An introduction to Everest package

Version 0.3.0

Xavier Domingo-Almenara (Maintainer)
xdomingo@scripps.edu

May 23, 2018

This vignette presents **Everest**, an R package for annotation of metabolites in untargeted LC–MS-based metabolomics. **Everest** allows annotation of a list of features from XCMS or other computational tools. **Everest** provides a full characterization of features, detecting adducts, common losses and dimmers, to computationally determine monoisotopic masses from underlying metabolites in the samples. **Everest** provides a list of putative identification based on accurate mass search of computed monoisotopic masses. If you use the package **Everest** in your analysis and publications please cite:

Domingo-Almenara X. Everest, an R package for metabolite annotation in untargeted LC-MS-based metabolomics. CRAN. URL https://CRAN.R-project.org/package=everest

Installation: **Everest** can be installed from any CRAN repository, by:

```
# Install
install.packages("everest")
# Load
library(everest)
```

Support: Any enquiries, bug reports or suggestions are welcome and they should be addressed to xdomingo@scripps.edu.

Contents

1 Introduction

Everest is an R package that allows the annotation of LC–MS-based untargeted data. **Everest** will first group those features that stem from the same metabolite, assigning to each one an 'AlignID' number. All these features sharing the same 'AlignID' number are considered a single pseudo-spectra resulting from the same molecule. Next, **Everest** will annotate these features within each pseudo-spectra, using known monoisotopic masses for common adducts, losses and dimmers. Features that show a m/z relation will be annotated with the same 'AnnID' number.

2 Importing feature tables

Everest can work with xcmsSet object directly from XCMS, or import tables. If you have processed your data with XCMS (Offline version), and you have the xcmsSet (e.g., xset3) loaded into your R enviroment, then proceed to the next section. Otherwise, if you have processed your data and you have a CSV or Excel file (e.g., diffreport from XCMS Online) then read the following steps.

First, edit the spreadsheet (Excel or CSV) so that the first column contains the features names, the second the *m/z* values and the third the Retention Times, and name them as: name, mz and RT, respectively. The rest of the columns should contain only the intensity or area (recommended) values of each feature in each sample. Remove any other columns or rows. Here is an example of how the table should look like:

```
name             mz        RT     Sample1     Sample2    ...
M50T350    50.22695  349.5595    375013.1   1997693.4
M51T316    51.28015  316.2429   2278437.1    213624.8
M51T349    51.28734  349.3183     23420.0   3988480.2
M51T3162   51.30070  316.3434   1800327.7   2046424.4
M51T3163   51.31288  316.4545   1208414.6   1546696.9
M53T350    52.94600  349.5218    318827.2   1227462.8
```

☞ **RT and column names:** The first column and the sample column can have any name. However, columns 2 and 3 have to follow this exact spelling: mz and RT. Also, the RT must be in seconds, so multiply the RT (rtmed) column per 60 if the data is in seconds (specially when using XCMS Online, as it is the default option)

Next, the file can be loaded into R using the native `read.csv2()` or the `read.xlsx()` function from openxlsx package[1].

```
# Load table (CSV)
feature.table <- read.csv2("featureTable.csv")
# Load table (tab delim TSV file)
feature.table <- read.delim("featureTable.tsv")
# Or an XLSX file:
feature.table <- read.xlsx("featureTable.xlsx")
# Make sure that it looks OK by taking a quick look:
head(feature.table)
```

If the output of `head(feature.table)` looks good, then we can proceed to the following section. Be careful, sometimes when loading the table, (e.g., if using TSV files) an additional column is introduced. We can remove it easily with R (see following section)

[1] You can install openxslx (https://CRAN.R-project.org/package=openxlsx) by typing the following into the R console: install.packages("openxlsx")

3 Annotation of features in untargeted LC/MS-based data

In this section we show the annotation of a feature list as a result of the analysis of blood samples in positive mode with processing via XCMS. The feature table used to reproduce this example can be download from the Github repository [2]. However, the following code includes a script to automatically download the file. All the listed commands (script) to reproduce the following demo can by found by executing:

```
help(package="everest")
```

and then click on *User guides, package vignettes and other documentation* and on *source* from the 'Everest Manual'.

