

Package ‘GoMiner’

May 14, 2025

Version 1.1

Date 2025-05-13

Title Automate the Mapping Between a List of Genes and Gene Ontology Categories

Maintainer Barry Zeeberg <barryz2013@gmail.com>

Author Barry Zeeberg [aut, cre]

Depends R (>= 4.2.0)

Imports minimalistGODB, HGNCHELPER, randomGODB, stats, gplots, grDevices, utils, vprint

LazyData true

LazyDataCompression xz

Description In gene-expression microarray studies, for example, one generally obtains a list of dozens or hundreds of genes that differ in expression between samples and then asks 'What does all of this mean biologically?' Alternatively, gene lists can be derived conceptually in addition to experimentally. For instance, one might want to analyze a group of genes known as housekeeping genes. The work of the Gene Ontology (GO) Consortium <geneontology.org> provides a way to address that question. GO organizes genes into hierarchical categories based on biological process, molecular function and subcellular localization. The role of 'GoMiner' is to automate the mapping between a list of genes and GO, and to provide a statistical summary of the results as well as a visualization.

License GPL (>= 2)

Encoding UTF-8

VignetteBuilder knitr

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

RoxygenNote 7.3.2

Config/testthat/edition 3

NeedsCompilation no

Repository CRAN

Date/Publication 2025-05-13 23:20:02 UTC

Contents

checkGeneListVsDB	2
cluster52	3
FDR	4
GOenrich3	5
GOGOA3small	5
GOheatmap	6
GOhypergeometric3	7
GoMiner	7
GOTable3	9
GOThresh	10
HCCS66	11
hitterBeforeAfterDriver	11
hitters2	12
Housekeeping_Genes	13
human	13
preprocessDB	14
randSubsetGeneList	15
RCPD	15
runGoMinerExamples	16
trimGOGOA3	18
validHGNCsymbols	19
x_fdr	19
x_hyper1	20
x_m	20
x_sampleList1	20
x_tablePop3	20
x_tablePop31	21
x_tableSample3	21
x_thresh	21
Index	22

checkGeneListVsDB	<i>checkGeneListVsDB</i>
-------------------	--------------------------

Description

determine if gene list and database contain compatible identifiers

Usage

```
checkGeneListVsDB(geneList, ontology, GOGOA3, thresh = 0.5, verbose = FALSE)
```

Arguments

geneList	character list of gene names
ontology	character string c("molecular_function", "cellular_component", "biological_process")
GOGOA3	return value of subsetGOGOA()
thresh	numeric acceptance threshold for fraction of gene list matching database identifiers
verbose	integer vector representing classes

Value

returns no value, but may have side effect of aborting the computation

Examples

```
## Not run:
# GOGOA3.RData is too large to include in the R package
# so I need to load it from a file that is not in the package.
# Since this is in a file in my own file system, I could not
# include this as a regular example in the package.
# you can generate it using the package 'minimalistGODB'
# or you can retrieve it from https://github.com/barryzee/GO/tree/main/databases
load("/Users/barryzeeberg/personal/GODB_RDATA/goa_human/GOGOA3_goa_human.RData")
checkGeneListVsDB(geneList=cluster52,ontology="biological_process",
  GOGOA3,thresh=0.5,verbose=TRUE)

# supposed to generate error message
load("/Users/barryzeeberg/personal/GODB_RDATA/sgd/GOGOA3_sgd.RData")
checkGeneListVsDB(geneList=xenopusGenes,ontology="biological_process",
  GOGOA3,thresh=0.5,verbose=TRUE)

## End(Not run)
```

cluster52

GoMiner data set

Description

GoMiner data set

Usage

data(cluster52)

FDR

FDR

Description

compute the false discovery rate (FDR) of the hypergeometric p values of genes mapping to gene ontology (GO) categories

Usage

```
FDR(sampleList, tablePop3, hyper, GOGO3, nrand, ontology, subd, opt = 0)
```

Arguments

sampleList	character vector of user-supplied genes of interest
tablePop3	return value of GOfable3()
hyper	return value of GOhypergeometric3()
GOGO3	return value of subsetGOGO3()
nrand	integer number of randomizations
ontology	c("molecular_function","cellular_component","biological_process")
subd	character string pathname for directory containing sink.txt
opt	integer 0:1 parameter used to determine randomization method

