########## Packages Required #########

library(plyr)

library(corpcor)

install.packages("mvinfluence")

library(mvinfluence)

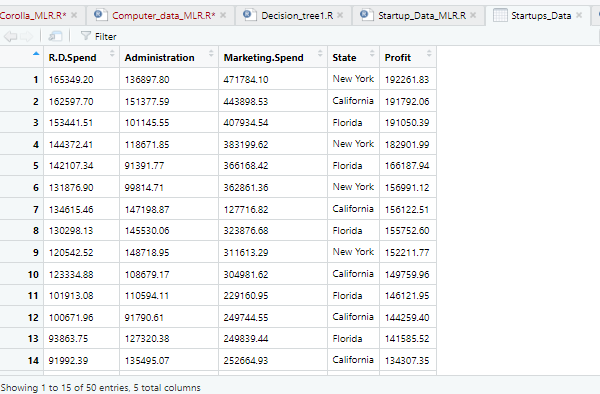
library(car)

library(corpcor)

#######reading and understanding the data #######

Startups\_Data<-read.csv(file.choose())

View(Startups\_Data)

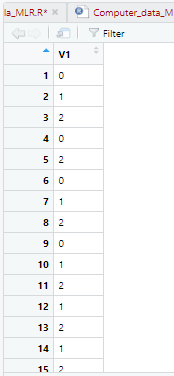


class(Startups\_Data)

Startups\_Data$State <- revalue(Startups\_Data$State,

c("New York"="0", "California"="1", "Florida"="2"))

View(Startups\_Data$State)



names(Startups\_Data)

C:\Users\home\Pictures\Screenshots\Screenshot (486).png

attach(Startups\_Data)

Startups\_Data <- cbind(RD\_Spend=R.D.Spend,Administration,Marketing\_Spend=Marketing.Spend,State,Profit)

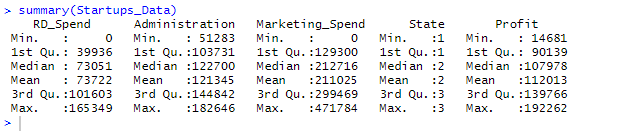
?cbind

names(Startups\_Data)

Startups\_Data <- as.data.frame(Startups\_Data)

attach(Startups\_Data) # Basically to avoid reference of Data Set name(Startups) in this report.

summary(Startups\_Data)



