Package ‘BRAIN’

March 28, 2017

Type Package

Title Baffling Recursive Algorithm for Isotope distribution calculations

Version 1.20.0

Date 2013-01-21

Author Piotr Dittwald, with contributions of Dirk Valkenborg and Jurgen Claesen

Maintainer Piotr Dittwald <piotr.dittwald@mimuw.edu.pl>

Depends R (>= 2.8.1), PolynomF, Biostrings, lattice

Description Package for calculating aggregated isotopic distribution and exact center-masses for chemical substances (in this version composed of C, H, N, O and S). This is an implementation of the BRAIN algorithm described in the paper by J. Claesen, P. Dittwald, T. Burzykowski and D. Valkenborg.

License GPL-2

biocViews MassSpectrometry, Proteomics

NeedsCompilation no

R topics documented:

- BRAIN-package
- calculateAverageMass
- calculateIsotopicProbabilities
- calculateMonoisotopicMass
- calculateNrPeaks
- getAtomsFromSeq
- useBRAIN

Index 10
Implementation of BRAIN (Baffling Recursive Algorithm for Isotope distribution calculations)

Description

This package implements BRAIN (Baffling Recursive Algorithm for Isotope distribution calculations) as described in full details by Claesen et al. The algorithm uses an algebraic approach (Viete’s formulas, Newton identities [Macd]) which is especially useful for large molecules due to its advantageous scaling properties. This version of the package provides functions for calculating the aggregated isotopic distribution and center-masses for each aggregated isotopic variant for chemical components built from carbon, hydrogen, oxygen, nitrogen and sulfur (e.g. peptides). The natural abundances and molecular masses for stable isotopes of C, H, N, O, S are taken from IUPAC 1997 [Rosm].

Details

<table>
<thead>
<tr>
<th>Package:</th>
<th>BRAIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Package</td>
</tr>
<tr>
<td>Version:</td>
<td>1.4.0</td>
</tr>
<tr>
<td>Date:</td>
<td>2013-01-18</td>
</tr>
<tr>
<td>License:</td>
<td>GPL-2</td>
</tr>
<tr>
<td>LazyLoad:</td>
<td>yes</td>
</tr>
</tbody>
</table>

Author(s)

Piotr Dittwald with contribution of Dirk Valkenborg and Jurgen Claesen
Maintainer: Piotr Dittwald <piotr.dittwald@mimuw.edu.pl>

References


Examples

```r
nrPeaks = 1000
aC <- list(C=23832, H=37816, N=6528, O=7031, S=170) # Human dynen heavy chain
res <- useBRAIN(aC = aC, nrPeaks = nrPeaks)
iso <- res$isoDistr
masses <- res$masses
mono <- res$monoisotopicMass
```
Function computing theoretical average masses.

**Description**

Function computing the theoretical average masses for chemical components composed of carbon, hydrogen, oxygen, nitrogen and sulfur (e.g. peptides).

**Usage**

calculateAverageMass(aC)

**Arguments**

aC List with fields C, H, N, O, S of integer non-negative values (if any field is omitted, then its value is set to 0).

**Details**

Mass is calculated in Daltons.

**Value**

Average mass (numeric)

**Author(s)**

Piotr Dittwald <piotr.dittwald@mimuw.edu.pl>

**References**


**See Also**

useBRAIN

**Examples**

aC <- list(C=23832, H=37816, N=6528, O=7031, S=170)  # Human dynein heavy chain
res <- calculateAverageMass(aC = aC)
calculateIsotopicProbabilities

*Function computing probabilities of aggregated isotopic variants using BRAIN algorithm.*

**Description**

Function computing probabilities of aggregated isotopic variants for chemical components built from carbon, hydrogen, oxygen, nitrogen and sulfur (e.g. peptides).

**Usage**

```r
calculateIsotopicProbabilities(aC, stopOption = "nrPeaks",
 nrPeaks, coverage, abundantEstim)
```

**Arguments**

- `aC`: List with fields C, H, N, O, S of integer non-negative values (if any field is omitted, then its value is set to 0).
- `stopOption`: one of the following strings: "nrPeaks" (default), "coverage", "abundantEstim"
- `nrPeaks`: Integer indicating the number of consecutive isotopic variants to be calculated, starting from the monoisotopic one. This value can always be provided, even if `stopOption` is not a default setting. In the latter case it is a hard stopping criterion.
- `coverage`: Scalar indicating the value of the cumulative aggregated distribution. The calculations will be stopped after reaching this value.
- `abundantEstim`: Integer indicating the number of consecutive isotopic variants to be calculated, starting from one after the most abundant one. All consecutive isotopic variants before the most abundant peak are also returned.

**Details**

Remember that the isotopic variants starts from the monoisotopic one. In case of large chemical molecules, first masses may have very low abundance values for the lower mass aggregated values. A sufficient number of peaks should be calculated to reach most abundant isotopic variant.

**Value**

Probabilities of aggregated isotopic variants (numeric vector)

**Note**

If also masses associated with the aggregated isotopic variants are needed, then the function use-BRAIN should be used.

**Author(s)**

Piotr Dittwald <piotr.dittwald@mimuw.edu.pl>
**calculateMonoisotopicMass**

Function computing theoretical monoisotopic masses.

