Lab1

Weijing Tang

January 12, 2018

R code for this Lab note can be found online [http://www-bcf.usc.edu/~gareth/ISL/data.html].

Chapter 2 Lab: Introduction to R

Installing R on your Personal Computer

Download from: [http://cran.mtu.edu/]

RStudio allows the user to run R in a more user-friendly enviroment. It is open-source and available at [http://www.rstudio.com/].

Basic Commands

```
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"
```

```
# ls returns a vector of character strings giving the names of the objects
# in the specified environment.
rm(x,y)
# remove objects from a specified environment
ls()
```

```
## character(0)
```

```r
rm(list=ls()) # remove all objects in this environment
# ?matrix
x=matrix(data=c(1,2,3,4), nrow=2, ncol=2)
x
```

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

```r
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
```

```r
sqrt(x)
```

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

```r
x^2
```

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

```r
x=rnorm(50)
y=x+rnorm(50,mean=50,sd=.1)
cor(x,y)
```

```
## [1] 0.9947947
```

```r
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
```

```r
set.seed(3)
y=rnorm(100)
mean(y)
```

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

```r
var(y)
```

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

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

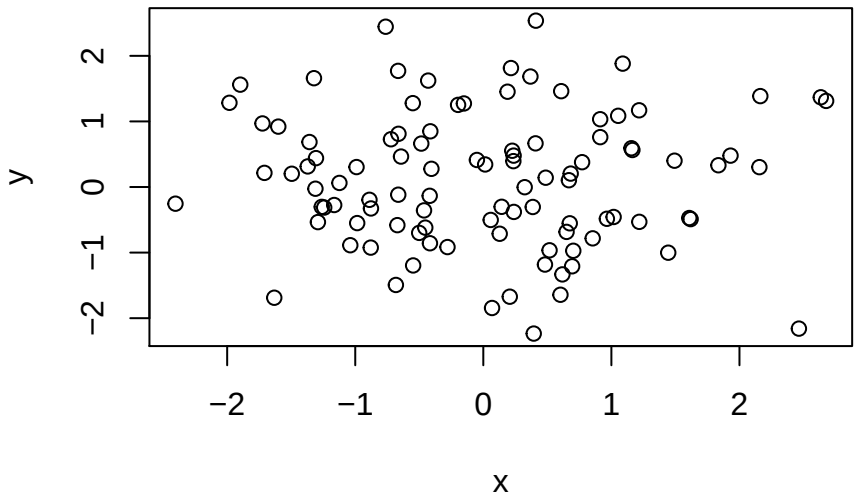
```
## [1] 0.8560768
```

```
sd(y)
```

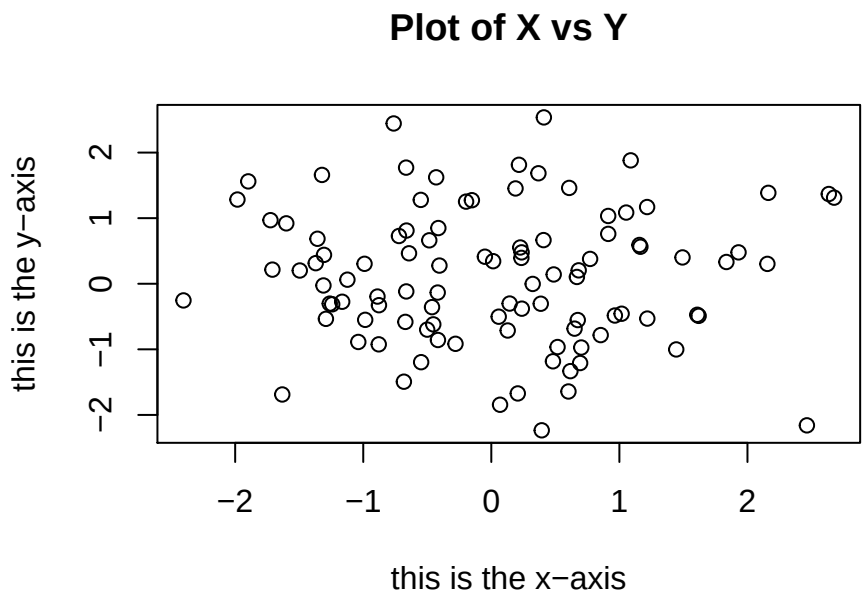
```
## [1] 0.8560768
```

Graphics

```
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()
```

```
## pdf
##   2
```

