DSA 8070 R Session 1: Characterizing and Displaying Multivariate Data

Whitney

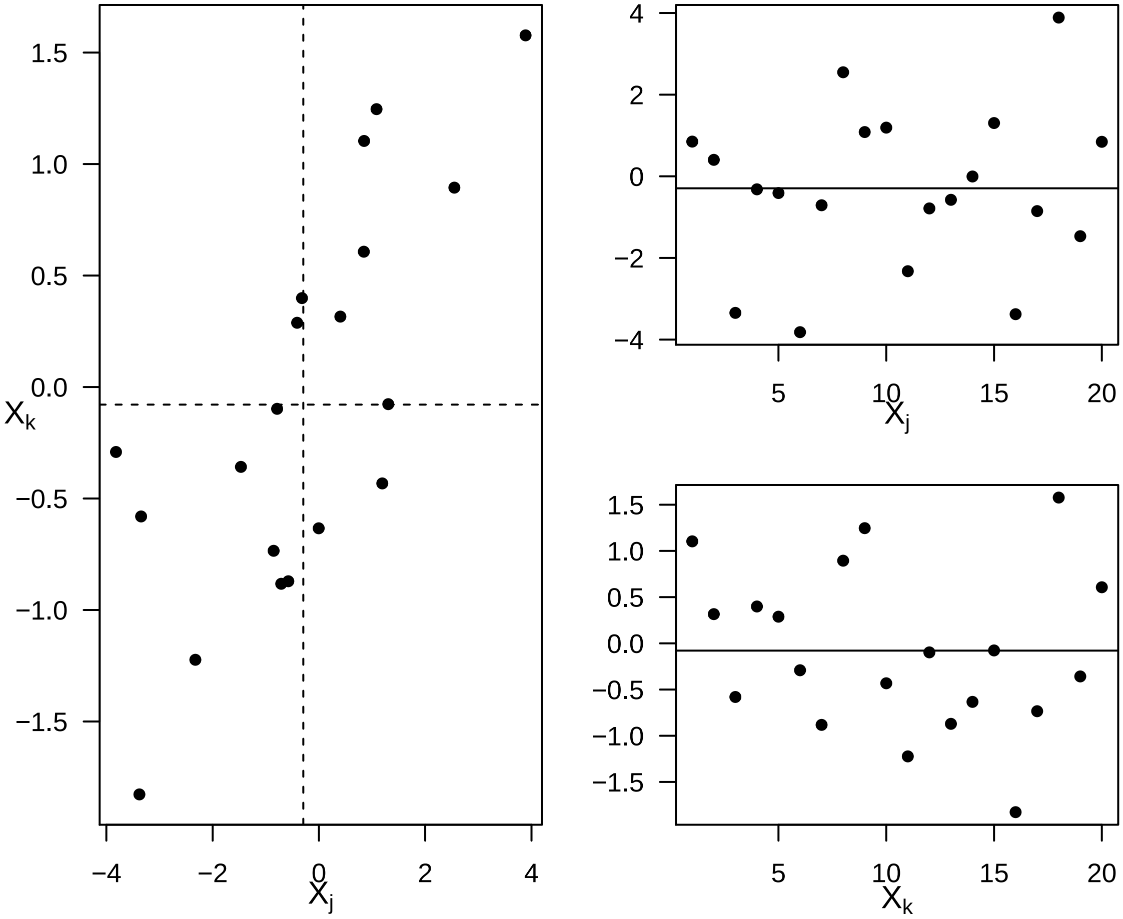
August 24, 2021

Contents

Descriptive Statistics

Sample covariance visualization

```r
set.seed(123)
library(MASS)
dat <- mvrnorm(n = 20, mu = c(0, 0), Sigma = matrix(c(4, 1.4, 1.4, 1), 2))
n <- dim(dat)[1]
par(mar = c(3.6, 3.6, 0.8, 0.6), las = 1)
layout(matrix(c(1, 1, 2, 3), nrow = 2, ncol = 2))
plot(dat, pch = 16, las = 1, xlab = "", ylab = "")
mtext(expression(X[j]), 1, line = 2); mtext(expression(X[k]), 2, line = 2)
text(-4, 2, expression(paste(S[jk], " = ")))
text(-3.3, 2, round(cov(dat[, 1], dat[, 2]), 2))
abline(h = mean(dat[, 2]), lty = 2); abline(v = mean(dat[, 1]), lty = 2)
plot(1:n, dat[, 1], pch = 16, xlab = "", ylab = "")
abline(h = mean(dat[, 1]))
mtext(expression(X[j]), 1, line = 2)
plot(1:n, dat[, 2], pch = 16, xlab = "", ylab = "")
abline(h = mean(dat[, 2]))
mtext(expression(X[k]), 1, line = 2)
```



