HW1 - STAT 4510/7510 - Spring 2024

Yang, Anton - #14405729

Due Wednesday, Jan. 31, 11:30 pm (upload PDF to Canvas)

Instructions: Please list your name and student number clearly. In order to receive credit for a problem, your solution must show sufficient detail so that the grader can determine how you obtained your answer.

Use R Markdown to create a WORD file. Before submitting, make sure you convert the WORD file to a PDF. All R code should be included, as well as all output produced. Upload your work to the Canvas course site.

Problem 1

Complete 2.3 Lab: Introduction to R, found on pages 42 - 52. *(You are expected to simply work through the textbook lab as written and execute the commands. Include all commands and output in your homework submission.)*

```
x<-c(1,3,2,5)
x
```

```
## [1] 1 3 2 5
```

```
x = c(1,6,2)
x
```

```
## [1] 1 6 2
```

```
y = c(1,4,3)
```

```
length(x)
```

```
## [1] 3
```

```
length(y)
```

```
## [1] 3
```

```
x+y
```

```
## [1]  2 10  5
```

```
ls()
```

```
## [1] "x" "y"
```

```
rm(x,y)

ls()

## character(0)

rm(list=ls())

?matrix

x<-matrix(data=c(1,2,3,4), nrow=2, ncol=2)
x

##      [,1] [,2]
## [1,]    1    3
## [2,]    2    4

x<-matrix(c(1,2,3,4),2,2)

matrix(c(1,2,3,4),2,2,byrow=TRUE)

##      [,1] [,2]
## [1,]    1    2
## [2,]    3    4

sqrt(x)

##          [,1]     [,2]
## [1,] 1.000000 1.732051
## [2,] 1.414214 2.000000

x^2

##      [,1] [,2]
## [1,]    1    9
## [2,]    4   16

x<-rnorm(50)
y<-x+rnorm(50, mean = 50, sd = .1)
cor(x,y)

## [1] 0.9951859

set.seed(1303)
rnorm(50)

##  [1] -1.1439763145  1.3421293656  2.1853904757  0.5363925179  0.0631929665
##  [6]  0.5022344825 -0.0004167247  0.5658198405 -0.5725226890 -1.1102250073
## [11] -0.0486871234 -0.6956562176  0.8289174803  0.2066528551 -0.2356745091
## [16] -0.5563104914 -0.3647543571  0.8623550343 -0.6307715354  0.3136021252
## [21] -0.9314953177  0.8238676185  0.5233707021  0.7069214120  0.4202043256
## [26] -0.2690521547 -1.5103172999 -0.6902124766 -0.1434719524 -1.0135274099
```

```
## [31]  1.5732737361  0.0127465055  0.8726470499  0.4220661905 -0.0188157917
## [36]  2.6157489689 -0.6931401748 -0.2663217810 -0.7206364412  1.3677342065
## [41]  0.2640073322  0.6321868074 -1.3306509858  0.0268888182  1.0406363208
## [46]  1.3120237985 -0.0300020767 -0.2500257125  0.0234144857  1.6598706557
```

```
set.seed(3)
y<-rnorm(100)
mean(y)
```

```
## [1] 0.01103557
```

```
var(y)
```

```
## [1] 0.7328675
```

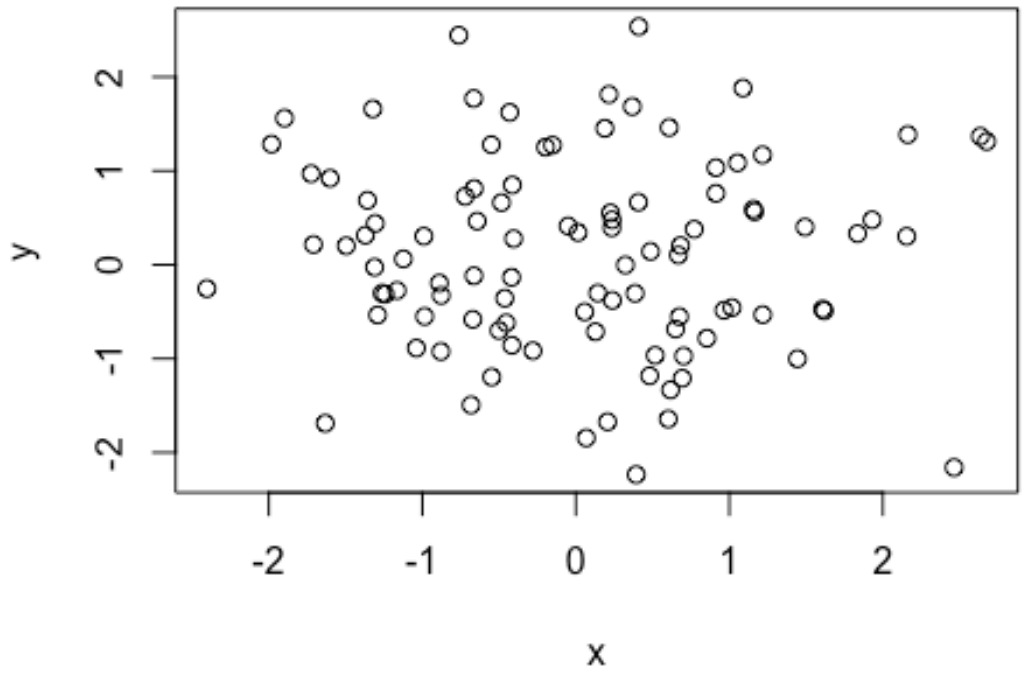
```
sqrt(var(y))
```

```
## [1] 0.8560768
```

