Ionomics analysis for human data set using Ionflow

Wanchang Lin

2020-12-21

Contents

Data preparation

The human ionomics data set has been pre-processed. We need to get the symbolic data:

```
dat <- read.table("./test-data/human.csv", header = T, sep = ",")
dat <- dat[!duplicated(dat[, 1]), ]
colnames(dat)[1] <- "Line"
dat_symb <- symbol_data(x = dat, thres_symb = 2)
```

Some of ionomics data and symbolic data are like:

```
dat %>% head(10) %>%
  kable(caption = 'Ionomics data', digits = 2, booktabs = T) %>%
  kable_styling(full_width = F, font_size = 10,
                latex_options = c("striped", "scale_down"))
```

Table 1: Ionomics data

Line	As	B	Ca	Cd	Co	Cu	Fe	K	Li	Mg	Mn	Mo	Na	Ni	P	S	Se	Zn
AARS	0.66	1.38	1.79	-0.78	0.89	-1.45	0.58	0.50	0.78	0.01	-0.92	0.71	1.72	0.47	0.60	0.26	0.56	-0.24
AARSL	0.74	-0.27	0.52	-2.86	0.38	1.13	-0.45	0.65	2.63	-0.20	1.42	-3.81	-0.58	0.65	-2.61	1.74	0.07	-0.57
ABCB7	0.38	0.84	1.34	1.09	1.29	-0.82	-3.06	0.73	1.69	1.04	-0.67	-0.12	0.87	0.04	0.60	1.42	-0.49	1.41
ABCC10	1.03	0.32	1.20	1.19	1.79	-0.27	-0.38	-1.64	1.07	-1.67	1.79	1.64	0.89	1.22	-2.29	1.50	1.03	-1.92
ABCC1	-0.49	1.24	2.50	-2.50	2.06	-1.60	-0.95	-0.28	1.32	-1.64	0.06	2.26	1.53	-0.11	-2.81	0.62	3.50	-2.19
ABCC12	-0.06	-2.16	0.59	-3.02	-0.20	2.18	0.55	0.99	-0.19	1.72	1.19	0.14	-1.27	-2.84	2.18	-0.47	-2.60	0.60
ABCC13	1.32	1.87	1.12	-0.46	0.76	-0.97	-0.16	1.13	-1.02	0.87	0.13	0.23	1.85	1.65	-0.62	0.30	1.59	-2.41
ABCC11	-0.83	1.90	-0.97	1.74	-0.88	-1.86	-0.91	-3.49	-1.84	-2.29	-1.74	0.50	0.56	-0.17	-2.04	2.49	3.38	-2.10
ABCD1	0.60	1.10	0.30	-0.75	2.00	2.59	0.17	-1.03	0.15	-1.83	2.49	0.26	1.04	-0.41	-2.73	2.57	0.62	-3.25
ABCD2	-1.88	-1.89	-2.42	-1.58	-1.75	0.64	-2.22	1.75	-1.52	1.11	-2.08	-1.90	-2.26	-0.29	2.05	-5.12	-0.37	1.95

```
dat_symb %>% head(10) %>%
  kable(caption = 'Symbolic data', booktabs = T) %>%
  kable_styling(full_width = F, font_size = 10,
                latex_options = c("striped", "scale_down"))
```

Table 2: Symbolic data

Line	As	B	Ca	Cd	Co	Cu	Fe	K	Li	Mg	Mn	Mo	Na	Ni	P	S	Se	Zn
AARS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
AARSL	0	0	0	-1	0	0	0	0	1	0	0	-1	0	0	-1	0	0	0
ABCB7	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0
ABCC10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0
ABCC1	0	0	1	-1	1	0	0	0	0	0	0	1	0	0	-1	0	1	-1
ABCC12	0	-1	0	-1	0	1	0	0	0	0	0	0	0	-1	1	0	-1	0
ABCC13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1
ABCC11	0	0	0	0	0	0	0	-1	0	-1	0	0	0	0	-1	1	1	-1
ABCD1	0	0	0	0	1	1	0	0	0	0	1	0	0	0	-1	1	0	-1
ABCD2	0	0	-1	0	0	0	-1	0	0	0	-1	0	-1	0	1	-1	0	0

