Package 'mseapca'

February 20, 2015

Type Package

Title Metabolite set enrichment analysis for factor loading in principal component analysis

Version 1.0

Date 2012-04-10

Author Hiroyuki Yamamoto

Maintainer Hiroyuki Yamamoto <h.yama2396@gmail.com>

Description This package provides functions for metabolite set enrichment analysis (MSEA) and principal component analysis (PCA), and converting metabolite set list from your own csv files or KEGG's tar.gz files to XML documents. This package is suitable for computation of MSEA for factor loading in PCA.

License BSD

Depends XML

Repository CRAN

Date/Publication 2012-04-15 16:42:20

NeedsCompilation no

R topics documented:

csv2list *Convert metabolite set / csv to list*

Description

This function converts your own metabolite set (csv file to list).

Usage

```
csv2list(filepath)
```

Arguments

filepath	file path of metabolite set (csv file)

Details

The first row of csv file are "metabolite set name" and "metabolite IDs" as header. The first column must be metabolite IDs and second column must be metabolite set name.

Value

list of metabolite set name and metabolite IDs

Author(s)

Hiroyuki Yamamoto

Examples

```
# ----------------------------
#  Convert csv file to list
# ----------------------------
# filepath <- "C:/pathway.csv" # filepath of csv file
# N <- csv2list(filepath) # convert csv file to list
```

list2xml *Convert metabolite set / list to XML format*

Description

This function converts metabolite set (list to XML format).

Usage

```
list2xml(filepath, M)
```

Arguments

filepath	file path of metabolite set (original file)
M	list of metabolite set and metabolite IDs

Details

The "filepath" is only used for writing a original path in XML.

Value

XML format of metabolite set

Author(s)

Hiroyuki Yamamoto

Examples

```
# ---------------------------------
#  Convert KEGG's tar.gz to list
# ---------------------------------
# filepath <- "C:/hsa.tar.gz" # location of original files of metabolite set
# Z <- pathway_class(filepath) # making metabolite set list
# L <- list2xml(filepath, Z) # xml format

# ----------------------
#  Convert csv to list
# ----------------------
# filepath <- "C:/pathway.csv" # csv file
# Z <- csv2list(filepath) # csv file to list
# L <- list2xml(filepath, Z) # xml format

# ----------------------
#  Convert list to xml
# ----------------------
# savefile <- "kegg_test.xml" # set filename (XML)
# fullpath <- paste(dirname(filepath),savefile,sep="/") # fullpath of saved XML file
# saveXML(L,fullpath) # save XML
```

---

msea_ora *MSEA by over representation analysis*

Description

This function performs metabolite set enrichment analysis by over representation analysis (ORA). Statistical hypothesis test of cross tabulation is performed by one-sided Fisher's exact test.

Usage

```
msea_ora(SIG, ALL, M)
```

Arguments

SIG	Metabolite IDs of significant metabolites
ALL	Metabolite IDs of all detected metabolites
M	list of metabolite set name and metabolite IDs

Value

list of p-value and q-value for metabolite set and selected (significant) metabolite IDs for each metabolite set

Author(s)

Hiroyuki Yamamoto

References

Draghici S, Khatri P, Martins RP, Ostermeier GC, Krawetz SA. Global functional profiling of gene expression. Genomics. 2003 Feb;81(2):98-104.

Examples

```
# -----------------------
#  load metabolite set
# -----------------------
# filename <- "C:/pathway.xml"
# M <- read_pathway(filename)  # load metabolite set list

# ----------------------
#  Set metabolite IDs
# ----------------------
## p : dataframe of metabolite IDs and p-value of factor loadings in PCA
## fl : dataframe of metabolite IDs and factor loadings in PCA

# ALL <- p[,1] # All metabolite IDs
# SIG <- p[p[,2] < 0.05 & fl[,2]<0,1]
            ## negatively significant metabolites selected by factor loading

# ---------------
#  MSEA by ORA
# ---------------
# B <- msea_ora (SIG, ALL, M)
```

msea_sub *MSEA by Subramanian et al.*

Description

This function performs metabolite set enrichment analysis implemented in the same fashion as gene set enrichment analysis (Subramanian et al. 2005). In this function, a permutation procedure is performed for a metabolite set rather than class label. This procedure corresponds to a "gene set" of permutation type in GSEA-P software (Subramanian et al. 2007). A leading-edge subset analysis is also undertaken following the standard GSEA procedure.

Usage

```
msea_sub(M, D, y, maxiter = 1000)
```

Arguments

M	list of metbolite set name and metabolite IDs
D	data.frame(metabolite ID, data matix)
y	response variable (e.g. PC score)
maxiter	maximum number of iterations in random permutation (default=1000)

Value

list of normalized enrichment score, p-value and q-value for metabolite set, and the results of leading edge subset

Author(s)

Hiroyuki Yamamoto

References

Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S. L., Golub, T. R., Lander, E. S. & Mesirov, J. P. (2005) Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. Proc. Natl. Acad. Sci. USA 102, 15545-15550.