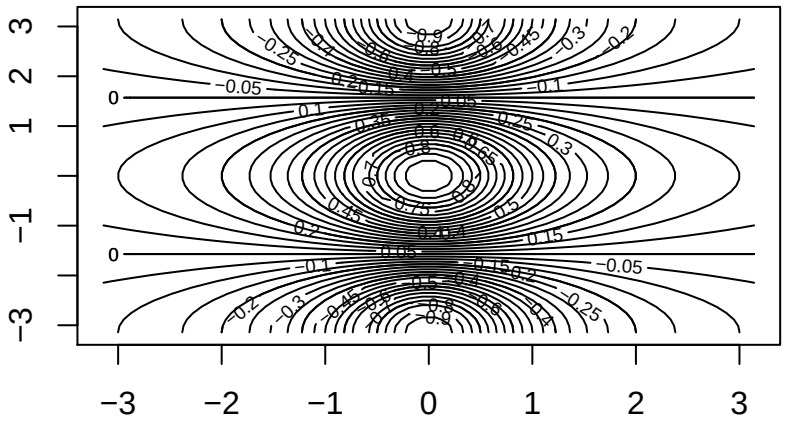
```
x=seq(1,10)
x
```

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

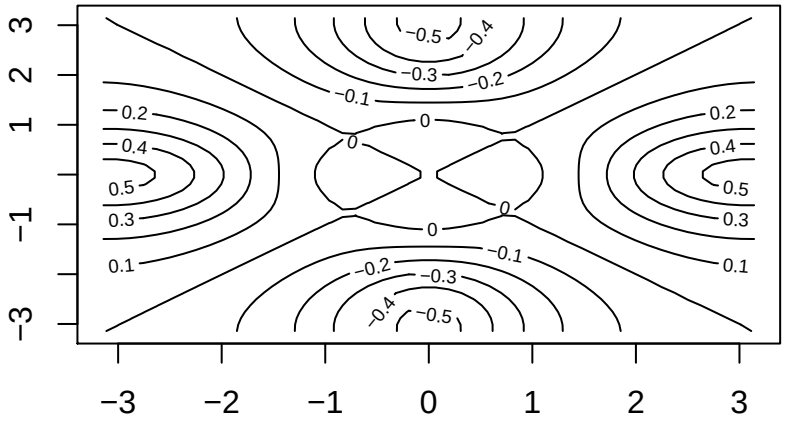
```
x=1:10
x
```

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

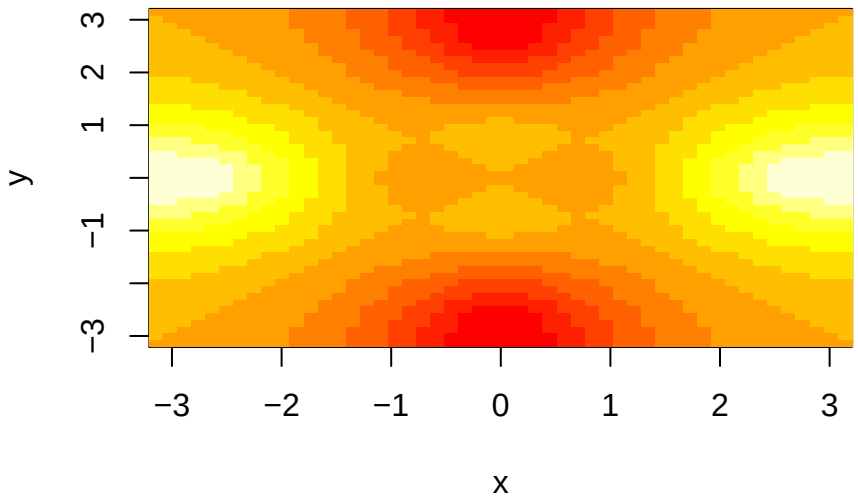
```
x=seq(-pi,pi,length=50)
y=x
f=outer(x,y,function(x,y)cos(y)/(1+x^2))
contour(x,y,f)
contour(x,y,f,nlevels=45,add=T)
```



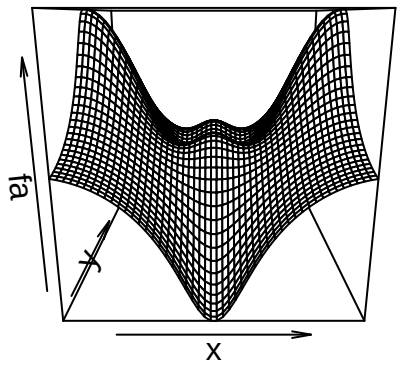
```r
# nlevels: number of contour levels desired
fa=(f-t(f))/2
contour(x,y,fa,nlevels=15)
```



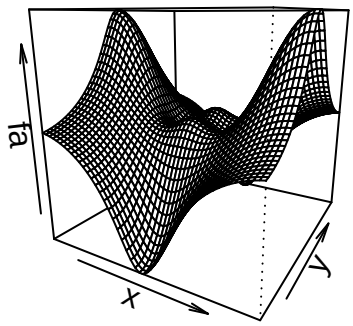
```r
image(x,y,fa)
```



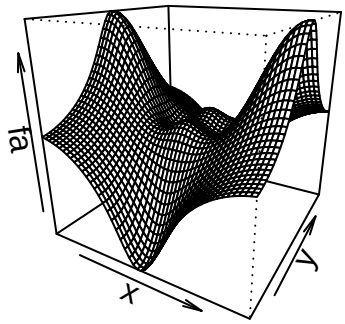

```
persp(x,y,fa)
```



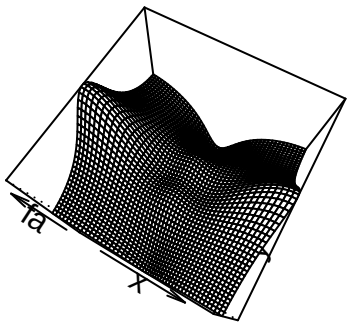

```
persp(x,y,fa,theta=30)
```



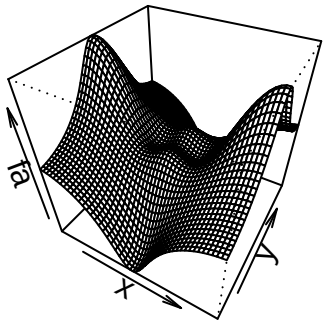

```
persp(x,y,fa,theta=30,phi=20)
```




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




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



Indexing Data

```
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
```

row column

```
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
```

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

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

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

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

```
A[1,]
```

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

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

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

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

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

```
dim(A)
```

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

Installing Packages

Package "ISLR" is a dataset package formulated by the author of the textbook. If you are to try the examples of the book, you have to install the package first.

```
# install.packages("ISLR",repos="http://cran.us.r-project.org")
library(ISLR)
```

Loading Data

Two methods to load Auto dataset:

1. loads specified data sets by data(). For example, Auto data set is included in the "ISLR" package. We need library("ISLR") at first.
2. reads a file from the working directory.

