Package 'JUMPsem'

October 27, 2025

Type Package

Title Enzyme Activity Inference with Structural Equation Modeling

Version 1.1.0

Date 2025-10-27

Maintainer Dehui Kong <dkong3@uthsc.edu>

Depends R (>= 3.5.0)

Imports lavaan (>= 0.6-11),
tidyr (>= 1.2.0),
stringr (>= 1.4.0),
dplyr (>= 1.0.9),
psych (>= 2.2.5),
devtools (>= 2.4.5),
EFAtools (>= 0.4.1),
data.table (>= 1.17.8),
tidyverse (>= 2.0.0),
MASS,
Matrix

Suggests knitr,
rmarkdown

Description Provides tools to infer enzyme activity using structural equation modeling (SEM). The package integrates latent variable modeling with omics data to estimate pathway-level enzyme activity.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

VignetteBuilder knitr

Contents

`aceAdjacency` *aceAdjacency*

Description

This script is used to build HUMAN HAT-substrate adjacency matrix.

Usage

```
aceAdjacency(databaseACE)
```

Arguments

`databaseACE`	Default database or customized database input.

`catchActRaw` *catchResults*

Description

Catch Activity and Affinity.

Usage

```
catchActRaw(result_list)
```

Arguments

`result_list`	The result list from modeling fitting.
`rawData`	Input raw data (quantitative phospho-proteomics / ubiqutin-proteomics data)
`eval_table`	eval_table A tibble or data.frame produced by ‘catchEvaluations()‘.

getFa *Build MEME input*

Description

script to extract MEME input .fa file

Usage

```
getFa(ss = NULL, es = NULL, filePath)
```

Arguments

ss	Vector; Substrate species; ("mouse", "rat", "human")
es	Vector; Kinase species; ("mouse", "rat", "human")
filePath	The path is used to save .fa files. Recommend to create a new folder to save all .fa.files.

Examples

```
getFa(ss = "human", es = c("human", "mouse", "rat"), filePath = getwd())
```

inputTransformPSP *inputTransform*

Description

This script is used to log2 transform AND/OR whole proteome normalize input data.

Usage

```
inputTransformPSP(
  input_raw,
  input_log2_trans = FALSE,
  relative.norm.p = TRUE,
  whole_log2_trans = FALSE,
  whole_proteome = NULL,
  relative.norm.w = TRUE
)
```

Arguments

input_raw	Input raw data (quantitative phospho-proteomics/ubiquitin-proteomics data)
input_log2_trans	FALSE or TRUE. Need program to do log2 transforming of the input file or not. Default is FALSE.
relative.norm.p	Logical. FALSE or TRUE. Need program to do relative normalization of the PTM input file or not. Default is TRUE.
whole_log2_trans	Need program to do log2 transforming of the whole proteome or not. Ignore if -whole.proteome is missing. Default is FALSE.
whole_proteome	Set up whole proteome used to normalize phospho-proteomics/ubiquitin-proteomics data by whole proteome. Default is NULL.
relative.norm.w	Logical. FALSE or TRUE. Need program to do relative normalization of the whole proteomic input file or not. Default is TRUE.

JUMPsem *JUMPsem*

Description

Infer enzyme activity from phosphoralation, ubiquitination, acetylation

Usage

```
JUMPsem(
  input = NULL,
  datatype = NULL,
  organism = NULL,
  enzyme.organism = NULL,
  database = NULL,
  kmo.off = 0,
  mdsite = TRUE,
  enzyList = NULL,
  input.log2.norm = FALSE,
  relative.norm.p = TRUE,
  whole.log2.trans = FALSE,
  whole.proteome = NULL,
  relative.norm.w = TRUE,
  motif = NULL,
  output.folder = getwd()
)
```

Arguments

input	Dataframe. Your quantitative ubiquiti-proteomics, phospho-proteomic and acetyl-proteomics data.
datatype	Factor. Data type: ubiquitination or phosphoralation or acetylation; Corresponding values are "ubi", "psp" and "ace".

`organism` Vector. Substrate species. Corresponding values are "human", "mouse" and "rat".

`enzyme.organism` Vector. Enzyme species; Corresponding values are "human", "mouse" and "rat".

`database` Dataframe. Customized database input. Default is NULL using internal database.

`kmo.off` Numeric. Set up KMO cutoff value 0-1. Default is 0.

`mdsite` Logical. Mapping with precise phosphorylated modification sites or not. Default is TRUE.

`enzyList` Vector. Program only calculate the enzyme in the enzyList. Default is to output ALL enzyme activities and affinities.

`input.log2.norm` Logical. FALSE or TRUE. Need program to do log2 transforming of the input file or not. Default is FALSE.

`relative.norm.p` Logical. FALSE or TRUE. Need program to do relative normalization of the PTM input file or not. Default is TRUE.