These data are filtered, i.e. remove all zero genes in symbolic data set:

```
idx <- rowSums(abs(dat_symb[, -1])) > 0
dat <- dat[idx, ]
dat_symb <- dat_symb[idx, ]
dim(dat)
#> [1] 710  19
```

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:

```
min <- 8
clust <- gene_clus(dat_symb[, -1], min_clust_size = min)
names(clust)
#> [1] "clus"    "idx"     "tab"     "tab_sub"
clust$tab_sub
#>   cluster nGenes
#> 1      19     10
#> 2      54      9
```

Gene network

The gene network uses both the ionomics and symbolic data. The similarity measures on ionomics data are used to construct the network. Before creating a network, these analyses are further filtered by:

- clustering of symbolic data;
- and the similarity threshold located between 0 and 1;

The methods implemented are: *pearson*, *spearman*, *kendall*, *cosine*, *mahal_cosine* or *hybrid_mahal_cosine*.

We use the Pearson correlation as similarity measure for network analysis:

```
net <- GeneNetwork(data = dat,
                   data_symb = dat_symb,
                   min_clust_size = min,
                   thres_corr = 0.6,
                   method_corr = "pearson")
```

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

```
net$plot.pnet1
```

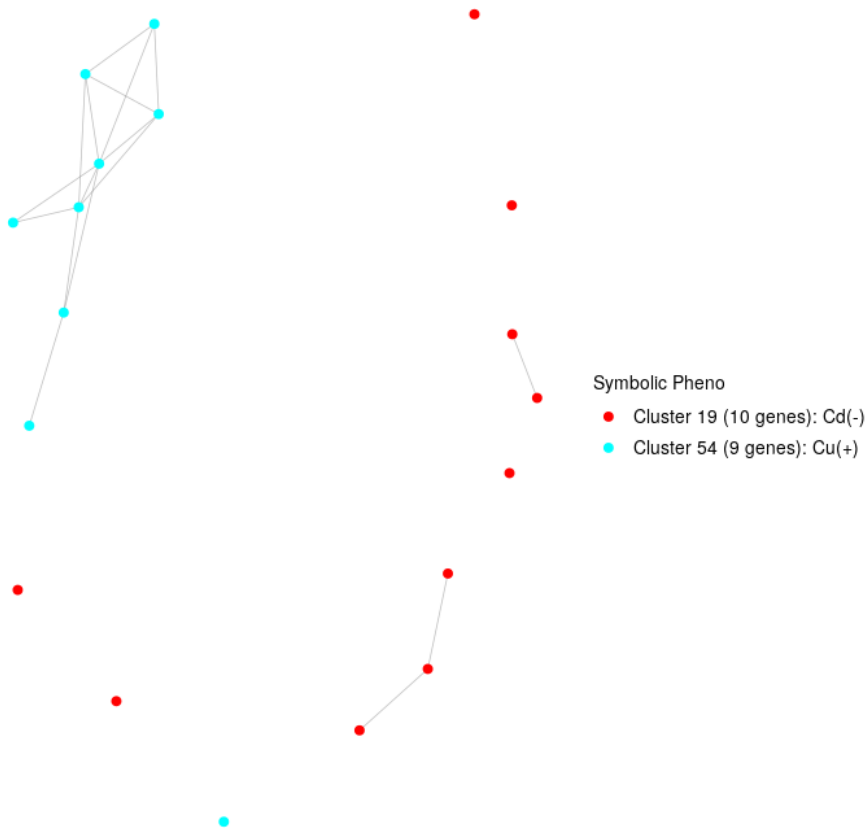

Figure 1: Network with Pearson correlation: symbolic clustering

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

```
net$plot.pnet2
```

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

```
net$plot.impact_betweenness
```

