Chapter 2 Lab: Introduction to R
2.3.1 Basic Commands

Code	Console execution

#vector of numbers
x <- c(1,3,2,5)
x
x = c(1,6,2)
x
 	
y = c(1,4,3)
 	
length(x)
length(y)
x+y
	> #vector of numbers
	> 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

# list all objects - ls; rm - remove the objects listed in the arguments 	
ls()
rm(x,y)
ls()
 	
x = c(1,6,2)
x
y = c(1,4,3)
ls()
# remove all the objects in the list 	
rm(list=ls())
	# list all objects - ls; rm - remove the objects listed in the arguments
	> ls()
	[1] "A" "cylinders" "f" "fa" "x"
	"y"
	> rm(x,y)
	> ls()
	[1] "A" "cylinders" "f" "fa"
	>
	> x = c(1,6,2)
	> x
	[1] 1 6 2
	> y = c(1,4,3)
	> ls()
	[1] "A" "cylinders" "f" "fa" "x"
	"y"
	> # remove all the objects in the list
	> rm(list=ls())

	>

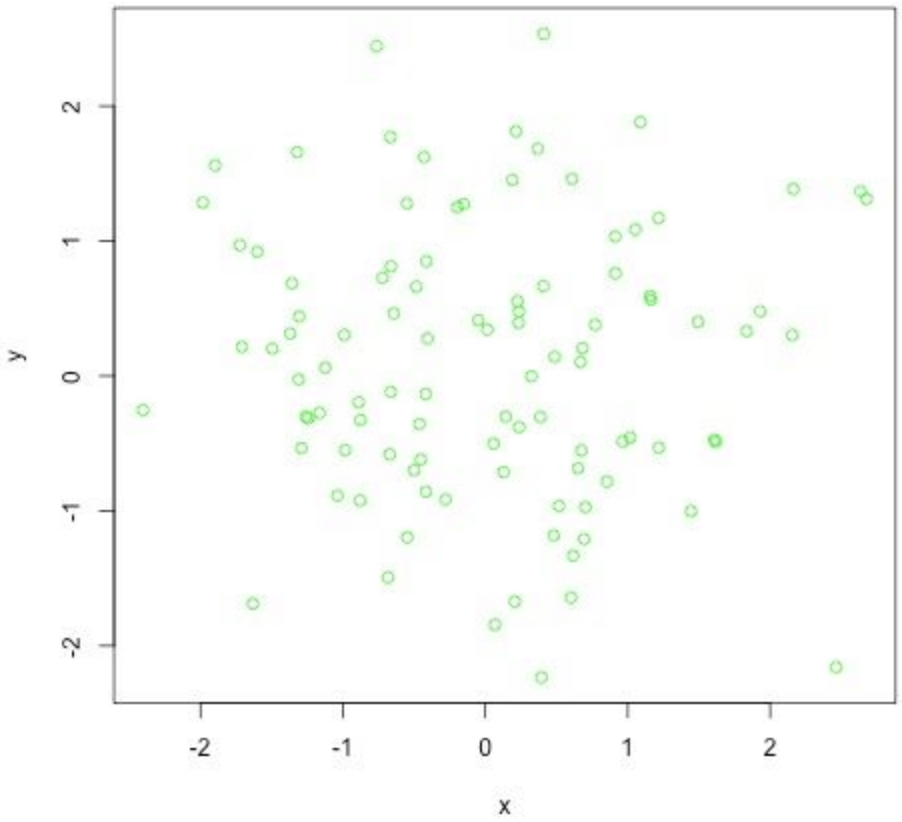
create matrix of numbers
 	
?matrix
x=matrix(data=c(1,2,3,4), nrow=2, ncol=2)
x
x=matrix(c(1,2,3,4),2,2)
#byrow = TRUE is required to populate the matrix in the order of the rows
matrix(c(1,2,3,4),2,2,byrow=TRUE)
 	
sqrt(x)
x^2
	> x=matrix(data=c(1,2,3,4), nrow=2, ncol=2)
	> x
	  [,1] [,2]
	[1,] 1 3
	[2,] 2 4
	> x=matrix(c(1,2,3,4),2,2)
	> #byrow = TRUE is required to populate the matrix in the order of the rows
	> 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

