```r
library(MASS)
library(reshape2)
library(dplyr)
library(ggcorrplot)
library(tidyverse)
library(corrr)
library(igraph)
library(ggraph)
```

Modify to use Spearman correlation

```r
trace(cor_pmat,edit=TRUE)
#tmp <- stats::cor.test(mat[, i], mat[, j], method = "spearman", exact=FALSE)
```

Modify to exclude 0 before decimal point

```r
trace(ggcorrplot,edit=TRUE)
#label <- sub("^(-?)0.", "\\1.", sprintf("%.2f", round(corr[, "value"], 2)))
```

Generate an arbitrary covariance matrix
https://stats.stackexchange.com/q/215505

```r
set.seed(123)
n <- 26 # number of samples
A <- matrix(runif(n^2)*2-1, ncol=n)
Sigma <- t(A) %*% A
```

Generate correlated data

```r
data1 <- as.data.frame(MASS::mvrnorm(n = 82, mu = c(runif(n, min = 2, max = 3
0)),
                                     Sigma = Sigma, empirical = TRUE))
colnames(data1) <- c(sprintf("miR-%d",seq(1:13)), sprintf("lncRNA-%d",seq(1:1
3)))
```

Compute a correlation matrix

```r
c.matrix <- cor(data1, method = "s", use = "pairwise.complete.obs")
```

Compute a matrix of correlation p-values

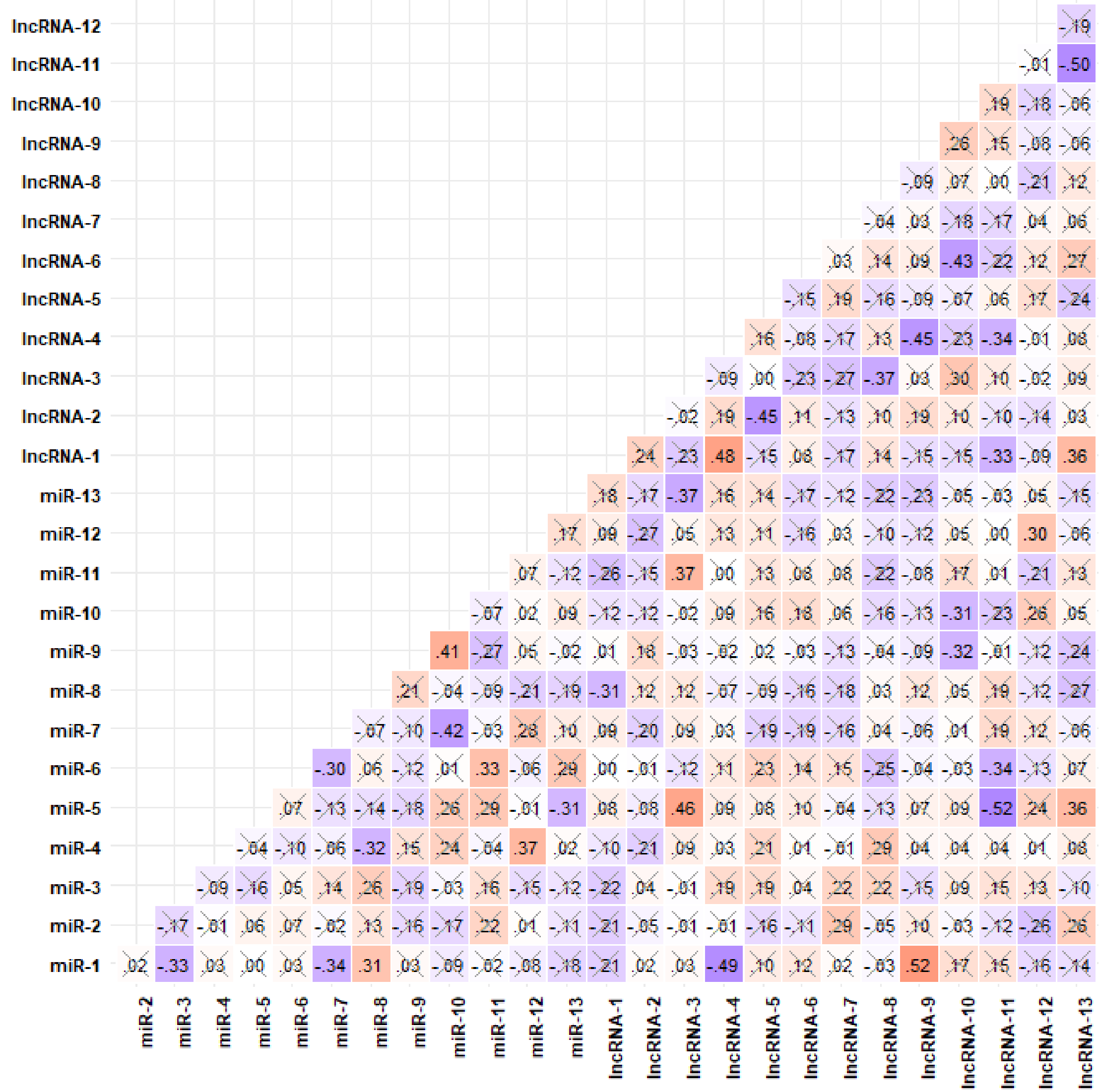
```r
p.mat <- cor_pmat(data1)
```

Adjust p-values using Benjamini & Hochberg (1995) correction

```r
p.mat2 <- matrix(p.adjust(as.vector(as.matrix(p.mat)), method = "BH"),
                 ncol = ncol(p.mat))
colnames(p.mat2) <- colnames(p.mat)
row.names(p.mat2) <- row.names(p.mat)
```

