DSA 8070 R Session 1: Characterizing and Displaying Multivariate Data

Whitney

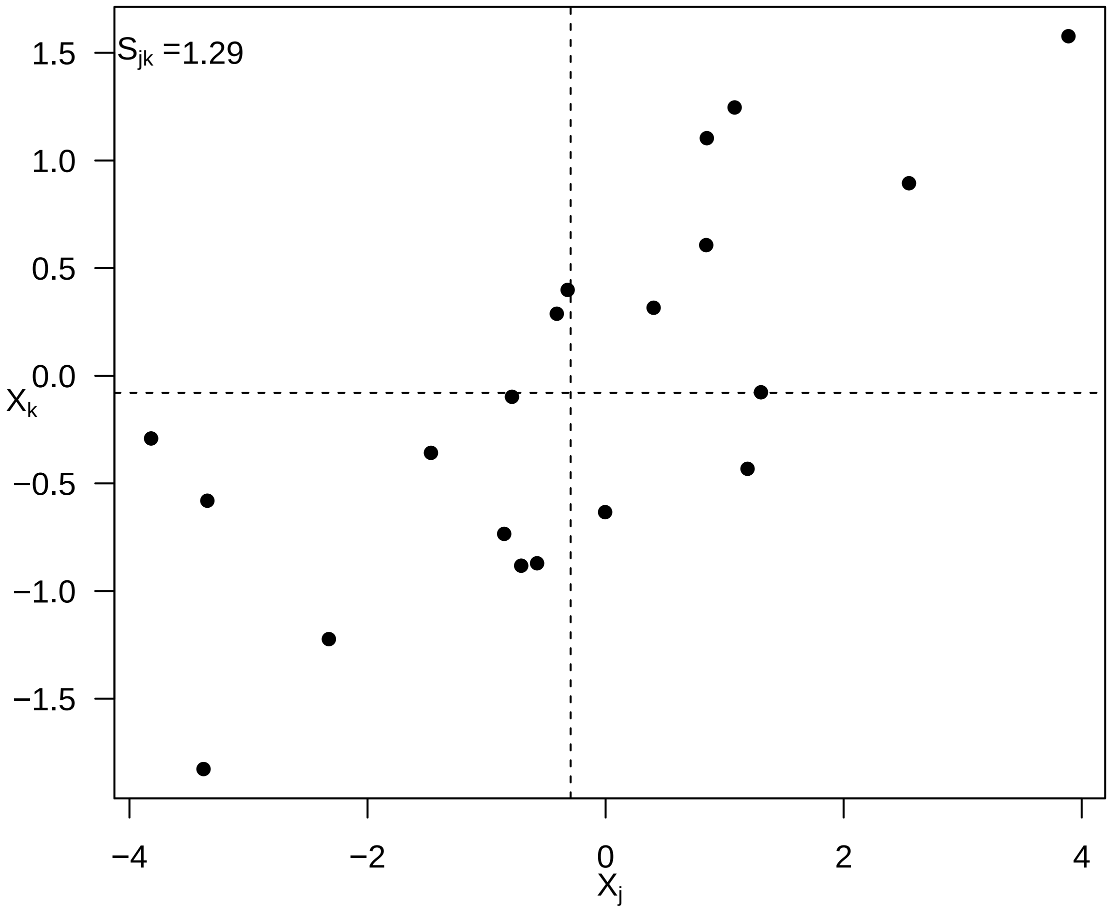
Contents

Descriptive Statistics

Sample covariance visualization

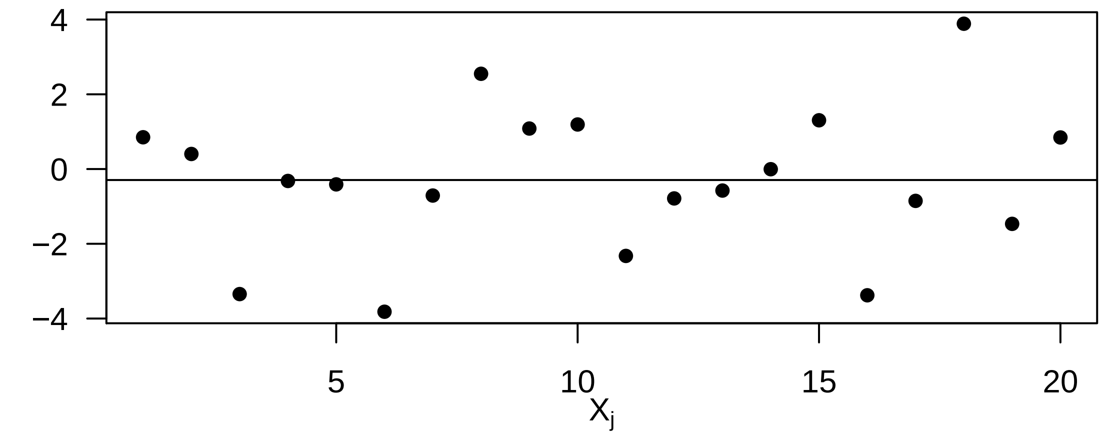
Here, we simulate a bivariate dataset that follows a bivariate normal distribution with a mean vector of $(0, 0)^T$ and variances of 4 and 1, respectively. Furthermore, these two variables are positively correlated with a population covariance of 1.4 (resulting in a population correlation of $\frac{1.4}{\sqrt{4}\times\sqrt{1}} = 0.7$). We will use a scatterplot to visualize the covariance.