- Dataset available online: [http://www-bcf.usc.edu/~gareth/ISL/data.html]
- Change directory using: setwd("your_own_working_directory")
- choose file using file.choose()

```
# setwd('data_sets')
Auto=read.table("Auto.data")
#fix(Auto)
Auto=read.table("Auto.data",header=T,na.strings="?")
#fix(Auto)
Auto=read.csv("Auto.csv",header=T,na.strings="?")
# alt: Auto = read.csv(file.choose(),header=T, na.strings="?")
#fix(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)
```

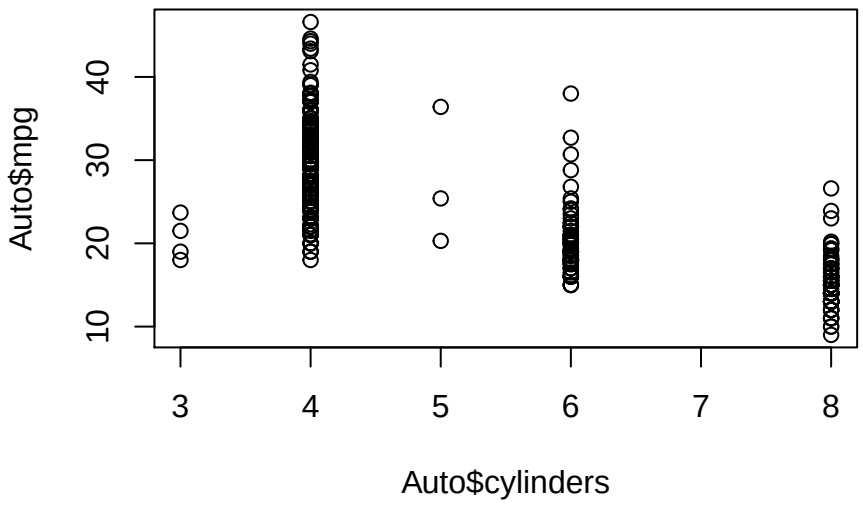
```
## [1] "mpg"          "cylinders"    "displacement" "horsepower"
## [5] "weight"       "acceleration" "year"         "origin"
## [9] "name"
```

Writing Data

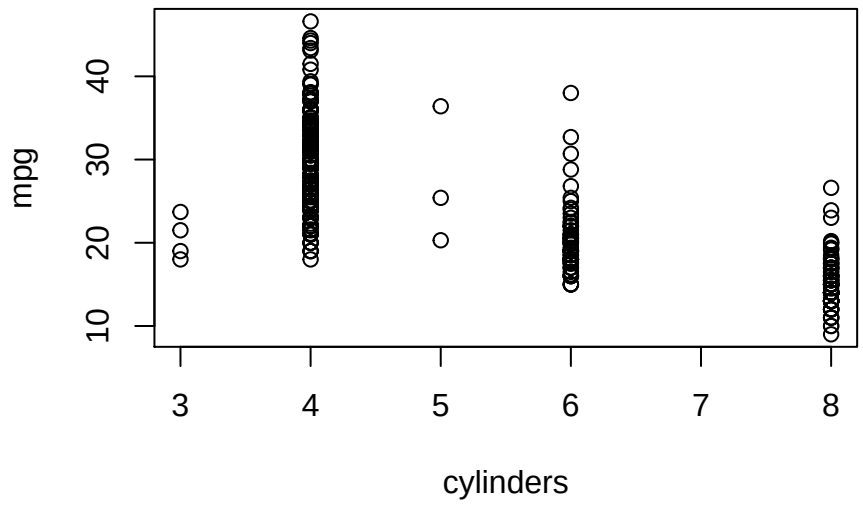
```
write.table(Auto, file="newauto.txt",col.names=TRUE, row.names=FALSE)
```

Additional Graphical

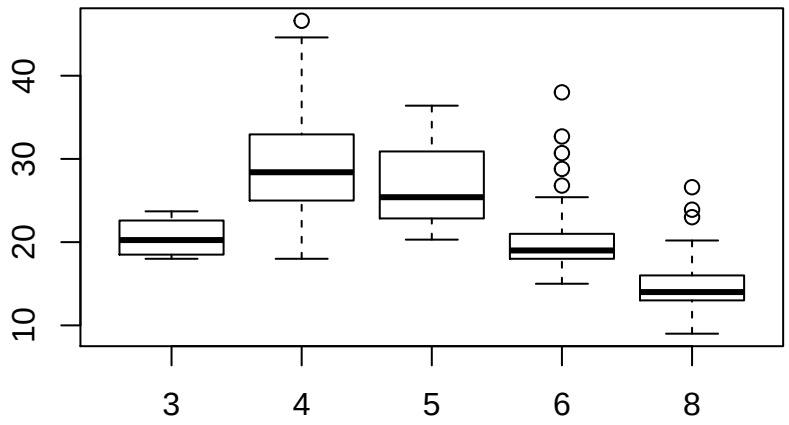
```
# plot(cylinders, mpg) error!
plot(Auto$cylinders, Auto$mpg)
```



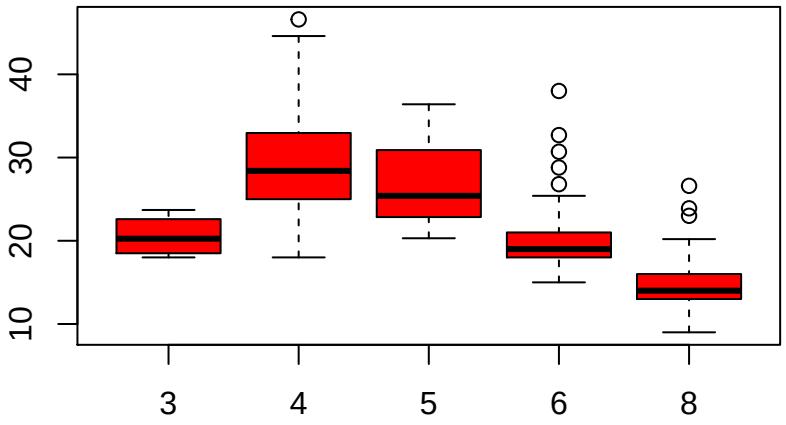

```
attach(Auto)
# The database is attached to the R search path, so objects in the database can be
# accessed by simply giving their names.
plot(cylinders, mpg)
```



