Package 'Miso'

March 9, 2019

Type Package

Title Multi-Isotope Labeling for Metabolomics Analysis

Version 0.2.1

Author Yonghui Dong

Maintainer Yonghui Dong <yonghui.dong@gmail.com>

Description An efficient approach for fishing out the dual (or multiple) isotope labeled analytes using dual labeling of metabolites for metabolome analysis (DLEMMA) approach, described in Liron (2018) <doi:10.1021/acs.analchem.8b01644>, and Dong (2019) <doi:10.1093/bioinformatics/btz092>.

Depends R (>= 3.1.0)

biocViews

Imports xcms, stats, utils, dplyr, ggplot2, scales, plotly

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2019-03-09 08:02:40 UTC

R topics documented:

diso *Isotope filtering*

Description

filtering isotopically labeled analytes according to RT and mass differences

Usage

```
diso(iso1, n11, n12, iso2 = "NO", n21 = 0, n22 = 0, exp.base,
  exp.iso, ppm = 10, rt.dif = 6, poly = 1)
```

Arguments

iso1	the first labled atom in precusor ion.
n11	the maximum numbers of the first labled atoms expected in the labeled intermediates.
n12	the minimum numbers of the first labled atoms expected in the labeled intermediates.
iso2	the second labled atom in the same precusor ion, default value 'NO' (not exist).
n21	the maximum numbers of the second labled atoms expected in the labeled intermediates, default value 0.
n22	the minimum numbers of the second labled atoms expected in the labeled intermediates, default value 0.
exp.base	the control group (fed with unlabled precusor).
exp.iso	isotope labeled group.
ppm	m/z tolarance, default value 30.
rt.dif	retention time tolarance, default value 6 seconds.
poly	polymer of the feeding precursor derived metabolites, e.g. dimer poly = 2, trimer poly =3, default value 1.

Value

results containing unlabled and their corresponding labled analytes, with RT and labeling information.

Examples

```
data(lcms)
explist <- prefilter(lcms, subgroup = c("B", "C", "D"), unlabel = "B")
exp.B <- explist$B
exp.C <- explist$C
exp.D <- explist$D
iso.C <- diso(iso1 = 'H2', n11 = 4, n12 = 2, exp.base = exp.B, exp.iso = exp.C)
iso.D <- diso(iso1 = 'C13', n11 = 9, n12 = 6, iso2 = 'N15', n21 = 1, n22 = 0,
exp.base = iso.C[,1:3], exp.iso = exp.D)
```

fold *fold change*

Description

calculate fold change among different samples.

Usage

```
fold(x)
```

Arguments

x	dataframe

Value

a dataframe with mean values and fold changes

Examples

```
vars <- 1000
samples <- 50
groups <- 3
dat <- replicate(vars, runif(n = samples))
f <- rep_len(1:groups, samples)
f <- LETTERS[f]
dat <- data.frame(dat, group = f)
ret <- fold(dat)
```

Fresult *Full result list*

Description

export full isotope labeled result list

Usage

```
Fresult(x, ...)
```

Arguments

x	results generated from diso() function. The input can be multiple depending on the experiment design: i.e. single labeling, dual labeling and multiple lableing.
...	results generated from diso() function

Value

file containing the all the possible combined results.

Examples

```
data(lcms)
explist <- prefilter(lcms, subgroup = c("B", "C", "D"), unlabel = "B")
exp.B <- explist$B
exp.C <- explist$C
exp.D <- explist$D
iso.C <- diso(iso1 = 'H2', n11 = 4, n12 = 2, exp.base = exp.B, exp.iso = exp.C)
iso.D <- diso(iso1 = 'C13', n11 = 9, n12 = 6, iso2 = 'N15', n21 = 1, n22 = 0,
exp.base = iso.C[,1:3], exp.iso = exp.D)
full_result <- Fresult(iso.C, iso.D)
```

---

getp *get p-values*

Description

get p-values from Post Hoc analysis

Usage

```
getp(dat)
```

Arguments

dat	peaklist

Value

a data frame

Examples

```
dat = as.data.frame(matrix(runif(2*300), ncol = 2, nrow = 300))
dat$group = rep(LETTERS[2:4], 100)
out <- getp(dat)
```

getp0 *get artifical p-values*

Description

get artifical p-values for groups without replicates

Usage

```
getp0(x)
```

Arguments

x	dataframe

Value

a data frame

Examples

```
dat = as.data.frame(matrix(runif(2*3), ncol = 2, nrow = 3))
dat$group = rep(LETTERS[2:4], 1)
out <- getp0(dat)
```

isoplot *plot isotopologues*

Description

plot unlabled and labeled Isotopologues from filtering result

Usage

```
isoplot(dat, rinx)
```

Arguments

dat	isotope filtering result
rinx	row index

Value

interactive plot

Examples

```
data(lcms)
explist <- prefilter(lcms, subgroup = c("B", "C", "D"), unlabel = "B")
exp.B <- explist$B
exp.C <- explist$C
exp.D <- explist$D
iso.C <- diso(iso1 = 'H2', n11 = 4, n12 = 2, exp.base = exp.B, exp.iso = exp.C)
iso.D <- diso(iso1 = 'C13', n11 = 9, n12 = 6, iso2 = 'N15', n21 = 1, n22 = 0,
exp.base = iso.C[,1:3], exp.iso = exp.D)
full_result <- Fresult(iso.C, iso.D)
isoplot(full_result, 1)
```

---

lcms *An S4 class to represent a LC-MS dataset*

Description

An S4 class to represent a LC-MS dataset.

Usage

```
data(lcms)
```

Slots

`peaks` peaks

`groups` groupd

`filled` filled

`phenoData` phenoData

`rt` retention time

`polarity` polsrity

`mslevel` MS level

`scanrange` scan range

Examples

```
data(lcms)
```

prefilter *Prefilter*

Description

prefiltering isotopically labeled analytes according to the experiment design.

Usage

```
prefilter(xset, subgroup = NULL, unlabel = NULL, reps = TRUE,
  p = 0.05, folds = 10)
```

Arguments

xset	xcms object.
subgroup	subset the xcms groups. The name should be the same as in phboData$class. default = NULL, which means no subset will be performed.
unlabel	specify which is unlabeled group.
reps	if there are replicates in the sample.
p	p-value threshold, default value = 0.05
folds	fold change threshold, default value = 10

Value

a filtered peaklist

Examples

```
data(lcms)
explist <- prefilter(lcms, subgroup = c("B", "C", "D"), unlabel = "B")
```

Rresult *Reduced result list*

Description

export reduced isotope labeled result list

Usage

```
Rresult(full_Result)
```

Arguments

full_Result	full result list

Value

csv file with repeated results being removed

Examples

```
data(lcms)
explist <- prefilter(lcms, subgroup = c("B", "C", "D"), unlabel = "B")
exp.B <- explist$B
exp.C <- explist$C
exp.D <- explist$D
iso.C <- diso(iso1 = 'H2', n11 = 4, n12 = 2, exp.base = exp.B, exp.iso = exp.C)
iso.D <- diso(iso1 = 'C13', n11 = 9, n12 = 6, iso2 = 'N15', n21 = 1, n22 = 0,
exp.base = iso.C[,1:3], exp.iso = exp.D)
full_result <- Fresult(iso.C, iso.D)
reduced_result <- Rresult(full_result)
```

Index