# Package ‘rrvgo’

May 7, 2024

<table>
<thead>
<tr>
<th>Type</th>
<th>Package</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Reduce + Visualize GO</td>
</tr>
<tr>
<td>Version</td>
<td>1.16.0</td>
</tr>
<tr>
<td>Description</td>
<td>Reduce and visualize lists of Gene Ontology terms by identifying redundancy based on semantic similarity.</td>
</tr>
<tr>
<td>License</td>
<td>GPL-3</td>
</tr>
<tr>
<td>Encoding</td>
<td>UTF-8</td>
</tr>
<tr>
<td>Imports</td>
<td>GOSemSim, AnnotationDbi, GO.db, pheatmap, ggplot2, ggrepel, treeMap, tm, wordcloud, shiny, grDevices, grid, stats, methods, umap</td>
</tr>
<tr>
<td>Suggests</td>
<td>knitr, rmarkdown, BiocStyle, testthat (&gt;= 2.1.0), shinydashboard, DT, plotly, heatmaply, magrittr, utils, clusterProfiler, DOSE, slam, org.Ag.eg.db, org.At.tair.db, org.Bt.eg.db, org.Ce.eg.db, org.Cf.eg.db, org.Dm.eg.db, org.Dr.eg.db, org.EcK12.eg.db, org.EcSakai.eg.db, org.Gg.eg.db, org.Hs.eg.db, org.Mm.eg.db, org.Mmu.eg.db, org.Pt.eg.db, org.Rn.eg.db, org.Sc.sgd.db, org.Ss.eg.db, org.Xl.eg.db</td>
</tr>
<tr>
<td>VignetteBuilder</td>
<td>knitr</td>
</tr>
<tr>
<td>RoxygenNote</td>
<td>7.2.3</td>
</tr>
<tr>
<td>biocViews</td>
<td>Annotation, Clustering, GO, Network, Pathways, Software</td>
</tr>
<tr>
<td>NeedsCompilation</td>
<td>no</td>
</tr>
<tr>
<td>git_url</td>
<td><a href="https://git.bioconductor.org/packages/rrvgo">https://git.bioconductor.org/packages/rrvgo</a></td>
</tr>
<tr>
<td>git_branch</td>
<td>RELEASE_3_19</td>
</tr>
<tr>
<td>git_last_commit</td>
<td>854dbd2</td>
</tr>
<tr>
<td>git_last_commit_date</td>
<td>2024-04-30</td>
</tr>
<tr>
<td>Repository</td>
<td>Bioconductor 3.19</td>
</tr>
<tr>
<td>Date/Publication</td>
<td>2024-05-06</td>
</tr>
</tbody>
</table>
calculateSimMatrix

**calculateSimMatrix**

Calculate the score similarity matrix between terms

**Description**

calculateSimMatrix Calculate the score similarity matrix between terms

**Usage**

calculateSimMatrix(
  x,
  orgdb,
  keytype = "ENTREZID",
  semdata = GOSemSim::godata(orgdb, ont = ont, keytype = keytype),
  ont = c("BP", "MF", "CC"),
  method = c("Resnik", "Lin", "Rel", "Jiang", "Wang")
)

**Arguments**

- **x**: vector of GO terms
- **orgdb**: one of org.* Bioconductor packages (the package name, or the package itself)
- **keytype**: keytype passed to AnnotationDbi::keys to retrieve GO terms associated to gene ids in your orgdb
getGoSize

semdata: object with prepared GO DATA for measuring semantic similarity
ont: ontology. One of c("BP", "MF", "CC")
method: distance method. One of the supported methods by GOSemSim: c("Resnik", "Lin", "Rel", "Jiang", "Wang")

Details

All similarity measures available are those implemented in the [GOSemSim package](https://www.bioconductor.org/packages/release/bioc/html/GOSemSim.html), namely the Resnik, Lin, Relevance, Jiang and Wang methods. See the [Semantic Similarity Measurement Based on GO](https://www.bioconductor.org/packages/release/bioc/vignettes/GOSemSim/inst/doc/GOSemSim.html#semantic-similarity-measurement-based-on-go) section from the GOSemSim documentation for more details.

