**RScript 1.** **R Script for eliminating in-source fragment ions by rapidly comparing retention times and searching for special neutral losses in the feature list.**

setwd("C:/Users/YT/Desktop/Sample")

data<-read.csv("Sample.csv")

Water <- rep(NA, length(data$Mass))

Na <- rep(NA, length(data$Mass))

Na\_H2O <- rep(NA, length(data$Mass))

C4H8 <- rep(NA, length(data$Mass))

CH3COOH <- rep(NA, length(data$Mass))

C4H8O <- rep(NA, length(data$Mass))

C3H2O3<- rep(NA, length(data$Mass))

C4H6O2 <- rep(NA, length(data$Mass))

xyl <- rep(NA, length(data$Mass))

glu <- rep(NA, length(data$Mass))

dM\_Na <- rep(NA, length(data$Mass))

data$Water <- Water

data$Na <- Na

data$Na\_H2O <- Na\_H2O

data$C4H8 <- C4H8

data$CH3COOH <- CH3COOH

data$C4H8O <- C4H8O

data$C3H2O3 <- C3H2O3

data$C4H6O2 <- C4H6O2

data$xyl <- xyl

data$glu <- glu

data$dM\_Na <- dM\_Na

for(i in 1:nrow(data)) {

data\_n<-data[i:nrow(data), ]

ms\_error1<-abs(data$Mass[i]-data\_n$Mass)-18.0106

id\_water<-which(abs(ms\_error1)<0.01)

ms\_error2<-abs(data$Mass[i]-data\_n$Mass)-21.982

id\_Na<-which(abs(ms\_error2) < 0.01)

ms\_error3<-abs(data$Mass[i]-data\_n$Mass)-39.9926

id\_Na\_H2O<-which(abs(ms\_error3) < 0.01)

ms\_error4<-abs(data$Mass[i]-data\_n$Mass)-56.0626

id\_C4H8<-which(abs(ms\_error4) < 0.01)

ms\_error5<-abs(data$Mass[i]-data\_n$Mass)-60.0211

id\_CH3COOH<-which(abs(ms\_error5) < 0.01)

ms\_error6<-abs(data$Mass[i]-data\_n$Mass)-72.0575

id\_C4H8O<-which(abs(ms\_error6) < 0.01)

ms\_error7<-abs(data$Mass[i]-data\_n$Mass)-86.0004

id\_C3H2O3<-which(abs(ms\_error7) < 0.01)

ms\_error8<-abs(data$Mass[i]-data\_n$Mass)-86.0368

id\_C4H6O2<-which(abs(ms\_error8) < 0.01)

ms\_error9<-abs(data$Mass[i]-data\_n$Mass)-132.0423

id\_xyl<-which(abs(ms\_error9) < 0.01)

ms\_error10<-abs(data$Mass[i]-data\_n$Mass)-162.0528

id\_glu<-which(abs(ms\_error10) < 0.01)

ms\_error11<-abs(data$Mass[i]-22.9898-2\*(data\_n$Mass-22.9898))

id\_dM\_Na<-which(abs(ms\_error11) < 0.01)

id\_pre<-i

Rt\_error<-data$RT[i]-data\_n$RT

id\_RT<-which(abs(Rt\_error)<0.2)

if (length(c(id\_water))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_water)) {

if(is.na(data$Water[id])) {

data$Water[id+i-1]<-i

data$Water[i]<-i

}

}

}

}

if (length(c(id\_Na))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_Na)) {

if(is.na(data$Na[id])) {

data$Na[id+i-1]<-i

data$Na[i]<-i

}

}

}

}

if (length(c(id\_Na\_H2O))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_Na\_H2O)) {

if(is.na(data$Na\_H2O[id])) {

data$Na\_H2O[id+i-1]<-i

data$Na\_H2O[i]<-i

}

}

}

}

if (length(c(id\_C4H8))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_C4H8)) {

if(is.na(data$C4H8[id])) {

data$C4H8[id+i-1]<-i

data$C4H8[i]<-i

}

}

}

}

if (length(c(id\_CH3COOH))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_CH3COOH)) {

if(is.na(data$CH3COOH[id])) {

data$CH3COOH[id+i-1]<-i

data$CH3COOH[i]<-i

}

}

}

}

if (length(c(id\_C4H8O))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_C4H8O)) {

if(is.na(data$C4H8O[id])) {

data$C4H8O[id+i-1]<-i

data$C4H8O[i]<-i

}

}

}

}

if (length(c(id\_C3H2O3))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_C3H2O3)) {

if(is.na(data$C3H2O3 [id])) {

data$C3H2O3 [id+i-1]<-i

data$C3H2O3 [i]<-i

}

}

}

}

if (length(c(id\_C4H6O2))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_C4H6O2)) {

