

# Package ‘HiResTEC’

January 20, 2025

**Type** Package

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

**Version** 0.62.3

**Date** 2024-02-23

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

**Description** Identifying labeled compounds in a  $^{13}\text{C}$ -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://github.com/janlisec/HiResTEC>

**Depends** R (>= 2.10.0)

**Imports** beeswarm, CorMID, InterpretMSSpectrum, openxlsx, plyr

**Encoding** UTF-8

**Language** en-US

**LazyData** TRUE

**RoxygenNote** 7.3.1

**NeedsCompilation** yes

**Author** Jan Lisec [aut, cre] (<<https://orcid.org/0000-0003-1220-2286>>),  
Friederike Hoffmann [aut]

**Repository** CRAN

**Date/Publication** 2024-02-26 09:20:09 UTC

## Contents

DeconvoluteSpectrum . . . . .	2
EvaluateCandidateListAgainstRawData . . . . .	3
EvaluatePairsFromXCMSSet . . . . .	5
GenerateCandXLSX . . . . .	6
GenerateQCPlots . . . . .	7
getMultipleBPC . . . . .	8
mz_shift_corrector . . . . .	9
plotBPC . . . . .	9
plotMID . . . . .	11
res_list . . . . .	12
sam . . . . .	12
xcms_cand . . . . .	13
<b>Index</b>	<b>14</b>

---

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 <a href="#">getMultipleBPC</a> .

### 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 <sup>13</sup>C 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 <a href="#">getMultipleBPC</a> .

## 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 by 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 additionally 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 <sup>13</sup>C 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 separate Si and C isotopes but reports an intermediate mass as m/z. In this case you will find carbon isotopes below their 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.

### 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).
## Not run:
load(xcms_cand)
head(xcms_cand[order(xcms_cand$P), ])

## End(Not run)
```

---

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	<i>GenerateQCPlots.</i>
-----------------	-------------------------

---

**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**

<code>res_list</code>	A list of result objects (each testing an individual mz pair).
<code>pdf_file</code>	Either APCI or ESI. Choice will modify some internal parameters and checks performed.
<code>mfrow</code>	If NULL automatically determined, otherwise useful to specify a layout.
<code>skip_plots</code>	NULL or numeric vector in which case plots with numbers in <code>skip_plots</code> 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 or NULL (default) to return the TIC.
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 time point or NULL (default) to use full scan time.
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{plotBPC} for an example
```

---

mz_shift_corrector	<i>Predefined mass search windows to be used internally.</i>
--------------------	--

---

**Description**

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

**Usage**

```
data(mz_shift_corrector)
```

**Format**

An object of class `list` of length 3.

**Source**

```
jan.lisec@bam.de
```

---

plotBPC	<i>plotBPC.</i>
---------	-----------------

---

**Description**

`plotBPC` will plot for each item of a list of result-objects 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",  
  ann = c("mdev", "mz", "none")  
)
```

## Arguments

bpc	A bpc object (list of intensity matrixes, rt x mz, including several attributes as attached by <a href="#">getMultipleBPC</a> ).
mfrow	Specify mfrow explicitly (is optimized internally if NULL to cover n=length(bpc)).
skip_plots	Allows to block certain subplots in the mfrow matrix to better 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 explicitly provided.
type	Switch between co-plot of BPC and Spectrum ("both") or BPC alone ("bpc").
ann	Select value to annotate peaks in spectrum. Usually the mass deviation from the expected value in mDa.

## 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)])  
plotBPC(bpc = res_list[[1]][["bpc"]][c(1:2, 13:14)], ann="mz")
```

---

plotMID	<i>plotMID.</i>
---------	-----------------

---

## Description

plotMID will plot a Mass Isotopomer Distribution (MID).

## Usage

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

## Arguments

mid	Matrix of corrected mass isotopomer distributions. Samples in columns, MID values in rows.
gr	Groups, a factor vector of length ncol(mid).
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' or 'barplot' (depending on 'stackedbars').

## Details

Not yet.

## Value

A plot.

## 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, xlab = "MID", legend.text = c("x", "y"))  
lt <- paste0("M", 0:1)
```

```

rownames(mid) <- lt
plotMID(mid = mid, gr = gr, stackedbars = TRUE, las = 1, xlab = "MID", legend.text = lt)
plotMID(mid = mid[, 2, drop = FALSE], stackedbars = TRUE, col = c(3, 4))
colnames(mid) <- paste0("S", 1:4)
gr2 <- gl(n = 1, k = 1, labels = "bla")
plotMID(mid = mid[, 2, drop = FALSE], gr = gr2, stackedbars = TRUE, name = NULL)
plotMID(mid = mid, gr = factor(colnames(mid)), stackedbars = TRUE, name = NULL)

```

---

res_list	<i>The main results object of a non-targeted search for tracer incorporation.</i>
----------	---

---

### Description

This is a list containing the evaluations results established based on processing example data with [EvaluateCandidateListAgainstRawData](#).

### Usage

```
data(res_list)
```

### Format

An object of class `list` of length 14.

### Source

jan.lisec@bam.de

---

sam	<i>Sample table</i>
-----	---------------------

---

### Description

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

### Usage

```
data(sam)
```

### Format

An object of class `data.frame` with 18 rows and 8 columns.

### Source

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.

**Usage**

```
data(xcms_cand)
```

**Format**

An object of class `data.frame` with 72 rows and 19 columns.

**Source**

jan.lisec@bam.de

# Index

## \* data

mz\_shift\_corrector, [9](#)

res\_list, [12](#)

sam, [12](#)

xcms\_cand, [13](#)

DeconvoluteSpectrum, [2](#)

EvaluateCandidateListAgainstRawData, [3](#),

[12](#)

EvaluatePairsFromXCMSSet, [4](#), [5](#), [13](#)

GenerateCandXLSX, [6](#)

GenerateQCPlots, [7](#)

getMultipleBPC, [3](#), [4](#), [8](#), [9](#), [10](#)

mz\_shift\_corrector, [9](#)

plotBPC, [9](#)

plotMID, [11](#)

res\_list, [12](#)

sam, [12](#)

xcms\_cand, [13](#)