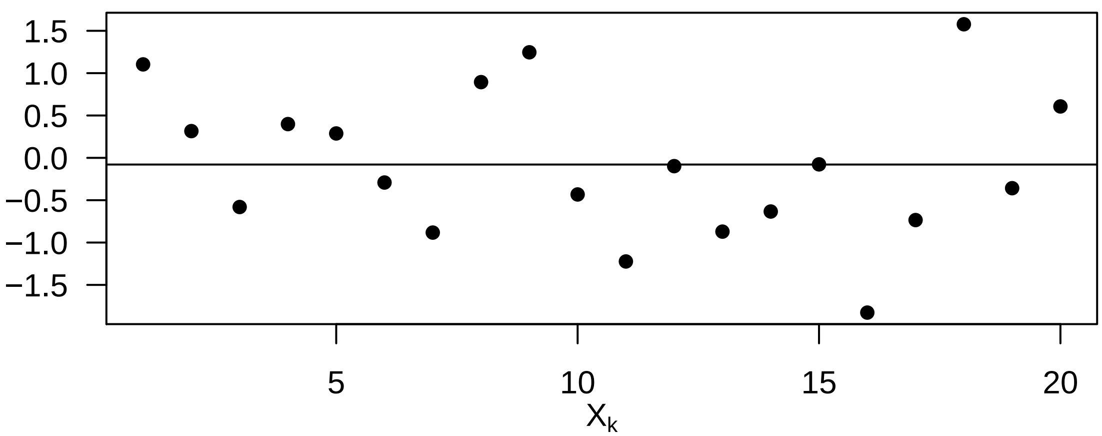
```
set.seed(123)
library(MASS)
dat <- mvrnorm(n = 20, mu = c(0, 0), Sigma = matrix(c(4, 1.4, 1.4, 1), 2))
n <- dim(dat)[1]
par(mar = c(3.6, 3.6, 0.8, 0.6), las = 1)
plot(dat, pch = 16, las = 1, xlab = "", ylab = "")
mtext(expression(X[j]), 1, line = 2); mtext(expression(X[k]), 2, line = 2)
text(-3.8, 1.5, expression(paste(S[jk], " = ")))
text(-3.3, 1.5, round(cov(dat[, 1], dat[, 2]), 2))
abline(h = mean(dat[, 2]), lty = 2); abline(v = mean(dat[, 1]), lty = 2)
```



