Package 'mvGST'

April 15, 2017

Type Package

Title Multivariate and directional gene set testing

Version 1.8.0

Date 2014-10-02

Author John R. Stevens and Dennis S. Mecham

Maintainer John R. Stevens <john.r.stevens@usu.edu>

Description mvGST provides platform-independent tools to identify GO terms (gene sets) that are differentially active (up or down) in multiple contrasts of interest. Given a matrix of one-sided p-values (rows for genes, columns for contrasts), mvGST uses meta-analytic methods to combine p-values for all genes annotated to each gene set, and then classify each gene set as being significantly more active (1), less active (-1), or not significantly differentially active (0) in each contrast of interest. With multiple contrasts of interest, each gene set is assigned to a profile (across contrasts) of differential activity. Tools are also provided for visualizing (in a GO graph) the gene sets classified to a given profile.

Depends R(>= 2.10.0), GO.db, Rgraphviz

Imports gProfileR, stringr, topGO, GOstats, annotate, AnnotationDbi, graph

Suggests hgu133plus2.db, org.Hs.eg.db

License GPL-3

biocViews Microarray, OneChannel, RNASeq, DifferentialExpression, GO, Pathways, GeneSetEnrichment, GraphAndNetwork

NeedsCompilation no

R topics documented:

`mvGST-package` *Multivariate and directional gene set testing*

Description

mvGST provides platform-independent tools to identify GO terms (gene sets) that are differentially active (up or down) in multiple contrasts of interest. Given a matrix of one-sided p-values (rows for genes, columns for contrasts), mvGST uses meta-analytic methods to combine p-values for all genes annotated to each gene set, and then classify each gene set as being significantly more active (1), less active (-1), or not significantly differentially active (0) in each contrast of interest. With multiple contrasts of interest, each gene set is assigned to a profile (across contrasts) of differential activity. Tools are also provided for visualizing (in a GO graph) the gene sets classified to a given profile.

Details

Package:	mvGST
Type:	Package
Version:	0.99.3
Date:	2014-10-02
License:	GPL-3

To access the tutorial document for this package, type in R: `vignette("mvGST")`

User must provide a matrix a p-values with rows representing genes and columns representing contrasts that were tested. The contrasts must be given in the form var(1).var(2)...var(n-1).var(n) var(1) is the variable that will define the possible significance profiles. Each profile is a set of zeros, ones, and negative ones meaning significantly greater than, less than, or not significant. For example, if var(1) has levels a and b, the profile 1,0 would indicate significance (greater than) at level a and no significance at level b. The main result of `profileTable` is a matrix with significance profiles for row names and contrasts tested (not including var(1)) for column names and the total number of gene sets that fit each profile for each contrast in the cells.

`pickOut` returns the Gene Ontology ID's for the gene sets in a given cell of the `results.table` produced by `profileTable`.

`graphCell` uses the gene sets from `pickOut` to make a GO graph and displays the names of the gene sets in a legend.

`go2Profile` returns a matrix similar to the `results.table` for each desired gene set. The only difference is that there is only one gene set included in these matrices.

Author(s)

John R. Stevens, Dennis S. Mecham and Garrett Saunders

Maintainer: John R. Stevens <john.r.stevens@usu.edu>

References

Stevens, J. R., and Isom, S. C., 2012. "Gene set testing to characterize multivariately differentially expressed genes." Conference on Applied Statistics in Agriculture Proceedings, 24, pp. 125-137.

Mecham, D. S. (2014) "mvGST: Multivariate and Directional Gene Set Testing". MS Project, Utah State University, Department of Mathematics and Statistics. http://digitalcommons.usu.edu/gradreports/382/

Saunders, G., 2014. "Family-wise error rate control in QTL mapping and gene ontology graphs with remarks on family selection." PhD thesis, Utah State University, Department of Mathematics and Statistics. http://digitalcommons.usu.edu/etd/2164/

Examples

```
# See examples in help files of four main functions:
#   ?profileTable; ?pickOut; ?go2Profile; ?graphCell

# See package vignette for larger examples with discussion:
#    vignette("mvGST")
```

`go2Profile` *Creates tables showing profiles of specific gene sets.*

Description

Creates tables similar to the results tables of an `mvGST` object except for one gene set at a time. For each gene set selected, a table is produced with a single one and all zeroes in each column.

