# Ionflow: Ionomics data network and enrichment analysis

*Wanchang Lin*

**01-12-2020**

## Contents

This vignette explains how to performs ionomics data analysis including gene network and enrichment analysis by using the modification of R package, ionflow. The modification(`ionflow_funcs`) was made by Wanchang Lin (w.lin@imperial.ac.uk) and Jacopo Iacovacci (j.iacovacci@imperial.ac.uk).

## Data preparation

To explore the pipeline, we'll use the ionomics data set:

```
ion_data <- read.table("../test-data/iondata.tsv", header = T, sep = "\t")
dim(ion_data)
#> [1] 9999   16
```

Ten random lines are shown as:

```
sample_n(ion_data, 10)
```

**Table 1: Samples of raw data**

| Knockout | Batch_ID | Ca | Cd | Co | Cu | Fe | K | Mg | Mn | Mo | Na | Ni | P | S | Zn |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| YOR239W | 89 | 20.88 | 0.95 | 0.14 | 1.35 | 7.70 | 2718.94 | 655.41 | 0.90 | 1.15 | 183.43 | 1.18 | 4324.82 | 427.11 | 12.12 |
| YOR178C | 26 | 42.56 | 0.99 | 0.13 | 1.55 | 8.09 | 3297.04 | 628.14 | 1.32 | 1.18 | 199.18 | 0.94 | 4772.22 | 491.50 | 15.70 |
| YOR185C | 26 | 43.31 | 1.19 | 0.15 | 1.65 | 6.97 | 3312.20 | 698.27 | 1.35 | 0.72 | 218.16 | 1.12 | 4856.83 | 472.15 | 17.20 |
| YCR017C | 21 | 46.21 | 0.89 | 0.21 | 2.10 | 10.78 | 2464.60 | 703.46 | 1.27 | 1.10 | 298.68 | 1.46 | 4894.45 | 545.89 | 16.03 |
| YLR396C | 39 | 56.05 | 1.04 | 0.16 | 1.57 | 12.87 | 1851.81 | 664.73 | 0.92 | 1.01 | 102.37 | 1.61 | 4395.39 | 663.09 | 16.91 |
| YCL045C | 20 | 59.45 | 1.35 | 0.21 | 2.05 | 9.85 | 2407.76 | 629.56 | 1.54 | 1.98 | 173.73 | 1.56 | 4332.01 | 883.05 | 16.98 |
| YHR203C | 20 | 46.85 | 1.16 | 0.21 | 1.93 | 19.09 | 2757.78 | 641.84 | 1.39 | 1.52 | 153.69 | 1.27 | 4089.49 | 444.12 | 16.93 |
| YJL151C | 28 | 41.28 | 1.09 | 0.14 | 1.68 | 8.56 | 2719.58 | 712.51 | 1.32 | 1.04 | 251.62 | 1.47 | 4544.00 | 606.34 | 16.41 |
| YDR207C | 35 | 65.51 | 1.17 | 0.21 | 0.99 | 7.29 | 1369.86 | 830.40 | 1.11 | 0.42 | 242.25 | 1.22 | 5372.71 | 531.86 | 16.48 |
| YDR209C | 35 | 29.50 | 1.09 | 0.23 | 1.29 | 7.84 | 2667.20 | 833.29 | 1.11 | 0.70 | 242.81 | 1.18 | 4996.98 | 552.91 | 17.36 |

The first few columns are meta information such as gene ORF and batch id. The rest is the ionomics data.

## Data pre-process

The raw data set should be pre-processed. The pre-processing function `PreProcessing` performs:

- log transformation
- batch correction
- outlier detection
- standardisation

The raw data are at first log transformed and then followed by the batch correction. The user can chose not to perform batch correction, otherwise the user can use either *median* or *median* plus *std* method. If there is quality control for the batch correction, the user can use it and indicates in the argument of `control_lines`. Also this function gives user option how to use these control line (`control_use`): If `control_use` is `control`, these control lines (data rows) are used for the batch correction factor; if `control.out`, lines except control lines are used.

This data set has a control line: **YDL227C** mutant. The code segment below is to identify it:

```
max(with(ion_data, table(Knockout)))
#> [1] 1617
which.max(with(ion_data, table(Knockout)))
#> YDL227C
#>     209
```

The next stage is outlier detection. Here only univariate methods are implemented, including *mad*, *IQR*, and *log.FC.dist*. And like batch correction, user can skip this procedure by setting `method_outliers = none` in the function argument. There is a threshold to control the number of outliers. The larger the threshold (`thres_outl`) the more outlier removal.

Standardisation provides three methods: *std*, *mad* or *custom*. If the method is *custom*, user must use specific std values such as:

```
std <- read.table("../test-data/user_std.tsv", header = T, sep = "\t")
std
#>    Ion     sd
#> 1   Ca 0.1508
#> 2   Cd 0.0573
#> 3   Co 0.0580
#> 4   Cu 0.0735
#> 5   Fe 0.1639
#> 6    K 0.0940
#> 7   Mg 0.0597
#> 8   Mn 0.0771
#> 9   Mo 0.1142
#> 10  Na 0.1075
#> 11  Ni 0.0784
#> 12   P 0.0597
#> 13   S 0.0801
#> 14  Zn 0.0671
```

The pre-process procedure returns not only processed ionomics data but also a symbolic data set. This data set is based on the ionomics data and is determined by a threshold(`thres_symb`):

- `0` if ionomics value is located between `[-thres_symb, thres_symb]`
- `1` if ionomics value is larger than `thres_symb`
- `-1` if ionomics value is smaller than `-thres_symb`

The core part of network and enrichment analysis, clustering, is based on the symbolic data.

Let's run the pre-process procedure:

```
pre <- PreProcessing(data = ion_data,
                     var_id = 1, batch_id = 2, data_id = 3,
                     method_norm = "median",
                     control_lines = "YDL227C",
                     control_use = "control",
                     method_outliers = "IQR",
                     thres_outl = 3,
                     stand_method = "std",
                     stdev = NULL,
                     thres_symb = 3)

names(pre)
#> [1] "stats.raw_data"    "stats.outliers"    "stats.batch_data"
#> [4] "data.long"         "data.gene.logFC"   "data.gene.zscores"
#> [7] "data.gene.symb"    "plot.dot"          "plot.hist"
```

The results includes summaries of raw data and processed data. The latter is:

```
pre$stats.batch_data %>%
  kable(caption = 'Processed data summary', digits = 2, booktabs = T) %>%
  kable_styling(full_width = F, font_size = 10)
```

**Table 2: Processed data summary**

| Ion | Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. | Variance |
|---|---|---|---|---|---|---|---|
| Ca | -4.45 | -0.28 | -0.13 | -0.12 | 0.02 | 2.35 | 0.11 |
| Cd | -1.70 | 0.03 | 0.10 | 0.11 | 0.17 | 0.93 | 0.03 |
| Co | -2.80 | 0.02 | 0.09 | 0.06 | 0.15 | 1.60 | 0.05 |
| Cu | -0.66 | -0.10 | -0.03 | -0.01 | 0.04 | 5.28 | 0.04 |
| Fe | -7.48 | -0.17 | -0.06 | -0.02 | 0.07 | 6.88 | 0.14 |
| K | -2.21 | -0.17 | -0.01 | -0.08 | 0.09 | 1.83 | 0.08 |
| Mg | -1.84 | -0.06 | 0.01 | -0.01 | 0.07 | 1.69 | 0.03 |
| Mn | -4.11 | -0.24 | -0.08 | -0.13 | 0.01 | 1.78 | 0.06 |
| Mo | -2.03 | -0.26 | -0.08 | -0.08 | 0.09 | 4.44 | 0.13 |
| Na | -7.41 | -0.53 | -0.22 | -0.33 | -0.04 | 1.25 | 0.24 |
| Ni | -2.40 | -0.01 | 0.09 | 0.12 | 0.21 | 7.90 | 0.12 |
| P | -1.18 | -0.06 | 0.00 | -0.01 | 0.06 | 1.45 | 0.02 |
| S | -2.38 | -0.03 | 0.05 | 0.06 | 0.16 | 2.38 | 0.04 |
| Zn | -0.46 | -0.08 | -0.03 | -0.01 | 0.03 | 4.60 | 0.02 |

The pre-processed data and symbolic data are like like:

```
pre$data.gene.zscores %>% head() %>%
  kable(caption = 'Processed data', digits = 2, booktabs = T) %>%
  kable_styling(full_width = F, font_size = 10,
                latex_options = c("striped", "scale_down"))
```

**Table 3:** Processed data

| Line | Ca | Cd | Co | Cu | Fe | K | Mg | Mn | Mo | Na | Ni | P | S | Zn |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| YAL004W | -1.16 | 0.75 | 1.19 | -0.47 | 0.04 | 0.61 | 0.51 | -0.84 | -0.08 | -1.84 | 1.71 | 0.52 | 0.33 | -0.09 |
| YAL005C | -1.67 | 0.84 | 0.55 | 0.58 | -2.79 | 0.59 | 0.31 | -1.16 | -1.42 | -0.12 | 1.48 | 0.73 | 0.13 | -0.13 |
| YAL007C | -2.12 | 0.64 | 0.23 | -0.53 | -0.24 | 0.79 | -0.09 | -0.14 | 1.22 | -0.92 | 0.00 | 0.09 | -0.29 | -0.65 |
| YAL008W | -2.34 | 1.13 | 0.21 | -0.73 | -2.16 | 0.52 | -0.02 | -0.87 | 0.93 | -0.58 | 0.02 | -0.09 | -0.73 | -0.47 |
| YAL009W | -1.18 | 0.66 | 0.55 | -1.11 | -3.91 | 0.22 | 0.09 | -0.18 | 1.50 | -0.84 | -0.09 | 0.14 | 0.01 | -0.36 |
| YAL010C | -1.28 | 1.43 | 2.27 | 0.46 | 1.53 | -2.75 | 0.04 | -0.74 | -9.71 | -4.30 | 2.42 | -0.98 | -0.05 | -0.01 |

```
pre$data.gene.symb %>% head() %>%
  kable(caption = 'Symbolic data', booktabs = T) %>%
  kable_styling(full_width = F, font_size = 10)
```

