object, ...)

## S4 method for signature 'enrichResult'
dotplot(
  object,
  x = "GeneRatio",
  color = "p.adjust",
  showCategory = 10,
  size = NULL,
  split = NULL,
  font.size = 12,
  title = "",
  orderBy = "x",
  label_format = 30,
  ...
)

## S4 method for signature 'gseaResult'
dotplot(
  object,
  x = "GeneRatio",
  color = "p.adjust",
  showCategory = 10,
  size = NULL,
  split = NULL,
  font.size = 12,
  title = "",
  orderBy = "x",
  label_format = 30,
  ...
)

## S4 method for signature 'compareClusterResult'
dotplot(
  object,
  x = "Cluster",
  color = "p.adjust",
  showCategory = 5,
  split = NULL,
  font.size = 12,
  title = "",
  by = "geneRatio",
  size = NULL,
  includeAll = TRUE,
  label_format = 30,
  ...
)
## S4 method for signature 'enrichResultList'
dotplot(
  object,
  x = "GeneRatio",
  color = "p.adjust",
  showCategory = 10,
  size = NULL,
  split = NULL,
  font.size = 12,
  title = "",
  orderBy = "x",
  label_format = 30,
  ...)

## S4 method for signature 'gseaResultList'
dotplot(
  object,
  x = "GeneRatio",
  color = "p.adjust",
  showCategory = 10,
  size = NULL,
  split = NULL,
  font.size = 12,
  title = "",
  orderBy = "x",
  label_format = 30,
  ...)

dotplot.enrichResult(
  object,
  x = "geneRatio",
  color = "p.adjust",
  showCategory = 10,
  size = NULL,
  split = NULL,
  font.size = 12,
  title = "",
  orderBy = "x",
  label_format = 30,
  decreasing = TRUE)

dotplot.compareClusterResult(
  object,
  x = "Cluster",
colorBy = "p.adjust",
showCategory = 5,
by = "geneRatio",
size = "geneRatio",
split = NULL,
includeAll = TRUE,
font.size = 12,
title = "",
label_format = 30,
group = FALSE,
shape = FALSE,
facet = NULL,
strip_width = 15)

Arguments

object: compareClusterResult object

...: additional parameters

x: variable for x-axis, one of 'GeneRatio' and 'Count'

color: variable that used to color enriched terms, e.g. 'pvalue', 'p.adjust' or 'qvalue'

showCategory: A number or a list of terms. If it is a number, the first n terms will be displayed. If it is a list of terms, the selected terms will be displayed.

size: variable that used to scale the sizes of categories, one of "geneRatio", "Percentage" and "count"

split: apply 'showCategory' to each category specified by the 'split', e.g., "ONTOL-OGY", "category" and "intersect". Default is NULL and do nothing

font.size: font size

title: figure title

orderBy: The order of the Y-axis

label_format: a numeric value sets wrap length, alternatively a custom function to format axis labels. by default wraps names longer that 30 characters

by: one of "geneRatio", "Percentage" and "count"

includeAll: logical

decreasing: logical. Should the orderBy order be increasing or decreasing?

colorBy: variable that used to color enriched terms, e.g. 'pvalue', 'p.adjust' or 'qvalue'

group: a logical value, whether to connect the nodes of the same group with wires.

shape: a logical value, whether to use nodes of different shapes to distinguish the group it belongs to

facet: apply 'facet_grid' to the plot by specified variable, e.g., "ONTOL-OGY", "cate-gory" and "intersect".

strip_width: width of strip text, a.k.a facet label.
drag_network

Description

Drag the nodes of a network to update the layout of the network

Usage

drag_network(p, g = NULL)
emapplot

Arguments

p the network diagram as a ggplot/gg/ggraph object.
g an corresponding igraph object. Default is to extract from the 'ggraph' attribute.

Value

an updated ggplot/gg/ggraph object

Examples

## Not run:
library(igraph)
library(ggraph)

flow_info <- data.frame(from = c(1,2,3,4,5,6),
                        to = c(5,5,6,7,6,7))
g = graph_from_data_frame(flow_info)
p <- ggraph(g, layout='nicely') + geom_node_point() + geom_edge_link()
pp <- drag_network(p)
## End(Not run)

emapplot

Description

Enrichment Map for enrichment result of over-representation test or gene set enrichment analysis

Usage

emapplot(x, ...)

## S4 method for signature 'enrichResult'
emapplot(x, showCategory = 30, ...)

## S4 method for signature 'gseaResult'
emapplot(x, showCategory = 30, ...)

## S4 method for signature 'compareClusterResult'
emapplot(x, showCategory = 30, ...)

emapplot.enrichResult(
  x,
  showCategory = 30,
  layout = NULL,
  coords = NULL,
color = "p.adjust",
min_edge = 0.2,
cex_label_category = 1,
cex_category = 1,
cex_line = 1,
shadowtext = TRUE,
label_style = "shadowtext",
repel = FALSE,
node_label = "category",
with_edge = TRUE,
group_category = FALSE,
group_legend = FALSE,
cex_label_group = 1,
nWords = 4,
label_format = 30,
clusterFunction = stats::kmeans,
nCluster = NULL,
layout.params = list(layout = NULL, coords = NULL),
edge.params = list(show = TRUE, min = 0.2),
cex.params = list(category_node = 1, category_label = 1, line = 1),
hilight.params = list(category = NULL, alpha_hilight = 1, alpha_no_hilight = 0.3),
cluster.params = list(cluster = FALSE, method = stats::kmeans, n = NULL, legend =
FALSE, label_style = "shadowtext", label_words_n = 4, label_format = 30),
...)