Usage

```
go2Profile(names, object)
```

Arguments

`names`	A character vector with the names, ID's, of the gene sets of interest. If the gene set names were not provided by the user, then this should be the GO ID's of the gene sets of interest.
`object`	A `mvGST` object with a final `results.table`.

Details

To access the tutorial document for this package, type in R: `vignette("mvGST")`

Value

A list of matrices. Each matrix has possible profiles as the row names and contrasts as the column names. Ones in the appropriate cells showing which profile the gene set fit for each contrast and zeroes elsewhere. The names of the list are the names, or ID's, provided.

Author(s)

John R. Stevens and Dennis S. Mecham

References

Stevens, J. R., and Isom, S. C., 2012. "Gene set testing to characterize multivariately differentially expressed genes." Conference on Applied Statistics in Agriculture Proceedings, 24, pp. 125-137.

Mecham, D. S. (2014) "mvGST: Multivariate and Directional Gene Set Testing". MS Project, Utah State University, Department of Mathematics and Statistics. http://digitalcommons.usu.edu/gradreports/382/

Examples

```
data(mvGSTsamples)

# object obatoclax.mvGST returned by profileTable

# Returns a list of 2 matrices: one matrix showing the significance profile for
# each contrast for the gene set GO:0000003, and one matrix for GO:0000019
go2Profile(c("GO:0000003", "GO:0000019"), obatoclax.mvGST)

# See package vignette for larger examples with discussion:
#    vignette("mvGST")
```

`graphCell` *Makes a GO graph highlighting the GO terms in the selected cell of the results table.*

Description

`graphCell` relies on tools in the Rgraphviz package. `graphCell` uses `pickOut` to get the GO terms in a specific cell of the results table. A GO graph is created from those GO terms, and can be interactive if desired. Also, if desired, a legend showing the names of the GO terms can be printed. If the graph is interactive, use esc to end interaction with graph.

Usage

```
graphCell(object, row, col = 1, set, ontology = "BP", interact = TRUE,
          legend.pos = "bottomleft", print.legend = TRUE,
  use.col="red", bg.col = "grey80")
```

Arguments

`object`	A `mvGST` object with a final `results.table`
`row`	The row of the desired cell.
`col`	The column of the desired cell. Column refers to the levels of Var2, if Var2 was used. It is the number of the column after the 1, 0, -1 columns that show the profiles. Default value is 1.
`set`	Optional argument that is a data frame with the first column containing the GO ID's that should be used to make the GO graph. The data frame returned by `pickOut` can be used.
`ontology`	The ontology, within Gene Ontology, that should be used ("BP", "MF", "CC").
`interact`	Indicates whether or not the graph should be interactive. If interactive, use esc to end interaction with graph.
`legend.pos`	If interactive, indicates the desired position of the legend that shows name and GO ID of selected node.
`print.legend`	Indicates if the legend should also be printed separately, showing GO names of all nodes.
`use.col`	Color to highlight the nodes representing gene sets of interest in the resulting graph.
`bg.col`	Color to use for the "background" in the graph when focusing on the gene sets of interest. This is the color used for the border of all nodes, the labels of all nodes NOT representing gene sets of interest, and all edges.

Details

To access the tutorial document for this package, type in R: `vignette("mvGST")`

Value

Invisibly returns `NULL`.

Author(s)

John R. Stevens and Dennis S. Mecham

References

Stevens, J. R., and Isom, S. C., 2012. "Gene set testing to characterize multivariately differentially expressed genes." Conference on Applied Statistics in Agriculture Proceedings, 24, pp. 125-137.

Mecham, D. S. (2014) "mvGST: Multivariate and Directional Gene Set Testing". MS Project, Utah State University, Department of Mathematics and Statistics. http://digitalcommons.usu.edu/gradreports/382/

Examples

```
data(mvGSTsamples)
# object obatoclax.mvGST returned by profileTable
obatoclax.mvGST

# plots a GO Graph highlighting the GO ID's from the cell
# in the fifth row and first column (the column for
# cell line RS4) of the results.table of the object
# returned by profileTable
graphCell(obatoclax.mvGST, 5, 1, ontology = "BP", interact = FALSE)

# See package vignette for larger examples with discussion:
#    vignette("mvGST")
```

`mvGST.other` *mvGST Other Functions*

Description

Internal functions used by the main functions of `mvGST` (`graphCell`, `profileTable`, `go2GeneSets`, `pickOut`, and `p.adjust.SFL`):

