

Package: assignments (via r-universe)

May 27, 2026

Title Molecular Formula Assignment For High Resolution ESI-MS Based Metabolomics Data

Version 1.0.2

Description A molecular formula assignment approach for electrospray ionisation high resolution mass spectrometry based metabolomics data.

Depends R (>= 3.5.0),

Imports cli, crayon, dplyr, furr, future, igraph, lubridate, magrittr, metabolysR, methods, mzAnnotation, purrr, rlang, stringr, tibble, tidygraph, tidyr

License GPL (>= 3)

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Collate parameters.R assignment.R correlations.R addIsoAssign.R transformationAssign.R relationships.R graph.R assign.R internals.R components.R feature_data.R assignments.R plotAdductDist.R plot_components.R plotSpectrum.R reexports.R zzz.R

Suggests testthat, covr, ggplot2, ggraph, ggrepel, ggtext, ggthemes, glue, graphlayouts, patchwork, knitr, rmarkdown

Remotes aberHRML/mzAnnotation, jassenfinch/metabolysR

URL <https://aberhrml.github.io/assignments/>

VignetteBuilder knitr

Config/pak/sysreqs cmake libglpk-dev make libicu-dev libuv1-dev libxml2-dev

Repository <https://aberhrml.r-universe.dev>

Date/Publication 2024-01-11 14:58:29 UTC

RemoteUrl <https://github.com/aberHRML/assignments>

RemoteRef HEAD

RemoteSha e11cad4a03df17e62b48e5f812a54f95a1ac4ff7

Contents

assignment	2
Assignment-class	3
assignmentParameters	3
AssignmentParameters-class	4
assignMFs	4
availableTechniques	6
calcCorrelations	6
feature_data	8
featureData	8
nodes	11
plotAdductDist	12
technique	14
Index	19

assignment	<i>Create an Assignment S4 class object</i>
------------	---

Description

Constructor methods for creating an object of S4 class Assignment.

Usage

```
assignment(feature_data, parameters, ...)

## S4 method for signature 'tbl_df,AssignmentParameters'
assignment(feature_data, parameters)

## S4 method for signature 'AnalysisData,AssignmentParameters'
assignment(feature_data, parameters)

## S4 method for signature 'Analysis,AssignmentParameters'
assignment(feature_data, parameters, type = c("pre-treated", "raw"))
```

Arguments

feature_data	a tibble or an object of S4 class AnalysisData or Analysis containing the feature intensity matrix of m/z for which to assign molecular formulas. See details.
parameters	an S4 object of class AssignmentParameters containing the parameters for molecular formula assignment
...	arguments to pass to the relevant method
type	type pre-treated or raw data on which to perform assignment when argument feature_data is of class Analysis

Value

An object of S4 class Assignment.

Examples

```
mf_assignments <- assignment(feature_data,assignmentParameters('FIE-HRMS'))
mf_assignments
```

Assignment-class *Assignment*

Description

An S4 class to store molecular formula assignment results.

Slots

log a list containing assignment logs
 data a tibble containing the m/z peak intensity matrix
 correlations a tibble containing the correlations analysis results
 relationships a tibble containing the calculated mathematical relationships
 addIsoAssign a list containing the results of the adduct and isotope assignment iterations
 transAssign a list containing the results of the transformation assignment iterationst
 assignments a tibble containing the assigned molecular formulas

assignmentParameters *Assignment parameters*

Description

Return the default molecular formula assignment parameters for a given analytical technique.

Usage

```
assignmentParameters(technique = availableTechniques())
```

Arguments

technique technique to use for assignment

Value

An object of S4 class AssignmentParameters

Examples

```
assignmentParameters('FIE-HRMS')
```

AssignmentParameters-class

S4 class for assignment parameters

Description

An S4 class to store assignment parameters.