Value

returns a list with FDR information

Examples

```
## Not run:  
# GOGO3.RData is too large to include in the R package  
# so I need to load it from a file that is not in the package.  
# Since this is in a file in my own file system, I could not  
# include this as a regular example in the package.  
# you can generate it using the package 'minimalistGODB'  
# or you can retrieve it from https://github.com/barryzee/GO/tree/main/databases  
load("/Users/barryzeeberg/personal/GODB_RDATA/goa_human/GOGO3_goa_human.RData")  
fdr<-FDR(x_sampleList1,x_tablePop31,x_hyper1,GOGO3,3,"biological_process",tempdir(),0)  
  
## End(Not run)
```

`GOenrich3`*GOenrich3*

Description

compute the gene enrichment in a GO category

Usage

```
GOenrich3(tableSample3, tablePop3)
```

Arguments

`tableSample3` sample return value of `GOTable3()`

`tablePop3` population return value of `GOTable3()`

Value

returns a matrix with columns `c("SAMPLE","POP","ENRICHMENT")`

Examples

```
m<-GOenrich3(x_tableSample3,x_tablePop3)
```

`GOGO3small`*GoMiner data set*

Description

GoMiner data set

Usage

```
data(GOGO3small)
```

GOheatmap

GOheatmap

Description

generate a matrix to be used as input to a heat map

Usage

```
GOheatmap(sampleList, x, thresh, fdrThresh = 0.105, verbose)
```

Arguments

sampleList	character list of gene names
x	DB component of return value of GOtable3()
thresh	output of GOthresh()
fdrThresh	numeric value of FDR acceptance threshold
verbose	integer vector representing classes

Value

returns a matrix to be used as input to a heat map

Examples

```
## Not run:  
# GOGOA3.RData is too large to include in the R package  
# so I need to load it from a file that is not in the package.  
# Since this is in a file in my own file system, I could not  
# include this as a regular example in the package.  
# you can generate it using the package 'minimalistGODB'  
# or you can retrieve it from https://github.com/barryzee/GO/tree/main/databases  
load("/Users/barryzeeberg/personal/GODB_RDATA/goa_human/GOGOA3_goa_human.RData")  
heatmap<-GOheatmap(cluster52,GOGOA3$ontologies[["biological_process"]],x_thresh,verbose=1)  
  
## End(Not run)
```

GOhypergeometric3	<i>GOhypergeometric</i>
-------------------	-------------------------

Description

compute the hypergeometric p value for gene enrichment in a GO category

Usage

```
GOhypergeometric3(tableSample3, tablePop3)
```

Arguments

tableSample3	sample return value of GOfilter3()
tablePop3	population return value of GOfilter3()

Value

returns a matrix with columns c("x","m","n","k","p")

Examples

```
hyper<-GOhypergeometric3(x_tableSample3,x_tablePop3)
```

GoMiner	<i>GoMiner</i>
---------	----------------

Description

driver to generate heatmap

Usage

```
GoMiner(  
  title = NULL,  
  dir,  
  sampleList,  
  GOGOA3,  
  ontology,  
  enrichThresh = 2,  
  countThresh = 5,  
  pvalThresh = 0.1,  
  fdrThresh = 0.1,  
  nrand = 100,  
  mn = 2,
```

```

    mx = 200,
    opt,
    verbose = 1
)

```

Arguments

title	character string descriptive title
dir	character string full pathname to the directory acting result repository
sampleList	character list of gene names
GOGO3	return value of subsetGOGO3()
ontology	character string c("molecular_function", "cellular_component", "biological_process")
enrichThresh	numerical acceptance threshold for enrichment
countThresh	numerical acceptance threshold for gene count
pvalThresh	numerical acceptance threshold for pval
fdrThresh	numerical acceptance threshold for fdr
nrand	numeric number of randomizations to compute FDR
mn	integer param passed to trimGOGO3, min size threshold for a category
mx	integer param passed to trimGOGO3, max size threshold for a category
opt	integer 0:1 parameter used to select randomization method
verbose	integer vector representing classes