**Description**

Function computing the theoretical monoisotopic masses for chemical components composed of carbon, hydrogen, oxygen, nitrogen and sulfur (e.g. peptides).

**Usage**

```r
calculateMonoisotopicMass(aC)
```

**Arguments**

- `aC` List with fields C, H, N, O, S of integer non-negative values (if any field is omitted, then its value is set to 0).

**Details**

Mass is calculated in Daltons.

**Value**

Monoisotopic mass (numeric)

**Author(s)**

Piotr Dittwald <piotr.dittwald@mimuw.edu.pl>

**References**


---

**References**

See Also

useBRAIN

Examples

```r
aC <- list(C=23832, H=37816, N=6528, O=7031, S=170) # Human dynein heavy chain
res <- calculateMonoisotopicMass(aC = aC)
```

### calculateNrPeaks

Function computing heuristically the required number of consecutive aggregated isotopic variants.

#### Description

Function computing heuristically the required number of consecutive aggregated isotopic variants (starting from the monoisotopic mass).

#### Usage

```r
calculateNrPeaks(aC)
```

#### Arguments

- `aC` List with fields C, H, N, O, S of integer non-negative values (if any field is omitted, then its value is set to 0).

#### Details

This function uses following rule of thumb: the difference between the theoretical monoisotopic mass and the theoretical average mass is computed and multiplied by two. Subsequently, the obtained number is rounded to the nearest integer greater than or equal to the multiplied difference. For small molecules, the minimal number of returned variants is five.

#### Value

Integer number not lower than 5.

#### Author(s)

Jurgen Claesen <jurgen.claesen@uhasselt.be>

#### References


#### See Also

useBRAIN
getAtomsFromSeq

Examples

```r
aC <- list(C=23832, H=37816, N=6528, O=7031, S=170) # Human dynein heavy chain
res <- calculateNrPeaks(aC = aC)
```

**Description**

Function computing an atomic composition from (naturally occuring) amino acid sequence.

**Usage**

```r
getAtomsFromSeq(seq)
```

**Arguments**


**Details**

The atomic composition is just a summaric atomic composition of all amino acids composing the sequence minus (n-1) times the water molecule, where n is a length of given amino acid sequence.

**Value**

Named list with the following fields with number of correponding atoms (integer non-negative values):

- C
- H
- N
- O
- S

**Author(s)**

Piotr Dittwald <piotr.dittwald@mimuw.edu.pl>

**Examples**

```r
seq1 <- "AACD"
ac1 <- getAtomsFromSeq(seq = seq1)
seq2 <- AAString("ACCD")
ac2 <- getAtomsFromSeq(seq = seq2)
```
**useBRAIN**

*Function computing probabilities of aggregated isotopic variants and their center-masses using BRAIN algorithm.*

**Description**

Function computing probabilities of isotopic variants and their aggregated masses for chemical components composed of carbon, hydrogen, oxygen, nitrogen and sulfur (e.g. peptides). Additionally the function returns also the monoisotopic mass and the average mass of given chemical component.

**Usage**

```r
useBRAIN(aC, stopOption = "nrPeaks", nrPeaks, coverage, abundantEstim)
```

**Arguments**

- `aC` List with fields C, H, N, O, S of integer non-negative values (if any field is omitted, then its value is set to 0).
- `stopOption` one of the following strings: "nrPeaks" (default), "coverage", "abundantEstim"
- `nrPeaks` Integer indicating the number of consecutive isotopic variants to be calculated, starting from the monoisotopic one. This value can always be provided, even if <stop.option> is not a default setting. In the latter case it is a hard stopping criterion.
- `coverage` Scalar indicating the value of the cumulative aggregated distribution. The calculations will be stopped after reaching this value.
- `abundantEstim` Integer indicating the number of consecutive isotopic variants to be calculated, starting from one after the most abundant one. All consecutive isotopic variants before the most abundant peak are also returned.

**Details**

Function uses recursive formulae based on algebraic Newton-Girard identity (see [Clae]).

**Value**

Named list with the following fields:

- `isoDistrProbabilities` probabilities of aggregated isotopic variants (numeric vector)
- `masses` Aggregated masses for isotopic variants (numeric vector)
- `monoisotopicMass` Monoisotopic mass (numeric)
- `avgMass` Average mass - weighted average of the isotopic variants contributing to the most abundant aggregated variant (numeric)

**Note**

Remember that the isotopic variants starts from monoisotopic one. For large chemical molecules, first masses may have very low abundances. So sufficient number of peaks should be calculated to reach most abundant isotopic variant.

If only isotopic probabilities are needed, then the function `calculateIsotopicProbabilities` should be used.
useBRAIN

Author(s)
Piotr Dittwald <piotr.dittwald@mimuw.edu.pl>

References

See Also
calculateIsotopicProbabilities

Examples
nrPeaks = 1000
aC <- list(C=23832, H=37816, N=6528, O=7031, S=170) # Human dynein heavy chain
res <- useBRAIN(aC = aC, stopOption="nrPeaks", nrPeaks = nrPeaks)
Index

*Topic package*
  BRAIN-package, 2
BRAIN (BRAIN-package), 2
BRAIN-package, 2

calculateAverageMass, 3
calculateIsotopicProbabilities, 4, 9
calculateMonoisotopicMass, 5
calculateNrPeaks, 6

getAtomsFromSeq, 7

useBRAIN, 3, 5, 6, 8