# Package 'HiResTEC'

October 12, 2022

**Type** Package

**Title** Non-Targeted Fluxomics on High-Resolution Mass-Spectrometry Data

**Version** 0.59

**Date** 2019-06-18

**Maintainer** Jan Lisec <jan.lisec@bam.de>

**Description** Identifying labeled compounds in a 13C-tracer experiment in non-targeted fashion is a cumbersome process. This package facilitates such type of analyses by providing high level quality control plots, deconvoluting and evaluating spectra and performing a multitude of tests in an automatic fashion.
The main idea is to use changing intensity ratios of ion pairs from peak list generated with 'xcms' as candidates and evaluate those against base peak chromatograms and spectra information within the raw measurement data automatically.
The functionality is described in Hoffmann et al. (2018) <doi:10.1021/acs.analchem.8b00356>.

**License** GPL-3

**URL** https://pubs.acs.org/doi/10.1021/acs.analchem.8b00356

**LazyData** TRUE

**Depends** R(>= 2.10.0)

**biocViews**

**Imports** plyr, openxlsx, InterpretMSSpectrum, Rdisop, beeswarm, Biobase

**RoxygenNote** 6.1.1

**NeedsCompilation** yes

**Author** Jan Lisec [aut, cre],
Friederike Hoffmann [aut]

**Repository** CRAN

**Date/Publication** 2019-06-18 13:40:03 UTC

# R topics documented:



---

`CalcMID` *CalcMID.*

---

## Description

`CalcMID` will compute a MID (Mass Isotopomer Distribution) based on measured ion intensities in GC-APCI-MS.

## Usage

```
CalcMID(int = NULL, fml = "", ratio = NULL, nmz = NULL,
  nbio = NULL)
```

## Arguments

| | |
|---|---|
| `int` | Vector of measured ion intensities of a fragment. |
| `fml` | Chemical formula of fragment. |
| `ratio` | If NULL M+H/M+ ratio will be determined from the data if necessary. Can be specified explicitly here. |
| `nmz` | Attached as attr to fml for CalcTheoreticalMDV. |
| `nbio` | Attached as attr to fml for CalcTheoreticalMDV. |

### Details

Let's assume we measured the ion intensities of all 3 isotopes of an individual compound containing 2 carbons and observe a vector of 978,22,0. We may calculate the enrichment (E) out of this data, i.e. the relative proportion of 13C vs total carbon which will amount to about 1.1 The equvivalent MID vector would be 1,0,0, indicating that the non-labeled isotopologue (where non-labeled means non-labled above the natural 1.1 During a labelling experiment we may change the measurement values in different ways (either labelling only one carbon or both), which potentially can translate into similar values for E being larger 1.1 The MIDs will provide additional information about the isotopolouge fraction which gave rise to the observed E's (cf. examples).

### Value

Percent representation of each isotopologue measured (=MIDs).

### Examples

```
#tbd
```

---

`CalcTheoreticalMDV` *CalcTheoreticalMDV.*

---

### Description

`CalcTheoreticalMDV` will compute the Mass Distribution Vectors of isotopologues as it is used for correction matrix in `CalcMID` computations.

### Usage

```
CalcTheoreticalMDV(fml = NULL)
```

### Arguments

| | |
|---|---|
| `fml` | The chemical formula of the compound. |

### Details

`CalcTheoreticalMDV` basically is a convenience function using Rdisop to generate the isotopologue distribution at natural abundance of 13C for a given formula. It will break this down into a matrix where the components of the MID constitute the rows and the expected relative ion intensities are within the columns. The number of exported ion intensities and MID components can be limited if numeric values for "nmz" and/or "nbio" provided as attributes with the formula.

### Value

A matrix with dimensions according to the attributes of `fml` or the number of carbons respectively.

## Examples

```
# standard distribution matrix
fml <- "C5H6Si1"
CalcTheoreticalMDV(fml=fml)
attr(fml,"nmz") <- 4
CalcTheoreticalMDV(fml=fml)
attr(fml,"nbio") <- 2
CalcTheoreticalMDV(fml=fml)
```

---

DeconvoluteSpectrum *DeconvoluteSpectrum.*

---

## Description

DeconvoluteSpectrum will evaluate a list of xcmsRaw objects at a given time (rt) and potential mass (mz1). The main purpose is to deconvolute the mass spectrum at rt including mz1.