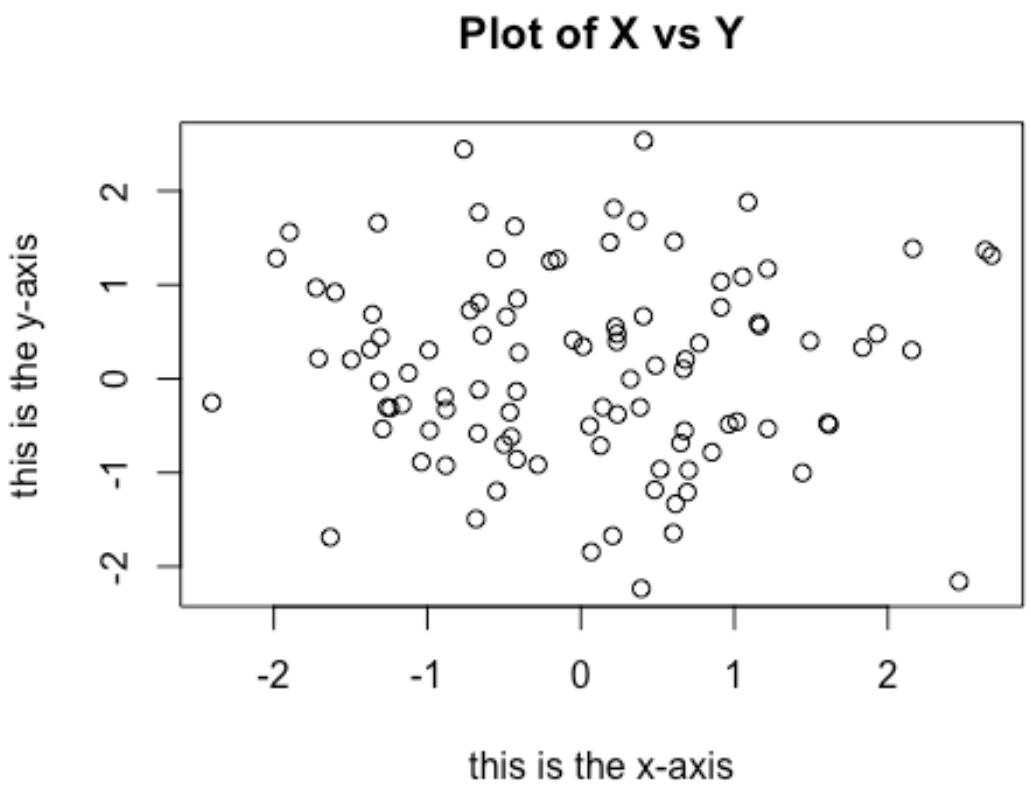
```
sd(y)
```

```
## [1] 0.8560768
```

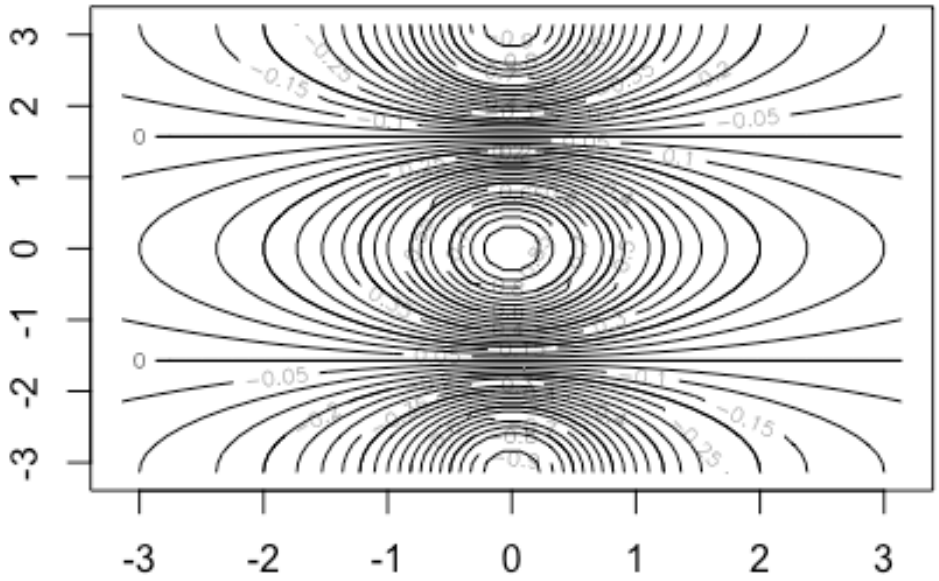
```
x<-rnorm(100)
y<-rnorm(100)
plot(x,y)
```



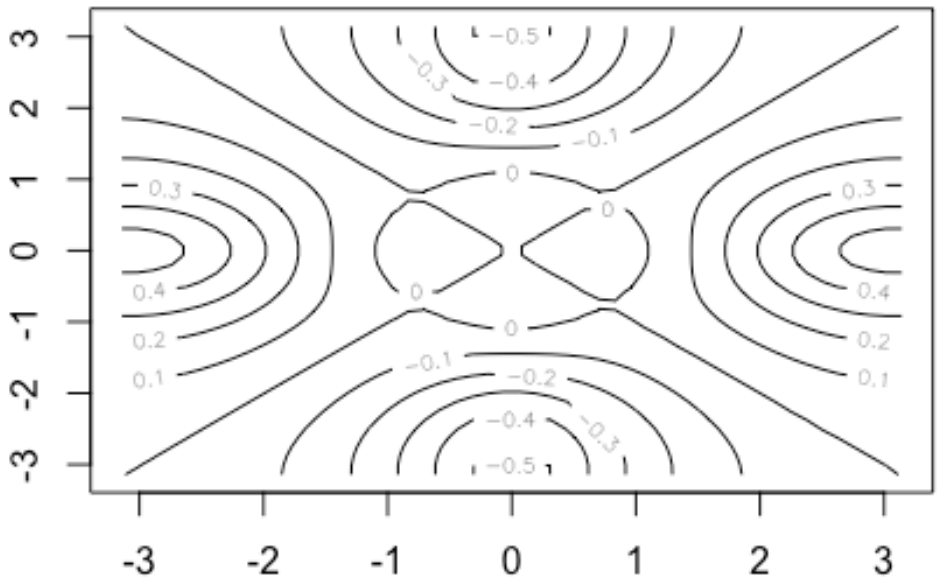

```
plot(x,y, xlab = "this is the x-axis",
     ylab = "this is the y-axis",
     main = "Plot of X vs Y")
```



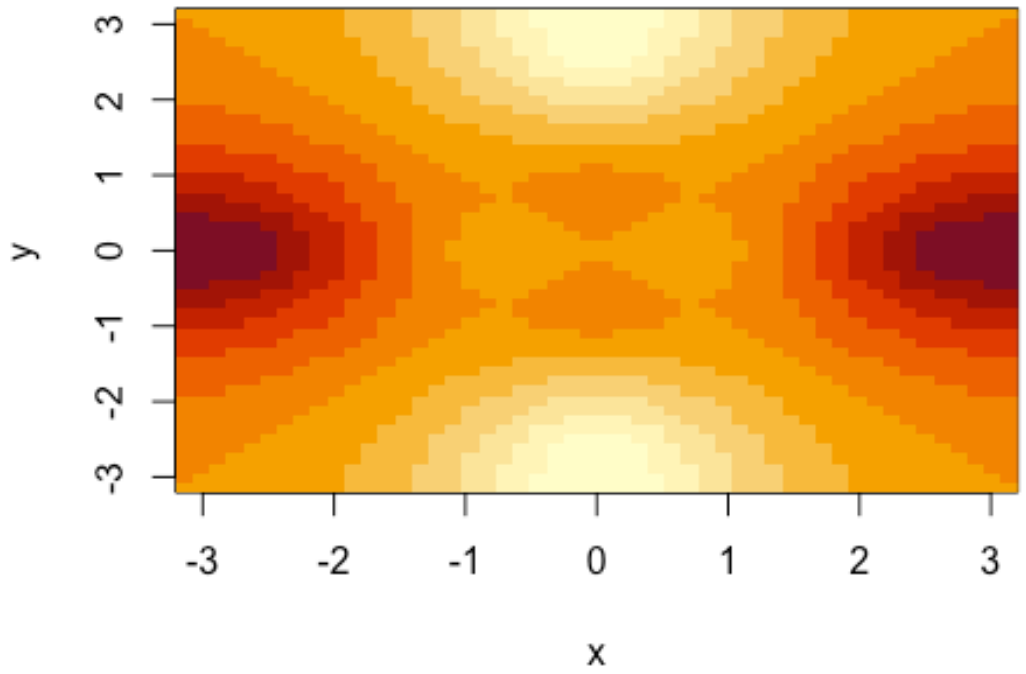

```
pdf("Figure.pdf")
plot(x,y, col="green")
dev.off()

## quartz_off_screen
##                 2

x<-seq(1,10)
x

##  [1]  1  2  3  4  5  6  7  8  9 10

x<-1:10
x

##  [1]  1  2  3  4  5  6  7  8  9 10

x<-seq(-pi,pi,length=50)
y<-x
f<-outer(x,y, function(x,y) cos(y) / (1+x^2))
contour(x,y,f)
contour(x, y, f, nlevels = 45, add = T)
```



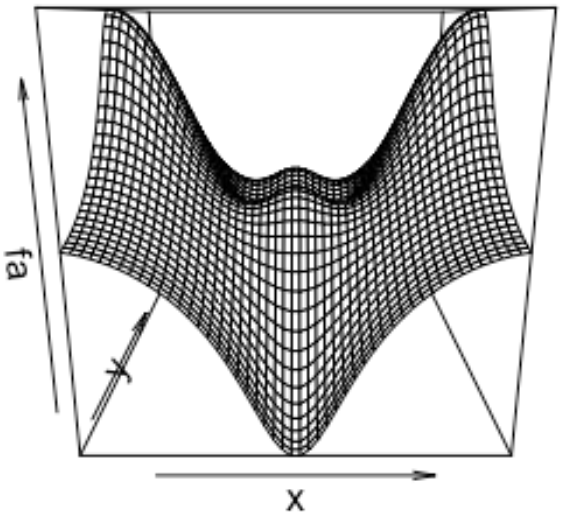

```
fa <- (f - t(f)) / 2
contour(x, y, fa, nlevels = 15)
```