**Table 4:** Symbolic data

| Line | Ca | Cd | Co | Cu | Fe | K | Mg | Mn | Mo | Na | Ni | P | S | Zn |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| YAL004W | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| YAL005C | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| YAL007C | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| YAL008W | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| YAL009W | 0 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| YAL010C | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | -1 | 0 | 0 | 0 | 0 |

The symbolic data are calculated from the processed data with control of `thres_symb` (here is 3). You can obtain a new symbol data set by re-assigning a new threshold to the function `symbol_data`:

```
data_symb <- symbol_data(pre$data.gene.zscores, thres_symb = 2)
data_symb %>% head() %>%
  kable(caption = 'Symbolic data with threshold of 2', booktabs = T) %>%
  kable_styling(full_width = F, font_size = 10)
```

The pre-processed data distribution is:

```
pre$plot.hist
```

**Table 5:** Symbolic data with threshold of 2

| Line | Ca | Cd | Co | Cu | Fe | K | Mg | Mn | Mo | Na | Ni | P | S | Zn |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| YAL004W | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| YAL005C | 0 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| YAL007C | -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| YAL008W | -1 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| YAL009W | 0 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| YAL010C | 0 | 0 | 1 | 0 | 0 | -1 | 0 | 0 | -1 | -1 | 1 | 0 | 0 | 0 |

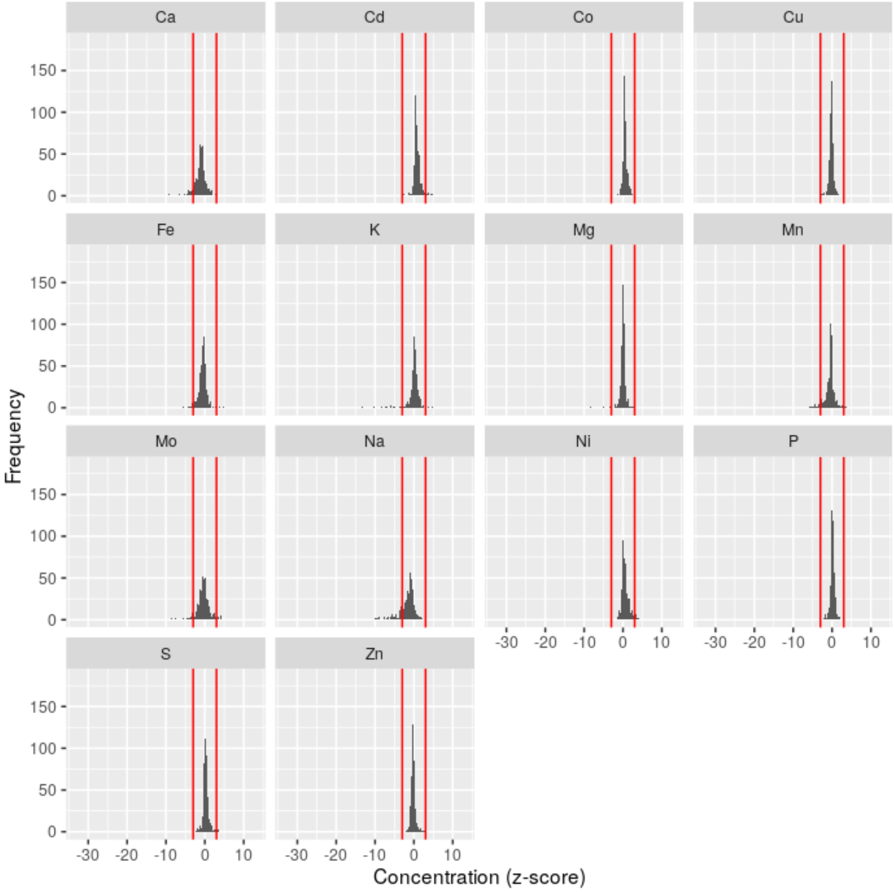

**Figure 1:** Ionomcs data distribution plot

## Data filtering

There are a lot of ways to filter genes. Here we filter genes based on symbolic data: remove genes with all values are zero.

```
data <- pre$data.gene.zscores
data_symb <- pre$data.gene.symb
```

```
idx <- rowSums(abs(data_symb[, -1])) > 0
dat <- data[idx, ]
dat_symb <- data_symb[idx, ]
dim(dat)
#> [1] 549  15
```

## Data clustering

The hierarchical cluster analysis is the key part of gene network and gene enrichment analysis. The methodology is as follow:

- Compute the distance of symbolic data
- Hierarchical cluster analysis on the distance
- Identify clusters/groups with a threshold of minimal number of cluster size

One example is:

```
clust <- gene_clus(dat_symb[, -1], min_clust_size = 10)
names(clust)
#> [1] "clus"    "idx"     "tab"     "tab_sub"
```

The cluster centres are:

```
clust$tab_sub
#>   cluster nGenes
#> 1      11     72
#> 2       7     36
#> 3       1     27
#> 4      18     15
#> 5       5     12
#> 6       3     11
#> 7       8     11
```

It indicates that clusters and their number of genes (larger than `min_cluster_size`).

## Gene network

The gene network uses both the ionomics and symbolic data. The similarity measures on the ionomics data are filtered by the similarity threshold located between 0 and 1, and cluster centres of symbolic data. The filter values are then used for network analysis.

The similarity measure method is one of *pearson*, *spearman*, *kendall*, *cosine*, *mahal_cosine* or *hybrid_mahal_cosine*. For the last two methods, see publication: Extraction and Integration of Genetic Networks from Short-Profile Omic Data Sets for details.

For example, we use the Pearson correlation as similarity measure for network analysis:

```
net <- GeneNetwork(data = dat,
                   data_symb = dat_symb,
                   min_clust_size = 10,
                   thres_corr = 0.75,
                   method_corr = "pearson")
```

The network with nodes coloured by the symbolic data clustering is:

```
net$plot.pnet1
```

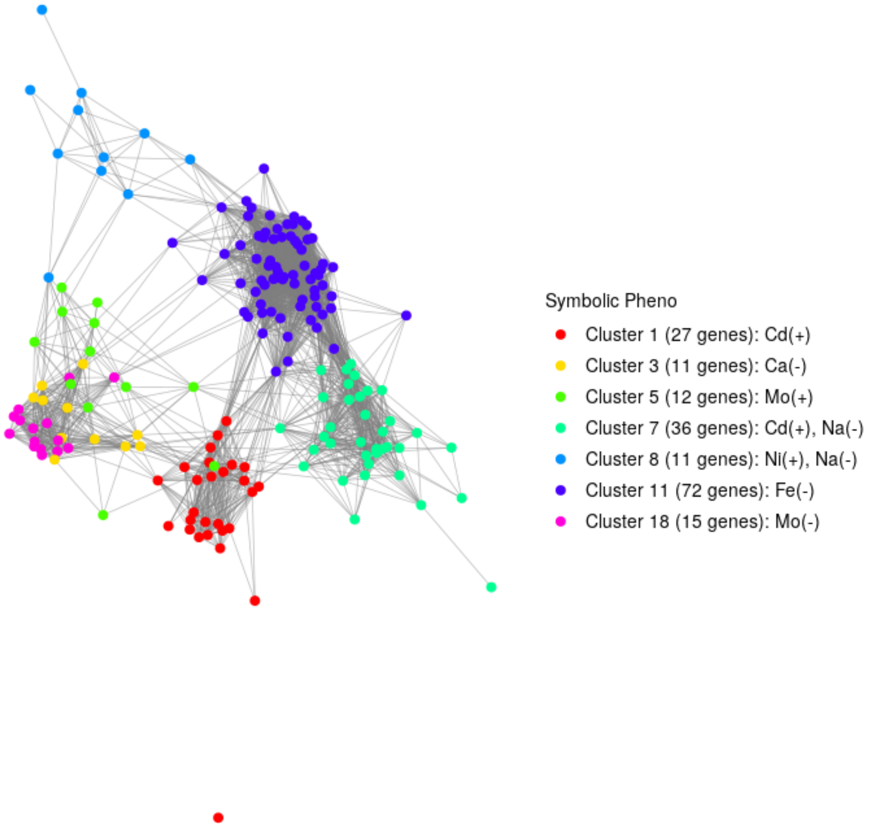
Symbolic Pheno
● Cluster 1 (27 genes): Cd(+)
● Cluster 3 (11 genes): Ca(-)
● Cluster 5 (12 genes): Mo(+)
● Cluster 7 (36 genes): Cd(+), Na(-)
● Cluster 8 (11 genes): Ni(+), Na(-)
● Cluster 11 (72 genes): Fe(-)
● Cluster 18 (15 genes): Mo(-)

**Figure 2:** Netwok analysis based on Pearson correlation: symbolic clustering

The same network, but nodes are coloured by the network community detection:

```
net$plot.pnet2
```

