Metabolomic Data Analysis with MetaboAnalyst 3.0

User ID: guest3415138159334245273

August 30, 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 Concentration Data

The concentration data should be uploaded in comma separated values (.csv) format. Samples can be in rows or columns, with class labels immediately following the sample IDs.

Samples are in columns and features in rows. The uploaded file is in comma separated values (.csv) format. The uploaded data file contains 12 (samples) by 481 (compounds) data matrix.

1.1.2 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)

1.1.3 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.

[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

43 variables were removed for threshold 50 percent. Missing variables were imputated using KNN

1.1.4 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.

No data filtering was applied

Table 1: Summary of data processing results

	Features (positive)	Missing/Zero	Features (processed)
ND_Veh_1	472	9	438
ND_Veh_2	466	15	438
ND_Veh_3	438	43	438
ND_Veh_4	472	9	438
HFD_Veh_1	393	88	438
HFD_Veh_2	358	123	438
HFD_Veh_3	369	112	438
HFD_Veh_4	388	93	438
HFD_Tx_1	451	30	438
HFD_Tx_2	395	86	438
HFD_Tx_3	456	25	438
HFD_Tx_4	457	24	438

[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

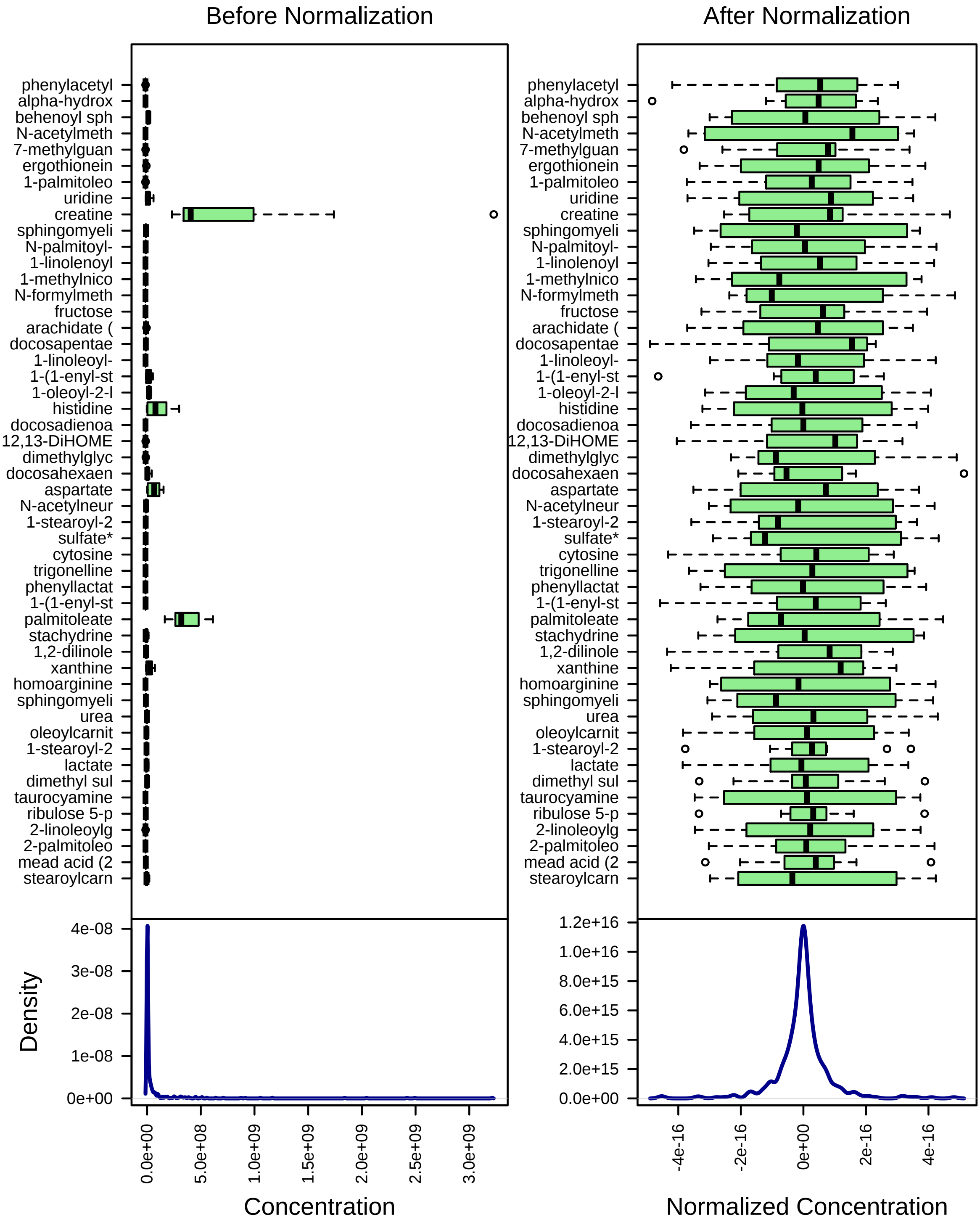

Figure 1: Box plots and kernel density plots before and after normalization. The boxplots show at most 50 features due to space limit. The density plots are based on all samples. Selected methods : Row-wise normalization: Normalization to sample median; Data transformation: Log Normalization; Data scaling: Range 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 One-way ANOVA

Univariate analysis methods are the most common methods used for exploratory data analysis. For multi-group analysis, MetaboAnalyst provides one-way Analysis of Variance (ANOVA). As ANOVA only tells whether the overall comparison is significant or not, it is usually followed by post-hoc analyses in order to identify which two levels are different. MetaboAnalyst provides two most commonly used methods for this purpose - Fisher's least significant difference method (Fisher's LSD) and Tukey's Honestly Significant Difference (Tukey's HSD). The univariate analyses provide a preliminary overview about features that are potentially significant in discriminating the conditions under study.

Figure 2 shows the important features identified by ANOVA analysis. Table 2 shows the details of these features. The `post-hoc Sig. Comparison` column shows the comparisons between different levels that are significant given the p value threshold.

Figure 2: Important features selected by ANOVA plot with p value threshold 0.05.

Table 2: Top 50 features identified by One-way ANOVA and post-hoc analysis

	Compounds	p.value	-log10(p)	FDR	Fisher's LSD
1	4-guanidinobutanoate	9.3386e-06	5.0297	0.0040903	0 - 1; 0 - 2
2	1-stearoyl-2-arachidonoyl-GPI (18:0/20:4)	2.9056e-05	4.5368	0.0048423	1 - 0; 2 - 0; 1 - 2
3	homostachydrine*	3.3166e-05	4.4793	0.0048423	0 - 1; 0 - 2; 1 - 2
4	N-delta-acetylornithine	4.5143e-05	4.3454	0.0049432	0 - 1; 0 - 2
5	sphingomyelin (d18:1/18:1, d18:2/18:0)	9.1013e-05	4.0409	0.0073321	1 - 0; 2 - 0; 1 - 2
6	homoarginine	1.0044e-04	3.9981	0.0073321	1 - 0; 2 - 0; 1 - 2
7	dehydroascorbate	2.1636e-04	3.6648	0.0120580	0 - 1; 2 - 1
8	1-palmitoleoyl-3-oleoyl-glycerol (16:1/18:1)*	2.4198e-04	3.6162	0.0120580	1 - 0; 2 - 0; 1 - 2
9	decanoylcarnitine	2.7583e-04	3.5594	0.0120580	1 - 0; 1 - 2
10	ergothioneine	2.9871e-04	3.5247	0.0120580	0 - 2; 1 - 2
11	3-hydroxybutyrylcarnitine (1)	3.1800e-04	3.4976	0.0120580	1 - 0; 1 - 2
12	cytidine	3.5279e-04	3.4525	0.0120580	0 - 1; 2 - 1
13	alpha-tocopherol	3.5788e-04	3.4463	0.0120580	1 - 0; 2 - 0
14	sphingomyelin (d18:1/20:1, d18:2/20:0)*	5.5401e-04	3.2565	0.0169860	1 - 0; 2 - 0; 1 - 2
15	pyridoxal	6.0236e-04	3.2201	0.0169860	0 - 1; 2 - 1
16	xanthine	6.5676e-04	3.1826	0.0169860	0 - 1; 2 - 1
17	sphingomyelin (d18:1/22:1, d18:2/22:0, d16:1/24:1)*	6.5929e-04	3.1809	0.0169860	1 - 0; 2 - 0; 1 - 2
18	3-phosphoglycerate	7.4280e-04	3.1291	0.0173140	0 - 1; 2 - 1
19	gamma-carboxyglutamate	7.5787e-04	3.1204	0.0173140	0 - 1; 2 - 1
20	stachydrine	7.9060e-04	3.1020	0.0173140	0 - 1; 0 - 2
21	dihydoxyphenylalanine (L-DOPA)	9.3512e-04	3.0291	0.0179240	1 - 0; 2 - 0; 1 - 2
22	2-hydroxybutyrate/2-hydroxyisobutyrate	9.4926e-04	3.0226	0.0179240	1 - 0; 2 - 0
23	sebacate (decanedioate)	1.0080e-03	2.9965	0.0179240	1 - 0; 1 - 2
24	phosphoenolpyruvate (PEP)	1.0697e-03	2.9707	0.0179240	0 - 1; 2 - 1
25	2-methylcitrate/homocitrate	1.0743e-03	2.9689	0.0179240	1 - 0; 1 - 2
26	1-(1-enyl-palmitoyl)-2-palmitoyl-GPC (P-16:0/16:0)*	1.1034e-03	2.9573	0.0179240	0 - 1; 2 - 1
27	1-stearoyl-2-arachidonoyl-GPE (18:0/20:4)	1.2087e-03	2.9177	0.0179240	1 - 0; 2 - 0; 1 - 2
28	hypoxanthine	1.2142e-03	2.9157	0.0179240	0 - 1; 2 - 1
29	uridine	1.2529e-03	2.9021	0.0179240	0 - 1; 2 - 1
30	imidazole lactate	1.2674e-03	2.8971	0.0179240	1 - 0; 1 - 2
31	heme	1.2696e-03	2.8963	0.0179240	1 - 0; 1 - 2
32	stearoyl sphingomyelin (d18:1/18:0)	1.3095e-03	2.8829	0.0179240	1 - 0; 2 - 0
33	isocitrate	1.3692e-03	2.8635	0.0181740	1 - 0; 1 - 2
34	1-palmitoleoyl-2-oleoyl-glycerol (16:1/18:1)*	1.4275e-03	2.8454	0.0183900	1 - 0; 2 - 0; 1 - 2
35	1-stearoyl-2-arachidonoyl-GPS (18:0/20:4)	1.4843e-03	2.8285	0.0185410	1 - 0; 2 - 0
36	retinol (Vitamin A)	1.5812e-03	2.8010	0.0185410	1 - 0; 1 - 2
37	gamma-aminobutyrate (GABA)	1.6015e-03	2.7955	0.0185410	0 - 1; 2 - 1
38	uridine 5'-monophosphate (UMP)	1.6589e-03	2.7802	0.0185410	1 - 0; 1 - 2
39	16-hydroxypalmitate	1.6816e-03	2.7743	0.0185410	1 - 0; 1 - 2
40	myristoleate (14:1n5)	1.7282e-03	2.7624	0.0185410	1 - 0; 2 - 0; 1 - 2
41	1-palmitoyl-2-oleoyl-GPG (16:0/18:1)	1.7705e-03	2.7519	0.0185410	0 - 1; 2 - 1
42	hexadecanedioate	1.8121e-03	2.7418	0.0185410	1 - 0; 2 - 0; 1 - 2
43	N-acetylglycine	1.8388e-03	2.7355	0.0185410	1 - 0; 1 - 2
44	adenosine 5'-monophosphate (AMP)	1.9192e-03	2.7169	0.0185410	1 - 0; 1 - 2
45	1-(1-enyl-palmitoyl)-2-linoleoyl-GPC (P-16:0/18:2)*	1.9243e-03	2.7157	0.0185410	0 - 1; 2 - 1
46	arachidonoyl ethanolamide	1.9881e-03	2.7016	0.0185410	1 - 0; 1 - 2
47	3-hydroxyisobutyrate	2.0161e-03	2.6955	0.0185410	1 - 0; 2 - 0
48	O-sulfo-L-tyrosine	2.0319e-03	2.6921	0.0185410	1 - 0; 1 - 2
49	tetradecanedioate	2.1171e-03	2.6743	0.0189250	1 - 0; 2 - 0; 1 - 2
50	myristate (14:0)	2.2057e-03	2.6565	0.0193220	1 - 0; 1 - 2