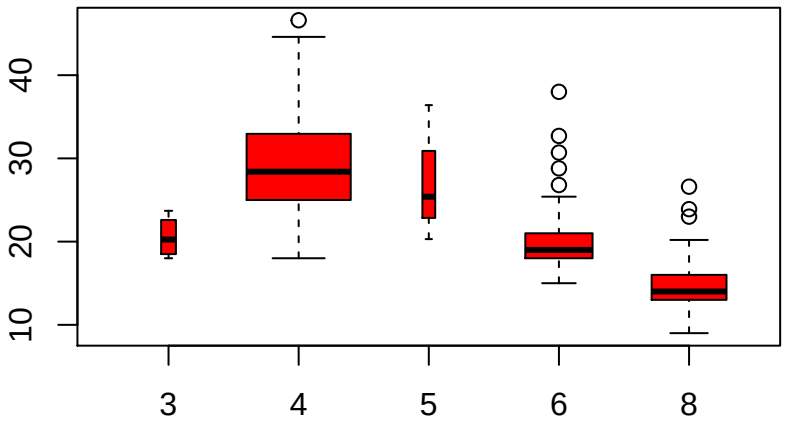

```
cylinders=as.factor(cylinders)
plot(cylinders, mpg)
```



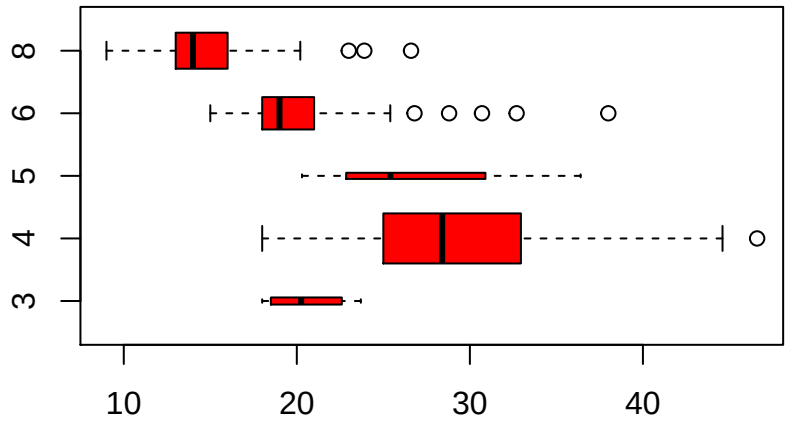

```
plot(cylinders, mpg, col="red")
```



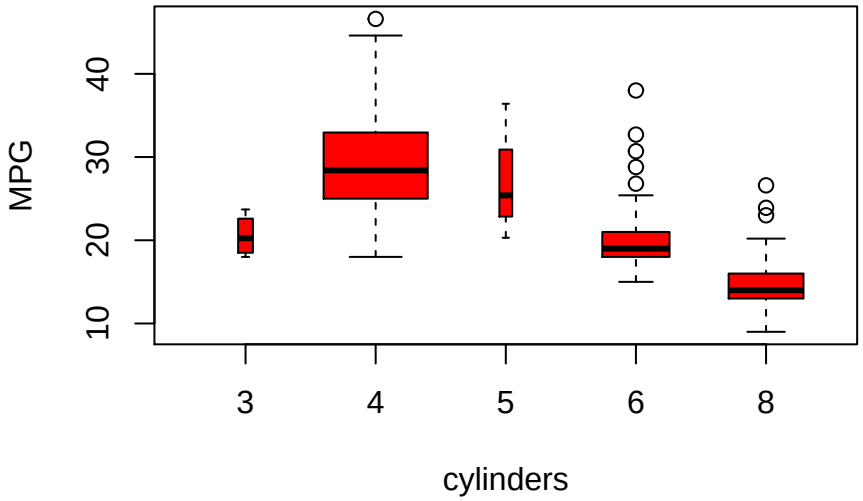

```
# varwidth=TRUE makes boxplot widths proportional to the square root of the sample size
plot(cylinders, mpg, col="red", varwidth=T)
```



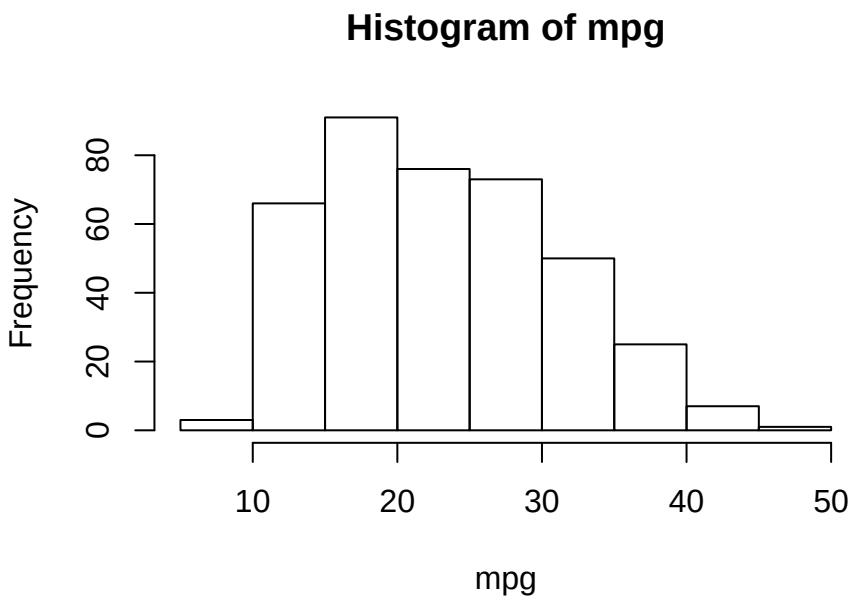

```
plot(cylinders, mpg, col="red", varwidth=T,horizontal=T)
```