Display correlation matrix

```
ggcorrplot(c.matrix, type = "lower", lab = TRUE,lab_size = 3,lab_col = "black
",
           outline.col = "white", p.mat = p.mat2, sig.level = 0.05, pch.col =
"#999999") +
  theme(axis.text.y = element_text(colour = "black", size = 8, face = "bold")
,
        axis.text.x = element_text(colour = "black", size = 8, angle = 90, fa
ce = "bold"),
        legend.position = "none")
```



Create a tidy data frame of correlations

```
tidy_cors <- data1 %>% correlate(method = "spearman") %>% stretch()

##
## Correlation method: 'spearman'
## Missing treated using: 'pairwise.complete.obs'
```

Convert stronger correlations to an undirected graph object

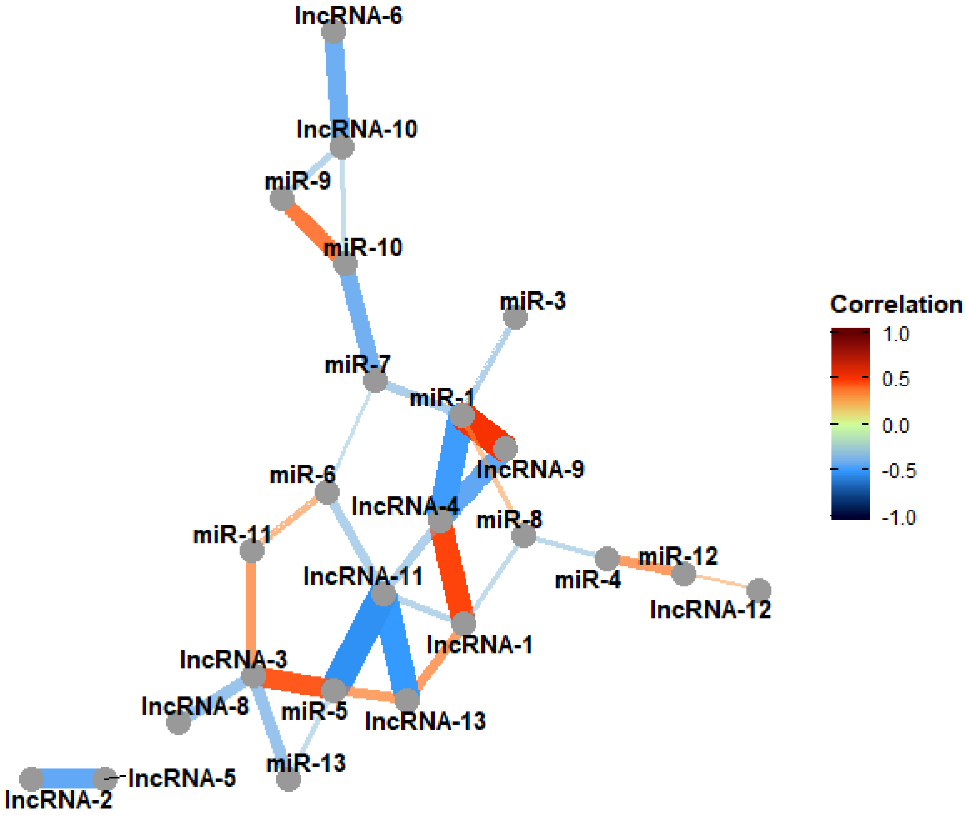
```
graph_cors <- tidy_cors %>% filter(abs(r) > .3)
```

Filter to remove non-significant correlation from the correlation network

```
p.mat.filter <- melt(p.mat2) %>% filter(value < .05) %>% as.data.frame()
colnames(p.mat.filter) <- c("x", "y", "value")
graph_cors_merged <- merge(p.mat.filter, graph_cors)
graph_cors_merged$value <- NULL
graph_cors_merged <- graph_cors_merged %>% graph_from_data_frame(directed = F
ALSE)
```

Display correlation network
https://drsimonj.svbtle.com/how-to-create-correlation-network-plots-with-corrr-and-ggraph

```
ggraph(graph_cors_merged) +
  geom_edge_link(aes(edge_alpha = abs(r), edge_width = abs(r), color = r)) +
  guides(edge_alpha = "none", edge_width = "none") +
  scale_edge_colour_gradientn("Correlation", limits = c(-1, 1),
                              colors = c("#000033", "#3399FF", "#CCFF99", "#F
F3300", "#660000")) +
  geom_node_point(color = "#999999", size = 5) +
  geom_node_text(aes(label = name), repel = TRUE, fontface = "bold") +
  theme_graph() + theme(legend.title = element_text(face = "bold")) +
  scale_edge_alpha(range = c(0.3, 1))
```


lncRNA-6
lncRNA-10
miR-9
miR-10
miR-3
miR-7
miR-1
lncRNA-9
miR-6
lncRNA-4
miR-8
miR-11
lncRNA-11
miR-4
miR-12
lncRNA-12
lncRNA-1
lncRNA-3
lncRNA-8
miR-5
lncRNA-13
miR-13
lncRNA-5
lncRNA-2
Correlation
1.0
0.5
0.0
-0.5
-1.0