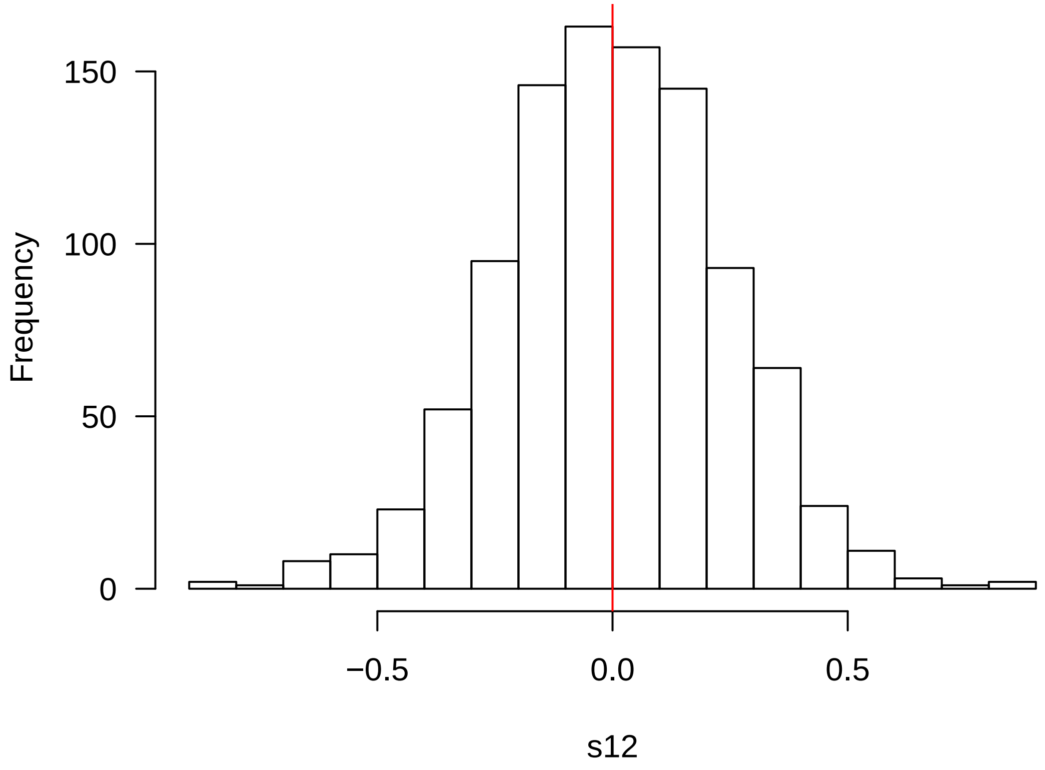
Sample and population covariance

Here we simulate data with size sample $n = 20$ from a bivariate normal distribution with *population covariance* $\rho_{12} = 0$. We calculate the *sample covariance* s_{12} for each simulated data set, and we repeat this process 1,000 times.

The main purpose of this exercise is to demonstrate that one can conduct *Monte Carlo* experiment to approximate the *sampling distribution* of s_{12}.

```r
dat <- replicate(1000, mvrnorm(n = 20, mu = c(0, 0), Sigma = matrix(c(1, 0, 0, 1), 2)))

s12 <- apply(dat, 3, function(x) cov(x[, 1], x[, 2]))
hist(s12, 20, las = 1, main = "")
abline(v = 0, col = "red")
```



Bivariate Data Example

```
data <- cbind(x1 = c(42, 52, 88, 58, 60), x2 = c(4, 5, 7, 4, 5))
(means <- apply(data, 2, mean))
```

```
## x1 x2
## 60  5
```

```
cov(data)
```

```
##     x1   x2
## x1 294 19.0
## x2  19  1.5
```

```
cor(data)
```

```
##           x1        x2
## x1 1.0000000 0.9047619
## x2 0.9047619 1.0000000
```