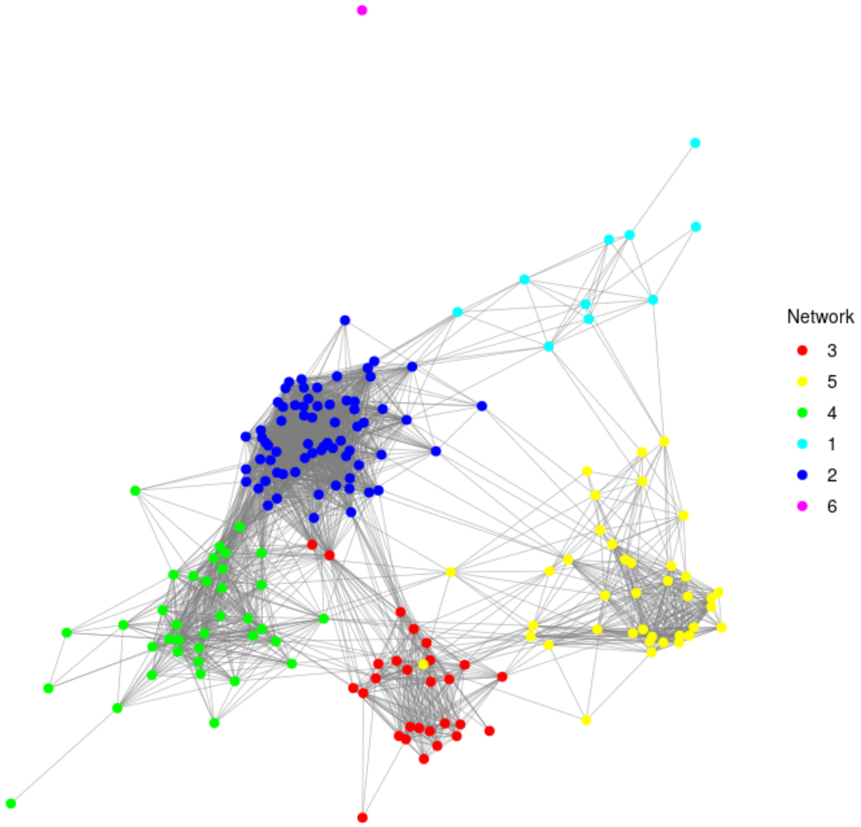

**Figure 3:** Netwok analysis based on Pearson correlation: community detction

The network analysis also returns a network impact and betweenness plot:

```
net$plot.impact_betweenness
```

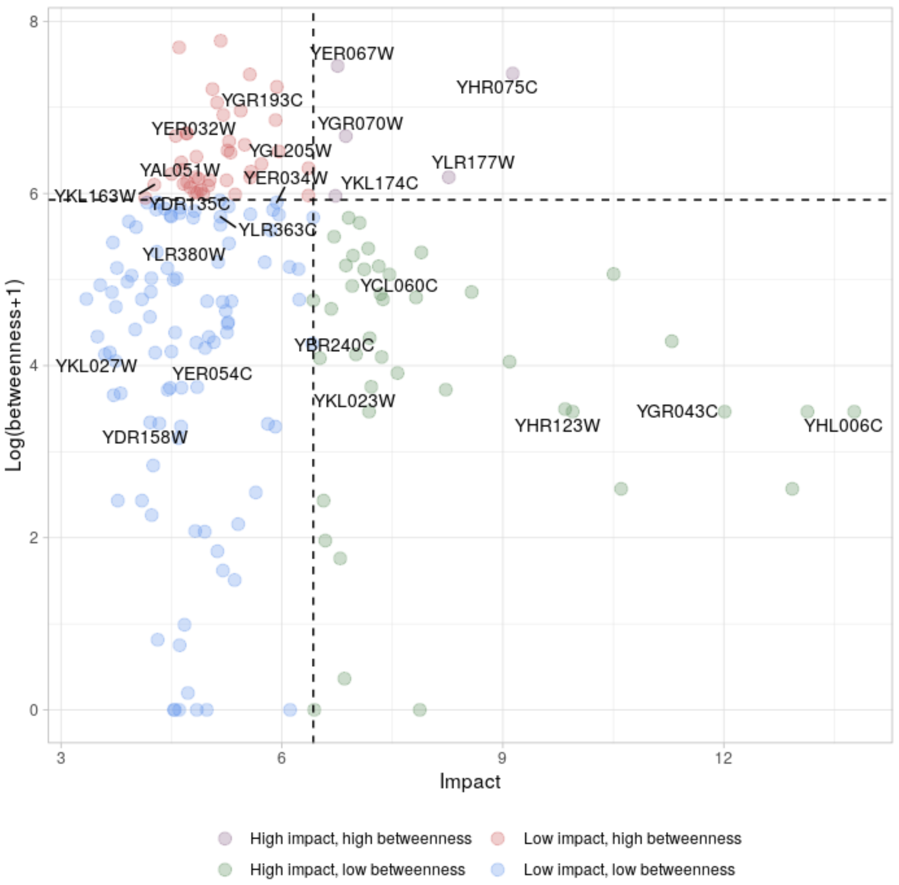

**Figure 4:** Netwok analysis based on Pearson correlation: impact and betweeness

For the comparison purpose, we use different similarity methods. Here we choose *Cosine*:

```
net_1 <- GeneNetwork(data = dat,
                     data_symb = dat_symb,
                     min_clust_size = 10,
                     thres_corr = 0.75,
                     method_corr = "cosine")
net_1$plot.pnet1
```

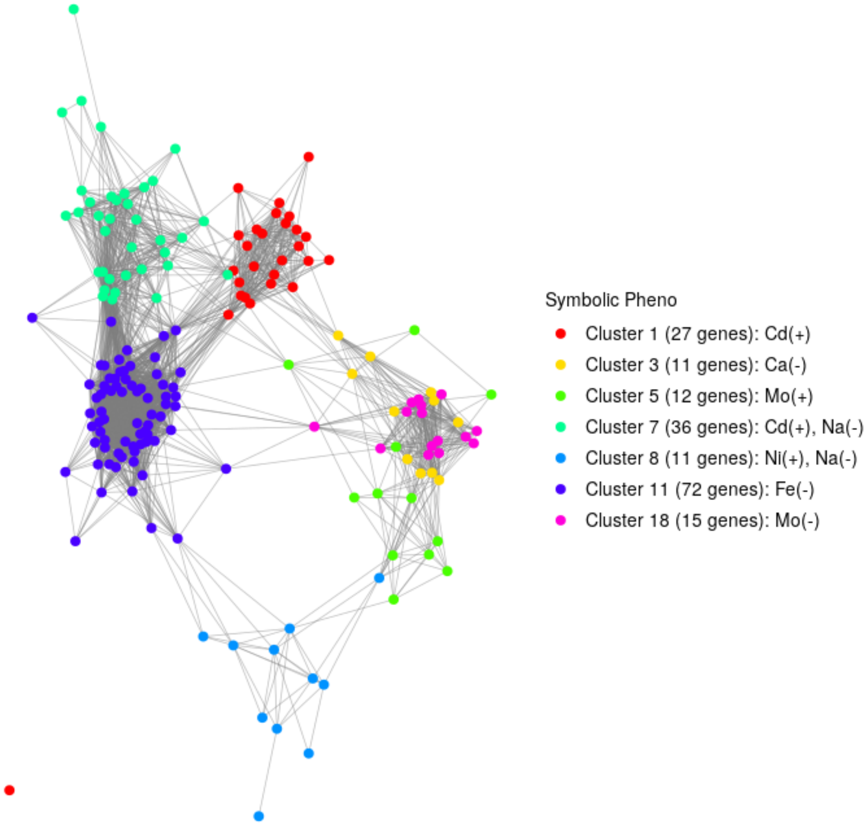
Symbolic Pheno
● Cluster 1 (27 genes): Cd(+)
● Cluster 3 (11 genes): Ca(-)
● Cluster 5 (12 genes): Mo(+)
● Cluster 7 (36 genes): Cd(+), Na(-)
● Cluster 8 (11 genes): Ni(+), Na(-)
● Cluster 11 (72 genes): Fe(-)
● Cluster 18 (15 genes): Mo(-)

**Figure 5: Netwok analysis based on Cosine**

```
net_1$plot.pnet2
```

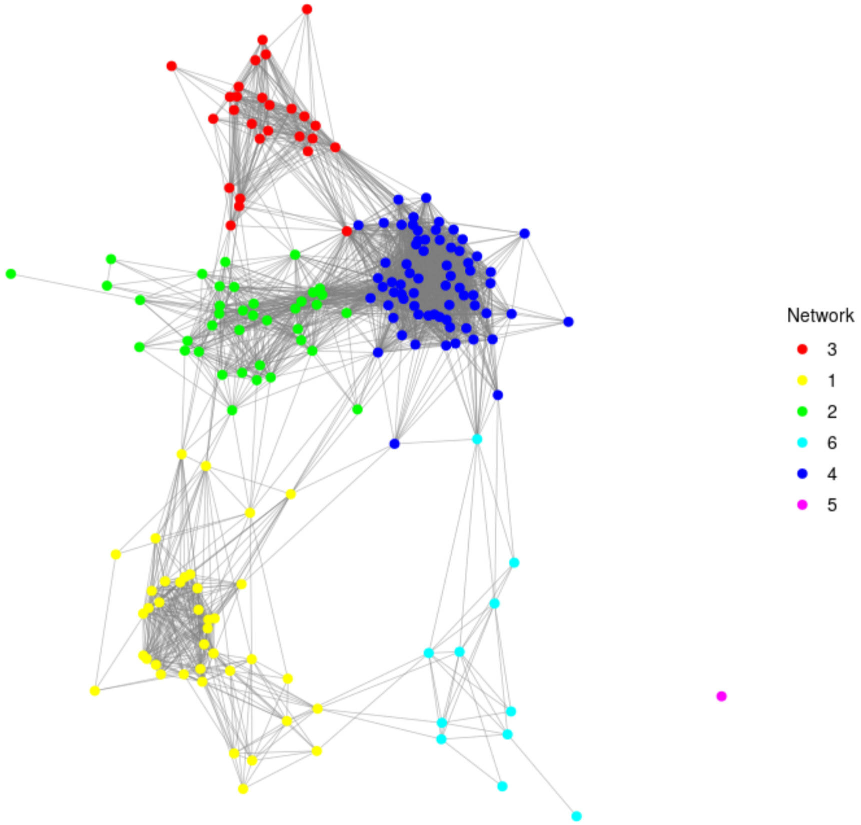

Figure 6: Netwok analysis based on Cosine

Use *Hybrid Mahalanobis Cosine*:

```
net_2 <- GeneNetwork(data = dat,
                     data_symb = dat_symb,
                     min_clust_size = 10,
                     thres_corr = 0.75,
                     method_corr = "mahal_cosine")
net_2$plot.pnet1
```