# rnorm creates standard normal random variables with a mean of 0 and SD of 1. 	
x=rnorm(50)
y=x+rnorm(50,mean=50,sd=.1)
 	
# corr is used to compute the correlation between two numbers 	
cor(x,y)
# set.seed function takes an arbitary integer argument and used in calculations involving 	
# random quantities. set.seed function uses our code to reproduce exact same set of random#. 	
set.seed(1303)
rnorm(50)
set.seed(3)
y=rnorm(100)
mean(y)
var(y)
	[1] 0.995529
	> # set.seed function takes an arbitary integer argument and used in calculations involving
	> # random quantities. set.seed function uses our code to reproduce exact same set of random#.
	> set.seed(1303)
	> rnorm(50)
	[1] -1.1439763145 1.3421293656
	2.1853904757 0.5363925179
	0.0631929665 0.5022344825
	[7] -0.0004167247 0.5658198405
	-0.5725226890 -1.1102250073
	-0.0486871234 -0.6956562176
	[13] 0.8289174803 0.2066528551
	-0.2356745091 -0.5563104914
	-0.3647543571 0.8623550343
	[19] -0.6307715354 0.3136021252
	-0.9314953177 0.8238676185
	0.5233707021 0.7069214120
	[25] 0.4202043256 -0.2690521547
	-1.5103172999 -0.6902124766

sqrt(var(y)) sd(y)	-0.1434719524 -1.0135274099 [31] 1.5732737361 0.0127465055 0.8726470499 0.4220661905 -0.0188157917 2.6157489689 [37] -0.6931401748 -0.2663217810 -0.7206364412 1.3677342065 0.2640073322 0.6321868074 [43] -1.3306509858 0.0268888182 1.0406363208 1.3120237985 -0.0300020767 -0.2500257125 [49] 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

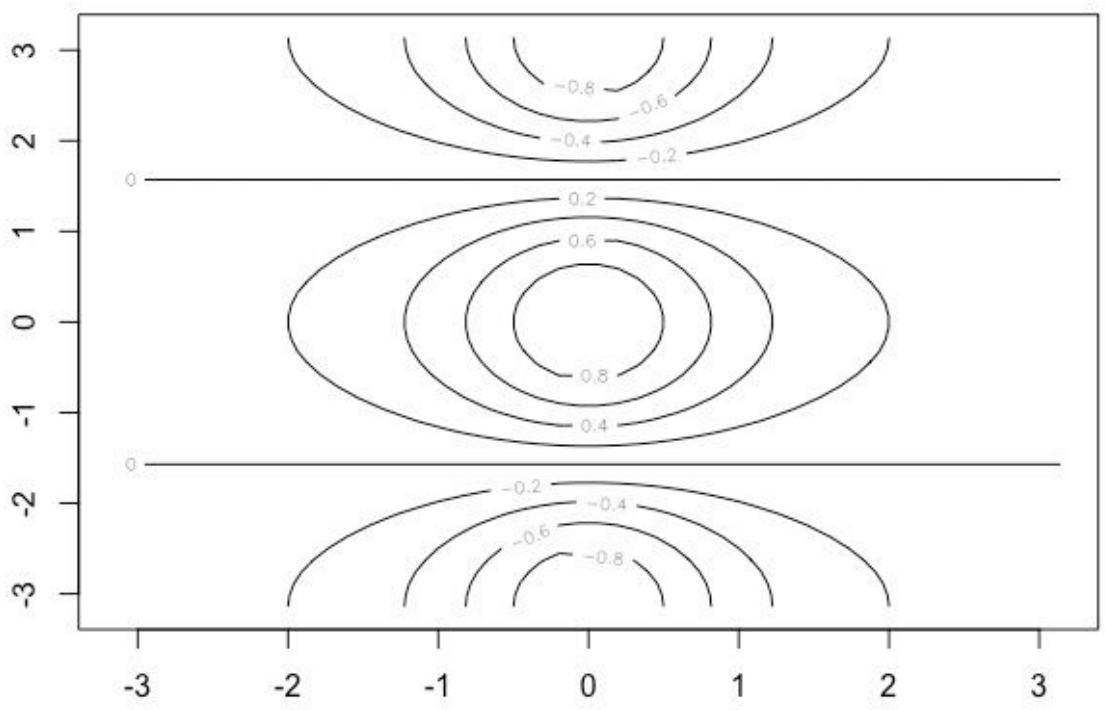
2.3.2 Graphics

```
x=rnorm(100)
y=rnorm(100)

#plot function is a primary way to plot data in R. produces a scatterplot of no.s in x vs y.
dev.new()
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() or jpeg() can be used
#pdf("Figure.pdf")
jpeg("ChandraShan_LabWeek1_plot.jpeg")
plot(x,y,col="green")
#dev.off indicates to R that we are done creating the plot
dev.off()
dev.cur()
```

