Assignment No	09
Title	Data pre-processing
Objective	Missing values, Data Reduction
Roll No	MCA2565

1) Data Pre-Processing

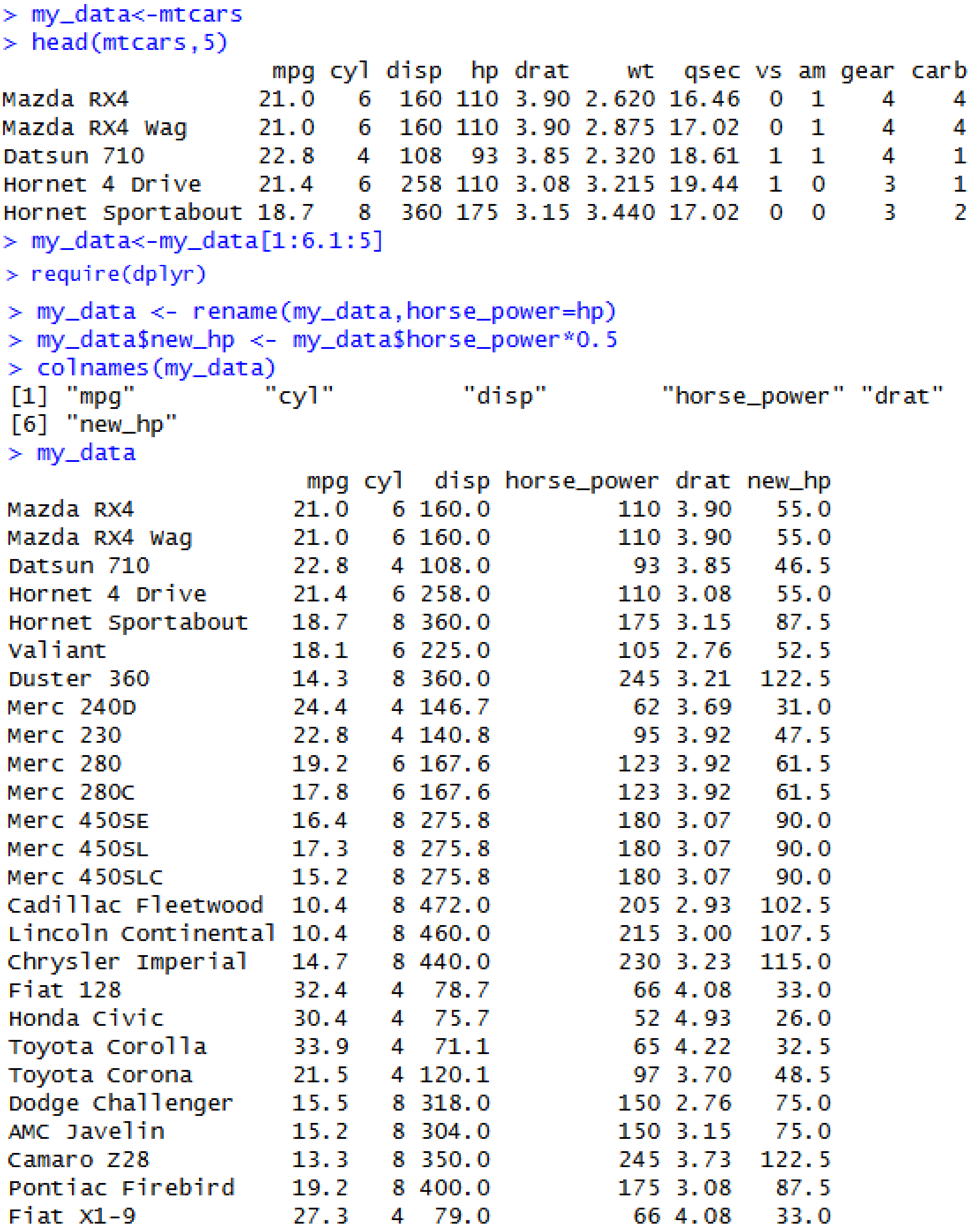
Source Code :-

Data preprocessing.R

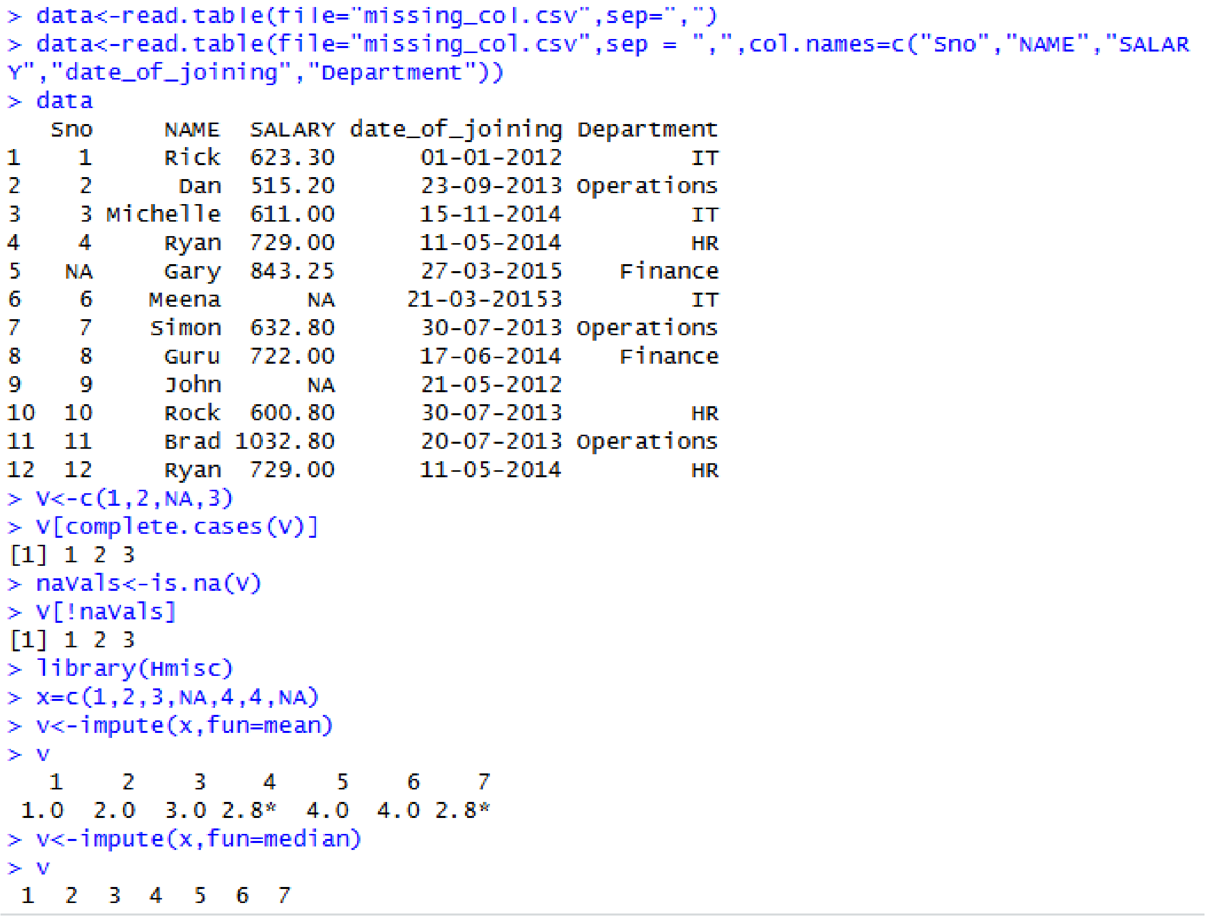
```
install.packages("dplyr")
library(dplyr)
install.packages("Hmisc")
library(Hmisc)
my_data<-mtcars
head(mtcars,5)


my_data<-my_data[1:6.1:5]
require(dplyr)
my_data <- rename(my_data,horse_power=hp)
my_data$new_hp <- my_data$horse_power*0.5
colnames(my_data)

my_data

data<-read.table(file="missing_col.csv",sep=",")
data<-read.table(file="missing_col.csv",sep =
",",col.names=c("Sno","NAME","SALARY","date_of_joining","Department"))
data

V<-c(1,2,NA,3)
V[complete.cases(V)]
naVals<-is.na(V)
V[!naVals]

library(Hmisc)
x=c(1,2,3,NA,4,4,NA)
v<-impute(x,fun=mean)
```

```
v

v<-impute(x,fun=median)
v
data1<-data.frame(Srno=c(1,2,3,NA,4,4,NA),
            Name=c("a","b","c","d","e","f","g"),
            Salary=c(400,200,NA,500,NA,800,900)
            )
v<-impute(data1$Srno,fun=mean)
v
v<-impute(data1$Salary,fun=median)
v
c1<-c("low","medium","high","low")
c1<-factor(c1,levels=c("low","medium","high"))
c1
data1<-read.csv("missing_col.csv",sep=",",col.names=
            c("Srno","Name","salary","DOJ","Department"))
View(data1)

x<-c(1,2,3,NA,4,NA,5)
#indicates which elements are missing
xn<-is.na(x)
x[!xn]
NA+4
#This will keep NA rows in data while removes them during calculation
median(x,na.rm=T)
#Return a logical vector indicating which cases are complete,i.e.,have no missing value
complete.cases(x)
is.na(data1)
datacompletecases<-data1[complete.cases(data1),]
datacompletecases

#detect if there are any NAs: any(is.na(datan)) Identify positions of NAs: which(is.na(datan$v1))
any(is.na(x))
which(is.na(data1$Srno))
na.omit(x)
```