Details

modes of FDR estimation: opt=0 use original database with randomized geneLists opt=1 use original geneList with internally scrambled genes databases (uses randomGODB())

databases that can be used with the real geneList: these are explicitly passed as parameter to GoMiner() (1) original GOGO3 (2) randomized version of GOGOSA GOGO3R<-randomGODB(GOGO3) (3) database containing a subset of the big hitters genes (randomGODB2driver()) attempts to compensate for the over-annotation of some genes, that might lead to false positive if gene G has a lot of mappings to categories, randomly sample G/category pairs to retain a reasonable number of them. e.g., reduce G from 100 category mappings to 7 category mappings, by omitting 93 of the mappings G/category mappings

Value

returns a matrix suitable to generate a heatmap

Examples

```

## Not run:
# GOGO3.RData is too large to include in the R package
# so I need to load it from a file that is not in the package.
# Since this is in a file in my own file system, I could not
# include this as a regular example in the package.

```



```

# you can generate it using the package 'minimalistGODB'
# or you can retrieve it from https://github.com/barryzee/GO/tree/main/databases
load("/Users/barryzeeberg/personal/GODB_RDATA/goa_human/GOGO3_goa_human.RData")
l<-GoMiner("Cluster52",tempdir(),cluster52,
  GOGO3=GOGO3,ontology="biological_process",enrichThresh=2,
  countThresh=5,pvalThresh=0.10,fdrThresh=0.10,nrand=2,mn=2,mx=200,opt=0,verbose=1)

# try out yeast database!
load("/Users/barryzeeberg/personal/GODB_RDATA/sgd/GOGO3_sgd.RData")
# make sure this is in fact the database for the desired species
GOGO3$species
# use database to find genes mapping to an interesting category
cat<-"GO_0042149_cellular_response_to_glucose_starvation"
w<-which(GOGO3$ontologies[["biological_process"]][,"GO_NAME"]==cat)
geneList<-GOGO3$ontologies[["biological_process"]][w,"HGNC"]
l<-GoMiner("YEAST",tempdir(),geneList,
  GOGO3,ontology="biological_process",enrichThresh=2,
  countThresh=3,pvalThresh=0.10,fdrThresh=0.10,nrand=2,mn=2,mx=200,opt=0)

## End(Not run)

```

GTable3

GTable3

Description

tabulate number of geneList mappings to GO categories

Usage

```
GTable3(hgncList, DB)
```

Arguments

hgncList	character list of gene names
DB	selected ontology branch of return value of subsetGOGO3

Value

returns a list whose components are c("DB","table","ngenes") where 'DB' is the GO DB subsetted to the desired ONTOLOGY, and 'table' is tabulation of number of occurrences of each GO category name within the desired ONTOLOGY, and ngenes is the total number of hgncList genes mapping to GOGO3

Examples

```
## Not run:
# GOGOA3.RData is too large to include in the R package
# so I need to load it from a file that is not in the package.
# Since this is in a file in my own file system, I could not
# include this as a regular example in the package.
# you can generate it using the package 'minimalistGODB'
# or you can retrieve it from https://github.com/barryzee/GO/tree/main/databases
load("/Users/barryzeeberg/personal/GODB_RDATA/goa_human/GOGOA3_goa_human.RData")
x<-G




```

G resh |*G resh |*

Description

retrieve lines of *m* that meet both *enrichThresh* and *countThresh*

Usage

```
G resh |(m, sampleFDR, enrichThresh, countThresh, pvalThresh, fdrThresh)
```

Arguments

<i>m</i>	return value of <code>GOenrich3()</code>
<i>sampleFDR</i>	component of return value of <code>RCPD()</code>
<i>enrichThresh</i>	numerical acceptance threshold for enrichment
<i>countThresh</i>	numerical acceptance threshold for gene count
<i>pvalThresh</i>	numerical acceptance threshold for <i>pval</i>
<i>fdrThresh</i>	numerical acceptance threshold for <i>fdr</i>

Value

returns a subset of matrix (*m* joined with *fdr*\$*sampleFDR*) with entries meeting all thresholds