## Usage

```
DeconvoluteSpectrum(dat = NULL, rt = NULL, rt_dev = 3, mz1 = NULL,
  mz_dev = 0.003, use.mz.adjust = FALSE, ionization = c("APCI",
  "ESI")[1], smooth = 0)
```

## Arguments

| | |
|---|---|
| dat | A list of xcmsRaws or an xcmsSet object. |
| rt | Retention time to search for maxima. |
| rt_dev | Allowed retention time window. |
| mz1 | If specified, ensure that this mass is included in the spectrum (assumed base peak). NULL otherwise. |
| mz_dev | Allowed mz deviation [Da]. |
| use.mz.adjust | Will adjust mz on an experiment wide basis. |
| ionization | Either APCI or ESI. Choice will modify some internal parameters and checks performed. |
| smooth | Smoothing parameter passed on to getMultipleBPC. |

## Details

Will test all mz at spectrum of base peak within range for co-apex, rt diff and ratio consistency/correlation over a set of samples.

## Value

A pseudo spectrum at rt (containing mz1 if specified). Effectively a 2-column matrix (mz, int) with rt as attribute

## Examples

```
# Please use examples from previous versions as xcms (and xcms objects)
# are no longer supported during CRAN checks leading to package rejection
# if included (and I do not know a work around). :(
```

---

`EvaluateCandidateListAgainstRawData` *EvaluateCandidateListAgainstRawData.*

---

## Description

`EvaluateCandidateListAgainstRawData` will analyze an xcmsSet result for mass pairs (mz1, mz2) with changes due to 13C incorporation.

## Usage

```
EvaluateCandidateListAgainstRawData(x = NULL, tp = NULL, gr = NULL,
  dat = NULL, dmz = 0.025, drt = 1, dEcut = 1, Pcut = 0.01,
  Icut = 1000, method = c("APCI", "ESI")[1], rolp = c("non", "pos",
  "neg", "all")[2], smooth = 0)
```

## Arguments

| | |
|---|---|
| `x` | Dataframe of results (output of EvaluatePairsFromXCMSet). |
| `tp` | Timepoint. |
| `gr` | group, e.g. different genotypes or concentrations. |
| `dat` | list of xcmsRaw's for deconvolution and plotting. |
| `dmz` | Allowed mass deviation in Da (for BPC extraction). |
| `drt` | Allowed rt deviation in seconds (for get extraction). |
| `dEcut` | Minimum required change in enrichment before a candidate ID is assigned. |
| `Pcut` | Maximum allowed P value before a candidate ID is assigned. |
| `Icut` | Minimum required median peak intensity before a candidate ID is assigned. |
| `method` | Either APCI or ESI. Choice will modify some internal parameters and checks performed. |
| `rolp` | RemoveOverLappingPeaks paramter. |
| `smooth` | Smoothing parameter passed to getMultipleBPC. |

## Details

This function will evaluate candidate mz pairs found within an xcmsSet object by EvaluatePairsFromXCMSSet against the raw measurement data. A special parameter is 'rolp' which can be set to 'non', 'pos', 'neg' or 'all'. It will influence the time performance of the function be determining how many peaks are effectively tested. If 'rolp' is set to 'non', no overlapping peaks will be skipped, every individual mz-pair will be sequentially evaluated (slow but most informative). If it is set to 'pos' or 'neg', overlapping peaks (determined by experiment wide deconvolution) will not be tested aditionally for positive or negative hits ('neg' is standard). If set to 'all' overlapping peaks will always be removed from the list of mz-pairs to be tested (fast).

## Value

A list of evaluation results.

## Examples

```
# Please use examples from previous versions as xcms (and xcms objects)
# are no longer supported during CRAN checks leading to package rejection
# if included (and I do not know a work around). :(
```

---

`EvaluatePairsFromXCMSSet` *EvaluatePairsFromXCMSSet.*

---

## Description

`EvaluatePairsFromXCMSSet` will analyze an xcmsSet result for mass pairs (mz1, mz2) with changes due to any 13C incorporation.

