Metabolomic Data Analysis with MetaboAnalyst 2.0

User ID: guest4903193057586512281

February 8, 2015

1 Data Processing and Normalization

1.1 Reading and Processing the Raw Data

MetaboAnalyst accepts a variety of data types generated in metabolomic studies, including compound concentration data, binned NMR/MS spectra data, NMR/MS peak list data, as well as MS spectra (NetCDF, mzXML, mzDATA). Users need to specify the data types when uploading their data in order for MetaboAnalyst to select the correct algorithm to process them. Table 1 summarizes the result of the data processing steps.

1.1.1 Reading MS Peak List and Intensities Data

MS peak list and intensities data should be uploaded as one zip file. It contains subfoulders with one for each group. Each folder contains peak list files, one per spectrum. The MS peak list format is either a two-column (mass and intensities) or three-column (mass, retention time, and intensities) comma separated values. The first line is assumed to be column labels. The files should be saved in .csv format. For paired analysis, users need to upload separately a text file specifying the paired information. Each pair is indicated by their sample names seperated by a colon ":" with one pair per line.

The uploaded files are peak lists and intensities data. A total of 6 samples were found. These samples contain a total of 23436 peaks. with an average of 3906 peaks per sample

1.1.2 Peak Matching and Alignment

Peaks need to be matched across samples in order to be compared. For two-column data, the program matches peaks by their m/z values. For three-column data, the program will further group peaks based on their retention time. During the process, mz and rt of each peak will be changed to their group median values. If a sample has more than one peak in a group, they will be replaced by their sum. Some peaks are excluded if they appear in less than half of both classes. The aligned peaks are reorganized into a single data matrix for further analysis. The name of the parent folder is used as class label for each sample.

A total of 3459 peak groups were formed. Peaks of the same group were summed if they are from one sample. Peaks appear in less than half of samples in each group were ignored.

1.1.3 Data Integrity Check

Before data analysis, a data integrity check is performed to make sure that all the necessary information has been collected. The class labels must be present and contain only two classes. If samples are paired, the class label must be from -n/2 to -1 for one group, and 1 to n/2 for the other group (n is the sample number and must be an even number). Class labels with same absolute value are assumed to be pairs.

Compound concentration or peak intensity values should all be non-negative numbers. By default, all missing values, zeros and negative values will be replaced by the half of the minimum positive value found within the data (see next section)

1.1.4 Missing value imputations

Too many zeroes or missing values will cause difficulties for downstream analysis. MetaboAnalyst offers several different methods for this purpose. The default method replaces all the missing and zero values with a small values (the half of the minimum positive values in the original data) assuming to be the detection limit. The assumption of this approach is that most missing values are caused by low abundance metabolites (i.e.below the detection limit). In addition, since zero values may cause problem for data normalization (i.e. log), they are also replaced with this small value. User can also specify other methods, such as replace by mean/median, or use K-Nearest Neighbours, Probabilistic PCA (PPCA), Bayesian PCA (BPCA) method, Singular Value Decomposition (SVD) method to impute the missing values [1]. Please choose the one that is the most appropriate for your data.

Zero or missing variables were replaced with a small value: 8.886447755

1.1.5 Data Filtering

The purpose of the data filtering is to identify and remove variables that are unlikely to be of use when modeling the data. No phenotype information are used in the filtering process, so the result can be used with any downstream analysis. This step can usually improves the results. Data filter is strongly recommended for datasets with large number of variables (> 250) datasets contain much noise (i.e.chemometrics data). Filtering can usually improve your results[2].

For data with number of variables < 250, this step will reduce 5% of variables; For variable number between 250 and 500, 10% of variables will be removed; For variable number bwteen 500 and 1000, 25% of variables will be removed; And 40% of variabled will be removed for data with over 1000 varaibles.

Reduce 40% features (1384) based on mean

Table 1: Summary of data processing results

	Peaks (raw)	Missing/Zero	Peaks (processed)
Diet_IR_Neg_1	3906	0	2075
Diet_IR_Neg_2	3906	2	2075
Diet_IR_Neg_3	3906	1	2075
Diet_NR_Neg_1	3906	2	2075
Diet_NR_Neg_2	3906	0	2075
Diet_NR_Neg_3	3906	0	2075

[1] Stacklies W, Redestig H, Scholz M, Walther D, Selbig J. *pcaMethods: a bioconductor package, providing PCA methods for incomplete data.*, Bioinformatics 2007 23(9):1164-1167

[2] Hackstadt AJ, Hess AM.*Filtering for increased power for microarray data analysis*, BMC Bioinformatics. 2009; 10: 11.

1.2 Data Normalization

The data is stored as a table with one sample per row and one variable (bin/peak/metabolite) per column. The normalization procedures implemented below are grouped into four categories. Sample specific normalization allows users to manually adjust concentrations based on biological inputs (i.e. volume, mass); row-wise normalization allows general-purpose adjustment for differences among samples; data transformation and scaling are two different approaches to make features more comparable. You can use one or combine both to achieve better results.

The normalization consists of the following options:

1. Sample specific normalization (i.e. normalize by dry weight, volume)
2. Row-wise procedures:
    - Normalization by the sum
    - Normalization by the sample median
    - Normalization by a reference sample (probabilistic quotient normalization)[3]
    - Normalization by a reference feature (i.e. creatinine, internal control)
3. Data transformation :
    - Generalized log transformation (glog 2)
    - Cube root transformation
4. Data scaling:
    - Unit scaling (mean-centered and divided by standard deviation of each variable)
    - Pareto scaling (mean-centered and divided by the square root of standard deviation of each variable)
    - Range scaling (mean-centered and divided by the value range of each variable)

Figure 1 shows the effects before and after normalization.

[3]Dieterle F, Ross A, Schlotterbeck G, Senn H. *Probabilistic quotient normalization as robust method to account for dilution of complex biological mixtures. Application in 1H NMR metabonomics*, 2006, Anal Chem 78 (13);4281 - 4290

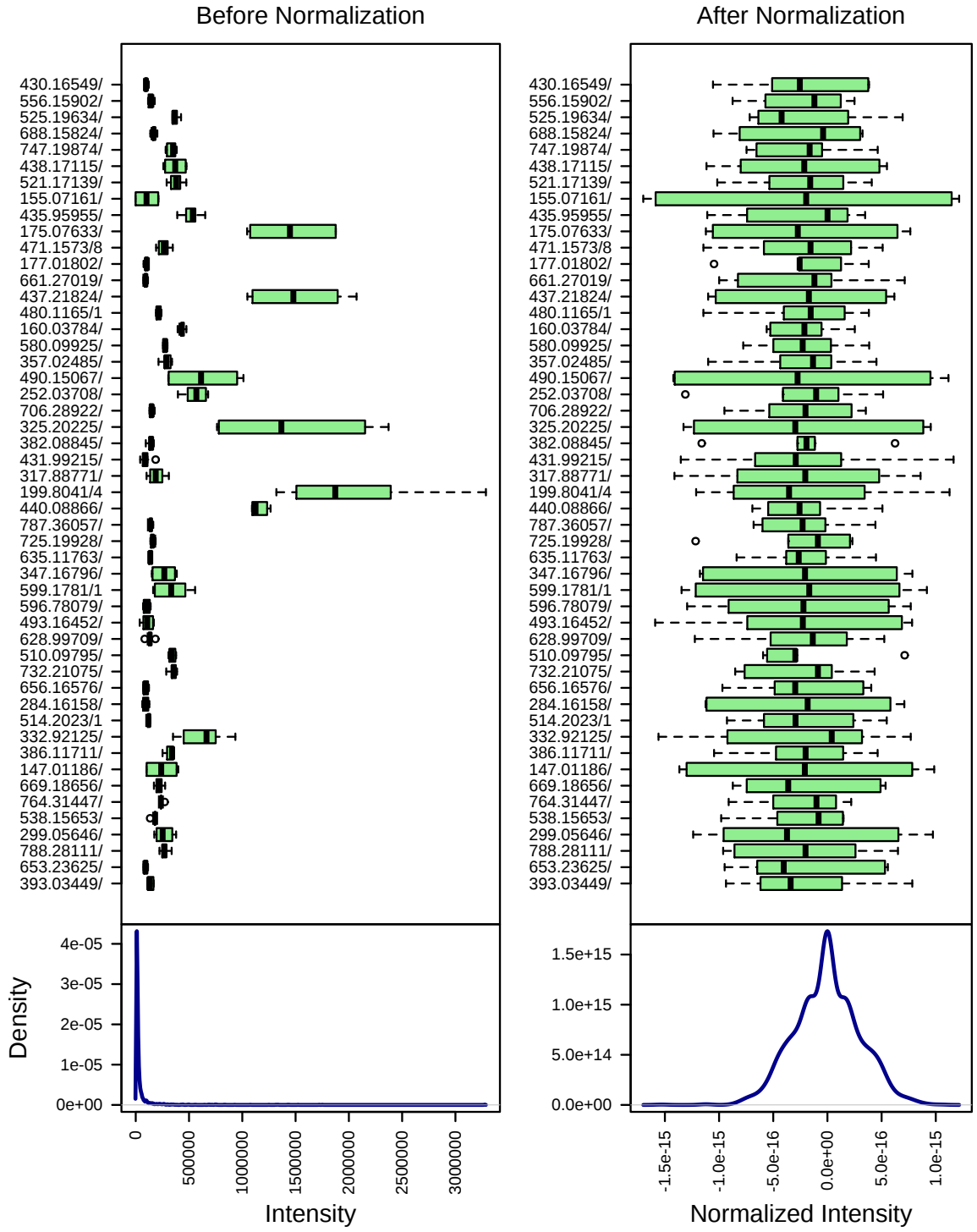


Figure 1: Box plots and kernel density plots before and after normalization. The boxplots show at most 50 features due to space limit. The density plots are based on all samples. Selected methods : Row-wise normalization: Normalization to constant sum; Data transformation: Log Normalization; Data scaling: Pareto Scaling.

2 Statistical and Machine Learning Data Analysis

MetaboAnalyst offers a variety of methods commonly used in metabolomic data analyses. They include:

1. Univariate analysis methods:
    - Fold Change Analysis
    - T-tests
    - Volcano Plot
    - One-way ANOVA and post-hoc analysis
    - Correlation analysis
2. Multivariate analysis methods:
    - Principal Component Analysis (PCA)
    - Partial Least Squares - Discriminant Analysis (PLS-DA)
3. Robust Feature Selection Methods in microarray studies
    - Significance Analysis of Microarray (SAM)
    - Empirical Bayesian Analysis of Microarray (EBAM)
4. Clustering Analysis
    - Hierarchical Clustering
        - Dendrogram
        - Heatmap
    - Partitional Clustering
        - K-means Clustering
        - Self-Organizing Map (SOM)
5. Supervised Classification and Feature Selection methods
    - Random Forest
    - Support Vector Machine (SVM)

`Please note:  some advanced methods are available only for two-group sample analyais.`

2.1 Univariate Analysis

Univariate analysis methods are the most common methods used for exploratory data analysis. For two-group data, MetaboAnalyst provides Fold Change (FC) analysis, t-tests, and volcano plot which is a combination of the first two methods. All three these methods support both unpaired and paired analyses. For multi-group analysis, MetaboAnalyst provides two types of analysis - one-way analysis of variance (ANOVA) with associated post-hoc analyses, and correlation analysis to identify signficant compounds that follow a given pattern. The univariate analyses provide a preliminary overview about features that are potentially significant in discriminating the conditions under study.

For paired fold change analysis, the algorithm first counts the total number of pairs with fold changes that are consistently above/below the specified FC threshold for each variable. A variable will be reported as significant if this number is above a given count threshold (default $> 75\%$ of pairs/variable)

Figure 2 shows the important features identified by fold change analysis. Table 2 shows the details of these features; Figure 3 shows the important features identified by t-tests. Table 3 shows the details of these features; Figure 4 shows the important features identified by volcano plot. Table 4 shows the details of these features.

Please note, the purpose of fold change is to compare absolute value changes between two group means. Therefore, the data before column normlaization will be used instead. Also note, the result is plotted in log2 scale, so that same fold change (up/down-regulated) will have the same distance to the zero baseline.

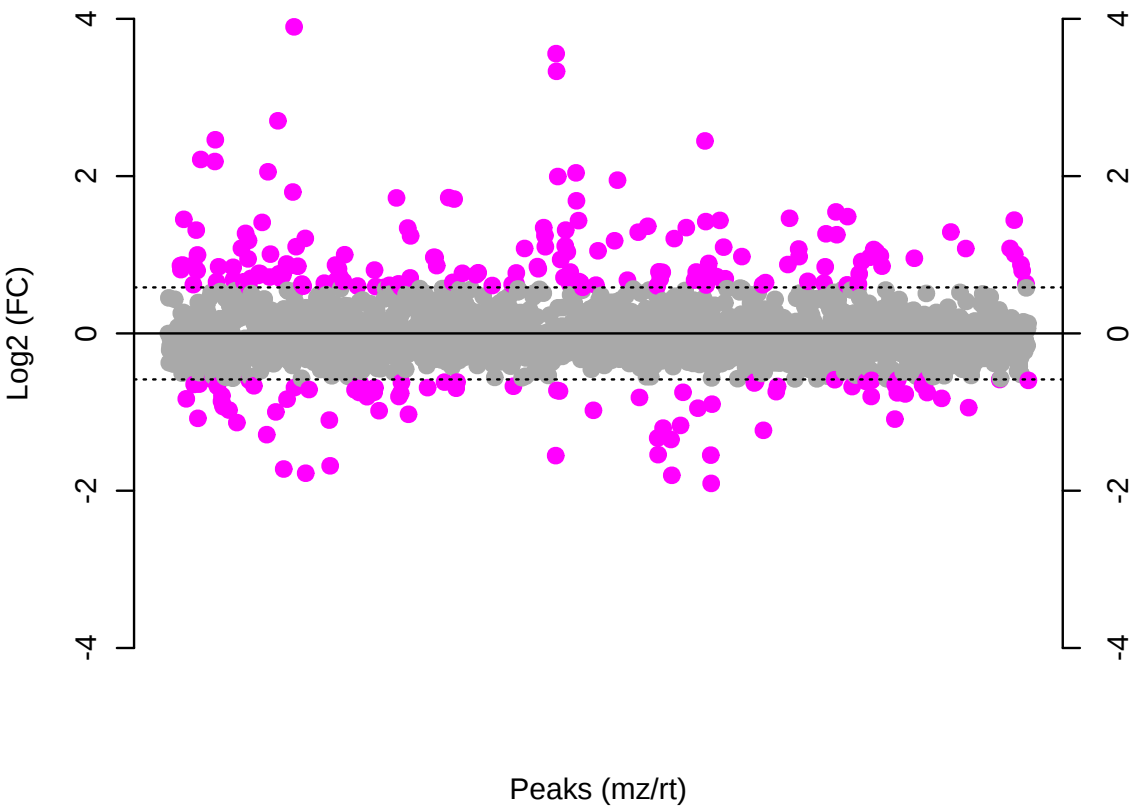


Figure 2: Important features selected by fold-change analysis with threshold 1.5. The red circles represent features above the threshold. Note the values are on log scale, so that both up-regulated and downregulated features can be plotted in a symmetrical way