emapplot.compareClusterResult(
  x,
  showCategory = 30,
  layout = NULL,
  coords = NULL,
  split = NULL,
  pie = "equal",
  legend_n = 5,
cex_category = 1,
cex_line = 1,
min_edge = 0.2,
cex_label_category = 1,
shadowtext = TRUE,
with_edge = TRUE,
group_category = FALSE,
label_format = 30,
group_legend = FALSE,
node_label = "category",
label_style = "shadowtext",
repel = FALSE,
cex_label_group = 1,
nWords = 4,
clusterFunction = stats::kmeans,
nCluster = NULL,
cex_pie2axis = 1,
pie.params = list(pie = "equal", legend_n = 5),
layout.params = list(layout = NULL, coords = NULL),
edge.params = list(show = TRUE, min = 0.2),
cluster.params = list(cluster = FALSE, method = stats::kmeans, n = NULL, legend = 
FALSE, label_style = "shadowtext", label_words_n = 4, label_format = 30),
cex.params = list(category_node = 1, category_label = 1, line = 1, pie2axis = 1, 
label_group = 1),
hilight.params = list(category = NULL, alpha_hilight = 1, alpha_no_hilight = 0.3),

Arguments

x Enrichment result.

... additional parameters

additional parameters can refer the following parameters.

• force Force of repulsion between overlapping text labels. Defaults to 1.
• nudge_x, nudge_y Horizontal and vertical adjustments to nudge the starting position of each text label.
• direction "both", "x", or "y" – direction in which to adjust position of labels.
• ellipse_style style of ellipse, one of "ggforce" an "polygon".
• ellipse_pro numeric indicating confidence value for the ellipses, it can be used only when ellipse_style = "polygon".
• alpha the transparency of ellipse fill.
• type The type of ellipse. The default "t" assumes a multivariate t-distribution, and "norm" assumes a multivariate normal distribution. "euclid" draws a circle with the radius equal to level, representing the euclidean distance from the center.

showCategory A number or a vector of terms. If it is a number, the first n terms will be displayed. If it is a vector of terms, the selected terms will be displayed.

layout Layout of the map, e.g. 'star', 'circle', 'gem', 'dh', 'graphopt', 'grid', 'mds', 'randomly', 'fr', 'kk', 'drl' or 'lgl'. Will be removed in the next version. Will be removed in the next version.

coords a data.frame with two columns: 'x' for X-axis coordinate and 'y' for Y-axis coordinate. Will be removed in the next version.

color Variable that used to color enriched terms, e.g. 'pvalue', 'p.adjust' or 'qvalue'.

min_edge The minimum similarity threshold for whether two nodes are connected, should be between 0 and 1, default value is 0.2. Will be removed in the next version.

cex_label_category Scale of category node label size. Will be removed in the next version.

cex_category Number indicating the amount by which plotting category nodes should be scaled relative to the default. Will be removed in the next version.
- **cex_line**: Scale of line width. Will be removed in the next version.
- **shadowtext**: a logical value, whether to use shadow font.
- **label_style**: style of group label, one of "shadowtext" and "ggforce". Will be removed in the next version.
- **repel**: whether to correct the position of the label. Defaults to FALSE.
- **node_label**: Select which labels to be displayed, one of 'category', 'group', 'all' and 'none'. Will be removed in the next version.
- **with_edge**: Logical, if TRUE, draw the edges of the network diagram. Will be removed in the next version.
- **group_category**: a logical, if TRUE, group the category. Will be removed in the next version.
- **group_legend**: Logical, if TRUE, the grouping legend will be displayed. The default is FALSE. Will be removed in the next version.
- **cex_label_group**: Numeric, scale of group labels size, the default value is 1. Will be removed in the next version.
- **nWords**: Numeric, the number of words in the cluster tags, the default value is 4. Will be removed in the next version.
- **label_format**: a numeric value sets wrap length, alternatively a custom function to format axis labels. Will be removed in the next version.
- **clusterFunction**: function of Clustering method, such as stats::kmeans(the default), cluster::clara, cluster::fanny or cluster::pam. Will be removed in the next version.
- **nCluster**: Numeric, the number of clusters, the default value is square root of the number of nodes. Will be removed in the next version.
- **layout.params**: list, the parameters to control the layout. see the layout.params in the following. layout.params control the attributes of layout, it can be referred to the following parameters:
  - **layout**: Layout of the map, e.g. 'star', 'circle', 'gem', 'dh', 'graphopt', 'grid', 'mds', 'randomly', 'fr', 'kk', 'drl' or 'lgl'.
  - **coords**: a data.frame with two columns: 'x' for X-axis coordinate and 'y' for Y-axis coordinate.
- **edge.params**: list, the parameters to control the edge. see the edge.params in the following. edge.params control the attributes of edge, it can be referred to the following parameters:
  - **show**: Logical, if TRUE (the default), draw the edges of the network diagram.
  - **min**: The minimum similarity threshold for whether two nodes are connected, should be between 0 and 1, default value is 0.2.
- **cex.params**: list, the parameters to control the edge. see the cex.params in the following. cex.params control the attributes of edge, it can be referred to the following parameters:
  - **category_node**: Number indicating the amount by which plotting category nodes should be scaled relative to the default.
  - **category_label**: Scale of category node label size.
• line Scale of line width.
• pie2axis It is used to adjust the relative size of the pie chart on the coordinate axis, the default value is 1.
• label_group Numeric, scale of group labels size, the default value is 1.