`whole.log2.trans` Logical. FALSE or TRUE. Need program to do log2 transforming of the whole proteome or not. (Ignore if -whole.proteome is missing). Default is FALSE.

`whole.proteome` Dataframe. Set up whole proteome used to normalize phosphor-proteome or ubiquitin-proteome whole proteome. Default is NULL.

`relative.norm.w` Logical. FALSE or TRUE. Need program to do relative normalization of the whole proteomic input file or not. Default is TRUE.

`motif` Matrix. Added kinase-substrate relationships from motif discovery.

`output.folder` Character. Character vector of location to save files if desired. Default is current directory.

Details

You can use JUMPsem package to infer enzyme activity.

Value

List

Examples

```
result <- JUMPsem(input = input_ubi_example,
                  datatype = "ubi",
                  organism = "human",
                  input.log2.norm = T)


result <- JUMPsem(input = input_psp_example,
                  datatype = "psp",
                  organism = "mouse",
                  enzyme.organism = c("human", "mouse", "rat"),
                  input.log2.norm = TRUE,
                  whole.log2.trans = TRUE,
                  motif = motif_example,
```

```
                    whole.proteome = wholeProteome_example)

result <- JUMPsem(input = input_ace_example,
                    datatype = "ace",
                    organism = "human",
                    input.log2.norm = FALSE)
```

---

mastInput *Build MAST input*

Description

Script to extract MAST input FASTA file.

Usage

```
mastInput(input, savePath = getwd())
```

Arguments

input	The same quantitative phosphoproteomics data as eSEM() input file.
savePath	The path to save built MAST input file. Default is the current directory.

Examples

```
mastInput(input = input_psp_example)
```

---

motifEx *motifEx*

Description

This script is used to extract kinase-substrate relationships from motif scanning and build motif_input file for KSEM().

Usage

```
motifEx(mastFile, savePath = getwd())
```

Arguments

mastFile	The output file "extract_result.txt" from Step: 03step_ExtractMASTresult.sh.
savePath	Path to save motif_input.txt file. Default is current directory.

pspAdjacency *pspAdjacency*

Description

This script is used to build kinase-substrate adjacency matrix with only PSP database.

Usage

```
pspAdjacency(sp, ep, databasePSP, mdsite)
```

Arguments

sp	Substrate species; ("mouse", "rat", "human").
ep	Enzyme species; ("mouse", "rat", "human").
databasePSP	Default database or customized database input.
mdsite	Logical. Mapping with precise phosphorylated modification sites or not. Default is TRUE.

pspMotifAdjacency *pspMotifAdjacency*

Description

This script is used to build enzyme-substrate adjacency matrix with PSP database + Motif.

Usage

```
pspMotifAdjacency(sp, ep, databasePSP, motif.ref, mdsite)
```

Arguments

sp	Substrate species; ("mouse", "rat", "human").
ep	Enzyme species; ("mouse", "rat", "human").
databasePSP	Default database or customized database input.
motif.ref	Added enzyme-subatrate relationships from motif discovery. Same as KSEM -motif parameter.
mdsite	Logical. Mapping with precise phosphorylated modification sites or not. Default is TRUE.

`singleEnzymeSEM` *singleEnzymeSEM*

Description

Estimate the latent activity of a single enzyme using Structural Equation Modeling (SEM). This function identifies substrates linked to an enzyme, filters based on data quality, and infers latent enzyme activity using lavaan. It also evaluates the model fit (CFI, TLI, RMSEA, SRMR) and provides a descriptive report.

Usage

```
singleEnzymeSEM(enzyme, input, adj, kmo.off = 0, mdsite = TRUE)
```

Arguments

`enzyme`	Character. Name of the enzyme (must match column name in adjacency matrix).
`input`	Data frame. Normalized and transformed phosphoproteomics data (rows = sites, cols = samples).
`adj`	Matrix. Enzyme–substrate adjacency matrix (rows = substrates, columns = enzymes).
`kmo.off`	Numeric. Cutoff for KMO (Kaiser-Meyer-Olkin) measure; default = 0.
`mdsite`	Logical. If TRUE, match substrates using exact PTM site IDs (e.g., "CDK16_S12"). If FALSE, match using gene-level mapping (e.g., "CDK16").

Value

A list with three components for the given enzyme:

activity Latent enzyme activity scores (sample × 1 matrix)

affinity Table of substrate loadings, z-values, and p-values

evaluation Model fit indices and descriptive interpretation

`ubiAdjacency` *ubiAdjacency*

Description

This script is used to build ligase-substrate adjacency matrix with only PSP database.

Usage

```
ubiAdjacency(sp, databaseUBI)
```

Arguments

`sp`	Substrate species; ("mouse", "rat", "human").
`databaseUBI`	Default database or customized database input.

Index