Examples

```
thresh<-G resh |(x_m,x_fdr$sampleFDR,enrichThresh=2,countThresh=2,pvalThresh=0.1,fdrThresh=0.100)
```

HCCS66	<i>GoMiner data set</i>
--------	-------------------------

Description

GoMiner data set

Usage

data(HCCS66)

hitterBeforeAfterDriver	<i>hitterBeforeAfterDriver</i>
-------------------------	--------------------------------

Description

driver to invoke hitters2() and trimGOGOA3()

Usage

hitterBeforeAfterDriver(GOGOA3, mn = 20, mx = 200, verbose)

Arguments

GOGOA3	return value of minimalistGODB::buildGODatabase()
mn	integer minimum category size
mx	integer maximum category size
verbose	integer vector representing classes

Value

returns the return value of trimGOGOA3()

Examples

```
## Not run:
# GOGOA3.RData is too large to include in the R package
# so I need to load it from a file that is not in the package.
# Since this is in a file in my own file system, I could not
# include this as a regular example in the package.
# This example is given in full detail in the package vignette.
# You can generate GOGOA3.RData using the package 'minimalistGODB'
# or you can retrieve it from https://github.com/barryzee/GO
dir<-"/Users/barryzeeberg/personal/GODB_RDATA/goa_human/"
load(sprintf("%s/%s", dir, "GOGOA3_goa_human.RData"))
```

```

geneList<-GOGO3$ontologies[["biological_process"]][1:10,"HGNC"]
GOGO3tr<-hitterBeforeAfterDriver(GOGO3,mn=20,mx=200,1)

## End(Not run)

```

hitters2

hitters2

Description

determine the number of mappings for the top several genes

Usage

```
hitters2(GOGO3, verbose = 1)
```

Arguments

GOGO3	return value of <code>minimalistGODB::buildGODatabase()</code>
verbose	integer vector representing classes

Value

returns no value, but has side effect of printing information

Examples

```

## Not run:
# GOGO3.RData is too large to include in the R package
# so I need to load it from a file that is not in the package.
# Since this is in a file in my own file system, I could not
# include this as a regular example in the package.
# This example is given in full detail in the package vignette.
# You can generate GOGO3.RData using the package 'minimalistGODB'
# or you can retrieve it from https://github.com/barryzee/GO
dir<-"/Users/barryzeeberg/personal/GODB_RDATA/goa_human/"
load(sprintf("%s/%s",dir,"GOGO3_goa_human.RData"))
geneList<-GOGO3$ontologies[["biological_process"]][1:10,"HGNC"]
hitters2(GOGO3,1)

## End(Not run)

```

Housekeeping_Genes	<i>GoMiner data set</i>
--------------------	-------------------------

Description

GoMiner data set

Usage

```
data(Housekeeping_Genes)
```

human	<i>human</i>
-------	--------------

Description

determine if database represents human species

Usage

```
human(GOGO3, verbose = TRUE)
```

Arguments

GOGO3	return value of subsetGOGO3()
verbose	integer vector representing classes

Value

returns Boolean TRUE if species is human

Examples

```
## Not run:  
# GOGO3.RData is too large to include in the R package  
# so I need to load it from a file that is not in the package.  
# Since this is in a file in my own file system, I could not  
# include this as a regular example in the package.  
# you can generate it using the package 'minimalistGODB'  
# or you can retrieve it from https://github.com/barryzee/GO/tree/main/databases  
load("/Users/barryzeeberg/personal/GODB_RDATA/goa_human/GOGO3_goa_human.RData")  
hum<-human(GOGO3)  
  
load("/Users/barryzeeberg/personal/GODB_RDATA/sgd/GOGO3_sgd.RData")  
hum<-human(XENOPUS,1)  
  
## End(Not run)
```

preprocessDB

*preprocessDB***Description**

driver to perform several preprocessing steps: quick peek trim small and large categories is the database for human species validate validated HGNC symbols in sampleList determine up to date (ie, contains GOGOA3\$species) or legacy version of human database

Usage

```
preprocessDB(sampleList, GOGOA3, ontology, mn, mx, thresh, verbose)
```