2.2 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 3 is pairwise score plots providing an overview of the various seperation patterns among the most significant PCs; Figure 4 is the scree plot showing the variances explained by the selected PCs; Figure 5 shows the 2-D scores plot between selected PCs; Figure 6 shows the 3-D scores plot between selected PCs; Figure 7 shows the loadings plot between the selected PCs; Figure 8 shows the biplot between the selected PCs.

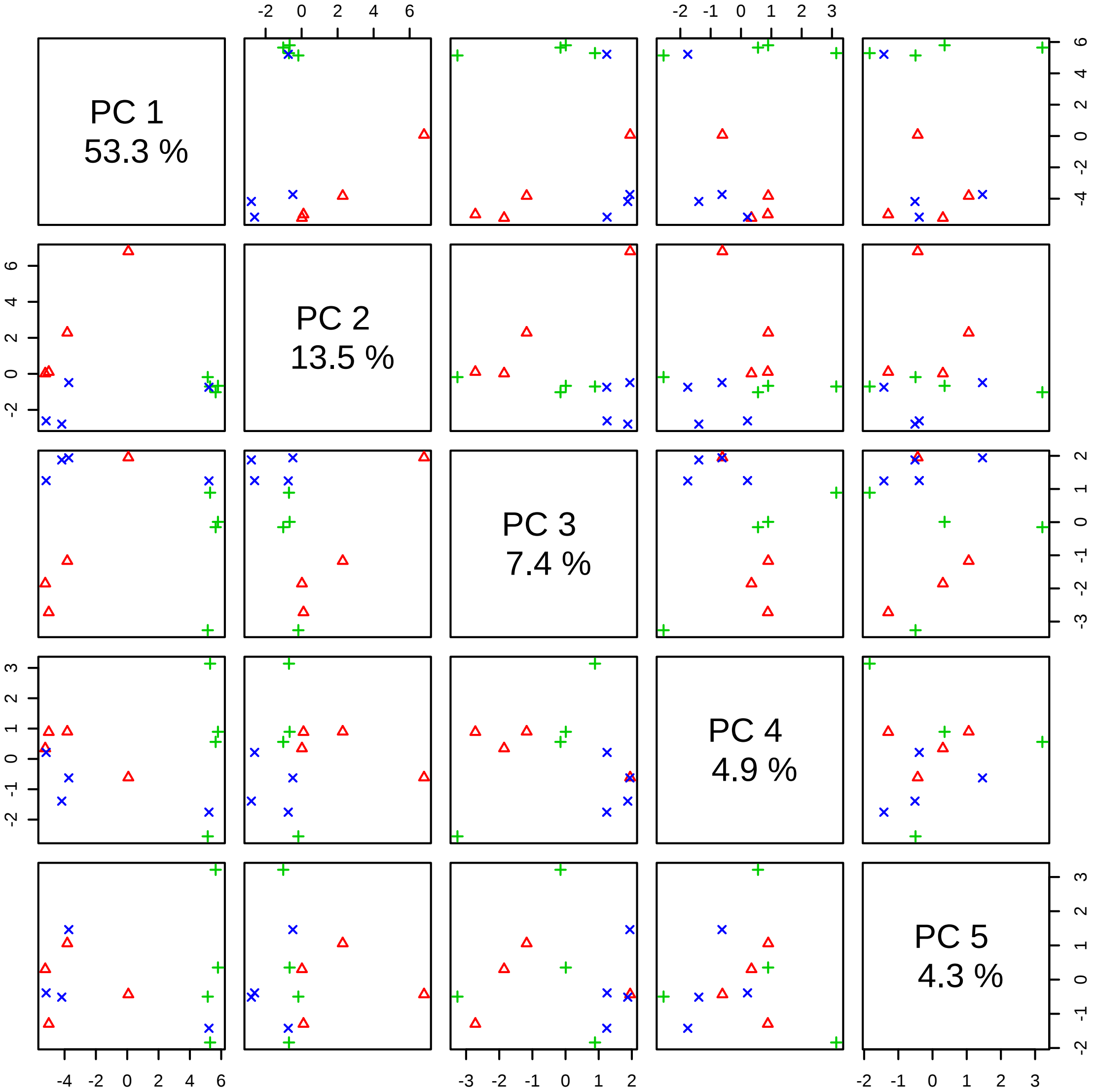

Figure 3: Pairwise score plots between the selected PCs. The explained variance of each PC is shown in the corresponding diagonal cell.

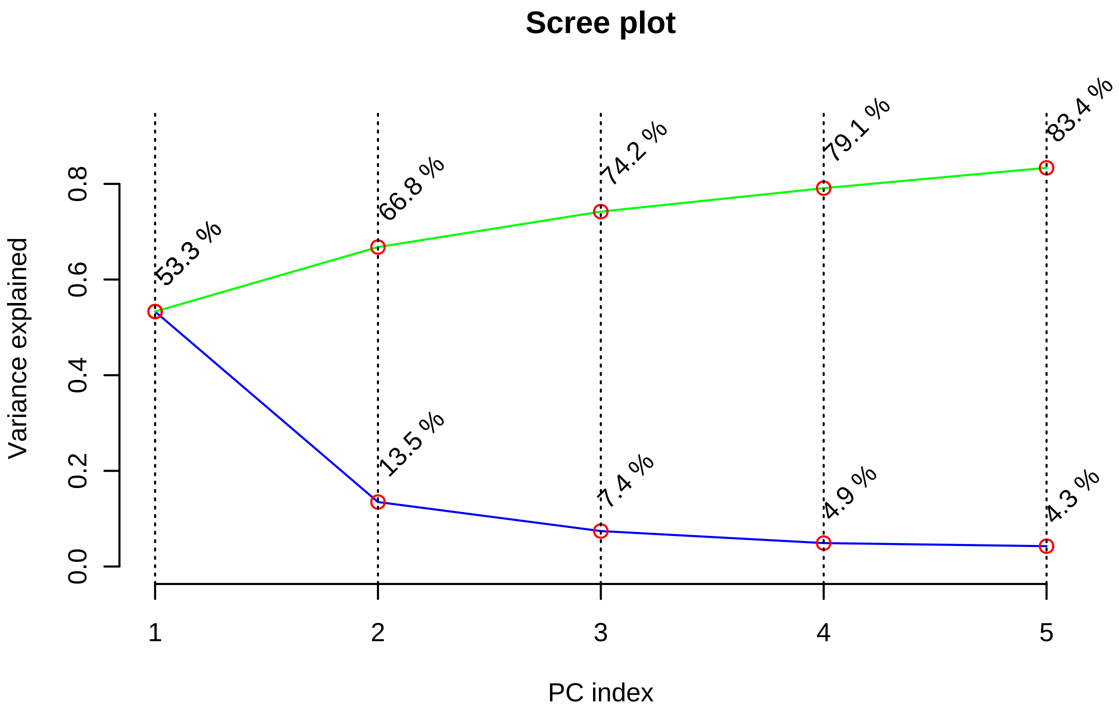

Figure 4: Scree plot shows the variance explained by PCs. The green line on top shows the accumulated variance explained; the blue line underneath shows the variance explained by individual PC.