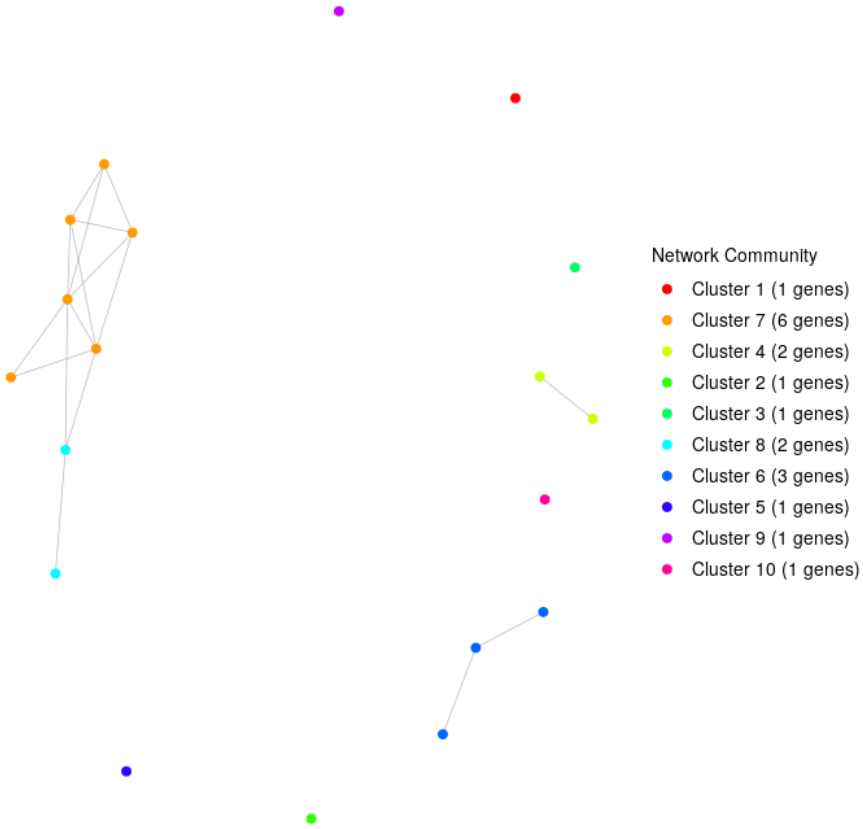

Figure 2: Network with Pearson correlation: community detction

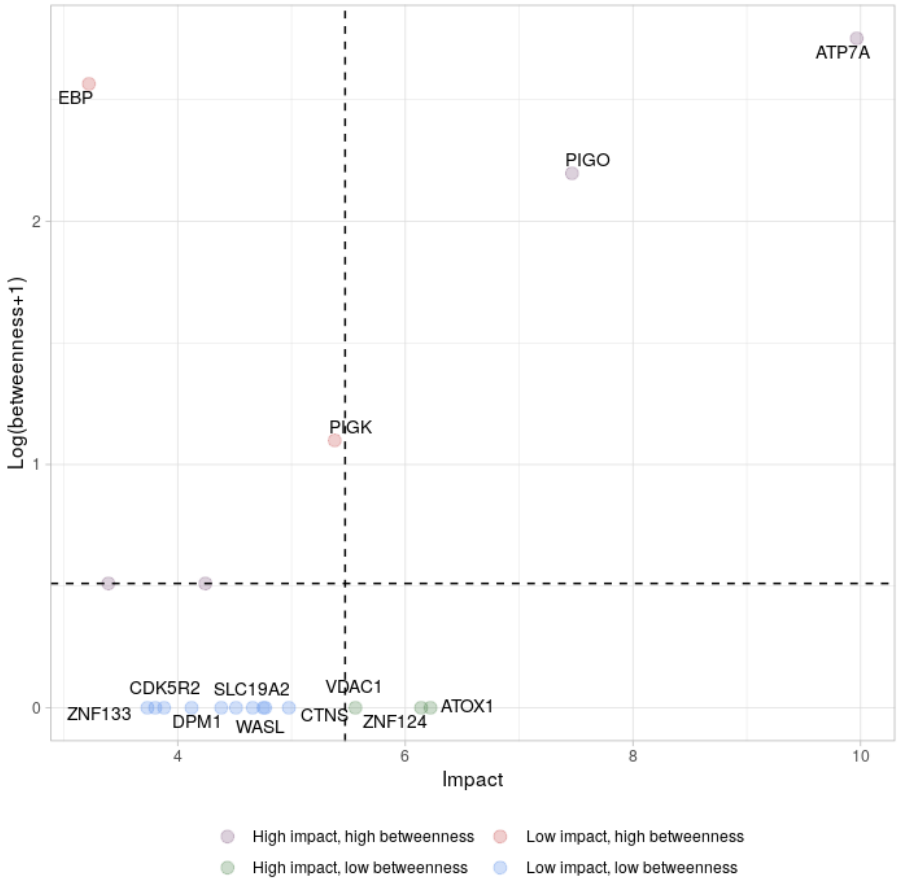

Figure 3: Network with Pearson correlation: impact and betweenness

For comparison purposes, we use *Mahalanobis Cosine*:

```
net_2 <- GeneNetwork(data = dat,
                     data_symb = dat_symb,
                     min_clust_size = min,
                     thres_corr = 0.6,
                     method_corr = "mahal_cosine")
net_2$plot.pnet1
```