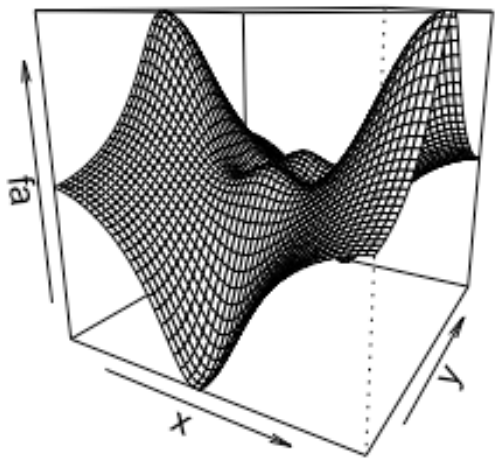

```
image(x, y, fa)
```



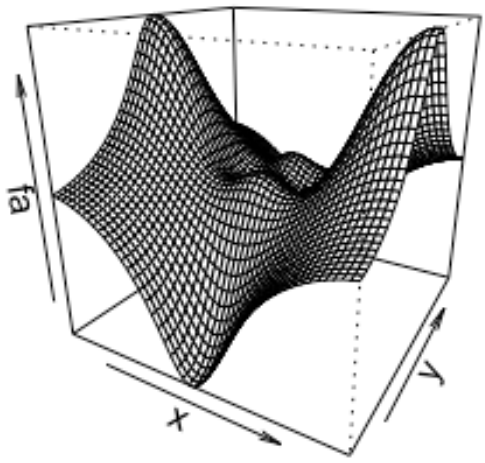

```
persp(x, y, fa)
```



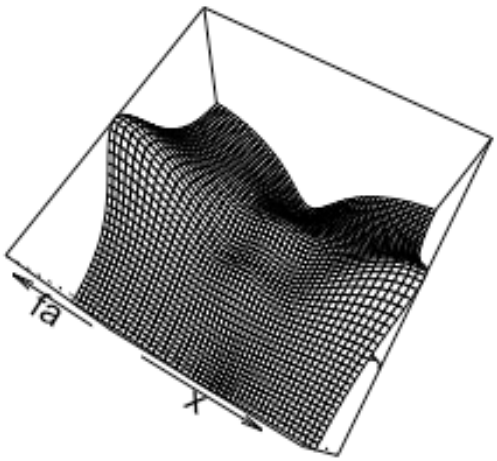

```
persp(x, y, fa, theta=30)
```



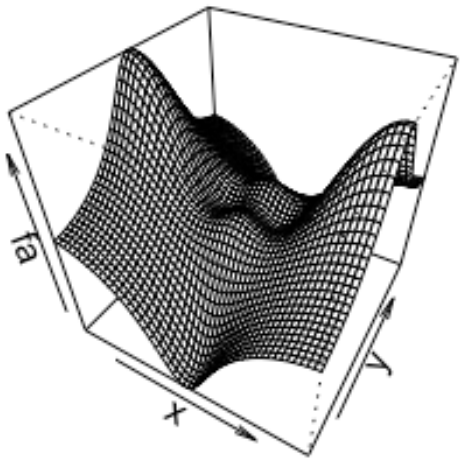

```
persp(x, y, fa, theta=30, phi=20)
```




```
persp(x, y, fa, theta=30, phi=70)
```




```
persp(x, y, fa, theta = 30, phi = 40)
```




```
A <-matrix(1:16, 4, 4)
A
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    5    9   13
## [2,]    2    6   10   14
## [3,]    3    7   11   15
## [4,]    4    8   12   16
```

```
A[2, 3]
```

```
## [1] 10
```

```
A[c(1, 3), c(2, 4)]
```

```
##      [,1] [,2]
## [1,]    5   13
## [2,]    7   15
```

```
A[1:3, 2:4]
```

```
##      [,1] [,2] [,3]
## [1,]    5    9   13
## [2,]    6   10   14
## [3,]    7   11   15
```

```r
A[1:2, ]
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    5    9   13
## [2,]    2    6   10   14
```

```r
A[, 1:2]
```

```
##      [,1] [,2]
## [1,]    1    5
## [2,]    2    6
## [3,]    3    7
## [4,]    4    8
```

```r
A[1, ]
```

```
## [1]  1  5  9 13
```

```r
A[-c(1,3), ]
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    2    6   10   14
## [2,]    4    8   12   16
```

```r
A[-c(1,3), -c(1,3,4)]
```

```
## [1] 6 8
```

```r
dim(A)
```

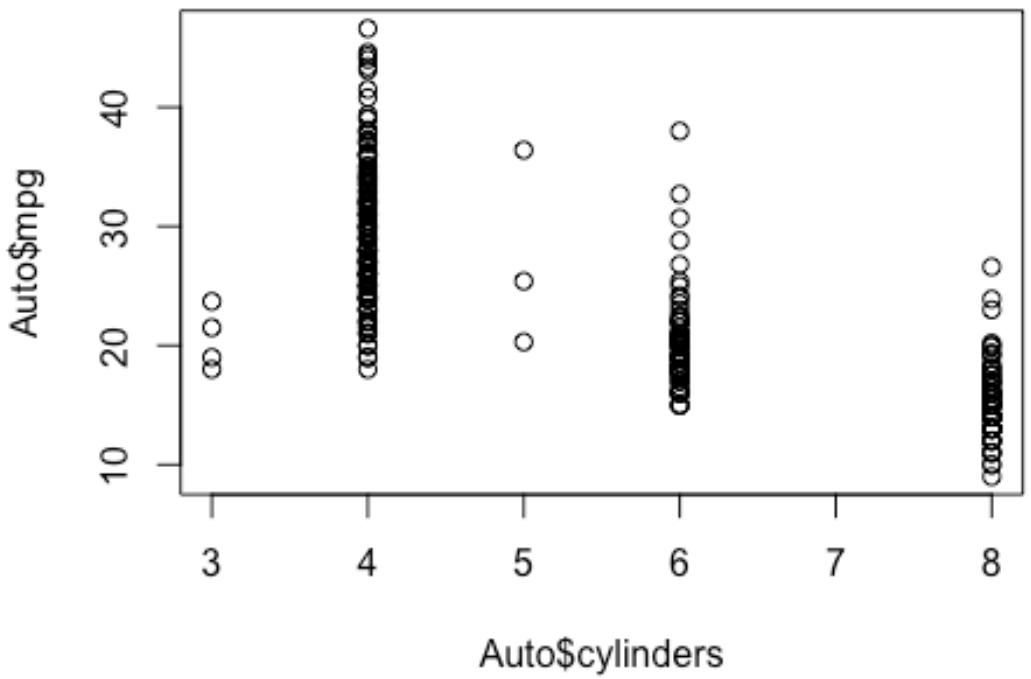
```
## [1] 4 4
```

```r
Auto <- read.table("Auto.data")
View(Auto)
head(Auto)
```

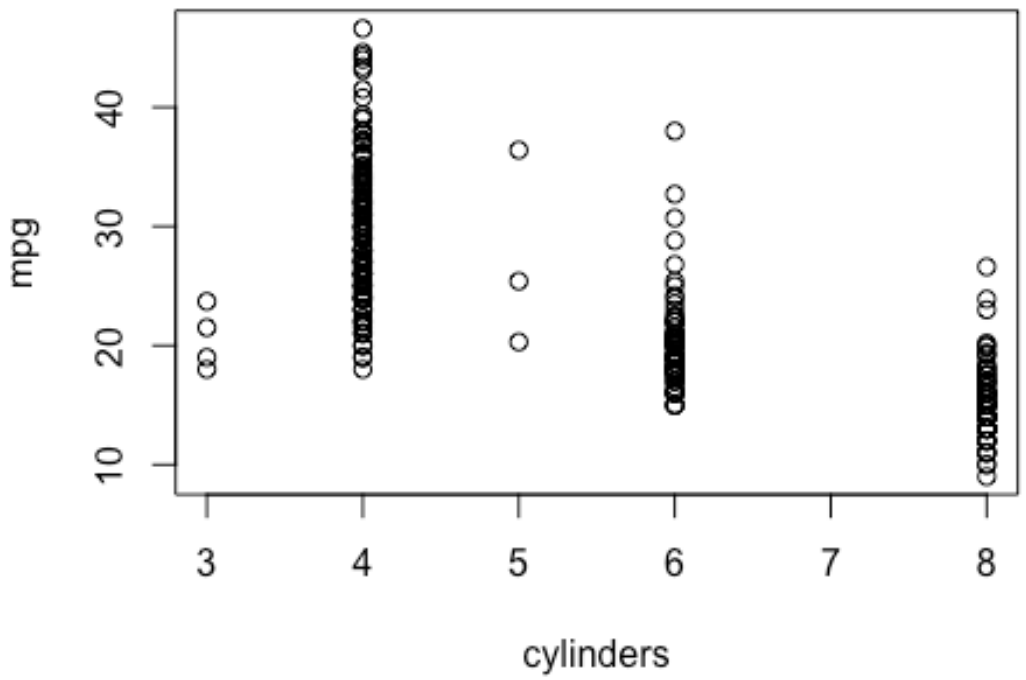
```
##     V1        V2           V3         V4     V5           V6   V7     V8
## 1  mpg cylinders displacement horsepower weight acceleration year origin
## 2 18.0         8        307.0      130.0  3504.         12.0   70      1
## 3 15.0         8        350.0      165.0  3693.         11.5   70      1
## 4 18.0         8        318.0      150.0  3436.         11.0   70      1
## 5 16.0         8        304.0      150.0  3433.         12.0   70      1
## 6 17.0         8        302.0      140.0  3449.         10.5   70      1
##                          V9
## 1                      name
## 2 chevrolet chevelle malibu
## 3         buick skylark 320
## 4        plymouth satellite
## 5             amc rebel sst
## 6               ford torino
```

```r
Auto <- read.table("Auto.data", header = T, na.strings = "?",
stringsAsFactors = T)
```