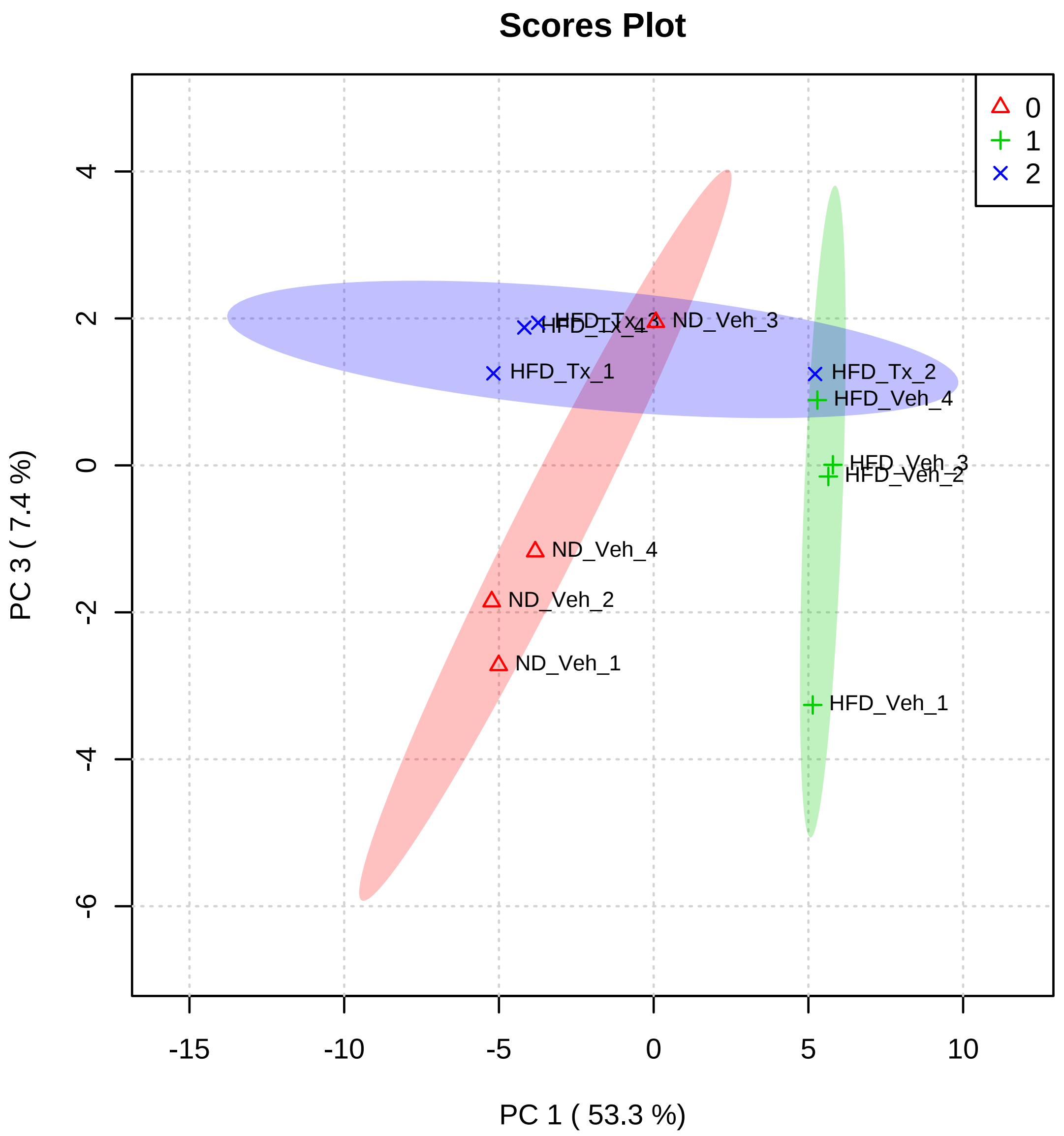

Figure 5: Scores plot between the selected PCs. The explained variances are shown in brackets.

Figure 6: 3D score plot between the selected PCs. The explained variances are shown in brackets.

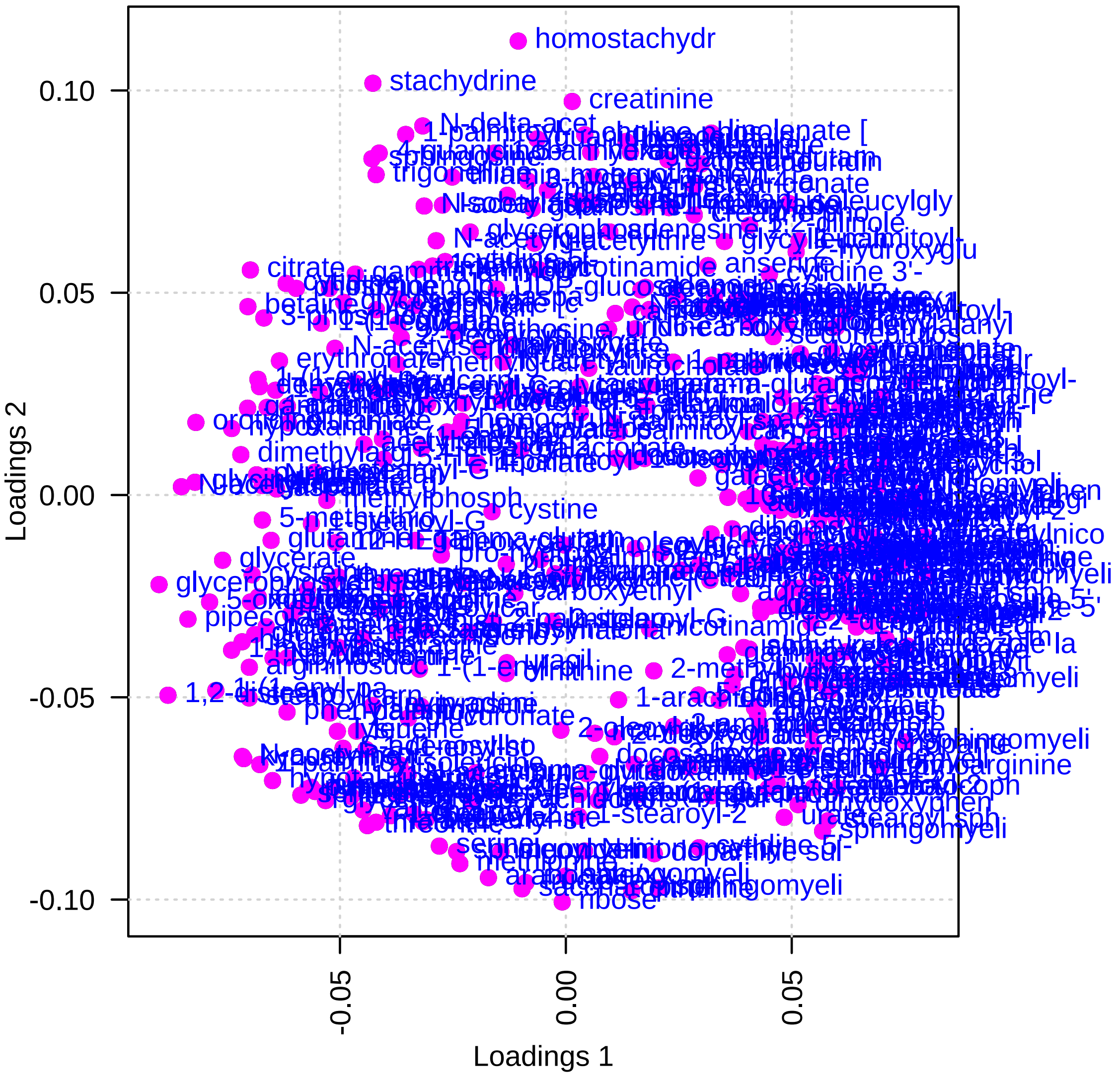

Figure 7: Loadings plot for the selected PCs.

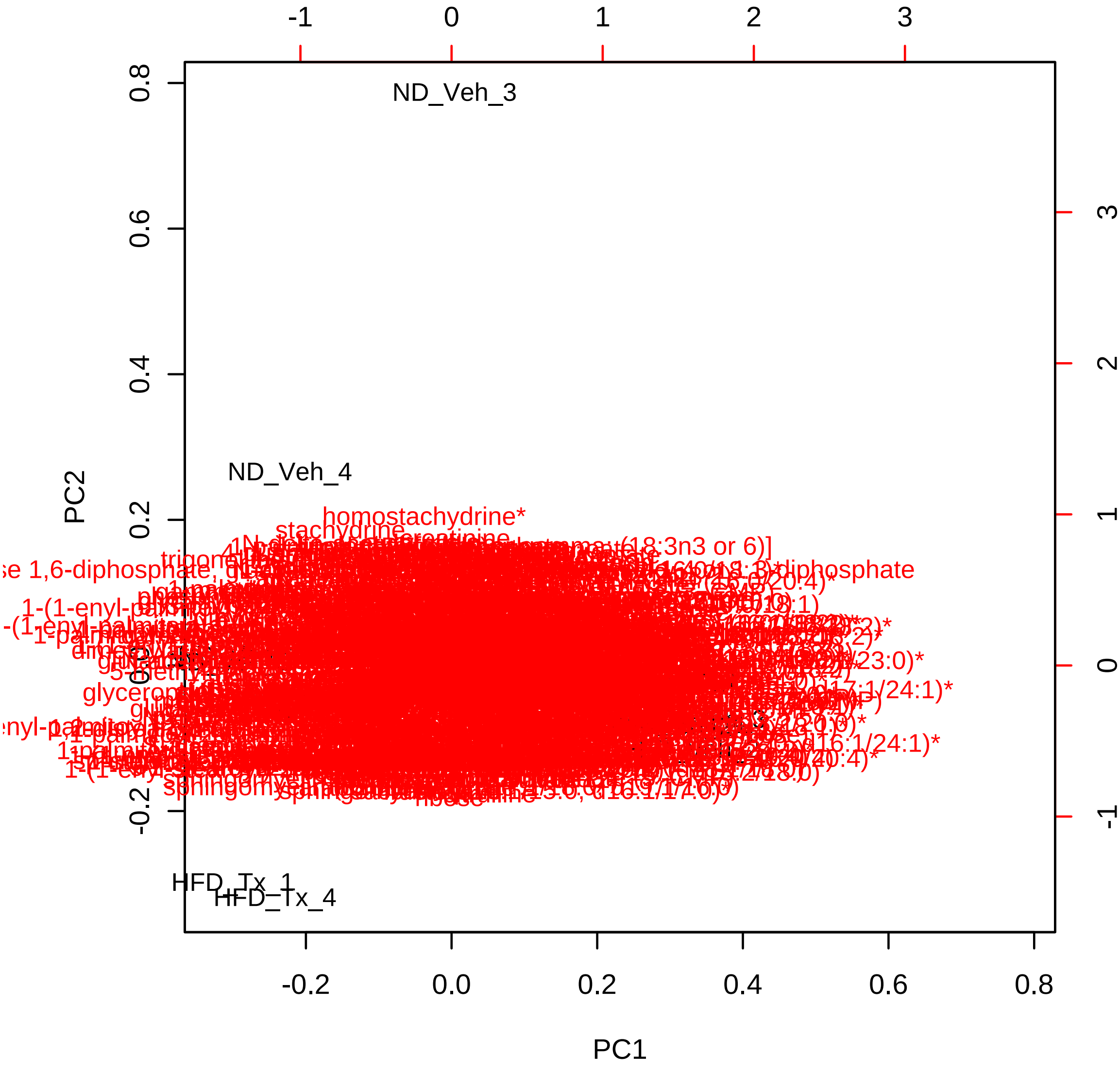

Figure 8: PCA biplot between the selected PCs. Note, you may want to test different centering and scaling normalization methods for the biplot to be displayed properly.

