A new pipeline to explore structural similarity across metabolite modules

Ebtesam Abdel-Shafy, Tadele Melak, David A. MacIntyre,
Giorgia Zadra, Luiz F. Zerbini, Silvano Piazza, and Stefano Cacciatore

2023-01-31

1 Introduction

MetChem is an R package used to perform structural and functional analysis of metabolites using a simple pipeline.

2 Installation

2.1 Installation via CRAN

The R package MetChem (current version 0.2) is part of the Comprehensive R Archive Network (CRAN)[1]. The simplest way to install the package is to enter the following command into your R session: `install.packages("MetChem")`. We suggest installing the following R packages: `pheatmap` and `RColorBrewer` to enable data visualization in heatmaps, `readxl` for the data reading of Excel files, and `impute` for the imputation of missing data.

```
# To install the pheatmap package
install.packages("pheatmap")

# To install the RColorBrewer package
install.packages("RColorBrewer")

# To install the readxl package
install.packages("readxl")

# To install the impute package
if (!require("BiocManager", quietly = TRUE))
    install.packages("BiocManager")
BiocManager::install("impute")
```

2.2 Manual installation from source

The package can be manually installed from source by opening the package's page in CRAN and then proceeding as follows:

- Download MetChem.tar.gz and save it to your hard disk
- Open a shell/terminal/command prompt window and change to the desired directory for installation of MetChem.tar.gz. Enter `R CMD INSTALL MetChem.tar.gz` to install the package. Note that this may require additional software on some platforms. Windows requires Rtools[2] to be installed and to be

[1] https://cran.r-project.org/
[2] https://developer.apple.com/xcode/

available in the default search path (environment variable PATH). MAC OS X requires installation of Xcode developers and command line tools.

2.3 Compatibility issues

The package downloadable from CRAN was built using R version, R.4.2.1. The package should work without major issues on R versions > 3.5.0 and KODAMA package >= 2.3.

3 Getting Started

The R package MetChem depends by the R package rcdk, which is partially implemented in Java. In Windows environment, we reported an issue related to the uploading of rcdk package. It can be solved running the following code before loading the package MetChem.

```
replacement <- function(category = "LC_ALL") {
  if (identical(category, "LC_MESSAGES"))
    return("")
  category <- match(category, .LC.categories)
  if (is.na(category))
    stop("invalid 'category' argument")
  .Internal(Sys.getlocale(category))

}
base <- asNamespace("base")
environment(replacement) <- base
unlockBinding("Sys.getlocale", base)
assign("Sys.getlocale", replacement, envir = base)
lockBinding("Sys.getlocale", base)
```

To load the package, enter the following command in your R session:

```
library("MetChem")
```

If this command terminates without any error messages, the package is installed successfully. The MetChem package is now ready for use.

The package includes both a user manual (this document) and a reference manual (help pages for each function). To view the user manual, enter `vignette("MetChem")`. Help pages can be viewed using the command `help(package="MetChem")`.

4 Example 1: murine prostate tissues metabolic profile

Here, we introduce an example for the analysis of metabolic structural information using MetChem package. For this, we used a data set of mass spectrometry dataset obtained from murine prostate tissue samples reported by Labbé and Zadra *et al.* (2019) (Supplementary Data 2). The metabolic data are obtained from ventral prostate tissues of mice that overexpress a human c-MYC transgene (MYC) in the prostate epithelium and wild-type littermates (WT). Mice were fed either a high fat diet (HFD; 60% kcal from fat; lard—rich in saturated fat) or a control diet (CTD; 10% kcal from fat). The data set includes six replicates for each group (*i.e.*, WT_CTD, MYC_CTD, WT_HFD, and MYC_HFD). To begin, download the data from the Labbé and Zadra (2019) study. Download it and save it to your hard disk. Metabolomic data is extracted using the instructions below. Data is then imputed using a k-nearest neighbour (kNN) algorithm using the function `impute` as described in the publication.

```
require("readxl")
require("impute")
d=as.data.frame(read_excel("41467_2019_12298_MOESM5_ESM.xlsx",skip = 3))
d=d[1:414,]
rownames(d)=d[,"Metabolite"]
met=d[,4:27]
label=rep(c("WT_CTD","MYC_CTD","WT_HFD","MYC_HFD"),each=6)
label_MYC=rep(c("WT","MYC","WT","MYC"),each=6)
colnames(met)=paste(label,1:6)
met=data.matrix(met)
met=impute.knn(met,k=5)$data
```

