Peak assignment for mass spectrometry data

October 22, 2013

1 Introduction

We report the peak assignment result of near-field laser ablation experiment in combination with laser desorption ionization (LDI) from silicon nanopost arrays (NAPA).

A 2.94μ m laser beam is passed through the end of the tip of the NSOM probe. After the near-field ablation occurs, the ejected material is transferred and de- posited on a silicon NAPA chip, and then analyzed by laser desorption ionization and mass spectrometry(LDI-MS). Our mass spectrometry data was collected by time-of-flight (TOF) mass spectrometer (AXima CFR,Shimadzu-Kratos, Manchester, U.K.).

2 Method

According to the Plant metabolic pathyway (PMN) database, we collected all of the metabolite information of *Arabidopsis*. All these files inclue Compound id, Compound common name, Molecular weight, Chemical formula, Reaction equations, Pathway, etc. Here is the database files: `ftp://ftp.plantcyc.org/Pathways/Data_dumps/PMN7_August2012/`

First, we calculated the monoisotpic mass according to each metabolite's chemical formula. By inserting Excel modules, we cacluated the monoistopic mass for each metabolite.

Second, we combined database files into one giant file, called ”giant.csv” for peak assignment processing.

Third, we matched the peak values with different ion mass values in database and output them.

```
ss = library(calibrate)

## Loading required package:  MASS

ss = library(base)

# giant.csv is the file used to store all of the database files
path <- "~/Dropbox/condmatbiophysics/R/data/"
```

```r
file <- paste(path, "giant.csv", sep = "")
s = read.csv(file, header = T)

# Replace -999999 with NA
s$Monoisotopic.mass[s$Monoisotopic.mass[] == -999999] <- NA

# Replace blank with NA in Molecular_weight
s$Molecular_weight[s$Molecular_weight[] == ""] <- NA

# ion_mass is used to store ions in positive mode
ion_mass <- matrix(nrow = dim(s)[1], ncol = 3)
colnames(ion_mass) <- c("[M+H]+", "[M+K]+", "[M+Na]+")
ion_mass_neg <- matrix(nrow = dim(s)[1], ncol = 3)
colnames(ion_mass_neg) <- c("[M-H]-", "[M+K-2H]-", "[M+Na-2H]-")

# ion_mass_pos is used to store ions
ion_mass[, 1] = s$Monoisotopic.mass[] + 1.00782 - 0.000548
ion_mass[, 2] = s$Monoisotopic.mass[] + 38.96371 - 0.000548
ion_mass[, 3] = s$Monoisotopic.mass[] + 22.9898 - 0.000548

# ion_mass_neg is used to store ions
ion_mass_neg[, 1] = s$Monoisotopic.mass[] - 1.00782 + 0.000548
ion_mass_neg[, 2] = s$Monoisotopic.mass[] + 38.96371 - 2 * (1.00782) + 0.000548
ion_mass_neg[, 3] = s$Monoisotopic.mass[] + 22.9898 - 2 * (1.00782) + 0.000548

# peak_pos.csv and peak_neg.csv are used to store peak values from
# positive mode and negative mode
file_pos <- paste(path, "peak_pos.csv", sep = "")
file_neg <- paste(path, "peak_neg.csv", sep = "")
pos = read.csv(file_pos)
neg = read.csv(file_neg)

output_pos <- c()
output_neg <- c()
peak_assign_pos <- c()
peak_assign_neg <- c()
pos_da <- c()
neg_da <- c()
sub_pos <- c()
sub_neg <- c()
peak_num_pos = 0
x_pos <- c()
peak_num_neg = 0
x_neg <- c()
```

```
for (i in 1:dim(pos)[1]) {
    t = abs(ion_mass[, 1:3] - pos[i, 1] * ones(dim(s)[1], 3))
    # Use ion mass minus the peak value, t is the abosolute value of this
    # difference.
    y <- which(t < 0.055, arr.ind = T)
    # return row number and column number of the matching ones
    if (dim(y)[1] > 0) {
        result <- matrix(nrow = dim(y)[1], ncol = 8)
        colnames(result) <- c("Compound name", "Ions", "Chemical_formula", "Measured_mass",