2.3 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 9 shows the overview of scores plots; Figure 10 shows the 2-D scores plot between selected components; Figure 11 shows the 3-D scores plot between selected components; Figure 12 shows the loading plot between the selected components;Figure 13 shows the classification performance with different number of components; Figure 14 shows the results of permutation test for model validation; Figure 15 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

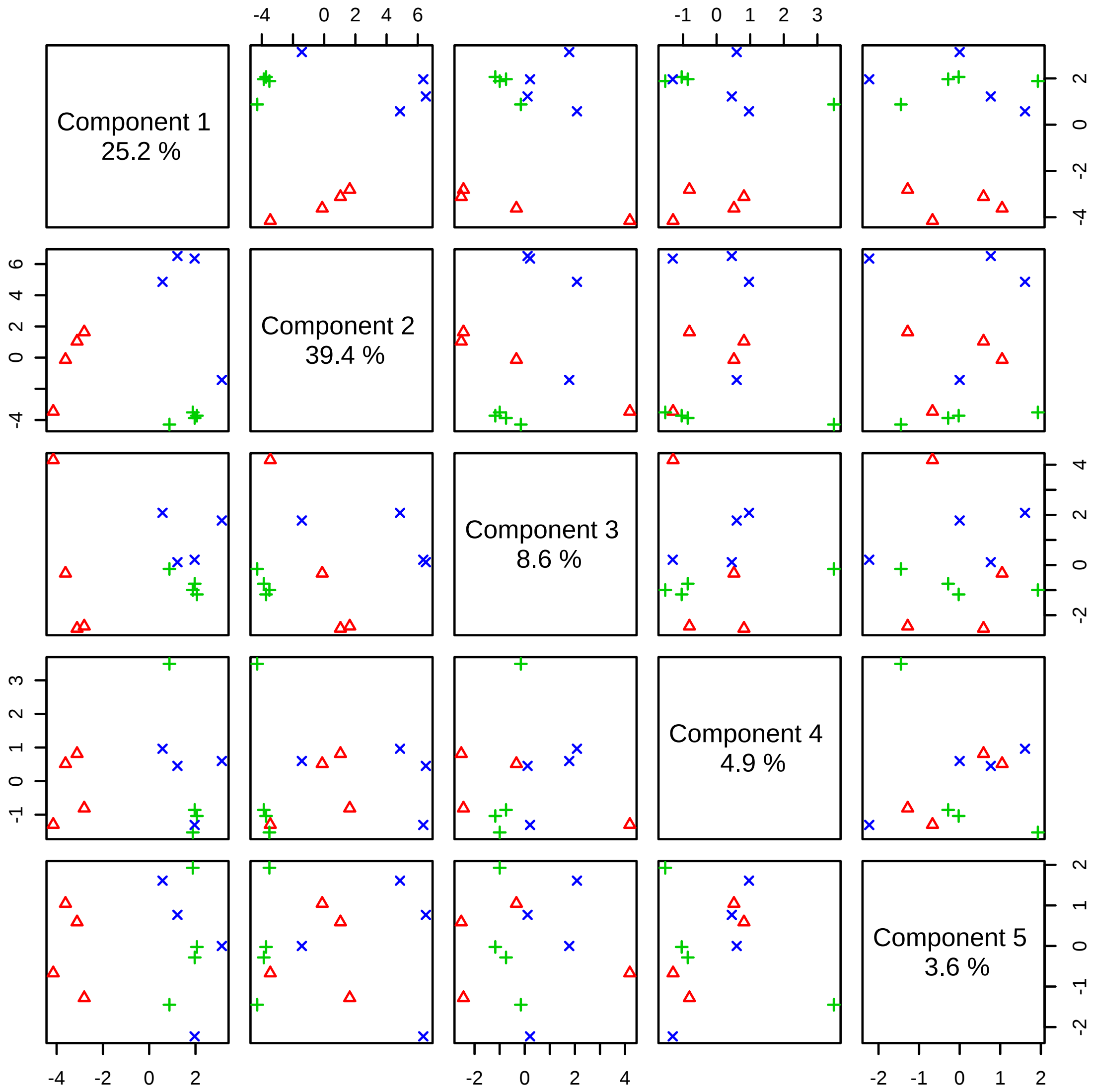

Figure 9: Pairwise scores plots between the selected components. The explained variance of each component is shown in the corresponding diagonal cell.

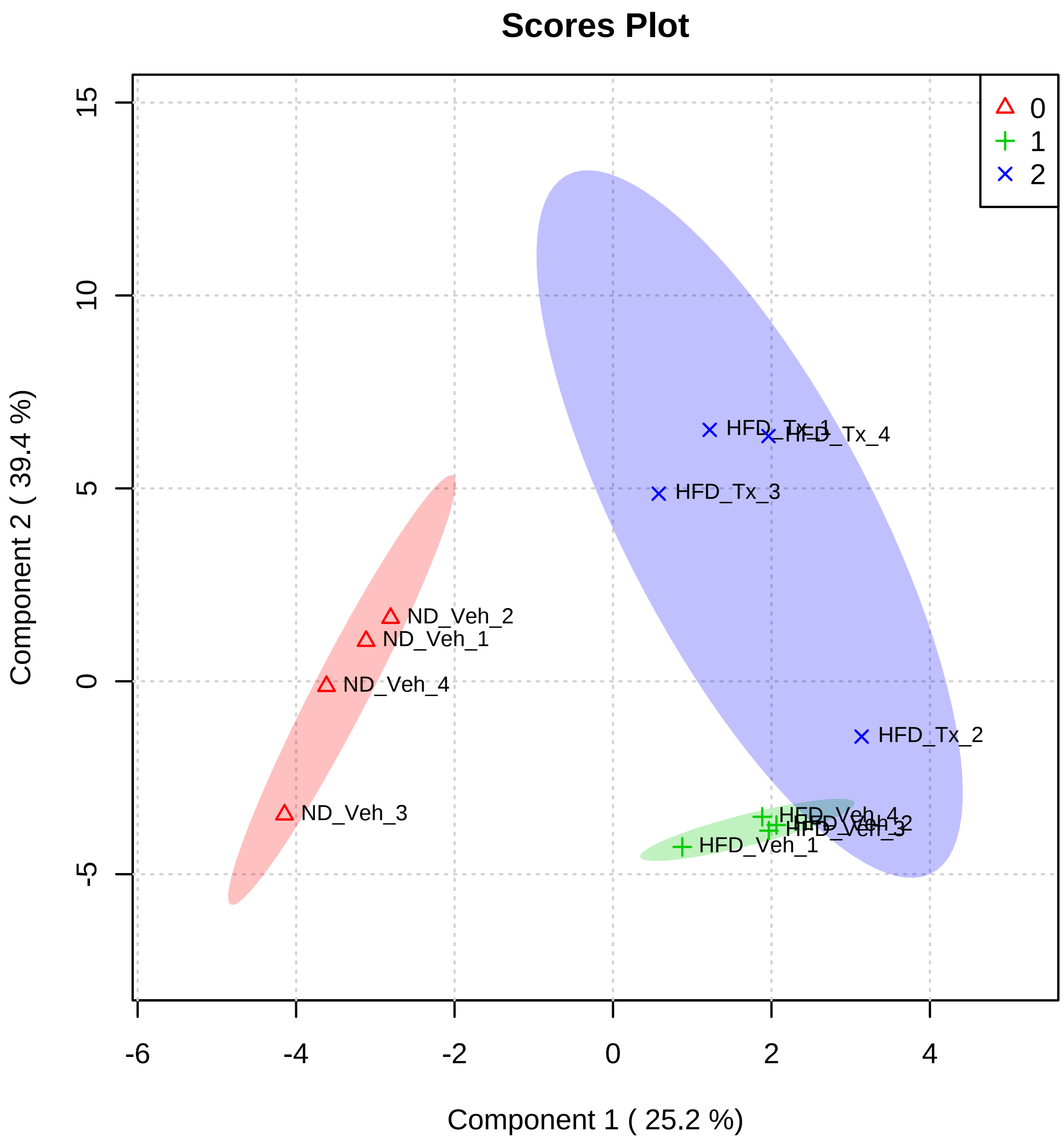

Figure 10: Scores plot between the selected PCs. The explained variances are shown in brackets.

Figure 11: 3D scores plot between the selected PCs. The explained variances are shown in brackets.