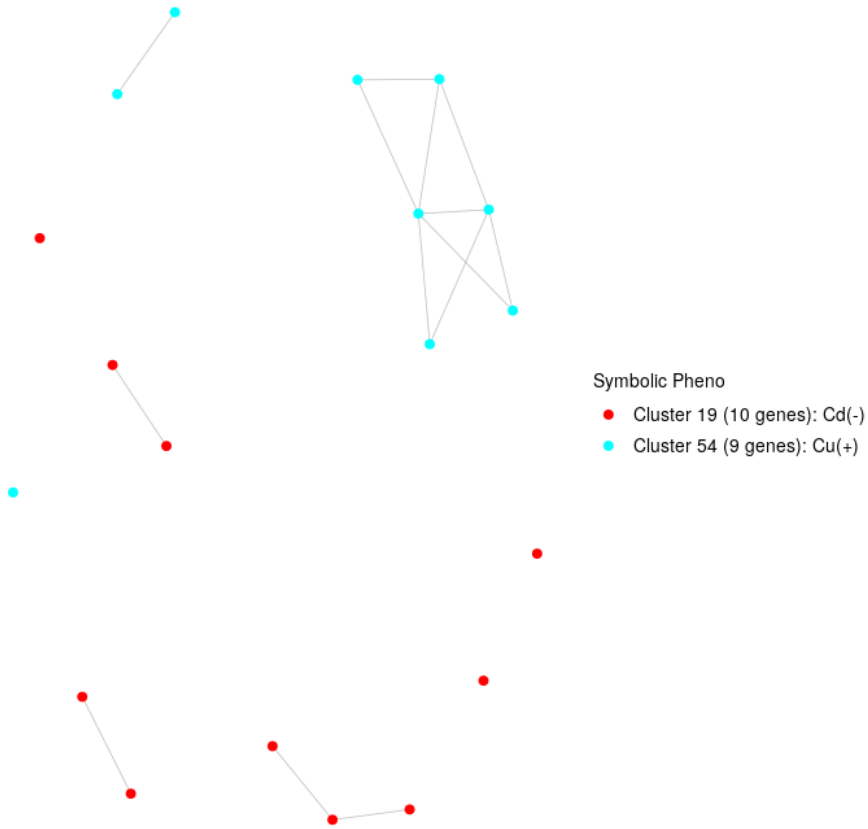

Figure 4: Network with Mahalanobis Cosine

```
net_2$plot.pnet2
```

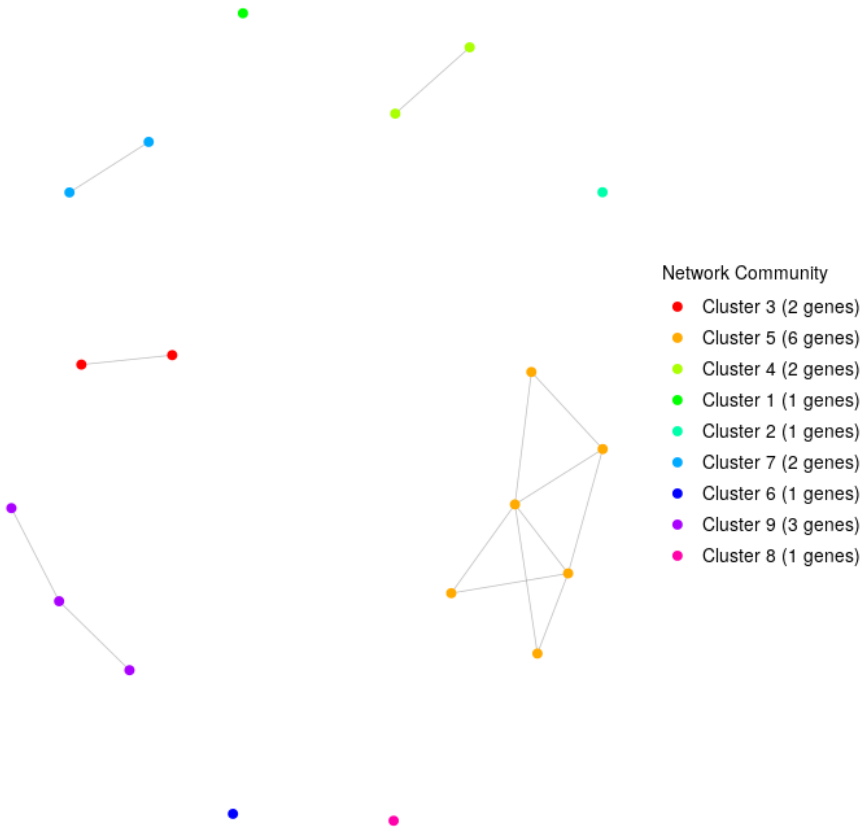

Figure 5: Network with Mahalanobis Cosine

Enrichment analysis

The enrichment analysis is for group data. The genes in the group are considered target gene sets while genes in the whole data set is the universal gene set.

The netowk analysis retunes a vertex attributes matrix:

```
head(net$net_node)
#>     Line                    symb_pheno         comm_centre
#> 1   AGXT Cluster 19 (10 genes): Cd(-) Cluster 1 (1 genes)
#> 2  ATOX1  Cluster 54 (9 genes): Cu(+) Cluster 7 (6 genes)
#> 3  ATP7A  Cluster 54 (9 genes): Cu(+) Cluster 7 (6 genes)
#> 4 CDK5R2 Cluster 19 (10 genes): Cd(-) Cluster 4 (2 genes)
#> 5   CTNS Cluster 19 (10 genes): Cd(-) Cluster 2 (1 genes)
#> 6   DPM1  Cluster 54 (9 genes): Cu(+) Cluster 3 (1 genes)
```

The second and third columns are symbolic clustering and network community cluster, respectively.

If we perform enrichment analysis on the network community centre, the matrix should include the first column (gene IDs) and the third colum:

```
mat <- net$net_node[, c(1,3)]
kegg <- kegg_enrich(mat = mat, pval = 0.05, annot_pkg =  "org.Hs.eg.db")

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

Table 3: KEGG enrichment analysis

comm_centre	KEGGID	Pvalue	Count	Size	Term

Note that there could be no results returned for KEGG enrichment analysis.

The GO Terms enrichment analysis with ontology of *BP* (other two are *MF* and *CC*):

```
go <- go_enrich(mat = mat, pval = 0.05, ont = "BP", annot_pkg =  "org.Hs.eg.db")