`changeTO10`	changes p-values in matrix to -1, 0, or 1, indicating significance.
`combinePvalues`	Uses Stouffer's method to combine p-values in gene sets
`convertPvalues`	converts p-values from one-sided to two-sided and vice versa
`cut`	cuts out any all 0 rows from the matrix of results
`fillInList`	fills in list of gene sets, ensuring that any gene in a child set is also in the parent set
`finalResults`	counts the number of gene sets corresponding to each profile and contrast and creates `res`
`geneNameConvertRows`	handles one-to-many and many-to-one problems of gene name translation
`generateGeneSets`	creates list of gene sets
`go2GeneSet`	creates a single matrix showing which profile a gene set fits into for each contrast tested
`hartung`	Combines a set of p-values using Hartung's modified inverse normal method

interactiveGraph	Creates a GO Graph of selected GO terms
method2	Uses Hartung's method to combine p-values and handles many-to-one problem of gene n
method3	Handles one-to-many problem of gene name translation by eliminating all but one of the
method4	Combines method2 and method3
mvGSTObject	Creates an mvGST object
mvSort	Sorts results.table by row totals
oneSideBYAdjust	Convert p-values to two-sided, then uses Benjamini-Yekutieli adjustment, and converts ba
print.mvGST	prints the results.table in a clean way
print.summary.mvGST	prints the summary of an mvGST object
profileCombine	Determines profiles of each gene set at each contrast
profiles	Defines the possible profiles
separate	splits genes (and their p-values) into gene sets
summary.mvGST	creates a summary of an mvGST object
tableColumns	Creates the column names for the results.table
distributeWeight	helper function for p.adjust.SFL
getCurrentChildren	helper function for p.adjust.SFL
makeCoherent	helper function for p.adjust.SFL
getAncestorsAndOffspring	helper function for p.adjust.SFL
turnListAround	From globaltest package, helper function for p.adjust.SFL

Details

To access the tutorial document for this package, type in R: `vignette("mvGST")`

Author(s)

John R. Stevens, Dennis S. Mecham, Garrett Saunders

References

Stevens, J. R., and Isom, S. C., 2012. "Gene set testing to characterize multivariately differentially expressed genes." Conference on Applied Statistics in Agriculture Proceedings, 24, pp. 125-137.

Mecham, D. S. (2014) "mvGST: Multivariate and Directional Gene Set Testing". MS Project, Utah State University, Department of Mathematics and Statistics. http://digitalcommons.usu.edu/gradreports/382/

Saunders, G., 2014. "Family-wise error rate control in QTL mapping and gene ontology graphs with remarks on family selection." PhD thesis, Utah State University, Department of Mathematics and Statistics. http://digitalcommons.usu.edu/etd/2164/

`mvGSTsamples` *mvGSTsamples: Sample objects for the mvGST package*

Description

This includes the following four objects:

obatoclax.pvals	a matrix with rows for p-values and columns for contrasts, based on GSE36149 data from NCBI GEO
parathyroid.pvals	a matrix with rows for p-values and columns for contrasts,

	based on parathyroidGenesSE data from *parathyroidSE* package
`obatoclax.mvGST`	object returned by `profileTable` run on `obatoclax.pvals`
`parathyroid.mvGST`	object returned by `profileTable` run on `parathyroid.pvals`

Usage

```
data(mvGSTsamples)
```

Format

This object contains the four objects described above.

Details

See *mvGST* package vignette for details on these objects (including how they were constructed): `vignette("mvGST")`

The intended use for these objects is to demonstrate the methods coded in the *mvGST* package.

Value

The name of the data set specified.

References

Love M., "parathyroidSE: SummarizedExperiment for RNA-Seq of primary culture parathyroid tumors by Haglund et al., J Clinical Endocrinol Metab 2012." R package version 1.2.0.

Urtishak K.A., Edwards A.Y., Wang L.S., Hudome A., et al. (2013), "Potent obatoclax cytotoxicity and activation of triple death mode killing across infant acute lymphoblastic leukemia," Blood 121(14):2689-703.

Examples

```
data(mvGSTsamples)
head(obatoclax.pvals)
head(parathyroid.pvals)
obatoclax.mvGST
parathyroid.mvGST

# See package vignette for discussion of these objects:
#    vignette("mvGST")
```

---

p.adjust.SFL *Short Focus Level adjustment*

Description

Takes a named numeric vector of raw p-values as input and returns the Short Focus Level adjusted p-values, where the adjustment is based on controlling the FWER at a specified level within the structure of the GO graph.