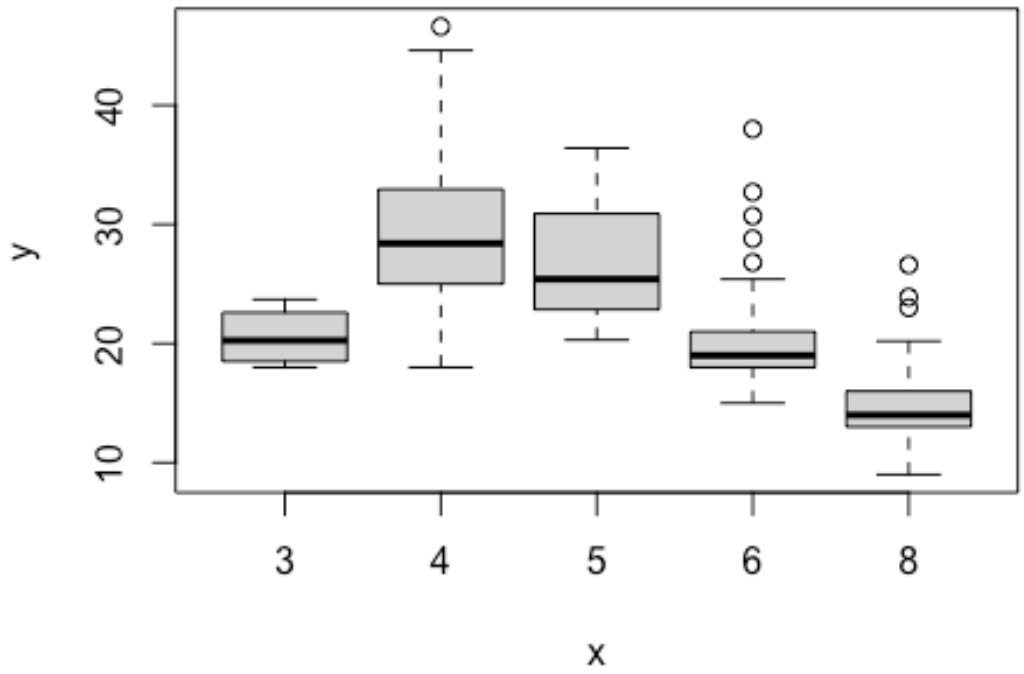
```
View(Auto)

Auto<-read.csv("Auto.csv", na.strings="?", stringsAsFactors = T)
View(Auto)
dim(Auto)

## [1] 397   9

Auto[1:4, ]

##   mpg cylinders displacement horsepower weight acceleration year origin
## 1  18         8          307        130   3504         12.0   70      1
## 2  15         8          350        165   3693         11.5   70      1
## 3  18         8          318        150   3436         11.0   70      1
## 4  16         8          304        150   3433         12.0   70      1
##                       name
## 1 chevrolet chevelle malibu
## 2         buick skylark 320
## 3        plymouth satellite
## 4             amc rebel sst

Auto<-na.omit(Auto)
dim(Auto)

## [1] 392   9

names(Auto)

## [1] "mpg"          "cylinders"    "displacement" "horsepower"   "weight"
## [6] "acceleration" "year"         "origin"       "name"

plot(Auto$cylinders, Auto$mpg)
```



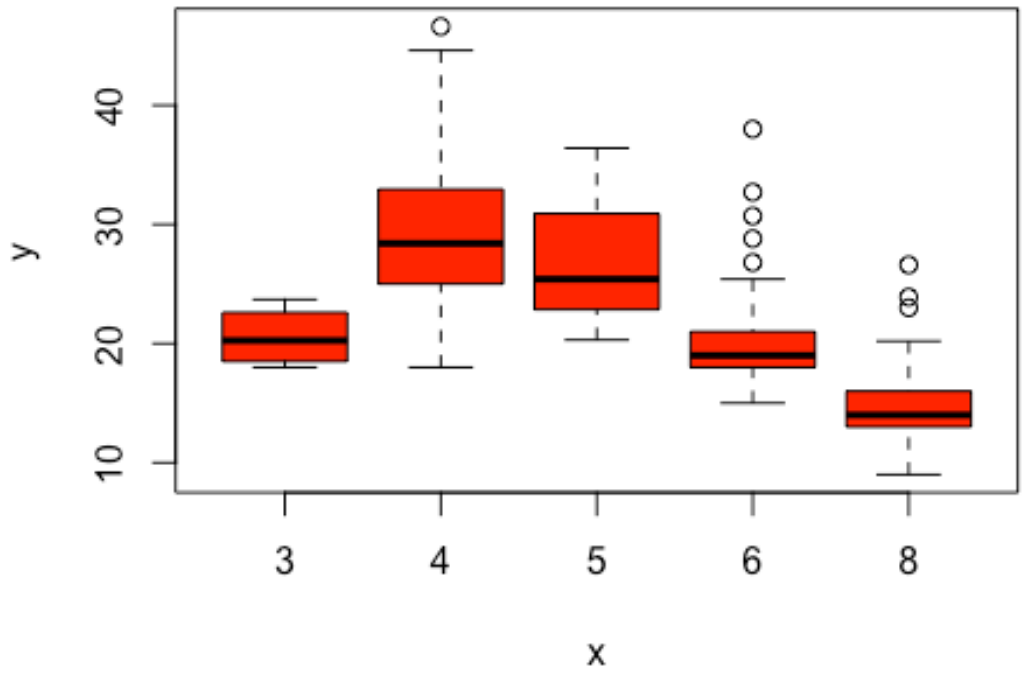

```
attach(Auto)
plot(cylinders, mpg)
```



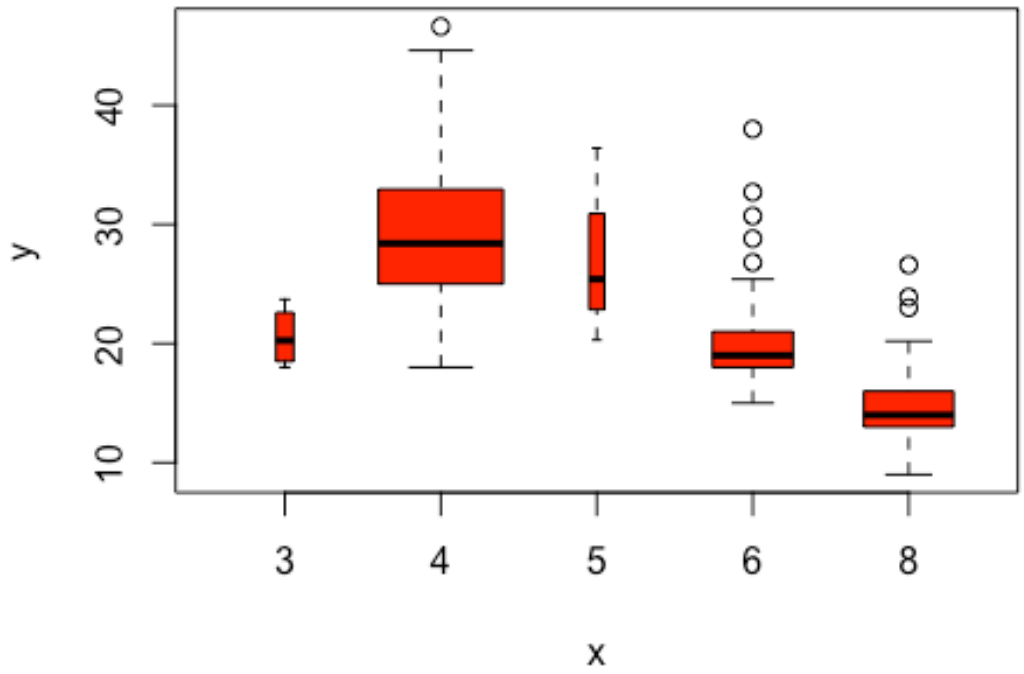

```
cylinders<-as.factor(cylinders)
plot(cylinders, mpg)
```



