Package 'mseapca'

November 4, 2023

Type Package

Title Metabolite Set Enrichment Analysis for Loadings

Version 2.0.3

Date 2023-11-5

Author Hiroyuki Yamamoto

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

Description Computing metabolite set enrichment analysis (MSEA) (Yamamoto, H. et al. (2014) <doi:10.1186/1471-2105-15-51>) and single sample enrichment analysis (SSEA) (Yamamoto, H. (2023) <doi:10.51094/jxiv.262>).

License LGPL-3

Encoding UTF-8

LazyData true

Imports XML

Depends loadings

NeedsCompilation no

Repository CRAN

Date/Publication 2023-11-04 11:20:02 UTC

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

```
## Not run:
# ---------------------------
#  Convert csv file to list
# ---------------------------
filepath <- "C:/pathway.csv" # filepath of csv file
N <- csv2list(filepath) # convert csv file to list

## End(Not run)
```

`list2xml` *Save compound set as XML file*

Description

This function save compound set of list format as XML file.

Usage

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

Arguments

`filepath`	filepath of XML file to save
`M`	list fomat of compound set and compound names

Details

This function is used to store a compound set. Saved xml file can be read using the read_pathway function.

Value

filepath of saved XML file

Author(s)

Hiroyuki Yamamoto

Examples

```
## Not run:
data(pathway)
M <- pathway$fasting
xml_file <- "pathway_fasting.xml"
N <- list2xml(xml_file, M)
# XML::saveXML(N,filepath)

## End(Not run)
```

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

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

```
## Example1 : Metabolome data
data(fasting)
data(pathway)

# pca and pca loading
pca <- prcomp(fasting$X, scale=TRUE)
pca <- pca_loading(pca)

# all detected metabolites
metabolites <- colnames(fasting$X)

# statistically significant negatively correlated metabolites in PC1 loading
SIG <- metabolites[pca$loading$R[,1] < 0 & pca$loading$p.value[,1] < 0.05]
ALL <- metabolites #all detected metabolites
```

```
# metabolite set list
M <- pathway$fasting

# MSEA by over representation analysis
B <- msea_ora(SIG, ALL, M)
B$`Result of MSEA(ORA)`

## Example2 : Proteome data
data(covid19)
data(pathway)

X <- covid19$X$proteomics
Y <- covid19$Y
D <- covid19$D
tau <- covid19$tau

protein_name <- colnames(X)

# pls-rog and pls-rog loading
plsrog <- pls_rog(X,Y,D)
plsrog <- plsrog_loading(plsrog)

# statistically significant proteins
index_prot <- which(plsrog$loading$R[,1]>0 & plsrog$loading$p.value[,1]<0.05)
sig_prot <- protein_name[index_prot]

# detected proteins
protein_name <- colnames(X)

# protein set list
M <- pathway$covid19$proteomics

# MSEA by over representation analysis
B <- msea_ora(sig_prot, protein_name, M)
B$`Result of MSEA(ORA)`
```

`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

```
data(fasting)
data(pathway)

# pca and pca loading
pca <- prcomp(fasting$X, scale=TRUE)
pca <- pca_loading(pca)

# all detected metabolites
metabolites <- colnames(fasting$X)

# statistically significant negatively correlated metabolites in PC1 loading
SIG <- metabolites[pca$loading$R[,1] < 0 & pca$loading$p.value[,1] < 0.05]
ALL <- metabolites #all detected metabolites

# Set response variable
y <- pca$x[,1]

# preparing dataframe
D <- data.frame(ALL,t(fasting$X))  # preparing dataframe
```

```
# MSEA by Subramanian et al.
M <- pathway$fasting
P <- msea_sub(M,D,y, maxiter = 10) # iteration was set ato 10 for demonstration
```

---

pathbank2list *Generate metabolite set list from PathBank database*

Description

This function generates metabolite set list of PathBank database by referencing the AHPathbankDbs Bioconductor package.