Usage

```
p.adjust.SFL(rawp, ontology=c("BP","CC","MF"),
    focus='rn', ancestors, offspring, trace=FALSE,
    recycle=TRUE, sig.level=0.05)
```

Arguments

rawp	named numeric vector of p-values where the names correspond to the GO ID for which the provided p-values correspond to the given GO Term. These must be 'two-sided' p-values, i.e., from a two-sided test.
ontology	The ontology of interest. Must be one of 'BP', 'CC', or 'MF'. All names of rawp must be from the same ontology. Defaults to 'BP'.
focus	the focus level of interest. Default is set at the root node, the logical place to start if there is no better place to start.
ancestors	named lists corresponding to the ancestor and offspring structure of all named GO IDs in rawp. These are optional, and simply result in faster computation of the adjusted p-values if they are already available from the R session when p.adjust.SFL is called.
offspring	named lists corresponding to the ancestor and offspring structure of all named GO IDs in rawp. These are optional, and simply result in faster computation of the adjusted p-values if they are already available from the R session when p.adjust.SFL is called.
trace	logical denoting whether or not progress about the algorithm is output to the user. Defaults to FALSE.
recycle	logical determining whether or not to recycle any threshold corresponding to a rejected leaf node back into the GO graph. Defaults to TRUE, as it can result in greater power for the Short Focus Level method at a slight increase to the computational burden.
sig.level	numeric value at which to control the family-wise error rate within the structure of the GO graph.

Details

To access the tutorial document for this package, type in R: `vignette("mvGST")`

Value

returns the adjusted p-values with naming and ordering identical to the original "rawp" values.

Author(s)

John R. Stevens and Garrett Saunders

References

Saunders G., Stevens J.R., and Isom S.C. "A shortcut for multiple testing on the directed acyclic graph of Gene Ontology." BMC Bioinformatics 2014 (under review).

Saunders, G., 2014. "Family-wise error rate control in QTL mapping and gene ontology graphs with remarks on family selection." PhD thesis, Utah State University, Department of Mathematics and Statistics. http://digitalcommons.usu.edu/etd/2164/

Examples

```
# Get GO terms of interest
library(GOstats); library(annotate)
GO.vec <- c("GO:0001775","GO:0007275")
g <- GOGraph(GO.vec, GOBPPARENTS)
g <- removeNode("all",g)
GOids <- names(nodes(g))

# Get p-values for all GO terms of interest
# (here, simulated for demonstration)
# Make sure names are GO term IDs
set.seed(1)
rawp <- rbeta(length(GOids), .2, 1)
names(rawp) <- GOids

# P-value adjustment using Short Focus Level
padj <- p.adjust.SFL(rawp, ontology='BP')
head(padj)
# These are in the same order as rawp, with
# names corresponding to GO terms.
# Calling GO terms significant when padj is
# less than alpha controls the FWER at alpha,
# within the context of the GO graph.

# See package vignette for larger examples with discussion:
#    vignette("mvGST")
```

---

pickOut *Returns gene sets in a cell of the* `results.table`.

Description

`pickOut` returns a character vector with the Gene Ontology ID's of the gene sets with a particular significance profile for a certain contrast (the gene sets in one cell of the `results.table` of an mvGST object).

Usage

```
pickOut(mvgst, row, col = 1)
```

Arguments

`mvgst`	A mvGST object. `mvGST$results.table` must not be NULL.
`row`	The row of the desired profile.
`col`	The column of the desire contrast. Column refers to the levels of Var2, if Var2 was used. It is the number of the column after the 1, 0, -1 columns that show the profiles. Default value is 1.

Details

To access the tutorial document for this package, type in R: `vignette("mvGST")`

Value

A data frame containing the ID's of the gene sets in the given row and column along with the GO descriptions of each gene set and the adjusted p-values for each contrast tested.

Author(s)

John R. Stevens and Dennis S. Mecham

References

Stevens, J. R., and Isom, S. C., 2012. "Gene set testing to characterize multivariately differentially expressed genes." Conference on Applied Statistics in Agriculture Proceedings, 24, pp. 125-137.