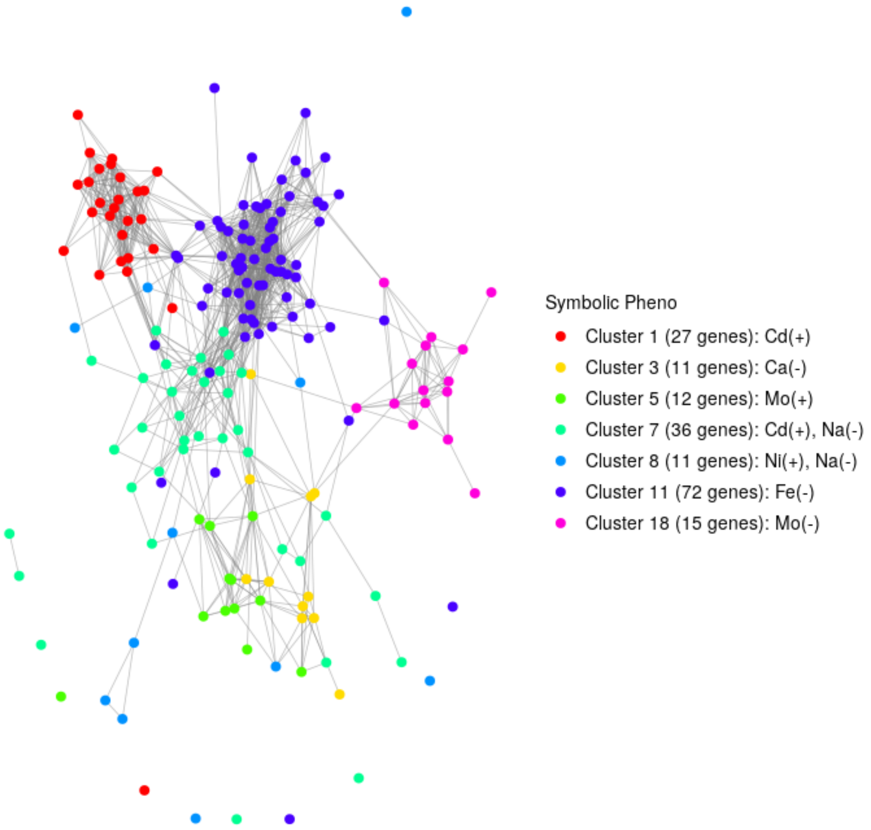

**Figure 7:** **Netwok analysis based on Mahalanobis Cosine**

```
net_2$plot.pnet2
```

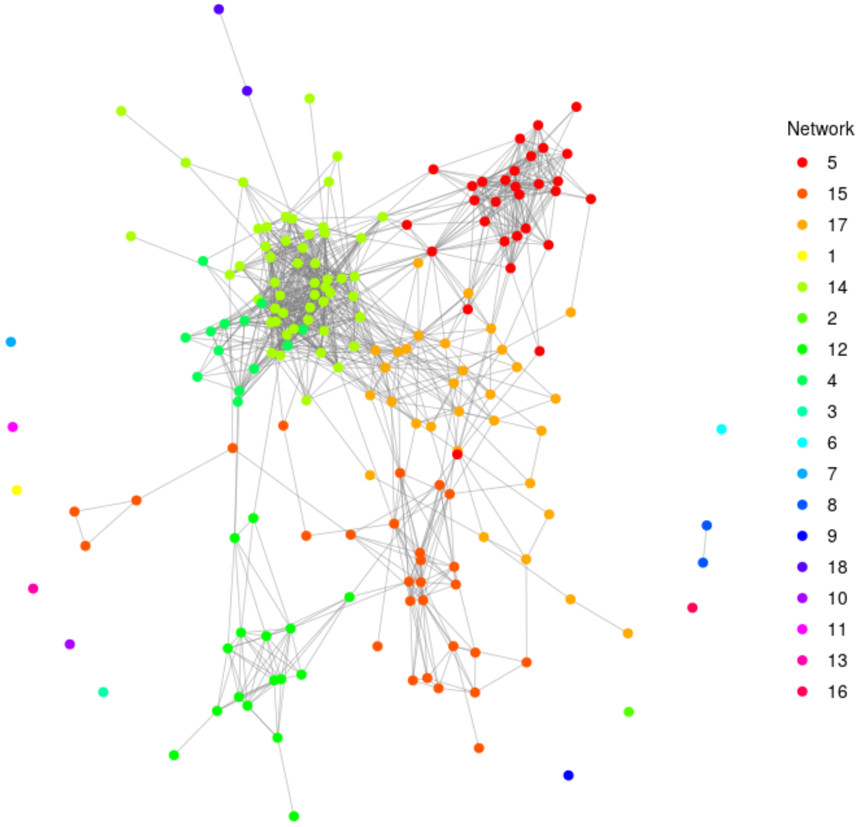

**Figure 8:** Netwok analysis based on Mahalanobis Cosine

Again, we use *Hybrid Mahalanobis Cosine*:

```
net_3 <- GeneNetwork(data = dat,
                     data_symb = dat_symb,
                     min_clust_size = 10,
                     thres_corr = 0.75,
                     method_corr = "hybrid_mahal_cosine")
net_3$plot.pnet1
```