We can also create two side-by-side run plots (i.e., plot data by order) to visualize the co-movement.

```
par(mfrow = c(2, 1), mar = c(3.6, 3.6, 0.8, 0.6), las = 1)
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)
```



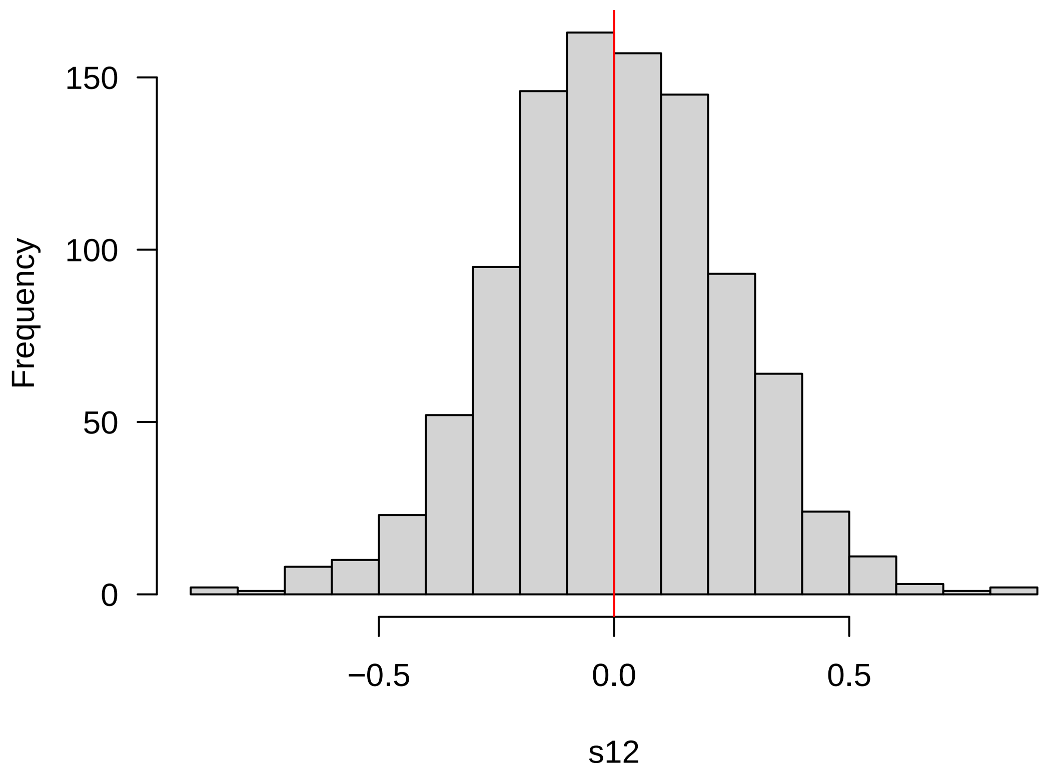


Sample and population covariance

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

The main purpose of this exercise is to demonstrate that one can conduct a *Monte Carlo* experiment to approximate the *sampling distribution* of s_{12} when two variables are independent to each other.

```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
```