############ Steps of analysis for MLR #####

# Exploratory data analysis:

# 1. Measures of central tendency

# 2. Measures of dispersion

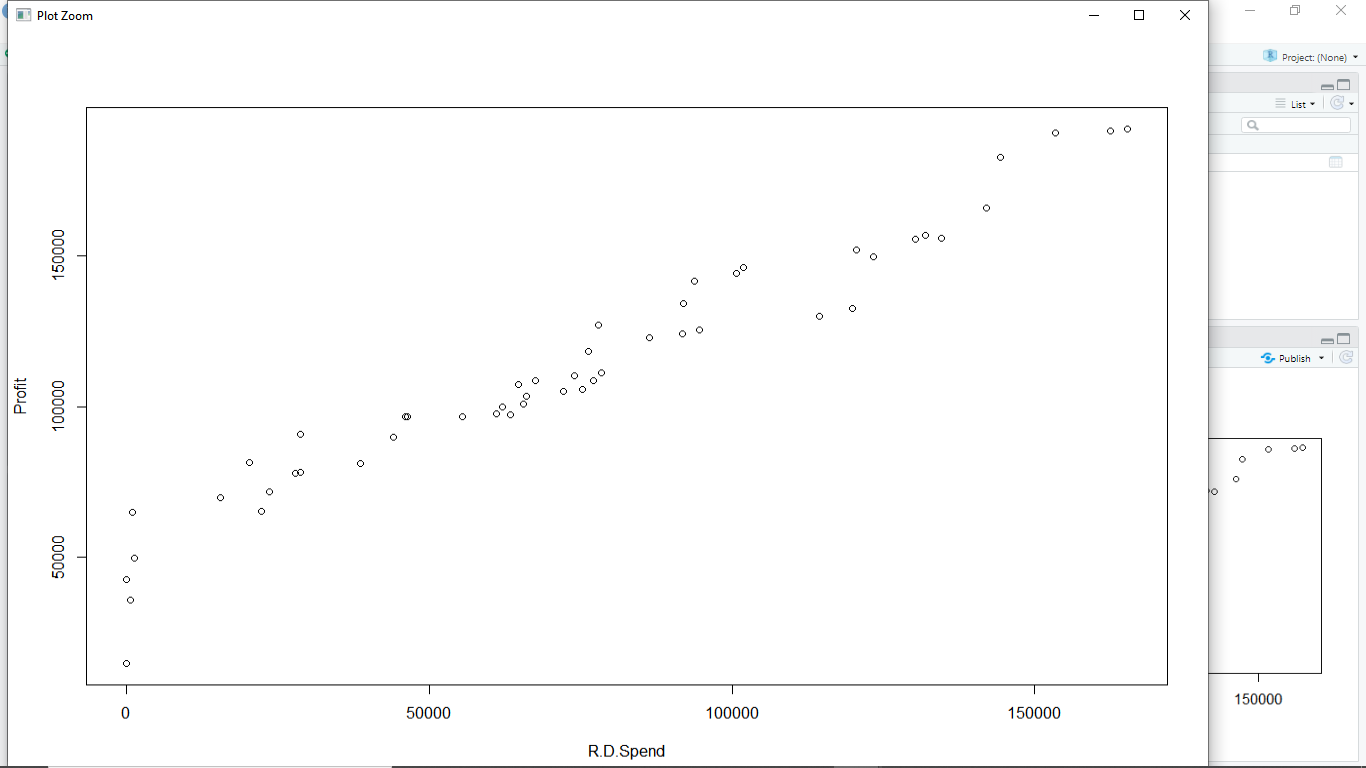
# 3. Third moment business decision

# 4. Fourth moment business decision

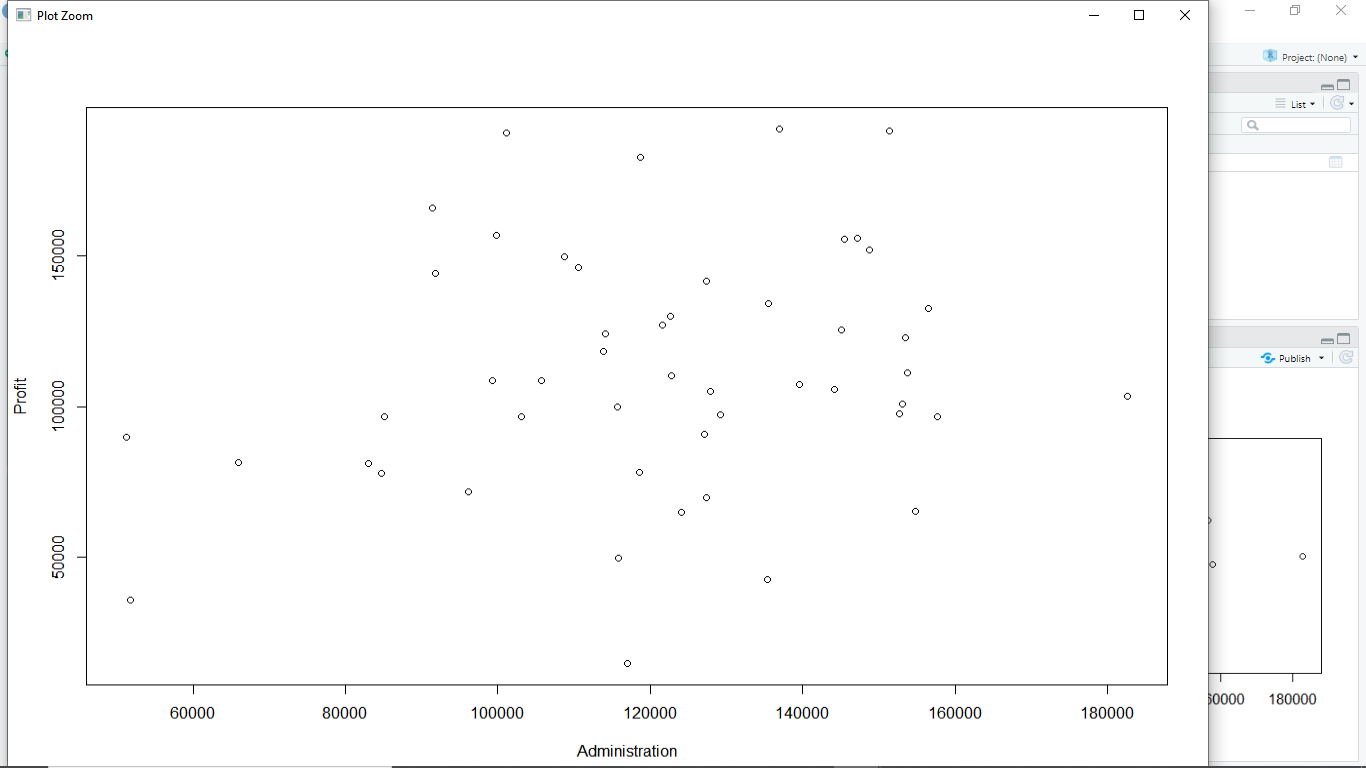
# 5. Probability distributions of variables