Table 2: Top 50 features identified by fold change analysis

	Peaks (mz/rt)	Fold Change	log2(FC)
1	242.11034/17.59	14.907	3.898
2	424.02817/8.12	11.773	3.5574
3	424.0396/8.14	10.074	3.3326
4	227.09225/15.85	6.5138	2.7035
5	170.09035/20.93	5.5106	2.4622
6	522.11941/13.88	5.4547	2.4475
7	155.07161/18.96	4.6327	2.2118
8	169.08711/20.44	4.5502	2.1859
9	217.10753/14.43	4.1557	2.0551
10	435.0655/16.99	4.1168	2.0415
11	425.04502/8.13	3.9857	1.9948
12	461.16711/12.08	3.8608	1.9489
13	525.29646/13.68	0.26662	-1.9072
14	499.71989/17.94	0.28634	-1.8042
15	241.10768/15.87	3.4771	1.7979
16	251.05942/16.66	0.29162	-1.7778
17	355.18757/21.87	3.3091	1.7264
18	231.15918/15.64	0.30268	-1.7242
19	315.14299/16.36	3.3012	1.723
20	359.20669/21.18	3.2682	1.7085
21	435.08298/16.95	3.2181	1.6862
22	267.12826/16.96	0.3113	-1.6836
23	423.26338/18.29	0.34056	-1.554
24	525.28412/13.69	0.34199	-1.548
25	631.27995/17.98	2.9181	1.545
26	490.15067/14.25	0.34351	-1.5416
27	640.77433/16.36	2.7994	1.4851
28	592.1392/14.93	2.7586	1.4639
29	129.05396/14.06	2.7332	1.4506
30	782.3709/17.72	2.7141	1.4405
31	533.17637/17.7	2.7074	1.4369
32	436.06435/16.98	2.7021	1.4341
33	522.17375/8.93	2.679	1.4217
34	213.11261/17.42	2.6604	1.4116
35	483.21826/17.62	2.5714	1.3625
36	499.21584/17.89	0.39257	-1.349
37	508.11125/12.33	2.5407	1.3452
38	417.10292/10.97	2.5404	1.345
39	323.18613/20.23	2.5319	1.3402
40	490.13636/14.17	0.39792	-1.3295
41	429.20741/12.98	2.4858	1.3137
42	147.01186/7.27	2.4853	1.3134
43	727.19418/9.59	2.4444	1.2895
44	216.11693/14.98	0.40969	-1.2874
45	475.2589/14.3	2.4405	1.2872
46	197.11032/18.33	2.415	1.272
47	622.22408/7.54	2.406	1.2666
48	632.27935/17.86	2.3863	1.2548
49	417.2123/18.49	2.3703	1.2451
50	325.20225/19.58	2.3617	1.2399

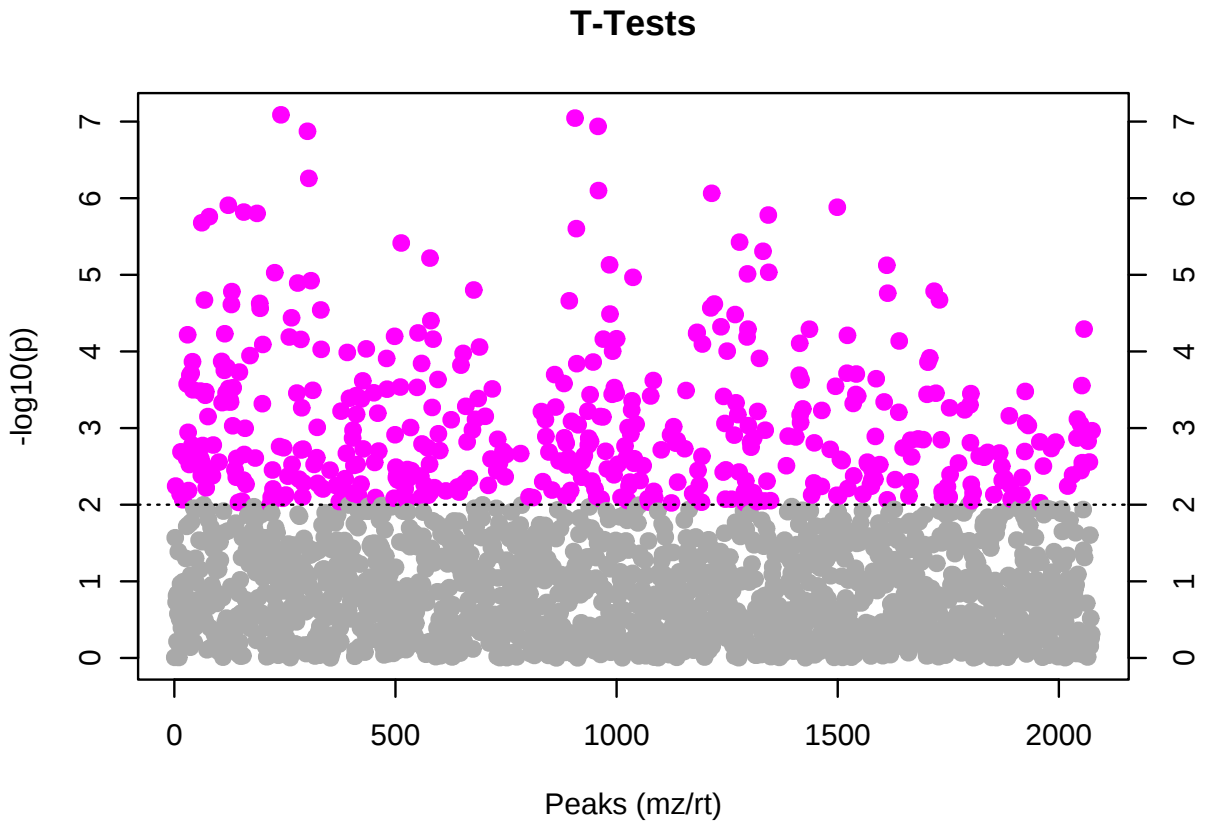


Figure 3: Important features selected by t-tests with threshold 0.01. The red circles represent features above the threshold. Note the p values are transformed by -log10 so that the more significant features (with smaller p values) will be plotted higher on the graph.

Table 3: Top 50 features identified by t-tests

	Peaks (mz/rt)	p.value	-log10(p)	FDR
1	217.10753/14.43	8.1686e-08	7.0879	6.9457e-05
2	417.10292/10.97	9.0108e-08	7.0452	6.9457e-05
3	429.18929/12.95	1.1557e-07	6.9372	6.9457e-05
4	241.10768/15.87	1.3389e-07	6.8732	6.9457e-05
5	242.11034/17.59	5.5121e-07	6.2587	0.00022875
6	429.20741/12.98	7.9521e-07	6.0995	0.00025596
7	499.71989/17.94	8.6347e-07	6.0638	0.00025596
8	173.08163/13.94	1.2364e-06	5.9078	0.0002779
9	592.1392/14.93	1.3117e-06	5.8822	0.0002779
10	187.09745/16.64	1.5173e-06	5.8189	0.0002779
11	197.11032/18.33	1.5792e-06	5.8016	0.0002779
12	537.11333/15.15	1.6592e-06	5.7801	0.0002779
13	155.07161/18.96	1.7411e-06	5.7592	0.0002779
14	144.04568/16.2	2.0924e-06	5.6793	0.00031013
15	417.2123/18.49	2.4965e-06	5.6027	0.00034535
16	517.19481/14.63	3.7438e-06	5.4267	0.00046904
17	304.06878/10.79	3.8427e-06	5.4154	0.00046904
18	533.17637/17.7	4.9285e-06	5.3073	0.00056815
19	323.18613/20.23	6.0488e-06	5.2183	0.0006606
20	435.0655/16.99	7.4065e-06	5.1304	0.00074128
21	631.27995/17.98	7.5021e-06	5.1248	0.00074128
22	537.12574/13.76	9.27e-06	5.0329	0.00084055
23	213.11261/17.42	9.3837e-06	5.0276	0.00084055
24	522.13493/16.42	9.722e-06	5.0122	0.00084055
25	447.25891/18.03	1.0785e-05	4.9672	0.00089519
26	243.1233/16.21	1.1935e-05	4.9232	0.00095253
27	231.15918/15.64	1.2807e-05	4.8926	0.00098423
28	355.18757/21.87	1.5771e-05	4.8021	0.0011449
29	665.28412/14.42	1.635e-05	4.7865	0.0011449
30	175.0975/14.87	1.6553e-05	4.7811	0.0011449
31	632.27935/17.86	1.737e-05	4.7602	0.0011627
32	669.29926/15.76	2.1234e-05	4.673	0.0013327
33	147.01186/7.27	2.1323e-05	4.6712	0.0013327
34	413.18019/16.22	2.1838e-05	4.6608	0.0013327
35	199.09784/15.58	2.3574e-05	4.6276	0.0013693
36	501.1806/16.44	2.4042e-05	4.619	0.0013693
37	175.07633/14.87	2.4417e-05	4.6123	0.0013693
38	499.21584/17.89	2.7042e-05	4.568	0.0014549
39	199.1334/19	2.7345e-05	4.5631	0.0014549
40	251.00445/14.65	2.8737e-05	4.5416	0.0014907
41	435.08298/16.95	3.2503e-05	4.4881	0.0016305
42	514.1562/11.71	3.3003e-05	4.4814	0.0016305
43	227.09225/15.85	3.6357e-05	4.4394	0.0017544
44	324.19066/9.25	3.9666e-05	4.4016	0.0018706
45	504.13812/10.12	4.7667e-05	4.3218	0.002198
46	789.22709/14.97	5.1066e-05	4.2919	0.0022162
47	522.17375/8.93	5.1145e-05	4.2912	0.0022162
48	574.26607/17.71	5.1267e-05	4.2902	0.0022162
49	490.15067/14.25	5.6267e-05	4.2497	0.0023446
50	315.14299/16.36	5.7354e-05	4.2414	0.0023446

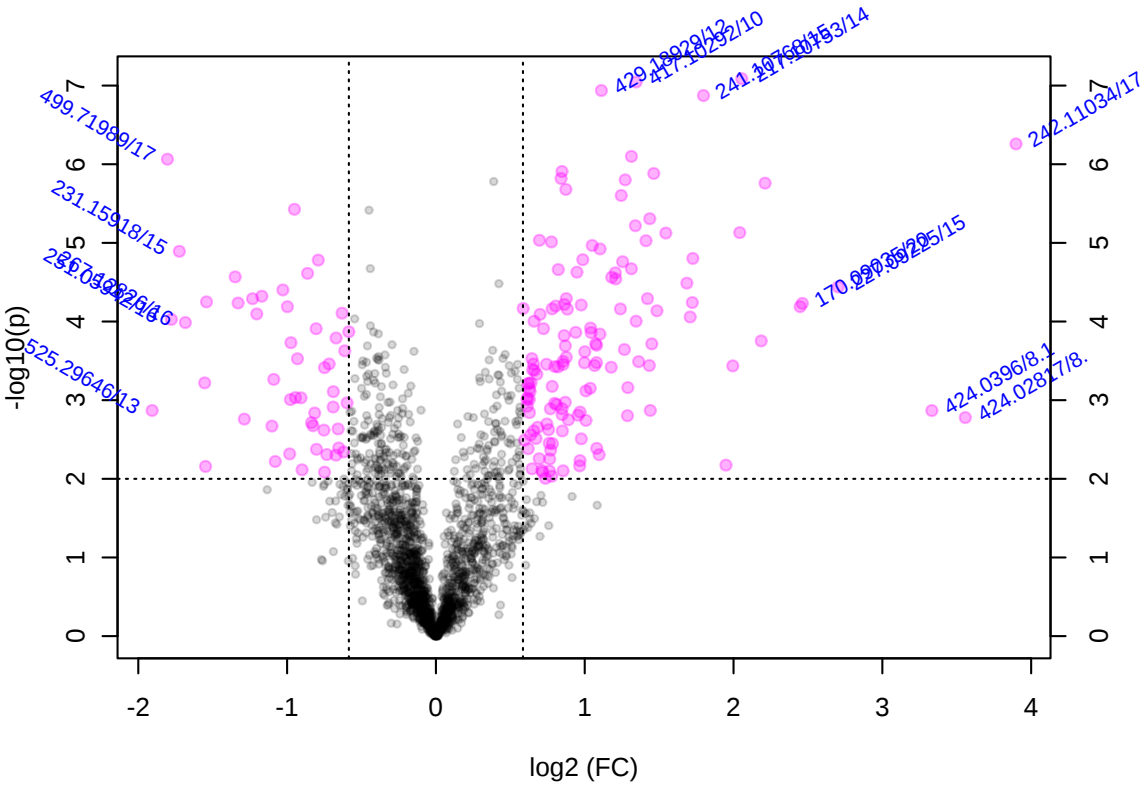


Figure 4: Important features selected by volcano plot with fold change threshold (x) 1.5 and t-tests threshold (y) 0.01. The red circles represent features above the threshold. Note both fold changes and p values are log transformed. The further its position away from the (0,0), the more significant the feature is.

Table 4: Top 50 features identified by volcano plot

	Peaks (mz/rt)	FC	log2(FC)	p.value	-log10(p)
1	217.10753/14.43	4.1557	2.0551	8.1686e-08	7.0879
2	417.10292/10.97	2.5404	1.345	9.0108e-08	7.0452
3	429.18929/12.95	2.1614	1.112	1.1557e-07	6.9372
4	241.10768/15.87	3.4771	1.7979	1.3389e-07	6.8732
5	242.11034/17.59	14.907	3.898	5.5121e-07	6.2587
6	429.20741/12.98	2.4858	1.3137	7.9521e-07	6.0995
7	499.71989/17.94	0.28634	-1.8042	8.6347e-07	6.0638
8	173.08163/13.94	1.7988	0.84702	1.2364e-06	5.9078
9	592.1392/14.93	2.7586	1.4639	1.3117e-06	5.8822
10	187.09745/16.64	1.7912	0.84092	1.5173e-06	5.8189
11	197.11032/18.33	2.415	1.272	1.5792e-06	5.8016
12	155.07161/18.96	4.6327	2.2118	1.7411e-06	5.7592
13	144.04568/16.2	1.8317	0.87319	2.0924e-06	5.6793
14	417.2123/18.49	2.3703	1.2451	2.4965e-06	5.6027
15	517.19481/14.63	0.51732	-0.95086	3.7438e-06	5.4267
16	533.17637/17.7	2.7074	1.4369	4.9285e-06	5.3073
17	323.18613/20.23	2.5319	1.3402	6.0488e-06	5.2183
18	435.0655/16.99	4.1168	2.0415	7.4065e-06	5.1304
19	631.27995/17.98	2.9181	1.545	7.5021e-06	5.1248
20	537.12574/13.76	1.6194	0.69543	9.27e-06	5.0329
21	213.11261/17.42	2.6604	1.4116	9.3837e-06	5.0276
22	522.13493/16.42	1.7128	0.77633	9.722e-06	5.0122
23	447.25891/18.03	2.071	1.0503	1.0785e-05	4.9672
24	243.1233/16.21	2.148	1.103	1.1935e-05	4.9232
25	231.15918/15.64	0.30268	-1.7242	1.2807e-05	4.8926
26	355.18757/21.87	3.3091	1.7264	1.5771e-05	4.8021
27	665.28412/14.42	1.9817	0.98674	1.635e-05	4.7865
28	175.0975/14.87	0.57815	-0.79049	1.6553e-05	4.7811
29	632.27935/17.86	2.3863	1.2548	1.737e-05	4.7602
30	147.01186/7.27	2.4853	1.3134	2.1323e-05	4.6712
31	413.18019/16.22	1.7679	0.82207	2.1838e-05	4.6608
32	199.09784/15.58	1.9287	0.9476	2.3574e-05	4.6276
33	501.1806/16.44	2.3066	1.2058	2.4042e-05	4.619
34	175.07633/14.87	0.54999	-0.86253	2.4417e-05	4.6123
35	499.21584/17.89	0.39257	-1.349	2.7042e-05	4.568
36	199.1334/19	2.2679	1.1813	2.7345e-05	4.5631
37	251.00445/14.65	2.3104	1.2082	2.8737e-05	4.5416
38	435.08298/16.95	3.2181	1.6862	3.2503e-05	4.4881
39	227.09225/15.85	6.5138	2.7035	3.6357e-05	4.4394
40	324.19066/9.25	0.49021	-1.0285	3.9666e-05	4.4016
41	504.13812/10.12	0.44444	-1.1699	4.7667e-05	4.3218
42	789.22709/14.97	1.8354	0.87612	5.1066e-05	4.2919
43	522.17375/8.93	2.679	1.4217	5.1145e-05	4.2912
44	574.26607/17.71	0.42568	-1.2321	5.1267e-05	4.2902
45	490.15067/14.25	0.34351	-1.5416	5.6267e-05	4.2497
46	315.14299/16.36	3.3012	1.723	5.7354e-05	4.2414
47	490.13636/14.17	0.39792	-1.3295	5.8167e-05	4.2353
48	170.09035/20.93	5.5106	2.4622	5.8756e-05	4.2309
49	121.0292/14.63	1.8184	0.86264	6.0596e-05	4.2176
50	599.20318/10.81	1.9676	0.97646	6.1623e-05	4.2103

2.2 Correlation Analysis

Correlation analysis can be used to visualize the overall correlations between different features It can also be used to identify which features are correlated with a feature of interest. Correlation analysis can also be used to identify if certain features show particular patterns under different conditions. Users first need to define a pattern in the form of a series of hyphenated numbers. For example, in a time-series study with four time points, a pattern of of `1-2-3-4` is used to search compounds with increasing the concentration as time changes; while a pattern of `3-2-1-3` can be used to search compounds that decrease at first, then bounce back to the original level.

