MetCleaning v1.1.0

Xiaotao Shen ([shenxt@sioc.ac.cn](mailto:shenxt@sioc.ac.cn)) and Zhengjiang Zhu

2016-11-25

## Introduction

### *MetCleaning* provides an integrated and automatic pipeline for data cleaning and statistical analysis of large scale mass spectrometry (MS) based-metabolomic data. It includes missing value (MV) filtering and imputation, zero value filtering, data normalization, data integration, data quality assessment, univariate statistical analysis, multivariate statistical analysis such as PCA and PLS-DA, potential marker selection and show. This document describes how to use the integrated functions, *MetClean* and *MetStat* in *MetCleaning* utilizing demo data.

D:\study\R\my git\MetCleaning\vignettes\workflow for instruction.tif

Figure1 Workflow of MetCleaning

## 安装及查看帮助文档

### *MetCleaning*存储在github上 [(link)](https://github.com/jaspershen/MetCleaning). 可以通过github进行安装.

#### code 1: 安装*MetCleaning*

##*pcaMethods*,*impute和pathifier需要通过bioconductor安装.*  
##*pcaMethods*  
source("http://bioconductor.org/biocLite.R")  
 biocLite("pcaMethods")  
##*impute*   
source("http://bioconductor.org/biocLite.R")  
 biocLite("impute")  
 if(!require(devtools)) {  
 install.packages("devtools")  
 }  
 library(devtools)  
 install\_github("jaspershen/MetCleaning")  
 library(MetCleaning)  
 help(package = "MetCleaning")

## 数据清洗

### 在*MetCleaning* 中数据清洗被整合为一个函数*MetClean*. 以包自带的数据作为例子进行练习. 拷贝下面的代码并粘贴在R操作台中.

#### code 2: *MetClean*示例数据

##demo data  
data(data, package = "MetCleaning")  
data(sample.information, package = "MetCleaning")   
##demo work directory  
dir.create("Demo for MetCleaning")  
setwd("Demo for MetCleaning")   
##write files  
write.csv(data, "data.csv", row.names = FALSE)  
write.csv(sample.information,"sample.information.csv",row.names=FALSE)

### 如图一所示, 示例数据这时候就已经放在了你的工作路径中. 包括两个文件, "data.csv"和"sample.information.csv". 如果需要做鉴定, 则需要另外新建立一个文件夹, 命名为"peak identification", 在里面放入二级鉴定的数据, 且文件命名必须含有"ms2"字样.

### 1. "data.csv" 是要处理的数据. 行是feature, 列是样品名以及feature的信息. Feature的信息必须包括"name" (feature名字), "mz" (质核比) and "rt" (保留时间). 其他的信息是可选的, 比如"isotopes"和"adducts". 样品名可以含有"."或者"\_", 但是不能含有"-"或者空格. 而且名字的开头不能是数字, 必须是字母. 比如, "A210.a"和"A210a"有效, 而 "210a" or "210-a"无效.

### 2. "sample.information.csv" 是要处理的数据的样品信息. 第一列是 "sample.name", 是subject和QC样品的名字.请一定要确认"sample.information.csv"和"data.csv"中的样品名字完全一致. 第二列是"injection.order", 是QC和subject samples的进样顺序. 第三列是"class", 是用来表明样品是属于"QC"还是"Subject", 请一定注意大小写. 第四列是"batch", 用来表明样品的批次信息. 第五列是"group", 用来表明subject样品的分组. 例如, "control" and "case". QC样品的"group"信息标为"QC".

D:\study\R\my git\MetCleaning\vignettes\data organisation.tif

Figure2 Data organization of MetCleaning

### 然后就可以使用*MetClean*函数来进行数据的清洗.

#### code 3: Running of *MetClean*

MetClean(data = "sz2016005 urine pos ms1.csv",   
 sample.information = "sample information.csv",  
 polarity = "positive",   
 obs.zero.cutoff = 0.5,  
 var.zero.cutoff = 0.5,  
 method = "svr",  
 threads = 2,   
 hmdb.matching = FALSE,  
 mass.tolerance = 30,  
 mz.tolerance = 30,  
 rt.tolerance = 180,  
 met.plot = TRUE)

#### 参数说明

##### (1) data: 要处理的数据的名字, 例如"data.csv".

##### (2) sample.information: 样品信息的文件名, 比如"sample.information.csv".

##### (3) polarity: 样品的极性, "positive", "negative". 如果是MRM数据, 可以设置为"none".

##### (4) obs.zero.cutoff: 零值筛选的样品阈值, 如果样品中的零值比例超过obs.zero.cutoff值, 那么这个样品就会被舍去, 默认为0.5, 是指一个样品中零值比例超过50%, 则该样品会被除去, 如果设置为100%, 是指一个样品中的所有峰都是0, 才会被舍去.

##### (5) var.zero.cutoff: 和obs.zero.cutoff相同, 是指对峰进行零值筛选.

##### (6) method: 数据标准化方法, 默认为 "svr".

##### (7) threads: 数据标准化时的多线程个数, 默认为2, 服务器上可以适当改大一些.

##### (8) hmdb.matching: 是否使用精确分子质量在HMDB数据库中进行匹配.

##### (9) mass.tolerance: HMDB数据库精确分子质量匹配的tolerance.

##### (10)mz.tolerance:一级二级数据匹配时mz的tolerance.

##### (11)rt.tolerance:一级二级数据匹配时rt的tolerance.

##### (12)mt.plot: 是否将所有feature的plot画出来.

### Running results of *MetClean*

#### 1. Missing or zero values filtering. In the missing or zero value filtering step, if there are samples which beyond the threshold you set, you should decide to filter them or not. We recommend to remove all of them as Figure 3 shows.

