R Codes of Regression analysis project:

```
d1=read.csv("C:/Users/Saikatk/Desktop/Statistics project/Regression/data.csv")
#the data file is read#
d1=na.omit(d1)
d1
#the data points which contain some missing observation are deleted#
attach(d1)
length(Y)
length(X[,2])
#length of Y is determined#
library(leaps)#calling the leaps library
X=matrix(0,143,6) #matrix of regressors is formed#
X[,1]=X1
X[,2]=X2
X[,3]=X3
X[,4]=X4
X[,5]=X5
X[,6]=X6
leaps(X,Y,int=TRUE,method=c("adjr2"))#Calculating the adjusted R^2
leaps(X,Y,int=TRUE,method=c("Cp"))#calculating the Cp
Y1=regsubsets(X,Y,int=TRUE,method=c("forward"),names=c(X[,1],X[,2],X[,3],X[,4],X[,5],X[,6]))
#forward elimination technique is carried out for model building#
par(mfrow=c(1,2))
plot(Y1,levels=Y1$Xname,scale=c("Cp"),main="Plot with respect to Cp")
plot(Y1,levels=Y1$Xname,scale=c("adjr2"),main="Plot with respect to adjusted R^2")
#model building procedure using adjR2 and Cp is shown in graph#
#From the graph it is evident that wrt Cp the est model is Y=a+b1X1+b2X2+b3X3+b4X4
and wrt R2adj is Y=a+b1X1+b2X2+b3x3+b4X4#
library(MPV)#calling the MPV library
#calculating the prediction criteria for each combination of regressors
PRESS(lm(Y~X[,1]))
PRESS(lm(Y~X[,2]))
PRESS(lm(Y~X[,3]))
PRESS(lm(Y~X[,4]))
PRESS(lm(Y~X[,5]))
PRESS(lm(Y~X[,6]))
#X1 gets selected#
PRESS(lm(Y~X[,1]+X[,2]))
PRESS(lm(Y~X[,1]+X[,3]))
PRESS(lm(Y~X[,1]+X[,4]))
PRESS(lm(Y~X[,1]+X[,5]))
PRESS(lm(Y~X[,1]+X[,6]))
#X1 and X2 gets selected#
PRESS(lm(Y~X[,1]+X[,2]+X[,3]))
PRESS(lm(Y~X[,1]+X[,2]+X[,4]))
PRESS(lm(Y~X[,1]+X[,2]+X[,5]))
PRESS(lm(Y~X[,1]+X[,2]+X[,6]))
#X1,X2 and X5 gets selected#
PRESS(lm(Y~X[,1]+X[,2]+X[,5]+X[,3]))
PRESS(lm(Y~X[,1]+X[,2]+X[,5]+X[,4]))
PRESS(lm(Y~X[,1]+X[,2]+X[,5]+X[,6]))
#The PRESS in all the last 3 cases increases.So the most parsimonious model is
Y=a+b1X1+b2X2+b3X5#
extractAIC(lm(Y~X[,1]))
extractAIC(lm(Y~X[,2]))
extractAIC(lm(Y~X[,3]))
extractAIC(lm(Y~X[,4]))
extractAIC(lm(Y~X[,5]))
extractAIC(lm(Y~X[,6]))
#X1 is selected#
extractAIC(lm(Y~X[,1]+X[,2]))
extractAIC(lm(Y~X[,1]+X[,3]))
extractAIC(lm(Y~X[,1]+X[,4]))
extractAIC(lm(Y~X[,1]+X[,5]))
extractAIC(lm(Y~X[,1]+X[,6]))
#X1 and X2 selected#
extractAIC(lm(Y~X[,1]+X[,2]+X[,3]))
extractAIC(lm(Y~X[,1]+X[,2]+X[,4]))
extractAIC(lm(Y~X[,1]+X[,2]+X[,5]))
extractAIC(lm(Y~X[,1]+X[,2]+X[,6]))
#X1,X2 and X3 selected#
extractAIC(lm(Y~X[,1]+X[,2]+X[,3]+X[,4]))
extractAIC(lm(Y~X[,1]+X[,2]+X[,3]+X[,5]))
extractAIC(lm(Y~X[,1]+X[,2]+X[,3]+X[,6]))
#X1,X2,X3 and X4 selected#
extractAIC(lm(Y~X[,1:4]+X[,6]))
extractAIC(lm(Y~X[,1:5]))
#The AIC increases,so we conclude on the basis of Cp,adjR2 and AIC that the best model is
Y=a+b1X1+b2X2+b3X3+b4X4#
```

```
l1=lm(Y~X1+X2+X3+X4)#Carrying out the normal linear regression#
summary(l1)
```

```
y1=fitted.values(l1)
y1
#the fitted values of y are determined#
r1=rstandard(lm(Y~X1+X2+X3+X4))
plot(Y,r1,ylab="Standardized Residuals")
#standardized residuals are plotted against Y#
abline(h=0)
abline(h=-3)
abline(h=3)
r2=rstudent(lm(Y~X1+X2+X3+X4))