Slots

technique the analytical technique

correlations_parameters a list of correlation parameters to be passed to metabolysR::correlations()

max_M the maximum molecular mass for which to assign molecular formulas

MF_rank_threshold rank threshold for molecular formula selection

ppm the parts per million error threshold

limit the atomic mass unit deviation limit for relationship calculation

RT_diff_limit the limit for retention time differences for correlated features in adduct and isotopic assignment

adducts a list of character vectors containing the adducts names. List element names should denote ionisation mode. The order that these adducts are provided denotes their expected relative importance to assignments with the first expected to be the most common and the last the least common within each ionisation mode.

isotopes a character vector of isotopes to use. Similarly to the adducts, their order denotes the expected commonality in the data.

transformations a character vector of transformations molecular formula changes

adduct_rules a tibble containing the adduct formation rules as returned by mzAnnotation::adduct_rules()

isotope_rules a tibble containing the isotope rules as returned by mzAnnotation::isotope_rules()

transformation_rules tibble containing the transformation rules as returned by mzAnnotation::transformation_rules()

assignMFs

Perform molecular formula assignment

Description

Perform automated molecular formula assignment.

Usage

```

assignMFs(
  feature_data,
  parameters = assignmentParameters("FIE-HRMS"),
  verbose = TRUE,
  ...
)

## S4 method for signature 'tbl_df'
assignMFs(
  feature_data,
  parameters = assignmentParameters("FIE-HRMS"),
  verbose = TRUE
)

## S4 method for signature 'AnalysisData'
assignMFs(
  feature_data,
  parameters = assignmentParameters("FIE"),
  verbose = TRUE
)

## S4 method for signature 'Analysis'
assignMFs(
  feature_data,
  parameters = assignmentParameters("FIE"),
  verbose = TRUE,
  type = c("pre-treated", "raw")
)

```

Arguments

feature_data	a tibble or an object of S4 class AnalysisData or Analysis containing the feature intensity matrix of m/z for which to assign molecular formulas. See details.
parameters	an S4 object of class AssignmentParameters containing the parameters for molecular formula assignment
verbose	should progress output be printed to the console
...	arguments to pass to the relevant method
type	pre-treated or raw data on which to perform assignment when argument feature_data is of S4 class Analysis

Details

If argument feature_data is specified as a tibble, this should be a feature intensity matrix where the columns are the m/z features to assign and the rows are the individual observations, with the cells as abundance values. The m/z features provided as column names should be in the form of

<ionisation_mode><m/z>@<retention_time>. Ionisation mode should be given as a prefix n or p for negative or positive ionisation modes respectively. Feature m/z should be provided to an accuracy of least 5 decimal places. The retention time portion (@<retention_time>) is only required for LC-MS data and should be provided in minutes.

Examples

```
plan(future::sequential)
p <- assignmentParameters('FIE-HRMS')

assignments <- assignMFs(feature_data,p)
```

availableTechniques *Available analytical techniques*

Description

The available analytical techniques for molecular formula assignment parameters.

Usage

```
availableTechniques()
```

Value

A character vector of technique names.

Examples

```
availableTechniques()
```

calcCorrelations *Molecular formula assignment methods*

Description

These methods provide access to performing the individual steps of the molecular formula assignment approach. See Details for more information of when it is best to use these instead of assignMFs().

Usage

```
calcCorrelations(assignment)

## S4 method for signature 'Assignment'
calcCorrelations(assignment)

addIsoAssign(assignment)

## S4 method for signature 'Assignment'
addIsoAssign(assignment)

transformationAssign(assignment)

## S4 method for signature 'Assignment'
transformationAssign(assignment)

calcRelationships(assignment)

## S4 method for signature 'Assignment'
calcRelationships(assignment)
```

Arguments

assignment an object of S4 class Assignment

Details

In circumstances where the molecular formula assignment approach has high memory requirements, such as where there are many correlations (> 2 million) or many high m/z (>700), it may be preferable to perform the assignment steps separately as opposed to using `assignMFs()`. This can reduce the memory overheads required to successfully assign molecular formulas to the data and also enable the possibility of objects to be saved and/or unloaded between the assignment steps where needed.

Value

An object of S4 class Assignment containing molecular formula assignments.

Examples

```
plan(future::sequential)
p <- assignmentParameters('FIE-HRMS')

mf_assignments <- assignment(feature_data,p)

mf_assignments <- mf_assignments %>%
  calcCorrelations() %>%
  calcRelationships() %>%
  addIsoAssign() %>%
  transformationAssign()
```

mf_assignments

feature_data	<i>Example feature data</i>
--------------	-----------------------------

Description

An example m/z peak intensity matrix containing total ion count normalised positive and negative mode flow infusion electrospray ionisation mass spectrometry *m/z* features.