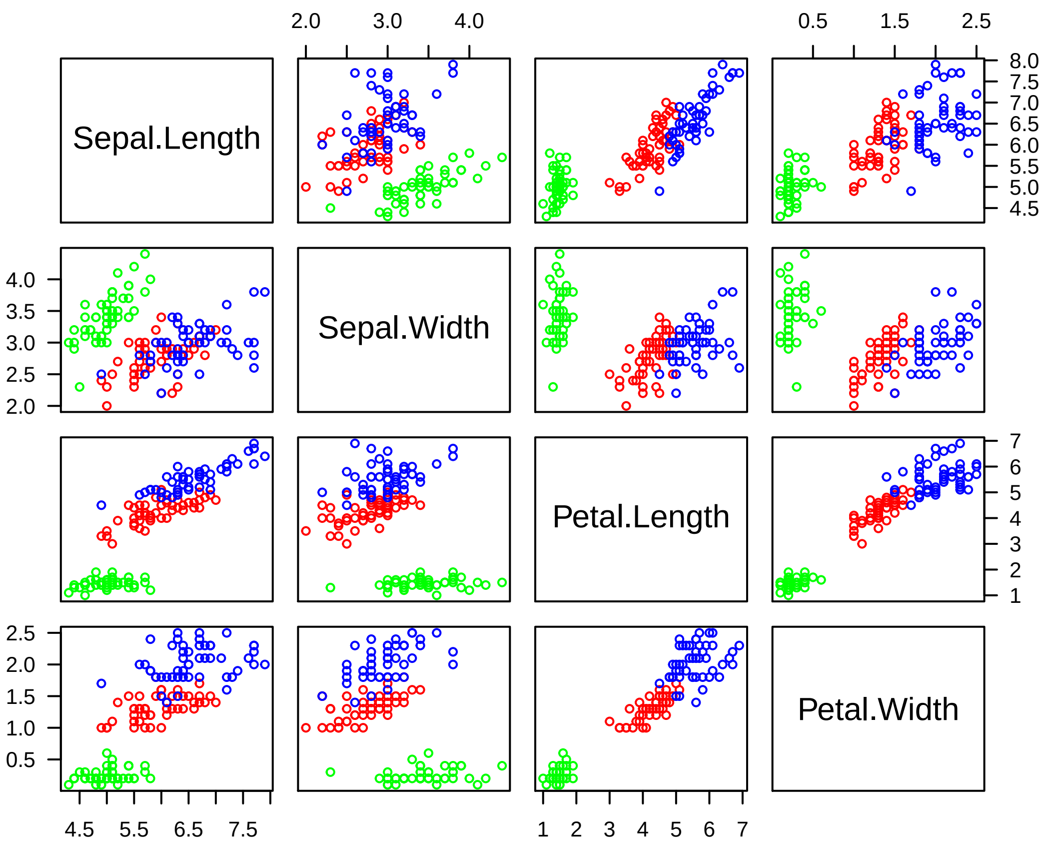
Graphs and Visualization

pairs

```r
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
```

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

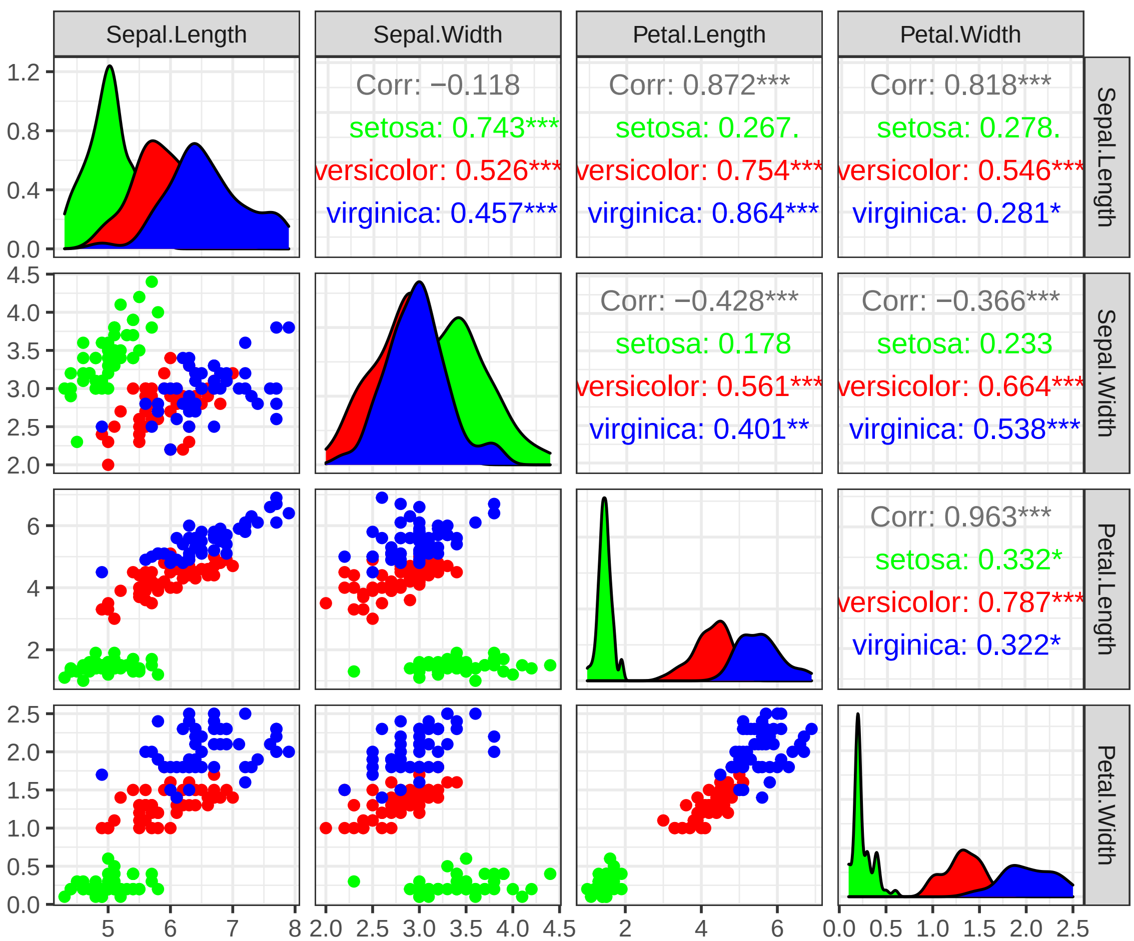


ggpairs

```r
library(GGally)
```

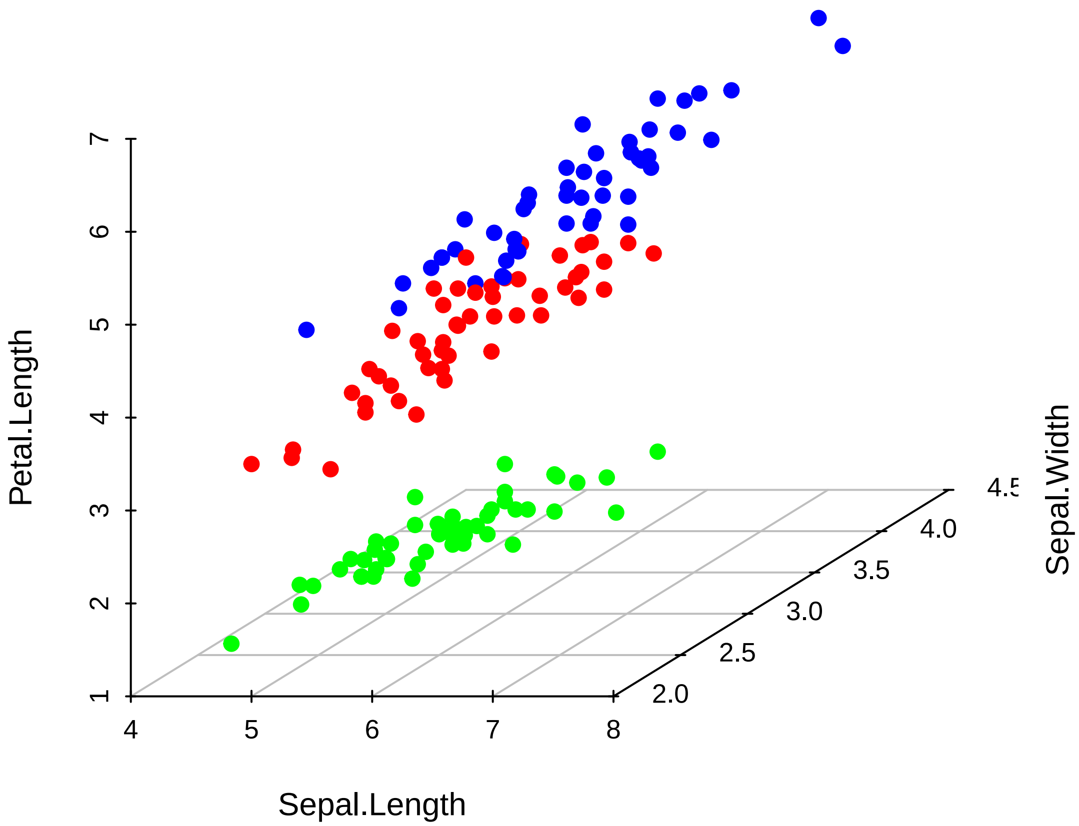
```
## Loading required package: ggplot2

## Warning: replacing previous import 'lifecycle::last_warnings' by
## 'rlang::last_warnings' when loading 'tibble'

## Warning: replacing previous import 'lifecycle::last_warnings' by
## 'rlang::last_warnings' when loading 'pillar'

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```r
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
```