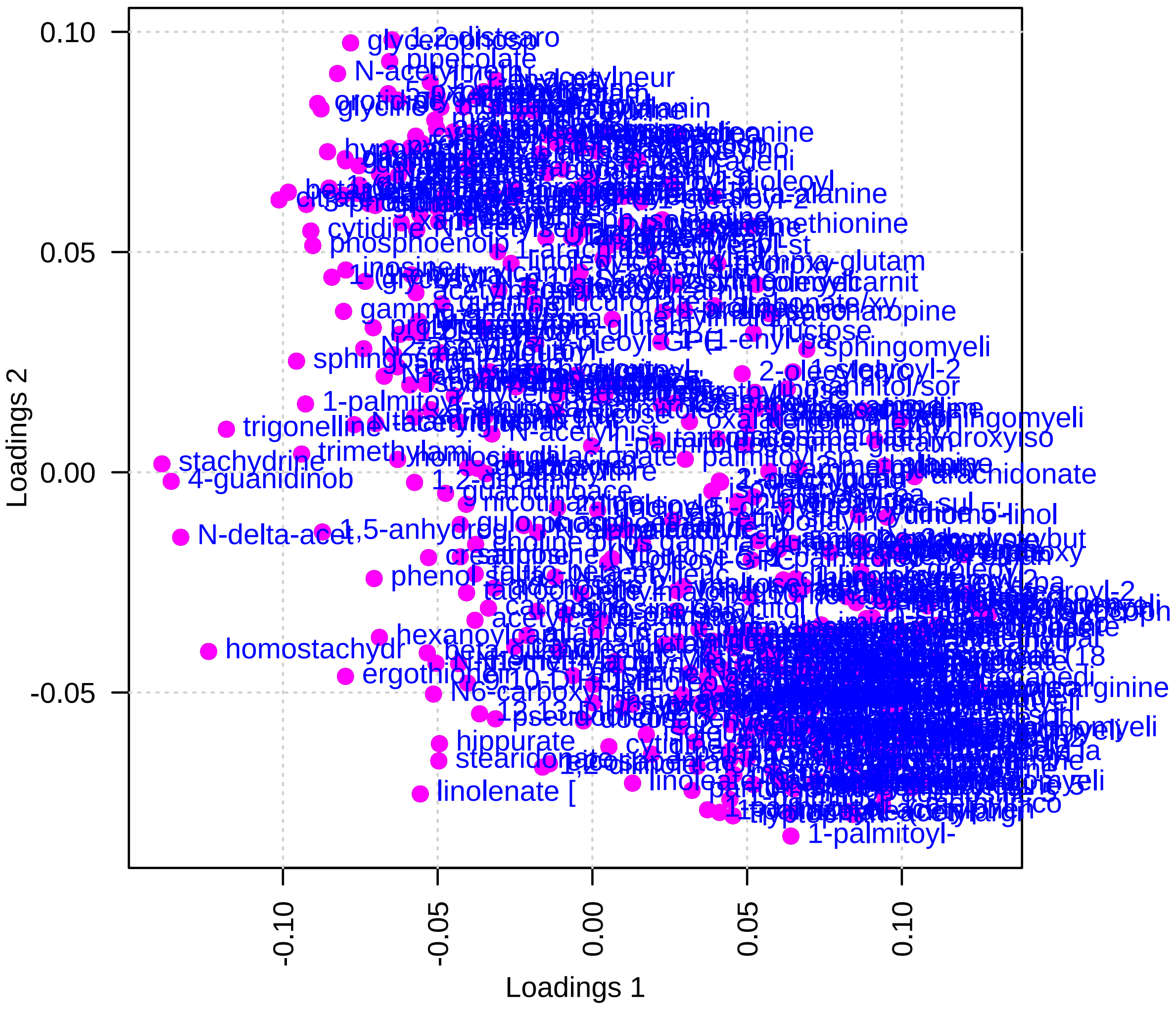

Figure 12: Loadings plot between the selected PCs.

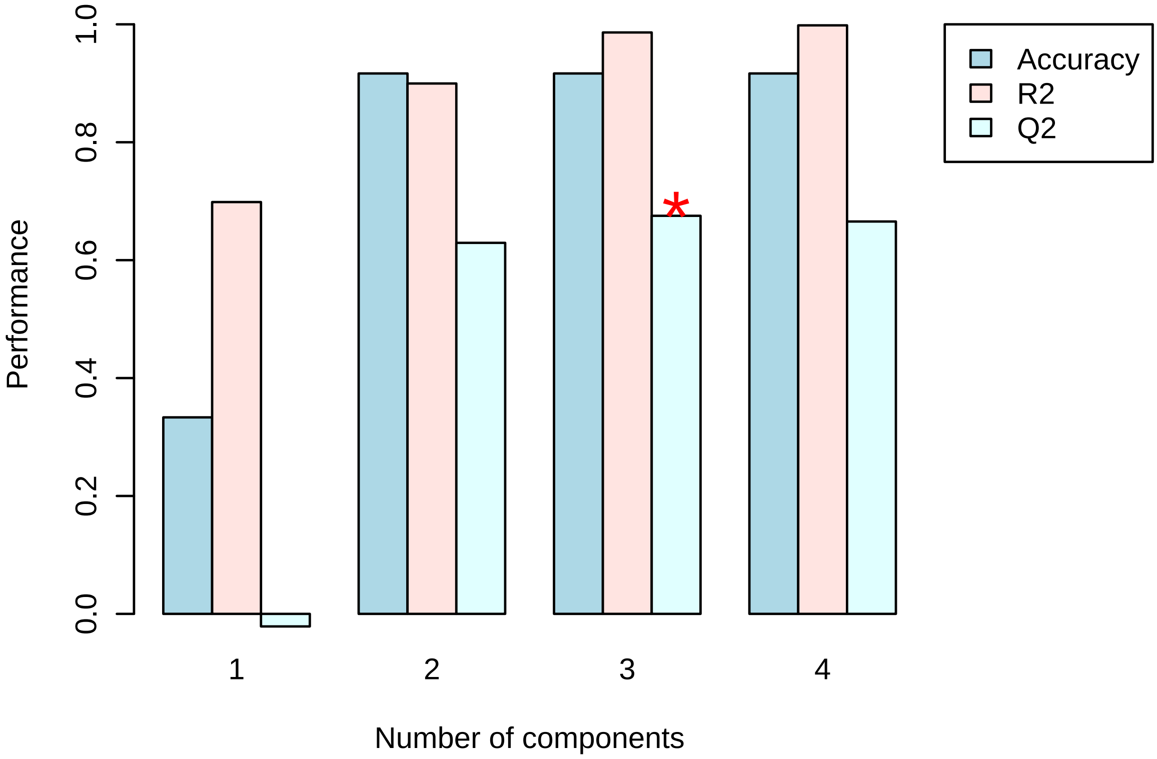

Figure 13: PLS-DA classification using different number of components. The red circle indicates the best classifier.

Figure 14: PLS-DA model validation by permutation tests based on . The p value based on permutation is .

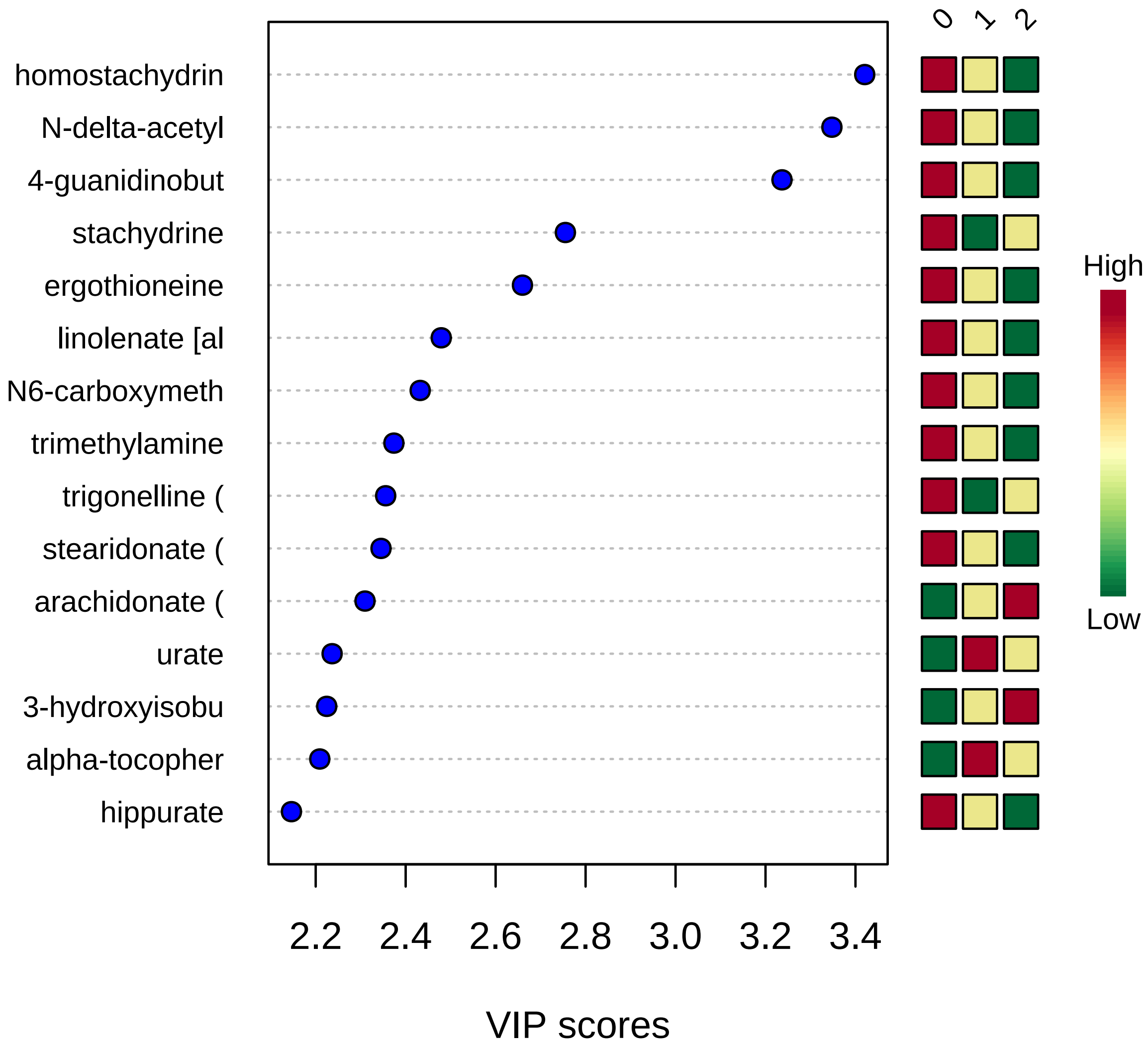

Figure 15: Important features identified by PLS-DA. The colored boxes on the right indicate the relative concentrations of the corresponding metabolite in each group under study.

2.4 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 16 shows the significant features identified by SAM. Table 3 shows the details of these features.