if(is.na(data$C4H6O2 [id])) {

data$C4H6O2 [id+i-1]<-i

data$C4H6O2 [i]<-i

}

}

}

}

if (length(c(id\_xyl))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_xyl)) {

if(is.na(data$xyl[id])) {

data$xyl[id+i-1]<-i

data$xyl[i]<-i

}

}

}

}

if (length(c(id\_glu))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_glu)) {

if(is.na(data$glu[id])) {

data$glu[id+i-1]<-i

data$glu[i]<-i

}

}

}

}

if (length(c(id\_dM\_Na))>0 & length(id\_RT) >0) {

for(id in id\_RT) {

if(id %in% c(id\_dM\_Na)) {

if(is.na(data$dM\_Na[id])) {

data$dM\_Na[id+i-1]<-i

data$dM\_Na[i]<-i

}

}

}

}

}

write.csv(data, " Sample.csv ")

**RScript 2. R Script for assigning the skeleton subtypes to the potential cycloartane triterpenoid features by monitoring specific fragments along MS1 and MS2 spectra on mzXML files.**

**## Prepare list of precursors and their MS2 spectra**

setwd("E:/DATA/POS-MSMS")

library("MSnbase")

library(xcms)

rawdata <- readMSData("Sample.MZXML", msLevel = 2, verbose = FALSE)

rawdata1 <- readMSData("Sample.MZXML", msLevel = 1, verbose = FALSE)

hd <- fData(rawdata)

prec <- rep(NA, length(hd$spectrum))

int<-rep(NA, length(hd$spectrum))

rt<-rep(NA, length(hd$spectrum))

for( i in 1:length(hd$spectrum)){

if(rawdata[[i]]@precursorMz>400&ceiling(rawdata[[i]]@precursorMz)%%2 ==0){

prec[i] <- rawdata[[i]]@precursorMz

int[i]<-rawdata[[i]]@precursorIntensity

rt[i]<-rawdata[[i]]@rt

}

}

hd$Precursor\_mz <- prec

hd$int<-int

hd$rt<-rt

list\_prec <- vector(mode = "list", length = length(prec))

names(list\_prec) <- c(round(prec, digits = 2))

for( i in 1:length(hd$spectrum)){

list\_prec[[c(i)]] <- c(list\_prec[[c(i)]], list("m/z"=NULL, "int"=NULL))

}

for( i in 1:length(hd$spectrum)){

list\_prec[[c(i, 1)]] <- c(list\_prec[[c(i, 1)]], rawdata[[i]]@mz[which(rawdata[[i]]@mz>250&rawdata[[i]]@intensity>50)])

list\_prec[[c(i, 2)]] <- c(list\_prec[[c(i, 2)]], rawdata[[i]]@intensity[which(rawdata[[i]]@mz>250&rawdata[[i]]@intensity>50)])

}

list\_prec<-list\_prec[!is.na(names(list\_prec))]

hd<-na.omit(hd)

prec<-na.omit(prec)

int<-na.omit(int)

rt<-na.omit(rt)

Struc\_type<-rep(NA, length(list\_prec))

hd$Struc\_type<-Struc\_type

rt<-array()

for(i in 1:length(rawdata1)) {

rt\_i<-rawdata1[[i]]@rt

rt<-c(rt,rt\_i)

}

rt<-rt[-1]

**## Look for 16,23-diketoshengmanol-type CTs in all spectra**

for (i in 1:nrow(hd)) {

id\_rt<-which(abs(hd[i,4]-rt)<30)

for(n in 1:length(id\_rt)) {

id<-id\_rt[n]

mz<-rawdata1[[id]]@mz

if(sum(abs(hd[i,2]-mz)<0.01)>0){

id\_mz<-which(abs(hd[i,2]-mz)<0.01)

id\_mz\_MH<-which(abs(hd[i,2]+72.0575-mz)<0.01)

if(length(id\_mz)>0&length(id\_mz\_MH)>0){

iNT\_MH\_72<-rawdata1[[id]]@intensity[id\_mz]

iNT\_MH<-rawdata1[[id]]@intensity[id\_mz\_MH]

if(iNT\_MH\_72/iNT\_MH >5){

Struc\_type[i]<-"D"

}

}

}

}

}

hd$Struc\_type<-Struc\_type

**## Look for cimigenol-, hydroshengmanol-, shengmanol- and cimiracerol-type CTs in all spectra**

for(i in 1:length(prec)) {

id132.0768 <- which((outer(prec[i],list\_prec[[c(i, 1)]], '-') > 132.0668) &

(outer(prec[i],list\_prec[[c(i, 1)]], '-') < 132.0868))

id72.0575 <- which((outer(prec[i],list\_prec[[c(i, 1)]], '-') > 72.0475) &

(outer(prec[i],list\_prec[[c(i, 1)]], '-') < 72.0675))

id\_72.0575 <- which((outer(list\_prec[[c(i, 1)]],list\_prec[[c(i, 1)]], '-') > 72.0475) &