#' go
dim(go)
#> [1] 0 7
go %>%
  kable(caption = 'GO Terms enrichment analysis',
```

```
        digits = 3, booktabs = T) %>%
  kable_styling(full_width = F, font_size = 10,
                latex_options = c("striped", "scale_down"))
```

Table 4: GO Terms enrichment analysis

comm_centre	ID	Description	Pvalue	Count	CountUniverse	Ontology

We can also perform enrichment analysis on the symbolic clustering. To do so, use the first and second columns:

```
mat <- net$net_node[, c(1,2)]
kegg <- kegg_enrich(mat = mat, pval = 0.05, annot_pkg =  "org.Hs.eg.db")

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

Table 5: KEGG enrichment analysis

symb_pheno	KEGGID	Pvalue	Count	Size	Term

Note that there could be no results returned for KEGG enrichment analysis.

The GO Terms enrichment analysis with ontology of *BP* (other two are *MF* and *CC*):

```
go <- go_enrich(mat = mat, pval = 0.05, ont = "BP", annot_pkg =  "org.Hs.eg.db")
#' go
dim(go)
#> [1] 0 7
go %>%
  kable(caption = 'GO Terms enrichment analysis',
        digits = 3, booktabs = T) %>%
  kable_styling(full_width = F, font_size = 10,
                latex_options = c("striped", "scale_down"))
```

Table 6: GO Terms enrichment analysis

symb_pheno	ID	Description	Pvalue	Count	CountUniverse	Ontology

Exploratory analysis

The explanatory analysis performs PCA and correlation analysis for ions in terms of genes. Note that this analysis treats ions as samples/replicates while genes are treated as variables/features. The explanatory analysis is initially employed at an early stage of the analysis.

