Atom count expectations with `compoundQuantiles`

Hendrik Treutler

October 29, 2019

1 Introduction

In [1] we propose an approach for the exhaustive detection and mass-specific validation of isotope clusters. In this approach we perform an targeted peak picking for the detection of isotopoloue features and we perform a mass-specific validation of putative isotope clusters. We validate putative isotope clusters in a mass-specific manner on basis of database statistics. Here, we compute quantiles for the ratio between the isotopes of database-specific sets of substances resulting in confidence intervals of isotope ratios. We check whether the ratios between the peaks of a putative isotope cluster are within the calculated confidence intervals and deconvolute the putative isotope cluster otherwise.
There are at least four cases for which the validation of isotope clusters can be beneficial as follows. First, valid isotope clusters can be verified which strengthens the trust in the data. Second, multiple coeluting substances with mass differences of a few dalton can result in isobaric ion species and thus in overlapping isotope clusters. These are potentially misinterpreted as a single isotope cluster affecting downstream analyses. This necessitates the deconvolution of the overlapping isotope cluster into at least two valid isotope clusters. Third, substances can be affected by hydrogen loss. This leads to mass differences similar to isotope peaks and can result in a small trailing peak which is potentially misinterpreted as monoisotopic peak of the putative isotope cluster. This may result in the assumption of a wrong monoisotopic mass and may even lead to the rejection of the entire isotope cluster on the basis of failed intensity-checks. Although this small trailing peak corresponds to the same substance, it needs to be removed from the isotope cluster in order to allow more precise molecular formula predictions. Fourth, the intensity of small peaks is systematically underestimated by some mass spectrometers

which leads to distorted ratios between different isotope peaks as reported previously in Boecker *et al.* 2009. This intensity bias would lead to distorted molecular formula predictions and the removal of these underestimated peaks from the isotope cluster allows more precise molecular formula predictions.

2 Usage

```
## attach
library("compoundQuantiles")
## instantiate
cpObj <- compoundQuantiles(compoundLibrary = "kegg")

## meta information
print(paste("Available libraries = {", paste(compoundLibraries(), collapse = ", "),
print(paste("Compound library = ", cpObj@compoundLibrary, sep = ""))
print(paste("Available mass window sizes = {", paste(massWindowSizes(cpObj@compoundL
print(paste("Mass window size = ", cpObj@massWindowSize, sep = ""))
print(paste("Elements = {", paste(cpObj@elementSet, collapse = ", "), "}", sep = "")
print(paste("Isotopes = {", paste(cpObj@isotopeSet, collapse = ", "), "}", sep = "")
print(paste("Mass interval = [", cpObj@minCompoundMass, ", ", cpObj@maxCompoundMass,
print(paste("Quantile levels = {", paste(cpObj@quantileSet, collapse = ", "), "}", s

## examples
compoundMass <- 503
quantileLow   <- 0.05
quantileHigh  <- 0.95

## example for element count
element <- "C"
countLow  <- getAtomCount(object = cpObj, element = element, mass = compoundMass, qu
countHigh <- getAtomCount(object = cpObj, element = element, mass = compoundMass, qu

print(paste("The ", (quantileHigh - quantileLow) * 100, "\% confidence interval for

## example for isotope proportion
isotope1 <- 0
isotope2 <- 1
propLow  <- getIsotopeProportion(object = cpObj, isotope1 = isotope1, isotope2 = iso
propHigh <- getIsotopeProportion(object = cpObj, isotope1 = isotope1, isotope2 = iso

print(paste("The ", (quantileHigh - quantileLow) * 100, "\% confidence interval for
```

In the above example, we create an S4 object `o` of class *compoundQuantiles*.

The instantiation causes the preparation of the raw data.
Next, we print meta information by addressing the corresponding slots of object `o`. E.g. the set of available quantiles is kept by slot *quantileSet*.
In the given example use case, we define the parameters of interest. I.e. the mass of the compound (*compoundMass*) is 503Da, the element (*element*) is carbon, and the quantiles (*quantileLow* and *quantileHigh*) are 5% and 95% and thus border a 90% confidence interval.
The defined parameters are assigned to the method *getAtomCount*, which returns the expected number of atoms (*countLow* and *countHigh*) of the given element in a compound of the given mass for the given quantiles. Terminatory, we print the result summed up.
The output of the given R snippet is as follows.

```
  [1] "Available libraries = {chebi, kegg, knapsack, LipidMaps, pubchem}"
  [1] "Compound library = kegg"
  [1] "Available mass window sizes = {10, 25, 50, 100, 250}"
  [1] "Mass window size = 50"
  [1] "Elements = {Br, Cl, C, F, H, I, N, O, P, Si, S, Unknown}"
  [1] "Isotopes = {1, 2, 3, 4, 5}"
  [1] "Mass interval = [0, 1050] Da; #mass windows = 21 a 50Da"
  [1] "Quantiles = {5e-06, 0.999995, 1e-05, 0.99999, 5e-05, 0.99995, 1e-
04, 0.9999, 5e-04, 0.9995, 0.001, 0.999, 0.005, 0.995, 0.01, 0.99, 0.025, 0.975, 0.05,
  [1] "The 90\% confidence interval for the number of atoms of element C in a compound
  [1] "The 90\% confidence interval for the proportion of isotopes 0 / 1 in a compound
```

[1] Submitted to Metabolites journal, specieal issue “Bioinformatics and Data Analysis”.