## Usage

```
EvaluatePairsFromXCMSSet(xg = NULL, tp = NULL, gr = NULL, drt = 1,
  dmz = 0.025, mz_iso = 1.00335, n = 6, method = c("APCI",
  "ESI")[1], specific_row = NULL, testing = FALSE, silent = FALSE)
```

## Arguments

| | |
|---|---|
| `xg` | xcmsSet object with group information. |
| `tp` | Timepoint information for all samples (obviously required, internally converted to factor). |
| `gr` | Group information for all samples, e.g. different genotypes or concentrations (optional, factor). |
| `drt` | Allowed rt deviation in time units of xcmsSet (usually seconds) to test for candidates. |
| `dmz` | Allowed mass deviation in Da. |

| | |
|---|---|
| mz_iso | Mass defect of the isotope under investigation. |
| n | Number of maximal incorporated carbons to test. |
| method | Currently APCI or ESI. If APCI, dmz will be modified depending on n (see details). |
| specific_row | A single row from groupval(xg) to process. |
| testing | Stop in function using browser() if specific_row is specified; can be a isotope number, i.e. 3 will stop at third isotope. |
| silent | Suppress warnings and console output if TRUE. |

## Details

Using 'APCI' as method assumes that (i) you analyze TMS-derivatized compounds and (ii) your MS resolution does not allow to seperate Si and C isotopes but reportes an intermediate mass as m/z. In this case you will find carbon isotopes below there expected masses, i.e. M+1 would be 1.001mDa apart from M+0 instead of 1.003. The effect is increased with isotope number, i.e. M+6 will be ~20mDa below the expected value. Hence, selecting method 'APCI' will combine your selected dmz with a allowed deviation due to Si-isotope caused mass shifts. Use 'ESI' if you are not sure if this effect takes place in your settings.

## Value

A dataframe with all observable pairs within the provided xcmsSet peak list including mean group intensities and P values.

---

# GenerateCandXLSX *GenerateCandXLSX.*

---

## Description

GenerateCandXLSX will produce a XLSX of a list containing test results objects.

## Usage

```
GenerateCandXLSX(res_list = NULL, xlsx_file = NULL, rejected = FALSE)
```

## Arguments

| | |
|---|---|
| res_list | A list of result objects (each testing an individual mz pair). |
| xlsx_file | File name. |
| rejected | Logical. Return rejected if TRUE. |

## Details

Not yet.

## Value

Candidate table as data.frame.

## Examples

```
#load evaluation result of example data
data(res_list)
#generate table within R (use xlsx_file to write to file)
str(GenerateCandXLSX(res_list))
GenerateCandXLSX(res_list)[,1:5]
```

---

GenerateQCPlots *GenerateQCPlots.*

---

## Description

GenerateQCPlots will produce QC plots for a list containing test results objects.

## Usage

```
GenerateQCPlots(res_list = NULL, pdf_file = NULL, mfrow = NULL,
  skip_plots = NULL)
```

## Arguments

| | |
|---|---|
| res_list | A list of result objects (each testing an individual mz pair). |
| pdf_file | Either APCI or ESI. Choice will modify some internal parameters and checks performed. |
| mfrow | If NULL automatically determined, otherwise useful to specify a layout. |
| skip_plots | NULL or numeric vector in which case plots with numbers in skip_plots will be empty. |

## Details

For individual candidates screen output is reasonable, otherwise a target PDF file should be specified.

## Value

NULL.

## Examples

```
#load evaluation result of example data
data(res_list)
#generate Figures on screen (use PDF output for mass evaluation)
GenerateQCPlots(res_list[1])
```

getMultipleBPC *getMultipleBPC.*

## Description

getMultipleBPC will extract multiple BPCs from an xcmsRaw object for a vector of mz within the limits given by rt, rt_dev and mz_dev.

## Usage

```
getMultipleBPC(x, mz = NULL, mz_dev = 0.005, rt = NULL, rt_dev = 2,
  zeroVal = NA, smooth = 0, returnEIC = FALSE)
```

## Arguments

| | |
|---|---|
| x | xcmsRaw object. |
| mz | mass vector. |
| mz_dev | allowed deviations (can be a single numeric, a vector, a matrix with one row (lower bound, upper bound) or a matrix with length(mz) rows giving lower and upper bound for each mz). |
| rt | target timepoint. |
| rt_dev | allowed window. |
| zeroVal | Set values <=0 to NA or keep as is with NULL. |
| smooth | Window size for moving average smoother, 0 = no smoothing. |
| returnEIC | Return EIC instead of BPC? |

## Details

While there are other functions to extract BPC information from raw data, this one is particularly useful to get all traces belonging to a isotopologue group. It will attach several derived values to the results object, i.e. describing the observed mass shift (deviation from expected value) which is helpful in QC for non-targeted tracer analyses.

