Package 'CorMID'

May 3, 2022

R topics documented:

CalcTheoreticalMDV   *CalcTheoreticalMDV.*

Description

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

Usage

```
CalcTheoreticalMDV(fml = NULL, nbio = NULL, nmz = NULL)
```

Arguments

fml	The chemical formula of the compound.
nbio	Provide the number of biological carbon within fml explicitly.
nmz	Provide the number of measured isotopes of fml explicitly.

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" are provided as parameters.

Value

A matrix of theoretical mass distribution vectors.

Examples

```
# standard distribution matrix
fml <- "C5H6Si1"
CalcTheoreticalMDV(fml=fml)

# extend to more columns (number of measured ions) if required
CalcTheoreticalMDV(fml=fml, nmz=4)

# limit to a smaller number of biological carbon (i.e. if compounds are silylated)
CalcTheoreticalMDV(fml=fml, nmz=4, nbio=2)
```

CorMID   *CorMID.*

Description

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

Usage

```
CorMID(
  int = NULL,
  fml = "",
  r = NULL,
  penalize = 7,
  mid_fix = NULL,
  trace_steps = FALSE,
  prec = 0.01
)
```

Arguments

`int`	Named numeric vector of measured ion intensities of a fragment. Names will give position of values relative to M+H (see details).
`fml`	Chemical formula of the fragment as string.
`r`	Either a character vector giving fragments to be considered OR a named numeric giving relative amounts of fragments OR NULL (all known fragments will be estimated) OR a 2-row matrix giving the lower and upper allowed ratio (see examples).
`penalize`	Numeric exponent penalizing solutions with low M+H occurence. Formula is $1+3*(1-x)^{penalty}$. NA to omit penalizing.
`mid_fix`	May provide a numeric vector used as a given MID. Allows to estimate *r* individually.
`trace_steps`	For testing purposes. Print the results of intermediate steps to console.
`prec`	Precision of the estimation of MID, set to 1% as default.

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 natural 13C abundance) under standard conditions. The equvivalent corMID vector would be 1,0,0, indicating that the non-labeled isotopologue (where non-labeled means non-labled above the natural 1.1%) is the only component observed. 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 isotopologue fraction which gave rise to the observed **E**'s (cf. examples). The *r* parameter indicates an overlay of chemical rearrangements which may occur.

Value

Estimated percent representation of each isotopologue measured (corMID).

References

https://doi.org/10.3390/metabo12050408

Examples

```
# make up some fake measurement data for Pyruvic acid 2TMS with 3 biological carbon
# assuming 10% labelling at M3 and 2 fragments
fml <- "C9H20O3Si2"
mid <- c(0.9,0,0,0.1)
r <- unlist(list("M+H"=0.8, "M+H2O-CH4"=0.2))
int <- CorMID::recMID(mid=mid, r=r, fml=fml)
plot(int)

# full estimation of M and r
CorMID::CorMID(int=int, fml=fml)

# get an improved result setting r to the correct values
CorMID::CorMID(int=int, fml=fml, r=r, prec=0.0001)

# provoke a wrong estimation using a fixed r
CorMID::CorMID(int=int, fml=fml, r=unlist(list("M+H"=1)))

# calculate r if you know the true corMID for a compound
r <- attr(CorMID::CorMID(int=int, fml=fml, mid_fix=c(0.9,0,0,0.1)), "ratio")
round(CorMID::CorMID(int=int, fml=fml, r=r, prec=0.0001),3)

# deal with missing intensity values
CorMID::CorMID(int=int[-3], fml=fml)


# perform estimation with banded r and observation of optimization steps
r <- matrix(c(0.5,1,0,0.5,0,0.5), nrow=2, dimnames=list(NULL,c("M+H","M+","M+H2O-CH4")))
CorMID::CorMID(int=int, fml=fml, r=r, trace=TRUE)

#process Gln data from publication
utils::data("prep", package = "CorMID")
int <- prep[[24]][["int"]][,6]
fml <- prep[[24]]$fml
CorMID::CorMID(int=int, fml=fml, trace=TRUE)

# check the effect of the penalize parameter on selection of adducts
int <- c(1560, 119203, 41927, 16932, 4438)
names(int) <- c(-2, 0, 1, 2, 3)
```

```
fml <- "C19H37NO4Si3"
CorMID::CorMID(int=int, fml=fml, r=NULL, trace=TRUE)
CorMID::CorMID(int=int, fml=fml, r=NULL, trace=TRUE, penalize=7)
```

CountChemicalElements   *CountChemicalElements.*

Description

`CountChemicalElements` will split a character (chemical formula) into its elements and count their occurence.

Usage

```
CountChemicalElements(x = NULL, ele = NULL)
```

Arguments

x	Chemical formula.
ele	Character vector of elements to count particularly or counting all contained in string if NULL.

Details

No testing for any chemical alphabet is performed. Elements may occur several times and will be summed up in this case without a warning. Information within brackets, i.e. [13]C will be removed prior to counting together with other symbols not contained in the R set 'alnum'. The result is filtered and oredered according to parameter 'ele' if provided.

Value

A named numeric with counts for all contained or specified elements.

Examples

```
# count every element
CountChemicalElements("C3H7Cl")

# remove additional symbols and sum up redundant elements (here 'C')
CountChemicalElements("[13]CC2H8Cl+")

# count specific elements and return in specified order
CountChemicalElements("[13]CC2H8Cl+", ele=c("Cl","O","H"))

# apply on a vector of formulas using sapply
sapply(c("C3H7Cl", "[13]CC2H8Cl+"), CorMID::CountChemicalElements, ele=c("Cl","O","H"))
```

| prep | *prep.* |

Description

prep.

Usage

```
data(prep)
```

Format

A list of example metabolites.

Source

IZ_Exp05 experiment

| recMID | *recMID.* |

Description

recMID will reconstruct a measured GC-APCI-MS spectrum of a compound given its true MID and the fragment ratio.

Usage

```
recMID(mid = NULL, r = list(`M+H` = 1), fml = NULL, cutoff = 0.001)

## S3 method for class 'recMID'
plot(x, ...)
```

Arguments

mid	A numeric vector with sum=1 and length of C atoms +1.
r	Fragment ratios. A numeric vector with sum=1.
fml	A compound formula.
cutoff	Remove values below this threshold from output vector.
x	Object of class recMID.
...	Further plotting parameters.

Details

recMID is basically the inverse function to CorMID. Providing a specific chemical formula together with information regarding the true MID and r, this function will compute a vector of ion intensities which can be expected in a GC-APCI-MS analysis for this cmpound.

Value

A reconstructed MID.

Examples

```
fml <- "C9H20O3Si2"
mid <- c(0.9,0,0,0.1)
r <- list("M+H"=0.8, "M-H"=0.1, "M+H2O-CH4"=0.1)
(rMID <- CorMID::recMID(mid=mid, r=r, fml=fml))
plot(rMID)
plot(x = rMID, ylim=c(0,max(rMID)))
plot(x = rMID, xlim=c(-2,12), ylim=NULL, col=2, lwd=12, las=2, xlab="label")

CorMID::CorMID(int = rMID, fml=fml, prec=0.001, r=unlist(r), trace_steps = TRUE)
```

Index