Figure 5 shows the overall correlation heatmap.

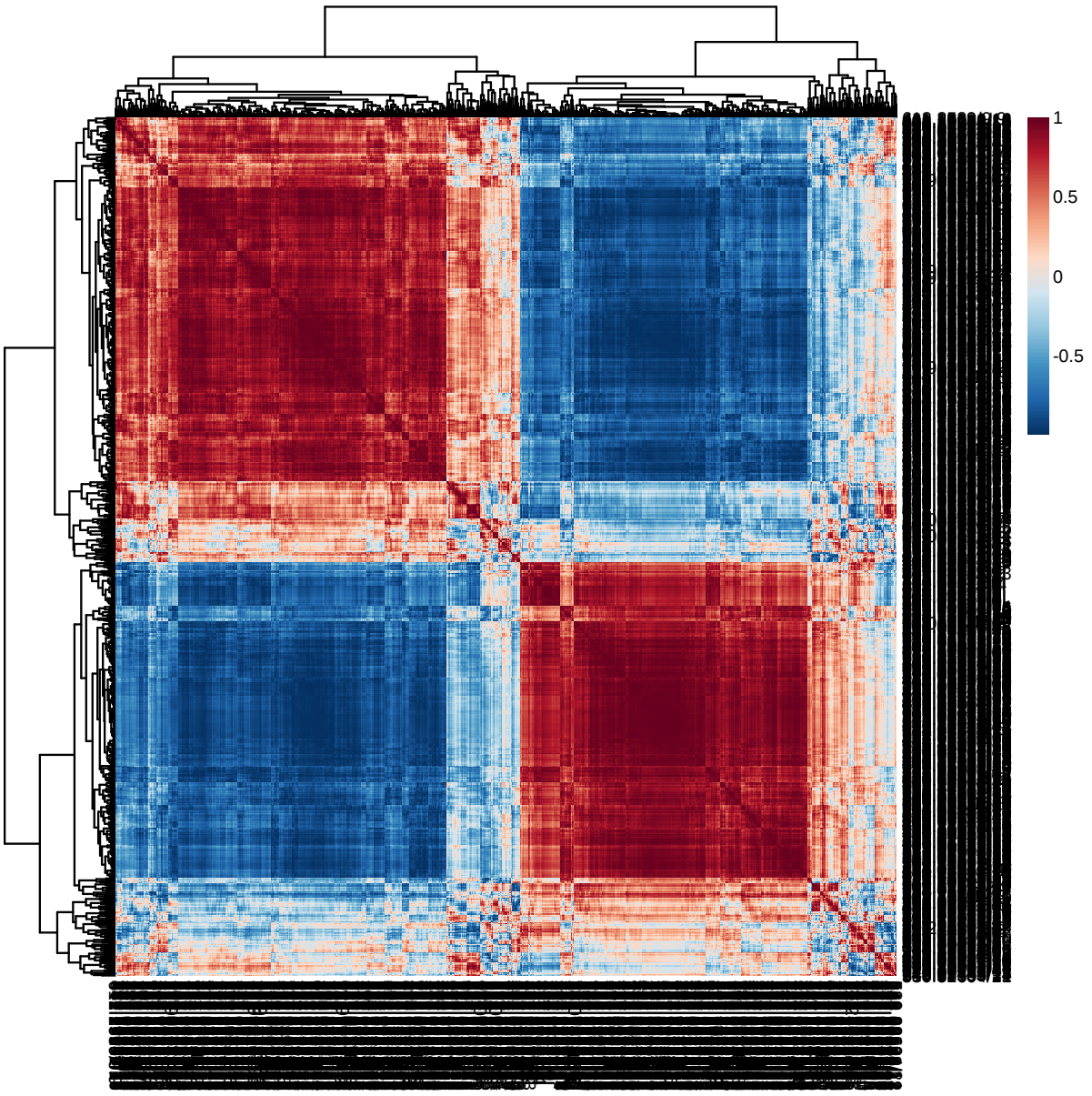


Figure 5: Correlation Heatmaps

2.3 Principal Component Analysis (PCA)

PCA is an unsupervised method aiming to find the directions that best explain the variance in a data set (X) without referring to class labels (Y). The data are summarized into much fewer variables called *scores* which are weighted average of the original variables. The weighting profiles are called *loadings*. The PCA analysis is performed using the `prcomp` package. The calculation is based on singular value decomposition.

The Rscript `chemometrics.R` is required. Figure 6 is pairwise score plots providing an overview of the various seperation patterns among the most significant PCs; Figure 7 is the scree plot showing the variances explained by the selected PCs; Figure 8 shows the 2-D scores plot between selected PCs; Figure 9 shows the 3-D scores plot between selected PCs; Figure 10 shows the loadings plot between the selected PCs; Figure 11 shows the biplot between the selected PCs.

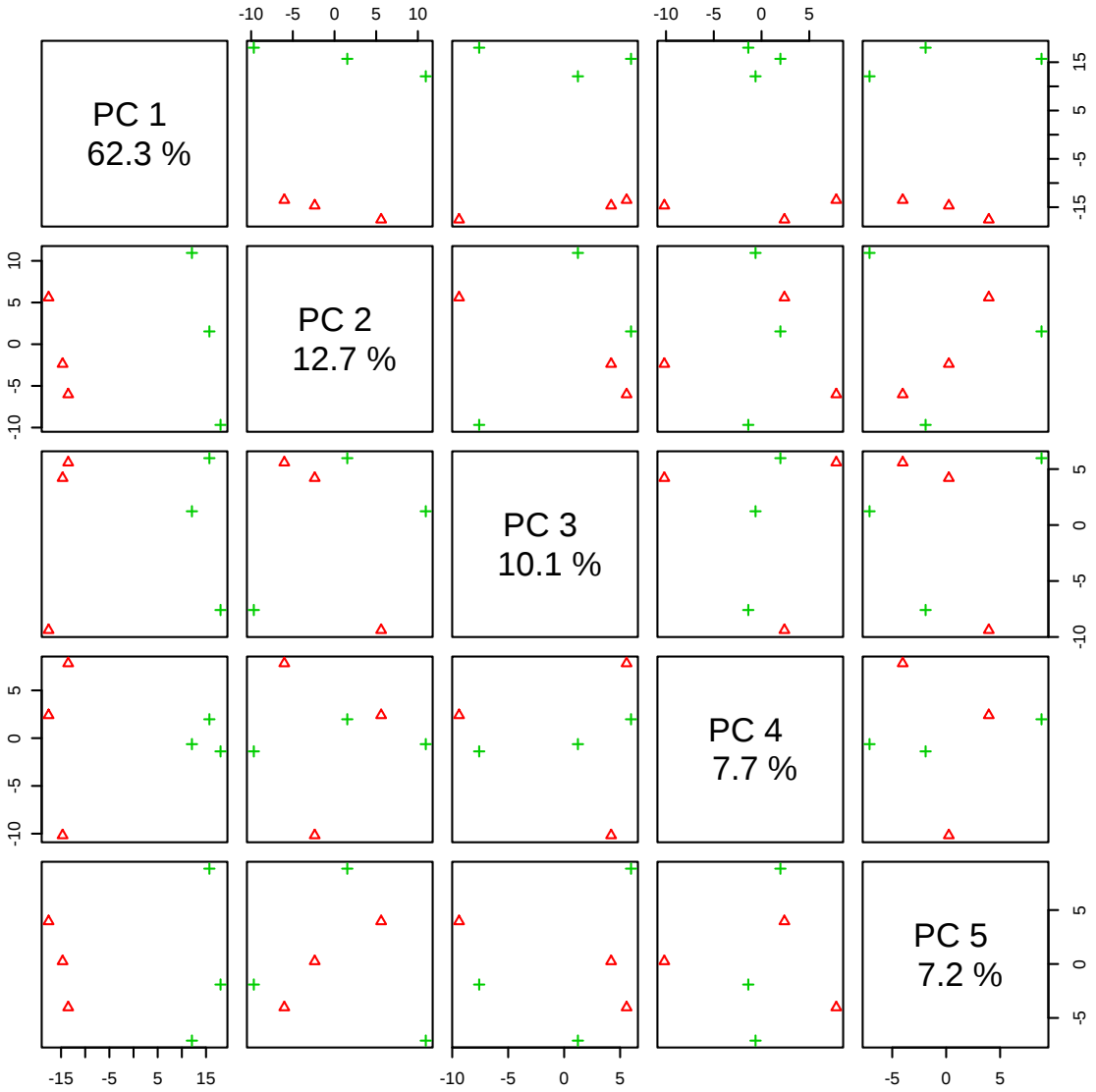


Figure 6: Pairwise score plots between the selected PCs. The explained variance of each PC is shown in the corresponding diagonal cell.

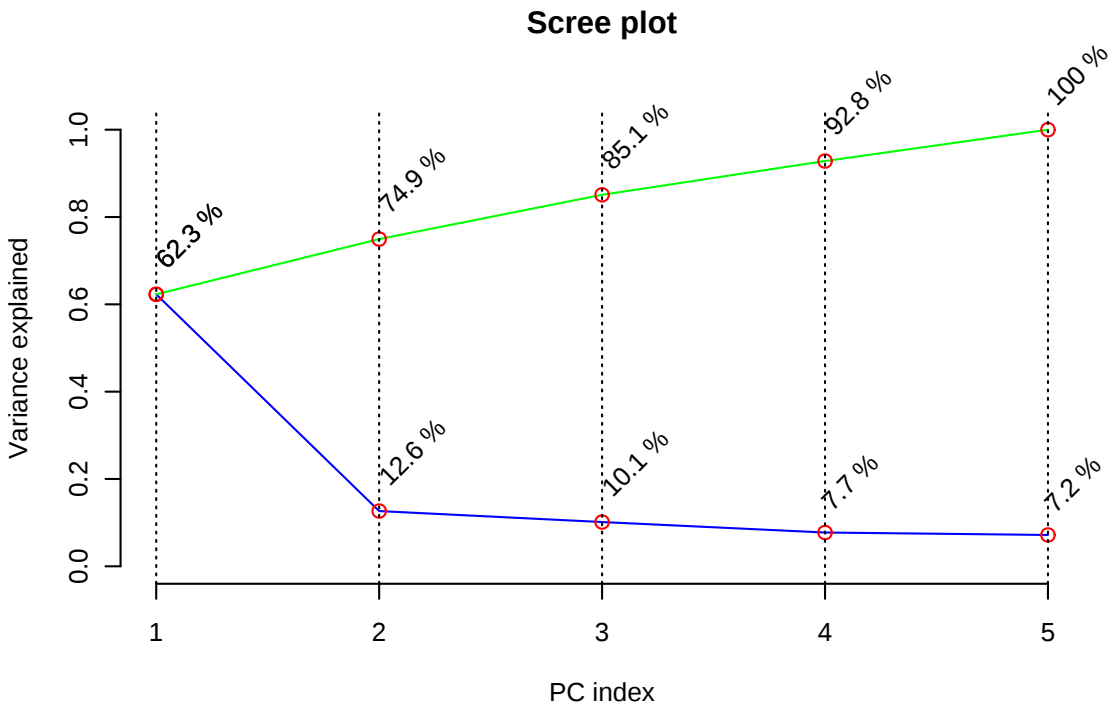


Figure 7: Scree plot shows the variance explained by PCs. The green line on top shows the accumulated variance explained; the blue line underneath shows the variance explained by individual PC.

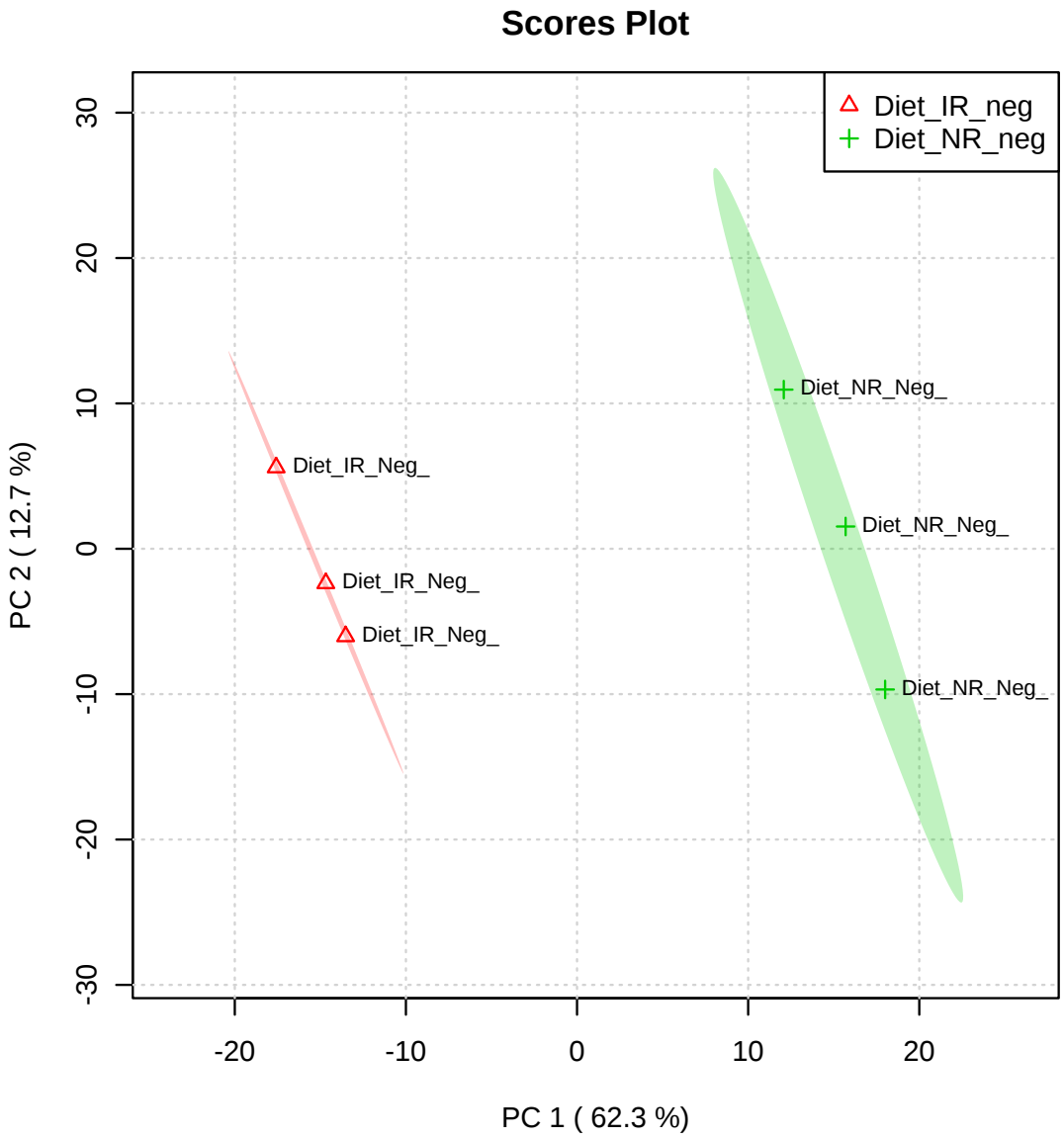


Figure 8: Scores plot between the selected PCs. The explained variances are shown in brackets.

Figure 9: 3D score plot between the selected PCs. The explained variances are shown in brackets.