            "Caculated_mass", "Mass_difference", "Reaction_equation", "Pathway")
        peak_num_pos = peak_num_pos + 1
        x1_pos <- c(peak_num_pos, pos[i, 1])
        x_pos <- rbind(x_pos, x1_pos, y)
        # peak_num_pos is to strore the number of peaks actually have been found
        # matching peaks x_pos is to store the row and column value of matching
        # metabolites from 'giant.csv'
        for (j in 1:dim(y)[1]) {
            result[j, 1] <- toString(s$Compound_common_name[y[j, 1]])
            result[j, 2] <- toString(colnames(ion_mass)[y[j, 2]])
            result[j, 3] <- toString(s$Chemical_formula[y[j, 1]])
            result[j, 4] = pos[i, 1]
            result[j, 5] = ion_mass[y[j, 1], y[j, 2]]
            result[j, 6] = t[y[j, 1], y[j, 2]]
            result[j, 7] = toString(s$Reaction_equation[y[j, 1]])
            result[j, 8] = toString(s$Pathway[y[j, 1]])
        }
        output_pos = rbind(output_pos, result)
        # output_pos includes all of the features of matched metabolites
        rm(result)

    }
}
write.csv(output_pos, "output_pos.csv")
# output_pos.csv is all the possible matching metabolites
pos_da <- data.frame(output_pos)
peak_assign_pos <- unique(output_pos[!duplicated(pos_da$Compound.name), ])
write.csv(peak_assign_pos, "peak_assign_pos.csv")
# peak_assign_pos.csv has all of the matching metabolites without
# redundancy
sub_pos <- unique(pos_da[c("Compound.name", "Ions")])
write.csv(pos_da[row.names(sub_pos), ], "peak_assign_pos1.csv")
# peak_assign_pos1.csv stores all of the matching metabolites with
# different possible ions
write.csv(x_pos, "assigned_peak_pos.csv")
# assigned_peak_pos.csv stores the assigned peaks and the matching row and
# column numbers in giant.csv

# Following codes are similar as above, just for negative mode.
for (i in 1:dim(neg)[1]) {
    t1 = abs(ion_mass_neg[, 1:3] - neg[i, 1] * ones(dim(s)[1], 3))
    y1 <- which(t1 < 0.055, arr.ind = T)
    if (dim(y1)[1] > 0) {
        result <- matrix(nrow = dim(y1)[1], ncol = 8)
        colnames(result) <- c("Compound name", "Ions", "Chemical_formula", "Measured_mass",
            "Caculated_mass", "Mass_difference", "Reaction_equation", "Pathway")
        peak_num_neg = peak_num_neg + 1
        x1_neg <- c(peak_num_neg, neg[i, 1])
        x_neg <- rbind(x_neg, x1_neg, y1)
        for (j in 1:dim(y1)[1]) {
            result[j, 1] <- toString(s$Compound_common_name[y1[j, 1]])
            result[j, 2] <- toString(colnames(ion_mass_neg)[y1[j, 2]])
            result[j, 3] <- toString(s$Chemical_formula[y1[j, 1]])
            result[j, 4] = neg[i, 1]
            result[j, 5] = ion_mass_neg[y1[j, 1], y1[j, 2]]
            result[j, 6] = t1[y1[j, 1], y1[j, 2]]
            result[j, 7] = toString(s$Reaction_equation[y1[j, 1]])
            result[j, 8] = toString(s$Pathway[y1[j, 1]])
        }
        output_neg = rbind(output_neg, result)
        rm(result)
    }
}
write.csv(output_neg, "output_neg.csv")
neg_da <- data.frame(output_neg)
peak_assign_neg <- unique(output_neg[!duplicated(neg_da$Compound.name), ])
write.csv(peak_assign_neg, "peak_assign_neg.csv")
sub_neg <- unique(neg_da[c("Compound.name", "Ions")])
write.csv(neg_da[row.names(sub_neg), ], "peak_assign_neg1.csv")
write.csv(x_neg, "assigned_peak_neg.csv")
```

3 Results

3.1 Peak assignment

The final results include several files and they are all for different purposes. Files which include ‘pos’ in their file name is for positive mode and ‘neg’ for negative mode.