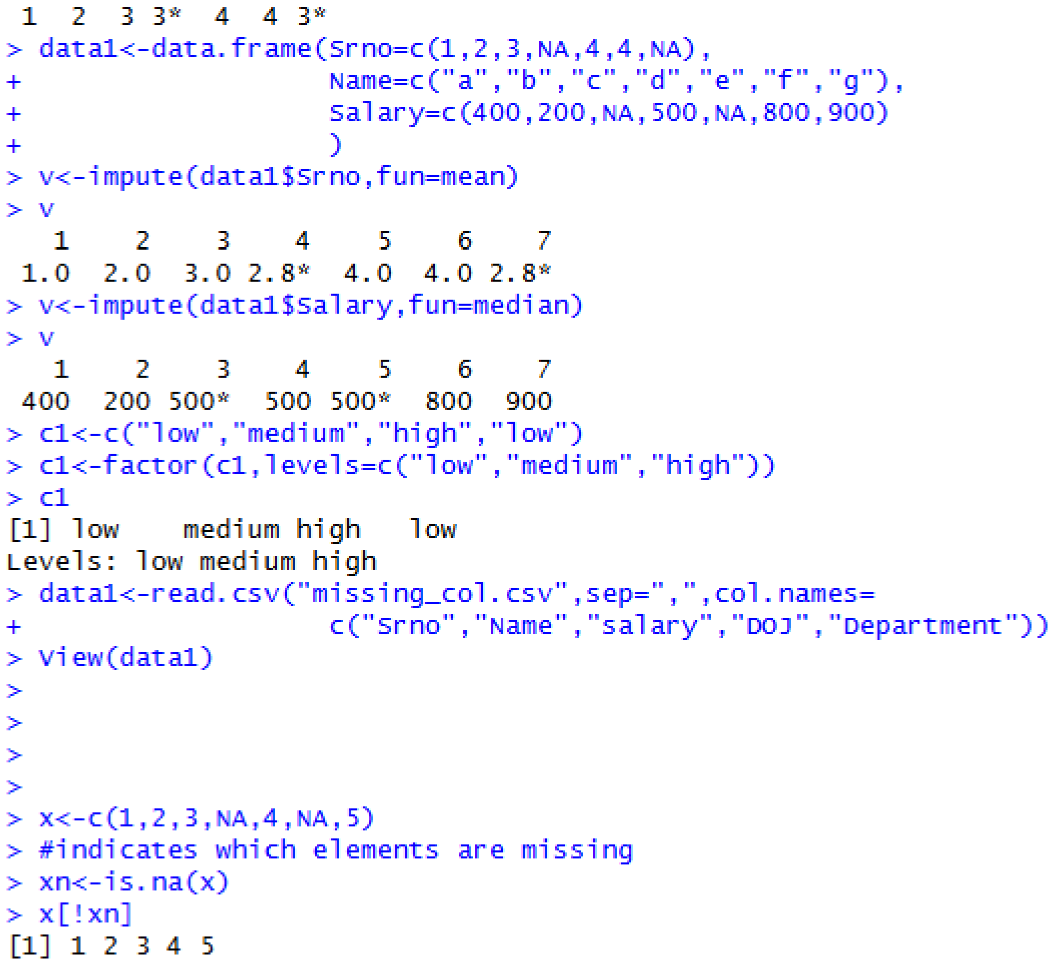
Output :-



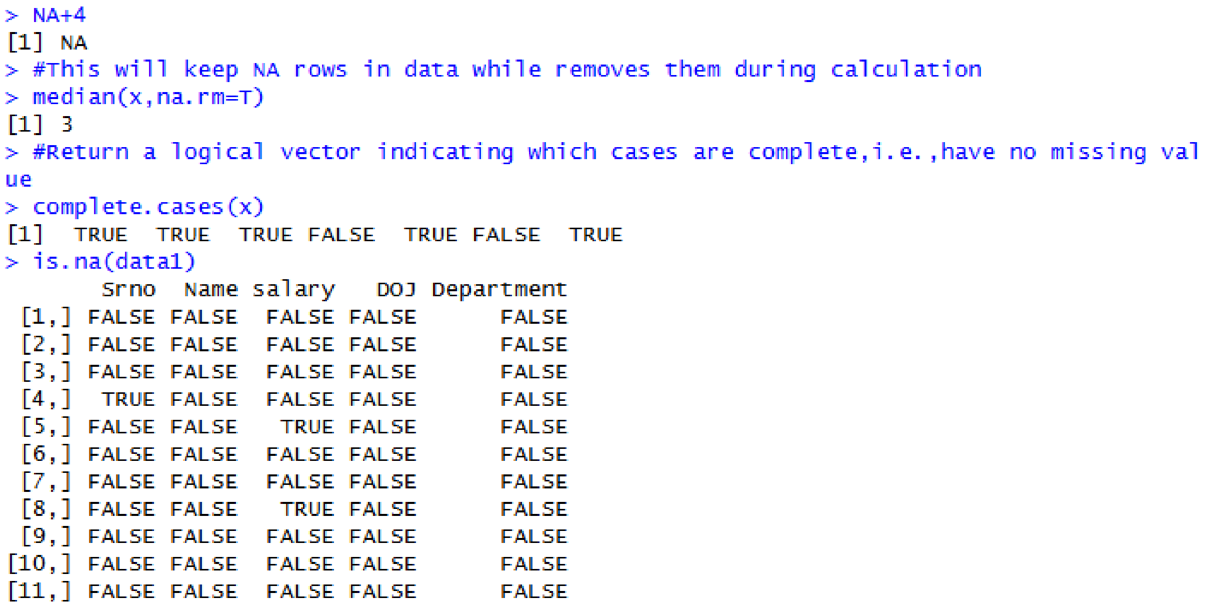
```
> my_data<-mtcars
> head(mtcars,5)
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
> my_data<-my_data[1:6.1:5]
> require(dplyr)
> my_data <- rename(my_data,horse_power=hp)
> my_data$new_hp <- my_data$horse_power*0.5
> colnames(my_data)
[1] "mpg"         "cyl"         "disp"        "horse_power" "drat"
[6] "new_hp"
> my_data
                     mpg cyl  disp horse_power drat new_hp
Mazda RX4           21.0   6 160.0         110 3.90   55.0
Mazda RX4 Wag       21.0   6 160.0         110 3.90   55.0
Datsun 710          22.8   4 108.0          93 3.85   46.5
Hornet 4 Drive      21.4   6 258.0         110 3.08   55.0
Hornet Sportabout   18.7   8 360.0         175 3.15   87.5
Valiant             18.1   6 225.0         105 2.76   52.5
Duster 360          14.3   8 360.0         245 3.21  122.5
Merc 240D           24.4   4 146.7          62 3.69   31.0
Merc 230            22.8   4 140.8          95 3.92   47.5
Merc 280            19.2   6 167.6         123 3.92   61.5
Merc 280C           17.8   6 167.6         123 3.92   61.5
Merc 450SE          16.4   8 275.8         180 3.07   90.0
Merc 450SL          17.3   8 275.8         180 3.07   90.0
Merc 450SLC         15.2   8 275.8         180 3.07   90.0
Cadillac Fleetwood  10.4   8 472.0         205 2.93  102.5
Lincoln Continental 10.4   8 460.0         215 3.00  107.5
Chrysler Imperial   14.7   8 440.0         230 3.23  115.0
Fiat 128            32.4   4  78.7          66 4.08   33.0
Honda Civic         30.4   4  75.7          52 4.93   26.0
Toyota Corolla      33.9   4  71.1          65 4.22   32.5
Toyota Corona       21.5   4 120.1          97 3.70   48.5
Dodge Challenger    15.5   8 318.0         150 2.76   75.0
AMC Javelin         15.2   8 304.0         150 3.15   75.0