Usage

```
feature_data
```

Format

A tibble containing 60 rows and 10 variables.

featureData	<i>Assignment accessors</i>
-------------	-----------------------------

Description

Access methods for Assignment S4 class

Usage

```
featureData(assignment)

## S4 method for signature 'Assignment'
featureData(assignment)

correlations(assignment)

## S4 method for signature 'Assignment'
correlations(assignment)

relationships(assignment)

## S4 method for signature 'Assignment'
relationships(assignment)

iterations(assignment)

## S4 method for signature 'Assignment'
```

```
iterations(assignment)

graph(assignment, iteration, type = c("selected", "all"))

## S4 method for signature 'Assignment'
graph(assignment, iteration, type = c("selected", "all"))

components(assignment, iteration, type = c("selected", "all"))

## S4 method for signature 'Assignment'
components(assignment, iteration, type = c("selected", "all"))

featureComponents(assignment, feature, type = c("selected", "all"))

## S4 method for signature 'Assignment'
featureComponents(assignment, feature, type = c("selected", "all"))

component(assignment, component, iteration, type = c("selected", "all"))

## S4 method for signature 'Assignment'
component(assignment, component, iteration, type = c("selected", "all"))

assignments(assignment)

## S4 method for signature 'Assignment'
assignments(assignment)

assignedData(assignment)

## S4 method for signature 'Assignment'
assignedData(assignment)

summariseAssignments(assignment)

## S4 method for signature 'Assignment'
summariseAssignments(assignment)
```

Arguments

assignment	S4 object of class Assignment
iteration	the assignment iteration
type	the graph type to return. filtered returns the assignment graph after component selection. all returns all assignment components.
feature	feature information to extract
component	component number to extract

Details

- featureData - Return the initially specified m/z feature data.
- correlations - Return the correlation analysis results.
- relationships - Return the calculated relationships.
- iterations - Return the assignment iteration performed.
- graph - Return a selected graph.
- components - Return the component information for an assignment iteration.
- featureComponents - Return the component information for a selected feature.
- component - Extract a component graph.
- assignments - Return the molecular formulas assigned to the m/z features.
- assignedData - Return the m/z peak intensity matrix with the molecular formula assignments included in the column names.
- summariseAssignments - Return a tibble of the assignments summarised by molecular formula.

Value

A tibble or tbl_graph containing assignment results depending on the method used.

Examples

```
plan(future::sequential)
p <- assignmentParameters('FIE-HRMS')

mf_assignments <- assignMFs(feature_data,p)

## Return feature data
featureData(mf_assignments)

## Return correlations
correlations(mf_assignments)

## Return relationships
relationships(mf_assignments)

## Return the available iterations
iterations(mf_assignments)

## Return a selected graph
graph(mf_assignments, 'A&I1')

## Return a component information for a selected graph
components(mf_assignments, 'A&I1')

## Return a component information for a selected feature
featureComponents(mf_assignments, 'n191.01962')

## Extract a component graph
```

```
component(mf_assignments,1,'A&I1')

## Return assignments
assignments(mf_assignments)

## Return an m/z intensity matrix with the assignments included
## in the column names
assignedData(mf_assignments)

## Return the assignments summarised by molecular formula
summariseAssignments(mf_assignments)
```

nodes	<i>Extract graph attributes</i>
-------	---------------------------------

Description

Extract node or edge attributes from a *tidygraph* `tbl_graph` object.

Usage

```
nodes(graph)
```

```
edges(graph)
```

Arguments

graph object of class `tbl_graph`

Examples

```
a_graph <- tidygraph::tbl_graph(
  nodes = data.frame(
    name = c('a', 'b', 'c')
  ),
  edges = data.frame(
    from = c(1,2),
    to = c(2,3),
    type = c(1,2)
  )
)

## Extract graph nodes
nodes(a_graph)

## Extract graph edges
edges(a_graph)
```

plotAdductDist	<i>Plot assignment results</i>
----------------	--------------------------------

Description

Plot molecular formula assignment results.