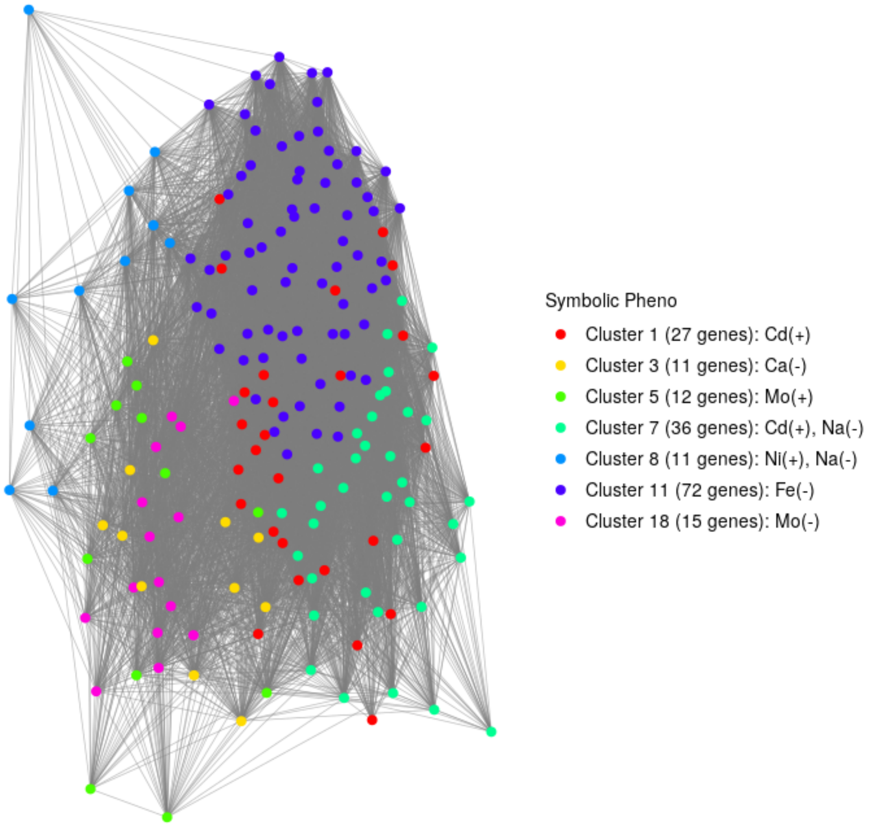

**Figure 9:** **Netwok analysis based on Hybrid Mahalanobis Cosine**

```
net_3$plot.pnet2
```

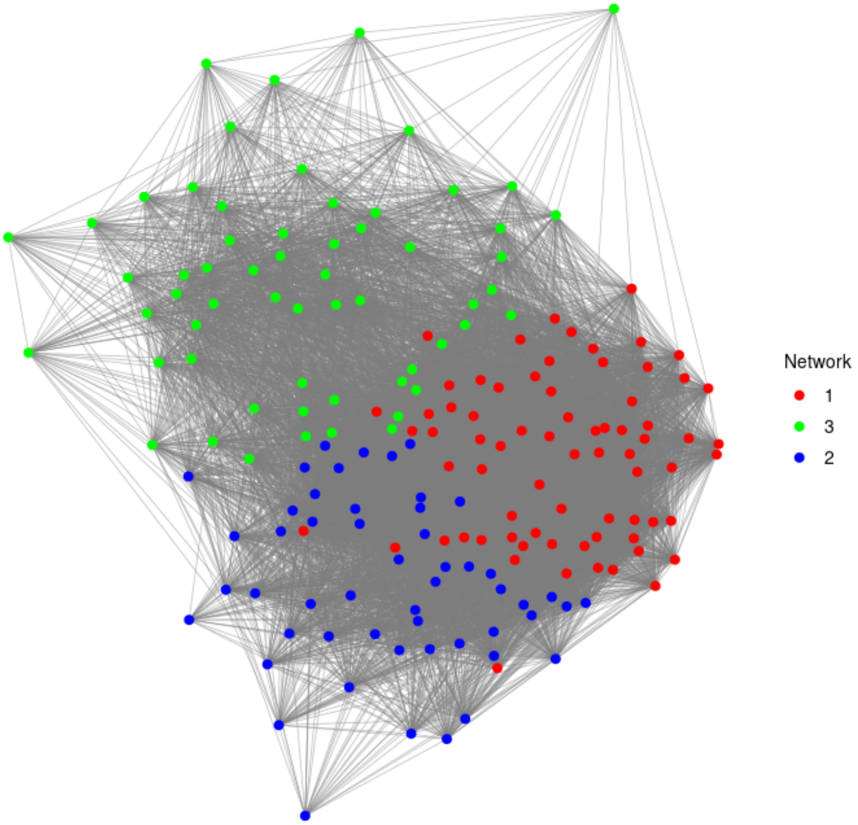

**Figure 10:** Netwok analysis based on Hybrid Mahalanobis Cosine

# Enrichment analysis

The enrichment analysis is based on symbolic data clustering. The genes in clusters are considered target gene sets while genes in the whole data set is the universe gene set.

The KEGG enrichment analysis:

```
kegg <- kegg_enrich(data = dat_symb, min_clust_size = 10, pval = 0.05,
                    annot_pkg =  "org.Sc.sgd.db")

#' kegg
kegg %>%
  kable(caption = 'KEGG enrichmenat analysis', digits = 3, booktabs = T) %>%
  kable_styling(full_width = F, font_size = 10,
                latex_options = c("striped", "scale_down"))
```

**Table 6:** KEGG enrichmenat analysis

| Cluster | KEGGID | Pvalue | Count | Size | Term |
|---|---|---|---|---|---|
| Cluster 18 (15 genes) | 00290 | 0.009 | 2 | 2 | Valine, leucine and isoleucine biosynthesis |
| Cluster 18 (15 genes) | 00520 | 0.009 | 2 | 2 | Amino sugar and nucleotide sugar metabolism |
| Cluster 18 (15 genes) | 00260 | 0.012 | 3 | 6 | Glycine, serine and threonine metabolism |
| Cluster 18 (15 genes) | 00010 | 0.024 | 2 | 3 | Glycolysis / Gluconeogenesis |
| Cluster 18 (15 genes) | 01110 | 0.037 | 5 | 22 | Biosynthesis of secondary metabolites |
| Cluster 3 (11 genes) | 00400 | 0.009 | 2 | 2 | Phenylalanine, tyrosine and tryptophan biosynthesis |
| Cluster 8 (11 genes) | 01100 | 0.006 | 6 | 55 | Metabolic pathways |
| Cluster 8 (11 genes) | 00564 | 0.027 | 2 | 6 | Glycerophospholipid metabolism |

Note that there can be none results for KRGG enrichment analysis. Change arguments such as `thres_clus` as appropriate.

The GO Terms enrichment analysis:

```
go <- go_enrich(data = dat_symb, min_clust_size = 10, pval = 0.05,
                ont = "BP", annot_pkg =  "org.Sc.sgd.db")
#' go
go %>%
  kable(caption = 'GO Terms enrichmenat analysis', digits = 3, booktabs = T) %>%
  kable_styling(full_width = F, font_size = 10,
                latex_options = c("striped", "scale_down"))
```

Table 7: GO Terms enrichmenat analysis

| Cluster | ID | Description | Pvalue | Count | CountUniverse | Ontology |
|---|---|---|---|---|---|---|
| Cluster 11 (72 genes) | GO:0051336 | regulation of hydrolase activity | 0.0018 | 4 | 12 | BP |
| Cluster 11 (72 genes) | GO:0043085 | positive regulation of catalytic activity | 0.0044 | 4 | 15 | BP |
| Cluster 11 (72 genes) | GO:0035303 | regulation of dephosphorylation | 0.0068 | 2 | 3 | BP |
| Cluster 11 (72 genes) | GO:0046889 | positive regulation of lipid biosynthetic process | 0.0068 | 2 | 3 | BP |
| Cluster 11 (72 genes) | GO:1903727 | positive regulation of phospholipid metabolic process | 0.0068 | 2 | 3 | BP |
| Cluster 11 (72 genes) | GO:0044764 | multi-organism cellular process | 0.0074 | 3 | 9 | BP |
| Cluster 11 (72 genes) | GO:0045833 | negative regulation of lipid metabolic process | 0.0132 | 2 | 4 | BP |
| Cluster 11 (72 genes) | GO:0009890 | negative regulation of biosynthetic process | 0.0203 | 5 | 34 | BP |
| Cluster 11 (72 genes) | GO:0032880 | regulation of protein localization | 0.0214 | 2 | 5 | BP |
| Cluster 11 (72 genes) | GO:0048869 | cellular developmental process | 0.0231 | 4 | 24 | BP |
| Cluster 11 (72 genes) | GO:0019220 | regulation of phosphate metabolic process | 0.0259 | 2 | 6 | BP |
| Cluster 11 (72 genes) | GO:0042180 | cellular ketone metabolic process | 0.0311 | 2 | 6 | BP |
| Cluster 11 (72 genes) | GO:0043547 | positive regulation of GTPase activity | 0.0311 | 2 | 6 | BP |
| Cluster 11 (72 genes) | GO:0031324 | negative regulation of cellular metabolic process | 0.0471 | 5 | 42 | BP |
| Cluster 7 (36 genes) | GO:0007031 | peroxisome organization | 0.0093 | 2 | 8 | BP |
| Cluster 1 (27 genes) | GO:0006974 | cellular response to DNA damage stimulus | 0.0122 | 3 | 22 | BP |
| Cluster 1 (27 genes) | GO:0048522 | positive regulation of cellular process | 0.0405 | 2 | 15 | BP |
| Cluster 18 (15 genes) | GO:0043436 | oxoacid metabolic process | 0.0037 | 8 | 44 | BP |
| Cluster 18 (15 genes) | GO:0006084 | acetyl-CoA metabolic process | 0.0039 | 2 | 2 | BP |
| Cluster 18 (15 genes) | GO:0006086 | acetyl-CoA biosynthetic process from pyruvate | 0.0039 | 2 | 2 | BP |
| Cluster 18 (15 genes) | GO:0006567 | threonine catabolic process | 0.0039 | 2 | 2 | BP |
| Cluster 18 (15 genes) | GO:0009097 | isoleucine biosynthetic process | 0.0039 | 2 | 2 | BP |
| Cluster 18 (15 genes) | GO:0033866 | nucleoside bisphosphate biosynthetic process | 0.0039 | 2 | 2 | BP |
| Cluster 18 (15 genes) | GO:0071616 | acyl-CoA biosynthetic process | 0.0039 | 2 | 2 | BP |
| Cluster 18 (15 genes) | GO:0009066 | aspartate family amino acid metabolic process | 0.0062 | 3 | 7 | BP |
| Cluster 18 (15 genes) | GO:1901606 | alpha-amino acid catabolic process | 0.0104 | 3 | 8 | BP |
| Cluster 18 (15 genes) | GO:0046394 | carboxylic acid biosynthetic process | 0.0109 | 5 | 23 | BP |
| Cluster 18 (15 genes) | GO:0033875 | ribonucleoside bisphosphate metabolic process | 0.0112 | 2 | 3 | BP |
| Cluster 18 (15 genes) | GO:0034032 | purine nucleoside bisphosphate metabolic process | 0.0112 | 2 | 3 | BP |
| Cluster 18 (15 genes) | GO:0035383 | thioester metabolic process | 0.0112 | 2 | 3 | BP |
| Cluster 18 (15 genes) | GO:0044272 | sulfur compound biosynthetic process | 0.0204 | 3 | 10 | BP |
| Cluster 18 (15 genes) | GO:0046395 | carboxylic acid catabolic process | 0.0268 | 3 | 11 | BP |
| Cluster 18 (15 genes) | GO:0051186 | cofactor metabolic process | 0.0368 | 4 | 21 | BP |
| Cluster 18 (15 genes) | GO:0055086 | nucleobase-containing small molecule metabolic process | 0.0368 | 4 | 21 | BP |
| Cluster 18 (15 genes) | GO:0044283 | small molecule biosynthetic process | 0.0402 | 5 | 32 | BP |
| Cluster 18 (15 genes) | GO:0006164 | purine nucleotide biosynthetic process | 0.0496 | 2 | 6 | BP |
| Cluster 18 (15 genes) | GO:0009069 | serine family amino acid metabolic process | 0.0496 | 2 | 6 | BP |
| Cluster 3 (11 genes) | GO:0002376 | immune system process | 0.0173 | 2 | 2 | BP |
| Cluster 3 (11 genes) | GO:0006952 | defense response | 0.0173 | 2 | 2 | BP |
| Cluster 3 (11 genes) | GO:0009073 | aromatic amino acid family biosynthetic process | 0.0173 | 2 | 2 | BP |
| Cluster 3 (11 genes) | GO:0009423 | chorismate biosynthetic process | 0.0173 | 2 | 2 | BP |
| Cluster 3 (11 genes) | GO:0009607 | response to biotic stimulus | 0.0173 | 2 | 2 | BP |
| Cluster 3 (11 genes) | GO:0035335 | peptidyl-tyrosine dephosphorylation | 0.0173 | 2 | 2 | BP |
| Cluster 3 (11 genes) | GO:0046835 | carbohydrate phosphorylation | 0.0173 | 2 | 2 | BP |
| Cluster 3 (11 genes) | GO:0051607 | defense response to virus | 0.0173 | 2 | 2 | BP |
| Cluster 3 (11 genes) | GO:0051707 | response to other organism | 0.0173 | 2 | 2 | BP |
| Cluster 3 (11 genes) | GO:0005975 | carbohydrate metabolic process | 0.0352 | 7 | 25 | BP |
| Cluster 3 (11 genes) | GO:0045814 | negative regulation of gene expression, epigenetic | 0.045 | 4 | 11 | BP |
| Cluster 3 (11 genes) | GO:0000291 | nuclear-transcribed mRNA catabolic process, exonucleolytic | 0.0475 | 2 | 3 | BP |
| Cluster 3 (11 genes) | GO:0018105 | peptidyl-serine phosphorylation | 0.0475 | 2 | 3 | BP |
| Cluster 3 (11 genes) | GO:0045815 | positive regulation of gene expression, epigenetic | 0.0475 | 2 | 3 | BP |
| Cluster 3 (11 genes) | GO:0046777 | protein autophosphorylation | 0.0475 | 2 | 3 | BP |
| Cluster 3 (11 genes) | GO:0060969 | negative regulation of gene silencing | 0.0475 | 2 | 3 | BP |
| Cluster 3 (11 genes) | GO:0070478 | nuclear-transcribed mRNA catabolic process, 3'-5' exonucleolytic nonsense-mediated decay | 0.0475 | 2 | 3 | BP |
| Cluster 3 (11 genes) | GO:0070481 | nuclear-transcribed mRNA catabolic process, non-stop decay | 0.0475 | 2 | 3 | BP |
| Cluster 8 (11 genes) | GO:0006646 | phosphatidylethanolamine biosynthetic process | 0.0021 | 2 | 3 | BP |
| Cluster 8 (11 genes) | GO:0046174 | polyol catabolic process | 0.0021 | 2 | 3 | BP |
| Cluster 8 (11 genes) | GO:1901616 | organic hydroxy compound catabolic process | 0.0068 | 2 | 5 | BP |
| Cluster 8 (11 genes) | GO:0006650 | glycerophospholipid metabolic process | 0.0089 | 2 | 6 | BP |
| Cluster 8 (11 genes) | GO:0006629 | lipid metabolic process | 0.0095 | 4 | 33 | BP |
| Cluster 8 (11 genes) | GO:0046165 | alcohol biosynthetic process | 0.0138 | 2 | 7 | BP |
| Cluster 8 (11 genes) | GO:0044282 | small molecule catabolic process | 0.0191 | 3 | 22 | BP |
| Cluster 8 (11 genes) | GO:0034599 | cellular response to oxidative stress | 0.0401 | 2 | 12 | BP |
| Cluster 8 (11 genes) | GO:0006796 | phosphate-containing compound metabolic process | 0.0417 | 3 | 31 | BP |
| Cluster 8 (11 genes) | GO:0045017 | glycerolipid biosynthetic process | 0.0466 | 2 | 13 | BP |
| Cluster 8 (11 genes) | GO:0019637 | organophosphate metabolic process | 0.0478 | 3 | 36 | BP |