# 6. Graphical representations (Histogram, Box plot, Dot plot, Stem & Leaf plot, Bar plot, etc.)

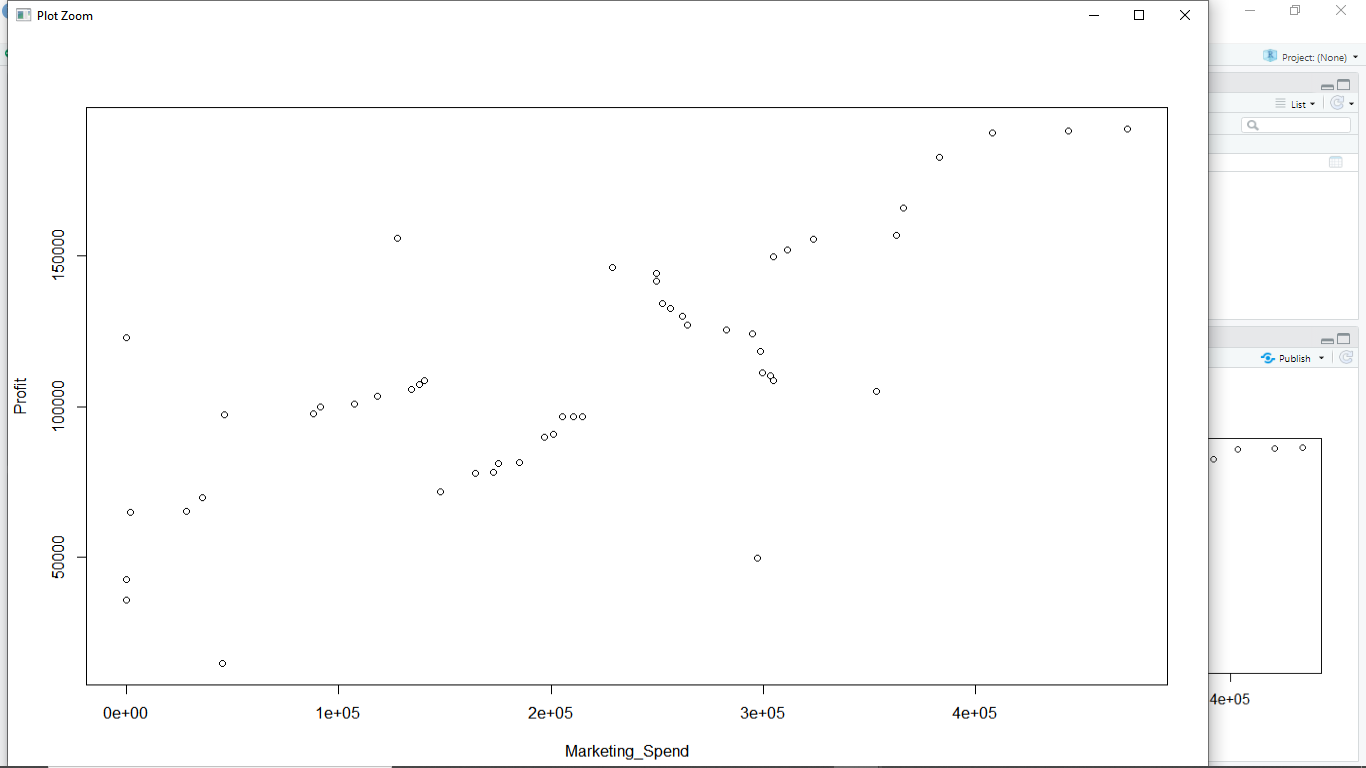
plot(R.D.Spend, Profit)