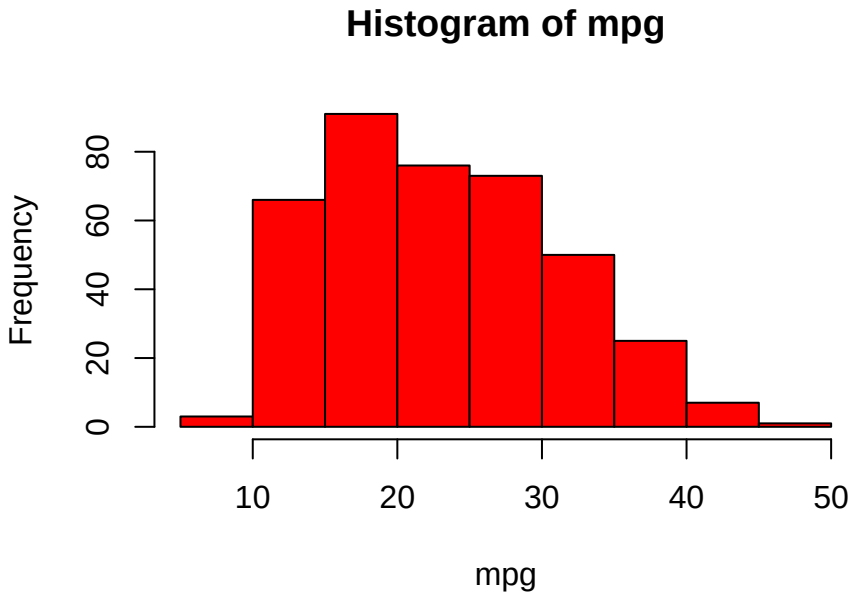

```
plot(cylinders, mpg, col="red", varwidth=T, xlab="cylinders", ylab="MPG")
```