Generliazed Variance

```
data(mtcars)
vars <- which(names(mtcars) %in% c("mpg", "disp", "hp", "drat", "wt"))
car <- mtcars[, vars]; S <- cov(car)
(genVar <- det(S))
```

```
## [1] 3951786
```

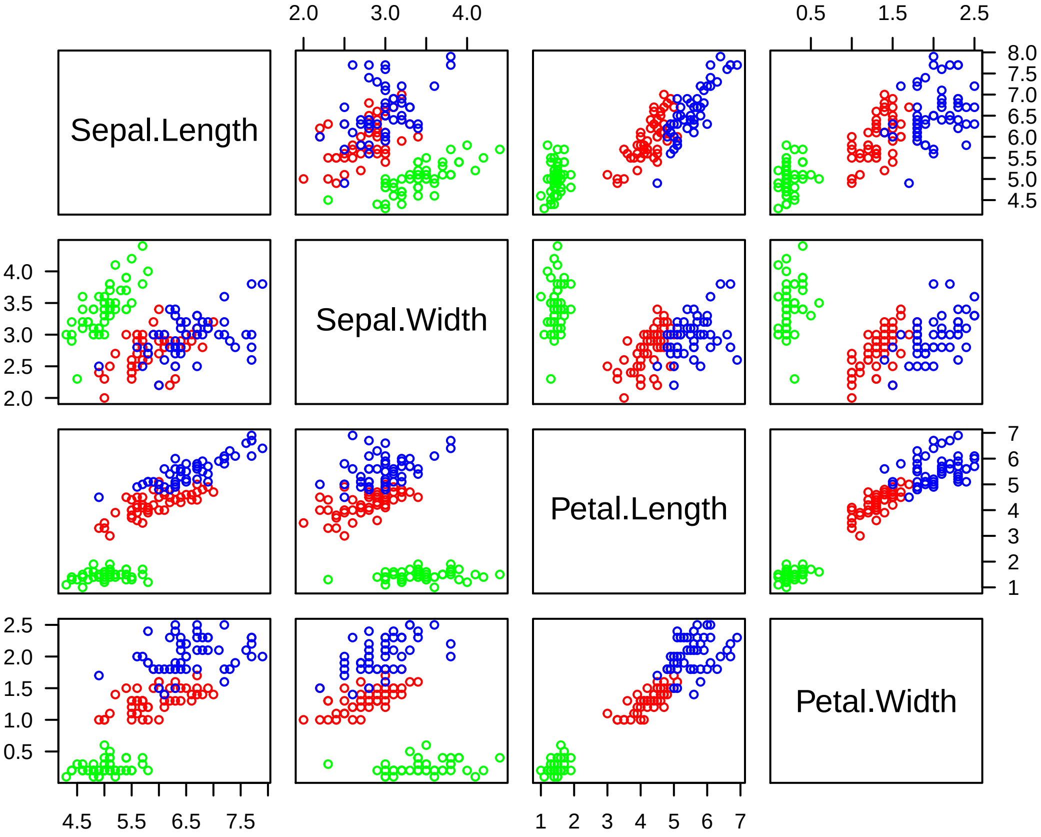
Graphs and Visualization

pairs

```
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```
pairs(iris[, -5], las = 1, col = rep(c("green", "red", "blue"), each = 50), cex = 0.8)
```