Value

a square matrix with similarity scores between terms

Examples

go_analysis <- read.delim(system.file("extdata/example.txt", package="rrvgo"))
simMatrix <- calculateSimMatrix(go_analysis$ID, orgdb="org.Hs.eg.db", ont="BP", method="Rel")

getGoSize

getGoSize Get GO term size (# of genes)

Description

getGoSize Get GO term size (# of genes)

Usage

getGoSize(terms, orgdb, keytype, children)

Arguments

terms: GO terms
orgdb: one of org.* Bioconductor packages (the package name, or the package itself)
keytype: keytype passed to AnnotationDbi::keys to retrieve GO terms associated to gene ids in your orgdb
children: include genes in children terms (based on relationships in the GO DAG hierarchy)

Value

number of genes associated with each term
getGoTerm

Description
getGoTerm Get the description of a GO term

Usage
getGoTerm(x)

Arguments
x GO terms

Value
the Term slot in GO.db::GOTERM[[x]]

getTermDisp

Description
getTermDisp Calculate the term dispensability score, defined as the semantic similarity threshold a term was assigned to a cluster (namely, the similarity of a term to the cluster representative term).

Usage
getTermDisp(simMatrix, cluster, clusterRep)

Arguments
simMatrix a (square) similarity matrix
cluster the cluster assignment for each term
clusterRep the cluster representative term

Value
a vector of term dispensability scores
getTermUniq

**getTermUniq** Calculate the term uniqueness score, defined as 1 minus the average semantic similarity of a term to all other terms.

**Description**

gTermUniq Calculate the term uniqueness score, defined as 1 minus the average semantic similarity of a term to all other terms.

**Usage**

gTermUniq(simMatrix, cluster = NULL)

**Arguments**

<table>
<thead>
<tr>
<th>simMatrix</th>
<th>a (square) similarity matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster</td>
<td>vector with the cluster each entry in the simMatrix belongs to. If NULL, a</td>
</tr>
</tbody>
</table>

**Value**

a vector of term uniqueness scores

---

**gg_color_hue**

**gg_color_hue** Emulate ggplot2 color palette.

**Description**

**gg_color_hue** Emulate ggplot2 color palette.

**Usage**

**gg_color_hue**(n)

**Arguments**

| n   | number of colors |

**Details**

It is just equally spaced hues around the color wheel, starting from 15:

**Value**

a vector with colors (alphanumeric)
heatmapPlot

### Not run:
```r
plot(1:10, pch=16, cex=2, col=gg_color_hue(10))
```  
### End(Not run)

#### Examples
```
go_analysis <- read.delim(system.file("extdata/example.txt", package="rrvgo"))
simMatrix <- calculateSimMatrix(go_analysis$ID, orgdb="org.Hs.eg.db", ont="BP", method="Rel")
scores <- setNames(-log10(go_analysis$qvalue), go_analysis$ID)
reducedTerms <- reduceSimMatrix(simMatrix, scores, threshold=0.7, orgdb="org.Hs.eg.db")
heatmapPlot(simMatrix, reducedTerms, annotateParent=TRUE, annotationLabel="parentTerm", fontsize=6)
```

## Description

heatmapPlot Plot similarity matrix as a heatmap

## Usage

```r
heatmapPlot(
  simMatrix,
  reducedTerms = NULL,
  annotateParent = TRUE,
  annotationLabel = "parentTerm",
  ...
)
```

## Arguments

- `simMatrix` a (square) similarity matrix.
- `reducedTerms` a data.frame with the reduced terms from reduceSimMatrix()
- `annotateParent` whether to add annotation of the parent
- `annotationLabel` display "parent" ids or "parentTerm" string
- `...` other parameters sent to pheatmap::pheatmap()

## Details

Matrix with similarity scores between terms is represented as a heatmap.