Usage

```
plotAdductDist(assignment)

## S4 method for signature 'Assignment'
plotAdductDist(assignment)

plotComponent(
  assignment,
  component,
  iteration,
  type = c("selected", "all"),
  label_size = 3,
  axis_offset = 0.1,
  border = NA,
  highlight = NA
)

## S4 method for signature 'Assignment'
plotComponent(
  assignment,
  component,
  iteration,
  type = c("selected", "all"),
  label_size = 3,
  axis_offset = 0.1,
  border = NA,
  highlight = NA
)

plotFeatureComponents(
  assignment,
  feature,
  iteration,
  type = c("all", "selected"),
  max_components = 6,
  label_size = 3,
  axis_offset = 0.1
)
```

```
## S4 method for signature 'Assignment'
plotFeatureComponents(
  assignment,
  feature,
  iteration,
  type = c("all", "selected"),
  max_components = 6,
  label_size = 2,
  axis_offset = 0.05
)

plotSpectrum(assignment, MF)

## S4 method for signature 'Assignment'
plotSpectrum(assignment, MF)
```

Arguments

assignment	an object of S4 class Assignment
component	component number to plot
iteration	the assignment iteration of the component or components
type	the graph type to return. selected returns the assignment graph after component selection. all returns all assignment components.
label_size	node label size
axis_offset	axis proportion by which to increase axis limits. Prevents cut off of node labels.
border	specify a plot border colour
highlight	specify a feature node to highlight
feature	the m/z feature to plot
max_components	the maximum number of components to plot
MF	the assigned molecular formula to plot

Details

- plotComponent - Plot a molecular formula component graph.
- plotFeatureComponents - Plot the possible component graphs for a given feature.
- plotAdductDist - Plot frequency distributions of the assigned adducts.
- plotSpectrum - Plot the spectrum of an assigned molecular formula.

Examples

```
library(ggraph)
plan(future::sequential)
p <- assignmentParameters('FIE-HRMS')

mf_assignments <- assignMFs(feature_data,p)
```

```
## Plot a component
plotComponent(mf_assignments,1,'A&I1')

## Plot the components for a feature
plotFeatureComponents(mf_assignments,'n191.01962','A&I1')

## Plot the adduct distributions
plotAdductDist(mf_assignments)

## Plot the spectrum of an assigned molecular formula
plotSpectrum(mf_assignments,'C6H807')
```

technique

Parameter get and set methods

Description

Get and set methods for the AssignmentParameters S4 class.

Usage

```
technique(x)

## S4 method for signature 'AssignmentParameters'
technique(x)

correlationsParameters(x)

## S4 method for signature 'AssignmentParameters'
correlationsParameters(x)

correlationsParameters(x) <- value

## S4 replacement method for signature 'AssignmentParameters,list'
correlationsParameters(x) <- value

limit(x)

## S4 method for signature 'AssignmentParameters'
limit(x)

limit(x) <- value

## S4 replacement method for signature 'AssignmentParameters'
limit(x) <- value

maxM(x)
```

```
## S4 method for signature 'AssignmentParameters'
maxM(x)

maxM(x) <- value

## S4 replacement method for signature 'AssignmentParameters'
maxM(x) <- value

MFrankThreshold(x)

## S4 method for signature 'AssignmentParameters'
MFrankThreshold(x)

MFrankThreshold(x) <- value

## S4 replacement method for signature 'AssignmentParameters'
MFrankThreshold(x) <- value

ppm(x)

## S4 method for signature 'AssignmentParameters'
ppm(x)

ppm(x) <- value

## S4 replacement method for signature 'AssignmentParameters'
ppm(x) <- value

isotopes(x)

## S4 method for signature 'AssignmentParameters'
isotopes(x)

isotopes(x) <- value

## S4 replacement method for signature 'AssignmentParameters'
isotopes(x) <- value

adducts(x)

## S4 method for signature 'AssignmentParameters'
adducts(x)

adducts(x) <- value

## S4 replacement method for signature 'AssignmentParameters'
adducts(x) <- value
```

```
transformations(x)

## S4 method for signature 'AssignmentParameters'
transformations(x)

transformations(x) <- value

## S4 replacement method for signature 'AssignmentParameters'
transformations(x) <- value

adductRules(x)

## S4 method for signature 'AssignmentParameters'
adductRules(x)

adductRules(x) <- value

## S4 replacement method for signature 'AssignmentParameters'
adductRules(x) <- value

isotopeRules(x)

## S4 method for signature 'AssignmentParameters'
isotopeRules(x)

isotopeRules(x) <- value

## S4 replacement method for signature 'AssignmentParameters'
isotopeRules(x) <- value

transformationRules(x)

## S4 method for signature 'AssignmentParameters'
transformationRules(x)

transformationRules(x) <- value

## S4 replacement method for signature 'AssignmentParameters'
transformationRules(x) <- value
```