We apply it to the pre-processed data `dat` before any other analysis:

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

The PCA plot is:

```
expl$plot.pca
```

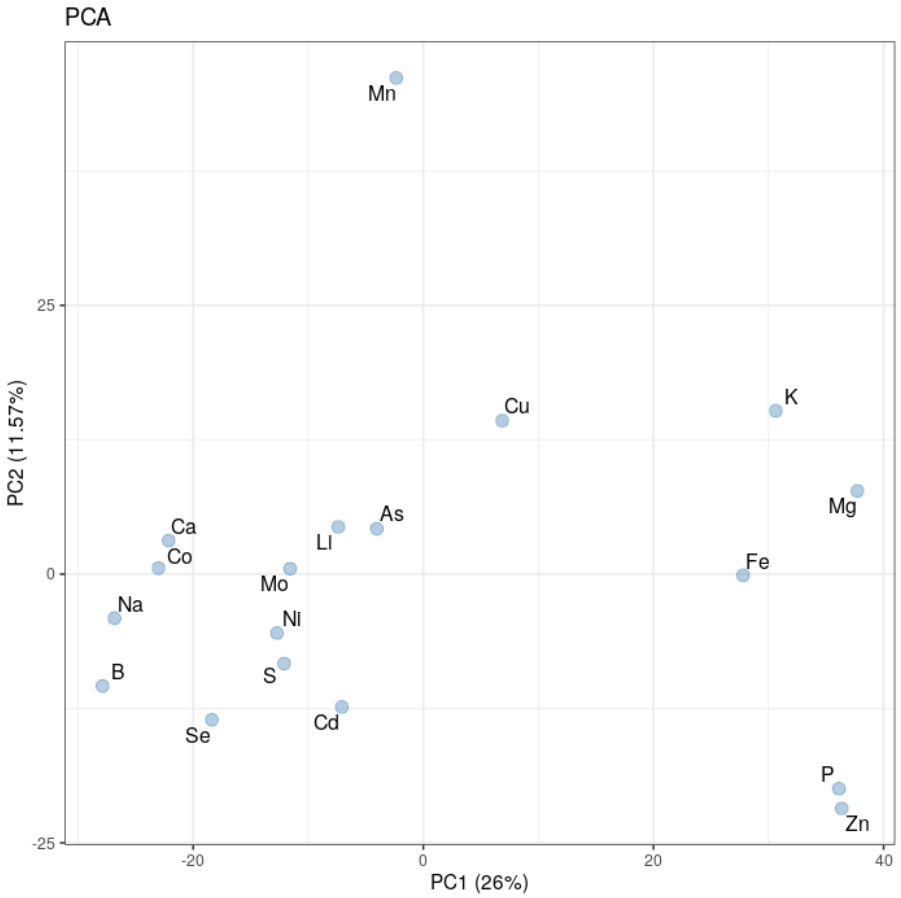

Figure 6: Ion PCA plot on pre-processed data

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