Camaro Z28          13.3   8 350.0         245 3.73  122.5
Pontiac Firebird    19.2   8 400.0         175 3.08   87.5
Fiat X1-9           27.3   4  79.0          66 4.08   33.0
```



```
> data<-read.table(file="missing_col.csv",sep=",")
> data<-read.table(file="missing_col.csv",sep = ",",col.names=c("Sno","NAME","SALAR
Y","date_of_joining","Department"))
> data
   Sno     NAME  SALARY date_of_joining Department
1    1     Rick  623.30      01-01-2012         IT
2    2      Dan  515.20      23-09-2013 Operations
3    3 Michelle  611.00      15-11-2014         IT
4    4     Ryan  729.00      11-05-2014         HR
5   NA     Gary  843.25      27-03-2015    Finance
6    6    Meena      NA     21-03-20153         IT
7    7    Simon  632.80      30-07-2013 Operations
8    8     Guru  722.00      17-06-2014    Finance
9    9     John      NA      21-05-2012
10  10     Rock  600.80      30-07-2013         HR
11  11     Brad 1032.80      20-07-2013 Operations
12  12     Ryan  729.00      11-05-2014         HR
> v<-c(1,2,NA,3)
> v[complete.cases(v)]
[1] 1 2 3
> navals<-is.na(v)
> v[!navals]
[1] 1 2 3
> library(Hmisc)
> x=c(1,2,3,NA,4,4,NA)
> v<-impute(x,fun=mean)
> v
   1    2    3    4    5    6    7
 1.0  2.0  3.0 2.8*  4.0  4.0 2.8*
> v<-impute(x,fun=median)
> v
 1  2  3  4  5  6  7
```



```
  1   2   3 3*   4   4 3* 
> data1<-data.frame(Srno=c(1,2,3,NA,4,4,NA),
+                   Name=c("a","b","c","d","e","f","g"),
+                   Salary=c(400,200,NA,500,NA,800,900)
+                   )
> v<-impute(data1$Srno,fun=mean)
> v
   1    2    3    4    5    6    7 
 1.0  2.0  3.0 2.8*  4.0  4.0 2.8* 
> v<-impute(data1$Salary,fun=median)
> v
   1    2    3    4    5    6    7 
 400  200 500*  500 500*  800  900 
> c1<-c("low","medium","high","low")
> c1<-factor(c1,levels=c("low","medium","high"))
> c1
