Package 'PEIMAN2'

April 8, 2025

Title Post-Translational Modification Enrichment, Integration, and Matching Analysis

Version 1.0.0

Description Functions and mined database from 'UniProt' focusing on post-translational modifications to do single enrichment analysis (SEA) and protein set enrichment analysis (PSEA). Payman Nickchi, Uladzislau Vadadokhau, Mehdi Mirzaie, Marc Baumann, Amir Ata Saei, Mohieddin Jafari (2025) <doi:10.1002/pmic.202400238>.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

VignetteBuilder knitr

Depends R (>= 2.10)

Imports ggplot2, dplyr, glue, lifecycle, purrr, rlang, stringr, graphics, forcats, stats, magrittr

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

LazyData true

NeedsCompilation no

Author Mohieddin Jafari [aut], Payman Nickchi [aut, cre]

Maintainer Payman Nickchi <payman.nickchi@gmail.com>

Repository CRAN

Date/Publication 2025-04-08 07:00:02 UTC

Contents

`exmplData1` *Example dataset1*

Description

A dataset with randomly selected proteins from UniProt.

Usage

```
exmplData1
```

Format

A list with 2 elements:

pl1 97 randomly selected Homo sapiens (Human) proteins randomly selected from UniProt.

pl2 45 randomly selected Homo sapiens (Human) proteins randomly selected from UniProt. ...

Source

https://www.uniprot.org/

`exmplData2` *Example dataset 2*

Description

Proteins of rat hippocampus proteome.

Usage

```
exmplData2
```

Format

A dataframe with 209 rows and 2 columns.

UniProtAC UniProt accession code of proteins

Score Check with MJ ...

Source

https://pubmed.ncbi.nlm.nih.gov/33632781/

---

getTaxonomyName *Return the exact taxonomy name for list of protein*

Description

getTaxonomyName get a character vector of proteins with their UniProt accession code and returns the exact taxonomy code.

Usage

```
getTaxonomyName(x)
```

Arguments

x	A character vector with each entry presenting a protein UniProt accession code.

Value

The exact taxonomy name

Examples

```
getTaxonomyName(x = exmplData1$pl1)
```

---

mod_ont *Database of protein modifications*

Description

Ontology database for post-translational modification terms. For more details, see the reference.

Usage

```
data(mod_ont)
```

Format

A data frame with 2102 rows and 3 variables

Details

- id
- name
- def

Source

https://raw.githubusercontent.com/HUPO-PSI/psi-mod-CV/master/PSI-MOD.obo

---

`plotEnrichment` *Plot and match singular enrichment results*

Description

This function can be used to plot results of singular enrichment analysis for one set of protein. It can also be used to integrate and match the results of two separate singular enrichment analysis and plot the common PTMs. For more details please see examples.

Usage

```
plotEnrichment(x, y = NULL, sig.level = 0.05, number.rep = NULL, plotit = TRUE)
```

Arguments

`x`	A data frame that contains singular enrichment results generated by `runEnrichment`
`y`	Default value is NULL. If provided by a singular enrichment results, the matching results of x and y are plotted.
`sig.level`	The significance level to select post-translational modification (based on their corrected p-value). Note that `sig.level` applies to both x and y simultaneously.
`number.rep`	Only plot PTM terms that occurred more than a specific number of times in UniProt database. This number is set by number.rep parameter. The default value is NULL.
`plotit`	a logical indicating whether you want to draw the plot (TRUE, default value) or you want to return the plot (FALSE).

Value

Plot.

Examples

```
## Enrichment analysis for the first protein list
enrich1 <- runEnrichment(protein = exmplData1$pl1, os.name = 'Homo sapiens (Human)')
## Plot results for first protein list
plotEnrichment(x = enrich1)

## Enrichment analysis for the second protein list
enrich2 <- runEnrichment(protein = exmplData1$pl2, os.name = 'Homo sapiens (Human)')
## Plot results for second protein list
plotEnrichment(x = enrich2)

## Integrate and match the results of two separate singular enrichment analysis
plotEnrichment(x = enrich1, y = enrich2)
plotEnrichment(x = enrich1, y = enrich2, number.rep = 5)
```

plotPSEA *Plot the results of protein set enrichment analysis (PSEA)*

Description

plotPSEA can be used to plot the results of protein set enrichment analysis (psea) for a set of proteins obtained from an experiment.