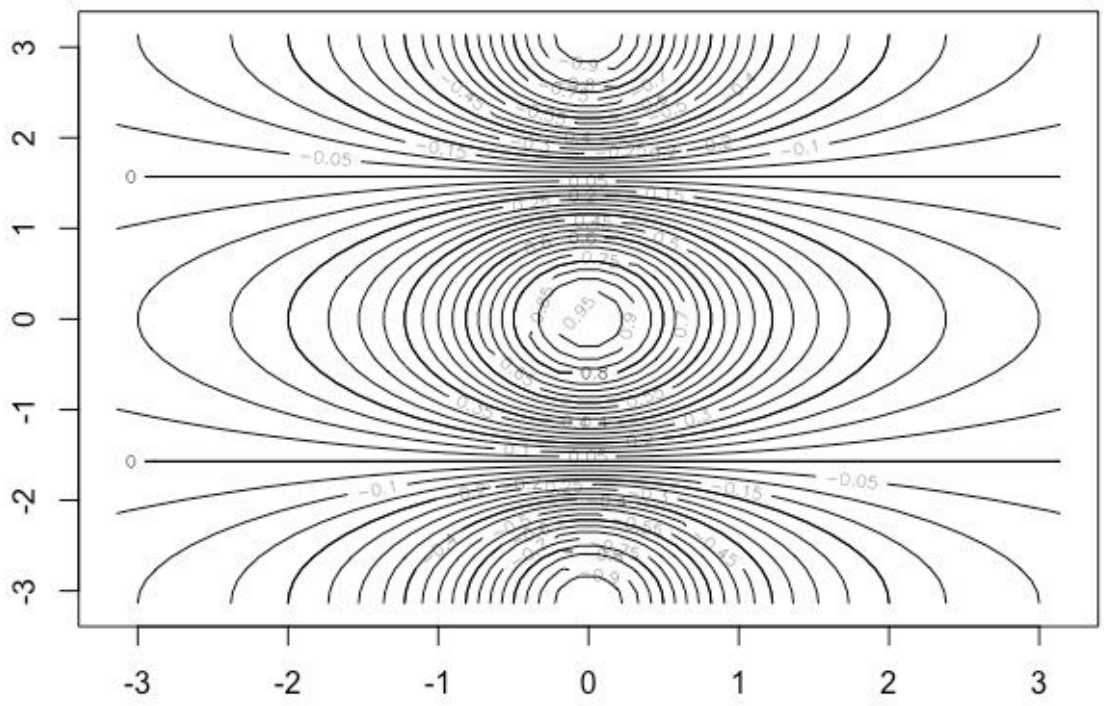


#seq used to create a sequence of numbers
x=seq(1,10)
x
x=1:10
x
#Contour function produces a contour plot inorder to represent 3D data. like a topographical map
#image function provides a color coded plot. same as contour. also known as heatmap
#persp provides a 3D plot , theta and phi control the angles to be viewed

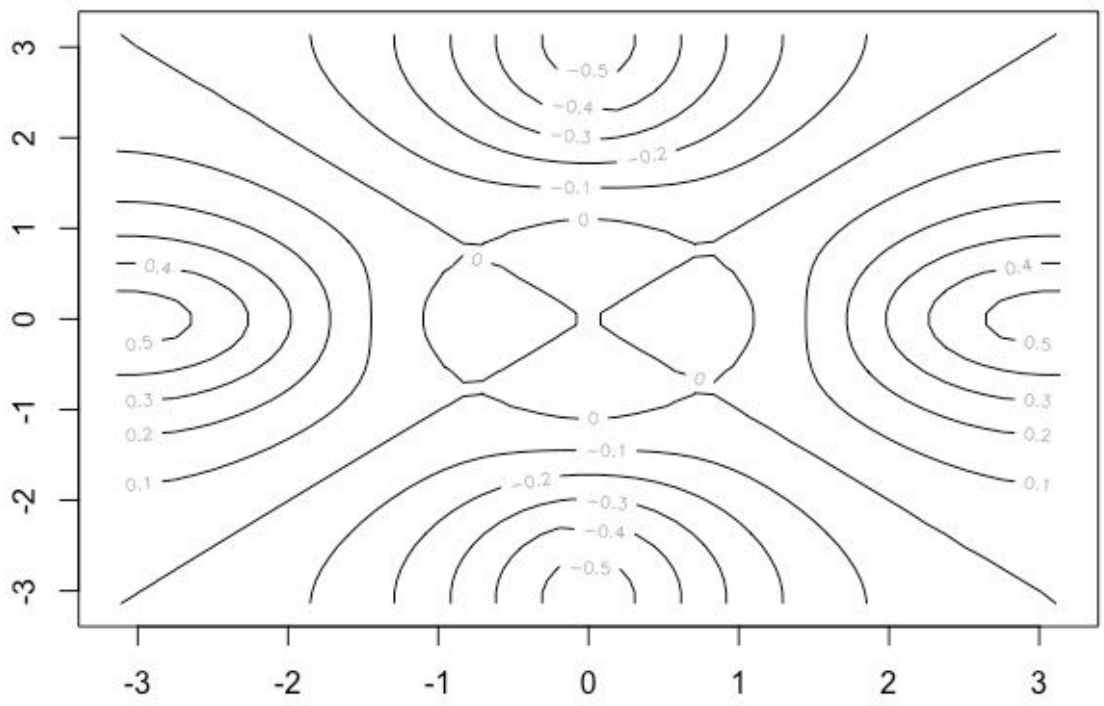
```
x=seq(-pi,pi,length=50)
y=x
f=outer(x,y,function(x,y)cos(y)/(1+x^2))
contour(x,y,f)
```