## Exploratory analysis

The explanatory analysis performs PCA and correlation analysis for ions in terms of genes, i.e. ions are samples/replicates while genes are variables/features.

We can use the pre-processed data `dat` for explanatory analysis:

```
expl <- ExploratoryAnalysis(data = dat)
names(expl)
#> [1] "plot.pca"       "data.pca.load"  "plot.corr"      "plot.corr.heat"
#> [5] "plot.heat"      "plot.net"
```

The ionome PCA plot is:

```
expl$plot.pca
```

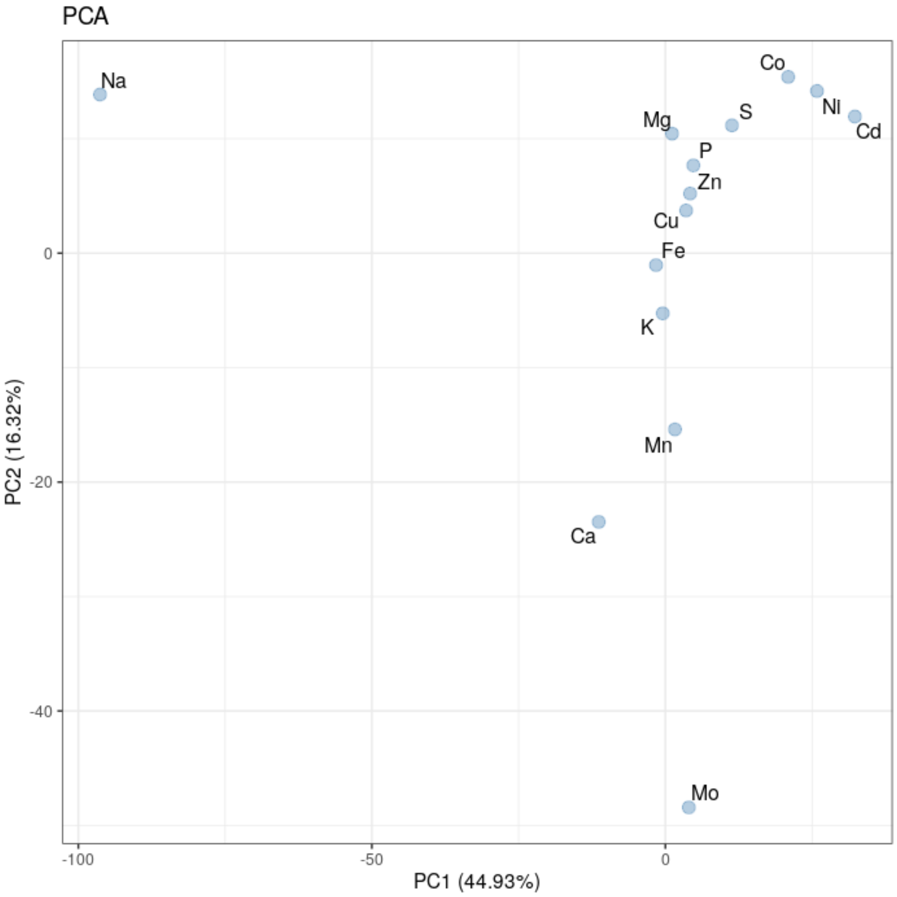

**Figure 11: Ion PCA plot**

The Person correlation of ions are shown in correlation plot, heatmap and network plot:

```
expl$plot.corr
```

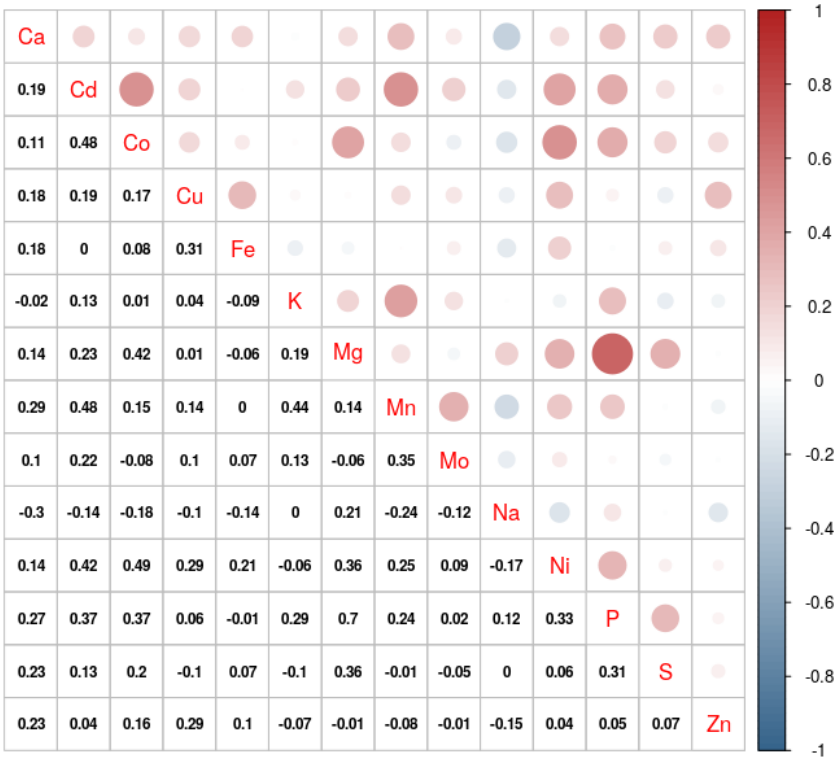

**Figure 12:** Ion correlation plots

```
expl$plot.corr.heat
```

```
expl$plot.net
```

The correlation between ions and genes are shown in heatmap with dendrogram:

```
expl$plot.heat
```