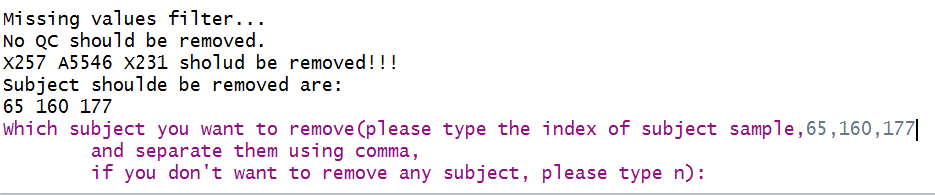


Figure3 Missing or zero value filtering

#### 2. Sample filtering. In the QC or subject sample filtering step (based on PCA), if there are samples which beyond the threshold you set, you should decide to filter them or not. We don't recommend to remove them as Figure 4 shows, because they should be considered combined other information.

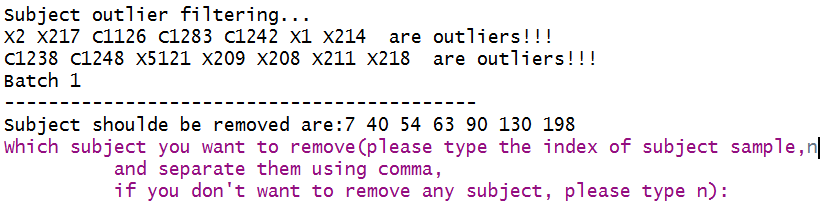


Figure4 Sample filtering

#### 3. Output files. Output files of MetClean are listed as Figure 5 shows.

##### (1) "1MV overview", "2MV filter", "3Zero overview" and "4Zero filter" are missing and zero values filtering information.

##### (2) "5QC outlier filter" and "6Subject outlier filter" are sample filtering based on PCA information.

##### (3) "7Normalization result" is the data normalization information for each batch.

##### (4) "8Batch effect" is the batch effect both in before and after data cleaning.

##### (5) "9metabolite plot" is the scatter plot for each feature.

##### (6) "10Data overview" is the overview of data.

##### (7) "11RSD overview" is the RSD distribution for each batch both before and after data cleaning.

##### (8) "data\_after\_pre.csv", "qc.info.csv" and "subject.info" are the data and sample information after data cleaning.

##### (9) "intermediate" is the intermediate data during processing.

D:\study\R\my git\MetCleaning\vignettes\output files of MetClean.tif

Figure5 Output files of MetClean

## Statistical analysis

### Data statistical analysis is integrated as a function named as *MetStat* in *MetCleaning*. We use the demo data as the example. Please note that now *MetStat* can only process two class data. Copy the code below and paste in you R console.

#### code 4: Demo data of *MetStat*

data("met.data.after.pre", package = "MetCleaning")  
data(new.group, package = "MetCleaning")   
##create a folder for MetStat demo  
dir.create("Demo for MetStat")  
setwd("Demo for MetStat")   
## export the demo data as csv  
write.csv(new.group, "new.group.csv", row.names = FALSE)

### The demo data have been added in your work directory. "new.group.csv" is a sample.information which has been changed the group information you want to use for statistical analysis. For the sample which you don't want to use them for statistical analysis, you can set their group information as NA like Figure 6 shows.

D:\study\R\my git\MetCleaning\vignettes\new.group.tif

Figure6 new group information

#### code 5: Running of *MetStat*

MetStat(MetFlowData = met.data.after.pre, new.group = TRUE)

### Running results of *MetStat*

#### 1. Sample removing. Firstly, you need to confirm the samples which you want to remove form dataset as Figure 7 shows.

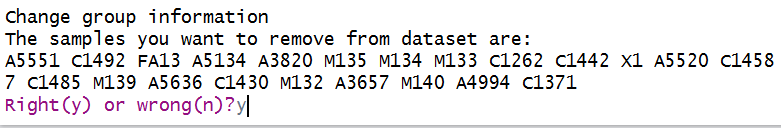


Figure7 sample removing confirmation

#### 2. Number of component selection in PLS-DA analysis. In PLS-DA analysis, you should manually select the best choice of the number of component. When the console show "How many comps do you want to see?", you can type 10 and hit "Enter" key. Then a MSE plot is showing, and the best number of component is the one has the smallest CV values. So type the number (in this example is 4) and hit "Enter" key.

D:\study\R\my git\MetCleaning\vignettes\PLS analysis.tif

Figure8 Number of component selection in PLS-DA analysis

#### 3. Output files. Output files of MetStat are listed as Figure 9 shows.

##### (1) "12PCA analysis" is the PCA score plot.

##### (2) "13PLS analysis" contains the PLS-DA results.

##### (3) "14heatmap" is the heatmap.

##### (4) "15marker selection" contains the information of markers, volcano plot and boxplots of markers.

##### (5) "data\_after\_stat.csv", "qc.info.csv" and "subject.info" are the data and sample information after statistical analysis.

##### (6) "intermediate" is the intermediate data during processing.

D:\study\R\my git\MetCleaning\vignettes\output files of MetStat.tif

Figure9 Output files of MetStat