hilight.params list, the parameters to control the attributes of highlighted nodes and edges. see the hilight.params in the following. hilight.params control the attributes of highlight, it can be referred to the following parameters:
• category category nodes to be highlight.
• alpha_hilight alpha of highlighted nodes.
• alpha_no_hilight alpha of unhighlighted nodes.

cluster.params list, the parameters to control the attributes of highlighted nodes and edges. see the cluster.params in the following. cluster.params control the attributes of highlight, it can be referred to the following parameters:
• cluster a logical, if TRUE, group the category.
• method function of Clustering method, such as stats::kmeans(the default), cluster::clara, cluster::fanny or cluster::pam.
• n Numeric, the number of clusters, the default value is square root of the number of nodes.
• legend Logical, if TRUE, the grouping legend will be displayed. The default is FALSE.
• label_style style of group label, one of "shadowtext" and "ggforce".
• label_words_n Numeric, the number of words in the cluster tags, the default value is 4.
• label_format a numeric value sets wrap length, alternatively a custom function to format axis labels.

split separate result by ’category’ variable
pie proportion of clusters in the pie chart, one of ’equal’ (default) and ’Count’. Will be removed in the next version.
legend_n number of circle in legend Will be removed in the next version.
cex_pie2axis It is used to adjust the relative size of the pie chart on the coordinate axis, the default value is 1. Will be removed in the next version.
pie.params list, the parameters to control the attributes of pie nodes. see the pie.params in the following. pie.params control the attributes of pie nodes, it can be referred to the following parameters:
• pie proportion of clusters in the pie chart, one of ’equal’ (default) and ’Count’.
• legend_n number of circle in legend.

Details

This function visualizes gene sets as a network (i.e., enrichment map). Mutually overlapping gene sets tend to cluster together, making it easier for interpretation. When the similarity between terms meets a certain threshold (default is 0.2, adjusted by parameter ‘min_edge’), there will be edges between terms. The stronger the similarity, the shorter and thicker the edges. The similarity between terms is obtained by function `pairwise_termsim`, the details of similarity calculation can be found in its documentation: `pairwise_termsim`. 
emapplot_cluster

Functional grouping network diagram for enrichment result of over-representation test or gene set enrichment analysis

Description

This function has been replaced by `emapplot`.

Usage

emapplot_cluster(x, ...)

Value

ggplot object

Author(s)

Guangchuang Yu

Examples

```r
## Not run:
library(DOSE)
data(geneList)
de <- names(geneList)[1:100]
x <- enrichDO(de)
x2 <- pairwise_termsim(x)
emapplot(x2)
  # use `layout` to change the layout of map
  emapplot(x2, layout = "star")
# use `showCategory` to select the displayed terms. It can be a number of a vector of terms.
emapplot(x2, showCategory = 10)
categorys <- c("pre-malignant neoplasm", "intestinal disease",
               "breast ductal carcinoma")
emapplot(x2, showCategory = categorys)

# It can also graph compareClusterResult
library(clusterProfiler)
library(DOSE)
library(org.Hs.eg.db)
data(gcSample)
xx <- compareCluster(gcSample, fun="enrichGO", OrgDb="org.Hs.eg.db")
xx2 <- pairwise_termsim(xx)
emapplot(xx2)

## End(Not run)
```
Arguments

x  enrichment result

...  additional parameters. Please refer to: emapplot.

Value

ggplot2 object

Description

convert compareClusterResult to a data.frame that ready for plot

convert enrichResult object for ggplot2

Usage

```r
## S3 method for class 'compareClusterResult'
fortify(
  model,
  data,
  showCategory = 5,
  by = "geneRatio",
  split = NULL,
  includeAll = TRUE,
  ...
)
```

```r
## S3 method for class 'enrichResult'
fortify(
  model,
  data,
  showCategory = 5,
  by = "Count",
  order = FALSE,
  drop = FALSE,
  split = NULL,
  ...
)
```
**Arguments**

- **model**: 'enrichResult' or 'compareClusterResult' object
- **data**: not use here
- **showCategory**: Category numbers to show
- **by**: one of Count and GeneRatio
- **split**: separate result by 'split' variable
- **includeAll**: logical
- **...**: additional parameter
- **order**: logical
- **drop**: logical

**Value**

- data.frame
- data.frame

**Author(s)**

Guangchuang Yu

---

**Description**

label genes in running score plot

**Usage**

```r
geom_gsea_gene(
  genes,
  mapping = NULL,
  geom = ggplot2::geom_text,
  ...,
  geneSet = NULL
)
```

**Arguments**

- **genes**: selected genes to be labeled
- **mapping**: aesthetic mapping, default is NULL
- **geom**: geometric layer to plot the gene labels, default is geom_text
- **...**: additional parameters passed to the 'geom'
- **geneSet**: choose which gene set(s) to be label if the plot contains multiple gene sets
**ggtable**

**Value**

ggplot object

**Author(s)**

Guangchuang Yu

---

**goplot**

**Description**

plot induced GO DAG of significant terms

**Usage**

```r
plot_sequence = c(
    goplot(d, p = NULL)
)
```

**Arguments**

- `d`: data frame
- `p`: ggplot object to extract color to color rownames(d), optional