Heatmap visualization is generated using the function `pheatmap`. Metabolites are hierarchically clustered according to their relative concentration The hierarchical clustering is performed using the distance matrix based on the KODAMA dimensions. KODAMA is a learning algorithm for unsupervised feature extraction specifically designed for analyzing noisy and high-dimensional data sets (Cacciatore *et al.*, 2014), implemented in the R package `KODAMA` (Cacciatore *et al.*, 2017). Additional information can be found in the review of Zinga *et al.*, 2023.

```
require("pheatmap")
require("RColorBrewer")

my_colour1 = list(genotype=c(MYC="#000000ff",WT="#eeeeeeff"),
                  group=c(MYC_CTD="#373898ff",MYC_HFD="#c11630ff",
                          WT_CTD="#00a4cfff",WT_HFD="#e40a81ff"))

set.seed(1)
kk1=KODAMA.matrix(t(met))
col=KODAMA.visualization(kk1)
hcol=hclust(dist(col),method="ward.D")

kk2=KODAMA.matrix(scale(met))
row=KODAMA.visualization(kk2)
hrow=hclust(dist(row),method="ward.D")
my_sample_col <- data.frame(group = label,genotype=label_MYC)
row.names(my_sample_col) <- colnames(met)

pheatmap(met,
         cluster_cols = hcol,
         cluster_rows = hrow,
         labels_row = rep("",nrow(met)),
         annotation_col = my_sample_col,
         annotation_colors = my_colour1,
```

```
color = colorRampPalette(rev(brewer.pal(n = 11, name ="RdBu")))(100))
```

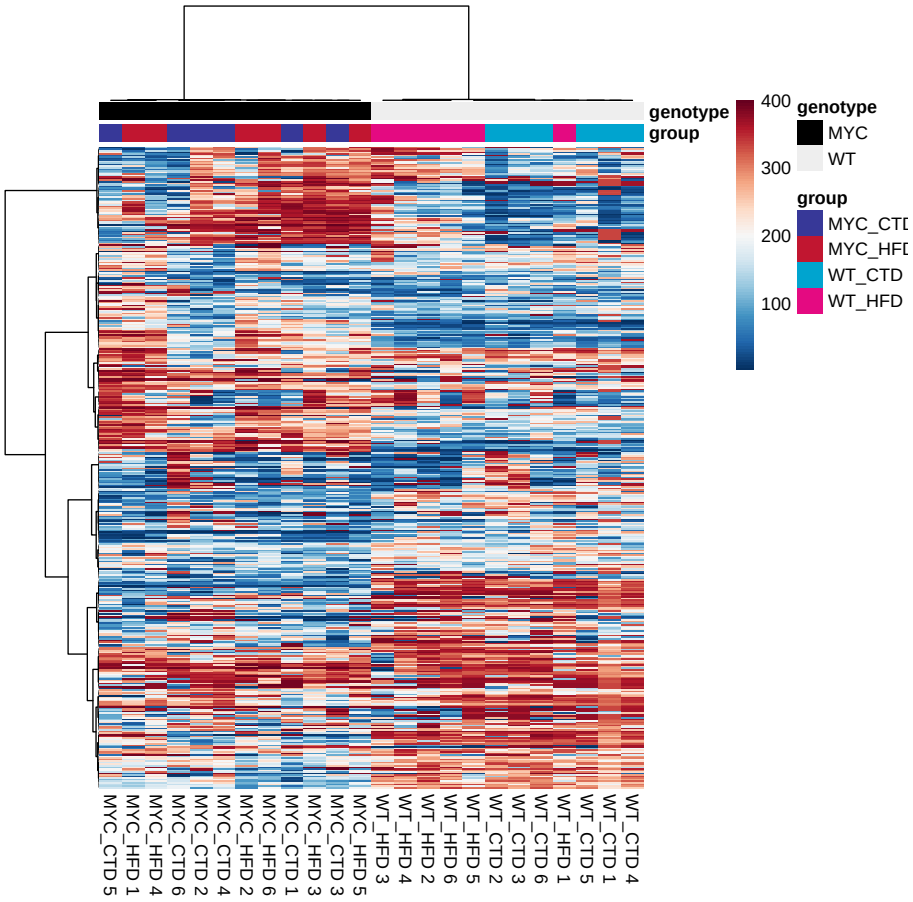

Figure 1: Heatmap of metabolites with hierarchical clustering based on their concentratrion.

To analyze the chemical similarities among metabolites, we need the Simplified Molecular-Input Line-Entry System (SMILES) of each metabolite is obtained. The SMILES of the previous data set is stored in the list `HFD` that can be loaded using the function `data(HFD)`. `MetChem` package includes the `modules.detection` function based on KODAMA analysis. This function repeats the follwing steps (10 times as default): i) transformation of the chemical structure dissimilarity matrix in a multidimensional space (with 50 dimensions as defaults) using multidimensional scaling; ii) KODAMA features extraction; iii) hierarchical clustering based on the KODAMA output; iv) Calculation of the silhoutte index from different number of clusters (from 2 to 30 as default). The average of the siloutte index is calculated for each cluster numbers to identity the optimal cluster number.

```
data(HFD)
met=met[rownames(HFD),]
clu=modules.detection(HFD$SMILES)
plot(clu$min_nc:clu$max_nc,clu$silhouette,type="l",
     ylab="Rousseeuw's Silhouette index",xlab="Number of clusters")
abline(v=5*(1:6),lty=2)

print(clu$main_cluster)
```