[1] low    medium high   low   
Levels: low medium high
> data1<-read.csv("missing_col.csv",sep=",",col.names=
+                   c("Srno","Name","salary","DOJ","Department"))
> View(data1)
> 
> 
> 
> 
> x<-c(1,2,3,NA,4,NA,5)
> #indicates which elements are missing
> xn<-is.na(x)
> x[!xn]
[1] 1 2 3 4 5
```



```
> NA+4
[1] NA
> #This will keep NA rows in data while removes them during calculation
> median(x,na.rm=T)
[1] 3
> #Return a logical vector indicating which cases are complete,i.e.,have no missing val
ue
> complete.cases(x)
[1]  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE
> is.na(data1)
       Srno  Name salary   DOJ Department
 [1,] FALSE FALSE  FALSE FALSE      FALSE
 [2,] FALSE FALSE  FALSE FALSE      FALSE
 [3,] FALSE FALSE  FALSE FALSE      FALSE
 [4,]  TRUE FALSE  FALSE FALSE      FALSE
 [5,] FALSE FALSE   TRUE FALSE      FALSE
 [6,] FALSE FALSE  FALSE FALSE      FALSE
 [7,] FALSE FALSE  FALSE FALSE      FALSE
 [8,] FALSE FALSE   TRUE FALSE      FALSE
 [9,] FALSE FALSE  FALSE FALSE      FALSE
[10,] FALSE FALSE  FALSE FALSE      FALSE
[11,] FALSE FALSE  FALSE FALSE      FALSE
```

```
> datacompletecases<-data1[complete.cases(data1),]
> datacompletecases
   Srno     Name salary        DOJ Department
1     2      Dan  515.2 23-09-2013 Operations
2     3 Michelle  611.0 15-11-2014         IT
3     4     Ryan  729.0 11-05-2014         HR
6     7    Simon  632.8 30-07-2013 Operations
7     8     Guru  722.0 17-06-2014    Finance
9    10     Rock  600.8 30-07-2013         HR
10   11     Brad 1032.8 20-07-2013 Operations
11   12     Ryan  729.0 11-05-2014         HR
> #detect if there are any NAs: any(is.na(datan)) Identify positions of NAs: which(is.n
a(datan$v1))
> any(is.na(x))
[1] TRUE
> which(is.na(data1$Srno))
[1] 4
> na.omit(x)
[1] 1 2 3 4 5
attr(,"na.action")
[1] 4 6
attr(,"class")
[1] "omit"
```