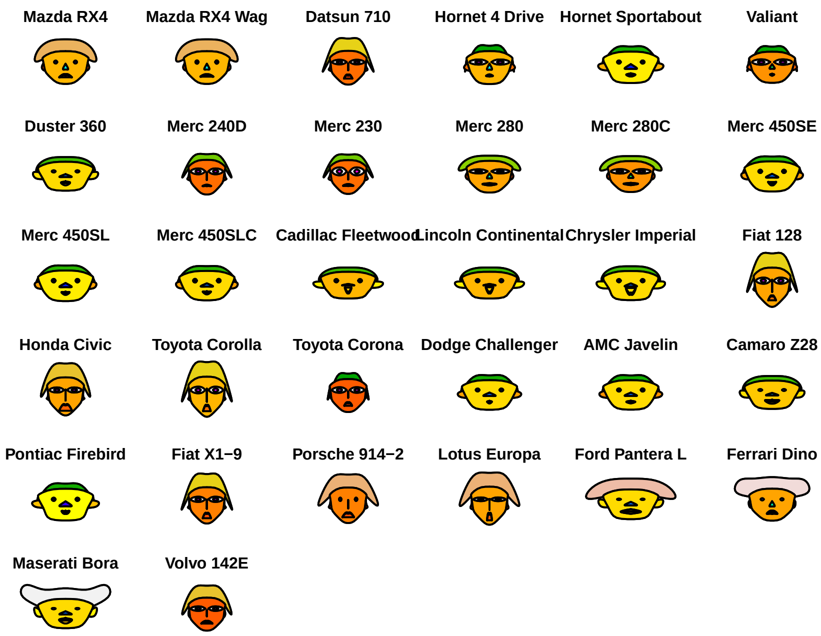
3D Scatter Plot

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



Chernoff Faces

```
library(aplpack)
par(mar = rep(0, 4))
faces(mtcars, cex = 0.8)
```