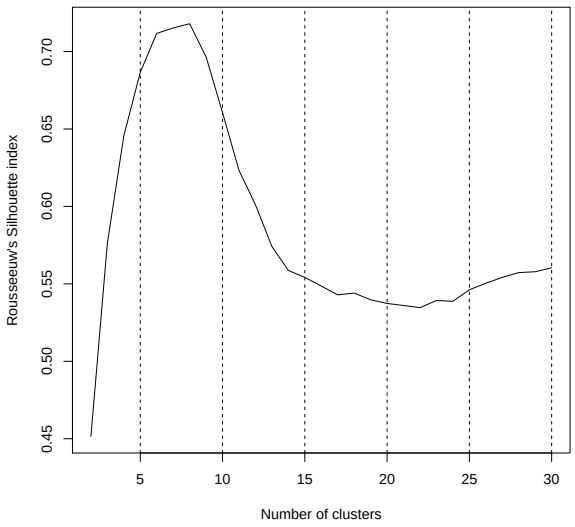

Figure 2: Silhouette index.

Based on the average Silhouette indeces, we have the optimal number of cluster for this data set is 8. Below is shown a graphical visualization of the final output of KODAMA. Each cluster is represented by a different color code. Each dot represents a different metabolite. Metabolites that are located near to each other share a similar chemical structure.

```
plot(clu$visualization,pch=21,bg=rainbow(8,alpha = 0.7)[clu$clusters[,"Clusters 8"]],cex=2)
legend(-30, 20, legend=paste("Cluster", unique(clu$clusters[,"Clusters 8"])),
   col= rainbow(8,alpha = 0.7), pch= 16, cex=1)
```

The following line code can be used to identify the points on the scatter plot. The cluster belonging and chemical name of the selected points will be displayed. ESC key to terminate the command.

```
data.frame(metabolite=rownames(met),
           cluster=clu$clusters[,"Clusters 8"])[identify(clu$visualization),]
```

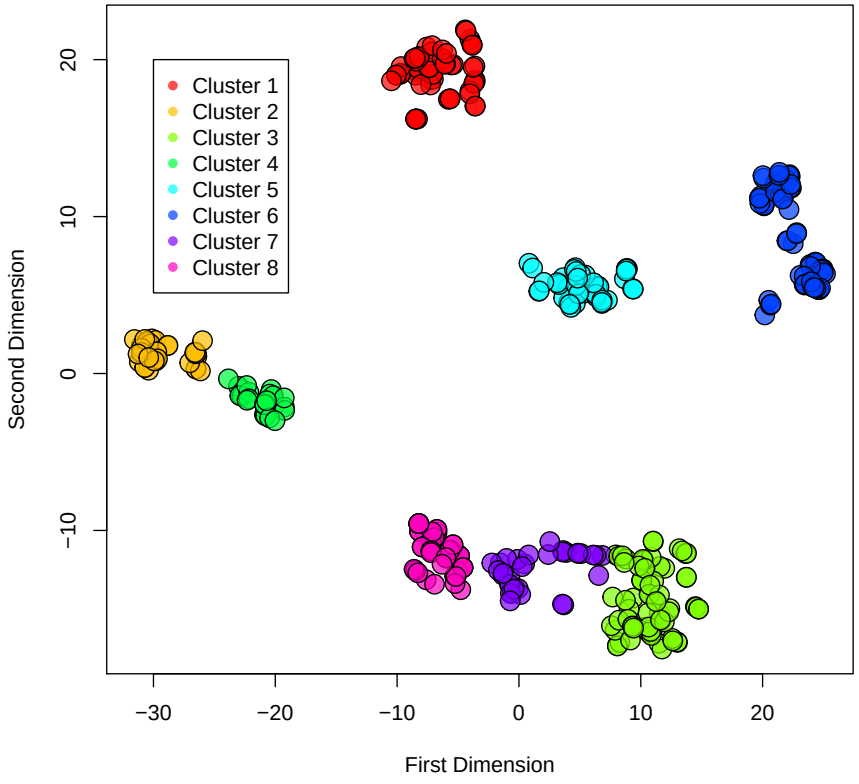

Figure 3: KODAMA plot.

