Package: KPIC (via r-universe)

November 2, 2024

Type Package

Title Mass Spectrometry-Based Metabolomics Using Pure Ion Chromatograms

Version 2.4.0

Author Hongchao Ji

Maintainer Hongchao Ji <ji.hongchao@foxmail.com>

Description KPIC2 is an effective platform for LC-MS based metabolomics using pure ion chromatograms, which is developed for metabolomics studies. KPIC2 can detect pure ions accurately, align PICs across samples, group PICs to annotate isotope and adduct PICs, fill missing peaks and pattern recognition. High-resolution mass spectrometers like TOF and Orbitrap are more suitable.

License GPL (>= 2)

Encoding UTF-8

LazyData true

Imports Rcpp, RcppArmadillo, mzR, parallel, shiny, plotly, data.table, GA, IRanges, dbscan, Ckmeans.1d.dp, jsonlite, randomForest, ropls, Matrix

LinkingTo Rcpp, RcppArmadillo

Repository https://rickhelmus.r-universe.dev

RemoteUrl https://github.com/rickhelmus/KPIC2

RemoteRef HEAD

RemoteSha dd0dddb0cac9bc7882eb1a420c82c9713a8aa0ce

Contents

`analyst.OPLS` *Analyst the peaks with PLS-DA or OPLS-DA*

Description

PLS, and OPLS classification

Usage

```
analyst.OPLS(labels, data)
```

Arguments

`labels`	A response vector.
`data`	The result of `getDataMatrix` or `fillPeaks` function

analyst.RF *Analyst the peaks with random forest*

Description

random forest algorithm (based on Breiman and Cutler's original Fortran code) for classification.

Usage

```
analyst.RF(labels, data)
```

Arguments

labels	A response vector.
data	The result of getDataMatrix or fillPeaks function

fillPeaks.EIBPC *identify missing peaks*

Description

For each sample, identify missing peaks resulting from peak detection or other steps. The EIBPC is used to achieve this aim.

Usage

```
fillPeaks.EIBPC(groups, extand_mz=20, extand_rt=5, min_snr=3, std='maxo')
```

Arguments

groups	The result of getDataMatrix function
extand_mz	PPM of m/z tolerance of filled peaks.
extand_rt	Retention time tolerance of filled peaks.
min_snr	The minimum SNR of peaks, which may be smaller than that of getPIC function
std	The standard for quantification, only 'maxo' is supported now.

getMS *get MS of a LC-MS data file.*

Description

get MS of a LC-MS data file.

Usage

```
getMS(filename)
```

Arguments

filename	The path of a LC-MS data file.

Value

a LIST, use

path	path of each LC-MS data file.
MS	MS.

getPeaks *Get peaks of the detected PICs.*

Description

Get the information peaks of the detected PICs, including m/z, retention time, snr, scale, height and peak area, etc. Note, only the information of highest peak of a PIC will be included.

Usage

```
getPeaks(pics)
```

Arguments

pics	The result object of getPIC, getPIC.kmeans, PICsplit, PICresolve or PICfit founction.

Value

scantime	The retention time of each scan.
pics	The extracted mass trace.
peaks	The detected peak of each mass trace.
peakInfo	The information of the peaks.

getPIC *Extract PICs from a LC-MS raw object based on m/z difference.*

Description

This method bases on the extension of mass trace depending on the m/z difference. The tolerence is described via mean and variance.

Usage

```
getPIC(raw, level, mztol = 0.1, gap = 3, width = 5, min_snr = 4, ...)
```

Arguments

raw	Raw LC-MS data object obtained by LoadData function.
level	Mass traces are only retained if their maximum values are over level.
mztol	The initial m/z tolerence.
gap	The number of gap points of a mass trace.
width	The minimum length of a mass trace.
min_snr	Minimum signal to noise ratio.
...	No use at present.

Value

a LIST of:

scantime	The retention time of each scan.
pics	The extracted mass trace.
peaks	The detected peak of each mass trace.