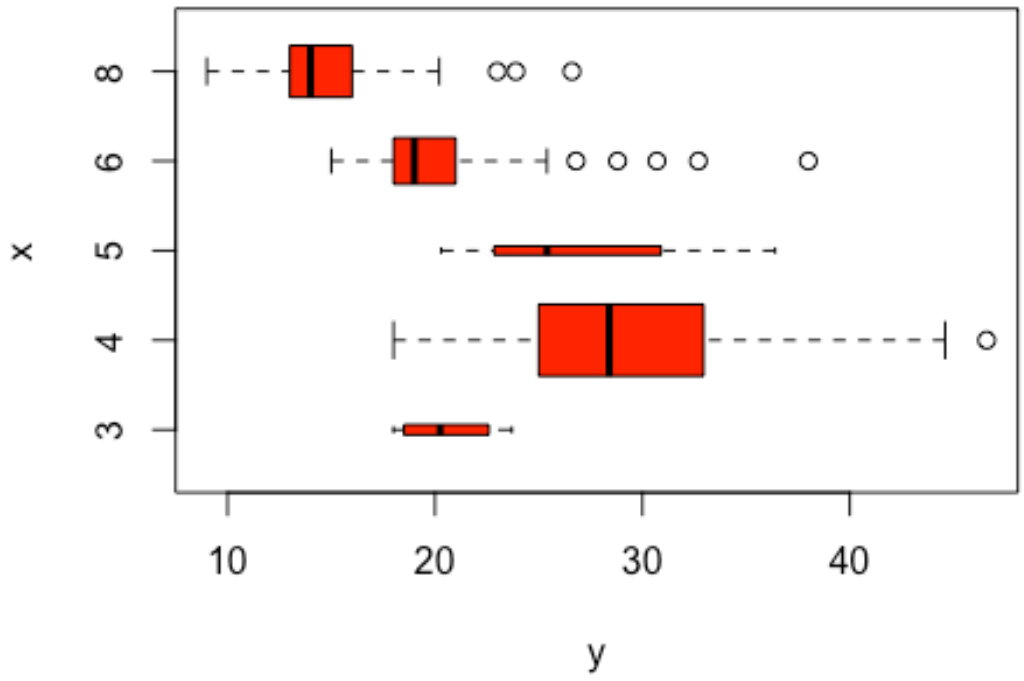

```
plot(cylinders, mpg, col = "red")
```



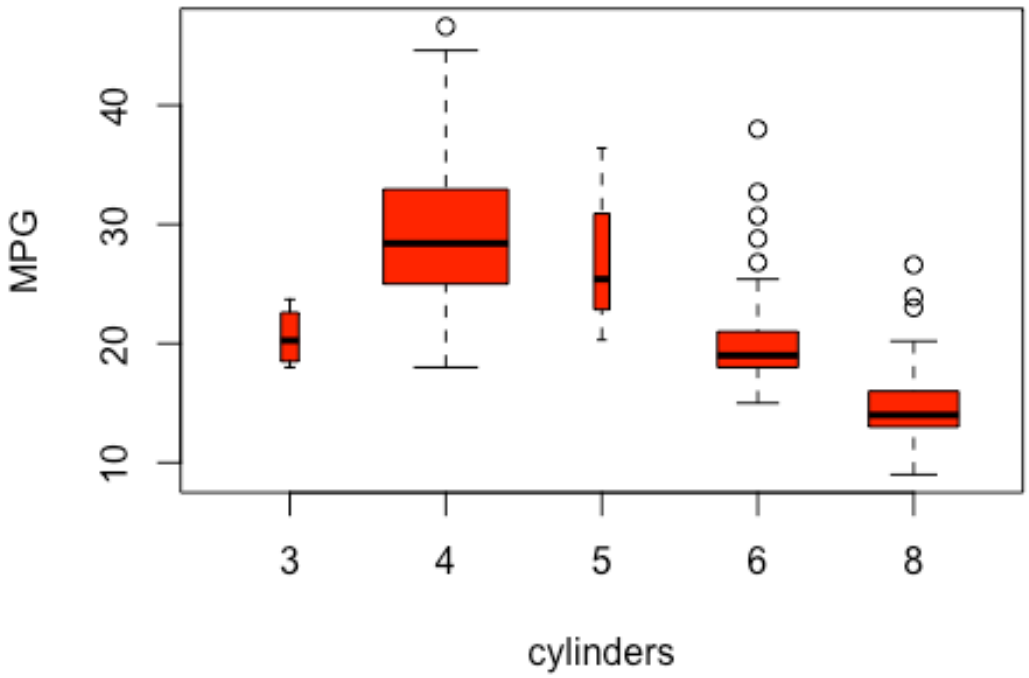

```
plot(cylinders, mpg, col = "red", varwidth = T)
```



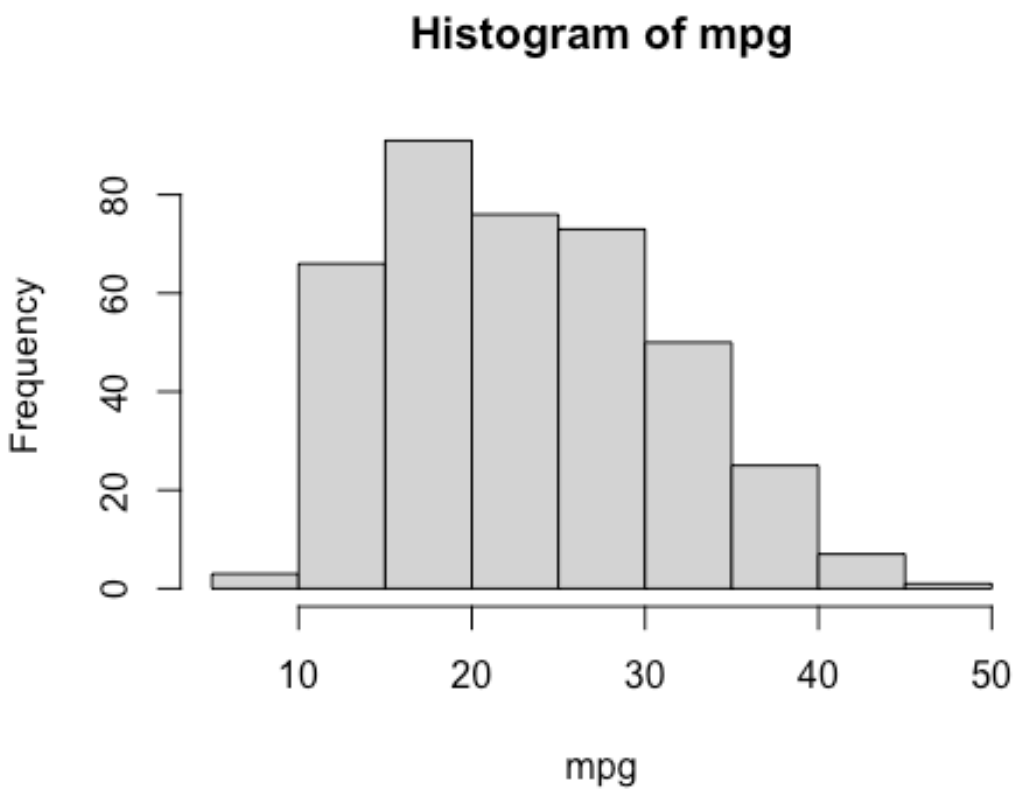

```
plot(cylinders, mpg, col = "red", varwidth = T, horizontal = T)
```