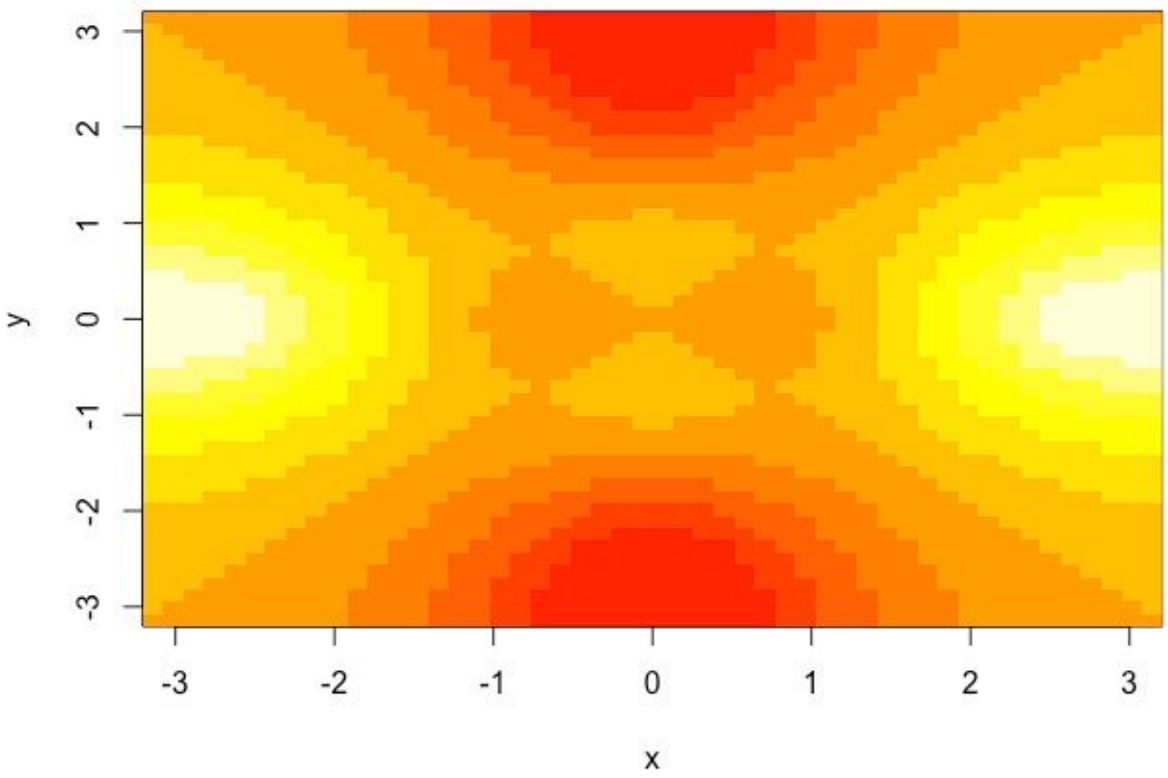

```
contour(x,y,f,nlevels=45,add=T)
```



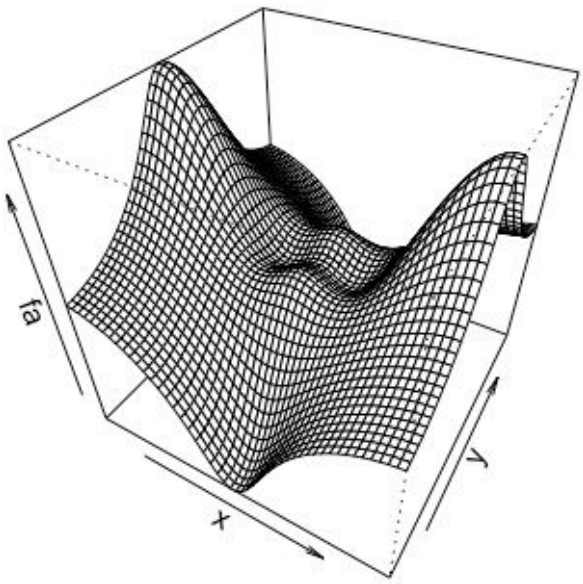

```
fa=(f-t(f))/2
contour(x,y,fa,nlevels=15)
```



image(x,y,fa)


y
3
2
1
0
-1
-2
-3
-3
-2
-1
0
1
2
3
x

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)