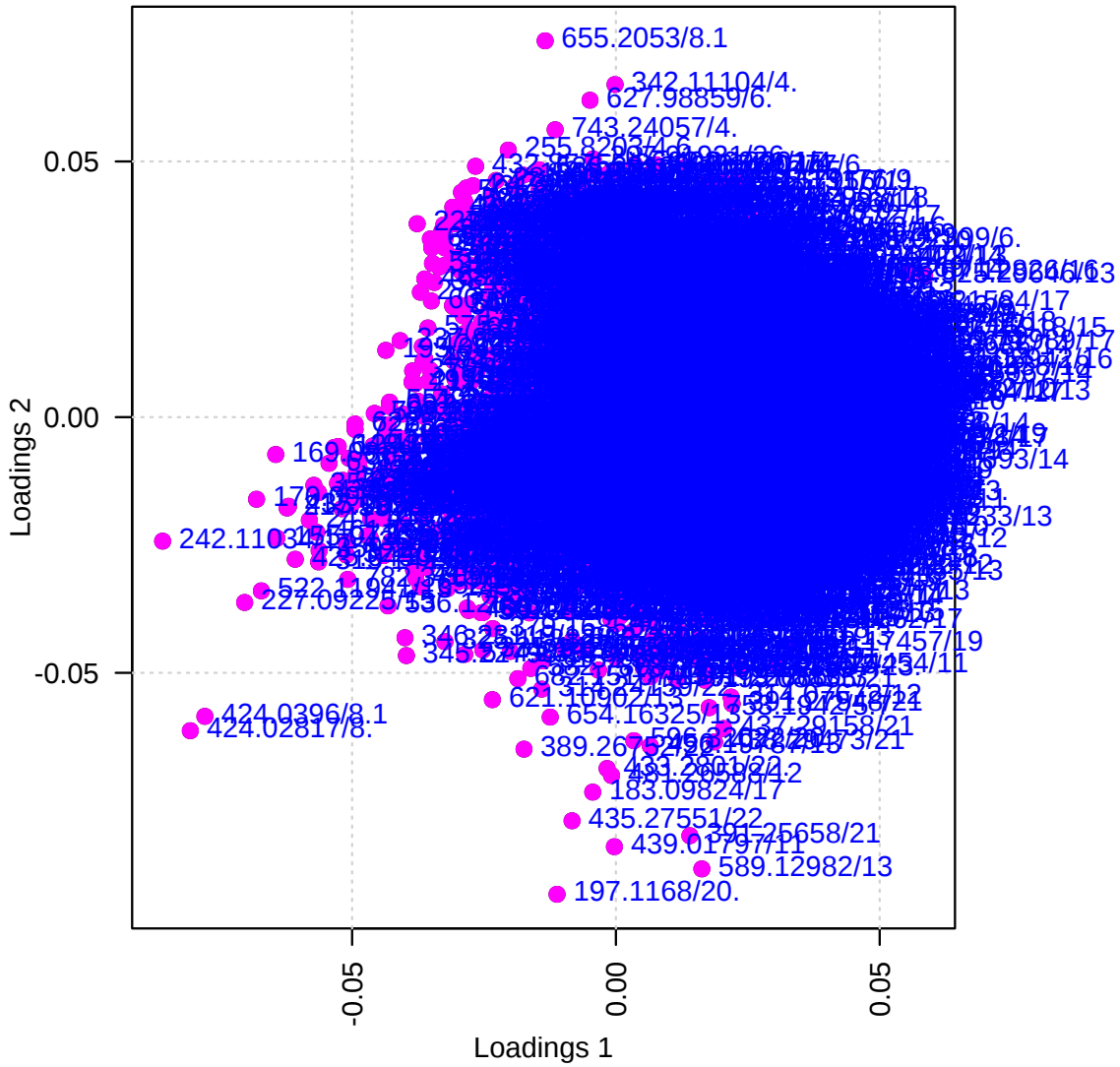


Figure 10: Loadings plot for the selected PCs.

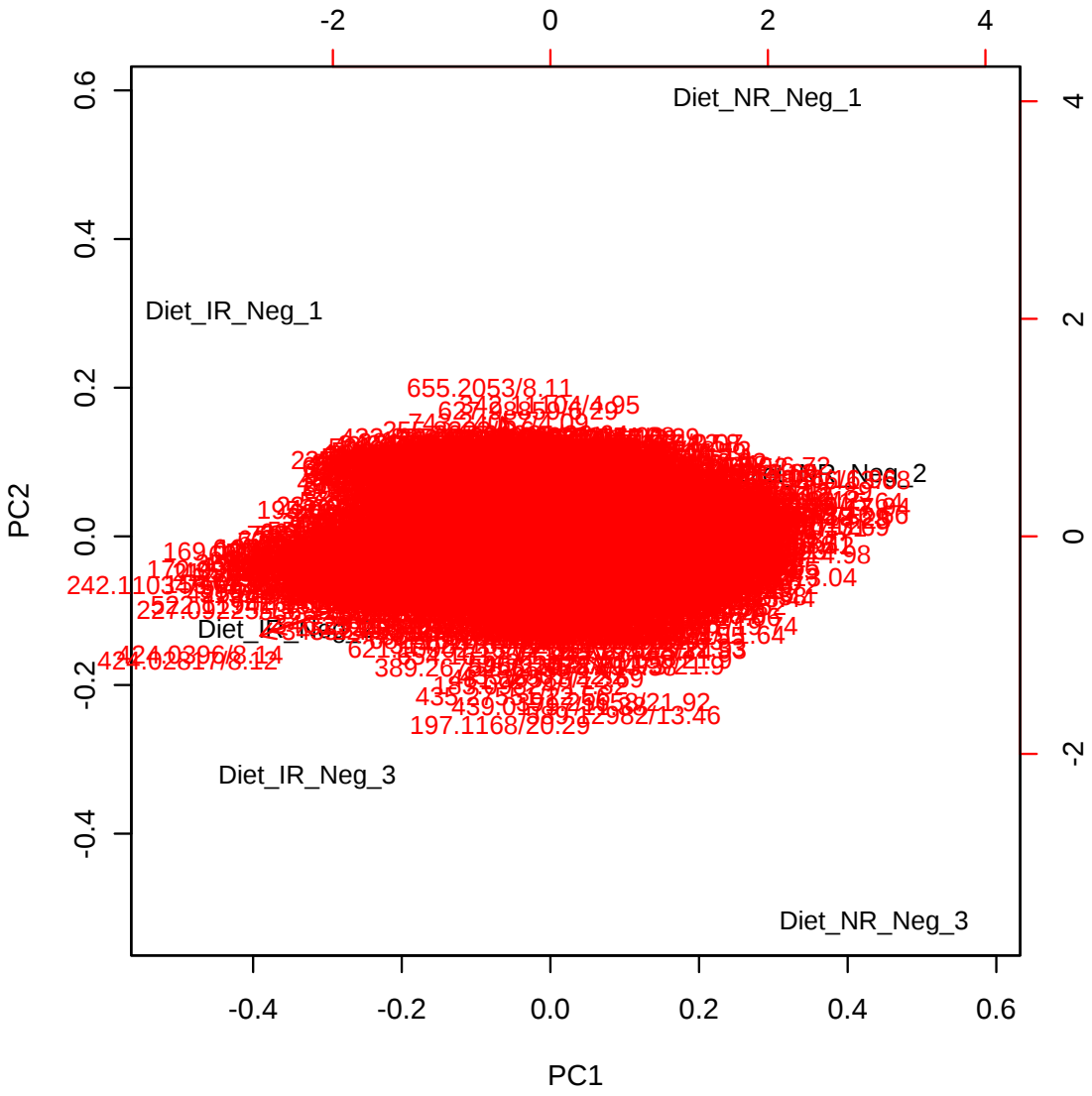


Figure 11: PCA biplot between the selected PCs. Note, you may want to test different centering and scaling normalization methods for the biplot to be displayed properly.

2.4 Partial Least Squares - Discriminant Analysis (PLS-DA)

PLS is a supervised method that uses multivariate regression techniques to extract via linear combination of original variables (X) the information that can predict the class membership (Y). The PLS regression is performed using the `plsr` function provided by R `pls` package[4]. The classification and cross-validation are performed using the corresponding wrapper function offered by the `caret` package[5].

To assess the significance of class discrimination, a permutation test was performed. In each permutation, a PLS-DA model was built between the data (X) and the permuted class labels (Y) using the optimal number of components determined by cross validation for the model based on the original class assignment. MetaboAnalyst supports two types of test statistics for measuring the class discrimination. The first one is based on prediction accuracy during training. The second one is separation distance based on the ratio of the between group sum of the squares and the within group sum of squares (B/W-ratio). If the observed test statistic is part of the distribution based on the permuted class assignments, the class discrimination cannot be considered significant from a statistical point of view.[6].

There are two variable importance measures in PLS-DA. The first, Variable Importance in Projection (VIP) is a weighted sum of squares of the PLS loadings taking into account the amount of explained Y-variation in each dimension. Please note, VIP scores are calculated for each components. When more than componetnts are used to calculate the feature importance, the average of the VIP scores are used. The other importance measure is based on the weighted sum of PLS-regression. The weights are a function of the reduction of the sums of squares across the number of PLS components. Please note, for multiple-group (more than two) analysis, the same number of predictors will be built for each group. Therefore, the coefficient of each feature will be different depending on which group you want to predict. The average of the feature coefficients are used to indicate the overall coefficient-based importance.

Figure 12 shows the overview of scores plots; Figure 13 shows the 2-D scores plot between selected components; Figure 14 shows the 3-D scores plot between selected components; Figure 15 shows the loading plot between the selected components;Figure 16 shows the classification performance with different number of components; Figure 17 shows the results of permutation test for model validation; Figure 18 shows important features identified by PLS-DA.

[4]Ron Wehrens and Bjorn-Helge Mevik.*pls: Partial Least Squares Regression (PLSR) and Principal Component Regression (PCR)*, 2007, R package version 2.1-0

[5]Max Kuhn. Contributions from Jed Wing and Steve Weston and Andre Williams.*caret: Classification and Regression Training*, 2008, R package version 3.45

[6]Bijlsma et al.*Large-Scale Human Metabolomics Studies: A Strategy for Data (Pre-) Processing and Validation*, Anal Chem. 2006, 78 567 - 574

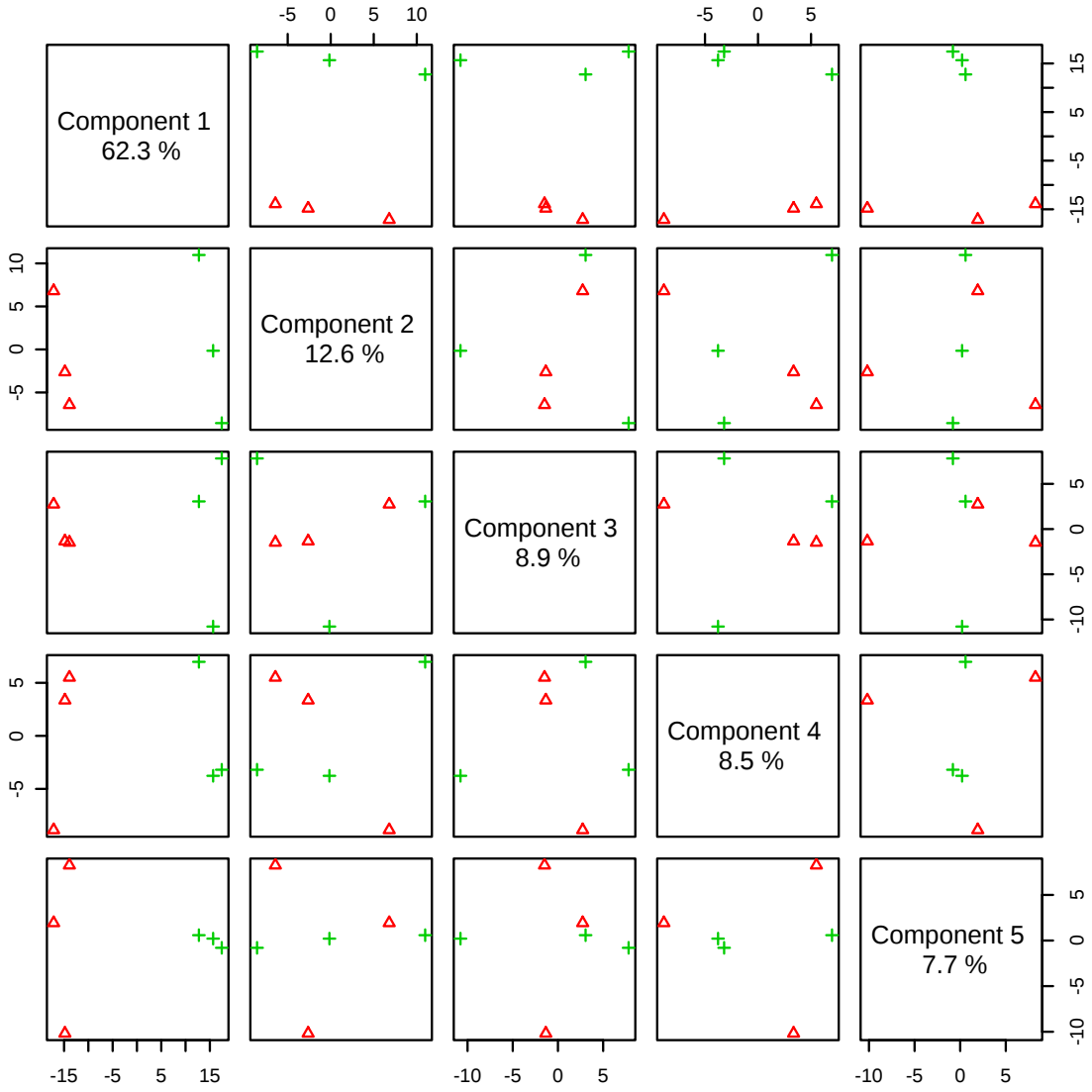


Figure 12: Pairwise scores plots between the selected components. The explained variance of each component is shown in the corresponding diagonal cell.

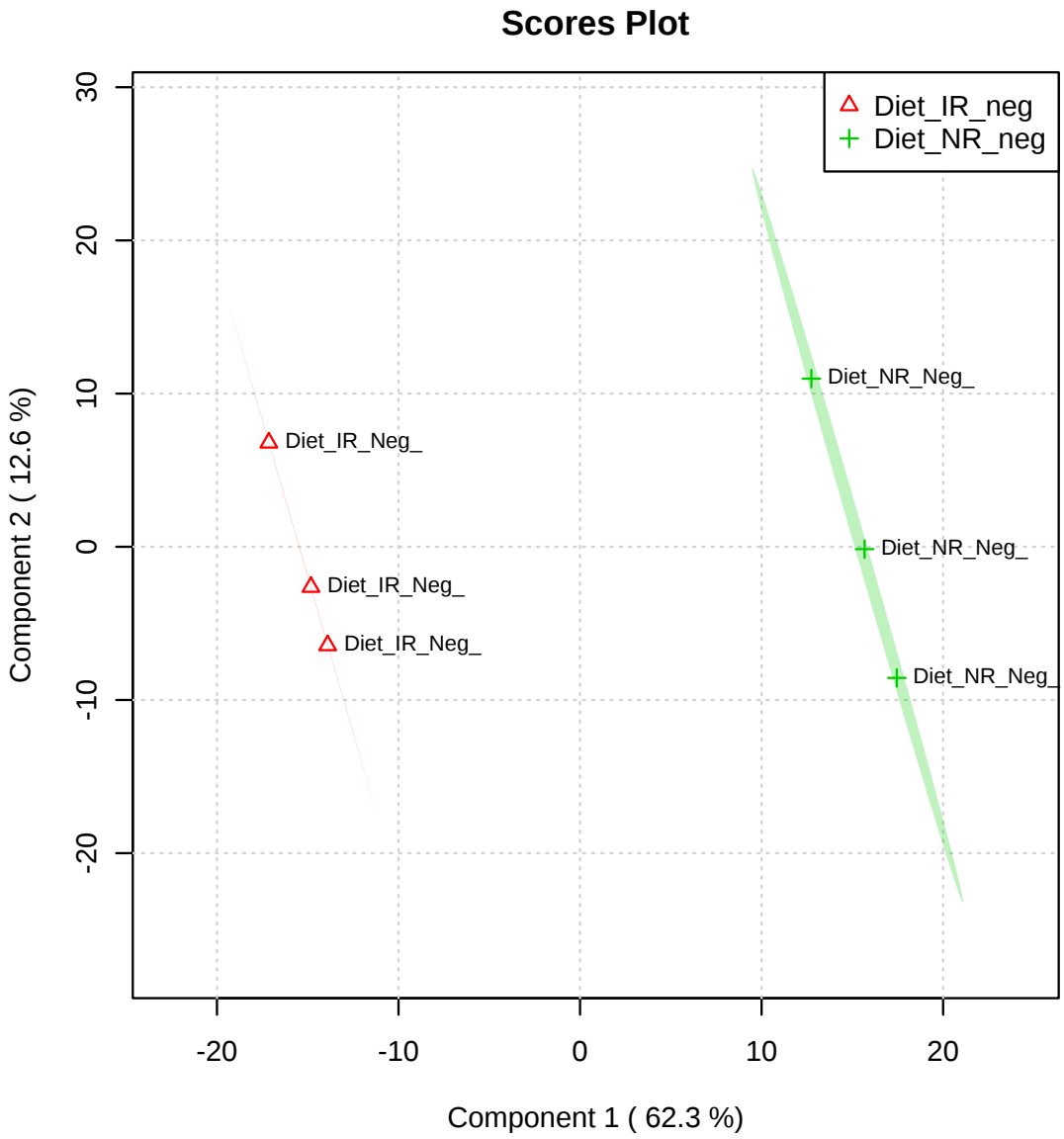


Figure 13: Scores plot between the selected PCs. The explained variances are shown in brackets.