Using this analysis code, 43 out of 75 predominant peaks have been assigned in the positive mode while 59 out of 91 peaks got assigned in the negative mode. Overall, 204 metabolites have been assigned in the positive mode and 329 metabolites have been assigned in the negative mode.

File 'output_pos.csv' stores all of the matching metabolites in the positive mode. Because the raw data has some reduntancy, this file includes redundant metabolite information as well.

'peak_assign_pos.csv' stores all the matching metabolites without redundancy. We removed the redundant metabolite information in this file.

'peak_assign_pos1.csv' stores all the matching metabolites with different possible ions. For example, 2-tridecanone has matching ion masses with $[M + H]^+$ and $[M + K]^+$. The first column indicates the row number of metabolites in 'output_pos.csv'.

'assigned_peak_pos.csv' includes the peaks assigned in the postive mode. Every row has 'x1_pos' indicate the number of the assigned peak, and the peak value. The rows underneath that peak indicate the row and column number in 'giant.csv' file.

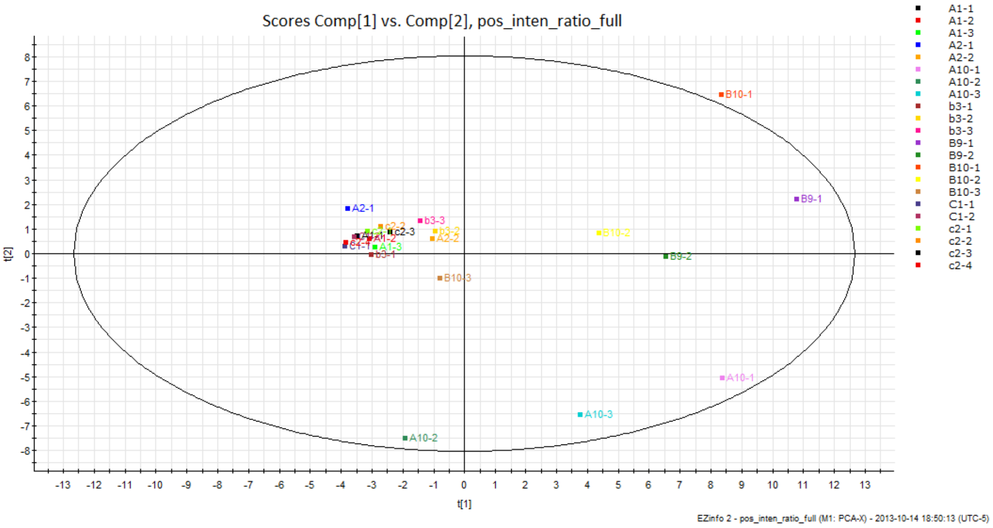
3.2 PCA analysis

In the previous primary metabolite study, we analyzed all the average predominant peaks. After removing the redundancy, we finally have 50 peaks as variables in the positive mode and 73 in the negative mode.

For the positive mode, you can see from Fig.1, it has similar results from what we got using only 13 primary metabolites (Fig.2). They all have a group of data in the center of the plot, and B10-1, B10-2, B9-1, B9-2 and A10-1 are far way from the group. In PCA analysis result from full metabolite variables, A10-2 and A10-3 are far from the center group while the results from primary metabolites shows that they are actually in the center group.