A new heatmap is generated where metabolites are clustered according to their chemical similarity.

```
my_colour2 = list(cluster = c("1"=rainbow(8,alpha = 1)[1],
                              "2"=rainbow(8,alpha = 1)[2],
                              "3"=rainbow(8,alpha = 1)[3],
                              "4"=rainbow(8,alpha = 1)[4],
                              "5"=rainbow(8,alpha = 1)[5],
                              "6"=rainbow(8,alpha = 1)[6],
                              "7"=rainbow(8,alpha = 1)[7],
                              "8"=rainbow(8,alpha = 1)[8]),
                  genotype=c(MYC="#000000ff",WT="#eeeeeeff"),
                  group=c(MYC_CTD="#373898ff",MYC_HFD="#c11630ff",
                          WT_CTD="#00a4cfff",WT_HFD="#e40a81ff"))

clusters8=clu$clusters[,"Clusters 8"]
my_sample_row <- data.frame(cluster = as.character(clusters8))
row.names(my_sample_row) <- rownames(met)

set.seed(1)
met=met[rownames(HFD),]
kk1=KODAMA.matrix(t(met))
col=KODAMA.visualization(kk1)
hcol=hclust(dist(col),method="ward.D")
hrow=clu$hclust

my_sample_col <- data.frame(group = label,genotype=label_MYC)
row.names(my_sample_col) <- colnames(met)

pheatmap(met,
         cluster_cols = hcol,
         cluster_rows = hrow,
         labels_row = rep("",nrow(met)),
         annotation_colors = my_colour2,
         annotation_col = my_sample_col,
         annotation_row = my_sample_row,
         cutree_rows = 8,
         color = colorRampPalette(rev(brewer.pal(n = 11, name ="RdBu")))(100))
```

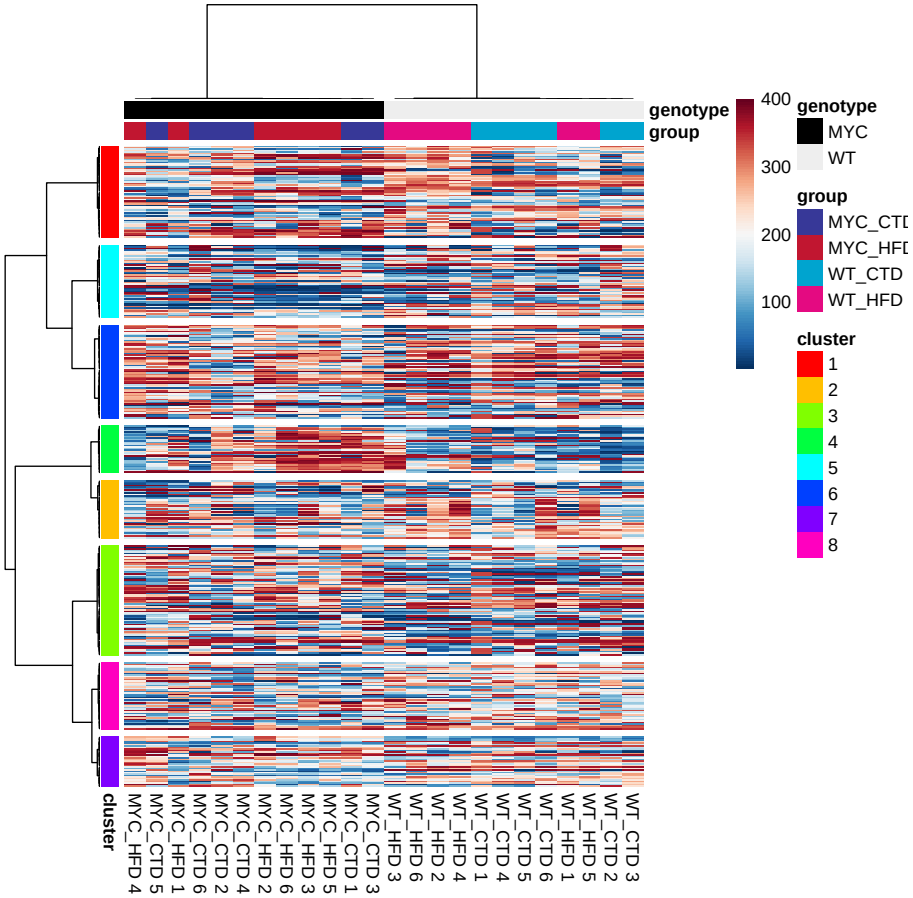

Figure 4: Heatmap of metabolites with vertical hierarchical clustering based on their molecular structure.