Usage

```
pathbank2list(tbl_pathbank, subject, id)
```

Arguments

tbl_pathbank	tibble from AHPathbankDbs
subject	Pathway subject (Metabolic, Disease, etc.) in tibble
id	database ID (HMDB ID, Uniprot ID, etc.) used for analysis

Details

AHPathbankDbs needs to be installed separately.

Value

list of metabolite or protein set

Author(s)

Hiroyuki Yamamoto

Examples

```
## Not run:
## PathBank
library(AnnotationHub)

ah <- AnnotationHub()
qr <- query(ah, c("pathbank", "Homo sapiens"))

#tbl_pathbank <- qr[[1]] # metabolomics
tbl_pathbank <- qr[[2]] # proteomics

ids <- names(tbl_pathbank)[-c(1:4)]
```

```
id <- ids[1] # Uniprot ID

subs <- unique(tbl_pathbank$`Pathway Subject`)
subject <- subs[6] # Protein

M <- pathbank2list(tbl_pathbank, subject, id)

## End(Not run)
```

---

pathway *Example dataset for fasting and covid19 datasets*

Description

This data includes a metabolite set list and metabolite name list for fasting, and a metabolite set list for covid19 dataset within the "loadings" package

Usage

```
data(pathway)
```

Arguments

The list object pathway contains the following elements:

fasting : metabolite set list for fasting mouse dataset

data$fasting : metabolite name list for fasting mouse dataset

covid19$proteomics : protein set list for covid19 dataset.

References

Yamamoto H., Fujimori T., Sato H., Ishikawa G., Kami K., Ohashi Y. (2014). "Statistical hypothesis testing of factor loading in principal component analysis and its application to metabolite set enrichment analysis". BMC Bioinformatics, (2014) 15(1):51.

B. Shen, et al, Proteomic and Metabolomic Characterization of COVID-19 Patient Sera, Cell. 182 (2020) 59-72.e15.

Examples

```
data(pathway)
```

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

```
## Not run:
filename <- "C:/R/pathway.xml" # load metabolite set file
M <- read_pathway(filename) # Convert XML to metabolite set (list)

## End(Not run)
```

setlabel *Generate binary label matrix of metabolite set*

Description

This function generates binary label matrix of metabolite names 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	detected metabolites
M	list of metabolite set and metabolite names

Details

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

Value

binary label matrix of metabolite names in metabolite sets

Author(s)

Hiroyuki Yamamoto

Examples

```
data(fasting)
data(pathway)

M_ID <- colnames(fasting$X) # detected metabolites
M <- pathway$fasting # metabolite set list

L <- setlabel(M_ID,M) # binary label matrix
```

---

ssea_ora *Single sample enrichment analysis by over representation analysis*

Description

This function performs single sample enrichment analysis (SSEA) by over representation analysis (ORA). SSEA performs MSEA by ORA between detected and not detected metabolites in each sample."

Usage

```
ssea_ora(det_list, det_all, M)
```

Arguments

det_list	metabolite names of detected metabolites
det_all	metabolite names of all metabolites
M	list of metabolite set and metabolite names

Details

The threshold for determining whether a metabolite is detected or not is typically set by the signal-to-noise (S/N) ratio. If the S/N ratio is unavailable, one might consider using the signal intensity or peak area for each metabolite as an alternative. In such cases, all values below the threshold can be set to 0.

Value

A matrix where each row represents a sample and each column represents a set of metabolites.

Author(s)

Hiroyuki Yamamoto

References

Yamamoto H., Single sample enrichment analysisfor mass spectrometry-based omics data, Jxiv.(2023)

Examples

```
## Not run:
data(fasting)
data(pathway)

det_list <- pathway$data$fasting
M <- pathway$fasting
det_all <- unique(c(colnames(fasting$X), as.character(unlist(M))))

# SSEA
Z <- ssea_ora(det_list, det_all, M)

## PCA for SSEA score
pca <- prcomp(Z, scale=TRUE)
pca <- pca_loading(pca)

## End(Not run)
```

Index