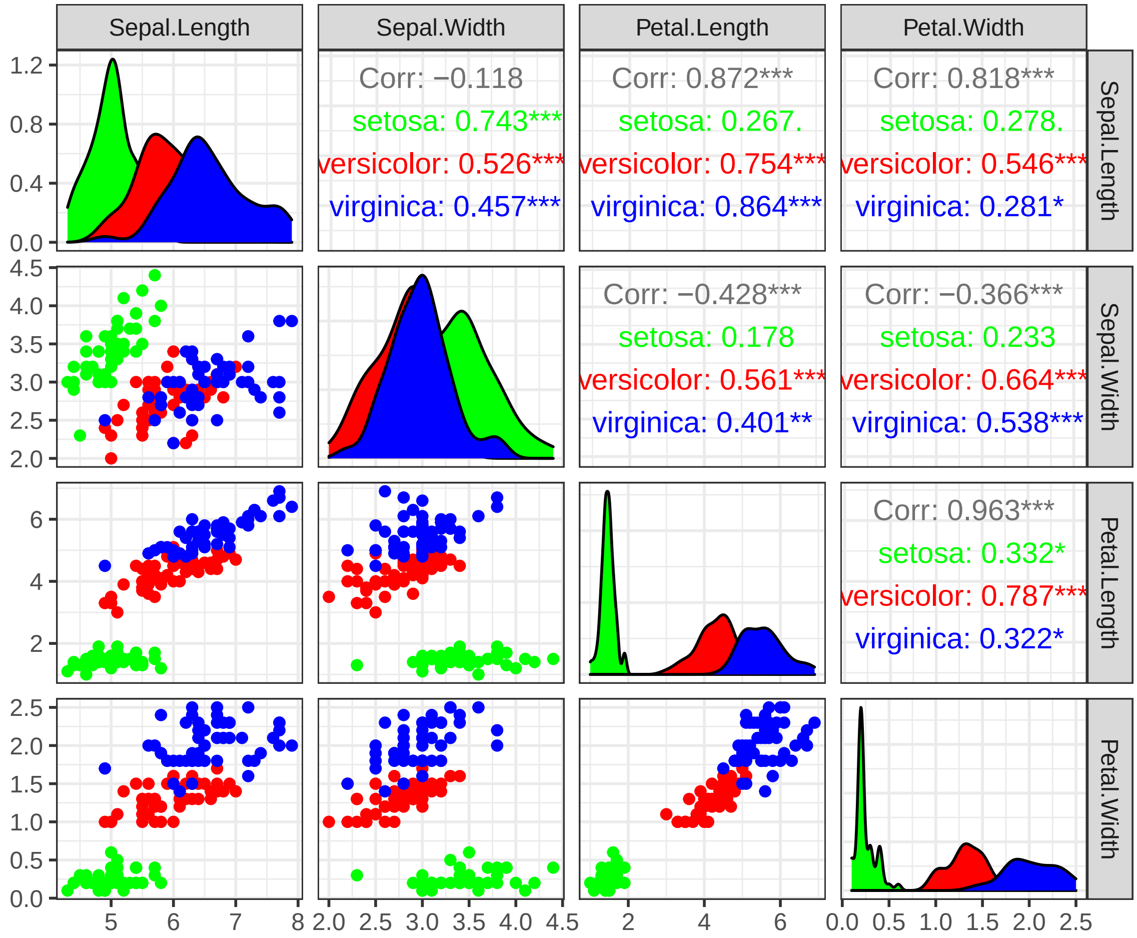


ggpairs

```
library(GGally)
```

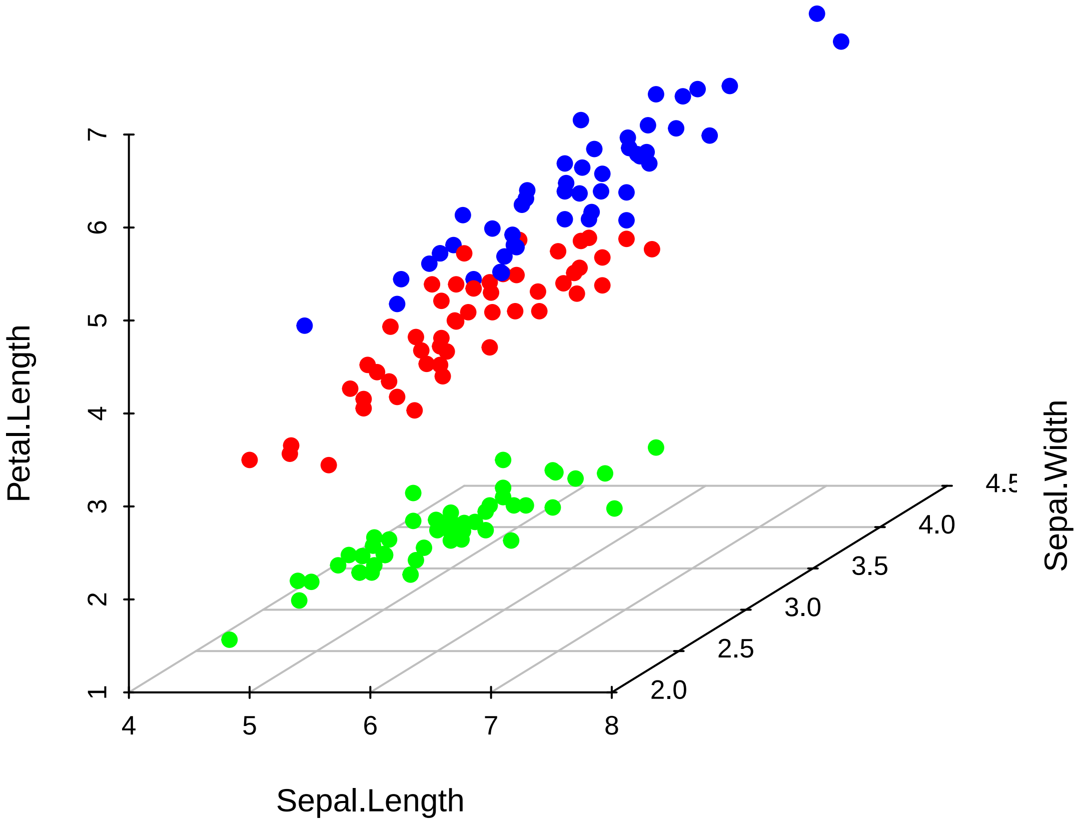
```
## Loading required package: ggplot2
```

```
library(ggplot2)
p <- ggpairs(iris[, -5], aes(color = iris$Species)) + theme_bw()
# Change color manually.
# Loop through each plot changing relevant scales
for(i in 1:p$nrow) {
  for(j in 1:p$ncol){
