Package ‘goTools’

April 25, 2017

Version 1.50.0
Date 2009-10-23
Title Functions for Gene Ontology database
Depends GO.db
Imports AnnotationDbi, GO.db, graphics, grDevices
Suggests hgu133a.db
biocViews Microarray, GO, Visualization
Author Yee Hwa (Jean) Yang <jean@biostat.ucsf.edu>, Agnes Paquet <paquetagnes@yahoo.com>
Maintainer Agnes Paquet <paquetagnes@yahoo.com>
Description Wrapper functions for description/comparison of oligo ID list using Gene Ontology database
License GPL-2
LazyLoad yes
NeedsCompilation no

R topics documented:

EndNodeList .................................................. 1
goTools ......................................................... 2
Internal functions ............................................. 4
probeID .......................................................... 4

Index 5

EndNodeList

Reference GO nodes list.

Description

The function EndNodeList builds the default end node list used in ontoCompare. CustomEndNodeList helps you build a list of children nodes starting from a GO id of interest.
Usage

EndNodeList()
CustomEndNodeList(id, rank=1)

Arguments

id Valid GO id: "GO:XXXXXXXX".
rank Number of levels of children of the GO DAG under id you want to add to your nodes list.

Value

EndNodeList returns a vector of all GO ids 1 level below MF("GO:0003674"), BP("GO:0008150") and CC("GO:0005575"). MF, BP and CC nodes are included. CustomEndNodeList returns a vector of all GO ids children of id, rank levels below it.

Author(s)

Yee Hwa (Jean) Yang, Agnes Paquet

See Also

ontoCompare

Examples

## Examples use the probeID dataset. For description type ? probeID.
## library(GO.db)
## EndNodeList()
## MFendnode <- CustomEndNodeList("GO:0003674", rank=2)

## Example (not run)
## data(probeID)
## ontoCompare(affylist, probeType="hgu133a", endnode=MFendnode, goType="MF")

---

goTools  Wrapper functions

Description

This functions will allow you to describe and compare sets of oligo ids using Gene Ontology database

Usage

ontoCompare(genelist, probeType=c("GO", "hgu133a"),
goType="All", endnode, method=c("TGenes", "TIDS", "none"), plot=FALSE, ...)

ontoPlot(objM, names.arg=NULL, beside=TRUE, las=2, legend.text=TRUE, ...)
Arguments

`genelist` list of list of valid probe ids.

`method` method used to evaluate the percentage of oligos for each end-node. 'TGenes' = for each end node, return the number of direct children found / total number of probe ids. (default). This includes oligos which do not have GO annotations. 'TIDS' = for each end node, return the number of direct children found / total number of GO ids describing the list. 'none' = for each end node, return the number of direct children found.

`probeType` type of input given to the function. Valid probe types include GO ids and any probes ids for which a BioC annotation package providing a mapping to GO is available. ontoCompare is expecting valid probe ids.

`goType` help sort the data by type. If 'All' (default), all oligos are taken into account. 'BP' restricts information to Biological Process, 'CC' to Cellular Component, and 'MF' to Molecular Function.

`plot` logical: if 'TRUE', results are output as a graph.

`endnode` list of GO ids corresponding to end-nodes of interest.

`beside` Logical. If 'TRUE', the bars of the barplot are portrayed as juxtaposed bars. See ?barplot for more details.

`las` numeric: if las=2, the axis labels are displayed perpendicular to the axis. See ?par for more details.

`legend.text` vector of text used to construct a legend for the plot. See ?barplot for more details.

`objM` results from ontoCompare.

`names.arg` Labels to use in ontoPlot.

`...` extra layout parameters to be passed to ontoPlot.

Value

Returns the percentage of probes children of nodes contained in endnode. If 'plot' = TRUE, results are plotted as a pie chart or a bargraph.

Author(s)

Yee Hwa (Jean) Yang, Agnes Paquet

Examples

```
# Examples use the probeID dataset. For description type ?probeID.
# Not run

#library(GO.db)
data(probeID)
#ontoCompare(affylist, probeType="hgu133a", plot=TRUE)
#res <- ontoCompare(operonlist["L1"], probeType="operon", method="TIDS")
#ontoPlot(res, cex=0.7)
```
**probeID**

<table>
<thead>
<tr>
<th>Internal functions</th>
<th>Internal goTools functions</th>
</tr>
</thead>
</table>

**Description**

Internal goTools functions

**Details**

These are not to be called by the user.

<table>
<thead>
<tr>
<th>probeID</th>
<th>List of probe ids from Affymetrix hgu133a chip and Operon Version 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Human oligos</td>
</tr>
</tbody>
</table>

**Description**

The probeID dataset consists of two lists of randomly chosen probe IDs. `affylist` contains 3 sets of ids from Affymetrix hgu133a. `operonlist` contains 2 sets of ids from Operon Version 2.

**Usage**

data(probeID)
Index

*Topic **datasets**
  probeID, 4

*Topic **file**
  EndNodeList, 1
gTools, 2

*Topic **hplot**
gTools, 2

*Topic **manip**
  EndNodeList, 1

*Topic **methods**
  Internal functions, 4

affylist (probeID), 4

CustomEndNodeList (EndNodeList), 1

EndNodeList, 1

ggetGOID (Internal functions), 4
ggetOntology (Internal functions), 4
ggoChildren (Internal functions), 4
ggoParents (Internal functions), 4
ggTools, 2

Internal functions, 4
isEndNode (Internal functions), 4

ontoCompare, 1, 2
ontoCompare (gTools), 2
ontoCompare.main (Internal functions), 4
ontoPlot (gTools), 2
operonlist (probeID), 4

parentsVectWraper (Internal functions),
  4

probeID, 4