**Value**

ggplot object

**Author(s)**

guangchuang yu
Usage

goplot(
  x,
  showCategory = 10,
  color = "p.adjust",
  layout = "sugiyama",
  geom = "text",
  ...
)

## S4 method for signature 'enrichResult'
goplot(
  x,
  showCategory = 10,
  color = "p.adjust",
  layout = "sugiyama",
  geom = "text",
  ...
)

## S4 method for signature 'gseaResult'
goplot(
  x,
  showCategory = 10,
  color = "p.adjust",
  layout = "sugiyama",
  geom = "text",
  ...
)

goplot.enrichResult(
  x,
  showCategory = 10,
  color = "p.adjust",
  layout = "sugiyama",
  geom = "text",
  ...
)

Arguments

x  enrichment result.
showCategory  number of enriched terms to display
color  variable that used to color enriched terms, e.g. pvalue, p.adjust or qvalue
layout  layout of the map
geom  label geom, one of 'label' or 'text'
...  additional parameter
gseadist

Value

ggplot object

Author(s)

Guangchuang Yu

Examples

```r
## Not run:
library(clusterProfiler)
data(geneList, package = "DOSE")
de <- names(geneList)[1:100]
yy <- enrichGO(de, 'org.Hs.eg.db', ont="BP", pvalueCutoff=0.01)
gplot(yy)
gplot(yy, showCategory = 5)
## End(Not run)
```

---

gseadist  gseadist

description

plot logFC distribution of selected gene sets

Usage

gseadist(x, IDs, type = "density")

Arguments

- `x`: GSEA result
- `IDs`: gene set IDs
- `type`: one of 'density' or 'boxplot'

Value

distribution plot

Author(s)

Guangchuang Yu
gseaplot

Description
visualize analyzing result of GSEA

Usage

```r
gseaplot(x, geneSetID, by = "all", title = "", ...)  # S4 method for signature 'gseaResult'
gseaplot(
  x,
  geneSetID,
  by = "all",
  title = "",
  color = "black",
  color.line = "green",
  color.vline = "#FA5860",
  ...
)
```

gseaplot.gseaResult(
  x,
  geneSetID,
  by = "all",
  title = "",
  color = "black",
  color.line = "green",
  color.vline = "#FA5860",
  ...
)

Arguments

- `x` object of gsea result
- `geneSetID` geneSet ID
- `by` one of "runningScore" or "position"
- `title` plot title
- `...` additional parameters
- `color` color of line segments
- `color.line` color of running enrichment score line
- `color.vline` color of vertical line which indicating the maximum/minimal running enrichment score
Details
plotting function for gseaResult

Value
ggplot2 object

Author(s)
Guangchuang Yu

Examples
library(DOSE)
data(geneList)
x <- gseDO(geneList)
gseaplot(x, geneSetID=1)

gseaplot2

Description
GSEA plot that mimic the plot generated by broad institute’s GSEA software

Usage
gseaplot2(
  x,
  geneSetID,
  title = "",
  color = "green",
  base_size = 11,
  rel_heights = c(1.5, 0.5, 1),
  subplots = 1:3,
  pvalue_table = FALSE,
  ES_geom = "line"
)

Arguments
x gseaResult object

geneSetID gene set ID

title plot title

color color of running enrichment score line
Description
plot ranked list of genes with running enrichment score as bar height

Usage
```
gsearank(x, geneSetID, title = "", output = "plot")
```

Arguments
```
x           gseaResult object
geneSetID   gene set ID
title       plot title
output      one of 'plot' or 'table' (for exporting data)
```

Value
```
ggplot object
```

Author(s)
Guangchuang Yu
**gsInfo**

**Description**

extract gsea result of selected geneSet

**Usage**

gsInfo(object, geneSetID)

**Arguments**

- **object**: gseaResult object
- **geneSetID**: gene set ID

**Value**

data.frame

**Author(s)**

Guangchuang Yu

---

**heatplot**

**Description**

heatmap like plot for functional classification

**Usage**

heatplot(x, showCategory = 30, ...)

```r
## S4 method for signature 'enrichResult'
heatplot(x, showCategory = 30, ...)

## S4 method for signature 'gseaResult'
heatplot(x, showCategory = 30, ...)

heatplot.enrichResult(
  x,
  showCategory = 30,
  symbol = "rect",
```
foldChange = NULL,
pvalue = NULL,
label_format = 30
)

Arguments

x enrichment result.
showCategory number of enriched terms to display
... Additional parameters
symbol symbol of the nodes, one of "rect" (the default) and "dot" by default wraps names longer than 30 characters
foldChange fold Change.
pvalue pvalue of genes
label_format a numeric value sets wrap length, alternatively a custom function to format axis labels.

Value

ggplot object

Author(s)

guangchuang yu
Guangchuang Yu

Examples

library(DOSE)
data(geneList)
de <- names(geneList)[1:100]
x <- enrichDO(de)
heatmap(x)

Description

Get the similarity matrix
Usage

```r
pairwise_termsim(x, method = "JC", semData = NULL, showCategory = 200)
```

### S4 method for signature 'enrichResult'

```r
pairwise_termsim(x, method = "JC", semData = NULL, showCategory = 200)
```

### S4 method for signature 'gseaResult'

```r
pairwise_termsim(x, method = "JC", semData = NULL, showCategory = 200)
```

### S4 method for signature 'compareClusterResult'

```r
pairwise_termsim(x, method = "JC", semData = NULL, showCategory = 200)
```

pairwise_termsim.enrichResult(
  x,
  method = "JC",
  semData = NULL,
  showCategory = 200
)

pairwise_termsim.compareClusterResult(
  x,
  method = "JC",
  semData = NULL,
  showCategory = 200
)

Arguments

- **x**: enrichment result.
- **method**: method of calculating the similarity between nodes, one of "Resnik", "Lin", "Rel", "Jiang", "Wang" and "JC" (Jaccard similarity coefficient) methods.
- **semData**: GOSemSimDATA object, can be obtained through `godata` function in GOSemSim package.
- **showCategory**: number of enriched terms to display, default value is 200.

