Package 'Rdisop'

April 4, 2025

Title Decomposition of Isotopic Patterns

Version 1.67.5

Date 2025-01-21

Description In high resolution mass spectrometry (HR-MS), the measured masses can be decomposed into potential element combinations (chemical sum formulas). Where additional mass/intensity information of respective isotopic peaks is available, decomposition can take this information into account to better rank the potential candidate sum formulas. To compare measured mass/intensity information with the theoretical distribution of candidate sum formulas, the latter needs to be calculated.
This package implements fast algorithms to address both tasks, the calculation of isotopic distributions for arbitrary sum formulas (assuming a HR-MS resolution of roughly 30,000), and the ranked list of sum formulas fitting an observed peak or isotopic peak set.

Depends R (>= 2.10), Rcpp

LinkingTo Rcpp

Suggests knitr, rmarkdown, RUnit, testthat (>= 3.0.0)

SystemRequirements None

License GPL-2

StagedInstalll no

URL https://github.com/sneumann/Rdisop

BugReports https://github.com/sneumann/Rdisop/issues/new

biocViews ImmunoOncology, MassSpectrometry, Metabolomics

Config/testthat/edition 3

Encoding UTF-8

Language en-US

VignetteBuilder knitr

RoxygenNote 7.3.2

LazyData true

git_url https://git.bioconductor.org/packages/Rdisop

git_branch devel

git_last_commit e5e778d

git_last_commit_date 2025-02-14

Repository Bioconductor 3.21

Date/Publication 2025-04-03

Author Anton Pervukhin [aut],
Steffen Neumann [aut, cre] (ORCID:
<https://orcid.org/0000-0002-7899-7192>),
Jan Lisec [ctb] (ORCID: <https://orcid.org/0000-0003-1220-2286>),
Miao Yu [ctb],
Roberto Canteri [ctb]

Maintainer Steffen Neumann <sneumann@ipb-halle.de>

Contents

`addMolecules` *Add/subtract sum formulas*

Description

Simple arithmetic modifications of sum formulas.

Usage

```
addMolecules(formula1, formula2, elements = NULL, maxisotopes = 10)

subMolecules(formula1, formula2, elements = NULL, maxisotopes = 10)
```

Arguments

`formula1`	Sum formula (can be a vector).
`formula2`	Sum formula.
`elements`	List of allowed chemical elements, defaults to full periodic system of elements. See `initializeElements`.
`maxisotopes`	Maximum number of isotopes shown in the resulting molecules.

Details

'addMolecules' adds the second argument to the first. 'subMolecules' subtracts the second argument from the first. This can be useful to revert e.g. adduct/fragment formation found in ESI mass spectrometry, or to mimic simple chemical reactions. No chemical checks are performed.

Value

The input vector formula1 will be converted into molecule objects similar to function `getMolecule`. However, the results will be amended or reduced by formula2.

Examples

```
# Remove the proton-adduct from Ethanol
subMolecules("C2H7O", "H")
```

`decomposeIsotopes` *Mass Decomposition of Isotope Patterns*

Description

Calculate the elementary compositions from an exact mass or isotope pattern, obtained e.g. by FTICR or TOF mass spectrometers.

Usage

```
decomposeIsotopes(
  masses,
  intensities,
  ppm = 2,
  mzabs = 1e-04,
  elements = NULL,
  filter = NULL,
  z = 0,
  maxisotopes = 10,
  minElements = "C0",
  maxElements = "C999999"
)

decomposeMass(
  mass,
  ppm = 2,
  mzabs = 1e-04,
  elements = NULL,
  filter = NULL,
  z = 0,
  maxisotopes = 10,
```

```
  minElements = "C0",
  maxElements = "C999999"
)

isotopeScore(molecule, masses, intensities)
```

Arguments

masses	A vector of masses (or m/z values) of an isotope cluster.
intensities	Absolute or relative intensities of the masses peaks.
ppm	Allowed deviation of hypotheses from given mass.
mzabs	Absolute deviation in Dalton (mzabs and ppm will be added).
elements	List of allowed chemical elements, defaults to CHNOPS. See initializeElements.
filter	NYI, will be a selection of DU, DBE and Nitrogen rules.
z	Charge z of m/z peaks for calculation of real mass, keep z=0 for auto-detection.
maxisotopes	Maximum number of isotopes shown in the resulting molecules.
minElements	Molecular formula, defining lower boundaries of allowed elements.
maxElements	Molecular formula, defining upper boundaries of allowed elements.
mass	A single mass (or m/z value).
molecule	An initialized molecule as returned by getMolecule() or the decomposeMass() and decomposeIsotopes() functions.

Details

Sum formulas are calculated which explain the given mass or isotope pattern.

Value

A list of molecules, which contain the sub-lists ‘formulas‘ potential formulas, ‘exactmass‘ exact mass of each hypothesis (not monoisotopic), ‘score‘ calculated score, ‘isotopes‘ a list of isotopes.