In the next step, we apply the Weighted Metabolite Chemical Similarity Analysis (WMCSA). WMCSA is implemented in the function `WMCSA`. This function summarizes the relative concentration of metabolites within each module (a.k.a., cluster). Each module is defined according to the chemical similarity.

```
set.seed(1)
my_sample_col <- data.frame(group = label,genotype=label_MYC)
row.names(my_sample_col) <- colnames(met)
ww=WMCSA(met,clu,8)

kk1=KODAMA.matrix(t(ww))
col=KODAMA.visualization(kk1)
hcol=hclust(dist(col),method="ward.D")

hrow=hclust(dist(ww),method="ward.D")

pheatmap(ww,
         cluster_cols = hcol,
         cluster_rows = hrow,
         annotation_col = my_sample_col,
         annotation_colors = my_colour1,
         color = colorRampPalette(rev(brewer.pal(n = 11, name ="RdBu")))(100))
```

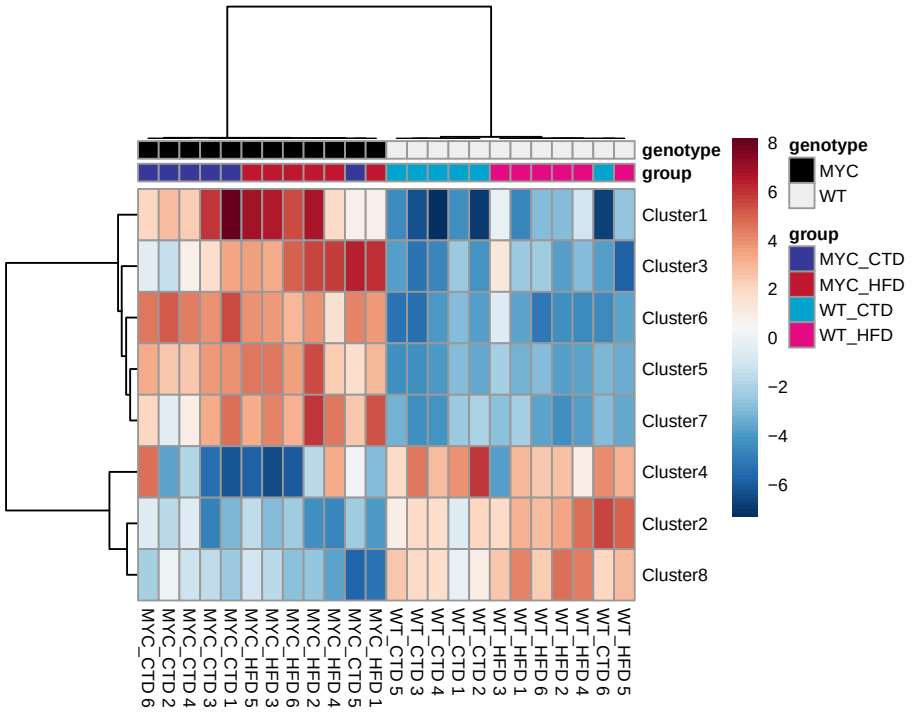

Figure 5: Heatmap of the output of WMCSA.

Differential analysis of the relevant modules can be performed using the function `multi_analysis` present in the R package `KODAMA`. In the example below, we perform a differential analysis between MYC transgenic mice fed with high-fat diet or control diet named as MYC_HFD and MYC_CTD, respectively.

```
multi_analysis(t(ww),label_MYC)
```

Feature	MYC	WT	p-value	FDR
Cluster1, median [IQR]	4.15 [2.003 6.629]	-4.379 [-6.376 -2.853]	3.66e-05	7.32e-05
Cluster2, median [IQR]	-2.57 [-4.056 -1.642]	2.423 [1.833 3.68]	6.01e-05	8.01e-05
Cluster3, median [IQR]	3.482 [1.47 5.657]	-3.779 [-4.305 -2.423]	7.66e-05	8.75e-05
Cluster4, median [IQR]	-3.325 [-5.853 -1.232]	2.934 [2.356 3.926]	6.1e-03	6.10e-03
Cluster5, median [IQR]	3.438 [2.472 4.082]	-3.376 [-3.839 -3.064]	3.66e-05	7.32e-05
Cluster6, median [IQR]	3.973 [3.794 4.45]	-4.192 [-4.657 -3.695]	3.66e-05	7.32e-05
Cluster7, median [IQR]	3.256 [2.382 4.548]	-3.406 [-3.94 -2.726]	3.66e-05	7.32e-05
Cluster8, median [IQR]	-2.279 [-2.99 -1.479]	2.413 [1.805 3.118]	4.69e-05	7.51e-05

We next build a heatmap of the metabolite belonging to the module 5.

```
sel=clu$clusters[,"Clusters 8"]==5
met.sel=met[sel,]
my_sample_col <- data.frame(group = label,genotype=label_MYC)
row.names(my_sample_col) <- colnames(met)
oo=order(row.names(my_sample_col))
my_sample_col=my_sample_col[oo,]
met.sel=met.sel[,oo]
hrow=hclust(dist(clu$visualization[sel,]),method="ward.D")

pheatmap(met.sel,fontsize = 7,
         cluster_cols = FALSE,
         cluster_rows = hrow,
         annotation_col = my_sample_col,
         annotation_colors = my_colour1,
         color = colorRampPalette(rev(brewer.pal(n = 11, name ="RdBu")))(100))
```