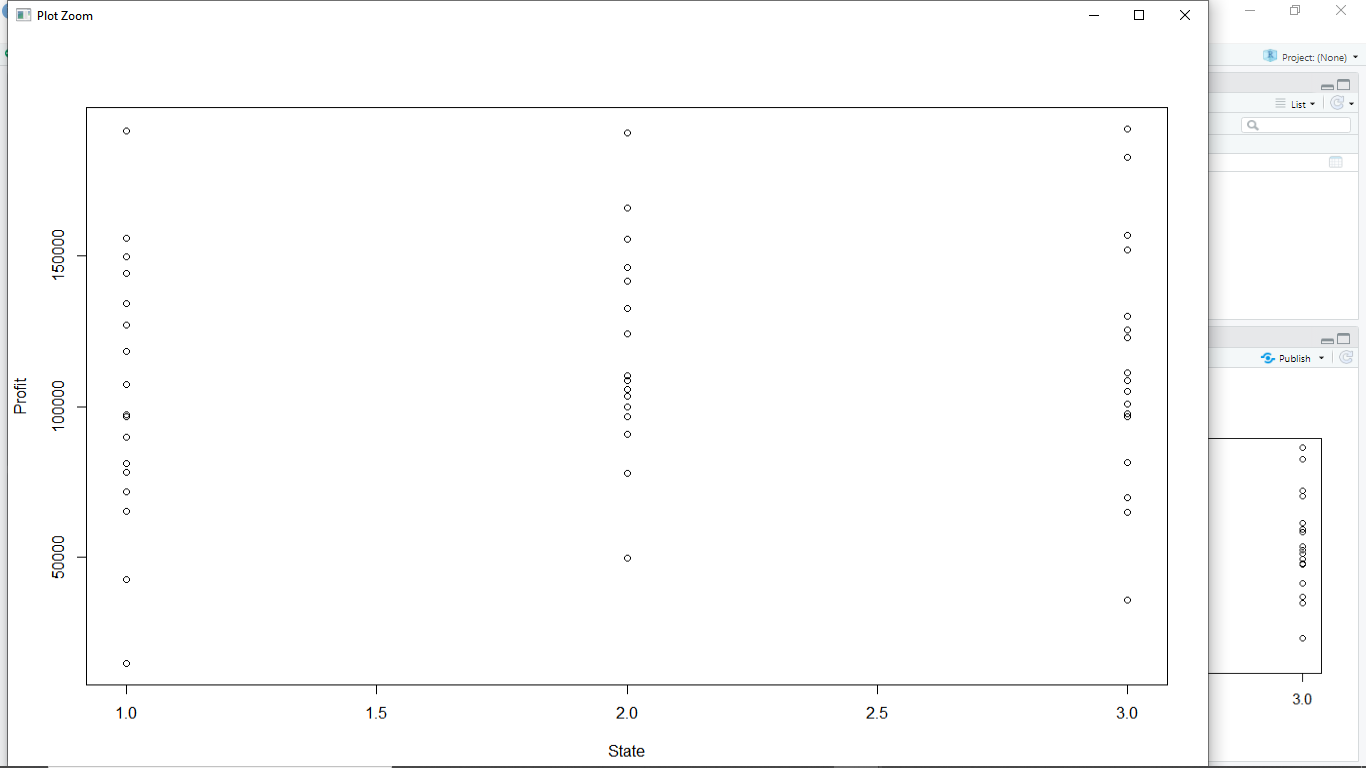
plot(Administration, Profit)



plot(Marketing\_Spend, Profit)