plot(Y,r2,ylab="Studentized Residuals")
#studentized residuals are plotted against Y#
library(car)
# Influential Observations are known with added variable plots#
avPlots(l1)
# Cook's D plot identify D values > 1#
cutoff <- 1
plot(l1, which=4, cook.levels=cutoff)
# Influence Plot#
influencePlot(l1, id.method="identify", main="Influence Plot", sub="Circle size is proportial to
Cook's Distance" )
#we delete outliers and influential observation#
d=read.csv("C:/Users/RahulC/Desktop/Statistics project/Regression/data1.csv")
#edited dataset is read#
d
d=na.omit(d)
d
attach(d)
l=lm(Y1~X11+X12+X13+X14)#Carrying out the normal linear regression#
summary(l)
length(Y1)
y=fitted.values(l)
y
X1=matrix(0,139,6) #matrix of regressors is formed#
X1[,1]=X11
X1[,2]=X12
X1[,3]=X13
X1[,4]=X14
X1[,5]=X15
X1[,6]=X16
e1=summary(l)$residuals
e1
plot(fitted.values(l),e1,ylab="residuals",xlab="fitted values")
```

```
#plot of residuals against fitted values#
#plots of regressors against residuals#
par(mfrow=c(2,2))
plot(X11,e1,xlab="X1",ylab="residuals")
abline(h=0)
plot(X12,e1,xlab="X2",ylab="residuals")
abline(h=0)
plot(X13,e1,xlab="X3",ylab="residuals")
abline(h=0)
plot(X14,e1,xlab="X4",ylab="residuals")
abline(h=0)
#gqtest is performed#
library(lmtest)
gqtest(Y1~X11,order.by=X11)
library(lmtest)
gqtest(Y1~X12,order.by=X12)
library(lmtest)
gqtest(Y1~X13,order.by=X13)
library(lmtest)
gqtest(Y1~X14,order.by=X14)
#variables are transformed#
Y1=Y1/X12
X1=X1/X12
e2=e1/X12
X21=X2[,1]
X22=X2[,2]
X23=X2[,3]
X24=X2[,4]
X25=X2[,5]
X26=X2[,6]
#plots of residuals against regressors#
par(mfrow=c(2,2))
plot(e2,X21,ylab="X1",xlab="residuals")
plot(e2,X22,ylab="X2",xlab="residuals")
plot(e2,X23,ylab="X3",xlab="residuals")
plot(e2,X24,ylab="X4",xlab="residuals")
#gqtest is performed#
gqtest(Y2~X21,order.by=X21)
library(lmtest)
gqtest(Y2~X22,order.by=X22)
library(lmtest)
gqtest(Y2~X23,order.by=X23)
library(lmtest)
gqtest(Y2~X24,order.by=X24)
```

```
#linear model on transformed variables#
l2=lm(Y2~X21+X22+X23+X24)
y2=fitted.values(l2)
e2=summary(l2)$residuals
plot(y2,e2,xlab="fitted values",ylab="residuals")
#checking normality of residuals#
qqnorm(e1)
qqline(e1)
library(car)
r3<- studres(l)
r3
library(MASS)
#histogrsm of residuals along with normal plot#
hist(r3, freq=FALSE,xlab="studentized residuals",
main="Distribution of Studentized Residuals")
xfit<-seq(min(r3),max(r3),length=50)
yfit<-dnorm(xfit)
lines(xfit, yfit)
#subjective transformation abs(y)^.5#
e3=summary(lm((abs(Y1)^(1/2))~X11+X12+X13+X14))$residuals
e3
qqnorm(e3)
qqline(e3)
#Box-Cox transformation#
library(MASS)
boxcox(Y2+1~X21+X22+X23+X24,lambda = seq(-2, 2, 1/10), plotit = TRUE,xlab =
expression(lambda),ylab = "log-Likelihood")
# Evaluate Collinearity
vif(l) # variance inflation factors
#variables are standardized#
Y1=(Y1-mean(Y1))/sd(Y1)
X11=(X11-mean(X11))/sd(X11)
X12=(X12-mean(X12))/sd(X12)
X13=(X13-mean(X13))/sd(X13)
X14=(X14-mean(X14))/sd(X14)
#fitting the regression on standardized observations#
lm(Y1~X11+X12+X13+X14)
summary(lm(Y1~X11+X12+X13+X14))
```