Usage

```
plotPSEA(x, y = NULL, sig.level = 0.05, number.rep = NULL)
```

Arguments

x	A data frame returned by runPSEA function.
y	Default value is NULL. If provided by a protein set enrichment results, the matching results of x and y are plotted.
sig.level	The significance level applied on adjusted p-value by permutation to filter pathways for plotting. The default value is 0.05
number.rep	Only plot PTM terms that occurred more than a specific number of times in UniProt. This number is set by number.rep parameter. The default value is NULL.

Value

Plot

Examples

```
# We recommend at least nperm = 1000.
# The number of permutations was reduced to 10
# to accommodate CRAN policy on examples (run time <= 5 seconds).
psea_res <- runPSEA(protein = exmplData2, os.name = 'Rattus norvegicus (Rat)', nperm = 10)
plotPSEA(psea_res, sig.level = 0.05)
```

plotRunningScore *Plot running score plot for the results of psea*

Description

This function takes results generated by runPSEA. It plots running enrichment score of ranked protein for each PTM.

Usage

```
plotRunningScore(
  x,
  nplot = length(x$psea.result),
  type = "l",
  lty = 1,
  lwd = 3,
  cex = 1.2,
  cex.axis = 1.2,
  cex.lab = 1.1,
  col = "blue"
)
```

Arguments

x	A list of 6 generated by runPSEA function.
nplot	An integer that defines the number of running score plots to show. Default value is the number of enriched PTMs in x.
type	Type of line used in the plot.
lty	A list of 6 generated by runPSEA function.
lwd	line width
cex	Specify the size of the title text
cex.axis	Specify the size of the tick label
cex.lab	Specify the size of the axis label text
col	Color of running enrichment score line

Value

Plot

Examples

```
# We recommend at least nperm = 1000.
# The number of permutations was reduced to 10
# to accommodate CRAN policy on examples (run time <= 5 seconds).
psea_res <- runPSEA(protein = exmplData2, os.name = 'Rattus norvegicus (Rat)', nperm = 10)
plotRunningScore(x = psea_res)
```

psea2mass *Translate PSEA results for Mass Spectrometry searching tools*

Description

This function translates protein set enrihment analysis results and extracts the required information for mass spectometry searching tools. The subset of protein modifications is from https://raw.githubusercontent.com/HUPO-PSI/psi-mod-CV/master/PSI-MOD.obo.

Usage

```
psea2mass(x, sig.level = 0.05, number.rep = NULL)
```

Arguments

x	A list of psea results generated by runPSEA function.
sig.level	The significance level to filter PTMs (applies on adjusted p-value). Default value is 0.05
number.rep	Only consider PTM terms that occurred more than a specific number of times in UniProt. This number is set by number.rep parameter. The default value is NULL.

Value

A database of subset of protein modifications:

- id: a unique identification for each subset of protein modifications, PSI-MOD.
- name: the name of modification.
- def: definition of PSI-MOD definition

Examples

```
# We recommend at least nperm = 1000.
# The number of permutations was reduced to 10
# to accommodate CRAN policy on examples (run time <= 5 seconds).
psea_res <- runPSEA(protein = exmplData2, os.name = 'Rattus norvegicus (Rat)', nperm = 10)
MS <- psea2mass(x = psea_res, sig.level = 0.05)
```

ptmlist — *Controlled vocabulary for post-translational modifications (PTM) terms*

Description

This dataframe lists the posttranslational modifications used in the UniProt knowledgebase (Swiss-Prot and TrEMBL). The columns in this dataframe are as follows:

Usage

```
data(ptmlist)
```

Format

A data frame with 686 rows and 5 variables

Details

- ID Identifier (FT description)
- AC Accession (PTM-xxxx)
- KW Keyword
- FT Feature key
- DR Cross-reference to external databases

Source

https://ftp.uniprot.org/pub/databases/uniprot/knowledgebase/complete/docs/ptmlist.txt

runEnrichment — *Run singular enrichment analysis (SEA) for a given list of protein*

Description

This function takes proteins with their UniProt accession code, runs singular enrichment (SEA) analysis, and returns enrichment results.

Usage

```
runEnrichment(protein, os.name, blist = NULL, p.adj.method = "BH")
```

Arguments

protein	A character vector with protein UniProt accession codes.
os.name	A character vector of length one with exact taxonomy name of species. If you do not know the the exact taxonomy name of species you are working with, please read getTaxonomyName.
blist	The background list will be substituted with the complete set of UniProt reviewed proteins to facilitate the analysis with a background list. The default value is NULL. Alternatively, if a vector of UniProt Accession Codes is provided, it will serve as the background list for the enrichment analysis.
p.adj.method	The adjustment method to correct for multiple testing. The default value is 'BH'. Run/see p.adjust.methods to get a list of possible methods.