Arguments

x	S4 object of class AssignmentParameters
value	the value to set

Details

- technique - Get the analytical technique.

- `correlationsParameters` - Get or set the correlation analysis parameters to be passed to `metabolysR::correlations()`.
- `limit` - Get or set the atomic mass unit limit for relationship calculation.
- `maxM` - Get or set the maximum molecular mass limit for which to assign molecular formulas.
- `MFrankThreshold` - Get or set the molecular formula rank threshold for molecular formula selection.
- `ppm` - Get or set the parts per million error threshold.
- `isotopes` - Get or set the isotope names. The order in which these are specified denotes the expected relative commonality within the data set.
- `adducts` - Get or set the adduct names for the ionisation modes. The order in which these are specified denotes the expected relative commonality within the data set for each ionisation mode.
- `transformations` - Get or set the transformation molecular formula changes.
- `isotopeRules` - Get or set the isotope rules table. The format of this tibble should match that of `mzAnnotation::isotope_rules()`.
- `adductRules` - Get or set the adduct rules table. The format of this tibble should match that of `mzAnnotation::adduct_rules()`.
- `techniqueRules` - Get or set the transformation rules table. The format of this tibble should match that of `mzAnnotation::transformation_rules()`.

Examples

```
assignment_parameters <- assignmentParameters('FIE')

## Return the analytical technique
technique(assignment_parameters)

## Return correlations parameters
correlationsParameters(assignment_parameters)

## Set correlations parameters
correlationsParameters(assignment_parameters)$minCoef <- 0.75

## Return limit
limit(assignment_parameters)

## Set limit
limit(assignment_parameters) <- 0.002

## Return max M
maxM(assignment_parameters)

## Set max M
maxM(assignment_parameters) <- 500

## Return MF rank threshold
MFrankThreshold(assignment_parameters)
```

```
## Set MF rank threshold
MFrankThreshold(assignment_parameters) <- 3

## Return ppm
ppm(assignment_parameters)

## Set ppm
ppm(assignment_parameters) <- 3

## Return isotopes
isotopes(assignment_parameters)

## Set isotopes
isotopes(assignment_parameters) <- '13C'

## Return adducts
adducts(assignment_parameters)

## Set adducts
adducts(assignment_parameters) <- list(n = c('[M-H]1-', '[M+Cl]1-'),
                                       p = c('[M+H]1+', '[M+K]1+'))

## Return transformations
transformations(assignment_parameters)

## Set transformations
transformations(assignment_parameters) <- "M - [O] + [NH2]"

## Return adduct rules
adductRules(assignment_parameters)

## Set adduct rules
adductRules(assignment_parameters) <- mzAnnotation::adduct_rules()

## Return isotope rules
isotopeRules(assignment_parameters)

## Set isotope rules
isotopeRules(assignment_parameters) <- mzAnnotation::isotope_rules()

## Return transformation rules
transformationRules(assignment_parameters)

## Set transformation rules
transformationRules(assignment_parameters) <- mzAnnotation::transformation_rules()
```