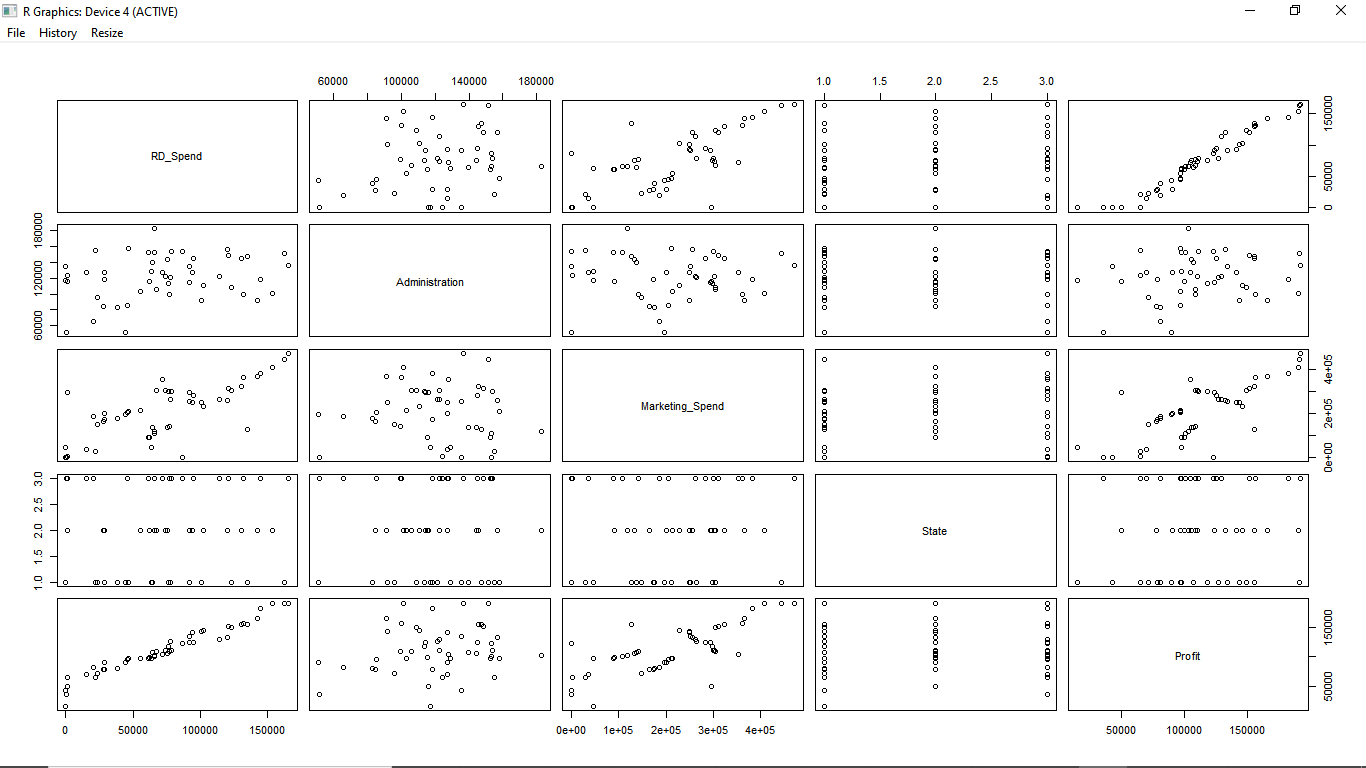
plot(State, Profit)



windows()

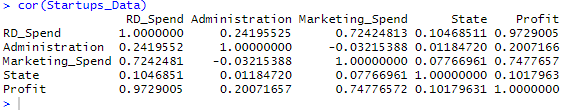
# 7. Find the correlation between Output (Profit) & inputs (R.D\_Spend, Administration, Marketing\_Spend, State) - SCATTER DIAGRAM

pairs(Startups\_Data)



# 8. Correlation coefficient - Strength & Direction of correlation

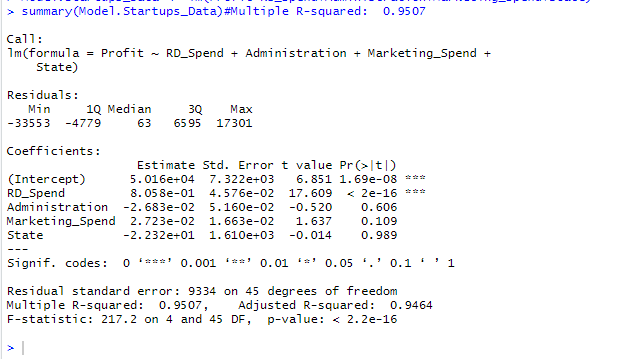
cor(Startups\_Data)



# The Linear Model of interest

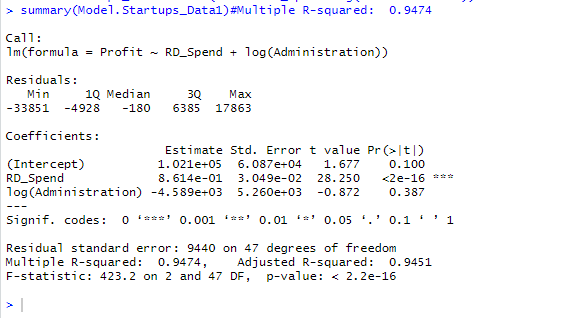
Model.Startups\_Data <- lm(Profit~RD\_Spend+Administration+Marketing\_Spend+State)

summary(Model.Startups\_Data)#Multiple R-squared: 0.9507



Model.Startups\_Data1 <- lm(Profit~RD\_Spend+log(Administration))

summary(Model.Startups\_Data1)#Multiple R-squared: 0.9474



###Model.Startups\_Data2 <- lm(Profit~RD\_Spend+Administration+Marketing\_Spend+log(State))