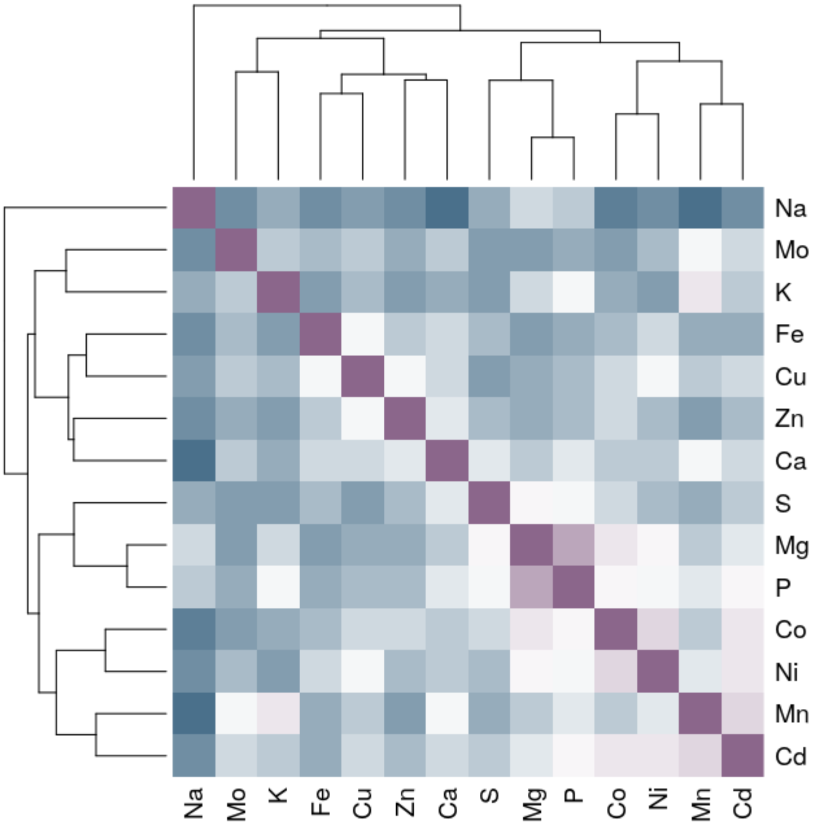

**Figure 13:** Ion correlation plots

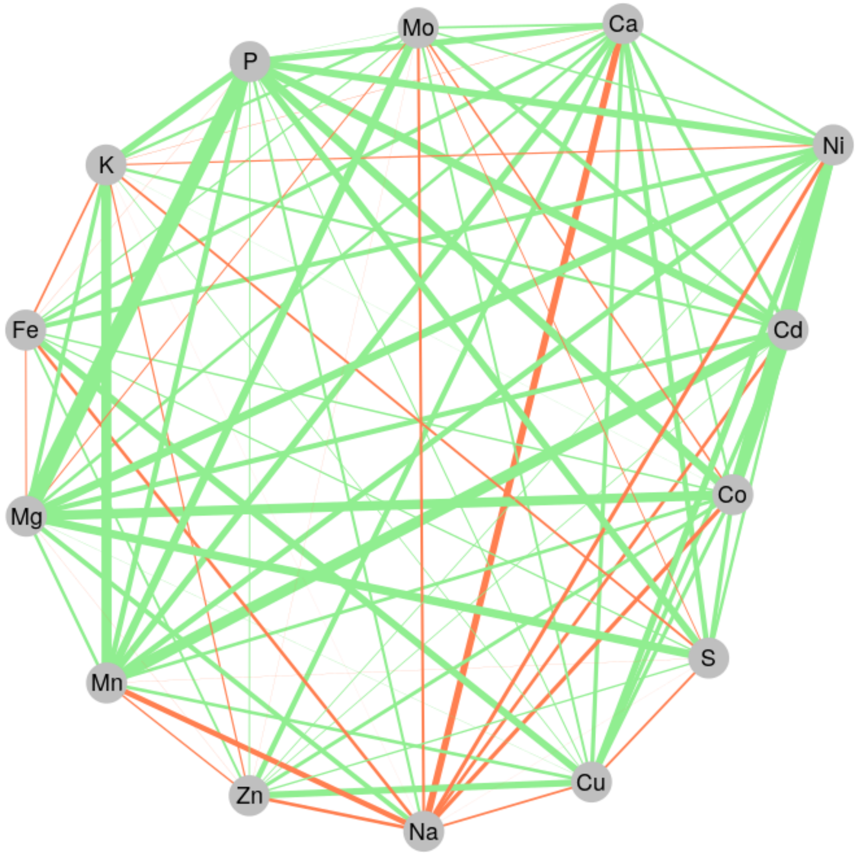

**Figure 14:** Ion correlation plots

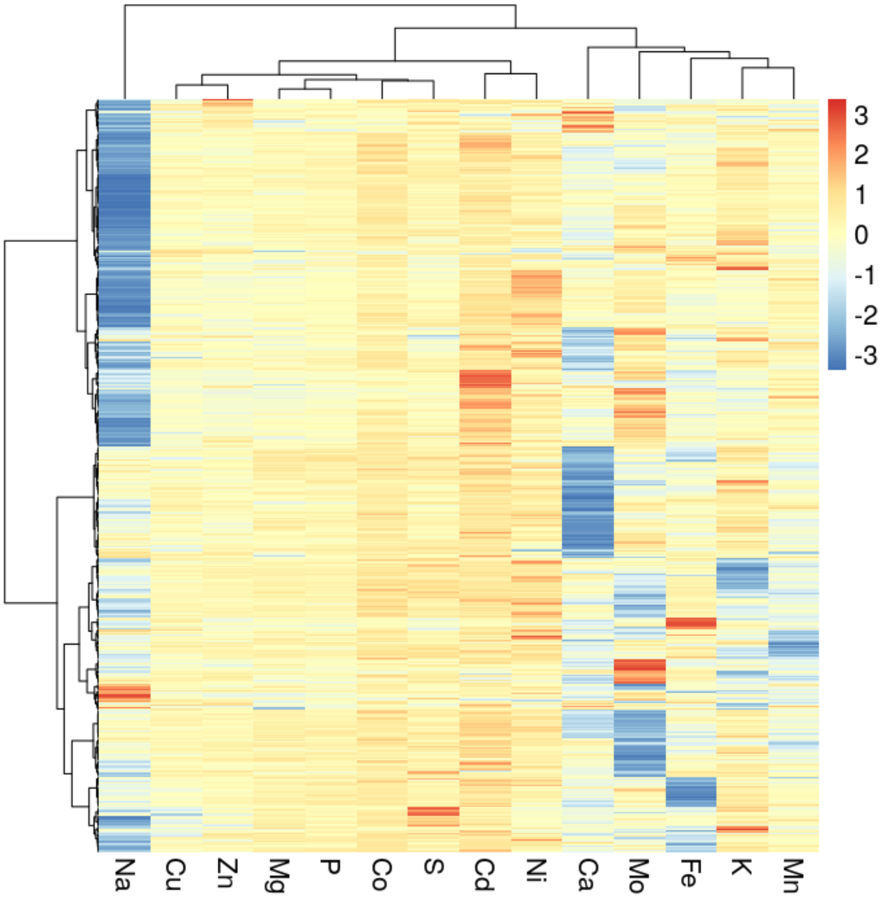

**Figure 15:** Correlation plots between ions and genes

Actually we can perform explanatory analysis using results of gene clustering:

```
#' update data set with results of gene clustering
dat_clus <- dat[clust$idx, ]
dim(dat_clus)
#> [1] 184  15

expl.1 <- ExploratoryAnalysis(data = dat_clus)
```