## Value

Invisibly a pheatmap object that is a list with components

## Examples

```r
go_analysis <- read.delim(system.file("extdata/example.txt", package="rrvgo"))
simMatrix <- calculateSimMatrix(go_analysis$ID, orgdb="org.Hs.eg.db", ont="BP", method="Rel")
scores <- setNames(-log10(go_analysis$qvalue), go_analysis$ID)
reducedTerms <- reduceSimMatrix(simMatrix, scores, threshold=0.7, orgdb="org.Hs.eg.db")
heatmapPlot(simMatrix, reducedTerms, annotateParent=TRUE, annotationLabel="parentTerm", fontsize=6)
```
**loadOrgdb**  
*loadOrgdb*  
*Load an orgdb object*

**Description**
*loadOrgdb*  
Load an orgdb object

**Usage**
*loadOrgdb*(*orgdb*)

**Arguments**
*orgdb*
  one of org.* Bioconductor packages

**Value**
the loaded orgdb

---

**reduceSimMatrix**  
*reduceSimMatrix*  
Reduce a set of GO terms based on their semantic similarity and scores.

**Description**
*reduceSimMatrix*  
Reduce a set of GO terms based on their semantic similarity and scores.

**Usage**
*reduceSimMatrix*(
  *simMatrix*,
  *scores* = c("uniqueness", "size"),
  *threshold* = 0.7,
  *orgdb*,
  *keytype* = "ENTREZID",
  *children* = TRUE
)

**Arguments**
*simMatrix*  
a (square) similarity matrix

*scores*  
one of c("uniqueness", "size"), or a *named* vector with scores provided for each term, where higher values favor choosing the term as the cluster representative. The default "uniqueness" uses a score reflecting how unique the term is. Note: if you like to use p-values as scores, consider "-1*log-transforming them ("-log(p)")"
scatterPlot

threshold  
similarity threshold (0-1). Some guidance: Large (allowed similarity=0.9), Medium (0.7), Small (0.5), Tiny (0.4) Defaults to Medium (0.7)

orgdb  
one of org.* Bioconductor packages (the package name, or the orgdb object itself)

keytype  
keytype passed to AnnotationDbi::keys to retrieve GO terms associated to gene ids in your orgdb

children  
when retrieving GO term size, include genes in children terms. (based on relationships in the GO DAG hierarchy). Defaults to TRUE

Details

Group terms which are at least within a similarity below ‘threshold’. Decide which term remains based on a score. If no score is provided, then decide based on the "uniqueness" or the term "size". Currently, rrvgo uses the similarity between pairs of terms to compute a distance matrix, defined as (1-simMatrix). The terms are then hierarchically clustered using complete linkage, and the tree is cut at the desired threshold, picking the term with the highest score as the representative of each group.

Therefore, higher thresholds lead to fewer groups, and the threshold should be read as the minimum similarity between group representatives.

Value

a data.frame identifying the different clusters of terms, the parent term representing the cluster, and some metrics of importance describing how unique and dispensable a term is.

Examples

go_analysis <- read.delim(system.file("extdata/example.txt", package="rrvgo"))
simMatrix <- calculateSimMatrix(go_analysis$ID, orgdb="org.Hs.eg.db", ont="BP", method="Rel")
scores <- setNames(-log10(go_analysis$qvalue), go_analysis$ID)
reducedTerms <- reduceSimMatrix(simMatrix, scores, threshold=0.7, orgdb="org.Hs.eg.db")

scatterPlot(simMatrix, reducedTerms, algorithm = c("pca", "umap"), onlyParents = FALSE)

Description

scatterPlot Plot GO terms as scattered points.