See Also

getPIC.kmeans

`getPIC.kmeans` *Extract PICs from a LC-MS raw object based on optimal k-means clustering.*

Description

This method bases on the optimal k-means clustering of m/z values of data points in ROI. see reference for details.

Usage

```
getPIC.kmeans(raw, level, mztol = 0.1, gap = 3, width = c(5, 60), alpha = 0.3, min_snr = 4, ...)
```

Arguments

`raw`	Raw LC-MS data object obtained by `LoadData` function.
`level`	Mass traces are only retained if their maximum values are over `level`.
`mztol`	The m/z range of ROI.
`gap`	The number of gap points of a mass trace.
`width`	The range of a mass trace.
`alpha`	The parameter of forecasting.
`min_snr`	Minimum signal to noise ratio.
`...`	No use at present.

Value

a LIST of:

`scantime`	The retention time of each scan.
`pics`	The extracted mass trace.
`peaks`	The detected peak of each mass trace.

References

Ji, H., et al. "KPIC2: An Effective Framework for Mass Spectrometry-Based Metabolomics Using Pure Ion Chromatograms." Analytical Chemistry (2017).

See Also

`getPIC`

getTICs *Get TICs of LC-MS data.*

Description

Get TICs of LC-MS data.

Usage

```
getTICs(files, method = "BPC")
```

Arguments

files	The path of LC-MS files.
method	TIC or BPC

Value

a LIST, use

rt	retention time of each scan.
tics	obtained tics.

groupCombine *Combine tailed, isotopic or/and adduct features into the same group.*

Description

Combine tailed, isotopic or/and adduct features into the same group.

Usage

```
groupCombine(groups, min_corr = 0.9, type = "tailed", window = 10)
```

Arguments

groups	The result of PICset.group function.
min_corr	the minimum coefficient between peaks, which are regarded as isotopes or adducts and the base feature.
type	'tailed' for tailed features; 'isotope' for tailed features and isotopic features; or 'all'.
window	the width of RT window.

Value

a LIST of:

peakmat	The peakmat with group index.
picset	The picset.
group.info	The information of each group.

LoadData *Load an LC-MS data file.*

Description

This function handles the task of reading a NetCDF/mzXML file containing LC-MS data.

Usage

```
LoadData(filename)
```

Arguments

filename	The path of LC-MS data file

Value

A LIST of:

mz	The vector of m/z values.
scans	The vector of scan indexes.
ints	The vector of intensity values.
times	The vector of unique time points.

PICset *Process a set of sample with* getPIC *method.*

Description

This function is used to process a dataset produced by LC-MS.

Usage

```
PICset(files, level, mztol = 0.1, gap = 3, width = 5, min_snr = 4, equal = TRUE, export=FALSE, par=TRUE,
```

Arguments

files	The path of the LC-MS files folder.
level	see getPIC
mztol	see getPIC
gap	see getPIC
width	see getPIC
min_snr	see getPIC
equal	Whether the retention times of samples are equaled or not. Equalization is need for alignment procedure.
export	Whether to export PICs of each sample as single files
par	Whether to use multi-core calculation
...	see getPIC

Value

a LIST of PICs, each element is the result of getPIC function.

PICset.align *Align each group of PICs.*

Description

This function is used to calculated the shifts of PICs in each group, and correct the retention times of the peakmat and picset obtain by PICset.group function.

Usage

```
PICset.align(groups, method = "fftcc", move = "direct", span = 1.5)
```

Arguments

groups	The result of PICset.group function.
method	Which method is used to calculated the shift. can be 'match' of 'fftcc'. 'match' means calculating the difference of the retention time of detected peak position. 'fftcc' means use fft cross correlation method to maximize the similarity of peak shape.
move	Which method is used to move the original to new position. can be 'direct' or 'loess'. 'direct' means directly move each PIC based on the calculated shift. 'loess' means use a loess regression to the obtained shift and predict a new shift of each PIC, then move each PIC based on the new values.
span	The parameter which controls the degree of smoothing. Only used when the move is 'loess'

Value

a LIST of:

peakmat	The peakmat with refreshed rt.
picset	The picset of refreshed rt.