Details

This function adds similarity matrix to the `termsim` slot of enrichment result. Users can use the `method` parameter to select the method of calculating similarity. The Jaccard correlation coefficient (JC) is used by default, and it applies to all situations. When users want to calculate the correlation between GO terms or DO terms, they can also choose "Resnik", "Lin", "Rel" or "Jiang" (they are semantic similarity calculation methods from GOSemSim packages), and at this time, the user needs to provide `semData` parameter, which can be obtained through `godata` function in GOSemSim package.

Examples

```r
## Not run:
```
library(clusterProfiler)
library(org.Hs.eg.db)
library(enrichplot)
library(GOSemSim)
library(DOSE)
data(geneList)
gene <- names(geneList)[abs(geneList) > 2]
eg0 <- enrichGO(gene = gene,
          universe = names(geneList),
          OrgDb = org.Hs.eg.db,
          ont = "BP",
          pAdjustMethod = "BH",
          pvalueCutoff = 0.01,
          qvalueCutoff = 0.05,
          readable = TRUE)
d <- godata('org.Hs.eg.db', ont="BP")
eg02 <- pairwise_termsim(ego, method="Wang", semData = d)
emappplot(ego2)
emappplot_cluster(ego2)

## End(Not run)

---

### plotting.clusterProfile

#### Description

Internal plot function for plotting compareClusterResult

#### Usage

```r
plotting.clusterProfile(
  clProf.reshape.df,
  x = ~Cluster,
  type = "dot",
  colorBy = "p.adjust",
  by = "geneRatio",
  title = "",
  font.size = 12
)
```

#### Arguments

- `clProf.reshape.df`
  - data frame of compareCluster result
- `x`
  - x variable
- `type`
  - one of dot and bar
pmcplot

colorBy | one of pvalue or p.adjust
by | one of percentage and count
title | graph title
font.size | graph font size

Value

ggplot object

Author(s)

Guangchuang Yu [https://guangchuangyu.github.io

Description

PubMed Central Trend plot

Usage

pmcplot(query, period, proportion = TRUE)

Arguments

query | query terms
period | period of query in the unit of year
proportion | If TRUE, use query_hits/all_hits, otherwise use query_hits

Value

ggplot object

Author(s)

guangchuang yu

reexports

Objects exported from other packages

Description

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

aplot plot_list
ggplot2 facet_grid, ggtitle
Description

ridgeline plot for GSEA result

Usage

ridgeplot(
  x,
  showCategory = 30,
  fill = "p.adjust",
  core_enrichment = TRUE,
  label_format = 30,
  ...
)

## S4 method for signature 'gseaResult'
ridgeplot(
  x,
  showCategory = 30,
  fill = "p.adjust",
  core_enrichment = TRUE,
  label_format = 30,
  ...
)

ridgeplot.gseaResult(
  x,
  showCategory = 30,
  fill = "p.adjust",
  core_enrichment = TRUE,
  label_format = 30,
  orderBy = "NES",
  decreasing = FALSE
)

Arguments

x gseaResult object

showCategory A number or a vector of terms. If it is a number, the first n terms will be displayed. If it is a vector of terms, the selected terms will be displayed.

fill one of "pvalue", "p.adjust", "qvalue"

core_enrichment whether only using core_enriched genes
Description

helper function to set color for enrichplot

Usage

```r
set_enrichplot_color(
  colors = get_enrichplot_color(2),
  type = "color",
  name = NULL,
  .fun = NULL,
  ...
)
```

Arguments

- **colors**: user provided color vector
- **type**: one of ‘color’, ‘colour’ or ‘fill’
- **name**: name of the color legend
- **.fun**: force to use user provided color scale function
- **...**: additional parameter that passed to the color scale function
Value

a color scale

---

Description

Similarity space plot of enrichment analysis results.

Usage

ssplot(x, ...)

## S4 method for signature 'enrichResult'
ssplot(x, showCategory = 30, ...)

## S4 method for signature 'gseaResult'
ssplot(x, showCategory = 30, ...)

## S4 method for signature 'compareClusterResult'
ssplot(x, showCategory = 30, ...)

ssplot.enrichResult(
  x,
  showCategory = 30,
  drfun = NULL,
  with_edge = FALSE,
  dr.params = list(),
  group_category = TRUE,
  node_label = "group",
  ...
)

ssplot.compareClusterResult(
  x,
  showCategory = 30,
  split = NULL,
  pie = "equal",
  drfun = NULL,
  with_edge = FALSE,
  cex_pie2axis = 0.0125,
  dr.params = list(),
  group_category = TRUE,
  node_label = "group",
  ...
)
ssplot

Arguments

x  Enrichment result.

...  additional parameters

additional parameters can refer the following parameters.