Value

The result is a dataframe with the following columns:

- PTM: Post-translational modification (PTM) keyword
- FreqinUniprot: The total number of proteins in UniProt with this PTM.
- FreqinList: The total number of proteins in the given list with this PTM.
- Sample: Number of proteins in the given list.
- Population: Total number of proteins in the current version of PEIMAN database with this PTM.
- pvalue: The p-value obtained from hypergeometric test (enrichment analysis).
- corrected pvalue: Adjusted p-value to correct for multiple testing.
- AC: Uniprot accession code (AC) of proteins with each PTM.

Examples

```
enrich1 <- runEnrichment(protein = exmplData1$pl1, os.name = 'Homo sapiens (Human)')
```

---

runPSEA *Run Protein Set Enrichment Analysis (PSEA)*

Description

This is the main function to run protein set enrichment analysis for a list of proteins and their score.

Usage

```
runPSEA(
  protein,
  os.name,
  blist = NULL,
  pexponent = 1,
  nperm = 1000,
  p.adj.method = "fdr",
  sig.level = 0.05,
  minSize = 1
)
```

Arguments

protein	A dataframe with two columns. Frist column should be protein accession code, second column is the score.
os.name	A character vector of length one with exact taxonomy name of species. If you do not know the the exact taxonomy name of species you are working with, please read getTaxonomyName.
blist	The background list will be substituted with the complete set of UniProt reviewed proteins to facilitate the analysis with a background list. The default value is NULL. Alternatively, if a vector of UniProt Accession Codes is provided, it will serve as the background list for the enrichment analysis.
pexponent	Enrichment weighting exponent, p. For values of $p < 1$, one can detect incoherent patterns in a set of protein. If one expects a small number of proteins to be coherent in a large set, then $p > 1$ is a good choice.
nperm	Number of permutation to estimate false discovery rate (FDR). Default value is 1000.
p.adj.method	The adjustment method to correct pvalues for multiple testing in enrichment. Run p.adjust.methods() to get a list of possible methods.
sig.level	The significance level to filter PTM (applies on adjusted p-value)
minSize	PTMs with the number of proteins below this threshold are excluded.

Value

Returns a list of 6: 1: A dataframe with protein set enrichment analysis (PSEA) results. Every row corresponds to a post-translational modification (PTM) keyword.

- PTM: PTM keyword
- pval: p-value obtained from singular enrichment analysis (SEA).
- pvaladj: adjusted p-value. This column is the adjusted pvalues with p.adj.method methods calculated in SEA method.
- FreqinPopulation: The frequency of PTM in UniProt.
- FreqinSample: The frequency of PTM in the given list.
- ES: enrichment score.

- NES: enrichmnt score normalized to mean enrichment of random samples of the same size.
- nMoreExtreme: number of times the permuted sample resulted in a profile with a larger ES value than abs(ES) of the sample.
- size: Number of proteins in the list having this specific PTM.
- Enrichment: Indicates if the proteins with the specific protein have been enriched in the list or not. NES positive is considered as enriched.
- AC: Uniprot accession code (AC) of proteins with the specific PTM.
- leadingEdge: the leading edge proteins are the proteins that show up in the ranked list at or before the point where the enrichment score (ES) reaches its maximum deviation from zero.

Examples

```
# We recommend at least nperm = 1000.
# The number of permutations was reduced to 10
# to accommodate CRAN policy on examples (run time <= 5 seconds).
psea_res <- runPSEA(protein = exmplData2, os.name = 'Rattus norvegicus (Rat)', nperm = 10)
```

`sea2mass` *Translate SEA results for Mass Spectrometry searching tools*

Description

This function translates singular enrichment analysis results and extracts the required information for mass spectometry searching tools. The subset of protein modifications is from https://raw.githubusercontent.com/HUPO-PSI/psi-mod-CV/master/PSI-MOD.obo.

Usage

```
sea2mass(x, sig.level = 0.05, number.rep = NULL)
```

Arguments

`x`	A dataframe of single enrichment analysis results generated by `runEnrichment` function.
`sig.level`	The significance level to filter pathways (applies on adjusted p-value). Default value is 0.05.
`number.rep`	Only consider PTM terms that occurred more than a specific number of times in UniProt. This number is set by number.rep parameter. The default value is NULL.

Value

A database of subset of protein modifications:

- id: a unique identification for each subset of protein modifications, PSI-MOD.
- name: the name of modification.
- def: definition of PSI-MOD definition

Examples

```
enrich1 <- runEnrichment(protein = exmplData1$pl1, os.name = 'Homo sapiens (Human)')
MS      <- sea2mass(x = enrich1, sig.level = 0.05)
```

Index