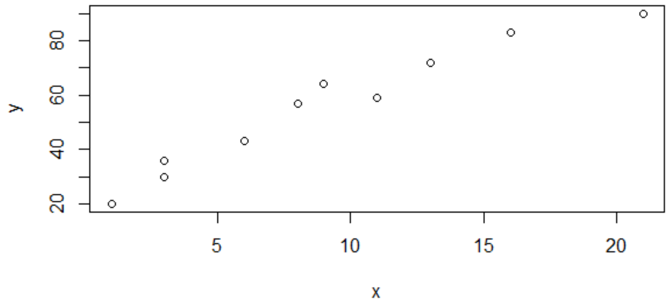
Assignment No	10
Title	Data mining Regression
Objective	Linear Regression
Roll No	MCA2565

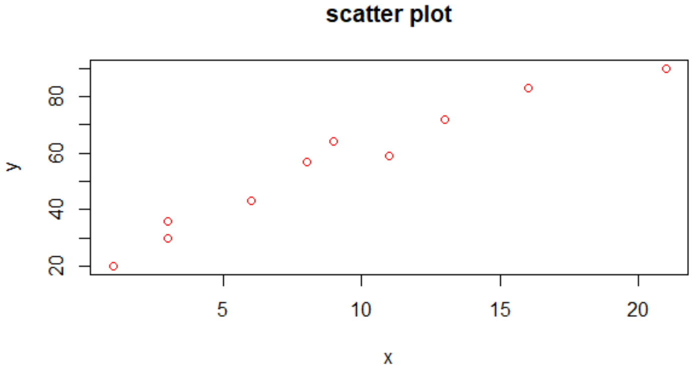
1) Linear Regression

Source Code :-

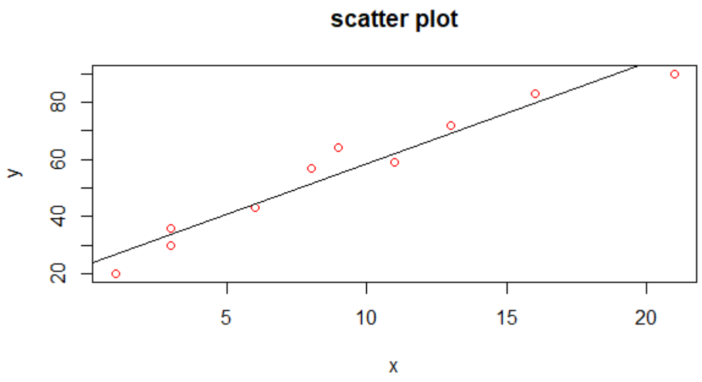
```
x<-c(3,8,9,13,3,6,11,21,1,16)
#response variable

y<-c(30,57,64,72,36,43,59,90,20,83)
plot(x,y)

plot (x,y,col='red',main="scatter plot")

model=lm(y~x)
model
attributes(model)

coef(model)
residuals(model)
summary(model)
abline(model)
#predicting values manually y=a+bx
x10<-model$coefficients[[1]]+model$coefficient[[2]]*10
x10
#using predict()
a<-data.frame(x=10)
a
pred<-predict(model,a)
pred

plot(model)
```