    p[i, j] <- p[i, j] +
        scale_fill_manual(values = c("green", "red", "blue")) +
        scale_color_manual(values = c("green", "red", "blue"))
  }
}
p
```



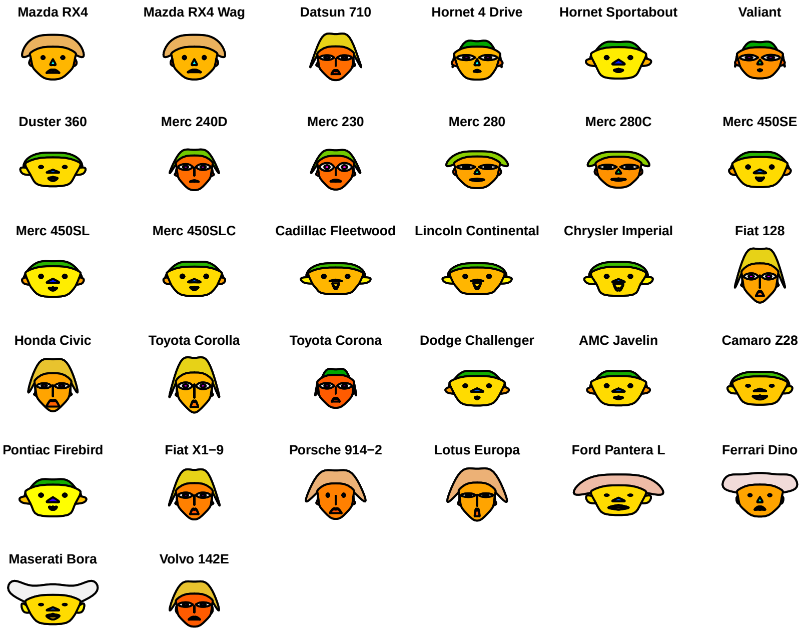
3D Scatter Plot

```
library(scatterplot3d)
scatterplot3d(iris[, 1:3], pch = 19, color = rep(c("green", "red", "blue"), each = 50), grid = TRUE, bo
```



