

# Package ‘GOpro’

April 16, 2025

**Type** Package

**Title** Find the most characteristic gene ontology terms for groups of human genes

**Version** 1.34.0

**Author** Lidia Chrabaszcz

**Maintainer** Lidia Chrabaszcz <chrabaszcz.lidia@gmail.com>

**Description** Find the most characteristic gene ontology terms for groups of human genes.  
This package was created as a part of the thesis which was developed under the auspices of MI<sup>2</sup> Group (<http://mi2.mini.pw.edu.pl/>, <https://github.com/geneticsMiNIng>).

**License** GPL-3

**Depends** R (>= 3.4)

**Imports** AnnotationDbi, dendextend, doParallel, foreach, parallel, org.Hs.eg.db, GO.db, Rcpp, stats, graphics, MultiAssayExperiment, IRanges, S4Vectors

**LinkingTo** Rcpp, BH

**RoxygenNote** 6.0.1

**Suggests** knitr, rmarkdown, RTCGA.PANCAN12, BiocStyle, testthat

**VignetteBuilder** knitr

**LazyData** true

**biocViews** Annotation, Clustering, GO, GeneExpression, GeneSetEnrichment, MultipleComparison

**URL** <https://github.com/mi2-warsaw/GOpro>

**BugReports** <https://github.com/mi2-warsaw/GOpro/issues>

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

**git\_branch** RELEASE\_3\_21

**git\_last\_commit** 8c40a18

**git\_last\_commit\_date** 2025-04-15

**Repository** Bioconductor 3.21

**Date/Publication** 2025-04-15

Contents

exrtcga . . . . .	2
findGO . . . . .	2
GOpro . . . . .	4
<b>Index</b>	<b>5</b>

---

exrtcga	<i>Expressions of human genes.</i>
---------	------------------------------------

---

Description

A dataset containing gene expressions of 300 human genes randomly chosen from a list returned by function prepareData(RTCGA = TRUE, cohorts = c('leukemia', 'colon', 'bladder')).

Usage

exrtcga

Format

A MultiAssayExperiment object of 3 listed experiments with user-defined names and respective classes. Containing an ExperimentList class object of length 3: [1] leukemia: matrix with 300 rows and 173 columns [2] colon: matrix with 300 rows and 190 columns [3] bladder: matrix with 300 rows and 122 columns.

Value

data

---

findGO	<i>Find top Gene Ontology terms for given genes.</i>
--------	--

---

Description

Find top Gene Ontology terms for given genes.

Usage

```
findGO(groups, topAOV = 50, sig.levelAOV = 0.05, parallel = FALSE,
grouped = "tukey", sig.levelGO = 0.05, minGO = 5, maxGO = 500,
clust.metric = NULL, clust.method = NULL, dist.matrix = NULL,
topGO = 3, sig.levelTUK = 0.05, onto = c("MF", "BP", "CC"),
extend = FALSE, over.rep = FALSE)
```

**Arguments**

<code>groups</code>	a <code>MultiAssayExperiment</code> object containing an <code>ExperimentList</code> class object representing gene expressions for at least 3 cohorts. Rows must be named with genes' aliases. The order of samples and genes has to be the same for each <code>ExperimentList</code> class object.
<code>topAOV</code>	A numeric value, a number of most significantly differentiated genes to be returned.
<code>sig.levelAOV</code>	a numeric value, a significance level used in BH correction for multiple testing ( <code>aovTopTest</code> ).
<code>parallel</code>	A logical value indicating if a task should be run on more than one core.
<code>grouped</code>	A method of grouping genes, one of 'tukey' and 'clustering'.
<code>sig.levelGO</code>	A numeric value, a significance level used in BH correction for multiple testing ( <code>findTopGOs</code> ).
<code>minGO</code>	A minimum number of functions that a gene needs to represent to be considered as frequent.
<code>maxGO</code>	A maximum number of functions that a gene needs to represent to be considered as frequent.
<code>clust.metric</code>	The method to calculate a distance measure used in hierarchical clustering, possible names: "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski".
<code>clust.method</code>	The agglomeration method used to cluster genes. This should be #'one of "ward.D", "ward.D2", "single", "complete", "average" (= UPGMA), "mcquitty" (= WPGMA), "median" (= WPGMC) or "centroid" (= UPGMC).
<code>dist.matrix</code>	A matrix with calculated distances to be used as a metric by <code>hclust</code> function.
<code>topGO</code>	A number of the most characteristic functions of groups of genes to be returned.
<code>sig.levelTUK</code>	A numeric value, a significance level used in Tukey's all pairwise comparison ( <code>groupByTukey</code> ).
<code>onto</code>	An ontology or ontologies to be searched for significant GO terms, at least one of 'MF' (molecular function), 'BP' (biological process), and 'CC' (cellular component).
<code>extend</code>	A logical value indicating if an extended version of the output should be presented.
<code>over.rep</code>	A logical value indicating if an over represented GO terms should be presented in the plot.

**Value**

A data frame containing the top gene ontology terms for each group of genes and the gene aliases.

**Examples**

```
findGO(exrtcga, grouped = 'clustering', topGO = 10, onto = 'MF')
findGO(exrtcga, grouped = 'tukey', topGO = 2, extend = TRUE)
```

---

GOpro*GOpro: find the most characteristic gene ontology terms for groups of genes*

---

**Description**

Based on the gene expressions find the structure somewhat comparable to gene signature. From all given genes, determine which are significantly different between sets. These sets may relate to different health conditions of patients, i.e. different types of cancer. Then divide interesting genes into subsets. Genes belong to a particular subset if they share the same feature. There are two implemented methods that can be used to create genes' subsets. The first method is so-called all pairwise comparisons by Tukey's procedure. Genes that have the same profile (a result of all comparisons) are assigned to one subset. The second way of determining subsets is a method of hierarchical clustering. When all genes are divided into subsets, then for each subset all relevant GO terms are searched for in org.Hs.eg.db database. Each found GO terms is tested using Fisher's test to find out which of them are the most characteristic for the given subset of genes.

**See Also**[findGO](#)

# Index

## \* **datasets**

exrtcga, [2](#)

exrtcga, [2](#)

findGO, [2](#), [4](#)

G0pro, [4](#)

G0pro-package (G0pro), [4](#)