• coords a data.frame with two columns: ‘x’ for X-axis coordinate and ‘y’ for Y-axis coordinate.
• color Variable that used to color enriched terms, e.g. 'pvalue', 'p.adjust' or 'qvalue'. the starting position of each text label.
• cex_line Scale of line width.
• min_edge The minimum similarity threshold for whether two nodes are connected, should between 0 and 1, default value is 0.2.
• cex_label_category Scale of category node label size.
• cex_category Number indicating the amount by which plotting category nodes should be scaled relative to the default.
• shadowtext a logical value, whether to use shadow font.
• label_style style of group label, one of "shadowtext" and "ggforce".
• repel whether to correct the position of the label. Defaults to FALSE.
• group_legend Logical, if TRUE, the grouping legend will be displayed. The default is FALSE.
• cex_label_group Numeric, scale of group labels size, the default value is 1.
• nWords Numeric, the number of words in the cluster tags, the default value is 4.
• label_format a numeric value sets wrap length, alternatively a custom function to format axis labels.
• clusterFunction function of Clustering method, such as stats::kmeans(the default), cluster::clara, cluster::fanny or cluster::pam.
• nCluster Numeric, the number of clusters, the default value is square root of the number of nodes.

additional parameters can refer the emapplot function: emapplot.

showCategory A number or a vector of terms. If it is a number, the first n terms will be displayed. If it is a vector of terms, the selected terms will be displayed.

drfun The function used for dimension reduction, e.g. stats::cmdscale (the default), vegan::metaMDS, or ape::pcoa.

with_edge Logical, if TRUE, draw the edges of the network diagram. Will be removed in the next version.

dr.params list, the parameters of tidydr::dr. one of 'category', 'group', 'all' and 'none'.

group_category a logical, if TRUE, group the category. Will be removed in the next version.

node_label Select which labels to be displayed, one of 'category', 'group', 'all' and 'none'.

split separate result by 'category' variable

pie proportion of clusters in the pie chart, one of 'equal' (default) and 'Count' Will be removed in the next version.

cex_pie2axis It is used to adjust the relative size of the pie chart on the coordinate axis, the default value is 0.0125.
## Not run:
library(clusterProfiler)
library(org.Hs.eg.db)
library(enrichplot)
library(GOSemSim)
library(DOSE)
data(geneList)
gene <- names(geneList)[abs(geneList) > 2]
ego <- enrichGO(gene = gene,
  universe = names(geneList),
  OrgDb = org.Hs.eg.db,
  ont = "BP",
  pAdjustMethod = "BH",
  pvalueCutoff = 0.01,
  qvalueCutoff = 0.05,
  readable = TRUE)
d <- godata('org.Hs.eg.db', ont="BP")
ego2 <- pairwise_termsim(ego, method = "Wang", semData = d)
ssplot(ego2)

## End(Not run)

description
Functional grouping tree diagram for enrichment result of over-representation test or gene set
enrichment analysis.

### Usage

treeplot(x, ...)

## S4 method for signature 'enrichResult'
treeplot(x, ...)

## S4 method for signature 'gseaResult'
treeplot(x, ...)

## S4 method for signature 'compareClusterResult'
treeplot(x, ...)
treeplot.enrichResult(
    x,
    showCategory = 30,
    color = "p.adjust",
    nWords = 4,
    nCluster = 5,
    cex_category = 1,
    label_format = NULL,
    label_format_cladelab = 30,
    label_format_tiplab = NULL,
    fontsize = 4,
    offset = rel(1),
    offset_tiplab = rel(1),
    hclust_method = "ward.D",
    group_color = NULL,
    extend = 0.3,
    hilight = TRUE,
    hexpand = 0.1,
    align = "both",
    hilight.params = list(hilight = TRUE, align = "both"),
    offset.params = list(bar_tree = rel(1), tiplab = rel(1), extend = 0.3, hexpand = 0.1),
    cluster.params = list(method = "ward.D", n = 5, color = NULL, label_words_n = 4,
        label_format = 30),
    ...
)

.treeplot.compareClusterResult(
    x,
    showCategory = 5,
    color = "p.adjust",
    nWords = 4,
    nCluster = 5,
    cex_category = 1,
    split = NULL,
    label_format = NULL,
    label_format_cladelab = 30,
    label_format_tiplab = NULL,
    fontsize = 4,
    offset = rel(1),
    pie = "equal",
    legend_n = 3,
    offset_tiplab = rel(1),
    hclust_method = "ward.D",
    group_color = NULL,
    extend = 0.3,
    hilight = TRUE,
    geneClusterPanel = "heatMap",
    hexpand = 0.1,
align = "both",
cluster.params = list(method = "ward.D", n = 5, color = NULL, label_words_n = 4,
                        label_format = 30),
hilight.params = list(hilight = TRUE, align = "both"),
color = NULL, label_words_n = 4,
colnames_angle = 0),
offset.params = list(bar_tree = rel(1), tiplab = rel(1), extend = 0.3, hexpand = 0.1),
...)