Usage

scatterPlot(simMatrix, reducedTerms, algorithm = c("pca", "umap"), onlyParents = FALSE)
scatterPlot

```r
size = "score",
addLabel = TRUE,
labelSize = 3
)
```

**Arguments**

- `simMatrix` a (square) similarity matrix.
- `reducedTerms` a data.frame with the reduced terms from `reduceSimMatrix()`.
- `algorithm` algorithm for dimensionality reduction. Either pca or umap.
- `onlyParents` plot only parent terms. Point size is the number of aggregated terms under the parent.
- `size` what to use as point size. Can be either GO term’s "size" or "score".
- `addLabel` add labels with the most representative term of the group.
- `labelSize` text size in the label.

**Details**

Distances between points represent the similarity between terms. Axes are the first 2 components of applying one of this dimensionality reduction algorithms: - a PCoA to the (di)similarity matrix. - a UMAP (Uniform Manifold Approximation and Projection,[1]) Size of the point represents the provided scores or, in its absence, the number of genes the GO term contains.

**Value**

ggplot2 object ready to be printed (or manipulated)

**References**


**Examples**

go_analysis <- read.delim(system.file("extdata/example.txt", package="rrvgo"))
simMatrix <- calculateSimMatrix(go_analysis$ID, orgdb="org.Hs.eg.db", ont="BP", method="Rel")
scores <- setNames(-log10(go_analysis$qvalue), go_analysis$ID)
reducedTerms <- reduceSimMatrix(simMatrix, scores, threshold=0.7, orgdb="org.Hs.eg.db")
scatterPlot(simMatrix, reducedTerms)
shiny_rrvgo Launch an interactive web interface.

Description

shiny_rrvgo Launch an interactive web interface.

Usage

shiny_rrvgo(...)

Arguments

... other params sent to shiny::runApp().

Value

Nothing

treemapPlot Plot GO terms as a treemap.

Description

treemapPlot Plot GO terms as a treemap.

Usage

treemapPlot(reducedTerms, size = "score", title = "", ...)}

Arguments

reducedTerms a data.frame with the reduced terms from reduceSimMatrix()
size what to use as point size. Can be either GO term’s "size" or "score"
title title of the plot. Defaults to nothing
... other parameters sent to treemap::treemap()

Value

A list from the call to the `treemap()` function is silently returned
Examples

```r
## Not run:
go_analysis <- read.delim(system.file("extdata/example.txt", package="rrvgo"))
simMatrix <- calculateSimMatrix(go_analysis$ID, orgdb="org.Hs.eg.db", ont="BP", method="Rel")
scores <- setNames(-log10(go_analysis$qvalue), go_analysis$ID)
reducedTerms <- reduceSimMatrix(simMatrix, scores, threshold=0.7, orgdb="org.Hs.eg.db")
treemapPlot(reducedTerms)

## End(Not run)
```

### Description
wordcloudPlot Plot GO reduced terms as a wordcloud.

### Usage
wordcloudPlot(reducedTerms, onlyParents = TRUE, ...)

### Arguments
- `reducedTerms` a data.frame with the reduced terms from reduceSimMatrix().
- `onlyParents` use only parent terms to calculate frequencies.
- `...` other parameters sent to wordcloud::wordcloud()

### Value
Nothing

### Examples
```r
go_analysis <- read.delim(system.file("extdata/example.txt", package="rrvgo"))
simMatrix <- calculateSimMatrix(go_analysis$ID, orgdb="org.Hs.eg.db", ont="BP", method="Rel")
scores <- setNames(-log10(go_analysis$qvalue), go_analysis$ID)
reducedTerms <- reduceSimMatrix(simMatrix, scores, threshold=0.7, orgdb="org.Hs.eg.db")
wordcloudPlot(reducedTerms, min.freq=1, colors="black")
```
Index

calculateSimMatrix, 2
getGoSize, 3
getGoTerm, 4
getTermDisp, 4
getTermUniq, 5
gg_color_hue, 5
heatmapPlot, 6
loadOrgdb, 7
reduceSimMatrix, 7
scatterPlot, 8
shiny_rrvgo, 10
treemapPlot, 10
wordcloudPlot, 11