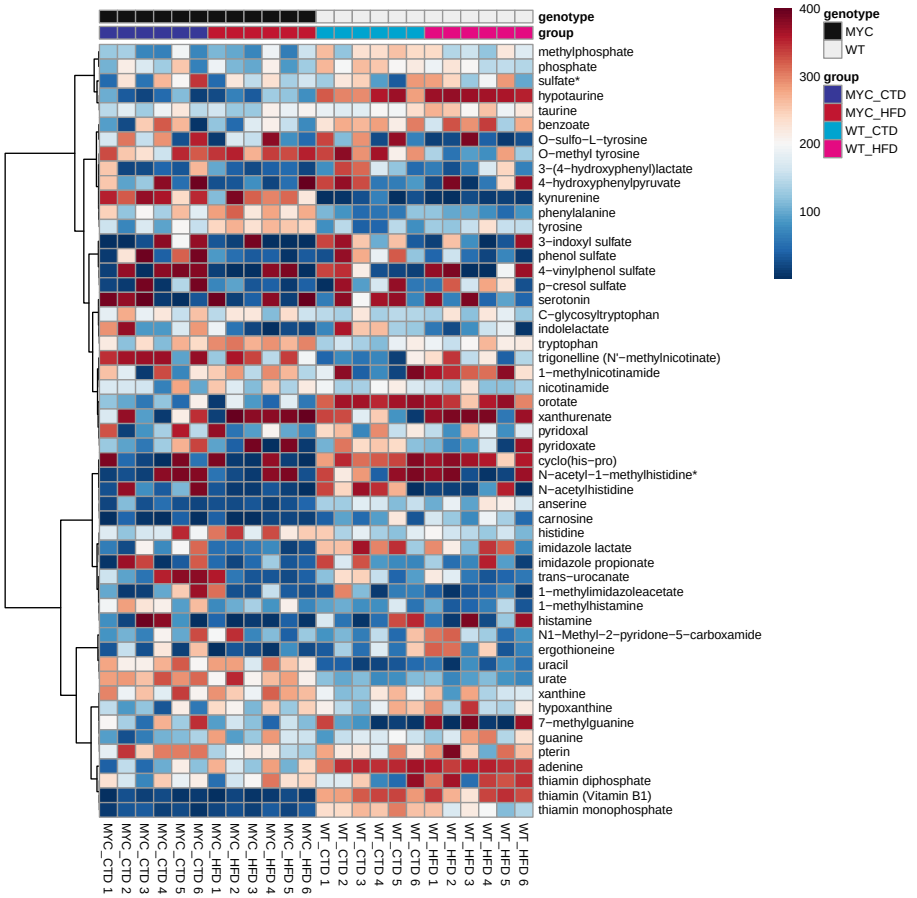

Figure 6: Heatmap

The function `readMet` connects R to the HMDB database [3]. to retrieve chemical and functional information of each metabolite. This can be summarized using different functions: `substituentsMet`, `diseaseMet`, `enzymeMet`, `pathwaysMet`, `taxonomyMet`. The function `features` associates the most prominent features to each module.

In this example, we characterized the modules by functional group.

```
doc=readMet(HFD$HMDB)
cla=substituentsMet(doc)
f=features(doc,cla,clu$clusters[,"Clusters 8"])
```

Fisher's test was used to rank the association of each module to the metabolite information. Below are reported the p-value for associations with the module 5.

```
f[[5]][f[[5]]<0.001]
```

[3] https://www.hmdb.ca

Substituents	p-value
Heteroaromatic compound	0.00e+00
Azacycle	0.00e+00
Aromatic heteromonocyclic compound	0.00e+00
Imidazolyl carboxylic acid derivative	3.00e-07
Aralkylamine	2.40e-06
Benzenoid	9.80e-06
Aromatic homomonocyclic compound	2.20e-05
1-hydroxy-2-unsubstituted benzenoid	6.85e-05
Azole	1.05e-04
Indole	2.26e-04
Pyrrole	2.26e-04
Substituted pyrrole	2.26e-04

4 Example 2: ChemRICH example data file

In the example, we analyzed the list of metabolites downloadable from [4]. HMDB IDs were retrieved from PubChem Identifier Exchange Service [5] and manually curated. The SMILES and metabolite's names are stored in the list `ChemRICH` that can be loaded using the function `data(ChemRICH)`. The `modules.detection` function was applied to the SMILES.

```
data(ChemRICH)
set.seed(1)
clu2=modules.detection(ChemRICH$SMILES)
plot(clu2$min_nc:clu2$max_nc,clu2$silhouette,type="l",
ylab="Rousseeuw's Silhouette index",xlab="Number of clusters")
abline(v=5*(1:6),lty=2)
```