Figure 14: 3D scores plot between the selected PCs. The explained variances are shown in brackets.

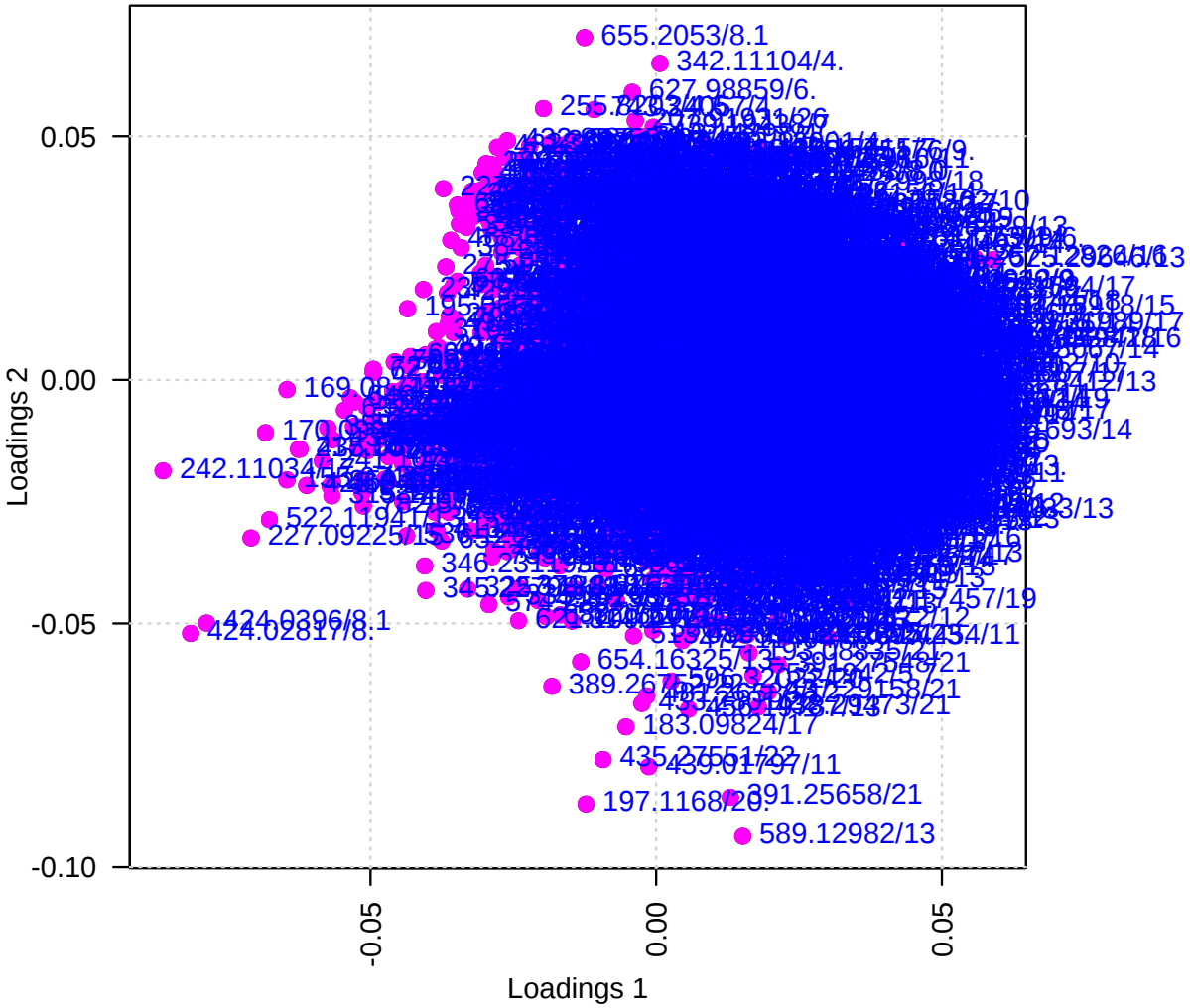


Figure 15: Loadings plot between the selected PCs.

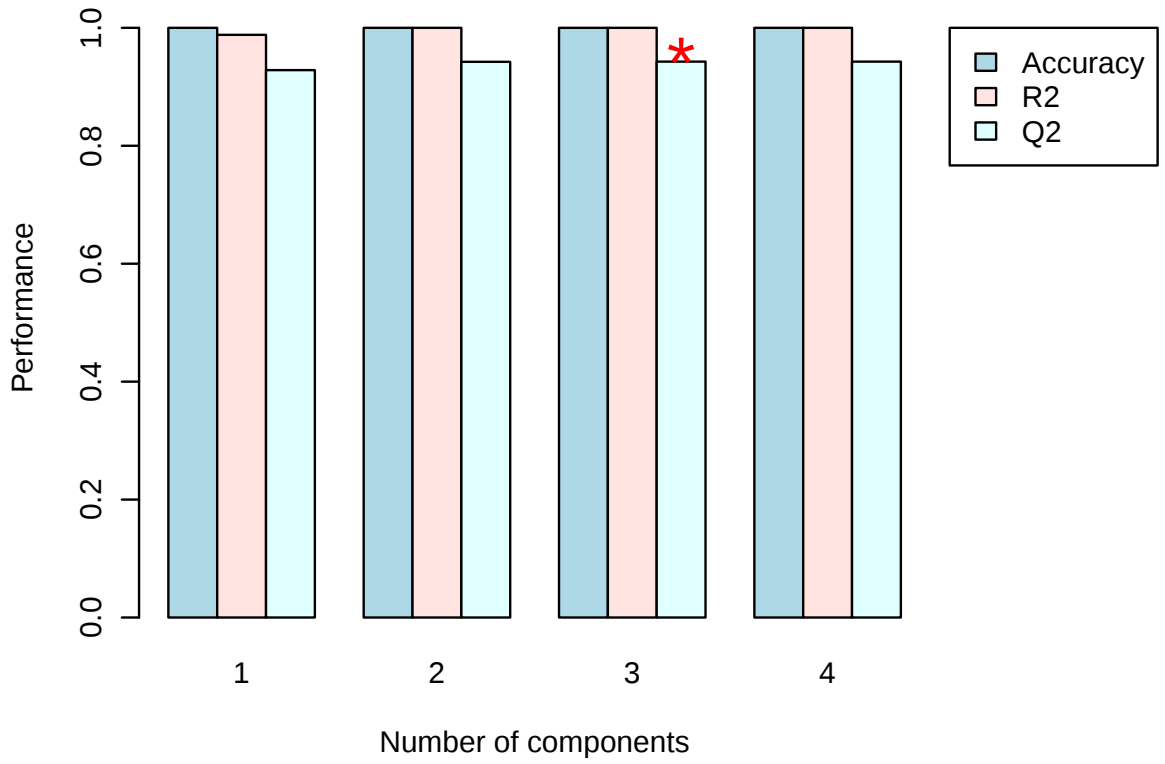


Figure 16: PLS-DA classification using different number of components. The red circle indicates the best classifier.

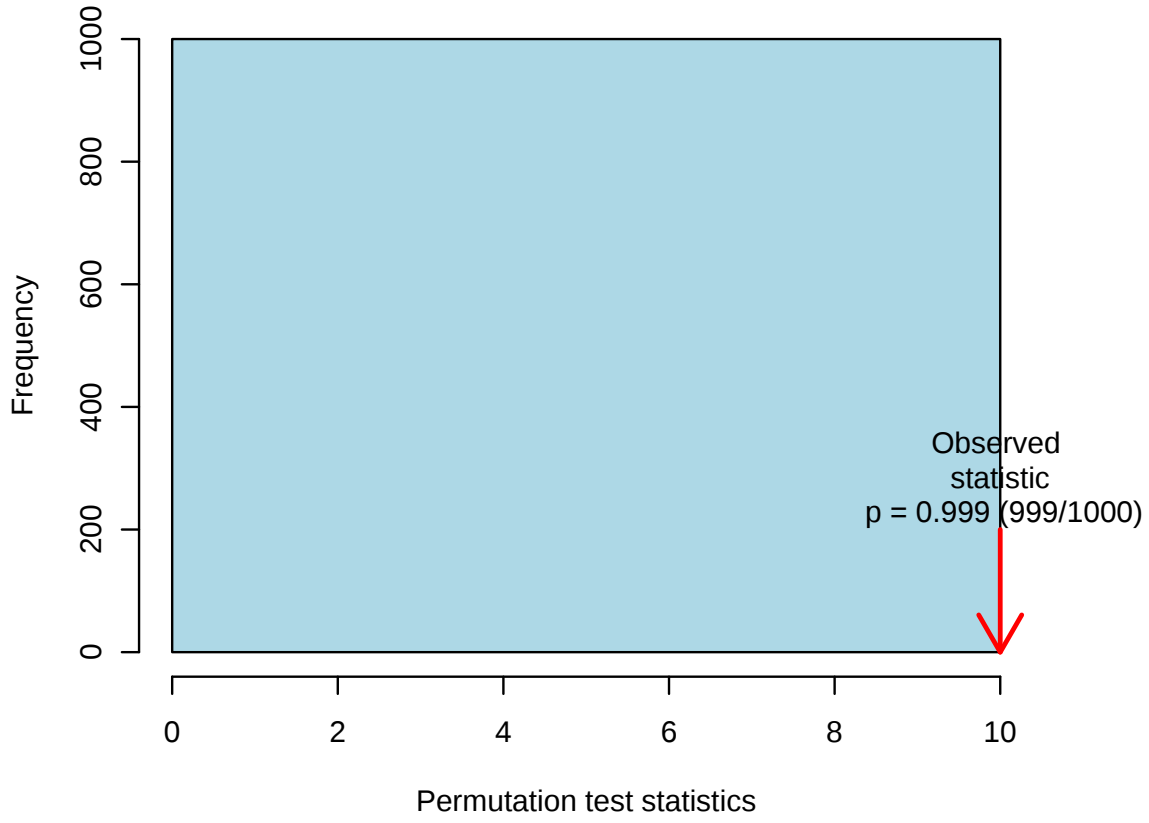


Figure 17: PLS-DA model validation by permutation tests based on separation distance. The p value based on permutation is p = 0.999 (999/1000).

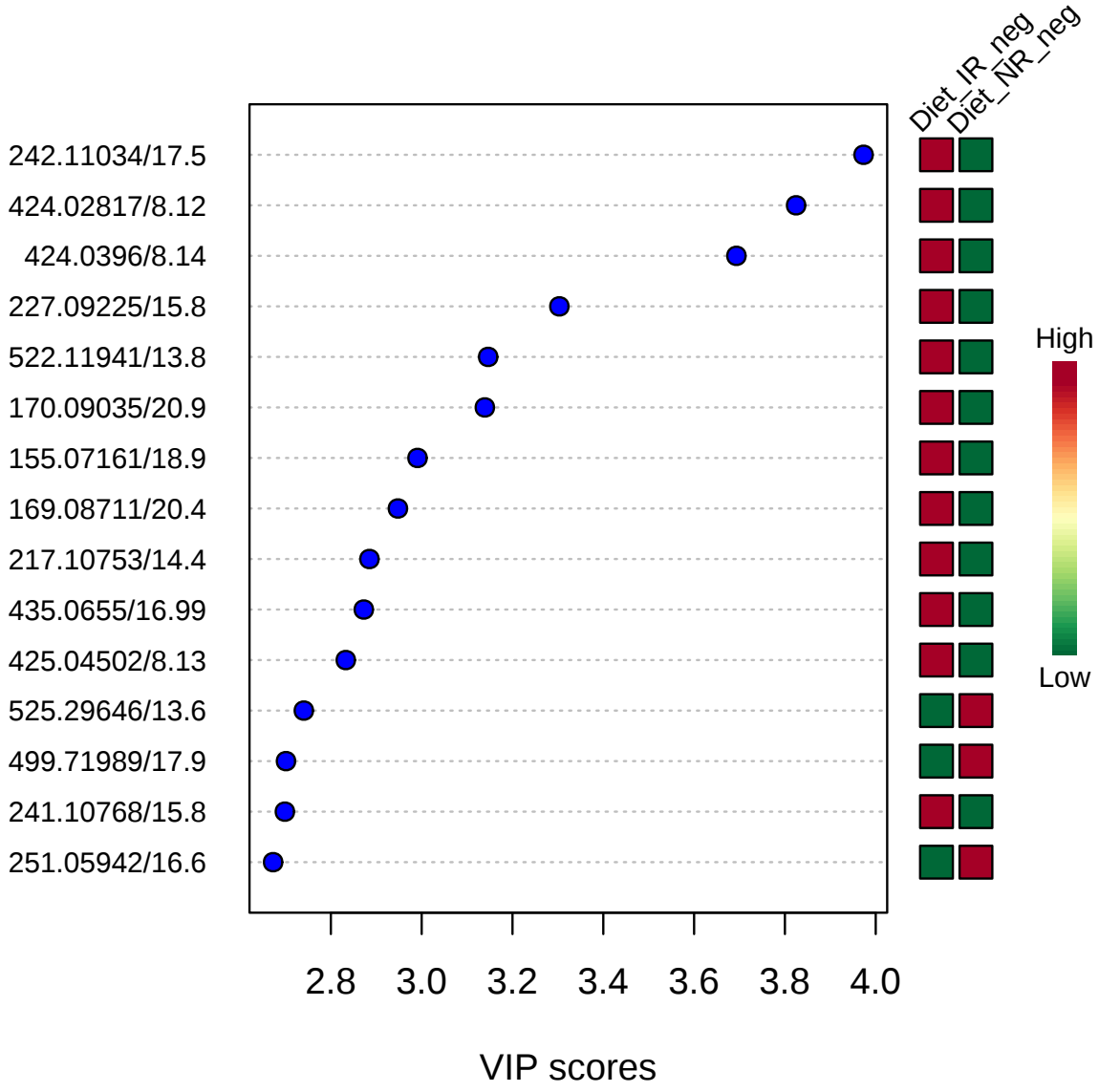


Figure 18: Important features identified by PLS-DA. The colored boxes on the right indicate the relative concentrations of the corresponding metabolite in each group under study.

2.5 Significance Analysis of Microarray (SAM)

SAM is a well-established statistical method for identification of differentially expressed genes in microarray data analysis. It is designed to address the false discovery rate (FDR) when running multiple tests on high-dimensional microarray data. SAM assigns a significance score to each variable based on its change relative to the standard deviation of repeated measurements. For a variable with scores greater than an adjustable threshold, its relative difference is compared to the distribution estimated by random permutations of the class labels. For each threshold, a certain proportion of the variables in the permutation set will be found to be significant by chance. The proportion is used to calculate the FDR. SAM is performed using the `siggenes` package[7]. Users need to specify the `Delta` value to control FDR in order to proceed.