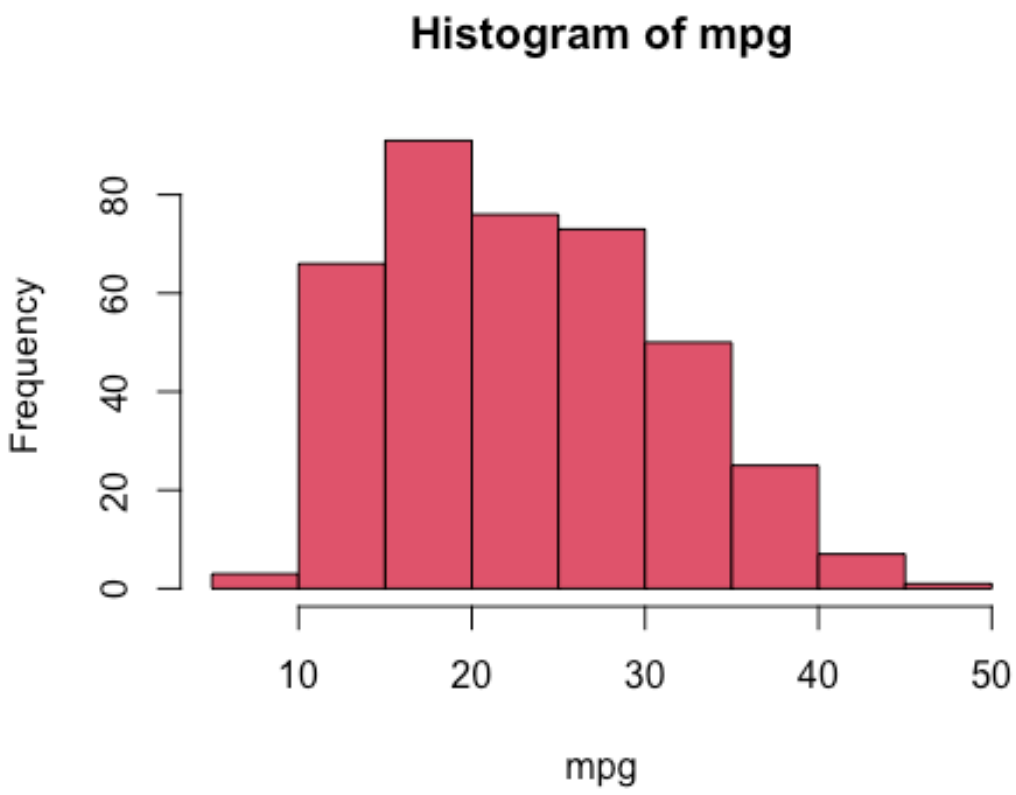

```
plot(cylinders, mpg, col = "red", varwidth = T, xlab = "cylinders", ylab =
"MPG")
```