PICset.getPeaks *The* getPeaks *function for a set of samples.*

Description

This function is used to apply PICfit method to a PIC set.

Usage

```
PICset.getPeaks(picset)
```

Arguments

picset	The result object of PICset, PICset.kmeans, PICset.split, PICset.resolve or PICset.fit founction.

Value

The processed picset object

PICset.group *group the features*

Description

This function is used to group the features across samples.

Usage

```
PICset.group(picset, tolerance = c(0.01, 10), weight = c(0.8, 0.2), method = "score", frac = 0.5)
```

Arguments

picset	The result of PICset.getPeaks function.
tolerance	Maximum allowed absolute m/z and RT difference
weight	The assigned weight for m/z and RT difference at the moment of match score calculation between peaks.
method	Which method is used. can be 'score' or 'dbscan'. 'dbscan' means group features with dbscan clustering method; 'score' means group features with calculated scores.
frac	Minimum fraction of samples necessary in at least one of the sample groups.

Value

a LIST of:

peakmat	The final peakmat of all sample with group id.
picset	The input picset

PICset.kmeans *Process a set of sample with* getPIC.kmeans *method.*

Description

This function is used to process a dataset produced by LC-MS.

Usage

```
PICset.kmeans(files, level, mztol = 0.1, gap = 3, width = c(5, 60), min_snr = 4, alpha = 0.3, equal = TRUE
```

Arguments

files	The path of the LC-MS files folder.
level	see getPIC.kmeans
mztol	see getPIC.kmeans
gap	see getPIC.kmeans
width	see getPIC.kmeans
min_snr	see getPIC.kmeans
alpha	see getPIC.kmeans
equal	Whether the retention times of samples are equaled or not. Equalization is need for alignment procedure.
export	Whether to export PICs of each sample as single files
par	Whether to use multi-core calculation
...	see getPIC.kmeans

Value

a LIST of PICs, each element is the result of getPIC.kmeans function.

PICset.split *The* PICsplit *function for a set of samples.*

Description

This function is used to apply PICsplit method to the result of PICset or PICset.kmeans founction.

Usage

```
PICset.split(picset, par = FALSE)
```

Arguments

picset	The result of PICset or PICset.kmeans founction.
par	Whether parallel method is used.

Value

The processed picset object

PICsplit *Spliting multiple-peak trace into single ones*

Description

If there is more than one peak in a mass trace, and they are obviously separated, they can be split with this function.

Usage

```
PICsplit(pics)
```

Arguments

pics	The result of getPIC or getPIC.kmeans function.

Value

a LIST of:

scantime	The retention time of each scan.
pics	The extracted mass trace.
peaks	The detected peak of each mass trace.

viewAlign *View the result of alignment.*

Description

View the result of alignment.

Usage

```
viewAlign(groups_raw, groups_align)
```

Arguments

groups_raw	The result of PICgroup
groups_align	The result of PICalign

Value

a shiny app.

viewGroups *View the result of group.*

Description

View the result of group.

Usage

```
viewGroups(groups)
```

Arguments

groups	The result of PICset.group function.

Value

a shiny app.

viewMS *View MS.*

Description

View MS.

Usage

```
viewMS(MS)
```

Arguments

MS	The result object of getMS function.

Value

A shiny app.

viewPICs *View the PICs.*

Description

View the PICs.

Usage

```
viewPICs(pics)
```

Arguments

pics	The result object of getPIC, getPIC.kmeans, PICsplit, PICresolve or PICfit founction.

Value

A shiny app.

viewTICs *View TICs*

Description

View TICs

Usage

```
viewTICs(tics)
```

Arguments

tics	The result object of getTICs

Value

A shiny app.

WMPD *Reslove overlapped peak based on mass spectrometry.*

Description

Reslove overlapped peak based on mass spectrometry.

Usage

```
WMPD(pic, min_snr, level, pval, iter)
```

Arguments

pic	Extracted ion trace.
min_snr	Minimum signal to noise ratio.
level	Peaks are only retained if their maximum values are over level.
pval	The p-value threshold of different peaks.
iter	Number of iteration

Value

The result of peak detection.

Index