Table 3: Top 50 features identified by SAM

	Compounds	d.value	stdev	rawp	q.value
1	4-guanidinobutanoate	4.8259	0.014954	0	0
2	N-delta-acetylornithine	4.4205	0.02084	4.5662e-05	0.00062498
3	homostachydrine*	4.2147	0.018119	6.8493e-05	0.00062498
4	homoarginine	4.1566	0.024471	6.8493e-05	0.00062498
5	sphingomyelin (d18:1/22:1, d18:2/22:0, d16:1/24:1)	3.8225	0.040334	0.00015982	0.0007222
6	alpha-tocopherol	3.6703	0.031263	0.00015982	0.0007222
7	dehydroascorbate	3.5768	0.025885	0.00022831	0.0007222
8	N-acetylmethionine	3.5627	0.058716	0.00022831	0.0007222
9	sphingomyelin (d18:1/18:1, d18:2/18:0)	3.5482	0.019843	0.00025114	0.0007222
10	gamma-carboxyglutamate	3.545	0.038518	0.00025114	0.0007222
11	stachydrine	3.5081	0.038582	0.00025114	0.0007222
12	sphingomyelin (d18:1/20:1, d18:2/20:0)*	3.4933	0.03399	0.00025114	0.0007222
13	1-stearoyl-2-arachidonoyl-GPI (18:0/20:4)	3.4112	0.013841	0.00027397	0.0007222
14	imidazole lactate	3.3592	0.043255	0.0002968	0.0007222
15	heme	3.3591	0.043281	0.0002968	0.0007222
16	orotidine	3.349	0.058382	0.00031963	0.00072914
17	hypoxanthine	3.321	0.041963	0.0003653	0.00078429
18	adenosine 5'-monophosphate (AMP)	3.1952	0.047414	0.00045662	0.00086536
19	3-phosphoglycerate	3.16	0.0332	0.00054795	0.00086536
20	decanoylcarnitine	3.1311	0.023891	0.00059361	0.00086536
21	retinol (Vitamin A)	3.131	0.042909	0.00059361	0.00086536
22	cytidine	3.1276	0.025748	0.00059361	0.00086536
23	1-palmitoleoyl-3-oleoyl-glycerol (16:1/18:1)*	3.1249	0.022904	0.00061644	0.00086536
24	glycerophosphoinositol*	3.1233	0.086715	0.00061644	0.00086536
25	pyridoxal	3.1089	0.030353	0.00061644	0.00086536
26	guanosine 5'- monophosphate (5'-GMP)	3.0795	0.055195	0.00061644	0.00086536
27	arachidonoyl ethanolamide	3.0428	0.045103	0.00075342	0.0010185
28	trigonelline (N'-methylnicotinate)	2.999	0.050464	0.00091324	0.0011904
29	stearoyl sphingomyelin (d18:1/18:0)	2.9542	0.037225	0.0010274	0.0012239
30	uridine 5'-monophosphate (UMP)	2.9527	0.040544	0.0010274	0.0012239
31	sphingomyelin (d18:2/23:0, d18:1/23:1, d17:1/24:1)	2.9477	0.062931	0.0010502	0.0012239
32	1-(1-enyl-palmitoyl)-2-linoleoyl-GPC (P-16:0/18:2)	2.9416	0.042652	0.0010731	0.0012239
33	1-stearoyl-2-arachidonoyl-GPS (18:0/20:4)	2.9103	0.038226	0.0012785	0.001397
34	3-hydroxybutyrylcarnitine (1)	2.9087	0.022925	0.0013014	0.001397
35	betaine	2.8785	0.051321	0.001347	0.0014047
36	isocitrate	2.853	0.036225	0.0015068	0.0014693
37	phosphoenolpyruvate (PEP)	2.8481	0.033148	0.0015297	0.0014693
38	1-palmitoyl-2-oleoyl-GPG (16:0/18:1)	2.8454	0.03964	0.0015297	0.0014693
39	dihydoxyphenylalanine (L-DOPA)	2.8086	0.031126	0.0017352	0.0016239
40	citrate	2.8009	0.062811	0.0018037	0.0016458
41	xanthine	2.7781	0.027322	0.002032	0.0018027
42	sebacate (decanedioate)	2.7542	0.031188	0.0021461	0.0018027
43	uridine	2.7423	0.033444	0.0022146	0.0018027
44	urate	2.7368	0.045532	0.0022603	0.0018027
45	1-methylnicotinamide	2.7291	0.064109	0.0022603	0.0018027
46	N-acetylarginine	2.7137	0.060877	0.0023288	0.0018027
47	myristate (14:0)	2.7099	0.040466	0.0023516	0.0018027
48	1-palmitoleoyl-2-oleoyl-glycerol (16:1/18:1)*	2.7087	0.034493	0.0023744	0.0018027
49	5-oxoproline	2.6783	0.068532	0.0024201	0.0018027
50	dimethylarginine (SDMA + ADMA)	2.6757	0.047862	0.0025114	0.0018333

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

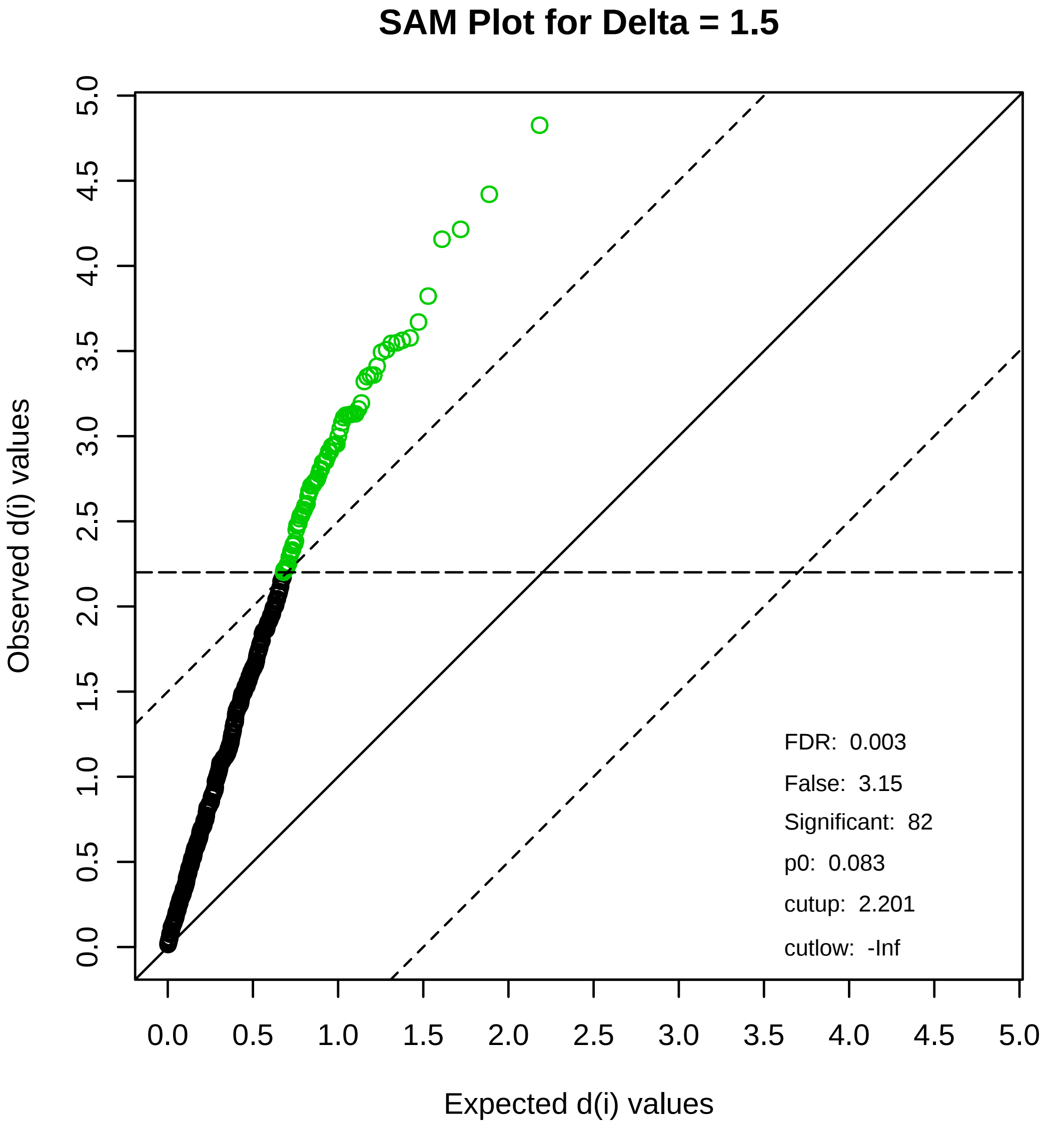

Figure 16: Significant features identified by SAM. The green circles represent features that exceed the specified threshold.

2.5 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 17 shows the clustering result in the form of a dendrogram. Figure 18 shows the clustering result in the form of a heatmap.

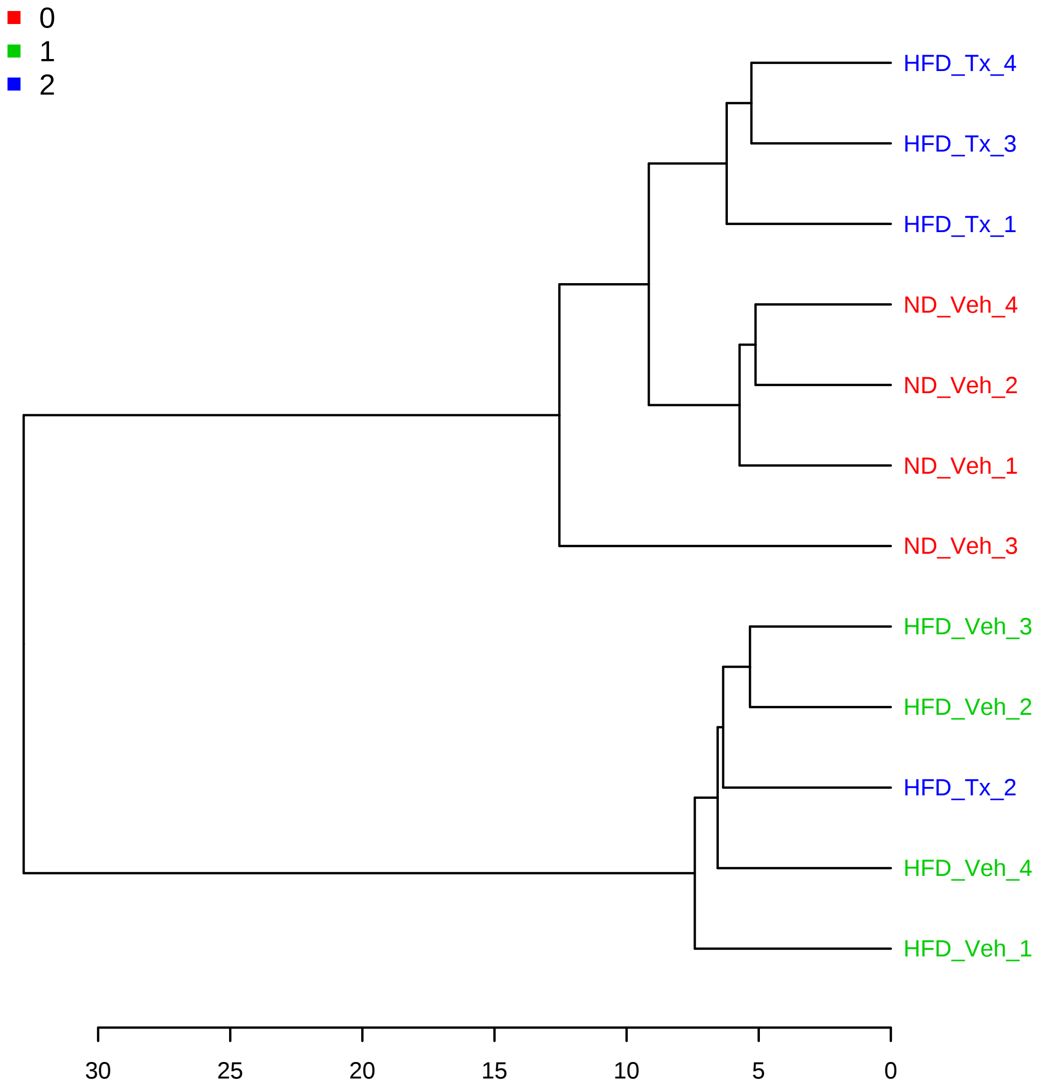

Figure 17: Clustering result shown as dendrogram (distance measure using `euclidean`, and clustering algorithm using `ward`).