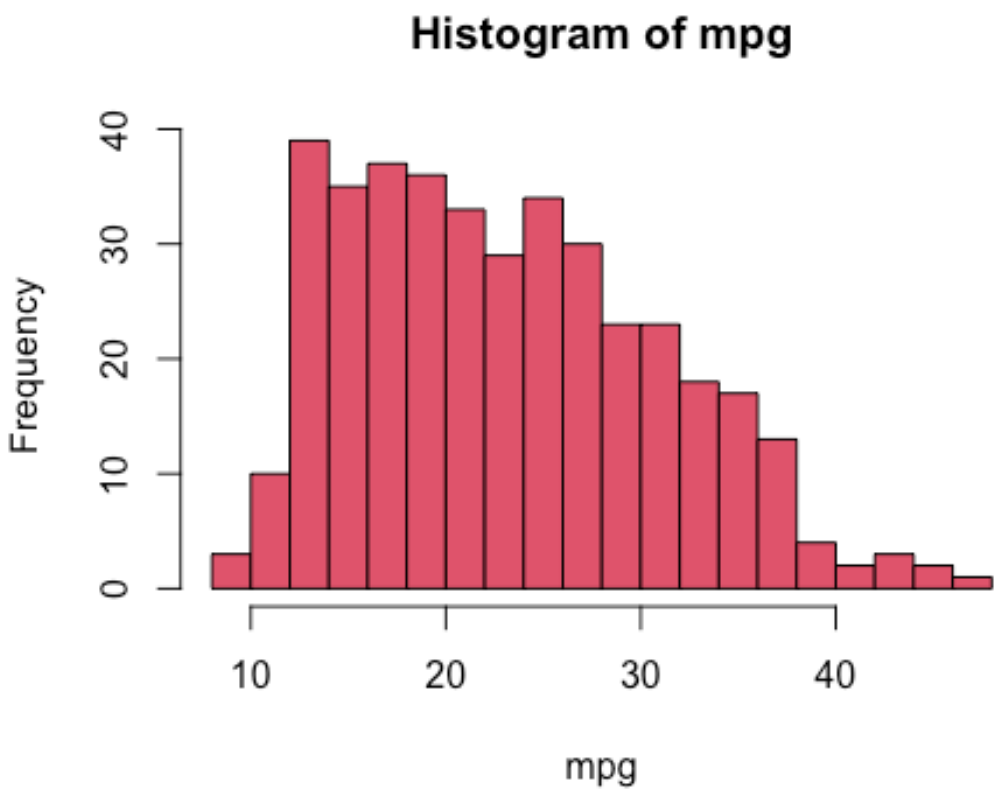

```
hist(mpg)
```



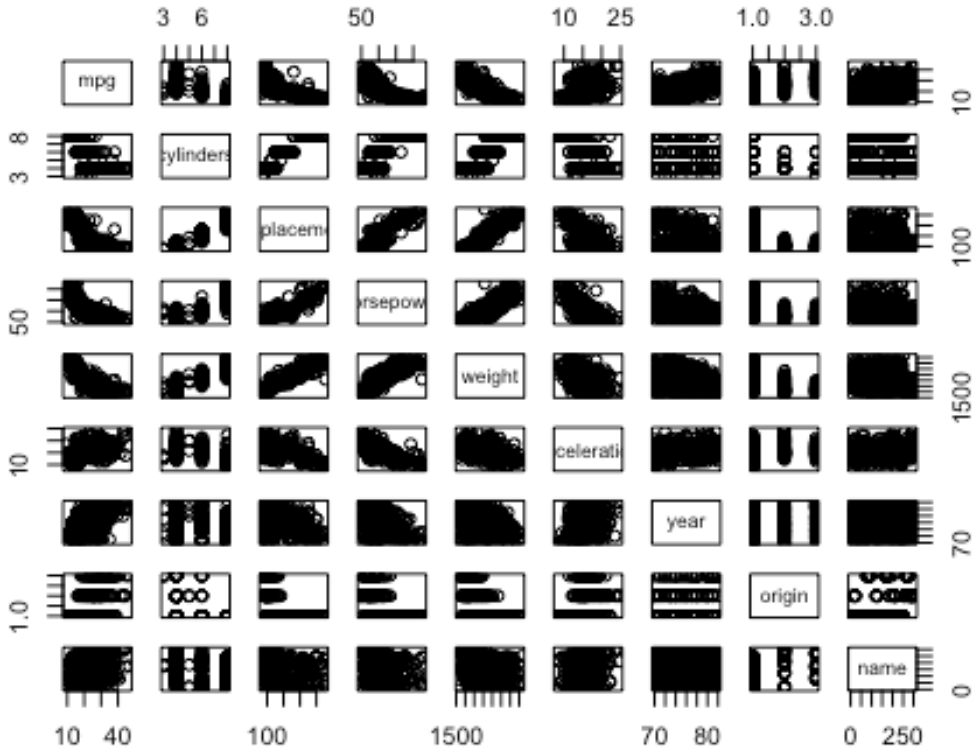

```
hist(mpg, col = 2)
```



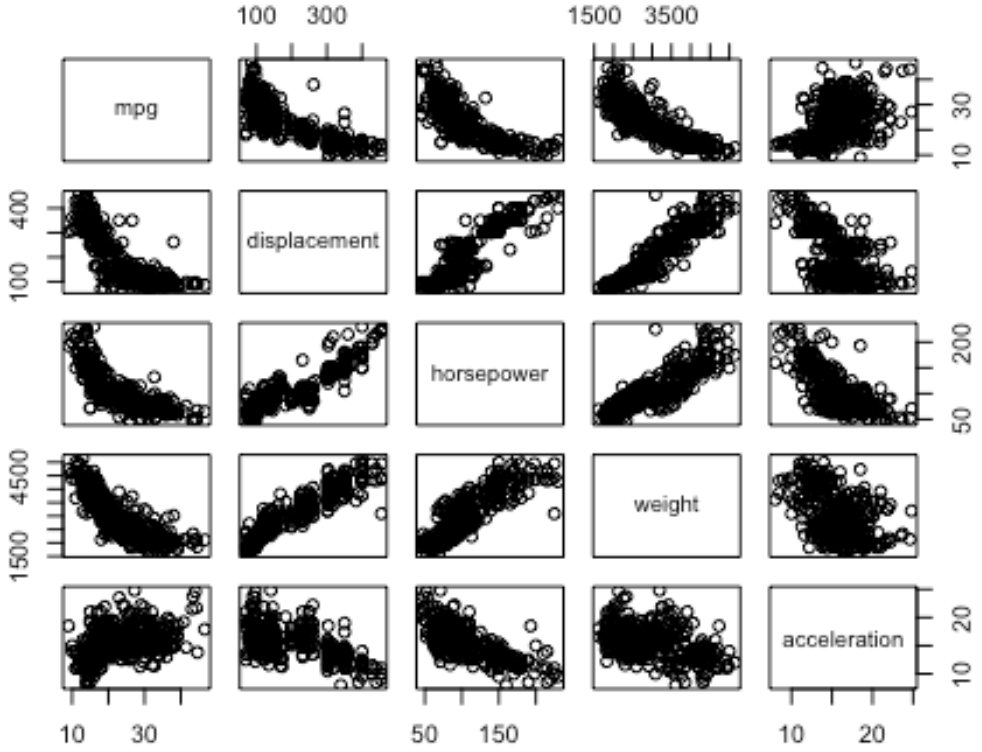

```
hist(mpg, col = 2, breaks = 15)
```



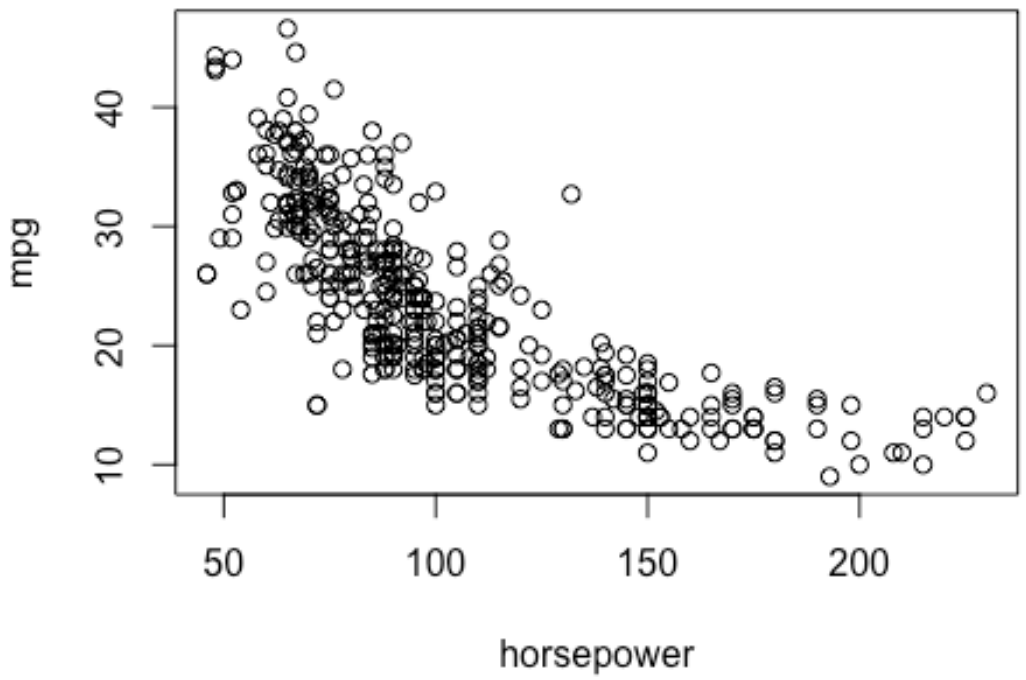

```
pairs(Auto)
```




```
pairs(
  ~mpg +displacement + horsepower + weight + acceleration, data = Auto
)
```




```
plot(horsepower, mpg)
identify(horsepower, mpg, name)
```




```
## integer(0)

summary(Auto)

##       mpg          cylinders      displacement     horsepower
weight
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0   Min.
:1613
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0   1st
Qu.:2225
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5   Median
:2804
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5   Mean
:2978
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0   3rd
Qu.:3615
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0   Max.
:5140
##
##   acceleration        year           origin                      name
##  Min.   : 8.00   Min.   :70.00   Min.   :1.000   amc matador       :  5
##  1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000   ford pinto        :  5
##  Median :15.50   Median :76.00   Median :1.000   toyota corolla    :  5
```

```
##  Mean   :15.54   Mean   :75.98   Mean   :1.577   amc gremlin       :  4
##  3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000   amc hornet        :  4
##  Max.   :24.80   Max.   :82.00   Max.   :3.000   chevrolet chevette:  4
##                                                  (Other)           :365

summary(mpg)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    9.00   17.00   22.75   23.45   29.00   46.60
```

Problem 2

The file `iris.csv` contains the famous (Fisher's or Anderson's) iris data set. It gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are Iris setosa, versicolor, and virginica.

a) Use the function `read.csv()` to read the data into `R` and call it `iris`.

```
iris<-read.csv("iris.csv")
```

b) Use the `str()` command to look at the structure of the data set. How many observations are there? How many variables? What type of data is each variable (*character, numeric, integer, logical, or complex*)?

```
str(iris)

## 'data.frame':    150 obs. of  5 variables:
##  $ sepal.length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ sepal.width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ petal.length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ petal.width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ variety     : chr  "Setosa" "Setosa" "Setosa" "Setosa" ...
```

There are 150 observations and 5 variables. Variables sepal.length, sepal. width, petal.length, petal.width are numeric and variety is character.

c) Change the variable `variety` to a factor variable using the `as.factor()` command.

```
iris$variety<-as.factor(iris$variety)
```

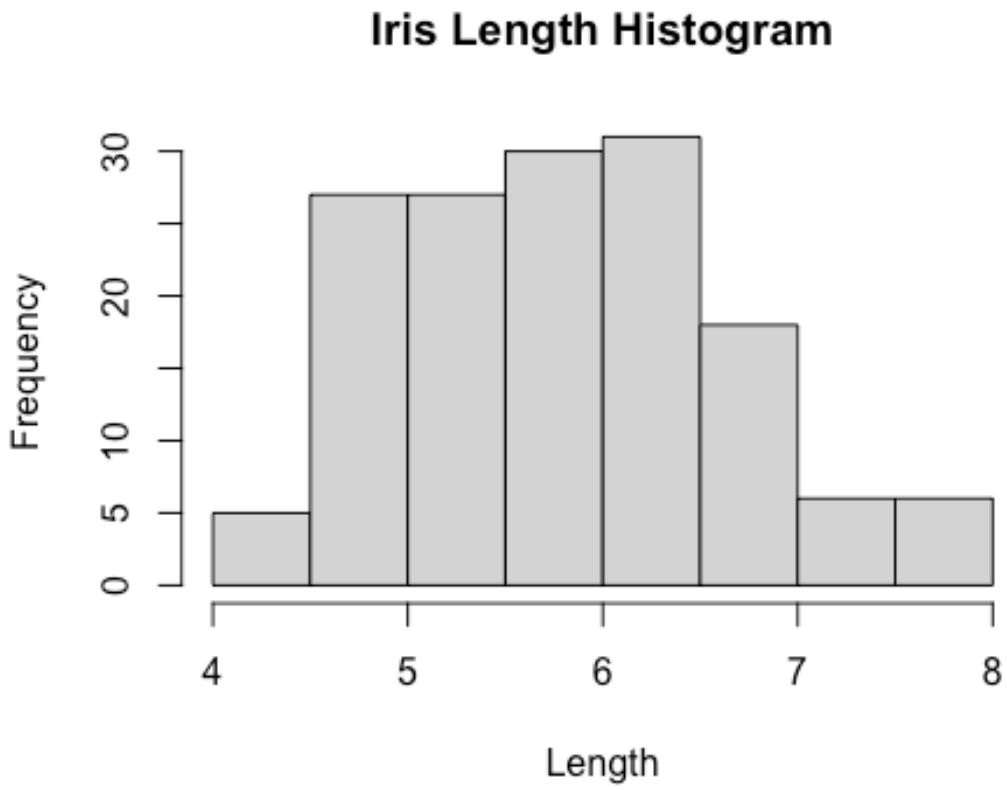
d) Produce a summary table of the data set.

```
summary(iris)

##   sepal.length    sepal.width     petal.length    petal.width
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
##        variety
##  Setosa    :50
```

```
##  Versicolor:50
##  Virginica :50
##
##
##
```