```
expl$plot.corr
```

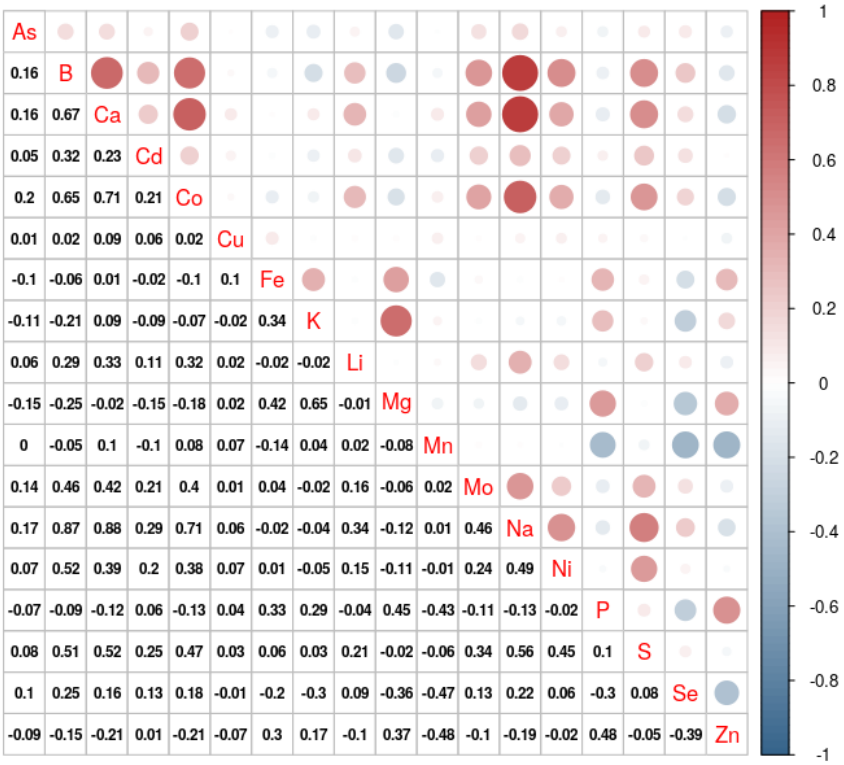

Figure 7: Ion correlation plots on pre-processed data

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

```
expl$plot.net
```

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

```
expl$plot.heat
```