Figure 18: Clustering result shown as heatmap (distance measure using `euclidean`, and clustering algorithm using `ward`).

2.6 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 19 shows clustering the results. Table 4 shows the members in each cluster from K-means analysis.

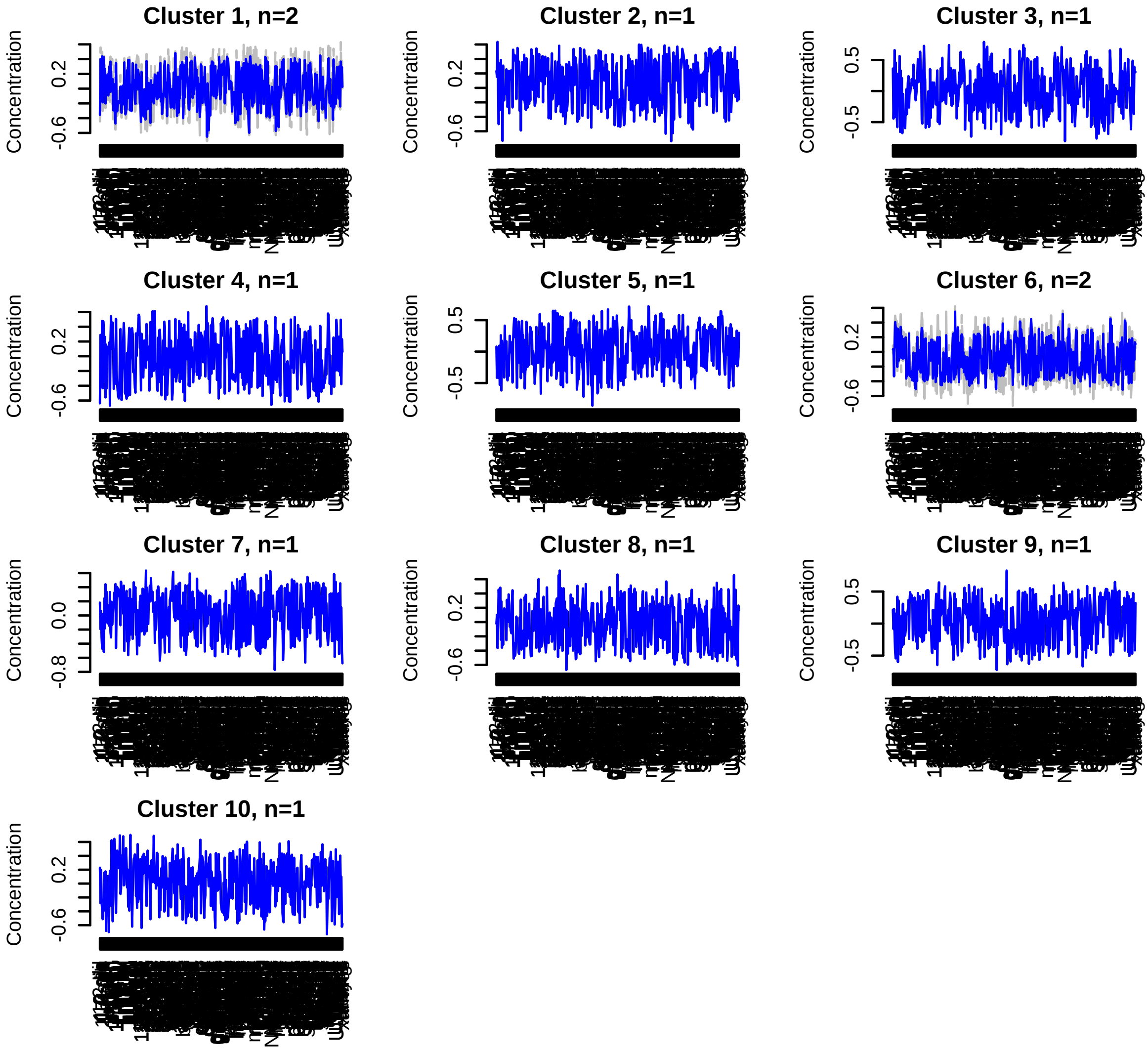

Figure 19: K-means cluster analysis. The x-axes are variable indices and y-axes are relative intensities. The blue lines represent median intensities of corresponding clusters

Table 4: Clustering result using K-means

	Samples in each cluster
Cluster(1)	HFD_Tx_3 HFD_Tx_4
Cluster(2)	HFD_Tx_2
Cluster(3)	ND_Veh_3
Cluster(4)	HFD_Tx_1
Cluster(5)	HFD_Veh_1
Cluster(6)	ND_Veh_2 ND_Veh_4
Cluster(7)	HFD_Veh_3
Cluster(8)	ND_Veh_1
Cluster(9)	HFD_Veh_2
Cluster(10)	HFD_Veh_4

2.7 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[8]. Figure 20 shows the SOM clustering results. Table 5 shows the members in each cluster from SOM analysis.

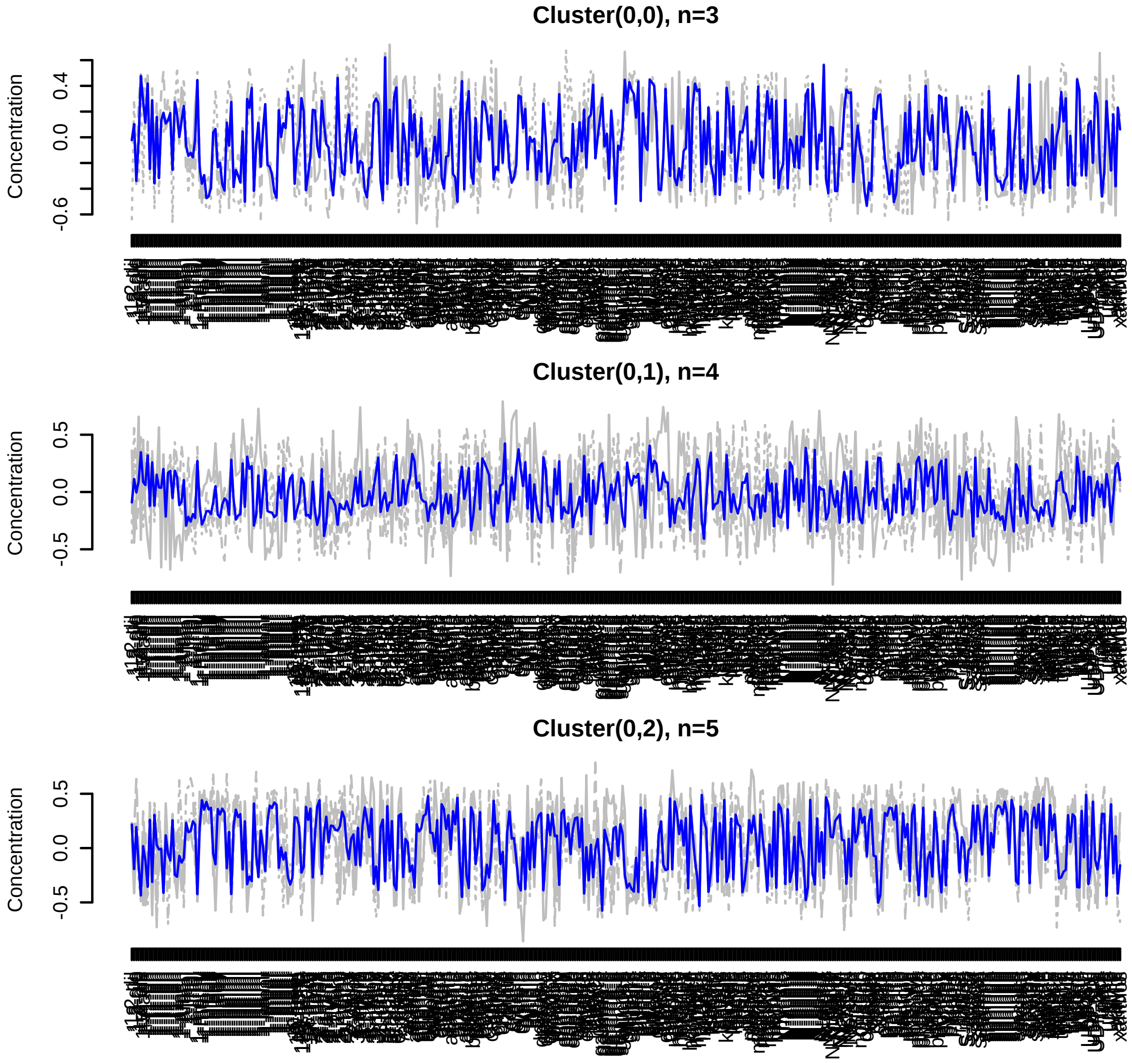

Figure 20: SOM cluster analysis. The x-axes are features and y-axes are relative intensities. The blue lines represent median intensities of corresponding clusters