2.3.3 Indexing data

Code	Console output
A=matrix(1:16,4,4) A A[2,3] A[c(1,3),c(2,4)] A[1:3,2:4] A[1:2,] A[,1:2] A[1,]	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]

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

```
[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
> 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
```

2.3.4 Loading data

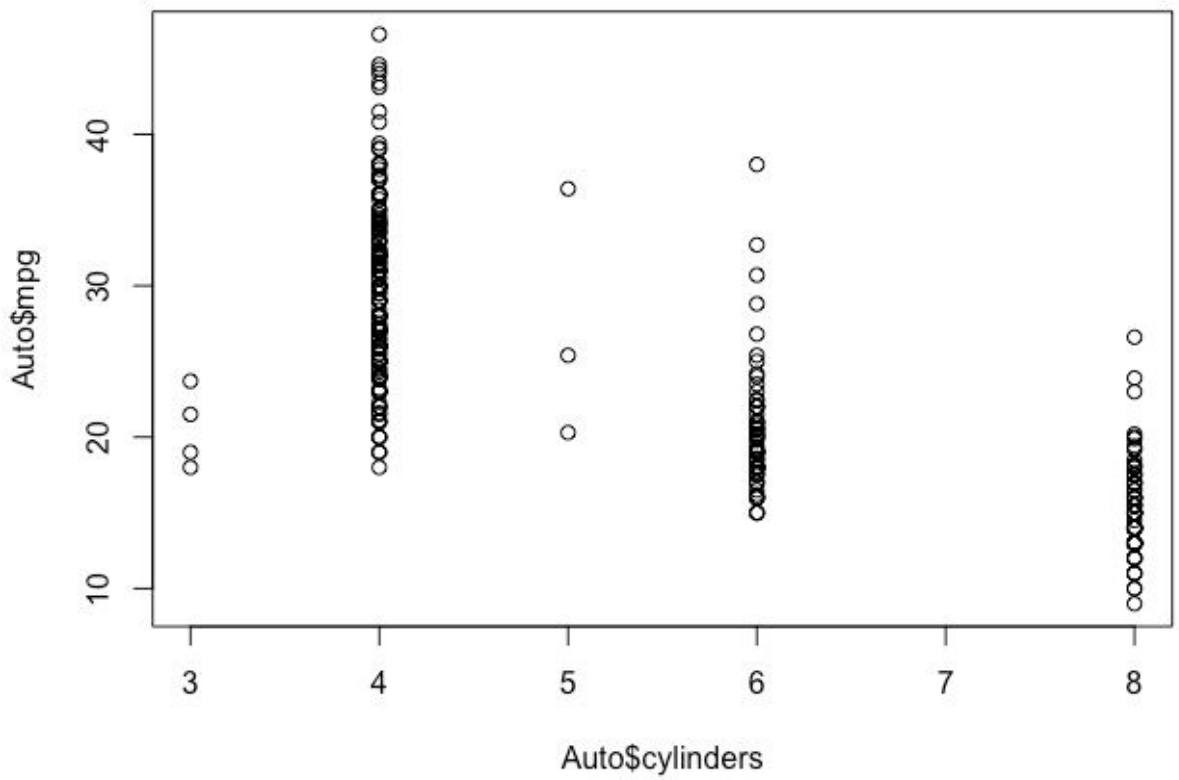
Installing ISLR package loaded the data automatically. Below shows the loaded data

```
head(Auto)
  mpg cylinders displacement horsepower weight acceleration year origin                 name
1  18         8          307        130   3504         12.0   70      1 chevrolet chevelle malibu
2  15         8          350        165   3693         11.5   70      1         buick skylark 320
3  18         8          318        150   3436         11.0   70      1        plymouth satellite
4  16         8          304        150   3433         12.0   70      1             amc rebel sst
5  17         8          302        140   3449         10.5   70      1               ford torino
6  15         8          429        198   4341         10.0   70      1          ford galaxie 500
```