Figure 19 shows the significant features identified by SAM. Table 5 shows the details of these features.

Table 5: Important features identified by SAM

	Peaks (mz/rt)	d.value	stdev	rawp	q.value
1	242.11034/17.59	-3.4198	0.046459	4.8193e-05	0.044263
2	227.09225/15.85	-2.6275	0.11047	9.6386e-05	0.044263
3	155.07161/18.96	-2.5737	0.046641	0.00014458	0.044263
4	217.10753/14.43	-2.5671	0.020923	0.00019277	0.044263
5	170.09035/20.93	-2.4734	0.1185	0.00024096	0.044263
6	522.11941/13.88	-2.4698	0.12172	0.00028916	0.044263
7	435.0655/16.99	-2.4168	0.064394	0.00033735	0.044263
8	241.10768/15.87	-2.3975	0.022147	0.00038554	0.044263
9	424.02817/8.12	-2.3897	0.3409	0.00043373	0.044263
10	424.0396/8.14	-2.3718	0.31183	0.00048193	0.044263
11	169.08711/20.44	-2.2472	0.14706	0.00057831	0.044263
12	355.18757/21.87	-2.2016	0.071558	0.0006747	0.044263
13	435.08298/16.95	-2.1399	0.084805	0.00072289	0.044263
14	315.14299/16.36	-2.1275	0.098987	0.00077108	0.044263
15	592.1392/14.93	-2.1257	0.035366	0.00081928	0.044263
16	631.27995/17.98	-2.1248	0.056213	0.00086747	0.044263
17	425.04502/8.13	-2.1041	0.17033	0.00096386	0.044263
18	359.20669/21.18	-2.0906	0.10961	0.001012	0.044263
19	417.10292/10.97	-2.087	0.01735	0.0010602	0.044263

[7]Holger Schwender. *siggenes: Multiple testing using SAM and Efron's empirical Bayes approaches*,2008, R package version 1.16.0

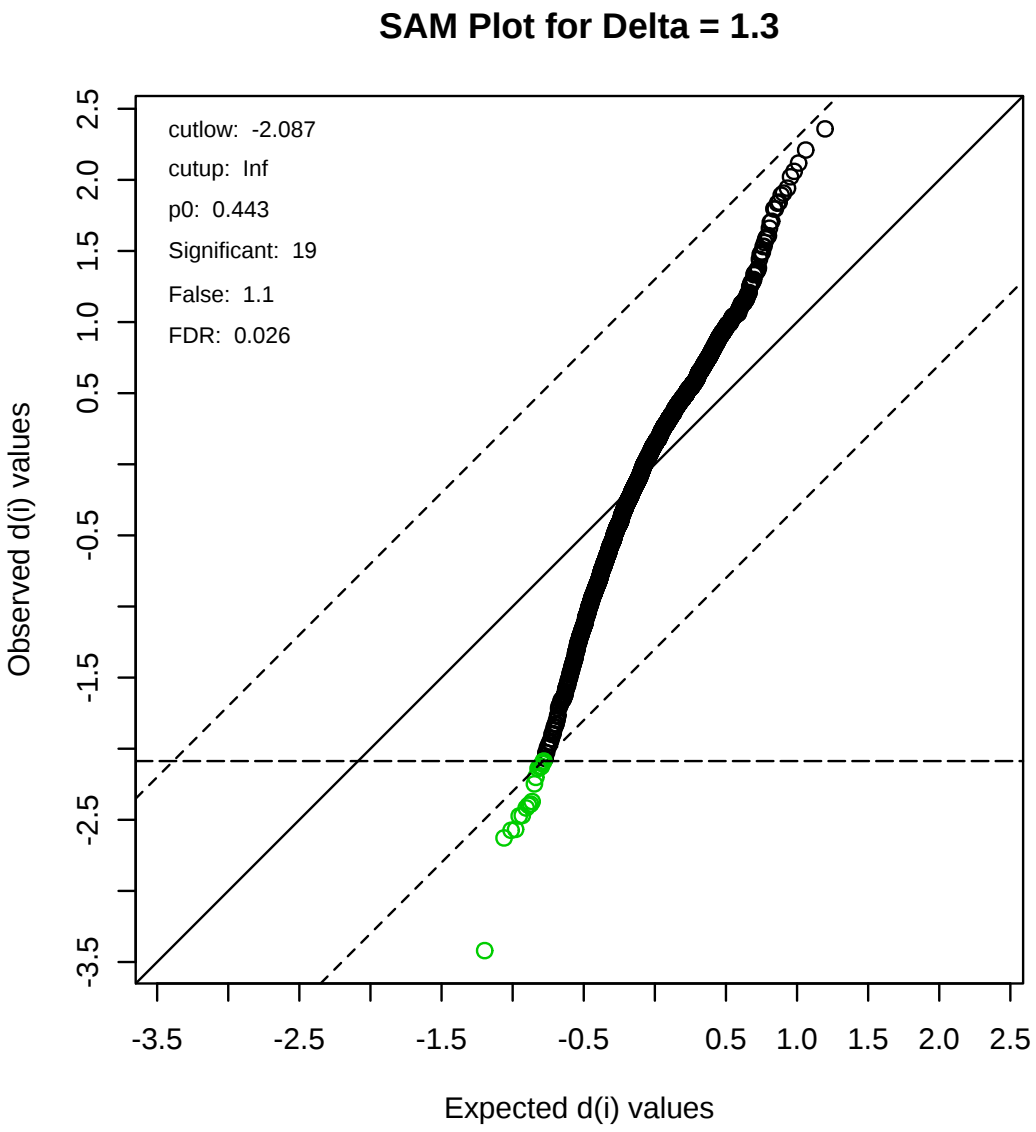


Figure 19: Significant features identified by SAM. The green circles represent features that exceed the specified threshold.

2.6 Empirical Bayesian Analysis of Microarray (EBAM)

EBAM is an empirical Bayesian method based on moderated t-statistics. EBAM uses a two-group mixture model for null and significant features. The prior and density parameters are estimated from the data. A feature is considered significant if its calculated posterior is larger than or equal to `delta` and no other features with a more extreme test score that is not called signicant. The default is `delta` = 0.9. The suggested fudge factor (`a0`) is chosen that leads to the largest number of significant features. EBAM is performed with `ebam` function in `siggenes` package[8].

Figure 20 shows the important features identified by EBAM.

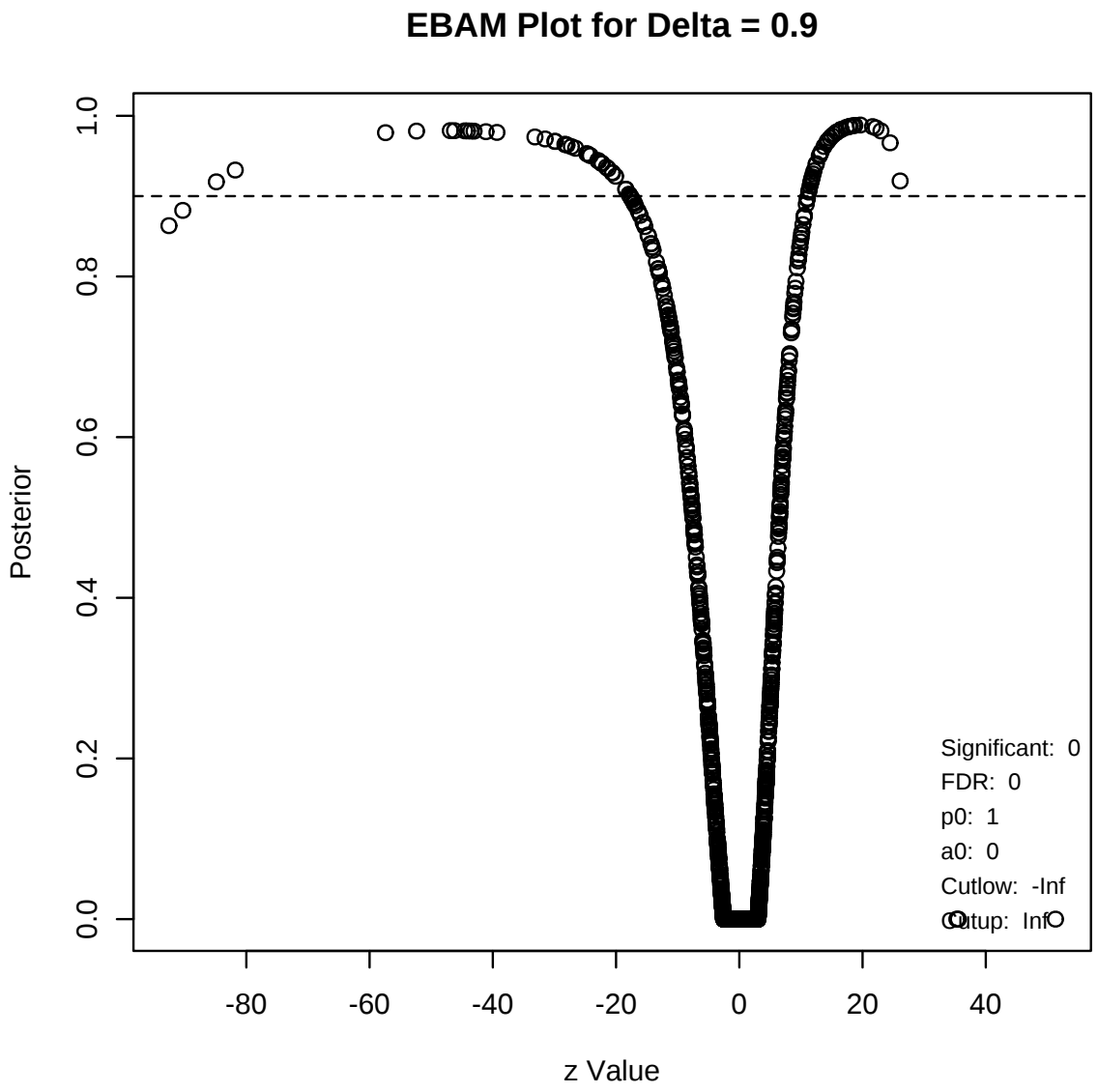


Figure 20: Significant features identified by EBAM. The green circles represent features that exceed the specified threshold.

[1] "No significant features were found using the given threshold for EBAM"

[8] Holger Schwender. *siggenes: Multiple testing using SAM and Efron's empirical Bayes approaches*,2008,R package version 1.16.0

2.7 Hierarchical Clustering

In (agglomerative) hierarchical cluster analysis, each sample begins as a separate cluster and the algorithm proceeds to combine them until all samples belong to one cluster. Two parameters need to be considered when performing hierarchical clustering. The first one is similarity measure - Euclidean distance, Pearson's correlation, Spearman's rank correlation. The other parameter is clustering algorithms, including average linkage (clustering uses the centroids of the observations), complete linkage (clustering uses the farthest pair of observations between the two groups), single linkage (clustering uses the closest pair of observations) and Ward's linkage (clustering to minimize the sum of squares of any two clusters). Heatmap is often presented as a visual aid in addition to the dendrogram.

Hierachical clustering is performed with the `hclust` function in package `stat`. Figure 21 shows the clustering result in the form of a dendrogram. Figure 22 shows the clustering result in the form of a heatmap.

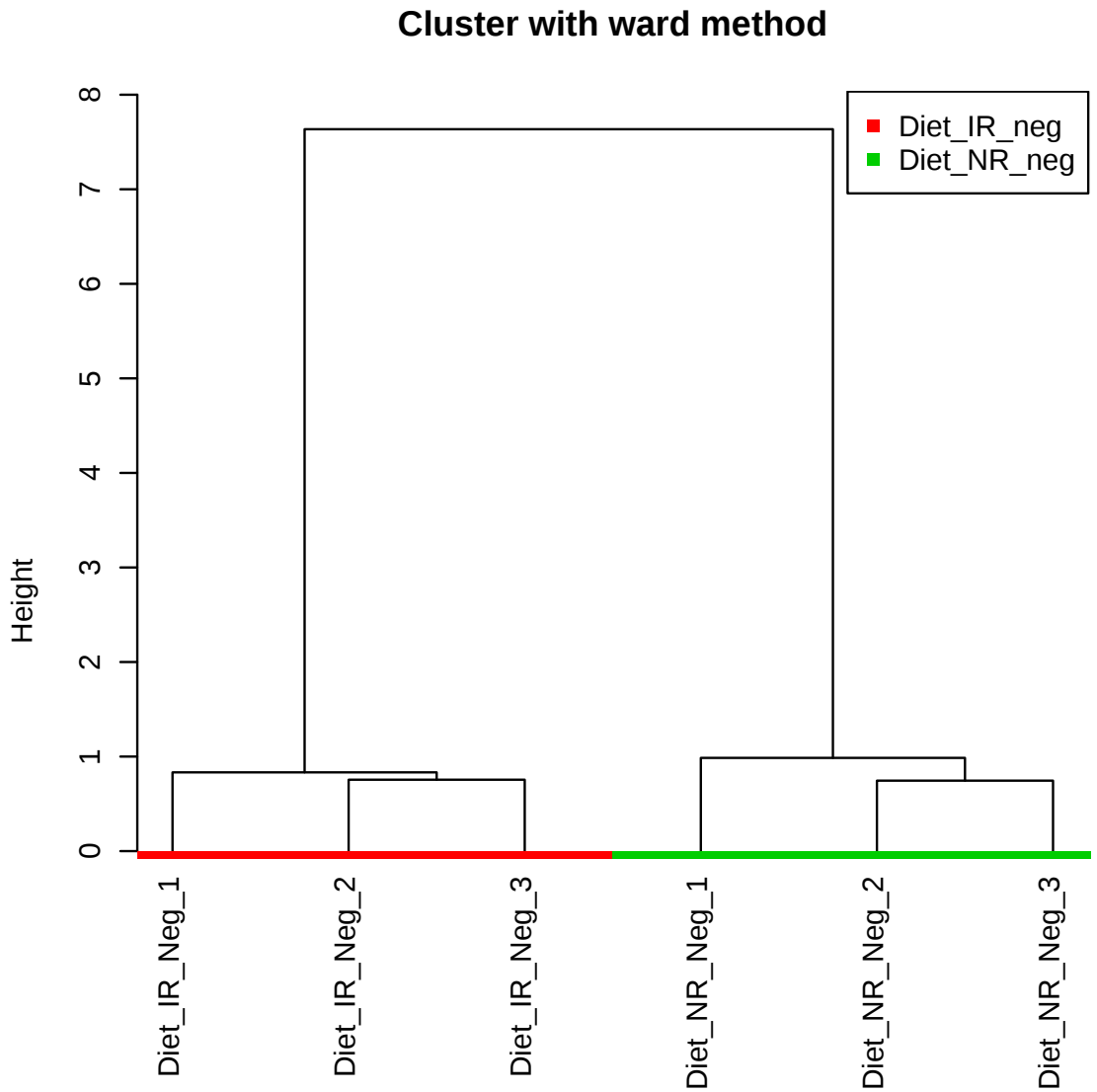
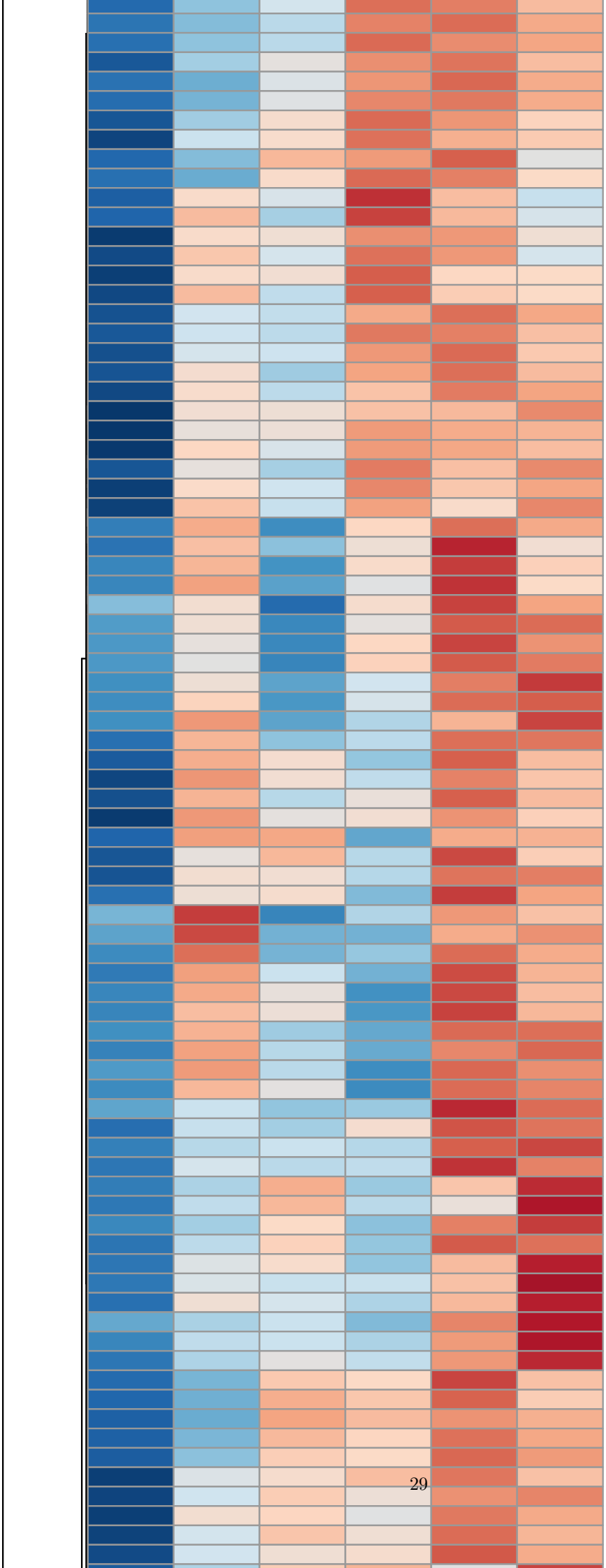


Figure 21: Clustering result shown as dendrogram (distance measure using `pearson`, and clustering algorithm using `ward`).