Table 5: Clustering result using SOM

	Samples in each cluster
Cluster(0 , 0)	ND_Veh_1 ND_Veh_2 HFD_Tx_1
Cluster(0 , 1)	ND_Veh_3 ND_Veh_4 HFD_Tx_3 HFD_Tx_4
Cluster(0 , 2)	HFD_Veh_1 HFD_Veh_2 HFD_Veh_3 HFD_Veh_4 HFD_Tx_2

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

2.8 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[9]. Table 6 shows the confusion matrix of random forest. Figure 21 shows the cumulative error rates of random forest analysis for given parameters. Figure 22 shows the important features ranked by random forest. Figure 23 shows the outlier measures of all samples for the given parameters. The OOB error is 0.167

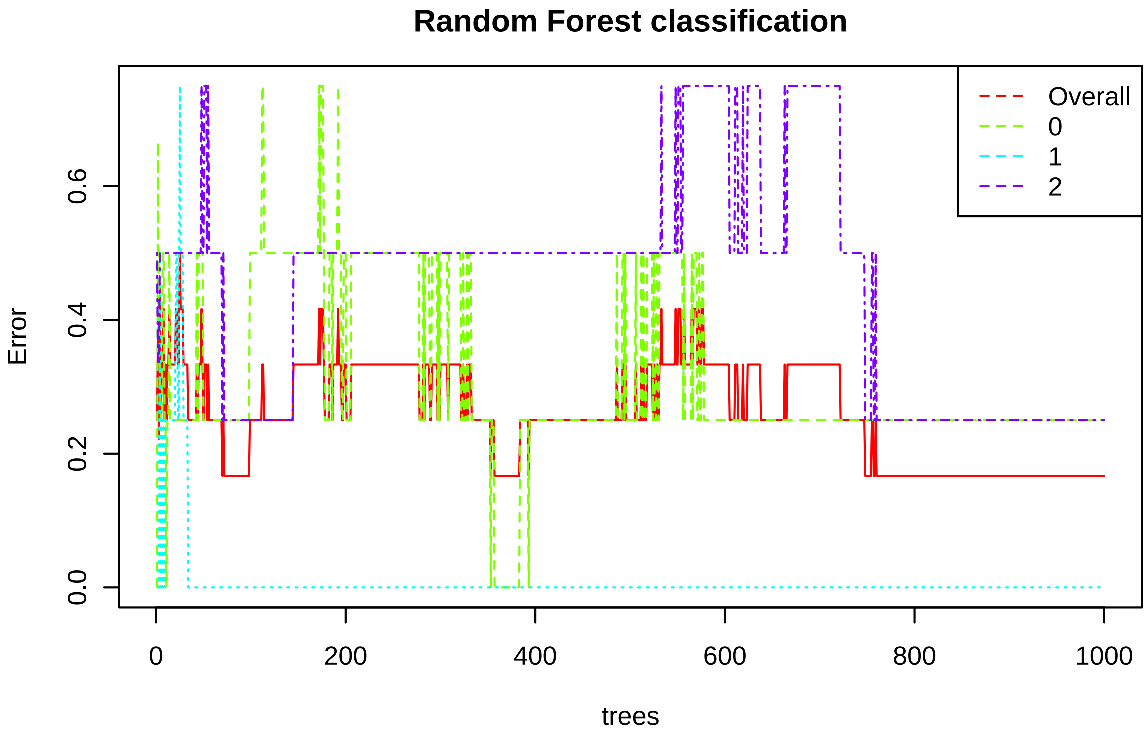

Figure 21: Cumulative error rates by Random Forest classification. The overall error rate is shown as the black line; the red and green lines represent the error rates for each class.

	0	1	2	class.error
0	3.00	0.00	1.00	0.25
1	0.00	4.00	0.00	0.00
2	0.00	1.00	3.00	0.25

Table 6: Random Forest Classification Performance

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

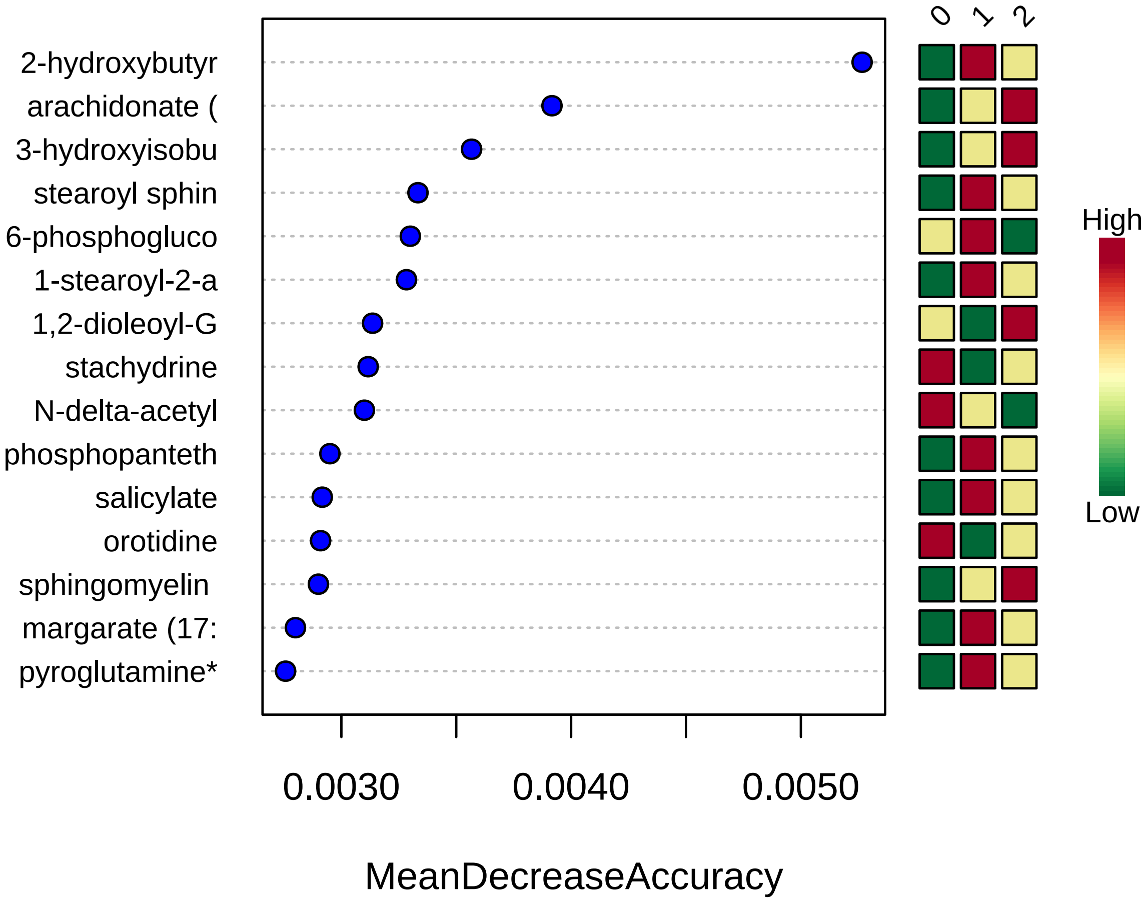

Figure 22: Significant features identified by Random Forest. The features are ranked by the mean decrease in classification accuracy when they are permuted.

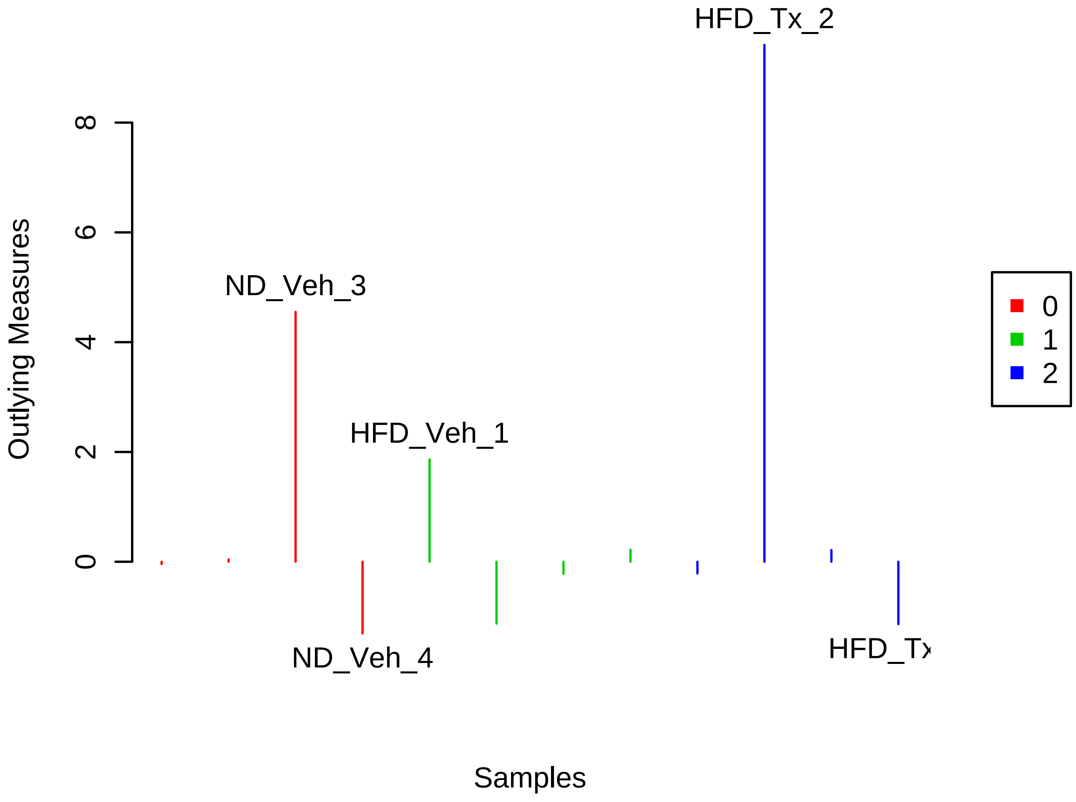

Figure 23: Potential outliers identified by Random Forest. Only the top five are labeled.

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 Aug 30 16:36:58 2015 with R version 3.0.3 (2014-03-06). Thank you for using MetaboAnalyst! For suggestions and feedback please contact Jeff Xia (*jeff.xia@mcgill.ca*).