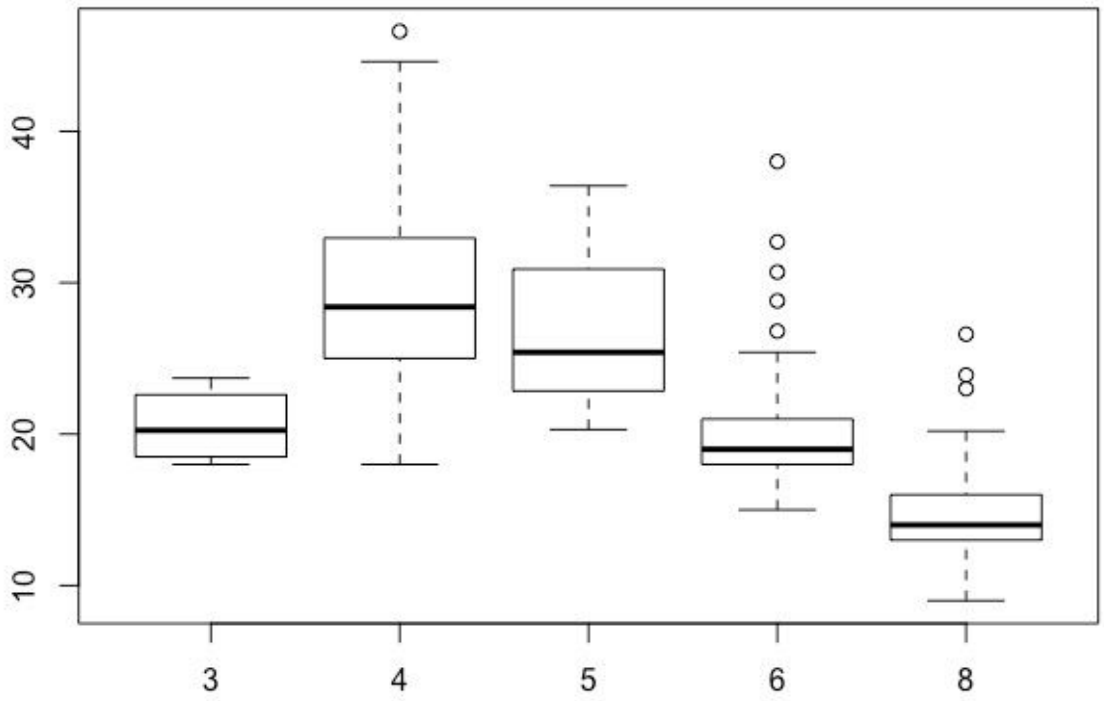
```
> names(Auto)
[1] "mpg"          "cylinders"    "displacement" "horsepower"   "weight"       "acceleration"
[7] "year"         "origin"       "name"
```

2.3.5 Additional Graphical and Numerical Summaries

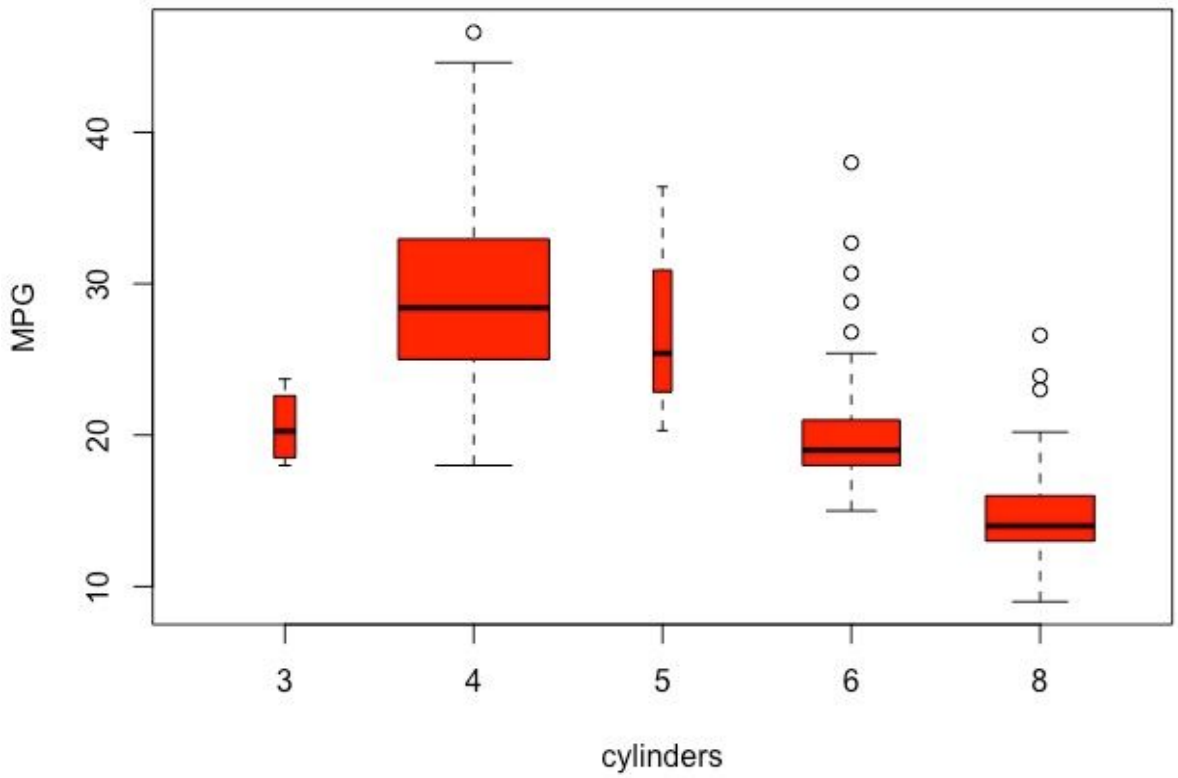
```
# below line will have error when executing as object cylinders not found
#plot(cylinders, mpg)

#attach func will tell R to make the variables in the dataframe available by name
#as.factor convert quantitative variables into qualitative variables.
#pairs create a scatterplot matrix for every pair of variables in a given dataset.
#identify identifies the rows for the selected points printed on the plot
plot(Auto$cylinders, Auto$mpg)
attach(Auto)
plot(cylinders, mpg)
```