## Value

A matrix with scan wise (rows) intensities for all requested masses (columns) as either EIC or BPC.

## References

Uses C code modified from XCMS (see citation("xcms")).

## Examples

```
# see \link{plotMID} for an example
```

## mz_shift_corrector *Predefined mass search windows to be used internally.*

### Description

This is a list defining windows for high res APCI or ESI instrumentation..

### Author(s)

Jan Lisec <jan.lisec@bam.de>

## plotBPC *plotBPC.*

### Description

`plotBPC` will plot for each item of a list of result-ojects from getMultipleBPC the BPC traces and the spectrum at the scan where the summed intensity of all ions is max.

### Usage

```
plotBPC(bpc = NULL, mfrow = NULL, skip_plots = NULL, ylim = NULL,
  col = NULL, ids = NULL, type = "both")
```

### Arguments

| | |
|---|---|
| `bpc` | A bpc object (list of intensity matrixes, rt x mz, including several attributes as attached by getMultipleBPC). |
| `mfrow` | Specify mfrow explicitely (is optimized internally if NULL to cover n=length(bpc)). |
| `skip_plots` | Allows to block certain subplots in the mfrow matrix to bettern align replicates. |
| `ylim` | Can be specified specifically, will be adjusted to overall min/max otherwise. |
| `col` | Specific color vector for masses may be provided. |
| `ids` | Specific plot ids may be explicitely provided. |
| `type` | Switch between co-plot of BPC and Spectrum ("both") or BPC alone ("bpc"). |

### Details

not yet

### Value

A plot to the graphics device and NULL as invisible.

## Examples

```
#load example raw data
data(res_list)
plotBPC(bpc = res_list[[1]][["bpc"]][c(1:2,13:14)])
```

---

plotMID *plotMID.*

---

## Description

plotMID will plot a Mass Isotopomer Distribution (MID) as calculated by CalcMID.

## Usage

```
plotMID(mid = NULL, gr = NULL, name = "unknown", contr = NULL,
  stackedbars = FALSE, subplot_ylim = 100, ...)
```

## Arguments

| | |
|---|---|
| mid | Matrix of measured ion intensities corrected using CalcMID. |
| gr | Groups, a factor. |
| name | Name of metabolite. |
| contr | Contrasts. Not yet clear if useful. |
| stackedbars | Alternative plotting layout using stacked bar plot. |
| subplot_ylim | Calculate ylim individually per subplot if 0, show full range in all subplots if 100 and limit to the minimal specified number otherwise. |
| ... | Further arguments to 'boxplot'. |

## Details

Not yet.

## Value

NULL.

## Examples

```
mid <- matrix(c(seq(0,0.3,0.1), seq(1,0.7,-0.1)), byrow=TRUE, nrow=2)
gr <- gl(2,2,labels=letters[1:2])
plotMID(mid=mid, gr=gr, name="Metabolite X")
plotMID(mid=mid, gr=gr, stackedbars=TRUE, las=1, col=2:3, xlab="MID")
```

---

`res_list` *The main results object of a non-targeted search for tracer incorporation.*

---

## Description

This is a list containing the evaluations results established based on processing example data with EvaluateCandidateListAgainstRawData.

## Author(s)

Jan Lisec <jan.lisec@bam.de>

---

`sam` *Sample table*

---

## Description

This data frame contains the sample definition of 18 samples from a larger experiment.

## Author(s)

Jan Lisec <jan.lisec@bam.de>

---

`xcms_cand` *Dataframe with putative candidates*

---

## Description

This data frame contains the analysis result of an xcmsSet which can not be provided via CRAN anymore using EvaluatePairsFromXCMSSet with respect to interesting m/z-pairs.

## Author(s)

Jan Lisec <jan.lisec@bam.de>

# Index