##summary(Model.Startups\_Data2)

### Scatter plot matrix with Correlations inserted in graph

panel.cor <- function(x, y, digits=2, prefix="", cex.cor)

{

usr <- par("usr"); on.exit(par(usr))

par(usr = c(0, 1, 0, 1))

r = (cor(x, y))

txt <- format(c(r, 0.123456789), digits=digits)[1]

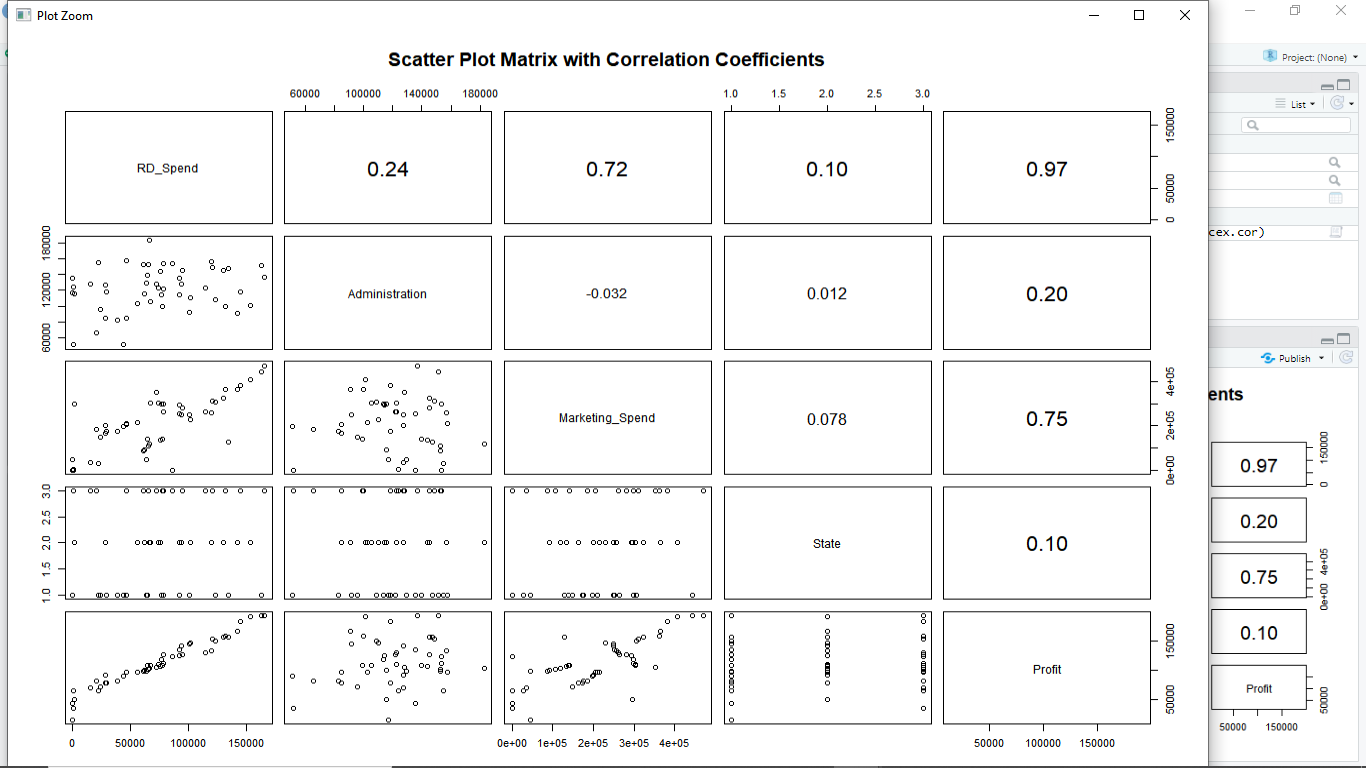
txt <- paste(prefix, txt, sep="")

if(missing(cex.cor)) cex <- 0.4/strwidth(txt)

text(0.5, 0.5, txt, cex = cex)