584.23904/
252.0821/1
611.27817/
245.89538/
154.02139/
395.20529/
239.09121/
579.26662/
157.05403/
753.18495/
455.15021/
584.17259/
450.20351/
530.04181/
339.0924/4
578.1236/1
243.89918/
284.06951/
744.33995/
353.07004/
373.06101/
760.23472/
146.96558/
773.22052/
208.93633/
442.11469/
677.05871/
113.9892/2
378.17229/
68.99716/2
112.98655/
290.09122/
277.03061/
290.07745/
571.12919/
454.08892/
518.15211/
87.00423/2
226.97979/
115.00418/
495.10859/
214.88095/
683.29796/
391.09961/
454.28831/
383.05676/
398.09594/
443.08999/
518.17119/
535.16183/
215.03182/
86.00802/2
732.21075/
634.18543/
717.65762/
132.91648/
393.0963/4
777.16138/
695.18589/
93.03288/1
427.03935/
192.02269/
523.20848/
234.98303/
405.02702/
137.02448/
111.00859/
288.98015/
191.01974/
182.0447/1
539.20125/
290.17003/
462.11188/
390.21328/
478.10319/
759.23537/
221.94646/
353.05165/
360.02988/
453.2863/2
492.27564/

2.8 K-means Clustering

K-means clustering is a nonhierarchical clustering technique. It begins by creating k random clusters (k is supplied by user). The program then calculates the mean of each cluster. If an observation is closer to the centroid of another cluster then the observation is made a member of that cluster. This process is repeated until none of the observations are reassigned to a different cluster.

K-means analysis is performed using the `kmeans` function in the package `stat`. Figure 23 shows clustering the results. Table 6 shows the members in each cluster from K-means analysis.

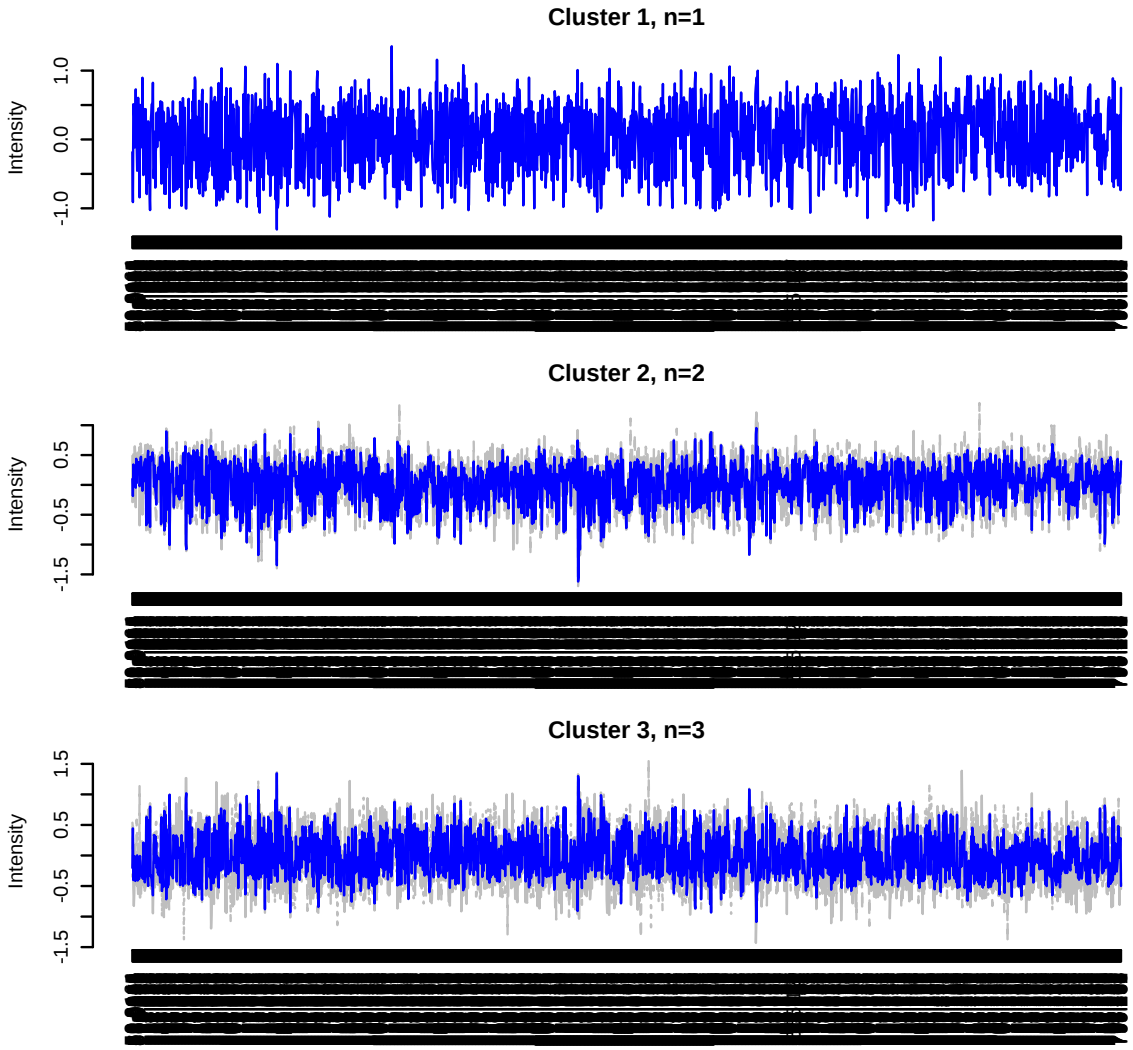


Figure 23: K-means cluster analysis. The x-axes are variable indices and y-axes are relative intensities. The blue lines represent median intensities of corresponding clusters

Table 6: Clustering result using K-means

	Samples in each cluster
Cluster(1)	Diet_NR_Neg_3
Cluster(2)	Diet_NR_Neg_1 Diet_NR_Neg_2
Cluster(3)	Diet_IR_Neg_1 Diet_IR_Neg_2 Diet_IR_Neg_3

2.9 Self Organizing Map (SOM)

SOM is an unsupervised neural network algorithm used to automatically identify major trends present in high-dimensional data. SOM is based on a grid of interconnected nodes, each of which represents a model. These models begin as random values, but during the process of iterative training they are updated to represent different subsets of the training set. Users need to specify the x and y dimension of the grid to perform SOM analysis.

The SOM is performed using the R `som` package[9]. Figure 24 shows the SOM clustering results. Table 7 shows the members in each cluster from SOM analysis.

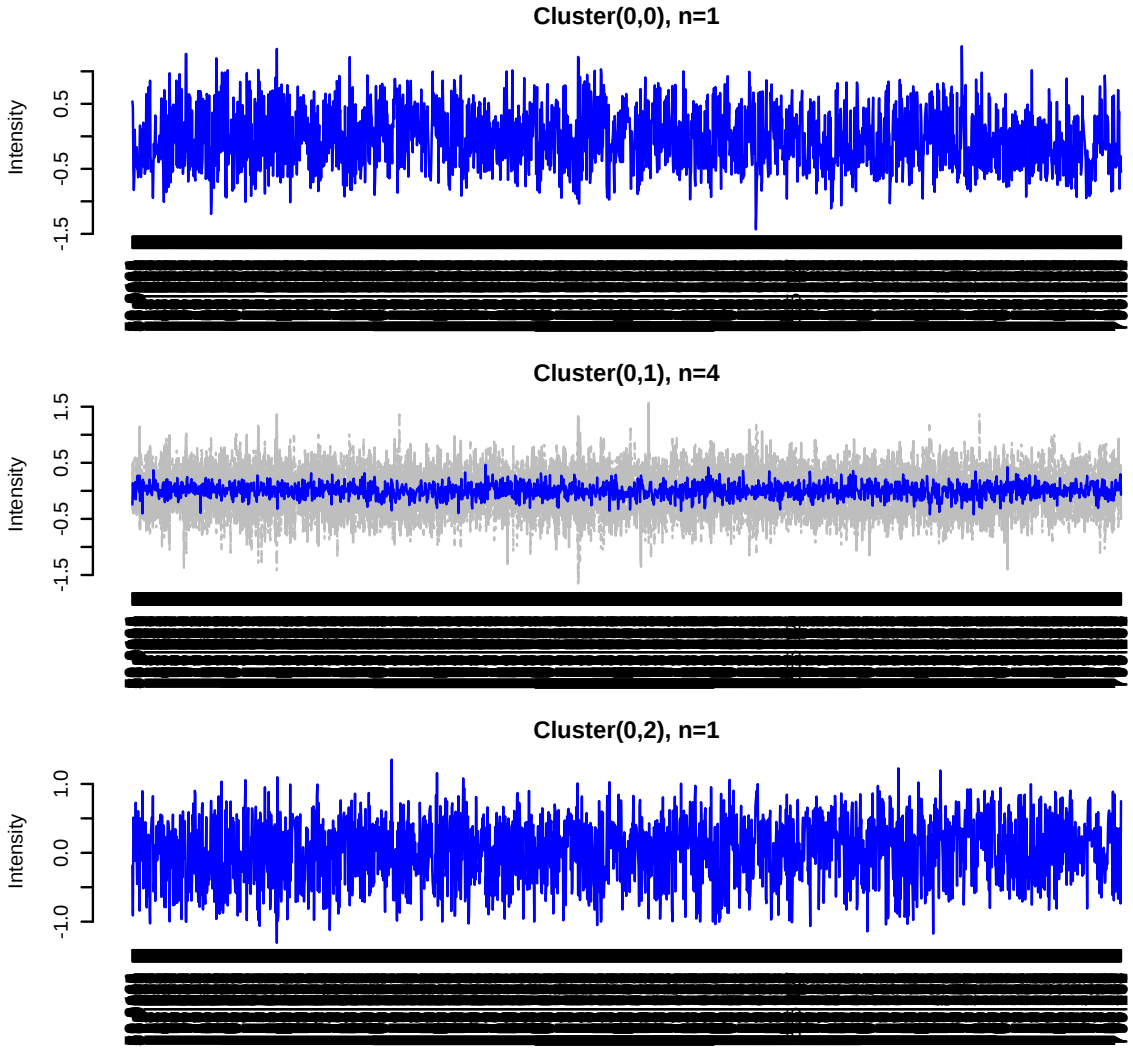


Figure 24: SOM cluster analysis. The x-axes are features and y-axes are relative intensities. The blue lines represent median intensities of corresponding clusters

Table 7: Clustering result using SOM

	Samples in each cluster
Cluster(0 , 0)	Diet_IR_Neg_1
Cluster(0 , 1)	Diet_IR_Neg_2 Diet_IR_Neg_3 Diet_NR_Neg_1 Diet_NR_Neg_2
Cluster(0 , 2)	Diet_NR_Neg_3

[9] Jun Yan. *som: Self-Organizing Map*, 2004, R package version 0.3-4

2.10 Random Forest (RF)

Random Forest is a supervised learning algorithm suitable for high dimensional data analysis. It uses an ensemble of classification trees, each of which is grown by random feature selection from a bootstrap sample at each branch. Class prediction is based on the majority vote of the ensemble. RF also provides other useful information such as OOB (out-of-bag) error, variable importance measure, and outlier measures. During tree construction, about one-third of the instances are left out of the bootstrap sample. This OOB data is then used as test sample to obtain an unbiased estimate of the classification error (OOB error). Variable importance is evaluated by measuring the increase of the OOB error when it is permuted. The outlier measures are based on the proximities during tree construction.

RF analysis is performed using the `randomForest` package[10]. Table 8 shows the confusion matrix of random forest. Figure 25 shows the cumulative error rates of random forest analysis for given parameters. Figure 26 shows the important features ranked by random forest. Figure 27 shows the outlier measures of all samples for the given parameters. The OOB error is 0

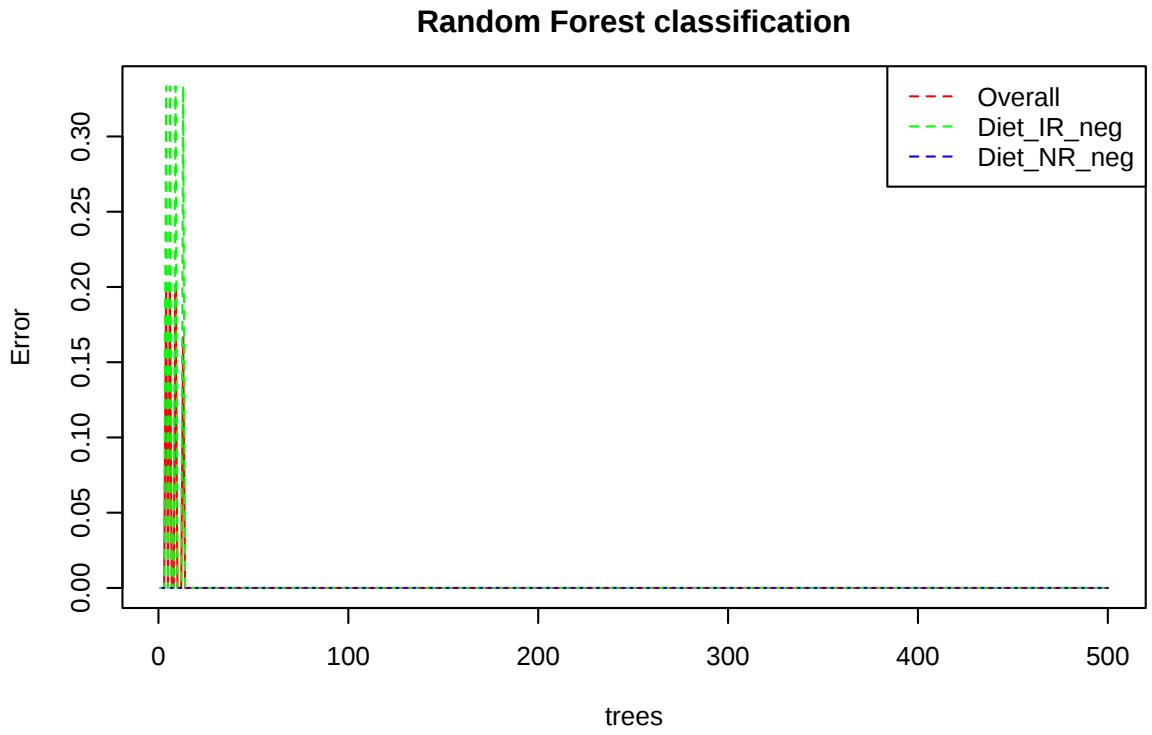


Figure 25: Cumulative error rates by Random Forest classification. The overall error rate is shown as the black line; the red and green lines represent the error rates for each class.