Arguments

sampleList	character list of gene names
GOGOA3	return value of subsetGOGOA()
ontology	character string c("molecular_function", "cellular_component", "biological_process")
mn	integer param passed to trimGOGOA3, min size threshold for a category
mx	integer param passed to trimGOGOA3, max size threshold for a category
thresh	numerical paramter passed to checkGeneListVsDB()
verbose	integer vector representing classes

Value

returns a list whose components are a trimmed version of GOGOA3 and (for human) a sampleList with validated HGNC symbols

Examples

```
## Not run:
# GOGOA3.RData is too large to include in the R package
# so I need to load it from a file that is not in the package.
# Since this is in a file in my own file system, I could not
# include this as a regular example in the package.
# you can generate it using the package 'minimalistGODB'
# or you can retrieve it from https://github.com/barryzee/GO/tree/main/databases
load("/Users/barryzeeberg/personal/GODB_RDATA/goa_human/GOGOA3_goa_human.RData")
pp<-preprocessDB(cluster52,GOGOA3,"biological_process",20,200,0.5,3)

## End(Not run)
```

randSubsetGeneList	<i>randSubsetGeneList</i>
--------------------	---------------------------

Description

retrieve n unique random genes

Usage

```
randSubsetGeneList(geneList, ngenes)
```

Arguments

geneList	character vector geneList
ngenes	integer desired number of random genes

Value

returns a character vector of genes

Examples

```
## Not run:
# GOGO3.RData is too large to include in the R package
# so I need to load it from a file that is not in the package.
# Since this is in a file in my own file system, I could not
# include this as a regular example in the package.
# you can generate it using the package 'minimalistGODB'
# or you can retrieve it from https://github.com/barryzee/GO/tree/main/databases
load("/Users/barryzeeberg/personal/GODB_RDATA/goa_human/GOGO3_goa_human.RData")
genes<-randSubsetGeneList(GOGO3$genes[["biological_process"]],20)

## End(Not run)
```

RCPD	<i>RCPD</i>
------	-------------

Description

prepare a cpd of p values from randomized gene sets

Usage

```
RCPD(GOGO3, tablePop, geneList, nrand, ontology, hyper, subd, opt)
```

Arguments

GOGOA3	return value of subsetGOGOA()
tablePop	return value of GOfable3()
geneList	character vector list of genes to randomize
nrand	integer number of randomizations
ontology	c("molecular_function","cellular_component","biological_process")
hyper	return value of GOhypergeometric3() from real (nonrandom) data
subd	character string pathname for directory containing sink.txt
opt	integer 0:1 parameter used to select randomization method

Details

the cpd of the randomizations is to be used for estimating the false discovery rate (FDR) of the real sampled genes

Value

returns a histogram of $\log_{10}(p)$

Examples

```
## Not run:
# GOGOA3.RData is too large to include in the R package
# so I need to load it from a file that is not in the package.
# Since this is in a file in my own file system, I could not
# include this as a regular example in the package.
# you can generate it using the package 'minimalistGODB'
# or you can retrieve it from https://github.com/barryzee/GO/tree/main/databases
load("/Users/barryzeeberg/personal/GODB_RDATA/goa_human/GOGOA3_goa_human.RData")
rcpd<-RCPD(GOGOA3,x_tablePop31,10,3,"biological_process",x_hyper1,tempdir(),0)

## End(Not run)
```

runGoMinerExamples *runGoMinerExamples*

Description

driver to run GoMiner under several randomization procedures

Usage

```
runGoMinerExamples(  
  title = NULL,  
  dir,  
  sampleList,  
  GOGO3,  
  ontology,  
  enrichThresh = 2,  
  countThresh = 5,  
  pvalThresh = 0.1,  
  fdrThresh = 0.1,  
  nrand = 2,  
  mn = 2,  
  mx = 200,  
  verbose = 1  
)
```

Arguments

title	character string descriptive title
dir	character string full pathname to the directory acting result repository
sampleList	character list of gene names
GOGO3	return value of subsetGOGO3()
ontology	character string c("molecular_function", "cellular_component", "biological_process")
enrichThresh	numerical acceptance threshold for enrichment
countThresh	numerical acceptance threshold for gene count
pvalThresh	numerical acceptance threshold for pval
fdrThresh	numerical acceptance threshold for fdr
nrand	numeric number of randomizations to compute FDR
mn	integer param passed to trimGOGO3, min size threshold for a category
mx	integer param passed to trimGOGO3, max size threshold for a category
verbose	integer vector representing classes