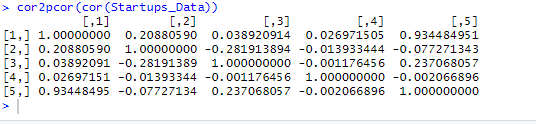
}

pairs(Startups\_Data, upper.panel=panel.cor,main="Scatter Plot Matrix with Correlation Coefficients")



### Partial Correlation matrix - Pure correlation between the variables

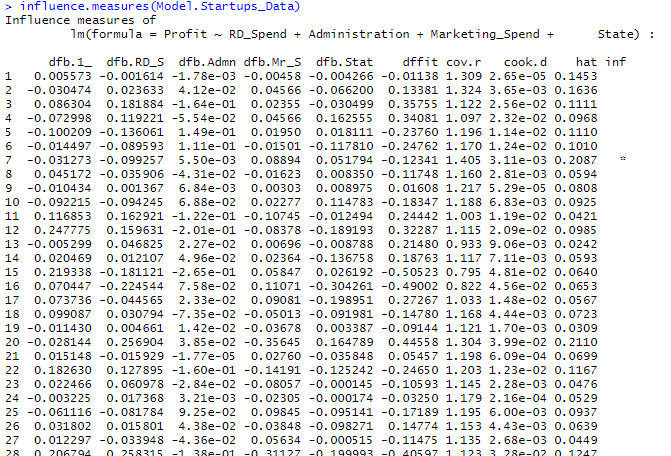
cor2pcor(cor(Startups\_Data))



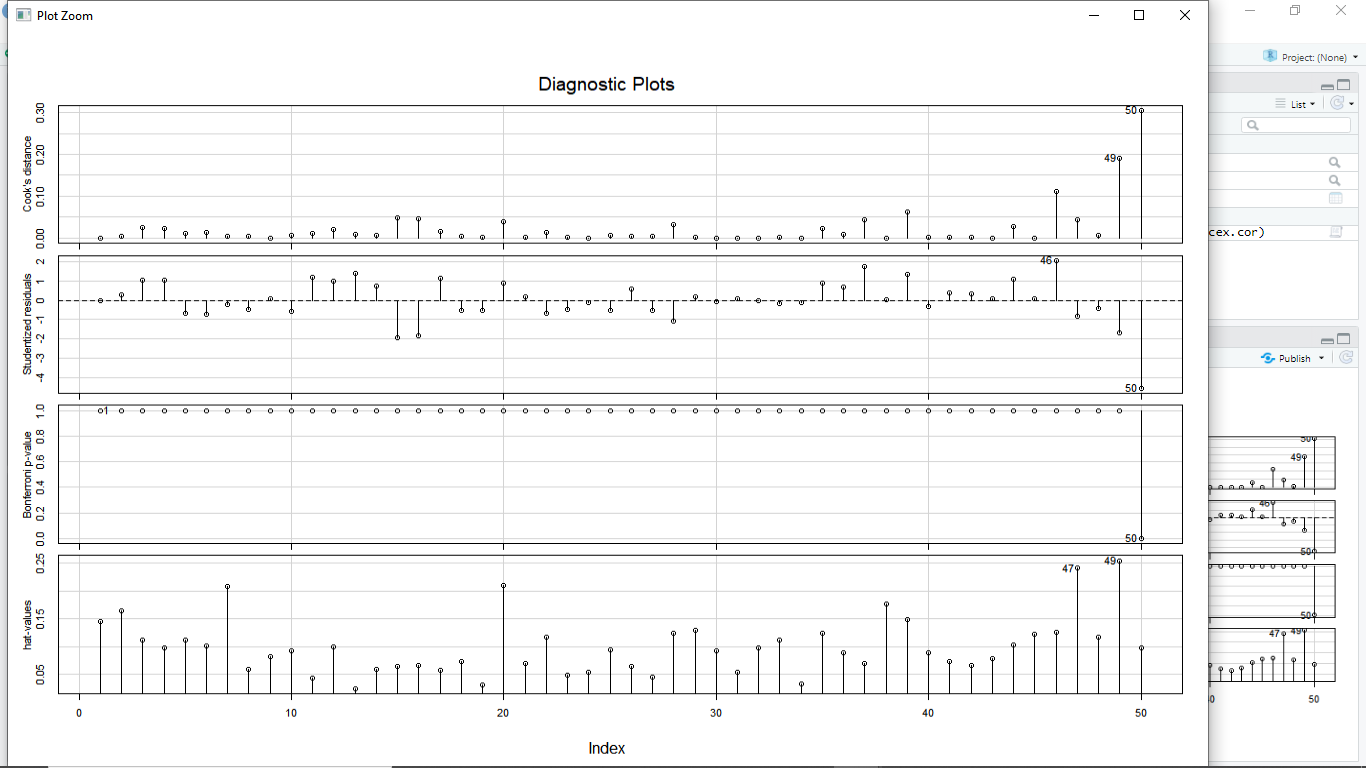
# It is better to delete a single observation rather than entire variable to get rid of collinearity problem