Author(s)

Steffen Neumann <sneumann@IPB-Halle.DE>

References

For a description of the underlying IMS see citation("Rdisop")

Examples

```
# query some measurement values from a Glutamate peak which will return two
# suggested/potential sum formulas
m <- c(147.0529, 148.0563, 149.0612)
i <- c(0.91, 0.06, 0.01)
mol <- decomposeIsotopes(m, i, maxisotopes = 3)
getFormula(mol)
```

```
# Rdisop returns the scores (how well does the exact data match the measured
# data) in a normalized fashion, but you can calculate the raw scores
getScore(mol)
isotopeScore(mol, m, i)

# using a 5 mDa window, the number of potential candidates is increased to 26
getFormula(decomposeIsotopes(m, i, mzabs = 0.005))

# elemental ranges can be specified to affect the result
# use maxElements to exclude all of the above suggestions containing more
# than one S and/or one P.
getFormula(decomposeIsotopes(m, i, mzabs = 0.005, maxElements = "S1P1"))
```

`getMolecule` *Calculate mass and isotope information for a molecule given as sum formula*

Description

Parse the sum formula and calculate the theoretical exact mass and the isotope distribution for an approximate MS resolution of 20,000 (i.e. not providing the isotopic fine structure).

Usage

```
getMolecule(formula, elements = NULL, z = 0, maxisotopes = 10)

getMass(molecule)

getMonoisotopic(molecule)

getIsotope(molecule, index)

getFormula(molecule)

getScore(molecule)

getValid(molecule)
```

Arguments

`formula`	Sum formula.
`elements`	List of allowed chemical elements, defaults to full periodic system of elements. See `initializeElements`.
`z`	Charge z of molecule for exact mass calculation.
`maxisotopes`	Maximum number of isotopes shown for the resulting molecule.

molecule	An initialized molecule as returned by getMolecule() or the decomposeMass() and decomposeIsotope() functions.
index	Return the n-th isotope mass/abundance pair(s) of the molecule

Details

getMolecule() will parse the sum formula and calculate the exact mass and the isotope distribution. The exact mass is the mass of the most abundant isotope and is not identical with the monoisotopic mass. The latter can be extracted using the function 'getMonoisotopic()'. This function can also be supplied with a vector of chemical formulas directly (in case that the isotopic distribution is of no interest). Since of version 1-65-3, if a charge is specified, the exact mass of the molecule will be reduced or increased by n-times the electron mass (depending on the sign). Also, isotopic masses will additionally be divided by the charge specified to reflect what would be measured in HR-MS.

Value

A list containing the elements 'formula' (repeated sum formula), 'mass' exact mass of molecule, 'score' probability, for given molecules a dummy value which is always 1.0, 'valid' result of neutrogen rule check, 'isotopes' a list of isotope masses and abundances.

Author(s)

Steffen Neumann <sneumann@IPB-Halle.DE>

References

For a description of the underlying IMS see citation("Rdisop")

Examples

```
# Ethanol
getMolecule("C2H6O")
```

---

initializeElements *Initialize (a subset of) elements of the periodic system of elements (PSE)*

Description

Initialize the information about name, mass and isotopes. To reduce the number of decomposition hypotheses, subsets of elements can be created.

Usage

```
initializeElements(names, method = c("NIST", "IUPAC"))

.getElement(name, elements = NULL)

initializeCHNOPS()

initializeCHNOPSMgKCaFe()

initializeCHNOPSNaK()

initializePSE(method = c("NIST", "IUPAC"))

initializeCharges()
```

Arguments

names	Vector of element names within PSE.
method	Use isotope mass and abundance data from either "NIST" (default) or "IUPAC".
name	Chemical element name (abbr.).
elements	Character vector of chemical element names.

Details

These functions return full, pre-defined or user-defined (sub-) lists of elements.

Value

A list with the elements ‘name‘ repeated sum formula, ‘mass‘ nominal mass of molecule, ‘isotope‘ a list of isotopes. The function ‘initializeCharges‘ is special, since it allows to parse charges as shown in examples.

Author(s)

Steffen Neumann <sneumann@IPB-Halle.DE>

References

For a description of the underlying IMS see citation("Rdisop"). Isotope patterns were obtained through wikipedia.org

Examples

```
initializeCHNOPS()
getMolecule("H3O+", elements=c(initializeCHNOPS(),initializeCharges()))
```

isotopes *isotope information.*

Description

isotope information.

Usage

```
isotopes
```

Format

A data frame of 4 columns for 398 chemical isotopes.

element The element name.

isotope The isotope name.

mass The absolute mass of this isotope in Dalton.

abundance The absolute abundance of this isotope.

Source

Isotope patterns were obtained through wikipedia.org.

mono_masses *Monoisotopic mass information..*

Description

Monoisotopic mass information..

Usage

```
mono_masses
```

Format

A named vector containing for each chemical element the most abundant isotope mass.

Source

Isotope patterns were obtained through wikipedia.org.

Index