(outer(list\_prec[[c(i, 1)]],list\_prec[[c(i, 1)]], '-') < 72.0675))

if(length(id72.0575)>0){

if(length(id132.0768)>0&length(id72.0575)>0){

standard<-list\_prec[[c(i, 2)]][id132.0768]-list\_prec[[c(i, 2)]][id72.0575]

if(standard>0){

Struc\_type[i]<-"C"

}else{

Struc\_type[i]<-"AE"

id\_AE\_rt<-which(abs(hd[i,4]-rt)<30)

for(n in 1:length( id\_AE\_rt)){

id\_AE\_n<- id\_AE\_rt[n]

mz<-rawdata1[[id\_AE\_n]]@mz

if(sum(abs(hd[i,2]-mz)<0.01)>0){

id\_mz<-which(abs(hd[i,2]-mz)<0.01)

id\_mz\_21.982<-which(abs(hd[i,2]+21.982-mz)<0.01)

id\_mz\_39.9926<-which(abs(hd[i,2]+39.9926-mz)<0.01)

if(length(id\_mz)>0){

iNT\_mz<-rawdata1[[id\_AE\_n]]@intensity[id\_mz]

iNT\_mz\_21.982<-rawdata1[[id\_AE\_n]]@intensity[id\_mz\_21.982]

iNT\_mz\_39.9926<-rawdata1[[id\_AE\_n]]@intensity[id\_mz\_39.9926]

if(length(id\_mz\_21.982)>0){

if(iNT\_mz\_21.982/iNT\_mz>0.1){

Struc\_type[i]<-"A1"

}

}

if(length(id\_mz\_39.9926)>0){

if(iNT\_mz\_39.9926/iNT\_mz>0.1){

Struc\_type[i]<-"A2"

}

}

}

}

}

}

} else{

Struc\_type[i]<-"AE"

id\_AE\_rt<-which(abs(hd[i,4]-rt)<30)

for(n in 1:length( id\_AE\_rt)){

id\_AE\_n<- id\_AE\_rt[n]

mz<-rawdata1[[id\_AE\_n]]@mz

if(sum(abs(hd[i,2]-mz)<0.01)>0){

id\_mz<-which(abs(hd[i,2]-mz)<0.01)

id\_mz\_21.982<-which(abs(hd[i,2]+21.982-mz)<0.01)

id\_mz\_39.9926<-which(abs(hd[i,2]+39.9926-mz)<0.01)

if(length(id\_mz)>0){

iNT\_mz<-rawdata1[[id\_AE\_n]]@intensity[id\_mz]

iNT\_mz\_21.982<-rawdata1[[id\_AE\_n]]@intensity[id\_mz\_21.982]

iNT\_mz\_39.9926<-rawdata1[[id\_AE\_n]]@intensity[id\_mz\_39.9926]

if(length(id\_mz\_21.982)>0){

if(iNT\_mz\_21.982/iNT\_mz>0.1){

Struc\_type[i]<-"A1"

}

}

if(length(id\_mz\_39.9926)>0){

if(iNT\_mz\_39.9926/iNT\_mz>0.1){

Struc\_type[i]<-"A2"

}

}

}

}

}

}

}else{

if(length(id132.0768)>0&length(id\_72.0575)>0){

Struc\_type[i]<-"B"

}

}

}

hd$Struc\_type<-Struc\_type

**## Look for acteol-type CTs in all spectra**

for(i in 1:length(prec)) {

id <- which((outer(prec[i],list\_prec[[c(i, 1)]], '-') > 86.0268) &

(outer(prec[i],list\_prec[[c(i, 1)]], '-') < 86.0468))

if(length(id)>0) {

Struc\_type[i]<-"F"

}

}

hd$Struc\_type<-Struc\_type

**## Look for anhydro-type CTs in all spectra**

for(i in 1:length(prec)) {

id <- which((outer(prec[i],list\_prec[[c(i, 1)]], '-') > 56.0526) &

(outer(prec[i],list\_prec[[c(i, 1)]], '-') < 56.0726))

if(length(id)>0) {

Struc\_type[i]<-"G"

}

}

hd$Struc\_type<-Struc\_type

Struc\_type<-na.omit(Struc\_type)

length(Struc\_type)

data.table::fwrite(hd,"E:/DATA/msdata.csv",sep=",")