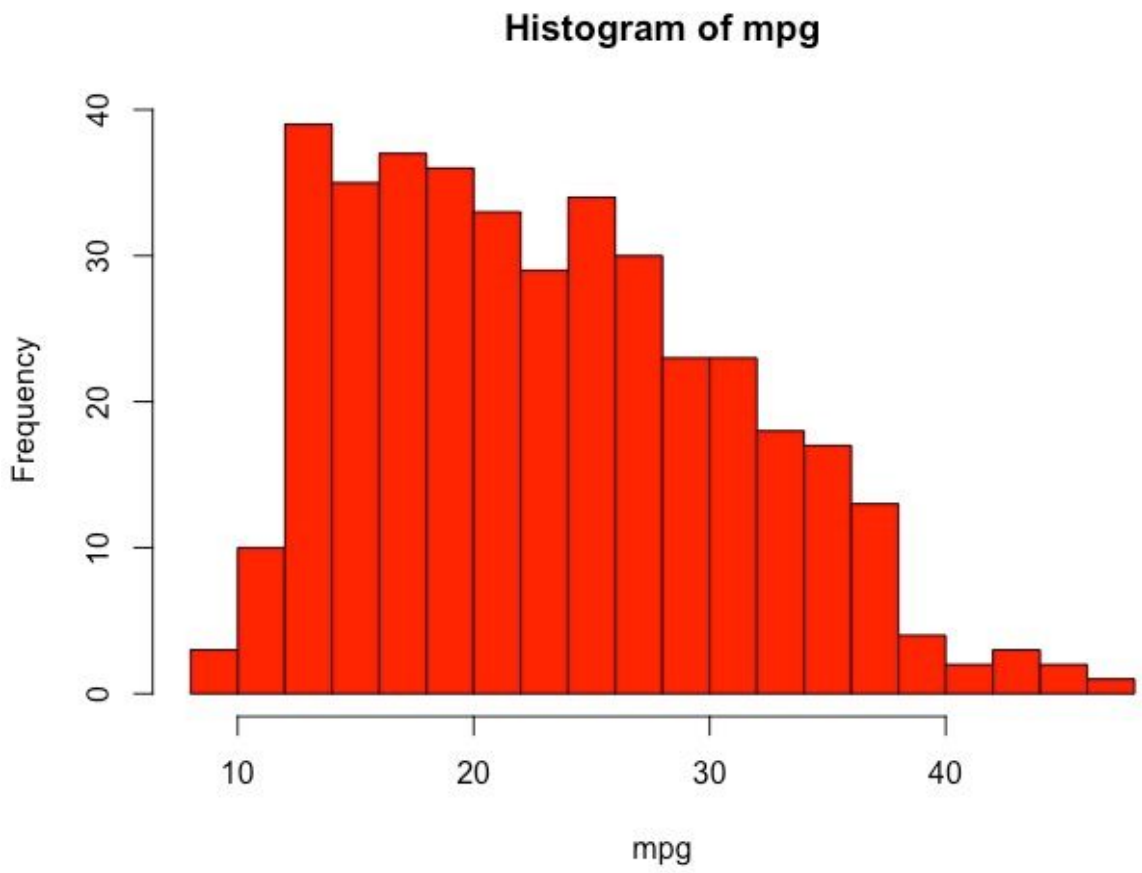
```
cylinders=as.factor(cylinders)
plot(cylinders, mpg)
```