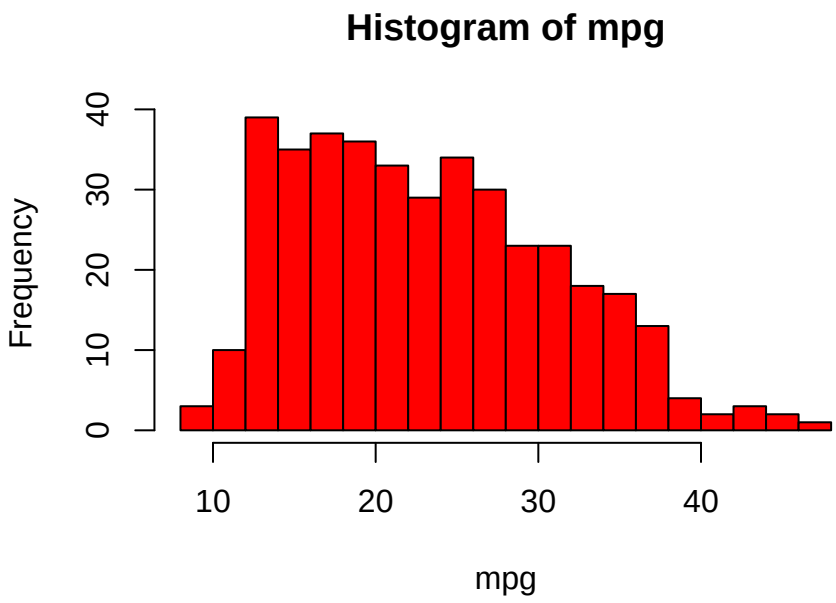

```
hist(mpg)
```



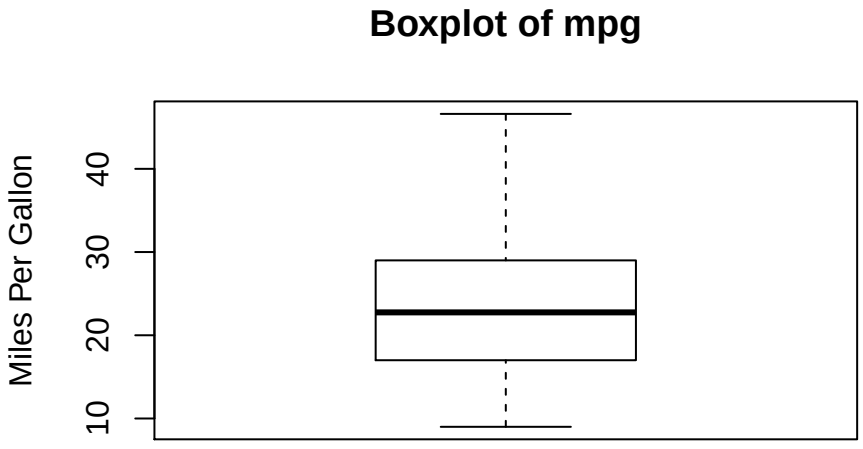

```
hist(mpg,col=2)
```



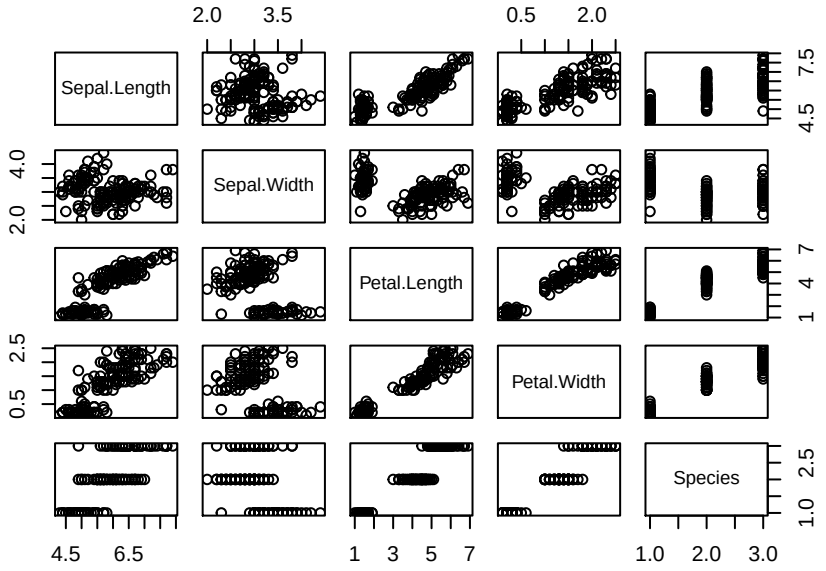

```
hist(mpg,col=2,breaks=15)
```



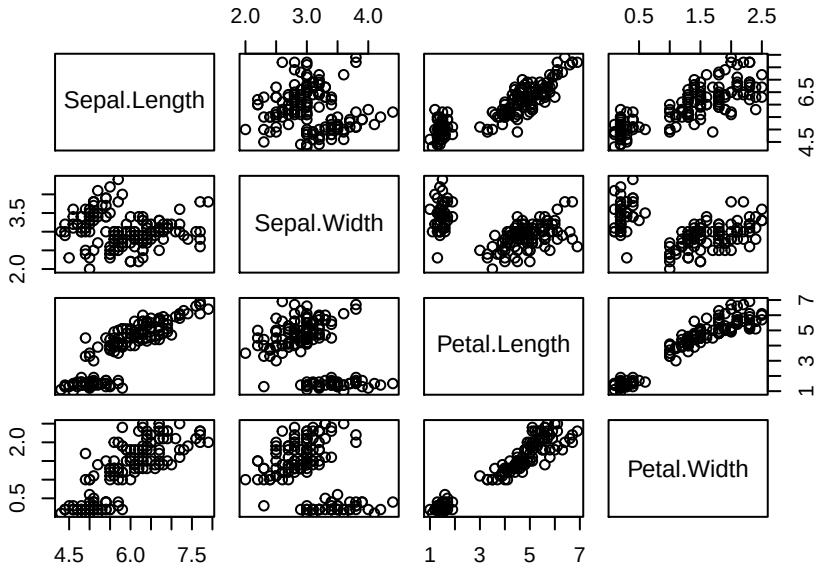

```r
boxplot(Auto$mpg,ylab = "Miles Per Gallon",main = "Boxplot of mpg")
```



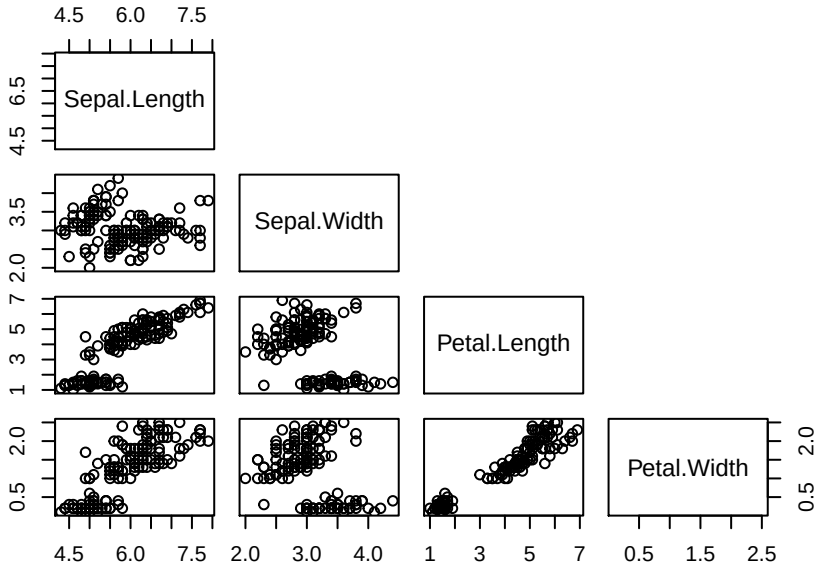

```r
# produce scatter plot matrix
pairs(iris)
```




```
pairs(~Sepal.Length + Sepal.Width + Petal.Length + Petal.Width,iris)
pairs(iris[1:4])
```



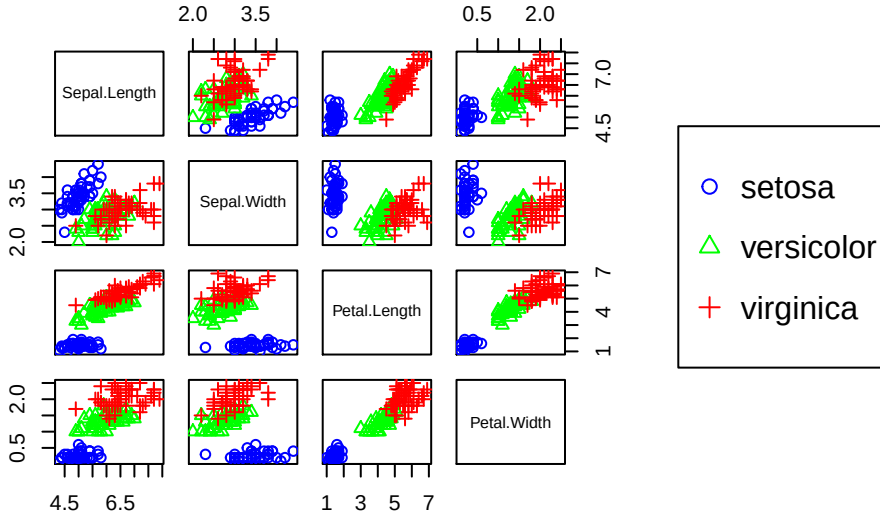

```
# show only lower triangle
pairs(iris[1:4],upper.panel = NULL)
```



