

# Package ‘rrvgo’

September 29, 2022

**Type** Package

**Title** Reduce + Visualize GO

**Version** 1.8.0

**Description** Reduce and visualize lists of Gene Ontology terms by identifying redundancy based on semantic similarity.

**URL** <https://www.bioconductor.org/packages/rrvgo>,  
<https://ssayols.github.io/rrvgo/index.html>

**License** GPL-3

**Encoding** UTF-8

**Imports** GOSemSim, AnnotationDbi, GO.db, pheatmap, ggplot2, ggrepel,  
treemap, tm, wordcloud, shiny, grDevices, grid, stats, methods

**Suggests** knitr, rmarkdown, BiocStyle, testthat (>= 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

**VignetteBuilder** knitr

**RoxygenNote** 7.1.1

**biocViews** Annotation, Clustering, GO, Network, Pathways, Software

**NeedsCompilation** no

**git\_url** <https://git.bioconductor.org/packages/rrvgo>

**git\_branch** RELEASE\_3\_15

**git\_last\_commit** de1ac17

**git\_last\_commit\_date** 2022-04-26

**Date/Publication** 2022-09-29

**Author** Sergi Sayols [aut, cre]

**Maintainer** Sergi Sayols <sergisayolspuig@gmail.com>

## R topics documented:

calculateSimMatrix . . . . .	2
getGoSize . . . . .	3
getGoTerm . . . . .	4
gg_color_hue . . . . .	4
heatmapPlot . . . . .	5
loadOrgdb . . . . .	6
reduceSimMatrix . . . . .	6
scatterPlot . . . . .	7
shiny_rrvgo . . . . .	8
treemapPlot . . . . .	9
wordcloudPlot . . . . .	9

<b>Index</b>	<b>11</b>
--------------	-----------

---

calculateSimMatrix	<i>calculateSimMatrix Calculate the score similarity matrix between terms</i>
--------------------	-------------------------------------------------------------------------------

---

### 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
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/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#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="Re1")
```

---

getGoSize	<i>getGoSize</i> Get GO term size (# of genes)
-----------	------------------------------------------------

---

**Description**

getGoSize Get GO term size (# of genes)

**Usage**

```
getGoSize(terms, orgdb, keytype)
```

**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

**Value**

number of genes associated with each term

---

getGoTerm	<i>getGoTerm</i> Get the description of a GO term
-----------	---------------------------------------------------

---

**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]]

---

gg_color_hue	<i>gg_color_hue</i> 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)

**Examples**

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

---

heatmapPlot	<i>heatmapPlot Plot similarity matrix as a heatmap</i>
-------------	--------------------------------------------------------

---

## Description

heatmapPlot Plot similarity matrix as a heatmap

## Usage

```
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

```
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	<i>loadOrgdb Load an orgdb object</i>
-----------	---------------------------------------

---

**Description**

loadOrgdb Load an orgdb object

**Usage**

```
loadOrgdb(orgdb)
```

**Arguments**

orgdb            one of org.\* Bioconductor packages

**Value**

the loaded orgdb

---

reduceSimMatrix	<i>reduceSimMatrix Reduce a set of GO terms based on their semantic similarity and scores.</i>
-----------------	------------------------------------------------------------------------------------------------

---

**Description**

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

**Usage**

```
reduceSimMatrix(
  simMatrix,
  scores = NULL,
  threshold = 0.7,
  orgdb,
  keytype = "ENTREZID"
)
```

**Arguments**

simMatrix        a (square) similarity matrix

scores            \*named\* vector with scores (weights) assigned to each term. Higher is better. Can be NULL (default, means no scores. In this case, a default score based on set size is assigned, thus favoring larger sets). Note: if you have p-values as scores, consider  $-1 \cdot \log$ -transforming them ( $-\log(p)$ )

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

### Details

Currently, rrvgo uses the similarity between pairs of terms to compute a distance matrix, defined as  $(1 - \text{simMatrix})$ . 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 expected similarity of terms within a group (though this is not entirely correct, and you'll see similarities below this threshold being put in the same group).

### Value

a data.frame with all terms and it's "reducer" (NA if the term was not reduced)

### 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

*scatterPlot Plot GO terms as scattered points.*

---

### Description

scatterPlot Plot GO terms as scattered points.

### Usage

```
scatterPlot(
  simMatrix,
  reducedTerms,
  size = "score",
  addLabel = TRUE,
  labelSize = 3
)
```

**Arguments**

<code>simMatrix</code>	a (square) similarity matrix.
<code>reducedTerms</code>	a data.frame with the reduced terms from <code>reduceSimMatrix()</code>
<code>size</code>	what to use as point size. Can be either GO term's "size" or "score"
<code>addLabel</code>	add labels with the most representative term of the group.
<code>labelSize</code>	text size in the label.

**Details**

Distances between points represent the similarity between terms. Axes are the first 2 components of applying a PCoA to the (di)similarity matrix. 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)

**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

*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	<i>treemapPlot</i> 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**

```
## Not run:
go_analysis <- read.delim(system.file("extdata/example.txt", package="rrvgo"))
simMatrix <- calculateSimMatrix(go_analysis$ID, orgdb="org.Hs.eg.db", ont="BP", method="Re1")
scores <- setNames(-log10(go_analysis$qvalue), go_analysis$ID)
reducedTerms <- reduceSimMatrix(simMatrix, scores, threshold=0.7, orgdb="org.Hs.eg.db")
treemapPlot(reducedTerms)

## End(Not run)
```

---

wordcloudPlot	<i>wordcloudPlot</i> Plot GO reduced terms as a wordcloud.
---------------	------------------------------------------------------------

---

**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**

```
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

`gg_color_hue`, 4

`heatmapPlot`, 5

`loadOrgdb`, 6

`reduceSimMatrix`, 6

`scatterPlot`, 7

`shiny_rrvgo`, 8

`treemapPlot`, 9

`wordcloudPlot`, 9