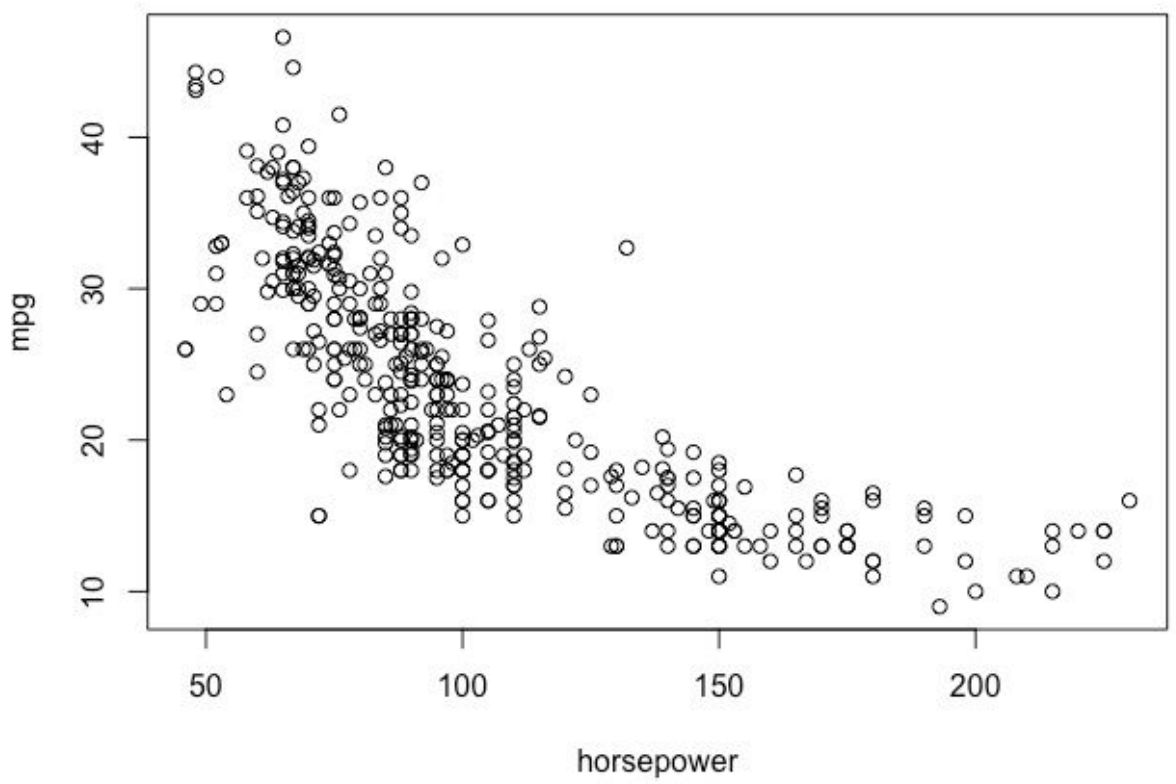

```
plot(cylinders, mpg, col="red")
plot(cylinders, mpg, col="red", varwidth=T)
plot(cylinders, mpg, col="red", varwidth=T,horizontal=T)
plot(cylinders, mpg, col="red", varwidth=T, xlab="cylinders", ylab="MPG")
```


MPG
10
20
30
40
3
4
5
6
8
cylinders

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



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




```
identify(horsepower,mpg,name)


summary(Auto)
summary(mpg)
```

```
      mpg          cylinders      displacement     horsepower       weight      acceleration
 Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0   Min.   :1613   Min.   : 8.00
 1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0   1st Qu.:2225   1st Qu.:13.78
 Median :22.75   Median :4.000   Median :151.0   Median : 93.5   Median :2804   Median :15.50
 Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5   Mean   :2978   Mean   :15.54
```

```
 3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0   3rd Qu.:3615   3rd Qu.:17.02
 Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0   Max.   :5140   Max.   :24.80

     year           origin                      name
 Min.   :70.00   Min.   :1.000   amc matador       :  5
 1st Qu.:73.00   1st Qu.:1.000   ford pinto        :  5
 Median :76.00   Median :1.000   toyota corolla    :  5
 Mean   :75.98   Mean   :1.577   amc gremlin       :  4
 3rd Qu.:79.00   3rd Qu.:2.000   amc hornet        :  4
 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
```