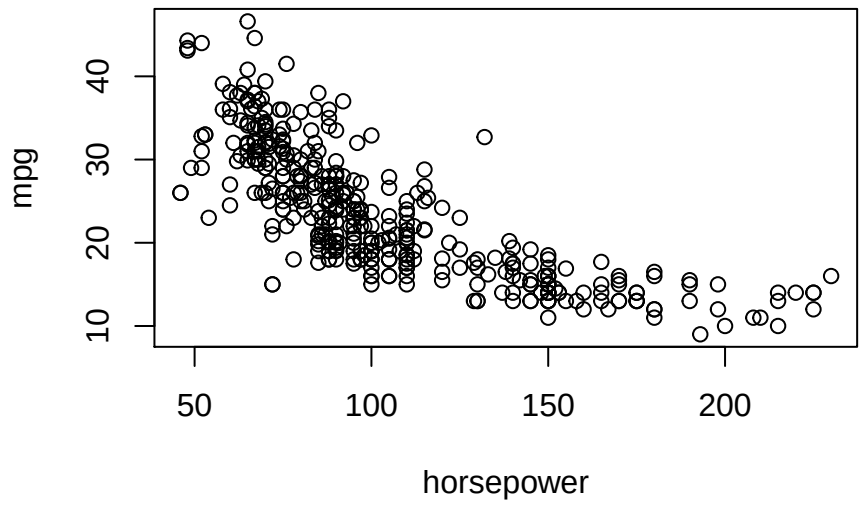

```
# a little bit fancy?
pairs(iris[1:4], col=c("blue", "green","red")[iris$Species],
      oma=c(4,4,6,12), pch=c(1,2,3)[iris$Species])
par(xpd=TRUE)
legend(0.85, 0.7, as.vector(unique(iris$Species)),
       col=c("blue", "green", "red"), pch=1:3)
```

puter margin area

current setting xpd = TRUE tells R that it is OK to plot outside the region horiz = TRUE

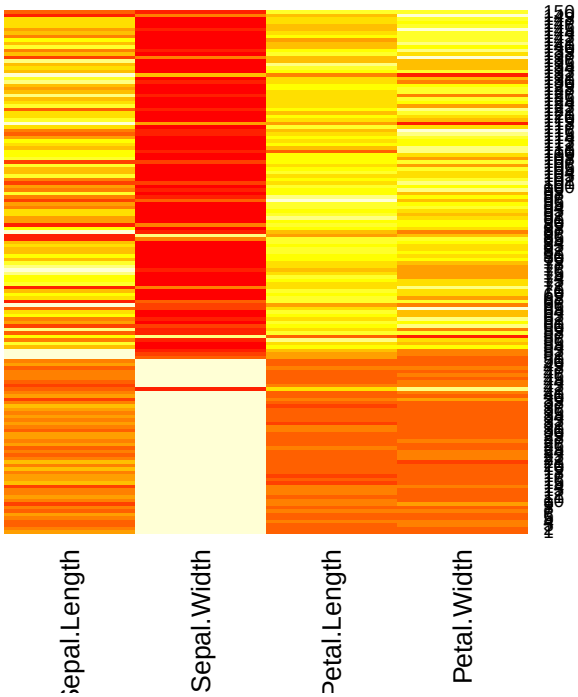



```
plot(horsepower,mpg)
identify(horsepower,mpg,name)
```