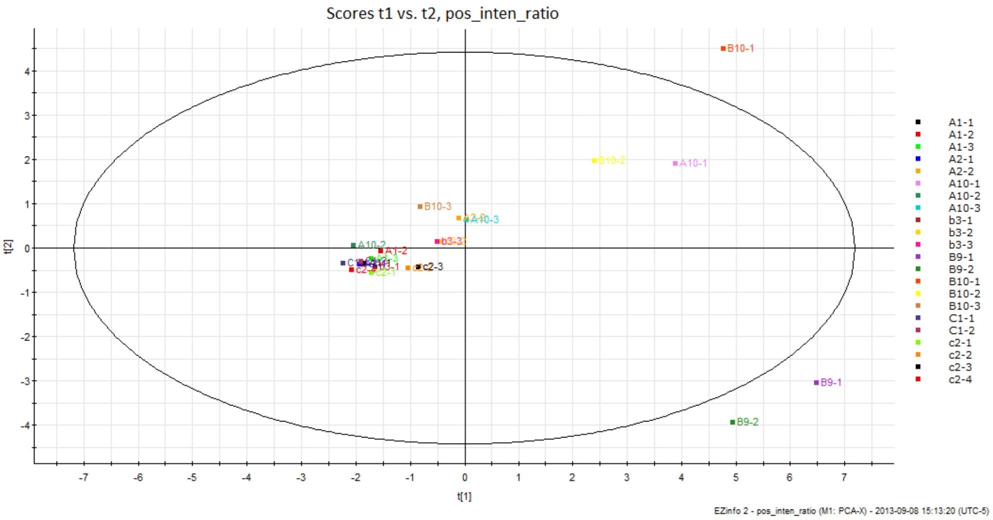
We chose the data in the center as one group and B9-2 as another group. The S-plot (Fig.3) and the VIP parameters(Fig.4) are shown. As you see, metabolites with peak values of 106.79, 74.098, 303.26, 104.114, 254.278 have the best impact of separating these groups. Actually, metabolites n-Butylamine (m/z=74.098), 4-aminobutyrate(GABA)(m/z=104.114) and Serine (m/z=106.79) were also pointed out as the best metabolites to separate these two groups while we were using only 13 primary metabolites as the variables. The other two metabolites are palmitoleate (m/z=254.278) and dehydroabietadiene-diol (etc.)(m/z=303.26).

The negative mode analysis plot is shown in Fig.5 and it is similar as the one we did before using only 14 metabolites. The data points scatter and do not indicate any groups.



EZinfo 2 - pos_inten_ratio_full (M1: PCA-X) - 2013-10-14 18:50:13 (UTC-5)

Figure 1: PCA results for intensity ratios in the positive mode. The results come from the analysis using 22 obseravations and 50 variables.



EZinfo 2 - pos_inten_ratio (M1: PCA-X) - 2013-09-08 15:13:20 (UTC-5)

Figure 2: PCA results for intensity ratios in the positive mode. The results come from the analysis using 22 obseravations and 13 variables.

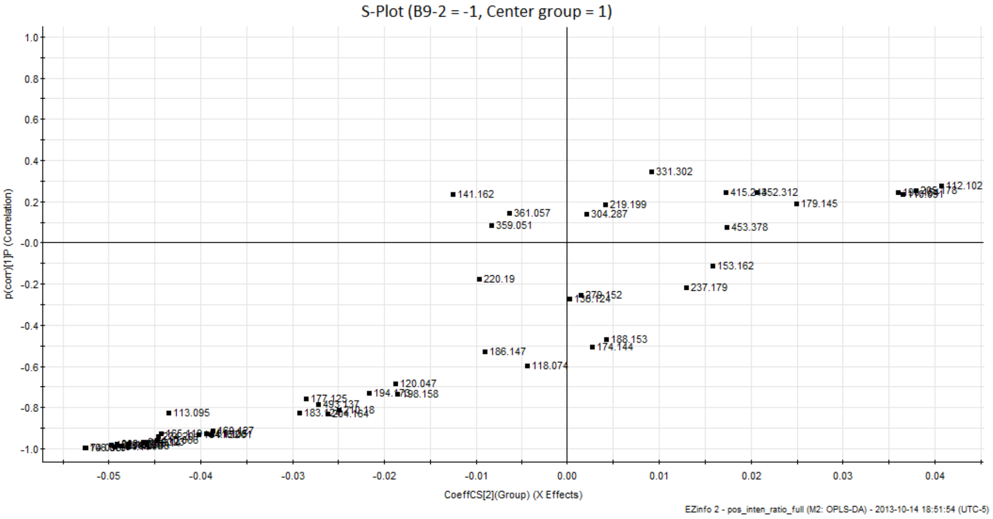


Figure 3: S-plot of using B9-2 as one group and the data in the center as another group.