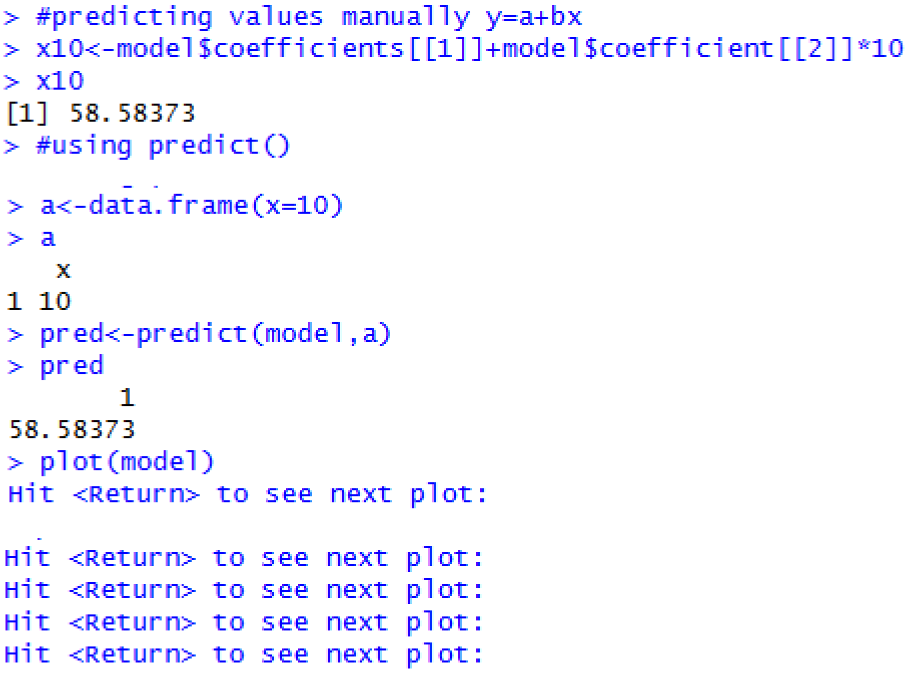
Output :-



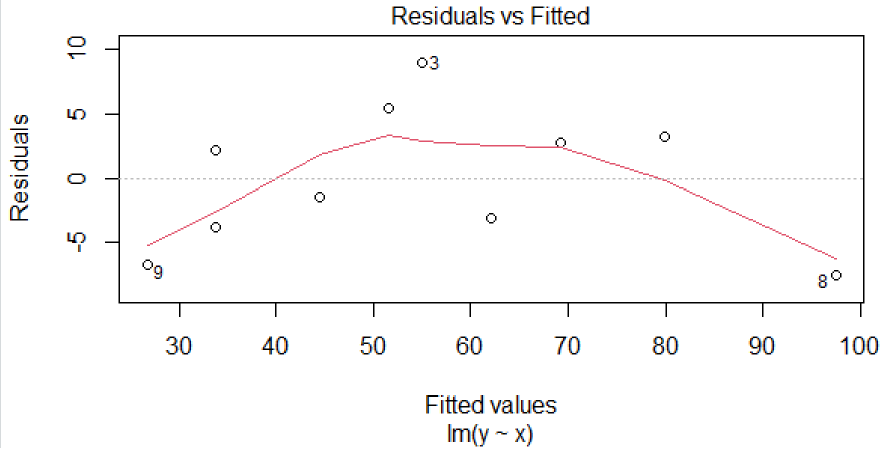


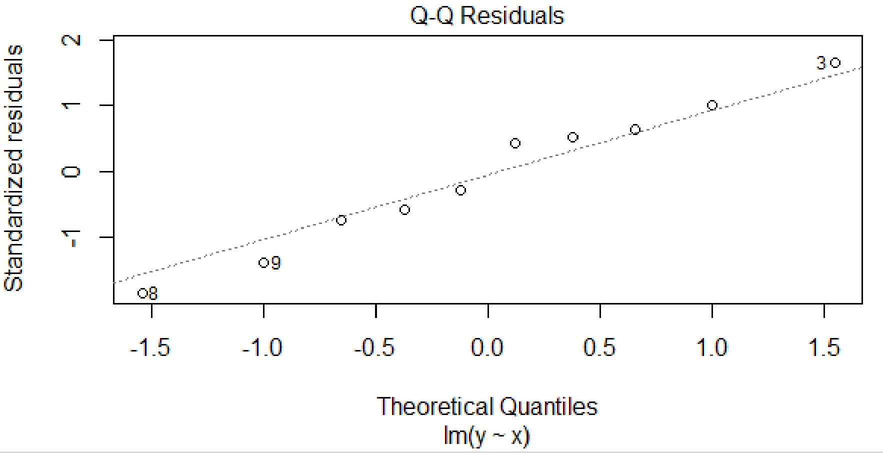
```
> model=lm(y~x)
> model

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x  
     23.209        3.537  

> attributes(model)
$names
 [1] "coefficients"  "residuals"     "effects"       "rank"          "fitted.values"
 [6] "assign"        "qr"            "df.residual"   "xlevels"       "call"         
[11] "terms"         "model"        

$class
[1] "lm"

> coef(model)
(Intercept)           x 
  23.208972    3.537476 
> residuals(model)
        1         2         3         4         5         6         7         8 
-3.821399  5.491223  8.953748  2.803845  2.178601 -1.433826 -3.121204 -7.495960 
        9        10 
-6.746447  3.191418 
> summary(model)

Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max 
-7.4960 -3.6463  0.3724  3.0945  8.9537 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  23.2090     3.2862   7.062 0.000106 ***
x             3.5375     0.3016  11.728 2.55e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.714 on 8 degrees of freedom

> abline(model)
```