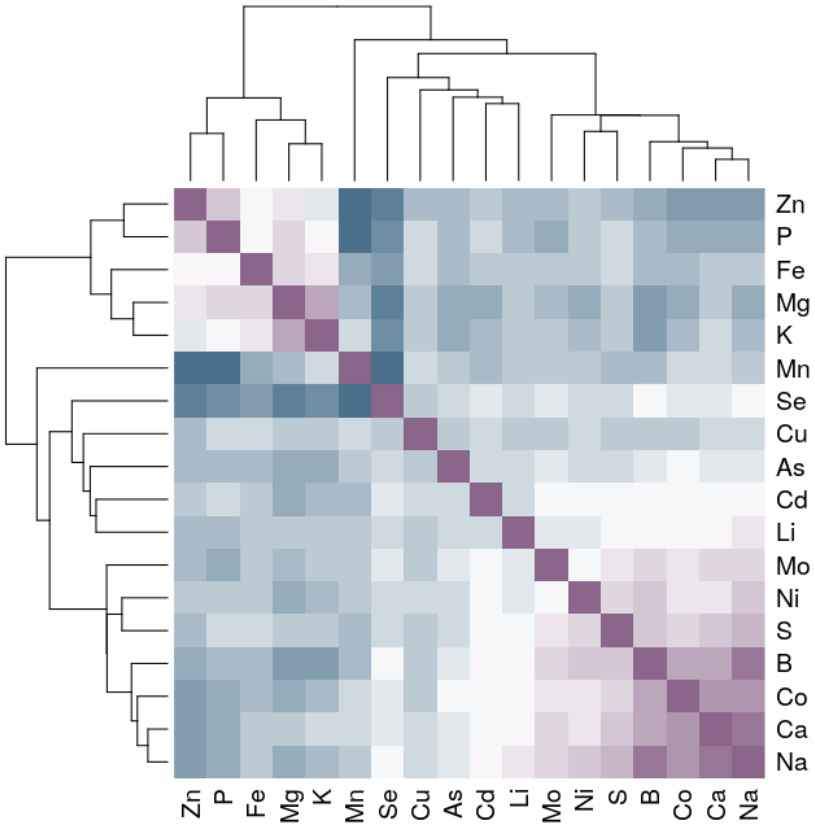

Figure 8: Ion correlation plots on pre-processed data

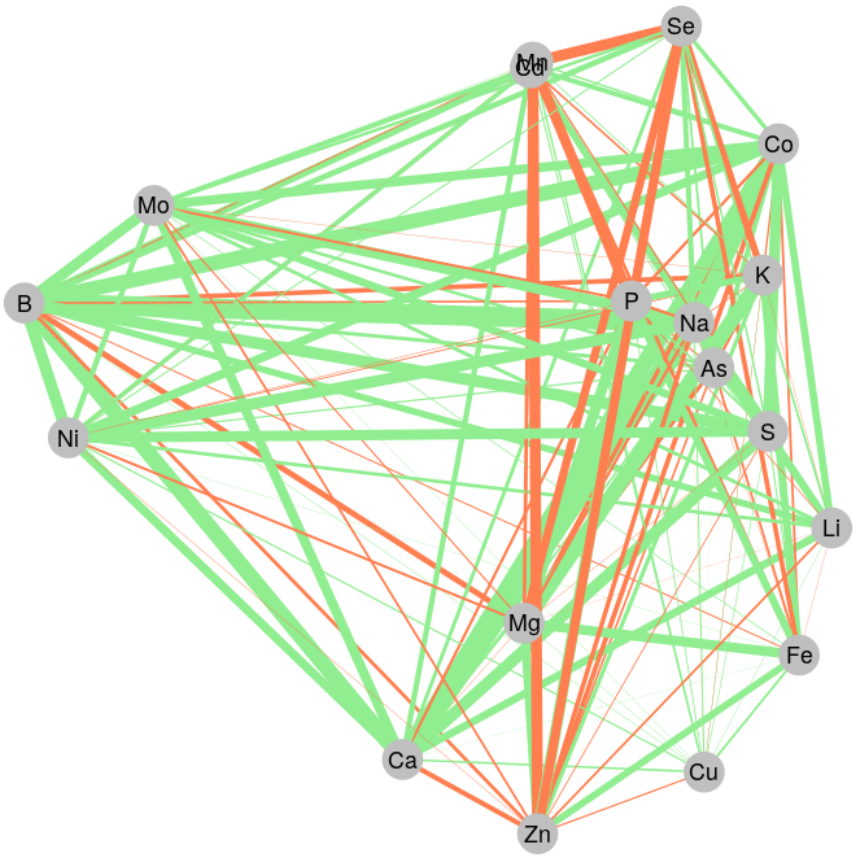

Figure 9: Ion correlation plots on pre-processed data

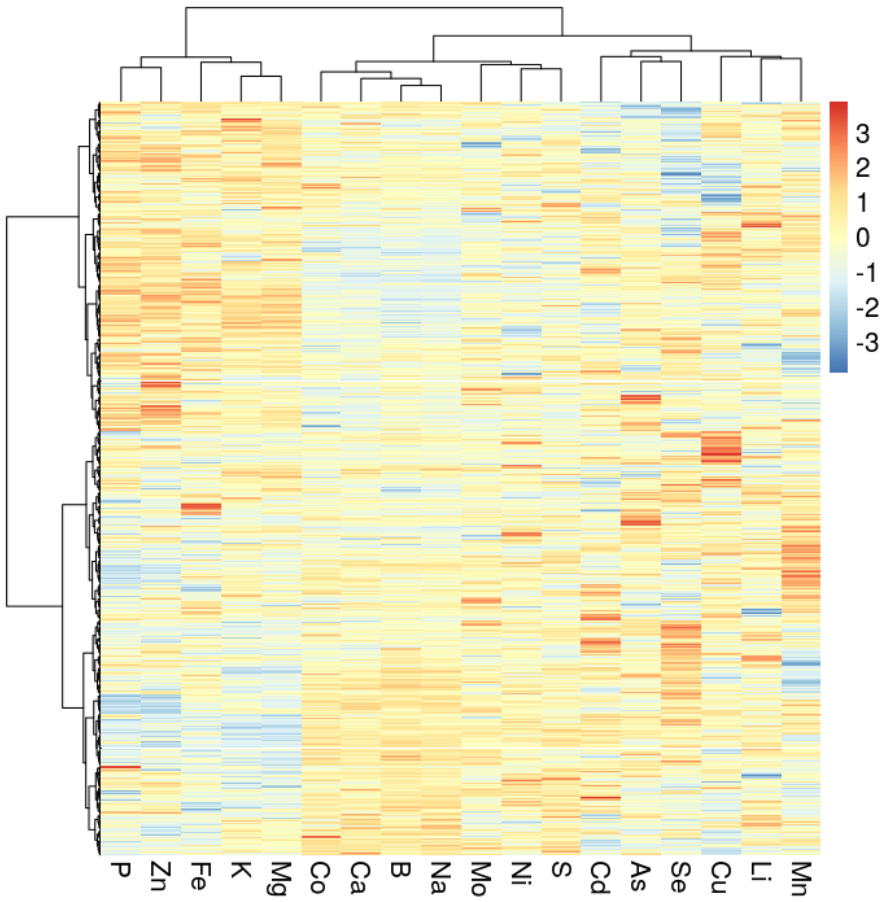

Figure 10: Correlation between ions and genes on pre-processed data