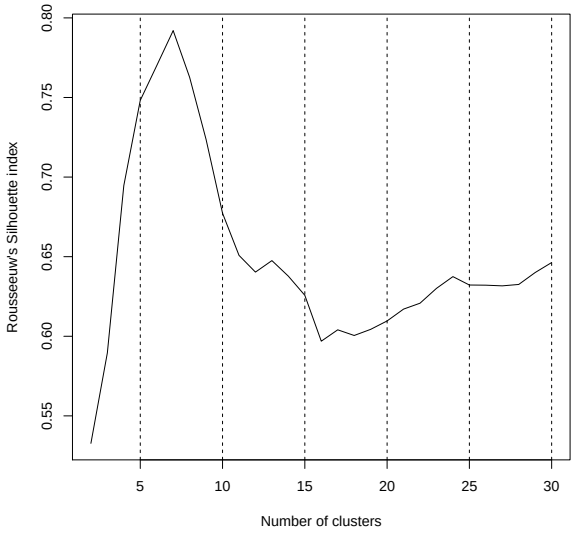

Figure 7: Silhouette index.

Based on the average Silhouette indeces, we have the optimal number of cluster for this data set is 7. Below is shown a graphical visualization of the final output of KODAMA. Each cluster is represented by a different

[4] https://chemrich.fiehnlab.ucdavis.edu/
[5] https://pubchem.ncbi.nlm.nih.gov/idexchange/idexchange.cgi

color code. Each dot represents a different metabolite. Metabolites that are located near to each other share a similar chemical structure.

```
plot(clu2$visualization,pch=21,bg=rainbow(7,alpha = 0.7)[clu2$clusters[,"Clusters 7"]],cex=2)
legend(-22, 30, legend=paste("Cluster", unique(clu2$clusters[,"Clusters 7"])),
   col= rainbow(7,alpha = 0.7), pch= 16, cex=1)
```

The following line code can be used to identify the points on the scatter plot. The cluster belonging and chemical name of the selected points will be displayed. ESC key to terminate the command.

```
data.frame(metabolite=ChemRICH$name,
           cluster=clu2$clusters[,"Clusters 7"])[identify(clu2$visualization),]
```

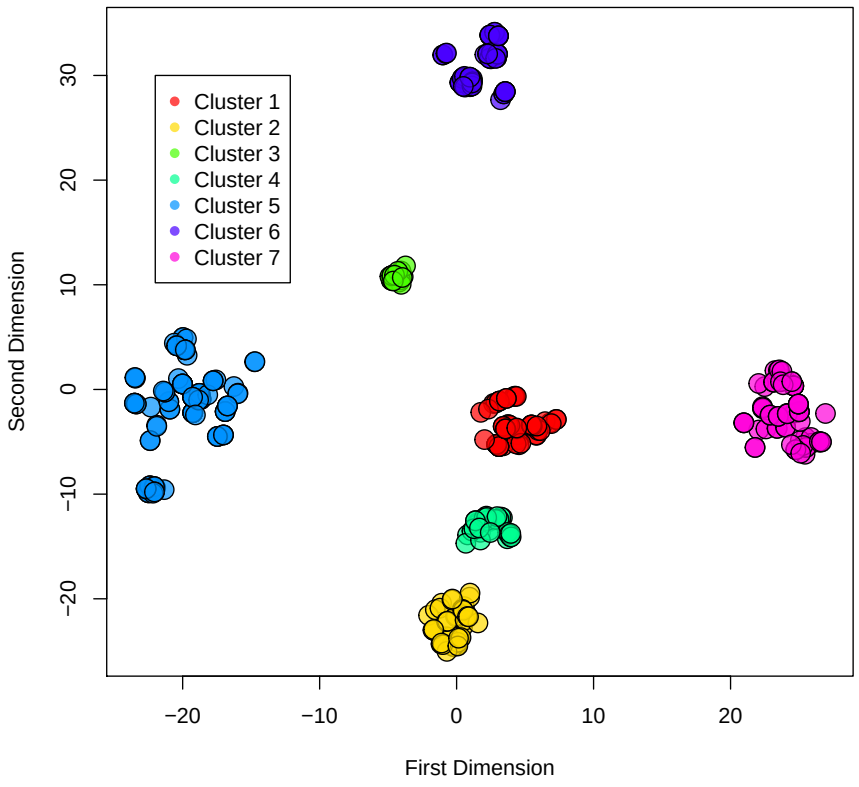

Figure 8: KODAMA plot.