Arguments

x enrichment result.
... additional parameters
showCategory number of enriched terms to display
color variable that used to color enriched terms, e.g. pvalue, p.adjust or qvalue
nWords The number of words in the cluster tags. Will be removed in the next version.
nCluster The number of clusters, the default value is 5. Will be removed in the next version.
cex_category Number indicating the amount by which plotting category. nodes should be scaled relative to the default. Will be removed in the next version.
label_format a numeric value sets wrap length, alternatively a custom function to format axis labels.
label_format_cladelab label_format for group labels, a numeric value sets wrap length, alternatively a custom function to format axis labels. Will be removed in the next version.
label_format_tiplab label_format for tiplabs, a numeric value sets wrap length, alternatively a custom function to format axis labels. Will be removed in the next version.
fontsize The size of text, default is 4.
offset rel object or numeric value, distance bar and tree, offset of bar and text from the clade, default is rel(1), meaning 1 * 1.2 * x_range_of_tree plus distance_between_tree_and_tiplab (1 * (1.2 * x_range_of_tree + distance_between_tree_and_tiplab)). Will be removed in the next version.
offset_tiplab tiplab offset, rel object or numeric value, the bigger the number, the farther the distance between the node and the branch. The default is rel(1), when geneClusterPanel = "pie", meaning 1 * max_radius_of_the_pies; when geneClusterPanel = "heatMap", meaning 1 * 0.16 * column_number_of_heatMap * x_range_of_tree; when geneClusterPanel = "dotplot", meaning 1 * 0.09 * column_number_of_dotplot * x_range_of_tree. Will be removed in the next version.
hclust_method Method of hclust. This should be (an unambiguous abbreviation of) one of "ward.D", "ward.D2", "single", "complete", "average" (= UPGMA), "mcquitty" (= WPGMA), "median" (= WPGMC) or "centroid" (= UPGMC). Will be removed in the next version.
group_color A vector of group colors, the length of the vector should be the same as nCluster.
                        Will be removed in the next version.
extend Numeric, extend the length of bar, default is 0.3. Will be removed in the next version.

hilight Logical value, if TRUE(default), add ggtree::geom_hilight() layer. Will be removed in the next version.

hexpand expand x limits by amount of xrange * hexpand. Will be removed in the next version.

align control the align direction of the edge of high light rectangular. Options is 'none', 'left', 'right', 'both (default)'. Will be removed in the next version.

hilight.params list, the parameters to control the attributes of highlight layer. see the hilight.params in the following. hilight.params control the attributes of highlight layer, it can be referred to the following parameters:
  • hilight Logical value, if TRUE(default), add ggtree::geom_hilight() layer.
  • align control the align direction of the edge of high light rectangular. Options is 'none', 'left', 'right', 'both (default)'.

offset.params list, the parameters to control the offset. see the offset.params in the following. offset.params control the attributes of offset, it can be referred to the following parameters:
  • bar_tree rel object or numeric value, distance bar and tree, offset of bar and text from the clade, default is rel(1), meaning 1 * 1.2 * x_range_of_tree plus distance_between_tree_and_tiplab (1 * (1.2 * x_range_of_tree + distance_between_tree_and_tiplab)).
  • tiplab tiplab offset, rel object or numeric value, the farther the distance between the node and the branch. The default is rel(1), when clusterPanel = "pie", meaning 1 * max_radius_of_the_pies; when clusterPanel = "heatMap", meaning 1 * 0.16 * column_number_of_heatMap * x_range_of_tree; when clusterPanel = "dotplot", meaning 1 * 0.09 * column_number_of_dotplot * x_range_of_tree.
  • extend Numeric, extend the length of bar, default is 0.3.
  • hexpand expand x limits by amount of xrange * hexpand.

cluster.params list, the parameters to control the attributes of highlighted nodes and edges. see the cluster.params in the following. cluster.params control the attributes of highlight, it can be referred to the following parameters:
  • method function of Clustering method, such as stats::kmeans(the default), cluster::clara, cluster::fanny or cluster::pam.
  • n Numeric, the number of clusters, the default value is square root of the number of nodes.
  • color A vector of group colors, the length of the vector should be the same as nCluster.
  • label_words_n Numeric, the number of words in the cluster tags, the default value is 4.
  • label_format A numeric value sets wrap length, alternatively a custom function to format axis labels.

split Separate result by 'category' variable.

pie Used only when geneClusterPanel = "pie", proportion of clusters in the pie chart, one of 'equal' (default) and 'Count'. Will be removed in the next version.
legend_n  Number of circle in legend, the default value is 3. Will be removed in the next version.

geneClusterPanel  one of "heatMap" (default), "dotplot", "pie". Will be removed in the next version.

clusterPanel.params  list, the parameters to control the attributes of cluster panel. see the clusterPanel.params in the following. clusterPanel.params control the attributes of cluster panel, it can be referred to the following parameters:

- clusterPanel one of "heatMap" (default), "dotplot", "pie".
- pie pUsed only when ClusterPanel = "pie", proportion of clusters in the pie chart, one of 'equal' (default) and 'Count'.
- legend_n number of circle in legend.
- colnames_angle set the angle of colnames.

Details

This function visualizes gene sets as a tree. Gene sets with high similarity tend to cluster together, making it easier for interpretation.