Mecham, D. S. (2014) "mvGST: Multivariate and Directional Gene Set Testing". MS Project, Utah State University, Department of Mathematics and Statistics. http://digitalcommons.usu.edu/gradreports/382/

Examples

```
data(mvGSTsamples)
# object obatoclax.mvGST returned by profileTable
obatoclax.mvGST

# returns the GO ID's from the cell in the third row and first column\
# (the column for cell line RS4) of the results.table of
# the object returned by profileTable
pickOut(obatoclax.mvGST, 3, 1)

# See package vignette for larger examples with discussion:
#    vignette("mvGST")
```

---

profileTable *Creates a table of significance profiles of gene sets from given matrix of p-values*

Description

`profileTable` takes a matrix of one-sided p-values with rows representing genes and columns representing contrasts. Rows (and p-values) are combined using Stouffer's method so that the new rows represent gene sets. P-values are then adjusted for multiple hypothesis testing using the Benjamini-Yekutieli adjustment and converted to 1 (p-value < alpha/2), -1 (alpha > 1-alpha/2) or 0 (not significant). Then each gene set is classified according to its significance profile (across one of the factors) for each of the remaining contrasts.

Usage

```
profileTable(pvals, gene.names, contrasts, list.groups, sig.level = 0.05,
             gene.ID, organism, affy.chip, ontology = "BP", method = 2,
             minsize = 1, maxsize = Inf, mult.adj = "BY")
```

Arguments

pvals	A matrix containing the one-sided p-values corresponding to the various genes (rows) and contrasts (columns)
gene.names	A character vector containing the gene names that correspond to the rows of the matrix of p-values. If NULL, then `gene.names <- rownames(pvals)`
contrasts	A character vector containing the contrasts that correspond to each column in the matirx of p-values. Must either be in format: Var1 or Var1.Var2 (Var2 is optional). The number of levels in Var1 determines the dimensions of the profiles (i.e. if Var1 has 2 levels, then there will be two columns for the profiles in the returned table). Var2 determines the number of columns, or strata, that will be reported in the returned table for each profile. If Var2 is not given, then there will only be one column reported, which will be the ontology chosen. If NULL, then `contrasts <- colnames(pvals)`
list.groups	An optional list containing user-defined gene sets
sig.level	The alpha level that should be used. Default is .05. This level is divided equally between the two sides of the test. So, for the default level of .05, any p-value less than .025 or greater than .975 is considered significant.
gene.ID	Gene naming system used for the gene names. Used to generate list of gene sets mapping genes to Gene Ontology sets. `gene.ID` can be "entrez", "genbank", "alias", "ensembl", "symbol", "genename", "unigene", or "affy" among others. (See `ID` argument in the `annFUN.org` function of the `topGO` package for supported levels.) If ID is all numeric and is not listed above, see http://biit.cs.ut.ee/gprofiler/gconvert.cgi for proper input.
organism	The organism that the genes come from. Used to generate list of gene sets mapping genes to Gene Ontology sets. The organism name should be the first letter of the scientific name and the second word of the scientific name, all lower case. For example, human would be "hsapiens". organism must be one of the following: agambiae, athaliana, btaurus, celegans, cfamiliaris, dmelanogaster, drerio, ecoliK12, ecoliSakai, ggallus, hsapiens, mmusculus, mmulatta, pfalciparum, ptrogldytes, rnorvegicus, scerevisiae, scoelicolor, sscrofa, tgondii, or xlaevis
affy.chip	The type of affy.chip used, if gene.ID == "affy". If `abatch` is an AffyBatch object, then use `affy.chip = annotation(abatch)`
ontology	The ontology that should be used for gene sets: "BP", "MF", or "CC". Default is "BP"
method	The method for handling gene name translation issues. Default is 2 (See "Details" below).
minsize	The minimum number of genes a gene set can have and still be included in the list of gene sets, if the list is not provided by the user.
maxsize	The maximum number of genes a gene set can have and still be included in the list of gene sets, if the list is not provided by the user.
mult.adj	The type of multiple hypothesis adjustment to make. BY is a Benjamini-Yekutieli adjustment. SFL is a Short Focus Level adjustment.