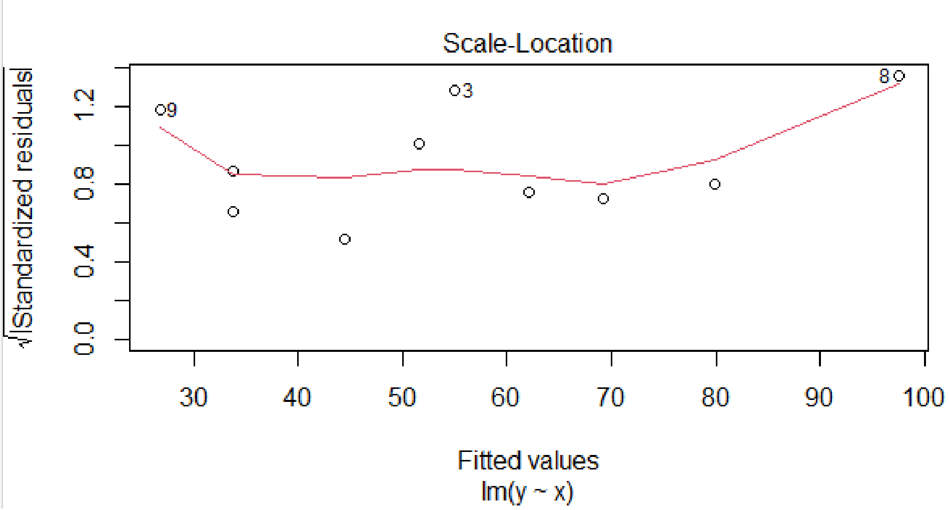


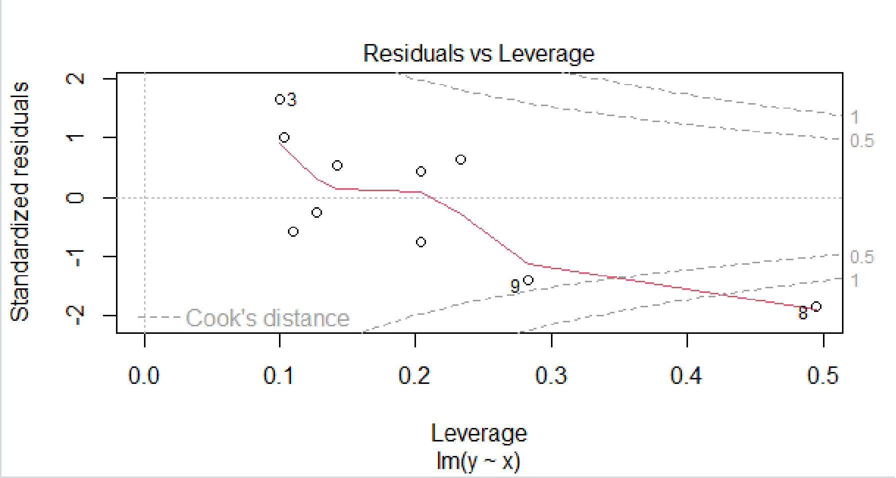


```
> #predicting values manually y=a+bx
> x10<-model$coefficients[[1]]+model$coefficient[[2]]*10
> x10
[1] 58.58373
> #using predict()

> a<-data.frame(x=10)
> a
   x
1 10
> pred<-predict(model,a)
> pred
       1
58.58373
> plot(model)
Hit <Return> to see next plot:

Hit <Return> to see next plot:
Hit <Return> to see next plot:
Hit <Return> to see next plot:
Hit <Return> to see next plot:
```






Scale-Location
√|Standardized residuals|
0.0
0.4
0.8
1.2
30
40
50
60
70
80
90
100
Fitted values
lm(y ~ x)
9
3
8


Residuals vs Leverage
Standardized residuals
-2
-1
0
1
2
0.0
0.1
0.2
0.3
0.4
0.5
Leverage
lm(y ~ x)
3
9
8
Cook's distance
1
0.5
0.5
1