

# Package ‘eSEM’

December 21, 2023

**Type** Package

**Title** Enzyme activity inference from Structural Equation Modeling

**Version** 1.0

**Date** 2023-09-13

**Author** Dehui Kong

**Maintainer** Dehui Kong <dkong3@uthsc.edu>

**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)

**Description** eSEM is used to infer enzyme activity.

**Depends** R (>= 3.5.0)

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.2.3

**Suggests** knitr,  
rmarkdown

**VignetteBuilder** knitr

## R topics documented:

aceAdjacency . . . . .	2
catchActRaw . . . . .	2
eSEM . . . . .	2
getFa . . . . .	4
inputTransformPSP . . . . .	5
mastInput . . . . .	5
motifEx . . . . .	6
pspAdjacency . . . . .	6
pspMotifAdjacency . . . . .	7
singleEnzymeAff . . . . .	7
singleHatAct . . . . .	8
singleKinaseAct . . . . .	8
singleLigaseAct . . . . .	9
ubiAdjacency . . . . .	9

---

`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. Activity contains Raw-Activity, Mean-Center, Z-scaled.

**Usage**

```
catchActRaw(x)
```

**Arguments**

`x`                      The results from modeling fitting.  
`rawData`              Input raw data (quantitative phospho-proteomics / ubiquitin-proteomics data)

---

`eSEM`*eSEM*

---

**Description**

Infer enzyme activity from phosphoralation, ubiquitination, acetylation

**Usage**

```
eSEM(
  input = NULL,
  datatype = NULL,
  organism = NULL,
  enzyme.organism = NULL,
  database = NULL,
  cor.off = 0.95,
  kmo.off = 0,
  enzyList = NULL,
  input.log2.norm = FALSE,
  whole.log2.trans = FALSE,
  whole.proteome = NULL,
  motif = NULL,
  output.folder = getwd()
)
```

**Arguments**

input	Dataframe. Your quantitative ubiquitin-proteomics, phospho-proteomic and acetyl-proteomics data.
datatype	Factor. Data type: ubiquitination or phosphorylation or acetylation; ("ubi", "psp", "ace")
organism	Vector. Substrate species; ("human", "mouse", "rat")
enzyme.organism	Vector. Enzyme species; ("human", "mouse", "rat")
database	Dataframe. Customized database input. Default is NULL using internal database.
cor.off	Numeric. Set up correlation cutoff value 0-1 to remove high collinear variables. Default is 0.95.
kmo.off	Numeric. Set up KMO cutoff value 0-1. Default is 0.
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.
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.
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 eSEM package to get enzyme activity.

**Value**

List

**Examples**

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

result <- eSEM(input = input_psp_example,
               datatype = "psp",
               organism = "mouse",
               enzyme.organism = c("human", "mouse", "rat"),
               cor.off = 0.8,
               input.log2.norm = TRUE,
               whole.log2.trans = TRUE,
               motif = motif_example,
               whole.proteome = wholeProteome_example)

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

---

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	<i>inputTransform</i>
-------------------	-----------------------

---

**Description**

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

**Usage**

```
inputTransformPSP(
  input_raw,
  input_log2_trans = FALSE,
  whole_log2_trans = FALSE,
  whole_proteome = NULL
)
```

**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.
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.

---

mastInput	<i>Build MAST input</i>
-----------	-------------------------

---

**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	<i>motifEx</i>
---------	----------------

---

**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	<i>pspAdjacency</i>
--------------	---------------------

---

**Description**

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

**Usage**

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

**Arguments**

sp	Substrate species; ("mouse", "rat", "human").
ep	Enzyme species; ("mouse", "rat", "human").
databasePSP	Default database or customized database input.

---

pspMotifAdjacency	<i>pspMotifAdjacency</i>
-------------------	--------------------------

---

## Description

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

## Usage

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

## Arguments

sp	Substrate species; ("mouse", "rat", "human").
ep	Enzyme species; ("mouse", "rat", "human").
databasePSP	Default database or customized database input.
motif.ref	Added enzyme-substrate relationships from motif discovery. Same as KSEM -motif parameter.

---

singleEnzymeAff	<i>singleEnzymeAff</i>
-----------------	------------------------

---

## Description

Get single enzyme affinity.

## Usage

```
singleEnzymeAff(enzyme, input, adj, cor.off, kmo.off)
```

## Arguments

enzyme	Single enzyme name.
input	Normalized and transformed phospho-proteomics / ubiquitin-proteomics data.
adj	Adjacency matrix.
cor.off	Set up correlation cutoff value 0-1 to remove high collinear variables. Default is 0.95.
kmo.off	Set up KMO cutoff value 0-1. Default is 0.

---

singleHatAct	<i>singleHatAct</i>
--------------	---------------------

---

**Description**

Get single HAT activity.

**Usage**

```
singleHatAct(HAT, input, adj, cor.off, kmo.off)
```

**Arguments**

HAT	Single HAT name.
input	Normalized and transformed acetyl-proteomics data.
adj	Adjacency matrix.
cor.off	Set up correlation cutoff value 0-1 to remove high collinear variables. Default is 0.95.
kmo.off	Set up KMO cutoff value 0-1. Default is 0.

---

singleKinaseAct	<i>singleKinaseAct</i>
-----------------	------------------------

---

**Description**

Get single kinase activity.

**Usage**

```
singleKinaseAct(kinase, input, adj, cor.off, kmo.off)
```

**Arguments**

kinase	Single kinase name.
input	Normalized and transformed phosphoproteomics data.
adj	Adjacency matrix.
cor.off	Set up correlation cutoff value 0-1 to remove high collinear variables. Default is 0.95.
kmo.off	Set up KMO cutoff value 0-1. Default is 0.



---

singleLigaseAct	<i>singleLigaseAct</i>
-----------------	------------------------

---

**Description**

Get single ligase activity.

**Usage**

```
singleLigaseAct(ligase, input, adj, cor.off, kmo.off)
```

**Arguments**

ligase	Single ligase name.
input	Normalized and transformed ubiquiti-proteomics data.
adj	Adjacency matrix.
cor.off	Set up correlation cutoff value 0-1 to remove high collinear variables. Default is 0.95.
kmo.off	Set up KMO cutoff value 0-1. Default is 0.

---

ubiAdjacency	<i>ubiAdjacency</i>
--------------	---------------------

---

**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.