Value

returns a list containing the return value of GoMiner()

`trimGOGOA3`*trimGOGOA3*

Description

remove categories from GOGOA3 that are too small or too large

Usage

```
trimGOGOA3(GOGOA3, mn, mx, verbose)
```

Arguments

GOGOA3	return value of subsetGOGOA()
mn	integer min size threshold for a category
mx	integer max size threshold for a category
verbose	integer vector representing classes

Details

If a category is too small, it is unreliable for statistical evaluation. Also, in the extreme case of size = 1, then that category is essentially equivalent to a gene rather than a category. Same is partially true for size = 2. If a category is too large, it is too generic to be useful for categorization. Finally, by trimming the database, analyses will run faster.

Value

returns trimmed version of GOGOA3

Examples

```
## Not run:  
# GOGOA3.RData is too large to include in the R package  
# so I need to load it from a file that is not in the package.  
# Since this is in a file in my own file system, I could not  
# include this as a regular example in the package.  
# This example is given in full detail in the package vignette.  
# You can generate GOGOA3.RData using the package 'minimalistGODB'  
# or you can retrieve it from https://github.com/barryzee/GO/tree/main/databases  
  
GOG03tr<-trimGOGOA3(GOGOA3, mn=2, mx=200, 1)  
  
## End(Not run)
```

validHGNCsymbols	<i>validHGNCsymbols</i>
------------------	-------------------------

Description

convert outdated HGNC symbols to current HGNC symbols

Usage

```
validHGNCsymbols(geneList)
```

Arguments

geneList character vector of HGNC symbols

Details

removes NA and /// from output of checkGeneSymbols()

Value

returns list of mapping table and vector of current HGNC symbols

Examples

```
geneList<-c("FN1", "tp53", "UNKNOWNGENE", "7-Sep",  
"9/7", "1-Mar", "Oct4", "4-Oct", "OCT4-PG4", "C19ORF71",  
"C19orf71")  
l<-validHGNCsymbols(geneList)
```

x_fdr	<i>GoMiner data set</i>
-------	-------------------------

Description

GoMiner data set

Usage

```
data(x_fdr)
```

x_hyper1	<i>GoMiner data set</i>
----------	-------------------------

Description

GoMiner data set

Usage

data(x_hyper1)

x_m	<i>GoMiner data set</i>
-----	-------------------------

Description

GoMiner data set

Usage

data(x_m)

x_sampleList1	<i>GoMiner data set</i>
---------------	-------------------------

Description

GoMiner data set

Usage

data(x_sampleList1)

x_tablePop3	<i>GoMiner data set</i>
-------------	-------------------------

Description

GoMiner data set

Usage

data(x_tablePop3)

<code>x_tablePop31</code>	<i>GoMiner data set</i>
---------------------------	-------------------------

Description

GoMiner data set

Usage

`data(x_tablePop31)`

<code>x_tableSample3</code>	<i>GoMiner data set</i>
-----------------------------	-------------------------

Description

GoMiner data set

Usage

`data(x_tableSample3)`

<code>x_thresh</code>	<i>GoMiner data set</i>
-----------------------	-------------------------

Description

GoMiner data set

Usage

`data(x_thresh)`

Index

checkGeneListVsDB, 2
cluster52, 3

FDR, 4

GOenrich3, 5
GOGOA3small, 5
GOheatmap, 6
GOhypergeometric3, 7
GoMiner, 7
GObtable3, 9
GOthresh, 10

HCCS66, 11
hitterBeforeAfterDriver, 11
hitters2, 12
Housekeeping_Genes, 13
human, 13

preprocessDB, 14

randSubsetGeneList, 15
RCPD, 15
runGoMinerExamples, 16

trimGOGOA3, 18

validHGNCsymbols, 19

x_fdr, 19
x_hyper1, 20
x_m, 20
x_sampleList1, 20
x_tablePop3, 20
x_tablePop31, 21
x_tableSample3, 21
x_thresh, 21