	Diet_IR_neg	Diet_NR_neg	class.error
Diet_IR_neg	3.00	0.00	0.00
Diet_NR_neg	0.00	3.00	0.00

Table 8: Random Forest Classification Performance

[10] Andy Liaw and Matthew Wiener. *Classification and Regression by randomForest*, 2002, R News

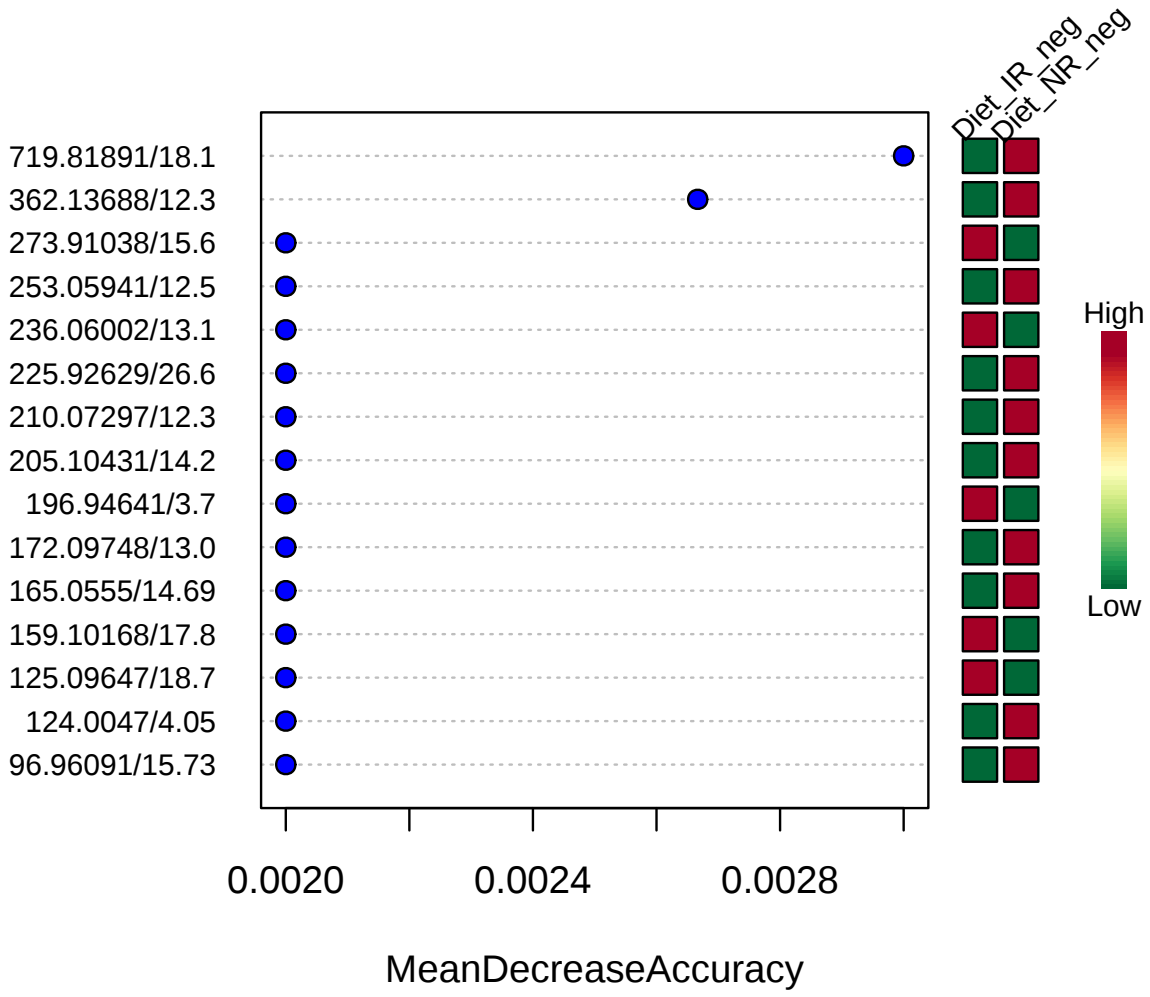


Figure 26: Significant features identified by Random Forest. The features are ranked by the mean decrease in classification accuracy when they are permuted.

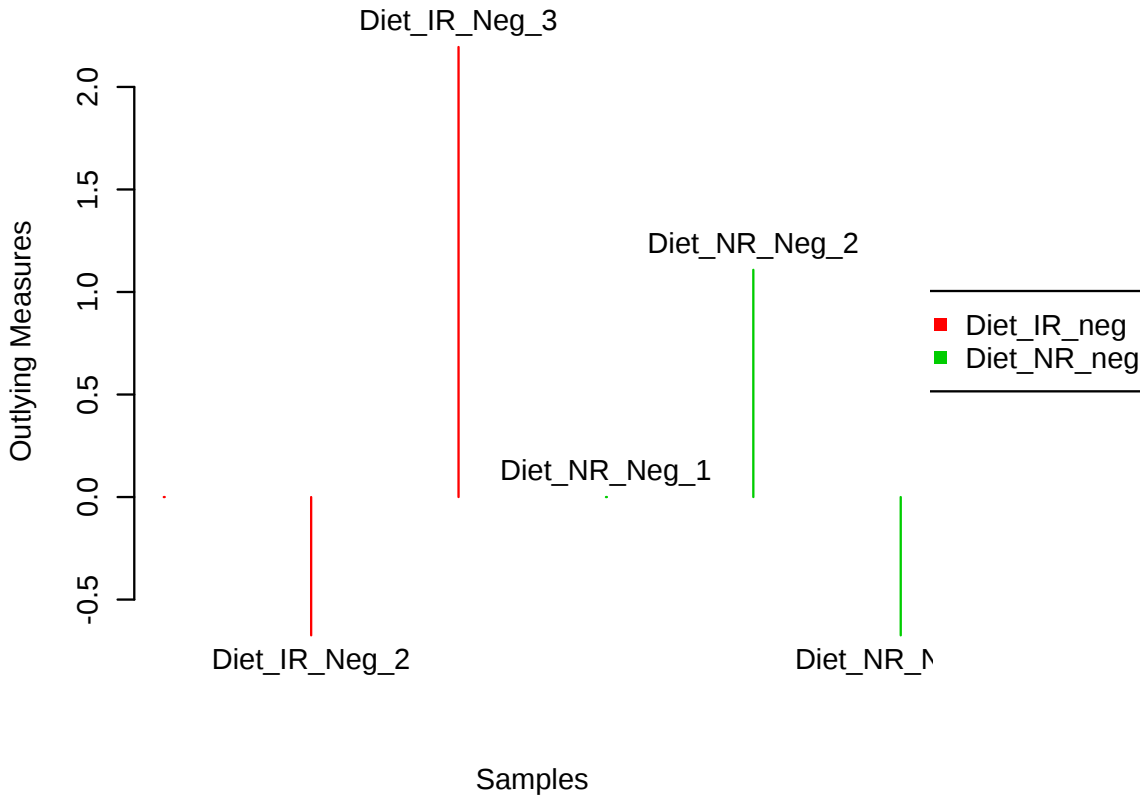


Figure 27: Potential outliers identified by Random Forest. Only the top five are labeled.

2.11 Support Vector Machine (SVM)

SVM aims to find a nonlinear decision function in the input space by mapping the data into a higher dimensional feature space and separating it there by means of a maximum margin hyperplane. The SVM-based recursive feature selection and classification is performed using the `R-SVM` script[11]. The process is performed recursively using decreasing series of feature subsets (`ladder`) so that different classification models can be calculated. Feature importance is evaluated based on its frequencies being selected in the best classifier identified by recursive classification and cross-validation. Please note, R-SVM is very computationally intensive. Only the top 50 features (ranked by their p values from t-tests) will be evaluated.

In total, 11 models (levels) were created using 2075, 830, 332, 133, 66, 33, 20, 15, 11, 8, 6 selected feature subsets. Figure 28 shows the SVM classification performance using recursive feature selection. Figure 29 shows the signicant features used by the best classifiers.

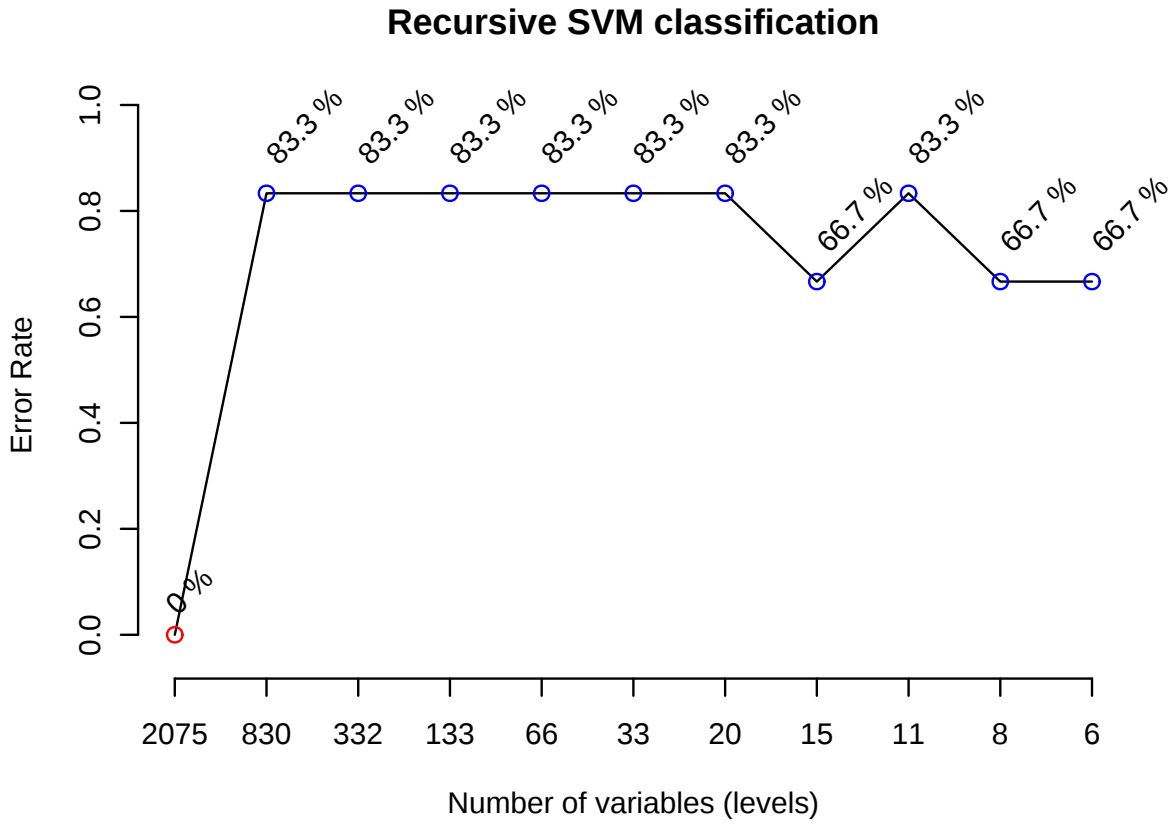


Figure 28: Recursive classification with SVM. The red circle indicates the best classifier.

[11] http://www.hsph.harvard.edu/bioinfocore/RSVMhome/R-SVM.html

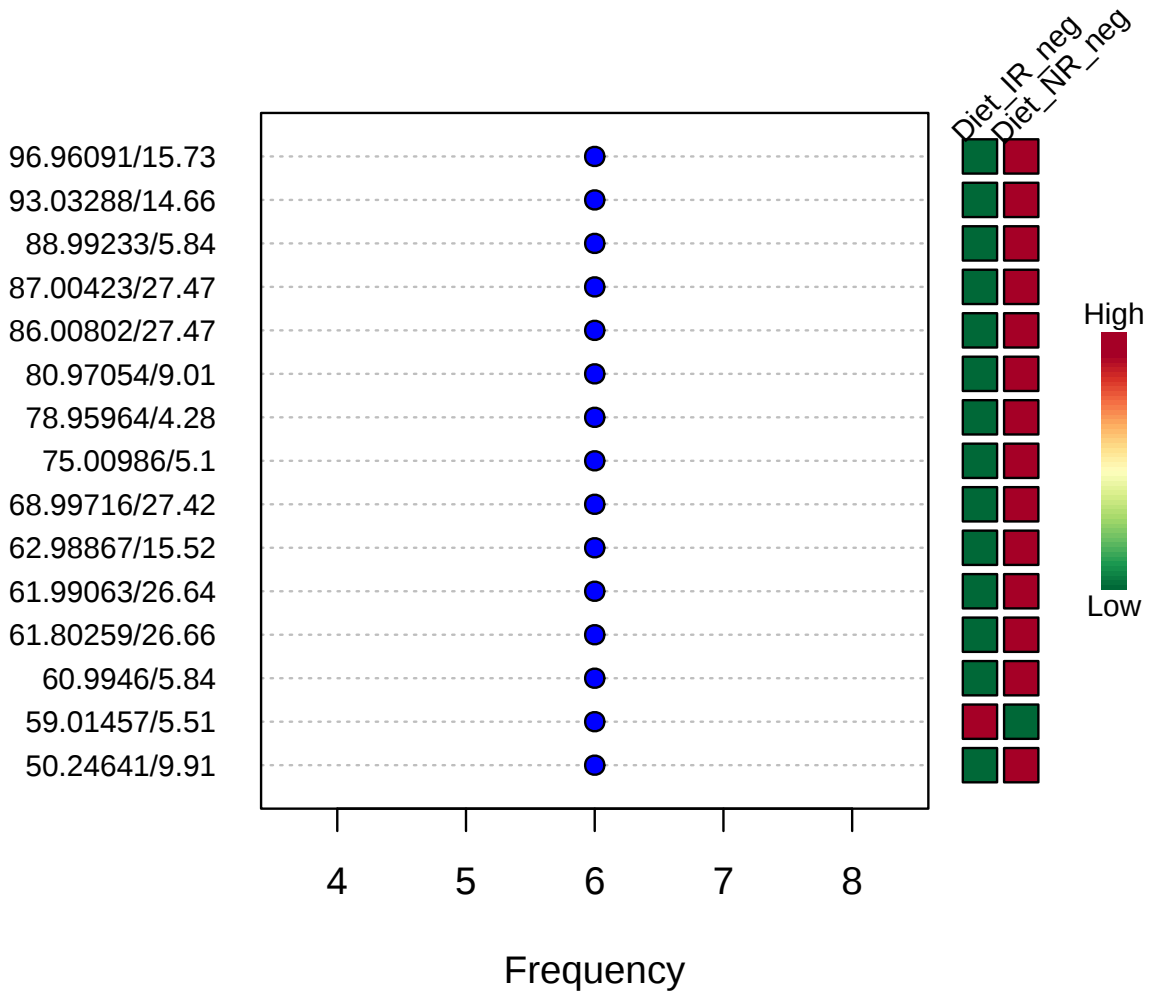


Figure 29: Significant features identified by R-SVM. Features are ranked by their frequencies of being selected in the classifer.

3 Data Annotation

Please be advised that MetaboAnalyst also supports metabolomic data annotation. For NMR, MS, or GC-MS peak list data, users can perform peak identification by searching the corresponding libraries. For compound concentration data, users can perform metabolite set enrichment analysis and metabolic pathway analysis.

The report was generated on Sun Feb 8 08:17:00 2015 with R version 3.0.3 (2014-03-06). Thank you for using MetaboAnalyst! For suggestions and feedback please contact Jeff Xia (*jianguox@ualberta.ca*).