Value

ggplot object

Examples

```r
## Not run:
library(clusterProfiler)
library(org.Hs.eg.db)
library(enrichplot)
library(GOSemSim)
library(ggplot2)
library(DOSE)
data(geneList)
gene <- names(geneList)[abs(geneList) > 2]
ego <- enrichGO(gene = gene,
   universe = names(geneList),
   OrgDb = org.Hs.eg.db,
   ont = "BP",
   pAdjustMethod = "BH",
   pvalueCutoff = 0.01,
   qvalueCutoff = 0.05,
   readable = TRUE)
d <- godata('org.Hs.eg.db', ont="BP")
eg02 <- pairwise_termsim(ego, method = "Wang", semData = d)
treepplot(ego2, showCategory = 30)
# use 'hilight' = FALSE to remove ggtree::geom_hilight() layer.
treepplot(ego2, showCategory = 30, hilight = FALSE)
# use 'offset' parameter to adjust the distance of bar and tree.
treepplot(ego2, showCategory = 30, hilight = FALSE, offset = rel(1.5))
# use 'offset_tiplab' parameter to adjust the distance of nodes and branches.
```

treeplot(ego2, showCategory = 30, hilight = FALSE, offset_tiplot = rel(1.5))
keep <- rownames(ego2@termsim)[c(1:10, 16:20)]
keep
treeplot(ego2, showCategory = keep)
treeplot(ego2, showCategory = 20,
    group_color = c("#999999", "#E69F00", "#56B4E9", "#009E73", "#F0E442"))
# It can also graph compareClusterResult
data(gcSample)
xx <- compareCluster(gcSample, fun="enrichKEGG",
    organism="hsa", pvalueCutoff=0.05)
xx <- pairwise_termsim(xx)
treeplot(xx)
# use "geneClusterPanel" to change the gene cluster panel.
treeplot(xx, geneClusterPanel = "dotplot")
treeplot(xx, geneClusterPanel = "pie")
## End(Not run)

**upsetplot**

### upsetplot method

#### Description

upsetplot method generics

#### Usage

upsetplot(x, ...)

```
## S4 method for signature 'enrichResult'
upsetplot(x, n = 10, ...)

## S4 method for signature 'gseaResult'
upsetplot(x, n = 10, ...)
```

#### Arguments

- **x**: object
- ...: additional parameters
- **n**: number of categories to be plotted

#### Value

plot
Author(s)

Guangchuang Yu

Examples

```r
require(DOSE)
data(geneList)
de=names(geneList)[1:100]
x <- enrichDO(de)
upsetplot(x, 8)
```
Index

* internal
  enrichplot-package, 2
  reexports, 31
autofacet, 3
barplot.enrichResult, 4
cnetplot, 5
cnetplot,compareClusterResult-method (cnetplot), 5
cnetplot,enrichResult-method (cnetplot), 5
cnetplot,gseaResult-method (cnetplot), 5
cnetplot,list-method (cnetplot), 5
cnetplot,enrichResult (cnetplot), 5
color_palette, 8
dotplot, 8
dotplot,compareClusterResult,ANY-method (dotplot), 8
dotplot,compareClusterResult-method (dotplot), 8
dotplot,enrichResult-method (dotplot), 8
dotplot,enrichResultList,ANY-method (dotplot), 8
dotplot,enrichResultList-method (dotplot), 8
dotplot,gseaResult-method (dotplot), 8
dotplot,gseaResultList,ANY-method (dotplot), 8
dotplot,gseaResultList-method (dotplot), 8
dotplot,compareClusterResult (dotplot), 8
dotplot,enrichResult (dotplot), 8
drag_network, 12
effapplot, 13, 19, 35
effapplot,compareClusterResult-method (effapplot), 13
emappplot,enrichResult-method (emappplot), 13
emappplot,gseaResult-method (emappplot), 13
effapplot.compareClusterResult (emappplot), 13
effapplot.enrichResult (emappplot), 13
effapplot_cluster, 18
effapplot (enrichplot-package), 2
effapplot-package, 2
facet_grid, 31
facet_grid (reexports), 31
fortify.compareClusterResult, 19
fortify.enrichResult
  (fortify.compareClusterResult), 19
geom_gsea_gene, 20
ggtable, 21
ggtitle, 31

ggtitle (reexports), 31
godata, 29
gplot, 21
gplot,compareClusterResult-method (gplot), 21
gplot,gseaResult-method (gplot), 21
gplot.enrichResult (gplot), 21
gseadist, 23
gseaplot, 24
gseaplot,gseaResult-method (gseaplot), 24
gseaplot2, 25
gsearank, 26
gsInfo, 27
heatplot, 27
heatplot,enrichResult-method (heatplot), 27
heatplot.gseaResult-method (heatplot), 27
heatplot.enrichResult (heatplot), 27
pairwise_termsim, 17, 28
pairwise_termsim,compareClusterResult-method (pairwise_termsim), 28
pairwise_termsim,enrichResult-method (pairwise_termsim), 28
pairwise_termsim,gseaResult-method (pairwise_termsim), 28
pairwise_termsim,compareClusterResult (pairwise_termsim), 28
pairwise_termsim,enrichResult (pairwise_termsim), 28
plot_list, 31
plot_list (reexports), 31
plotting.clusterProfile, 30
pmcplot, 31
reexports, 31
ridgeplot, 32
ridgeplot,gseaResult-method (ridgeplot), 32
ridgeplot.gseaResult (ridgeplot), 32
set_enrichplot_color, 33
ssplot, 34
ssplot,compareClusterResult-method (ssplot), 34
ssplot,enrichResult-method (ssplot), 34
ssplot,gseaResult-method (ssplot), 34
ssplot,compareClusterResult (ssplot), 34
ssplot,enrichResult (ssplot), 34
treeplot, 36
treeplot,compareClusterResult-method (treeplot), 36
treeplot,enrichResult-method (treeplot), 36
treeplot,gseaResult-method (treeplot), 36
treeplot,compareClusterResult (treeplot), 36
treeplot,enrichResult (treeplot), 36
upsetplot, 41
upsetplot,enrichResult,ANY-method (upsetplot), 41
upsetplot,gseaResult (upsetplot), 41
upsetplot,gseaResult-method (upsetplot), 41