Chernoff Faces

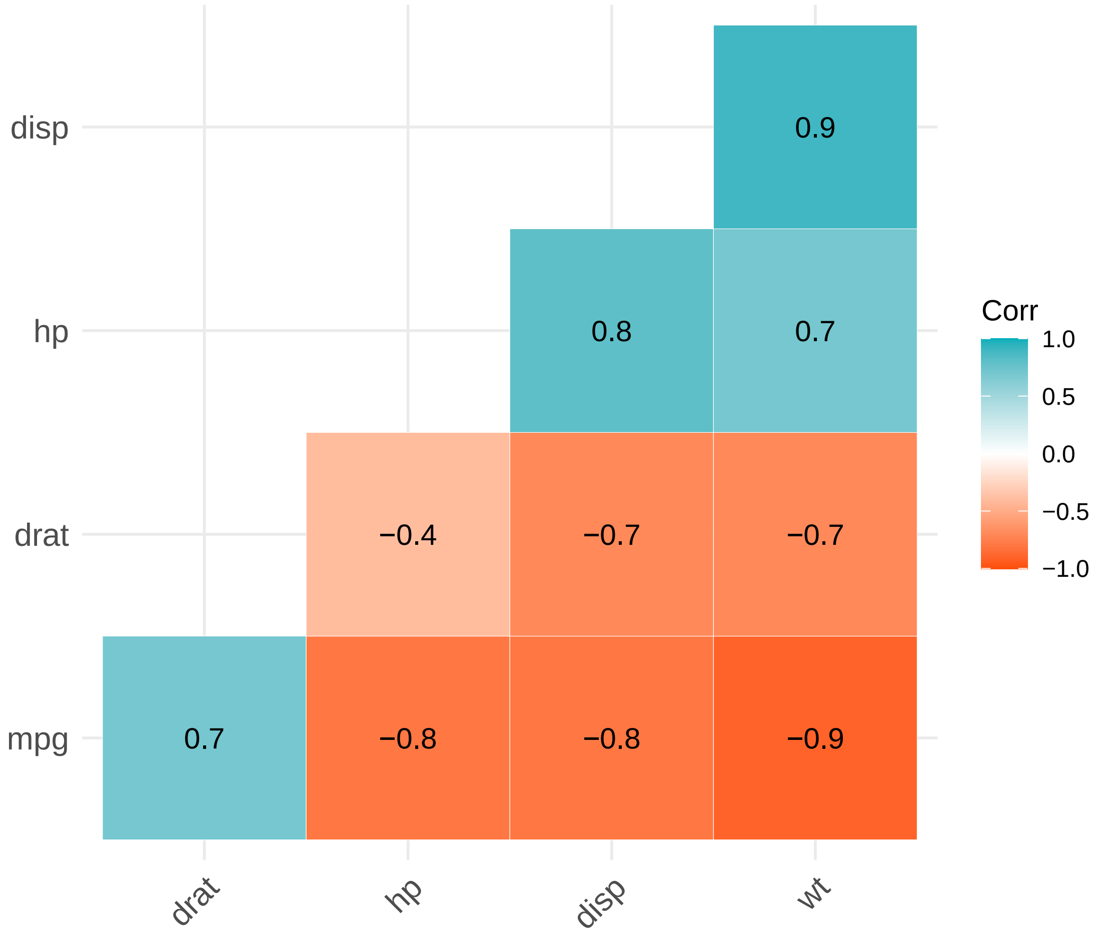
```r
library(aplpack)
par(mar = rep(0, 4))
faces(mtcars, cex = 0.6)
```



```
## effect of variables:
##  modified item       Var
##  "height of face   " "mpg"
##  "width of face    " "cyl"
##  "structure of face" "disp"
##  "height of mouth  " "hp"
##  "width of mouth   " "drat"
##  "smiling          " "wt"
##  "height of eyes   " "qsec"
##  "width of eyes    " "vs"
##  "height of hair   " "am"
##  "width of hair   "  "gear"
##  "style of hair   "  "carb"
##  "height of nose  "  "mpg"
##  "width of nose   "  "cyl"
##  "width of ear    "  "disp"
##  "height of ear   "  "hp"
```

Visualizing Summary Statistics

```r
library(ggcorrplot)
# Compute a correlation matrix
corr <- round(cor(car), 1)
# Visualize
ggcorrplot(corr, p.mat = cor_pmat(car),
           hc.order = TRUE, type = "lower",
           color = c("#FC4E07", "white", "#00AFBB"),
           outline.col = "white", lab = TRUE)
```


disp
hp
drat
mpg
0.9
0.8
0.7
−0.4
−0.7
−0.7
0.7
−0.8
−0.8
−0.9
drat
hp
disp
wt
Corr
1.0
0.5
0.0
−0.5
−1.0