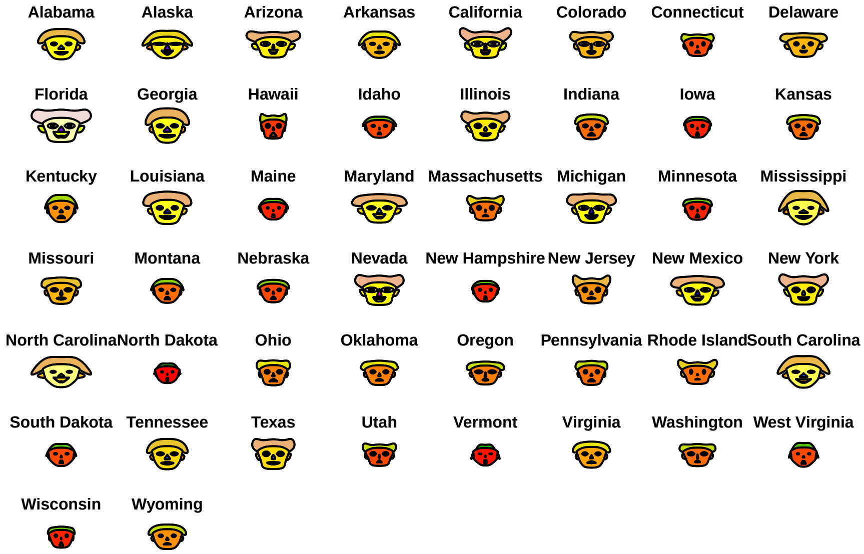

```
## effect of variables:
##  modified item       Var   
##  "height of face   " "mpg" 
##  "width of face    " "cyl" 
##  "structure of face" "disp"
##  "height of mouth  " "hp"  
##  "width of mouth   " "drat"
##  "smiling          " "wt"  
##  "height of eyes   " "qsec"
##  "width of eyes    " "vs"  
##  "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"
```

```
USArrests
```

```
##                Murder Assault UrbanPop Rape
## Alabama          13.2     236       58 21.2
## Alaska           10.0     263       48 44.5
## Arizona           8.1     294       80 31.0
## Arkansas          8.8     190       50 19.5
## California        9.0     276       91 40.6
## Colorado          7.9     204       78 38.7
## Connecticut       3.3     110       77 11.1
## Delaware          5.9     238       72 15.8
## Florida          15.4     335       80 31.9
## Georgia          17.4     211       60 25.8
## Hawaii            5.3      46       83 20.2
## Idaho             2.6     120       54 14.2
```

```
## Illinois         10.4     249       83 24.0
## Indiana           7.2     113       65 21.0
## Iowa              2.2      56       57 11.3
## Kansas            6.0     115       66 18.0
## Kentucky          9.7     109       52 16.3
## Louisiana        15.4     249       66 22.2
## Maine             2.1      83       51  7.8
## Maryland         11.3     300       67 27.8
## Massachusetts     4.4     149       85 16.3
## Michigan         12.1     255       74 35.1
## Minnesota         2.7      72       66 14.9
## Mississippi      16.1     259       44 17.1
## Missouri          9.0     178       70 28.2
## Montana           6.0     109       53 16.4
## Nebraska          4.3     102       62 16.5
## Nevada           12.2     252       81 46.0
## New Hampshire     2.1      57       56  9.5
## New Jersey        7.4     159       89 18.8
## New Mexico       11.4     285       70 32.1
## New York         11.1     254       86 26.1
## North Carolina   13.0     337       45 16.1
## North Dakota      0.8      45       44  7.3
## Ohio              7.3     120       75 21.4
## Oklahoma          6.6     151       68 20.0
## Oregon            4.9     159       67 29.3
## Pennsylvania      6.3     106       72 14.9
## Rhode Island      3.4     174       87  8.3
## South Carolina   14.4     279       48 22.5
## South Dakota      3.8      86       45 12.8
## Tennessee        13.2     188       59 26.9
## Texas            12.7     201       80 25.5
## Utah              3.2     120       80 22.9
## Vermont           2.2      48       32 11.2
## Virginia          8.5     156       63 20.7
## Washington        4.0     145       73 26.2
## West Virginia     5.7      81       39  9.3
## Wisconsin         2.6      53       66 10.8
## Wyoming           6.8     161       60 15.6
```