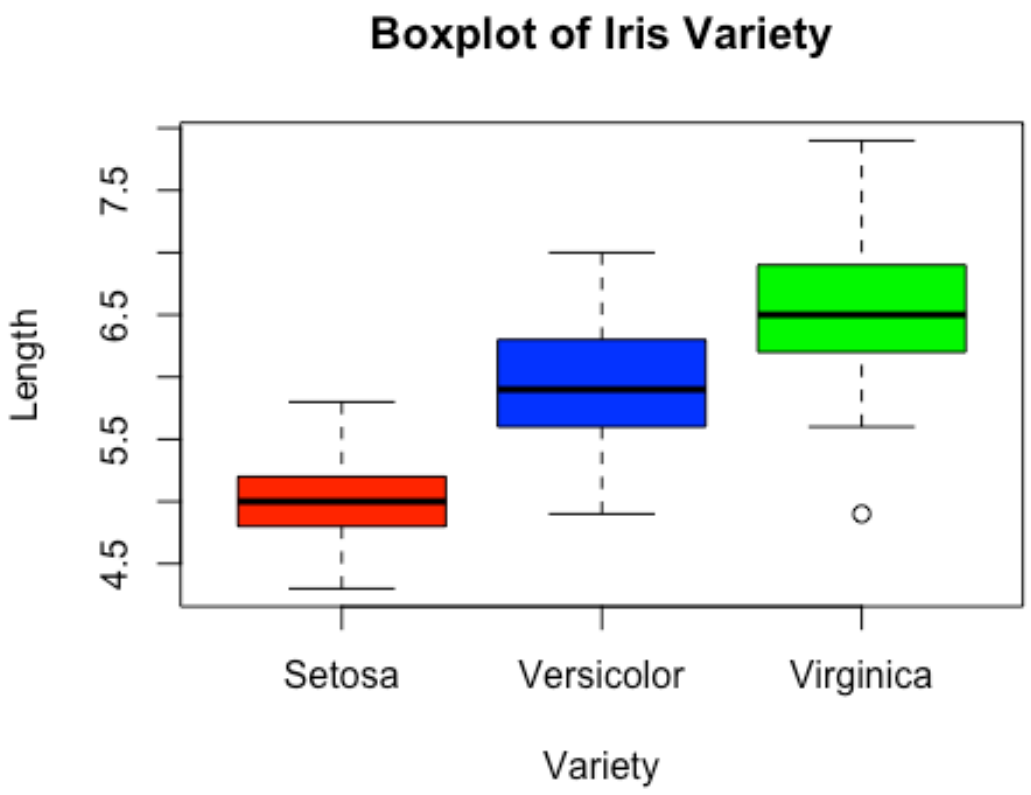
e) Use the `hist()` command to create a histogram for the variable `sepal.length`. Add a title and axis labels to the plot.

```
hist(iris$sepal.length,
     main = "Iris Length Histogram",
     xlab = "Length",
     ylab = "Frequency")
```



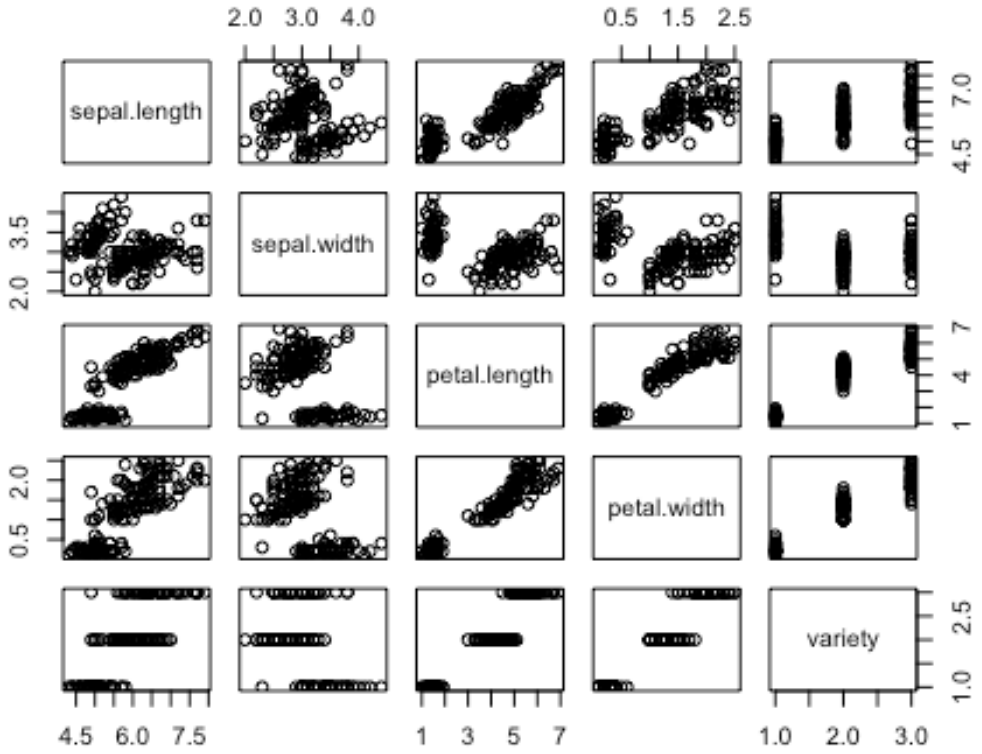
f) Use the plot() command to create a boxplot of `sepal.length` for each `variety`. Add a title and axis labels to each plot and use a different color for each `variety` of iris.

```
plot(iris$variety,iris$sepal.length,
     main = "Boxplot of Iris Variety",
     xlab = "Variety",
     ylab = "Length",
     col = c("red", "blue","green"))
```



g) Produce a scatterplot matrix of all variables and note some relationships between them. Which attributes appear to be highly related? Which attributes do a good job of distinguishing variety?

```
pairs(iris)
```



The variables Sepal Length and Petal Length, Sepal Length and Petal Width, Petal Length and Petal Width, and Petal Length and Petal Width appear highly related. Petal Width and Petal Length do a good job on distinguishing variety.

h) Use the `cor()` function to produce a correlation matrix for the data. Note that this function will only work for data that is numeric, so you will need to filter out any variables that are not numeric. Are there any variables with high correlations (values close to 1 or -1)?

```
iris_cordata<-iris[, !names(iris) %in% "variety"]
cor(iris_cordata)
```

```
##              sepal.length sepal.width petal.length petal.width
## sepal.length    1.0000000  -0.1175698    0.8717538   0.8179411
## sepal.width    -0.1175698   1.0000000   -0.4284401  -0.3661259
## petal.length    0.8717538  -0.4284401    1.0000000   0.9628654
## petal.width     0.8179411  -0.3661259    0.9628654   1.0000000
```

Petal Width and Petal Length have high correlation with 0.9628654 which is very highly correlated. Petal Length and Sepal Length with correlation 0.8717538, and Petal Width and Sepal Length with correlation 0.8179411 are also moderately high correlated, but not as good as Petal Width and Petal Length.