3.1 Metabolite annotation

The following code will download the DEMO feature table and proceed with the metabolite annotation.

```
library(everest)
# Download the feature table (if not downloaded manually before):
download.file("https://github.com/xdomingoal/everest-data/raw/master/MTBLS20.tsv"
, "MTBLS20.tsv")
# Load the table (we have to remove the first column of the table, as explained be-
fore):
feature.table <- read.delim("featureTable.tsv")[,-1]
# Annotate
ex <- evAnnotate(data.table = feature.table, ion.mode = "pos",
min.correlation = 0.7, max.time.dist = 2, ppm.error = 20)
```

The results can be accessed throught the function `annoTable()`, and we can save the resulting file as an Excel or CSV file:

```
# Access the annotation results:
anTab <- annoTable(ex)
# Write the table as an excel file:
require(openxlsx)
write.xlsx(anTab, file = "results.xlsx")
```

The resulting table will contain four additional columns:

`AlignID:` All the features sharing the same AlignID are considered to stem from the same metabolite.

`AnnID:` Within the same AlignID, these features sharing the same AnnID are considered to stem from a unique monoisotopic mass. A unique AlignID may have more than one AnnID, and each AnnID number leads to a different monoisotopic m/z (see example below).

`Isotope:` Indicates whether or not this feature is an isotopic peak.

`toMSMS:` Indicates whether or not **Everest** recommends these feature for subsequent tandem MS fragmentation.

[2] URL to download the file. https://github.com/xdomingoal/everest-data/raw/master/MTBLS20.tsv (click on File/Save File As on your web explorer if the file does not automatically downloads.)

To annotate a xset3 object from XCMS (offline version), use the following code:

```
# Annotate an xset3 object from XCMS
ex <- evAnnotate(xcmsSet = xset3, ion.mode = "pos",
min.correlation = 0.7, max.time.dist = 2, ppm.error = 20)
```

3.2 Ohter functionalities

Two other functions are included in , `showAn` and `showGroup`. The first displays access to a summary of the annotation results for a specific feature. Let us consider that we want to know the annotation of the feature 'M170T276_8', in that case:

```
showAn(ex, "M170T276_8")
AlignID AnnID Feature    Mass     Rt    Adduct IsoCount Score mIsoMass Isotope toMSMS
525     1     M152T276   152.0344 276.4 (M+H)+[-H2O]  1 99    169.0382
525     1     M170T276_8 170.0448 276.4 (M+H)+        0 99    169.0382         yes
525     1     M192T276   192.0268 276.1 (M+Na)+       0 99    169.0382
525     1     M339T276   339.0852 276.1 (2M+H)+       0 99    169.0382
525     1     M153T276   153.0376 276.3 M+1           0 99    169.0382 yes
525     2     M152T276   152.0344 276.4 (M+Na)+[-H2O] 1 99    147.0562
525     2     M170T276_8 170.0448 276.4 (M+Na)+       0 99    147.0562
525     2     M192T276   192.0268 276.1 [M-H+2Na]+    0 99    147.0562
525     2     M153T276   153.0376 276.3 M+1           0 99    147.0562 yes
```

In this example, the queried feature is shown together with the other features sharing the same `AlignID` number. We observe that the queried feature is a protonated (M+H) specie, and that shares its `AnnID` number with other four features, which correspond to adducts, losses, dimmers and isotopes. Additionally, other four features within the same `AlignID` group share a different `AnnID` number. This indicates that these `AlignID` group could actually be another metabolite (protonated specie). **Everest** ranks the different annotation hypothesis, and in this case, proposes the queried feature for subsquent tandem MS fragmentation (`toMSMS` column). The `Isocount` column indicates how many isotopic peaks have been found for that feature, whereas the `mIsoMass` indicates the monoisotopic mass to which all the features whithin a unique `AnnID` point to. Of note, decimals in this example are truncated for illustration purposes.

The second function, `showGroup` retrieves a summary of the group (AlignID) of that feature, and it contains specific information such as the pseudo-spectra or the list of features that compose that group.