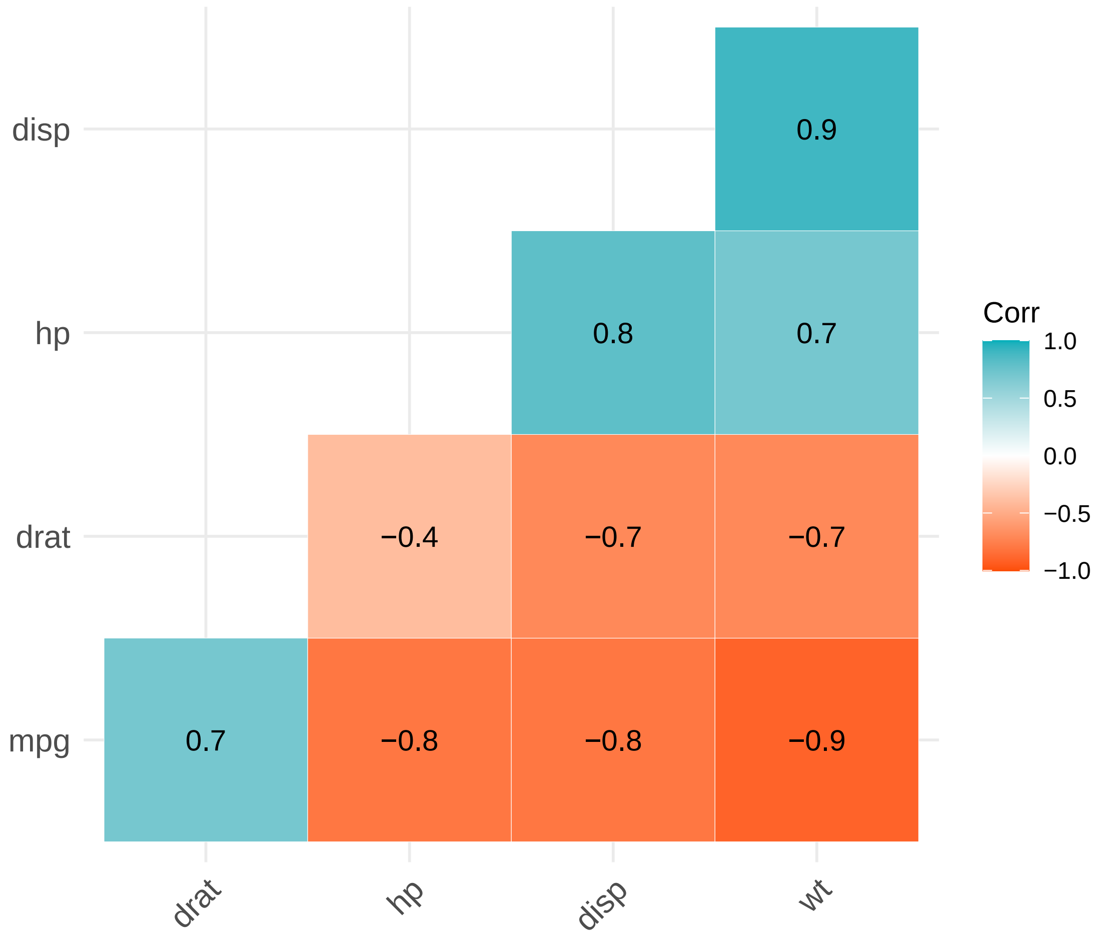
```r
faces(USArrests, cex = 0.8)
```




```
## effect of variables:
##  modified item       Var
##  "height of face   " "Murder"
##  "width of face    " "Assault"
##  "structure of face" "UrbanPop"
##  "height of mouth  " "Rape"
##  "width of mouth   " "Murder"
##  "smiling          " "Assault"
##  "height of eyes   " "UrbanPop"
##  "width of eyes    " "Rape"
##  "height of hair   " "Murder"
##  "width of hair   "  "Assault"
##  "style of hair   "  "UrbanPop"
##  "height of nose  "  "Rape"
##  "width of nose   "  "Murder"
##  "width of ear    "  "Assault"
##  "height of ear   "  "UrbanPop"
```

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