Index

- * **datasets**
 - feature_data, 8
- addIsoAssign (calcCorrelations), 6
- addIsoAssign, Assignment-method (calcCorrelations), 6
- adductRules (technique), 14
- adductRules, AssignmentParameters-method (technique), 14
- adductRules<- (technique), 14
- adductRules<-, AssignmentParameters-method (technique), 14
- adducts (technique), 14
- adducts, AssignmentParameters-method (technique), 14
- adducts<- (technique), 14
- adducts<-, AssignmentParameters-method (technique), 14
- assignedData (featureData), 8
- assignedData, Assignment-method (featureData), 8
- assignment, 2
- assignment, Analysis, AssignmentParameters-method (assignment), 2
- assignment, AnalysisData, AssignmentParameters-method (assignment), 2
- assignment, tbl_df, AssignmentParameters-method (assignment), 2
- Assignment-class, 3
- assignmentParameters, 3
- AssignmentParameters-class, 4
- assignments (featureData), 8
- assignments, Assignment-method (featureData), 8
- assignMFs, 4
- assignMFs, Analysis-method (assignMFs), 4
- assignMFs, AnalysisData-method (assignMFs), 4
- assignMFs, tbl_df-method (assignMFs), 4
- availableTechniques, 6
- calcCorrelations, 6
- calcCorrelations, Assignment-method (calcCorrelations), 6
- calcRelationships (calcCorrelations), 6
- calcRelationships, Assignment-method (calcCorrelations), 6
- component (featureData), 8
- component, Assignment-method (featureData), 8
- components (featureData), 8
- components, Assignment-method (featureData), 8
- correlations (featureData), 8
- correlations, Assignment-method (featureData), 8
- correlationsParameters (technique), 14
- correlationsParameters, AssignmentParameters-method (technique), 14
- correlationsParameters<- (technique), 14
- correlationsParameters<-, AssignmentParameters, list-method (technique), 14
- edges (nodes), 11
- feature_data, 8
- featureComponents (featureData), 8
- featureComponents, Assignment-method (featureData), 8
- featureData, 8
- featureData, Assignment-method (featureData), 8
- graph (featureData), 8
- graph, Assignment-method (featureData), 8
- isotopeRules (technique), 14
- isotopeRules, AssignmentParameters-method (technique), 14
- isotopeRules<- (technique), 14
- isotopeRules<-, AssignmentParameters-method (technique), 14

- isotopes (technique), 14
- isotopes, AssignmentParameters-method
(technique), 14
- isotopes<- (technique), 14
- isotopes<-, AssignmentParameters-method
(technique), 14
- iterations (featureData), 8
- iterations, Assignment-method
(featureData), 8

- limit (technique), 14
- limit, AssignmentParameters-method
(technique), 14
- limit<- (technique), 14
- limit<-, AssignmentParameters-method
(technique), 14

- maxM (technique), 14
- maxM, AssignmentParameters-method
(technique), 14
- maxM<- (technique), 14
- maxM<-, AssignmentParameters-method
(technique), 14
- MFrankThreshold (technique), 14
- MFrankThreshold, AssignmentParameters-method
(technique), 14
- MFrankThreshold<- (technique), 14
- MFrankThreshold<-, AssignmentParameters-method
(technique), 14

- nodes, 11

- plotAdductDist, 12
- plotAdductDist, Assignment-method
(plotAdductDist), 12
- plotComponent (plotAdductDist), 12
- plotComponent, Assignment-method
(plotAdductDist), 12
- plotFeatureComponents (plotAdductDist),
12
- plotFeatureComponents, Assignment-method
(plotAdductDist), 12
- plotSpectrum (plotAdductDist), 12
- plotSpectrum, Assignment-method
(plotAdductDist), 12
- ppm (technique), 14
- ppm, AssignmentParameters-method
(technique), 14
- ppm<- (technique), 14

- ppm<-, AssignmentParameters-method
(technique), 14

- relationships (featureData), 8
- relationships, Assignment-method
(featureData), 8

- summariseAssignments (featureData), 8
- summariseAssignments, Assignment-method
(featureData), 8

- technique, 14
- technique, AssignmentParameters-method
(technique), 14
- transformationAssign
(calcCorrelations), 6
- transformationAssign, Assignment-method
(calcCorrelations), 6
- transformationRules (technique), 14
- transformationRules, AssignmentParameters-method
(technique), 14
- transformationRules<- (technique), 14
- transformationRules<-, AssignmentParameters-method
(technique), 14
- transformations (technique), 14
- transformations, AssignmentParameters-method
(technique), 14
- transformations<- (technique), 14
- transformations<-, AssignmentParameters-method
(technique), 14