A higher number of clusters (for example 8) can be choosen and it can be visualized with the following code. In this example, the cluster 7 was splitted in two clusters. Now, Cluster 8 and Cluster 7 represents the group of phospolipids and Cluster 8 refers exclusively to the sphingomyelins.

```
plot(clu2$visualization,pch=21,bg=rainbow(8,alpha = 0.7)[clu2$clusters[,"Clusters 8"]],cex=2)
legend(-22, 30, legend=paste("Cluster", unique(clu2$clusters[,"Clusters 8"])),
   col= rainbow(8,alpha = 0.7), pch= 16, cex=1)
```

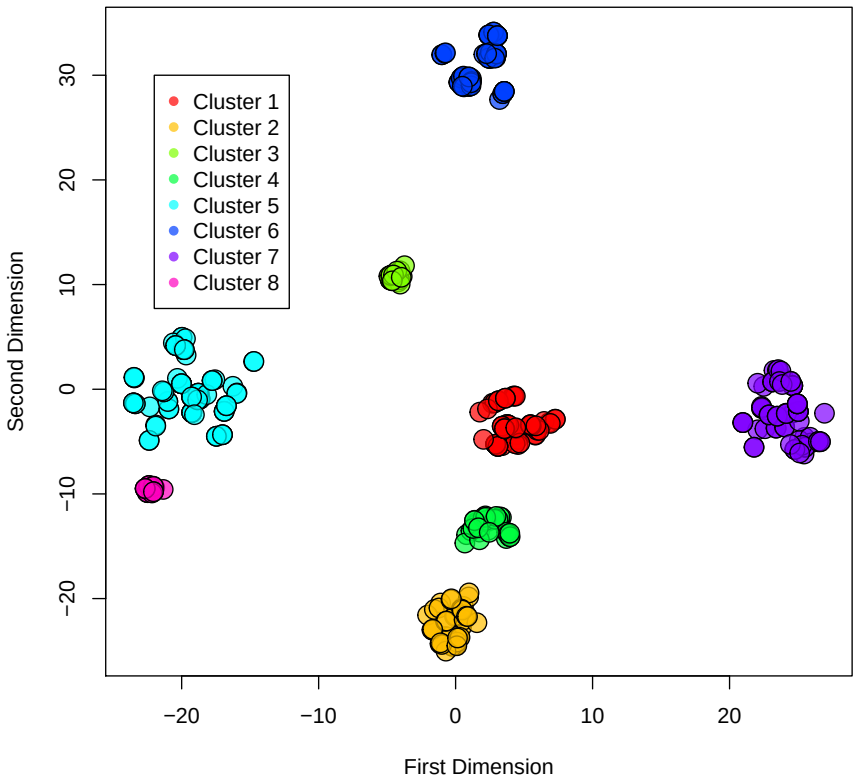

Figure 9: KODAMA plot.

The function `readMet` connects R to the HMDB database [6]. to retrieve chemical and functional information of each metabolite. This can be summarized using different functions: `substituentsMet`, `diseaseMet`, `enzymeMet`, `pathwaysMet`, `taxonomyMet`. The function `features` associates the most prominent features to each module.

```
doc2=readMet(ChemRICH$HMDB)
cla2=substituentsMet(doc2)
f_7clusters=features(doc2,cla2,clu2$clusters[,"Clusters 7"])
f_8clusters=features(doc2,cla2,clu2$clusters[,"Clusters 8"])
```

[6] https://www.hmdb.ca

8 How to Cite this Package

Ebtesam Abdel-Shafy, Tadele Melak, David A. MacIntyre, Giorgia Zadra, Luiz F. Zerbini, Silvano Piazza, and Stefano Cacciatore Publication in submission

To obtain BibTex entries of the two references, you can enter the following into your R session to Bibtex `citation("MetChem")`.

5 References

Cacciatore S, Luchinat C, Tenori L. Knowledge discovery by accuracy maximization. *Proc Natl Acad Sci USA* 2014; 111: 5117-22.

Cacciatore S, Tenori L, Luchinat C, Bennett PR, MacIntyre DA (2017) KODAMA: an R package for knowledge discovery and data mining. *Bioinformatics* 2017; 33(4): 621-623.

Labbé DP, Zadra G, Yang M, Reyes JM, Lin CY, Cacciatore S, Ebot EM, Creech AL, Giunchi F, Fiorentino M, Elfandy H, Syamala S, Karoly ED, Alshalalfa M, Erho N, Ross A, Schaeffer EM, Gibb EA, Takhar M, Den RB, Lehrer J, Karnes RJ, Freedland SJ, Davicioni E, Spratt DE, Ellis L, Jaffe JD, D'Amico AV, Kantoff PW, Bradner JE, Mucci LA, Chavarro JE, Loda M, Brown M. High-fat diet fuels prostate cancer progression by rewiring the metabolome and amplifying the MYC program. *Nat Commun* 2019; 10: 4358.

Zinga MM, Abdel-Shafy E, Melak T, Vignoli A, Piazza S, Zerbini LF, Tenori L, Cacciatore S. KODAMA exploratory analysis in metabolic phenotyping. *Front Mol Biosci* 2023; 9: 1436.