Details

User must provide a matrix a p-values with rows representing genes and columns representing contrasts that were tested. The contrasts must be given in the form Var1.Var2 or just Var1 (Var2 is optional). The possible significance profiles will be defined by Var1. The number of dimensions

of the profiles is the same as the number of levels of Var1. Each profile is a set of zeros, ones, and negative ones meaning significantly greater than, less than, or not significant. For example, if Var1 has levels a and b, the profile 1,0 would indicate significance (greater than) at level a and no significance at level b. The main result of `profileTable` is a matrix with significance profiles for row names and contrasts tested (not including Var1) for column names and the total number of gene sets that fit each profile for each contrast in the cells.

If the gene names given are not affy or entrez ID's, then they will have to be translated to entrez ID's if the user does not provide their own list of gene sets. Translation is done using `gconvert` from the `gProfileR` package. This may lead to one gene being translated to many, or many being translated to one. These problems are handled using a method of the user's choice (See Section 2.1.1 of Mecham, 2014): Method 1 does nothing. As a result, some rows of p-values will be duplicated when one name translates to many. Some rows will also have the same gene name when many names translate to just one. Method 2 uses Stouffer's inverse normal method to combine p-values when many names translate to just one. Method 3 accounts for when one name translates to many. Instead of duplicating rows of p-values, only the first of the new names is used. Method 4 combines methods 2 and 3. First method 2 is performed, then method 3.

To access the tutorial document for this package, type in R: `vignette("mvGST")`

Value

Returns an object of class "mvGST". An object of class "mvGST" is a list containing the following components:

`results.table` A matrix with possible profiles as row names and contrasts as column names. The cells of the matrix show how many gene sets have each profile for each contrast.

`raw.pvals` A matrix of the original p-values provided with gene names as row names and contrasts as column names.

`grouped.raw` A matrix of Stouffer combined p-values. Each row is for a gene set and each column is for a contrast.

`adjusted.group.pvals` The same matrix as in grouped.raw, but with a the requested adjustment ("BY" or "SFL") being performed within each column.

`ones.zeroes` A matrix showing the significance results of each of the BY adjusted p-values. 1 means significantly greater. -1 mean significantly less. 0 means not significant.

`ord.lev` The levels of the ordered variable (the variable that defines the profiles).

`contrasts` The contrasts that are the column names of $results.table.

`group.names` The Gene Ontology ID's of the gene sets in the selected ontology.

warning

If user does need to use gene ID's other than affy or entrez, it is strongly recommended that the user provides his or her own list of gene sets. Some of the translations that have been tested can take a VERY long time.

Author(s)

John R. Stevens and Dennis S. Mecham

References

Stevens, J. R., and Isom, S. C., 2012. "Gene set testing to characterize multivariately differentially expressed genes." Conference on Applied Statistics in Agriculture Proceedings, 24, pp. 125-137.

Mecham, D. S. (2014) "mvGST: Multivariate and Directional Gene Set Testing". MS Project, Utah State University, Department of Mathematics and Statistics. http://digitalcommons.usu.edu/gradreports/382/

Saunders, G., 2014. "Family-wise error rate control in QTL mapping and gene ontology graphs with remarks on family selection." PhD thesis, Utah State University, Department of Mathematics and Statistics. http://digitalcommons.usu.edu/etd/2164/

Examples

```
# A matrix of p-values with 3 rows (one for each gene) and 4 columns (one for
# each contrast)
pvals <- matrix(c(.001, .03, .7, .5, .01, .0002, .01, .85, .97, .99, 1, .98),
                nrow = 3)

# A character vector with the IDs of the genes (in this case, 3 Entrez IDs)
gene.names <- c("8510", "4361", "10111")
rownames(pvals) <- gene.names

# A character vector with the contrasts that were tested (the first factor
# had levels 1 and 2; the second factor had levels a and b; in each of the
# first factor's levels, second factor levels a and b were tested
# against a third level)
contrasts <- c("1.a", "1.b", "2.a", "2.b")
colnames(pvals) <- contrasts

# Creates an mvGST object with a results.table showing how many
# biological process (BP) gene sets fit each significance profile
# for each contrast.  Gene names are from Entrez, and data are human.
test <- profileTable(pvals, gene.ID = "entrez", organism = "hsapiens",
                     ontology = "BP")

# See how many gene sets were classified into each profile
# (across levels of the first factor) for each level of the
# second factor.  For example, the '0  0' line represents the
# profile of gene sets that are not differentially active in either
# level of the first factor.
test

# See package vignette for larger examples with discussion:
#    vignette("mvGST")
```

Index