Subramanian, A., Kuehn, H., Gould, J., Tamayo, P., Mesirov, J.P. (2007) GSEA-P: A desktop application for Gene Set Enrichment Analysis. Bioinformatics, doi: 10.1093/bioinformatics/btm369.

Examples

```
# ------------------------
#  Set response variable
# ------------------------
## T : PC scores
# y <- T[,1];  # 1st PC score
```

```
# ------------------------------------------------------
#  Preparing metabolome data and metabolite set list
# ------------------------------------------------------
## M : metabolite set list, M_ID : metabolite ID
## Z : data matrix (metabolite IDs * samples)
# filename <- "C:/pathway.xml"
# M <- read_pathway(filename)  # load metabolite set list
# D <- data.frame(M_ID,Z)  # preparing dataframe

# --------
#  MSEA
# --------
# P <- msea_sub(M,D,y) # MSEA by Subramanian et al.
```

`pathway_class` *Convert metabolite set (multiple) / tar.gz to list*

Description

Convertion of KEGG's tar.gz files (e.g. hsa.tar.gz) to metabolite set list

Usage

```
pathway_class(filepath)
```

Arguments

`filepath`	file path of KEGG's tar.gz files

Details

The tar.gz files should be downloaded from KEGG FTP according to your own licence.

Value

list of metabolite set name and metabolite IDs

Author(s)

Hiroyuki Yamamoto

Examples

```
# filepath <- "C:/hsa.tar.gz" # *.tar.gz downloaded from KEGG
# Z <- pathway_class(filepath) # convert to list
```

pca_scaled *PCA for autoscaled data*

Description

This function performs principal component analysis (PCA). In this function, data matrix is automatically scaled to zero mean and unit variance (autoscaling) for each metabolites. PC scores, factor loadings and the results of statistical test (p-value and q-value by Benjamini and Hochberg) are returned. In this function, factor loading is defined as a correlation coefficient between PC score and each metabolite levels.

Usage

```
pca_scaled(D)
```

Arguments

D	dataframe of metabolite IDs and data matrix (metabolites * samples)

Details

Blank must be set to missing values in data matrix. If standard deviation of metabolite levels is equal to zero, it is removed from PCA computation and the results of factor loadings are set to NA.

Value

list of PC scores, factor loadings, p-value and q-value (by Benjamini and Hochberg), and contribution ratios

Author(s)

Hiroyuki Yamamoto

References

Benjamini, Yoav and Hochberg, Yosef (1995). "Controlling the false discovery rate: a practical and powerful approach to multiple testing". Journal of the Royal Statistical Society, Series B (Methodological) 57 (1): 289-300.

Examples

```
# --------------
#  Sample data
# --------------
X <- matrix(runif(1000),nrow=100,ncol=10) # 100(metabolites)*10(samples)
M <- as.character(c(1:100)) # metabolite IDs
D <- data.frame(M,X) # dataframe of metabolite IDs and data matrix

# --------------------------
```

```
#  PCA for autoscaled data
# ---------------------------
A <- pca_scaled(D)  # automatically scaled in pca_scaled function

# ---------
#  Result
# ---------
A["score"]] # PC score
A["factor.loading"]]  # factor loading
A["p.value"]]  # p-value of factor loading
A["q.value"]]  # q-value of factor loading
```

`read_pathway` *Read metabolite set file (*.xml)*

Description

This function generates metabolite set list from metabolite set file (XML). This is mainly used to be called by other functions.

Usage

```
read_pathway(fullpath)
```

Arguments

`fullpath`	file path of metabolite set (XML)

Value

list of metabolite set name and metabolite IDs.

Author(s)

Hiroyuki Yamamoto

Examples

```
# filename <- "C:/R/pathway.xml" # load metabolite set file
# M <- read_pathway(filename) # Convert XML to metabolite set (list)
```

setlabel *Generate binary label matrix of metabolite set*

Description

This function generates binary label matrix of metabolite IDs and metabolite sets. This is mainly used to be called by other functions, and used to count the number of metabolites in a specific metabolite set.

Usage

```
setlabel(M_ID, M)
```

Arguments

M_ID	metabolite IDs of detected metabolites in data
M	list of metabolite set name and metabolite IDs

Details

If single peak has multiple metabolite IDs in M_ID, split by "," or ";".

Value

binary label matrix of metabolite IDs in metabolite sets

Author(s)

Hiroyuki Yamamoto

Examples

```
## M_ID : metabolite IDs, M : metabolite set list
# L <- setlabel(M_ID,M) # binary label matrix
# colSums(L) # the number of metabolites in each metabolite set
```

Index