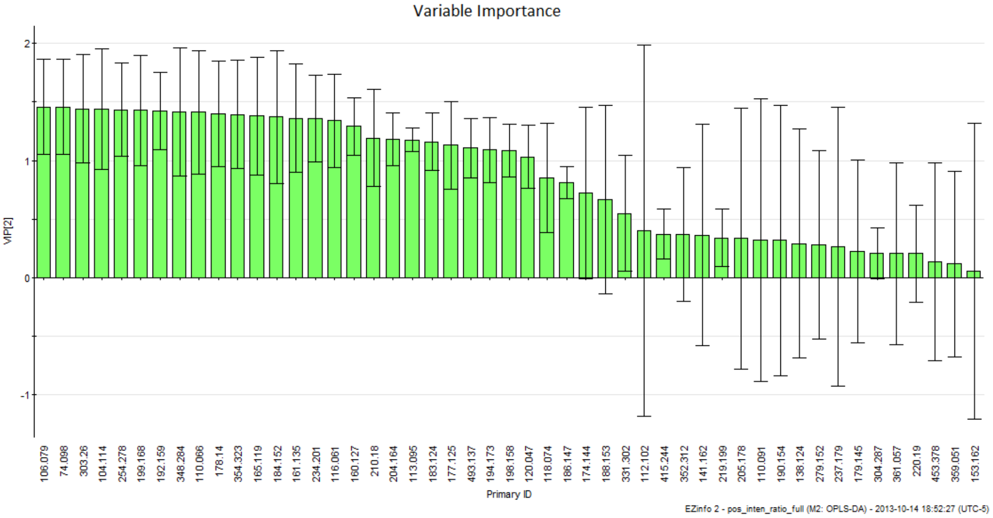


Figure 4: The VIP parameters by using B9-2 as one group and the data in the center as another group.

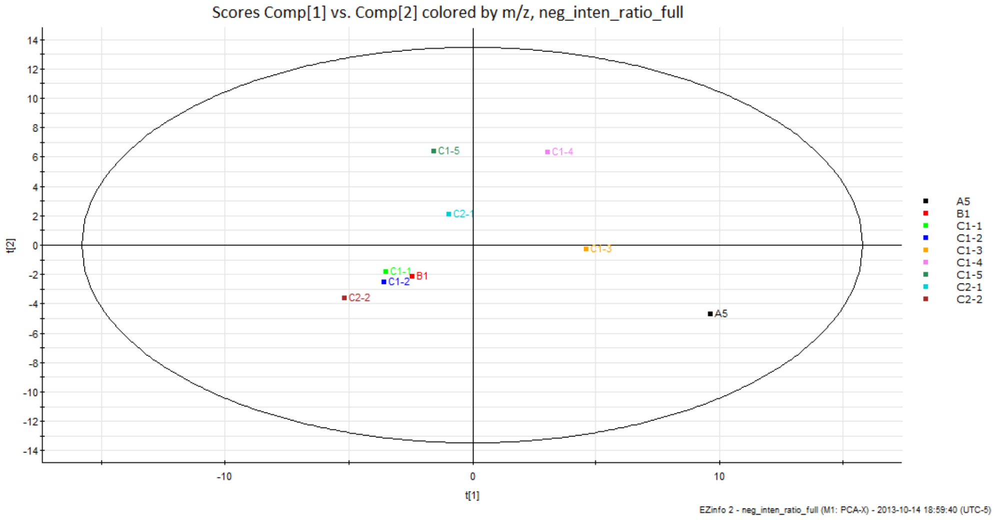


Figure 5: PCA results for intensity ratios in the negative mode. The results come from the analysis using 9 obseravations and 73 variables.