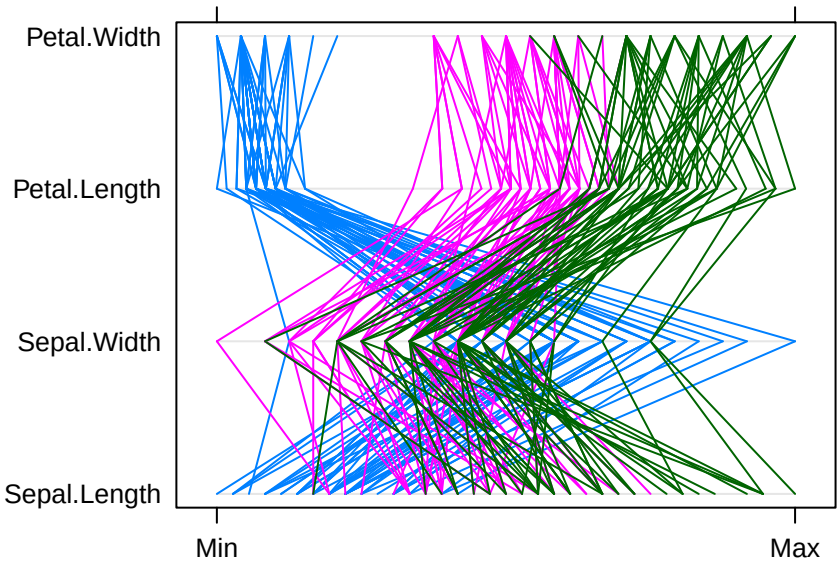



```
## integer(0)
```

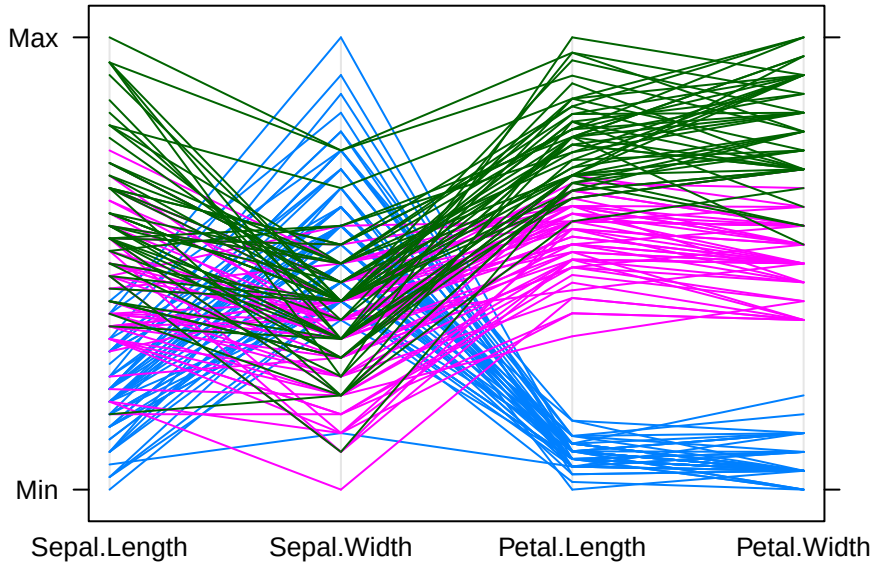
```
# heatmap is a graphical representation of data where the individual
# values contained in a matrix are represented as colors.
# We need matrix input for the function heatmap()
# and standardize the matrix before using heatmap.
dataMatrix = as.matrix(iris[,-5])
heatmap(scale(dataMatrix),Rowv=NA, Colv=NA, keep.dendro = FALSE,cexCol=1)
# parallel coordinate plot for iris dataset
# parallelplot() function is contained in "lattice" package
# install.packages("lattice")
library(lattice)
```



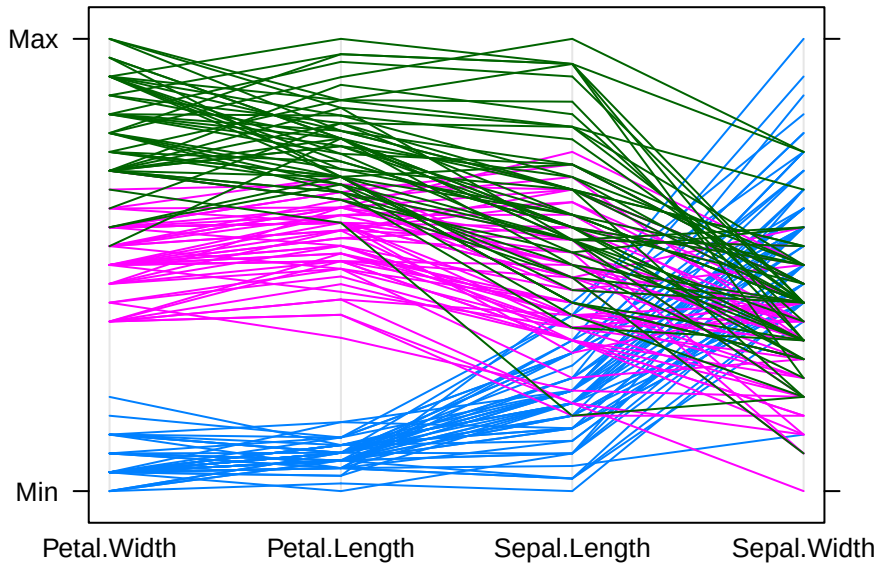

```
parallelplot(~ iris[1:4],group=Species,data=iris)
```




```
parallelplot(~ iris[1:4], iris, groups = Species, horizontal.axis = FALSE)
```




```
parallelplot(~ iris[c(4,3,1,2)], iris, groups = Species, horizontal.axis = FALSE)
```



Numerical Summaries

```
summary(Auto)
```

```
##       mpg          cylinders      displacement     horsepower
```

```
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0
##
##      weight      acceleration        year           origin
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000
##  1st Qu.:2225   1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000
##  Median :2804   Median :15.50   Median :76.00   Median :1.000
##  Mean   :2978   Mean   :15.54   Mean   :75.98   Mean   :1.577
##  3rd Qu.:3615   3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000
##
##                  name
##  amc matador       :  5
##  ford pinto        :  5
##  toyota corolla    :  5
##  amc gremlin       :  4
##  amc hornet        :  4
##  chevrolet chevette:  4
##  (Other)           :365
```

```
# origin is a categorical variable
Auto$origin = as.factor(Auto$origin)
summary(mpg)
```

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

```
data(iris)
mean(iris$Sepal.Length)
```

```
## [1] 5.843333
```

```
median(iris$Sepal.Length)
```

```
## [1] 5.8
```

```
var(iris$Sepal.Length)
```

```
## [1] 0.6856935
```

```
sd(iris$Sepal.Length)
```

```
## [1] 0.8280661
```

```
range(iris$Sepal.Length)
```

```
## [1] 4.3 7.9
```

```
quantile(iris$Sepal.Length)
```

```
##   0%  25%  50%  75% 100%
##  4.3  5.1  5.8  6.4  7.9
```

```
quantile(iris$Sepal.Length,.25)
```

```
## 25%
```

```
## 5.1
```

Subsetting of a dataframe

```
head(Auto)
```

```
##   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
## 5  17         8          302        140   3449         10.5   70      1
## 6  15         8          429        198   4341         10.0   70      1
##                       name
## 1 chevrolet chevelle malibu
## 2         buick skylark 320
## 3        plymouth satellite
## 4             amc rebel sst
## 5               ford torino
## 6          ford galaxie 500
```

```
tail(Auto)
```

```
##     mpg cylinders displacement horsepower weight acceleration year origin
## 392  27         4          151         90   2950         17.3   82      1
## 393  27         4          140         86   2790         15.6   82      1
## 394  44         4           97         52   2130         24.6   82      2
## 395  32         4          135         84   2295         11.6   82      1
## 396  28         4          120         79   2625         18.6   82      1
## 397  31         4          119         82   2720         19.4   82      1
##                 name
## 392 chevrolet camaro
## 393  ford mustang gl
## 394        vw pickup
## 395    dodge rampage
## 396      ford ranger
## 397       chevy s-10
```

```
col2 = Auto[,2]
row1 = Auto[1,]
col23 = Auto[,c(2,3)]
sum(col2==5)
```

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

```
which(col2==5)
```

```
## [1] 273 296 326
```

```
setosa = iris[which(iris$Species=="setosa"),]
dim(setosa)
```

```
## [1] 50  5
```