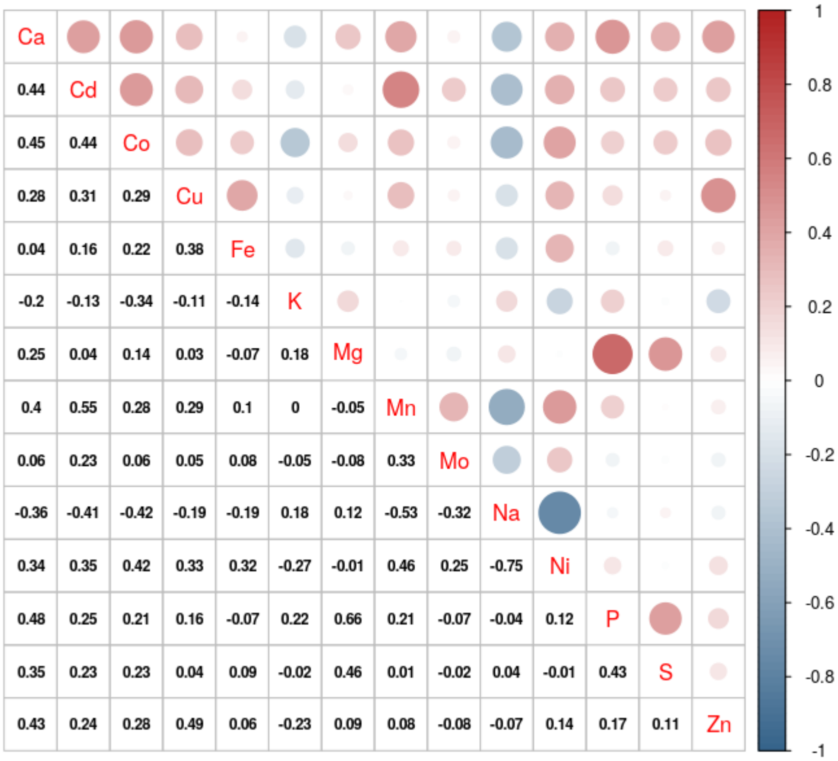

**Figure 16:** Explanotory analysis plots for gene clustering

```
expl.1$plot.pca
```

```
expl.1$plot.net
```

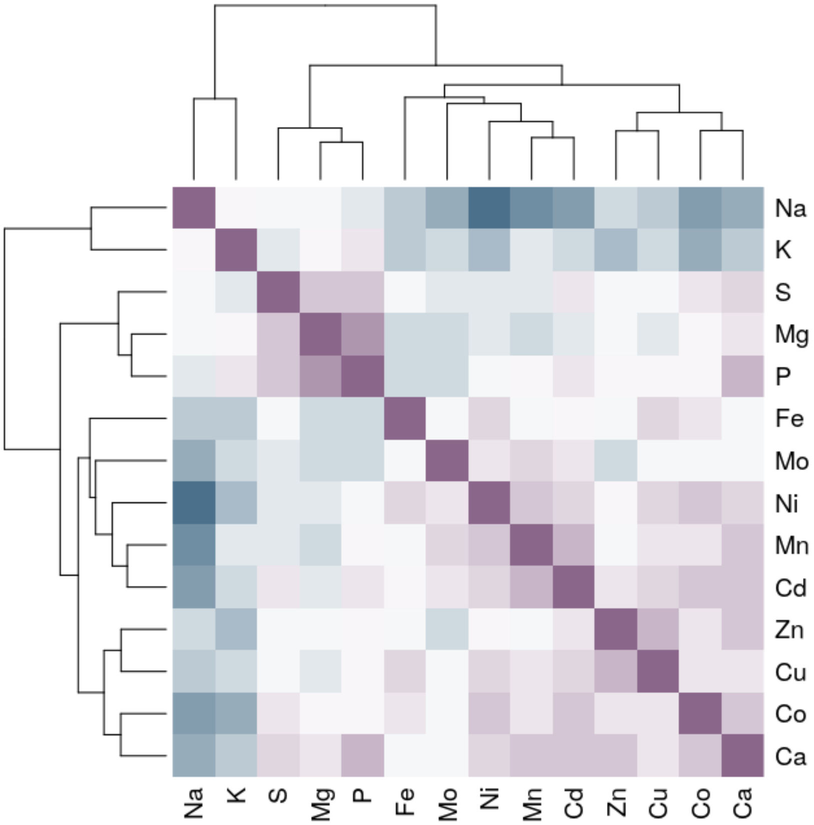

**Figure 17:** Explanotory analysis plots for gene clustering

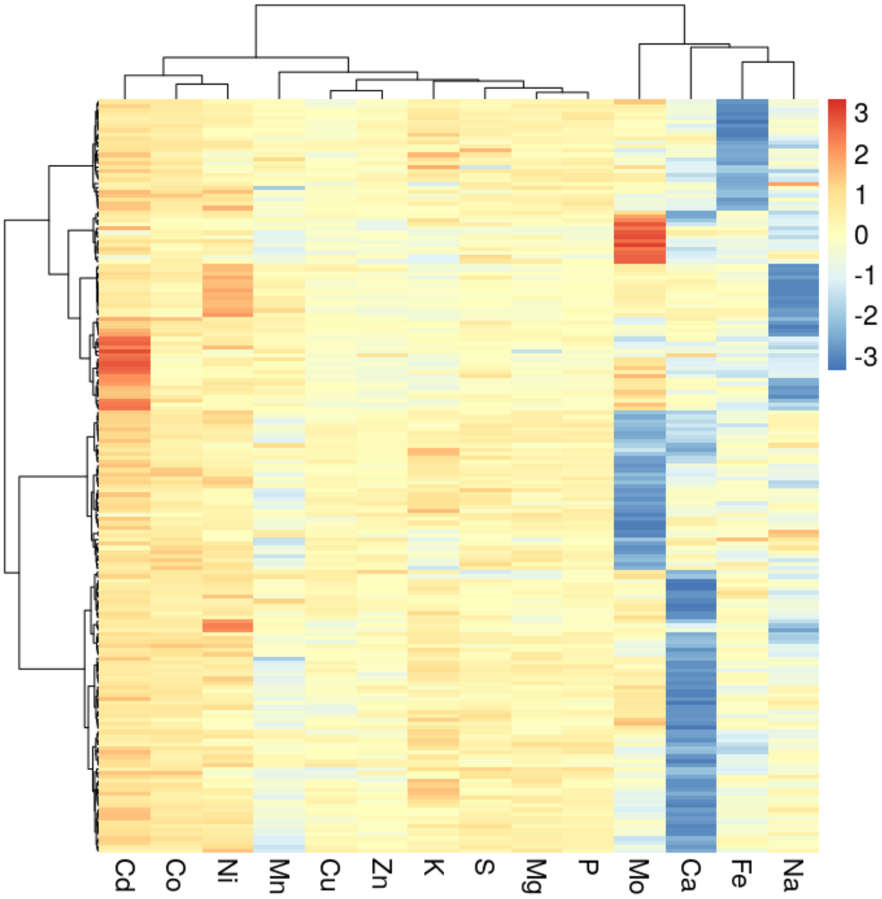

**Figure 18:** Explanotory analysis plots for gene clustering

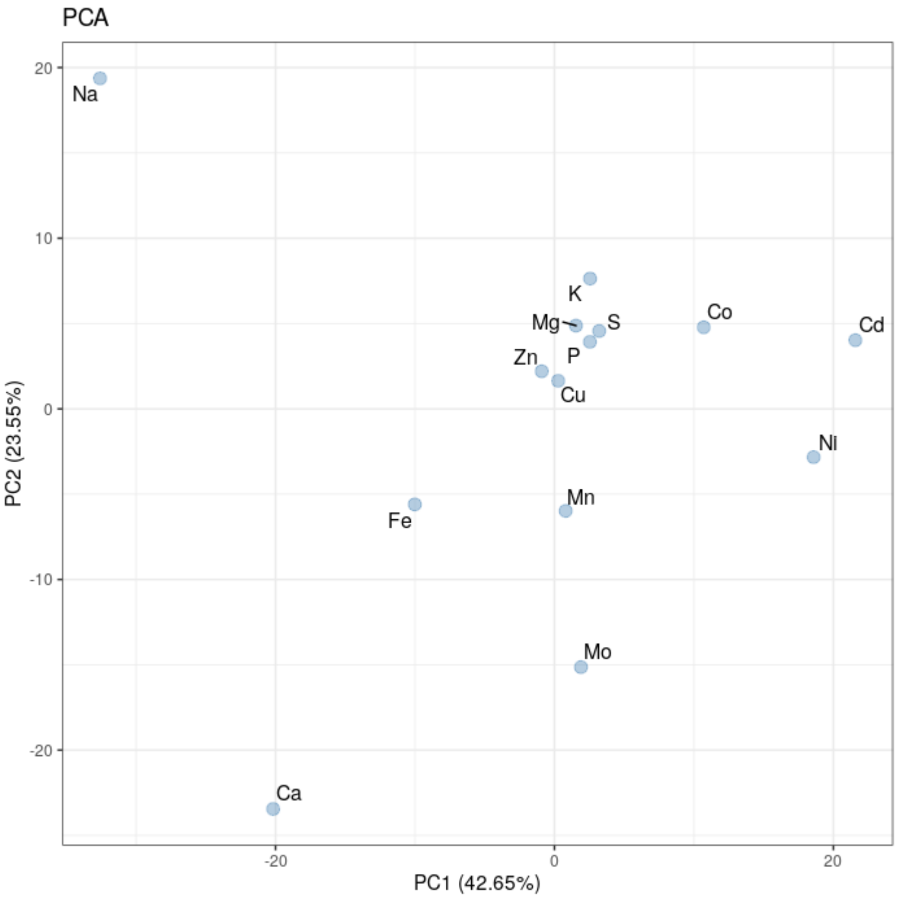

**Figure 19:** Explanotory analysis plots for gene clustering

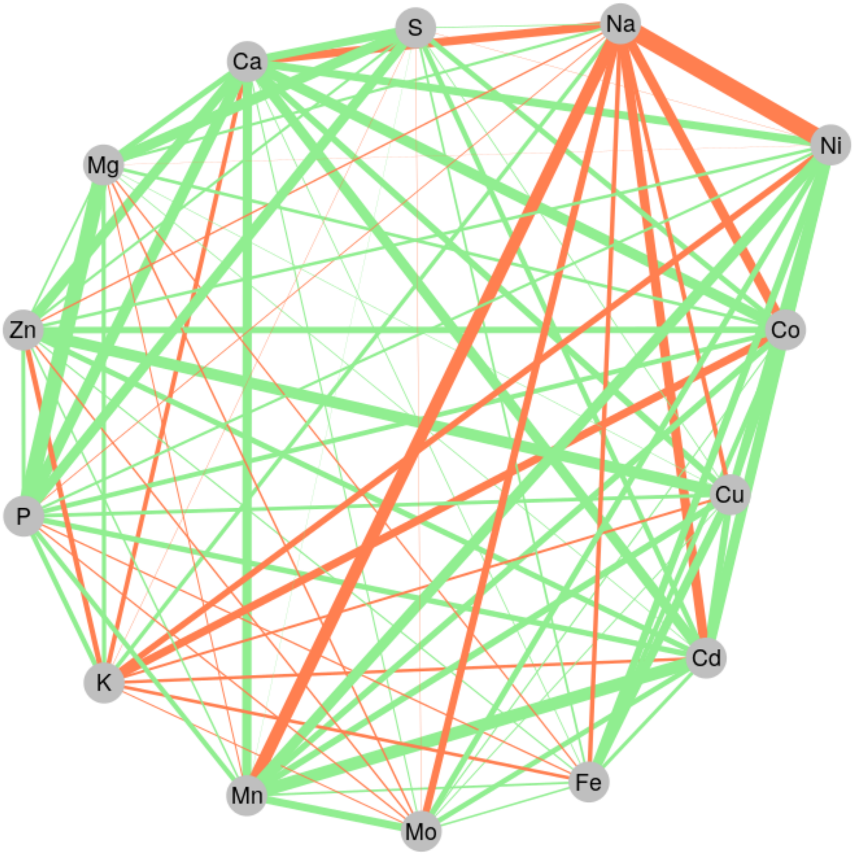

Figure 20: Explanotory analysis plots for gene clustering