# Deletion Diagnostics for identifying influential variable

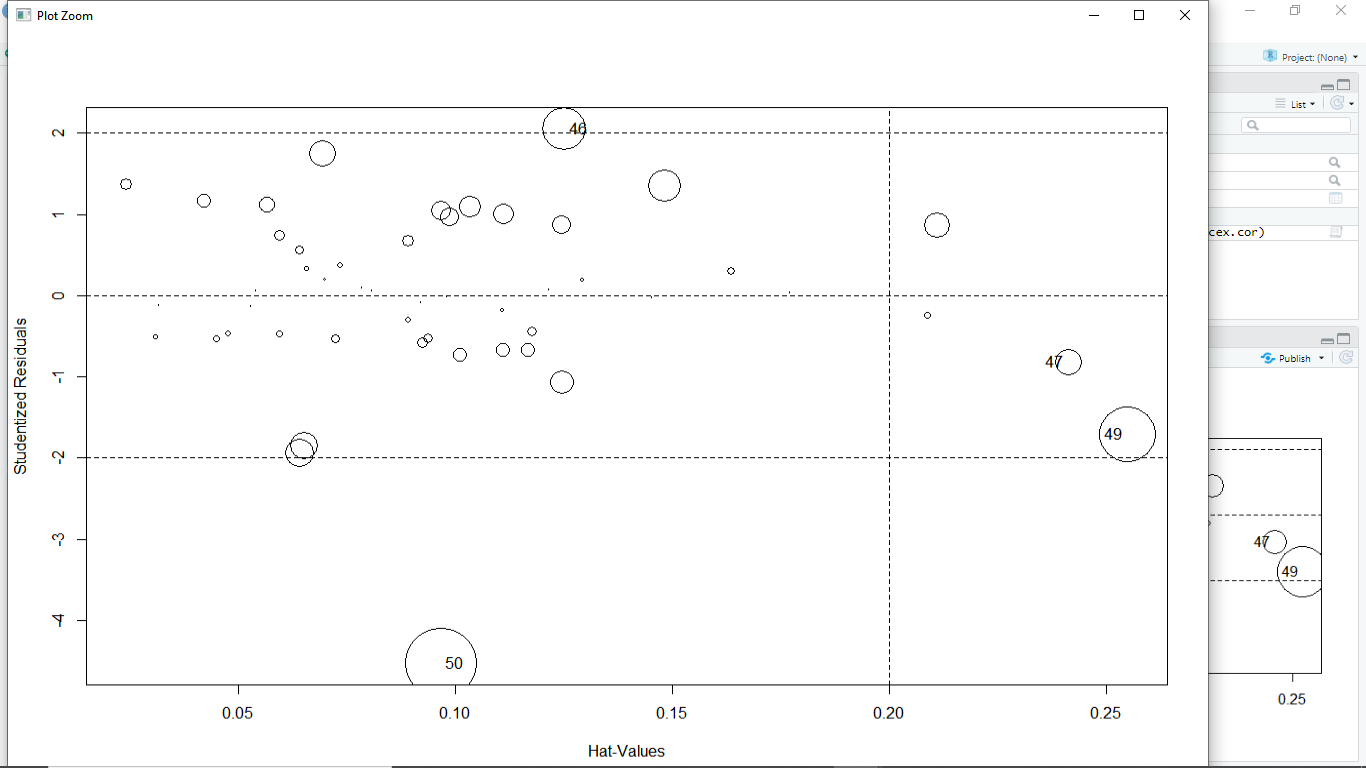
influence.measures(Model.Startups\_Data)



influenceIndexPlot(Model.Startups\_Data, id.n=3) # Index Plots of the influence measures



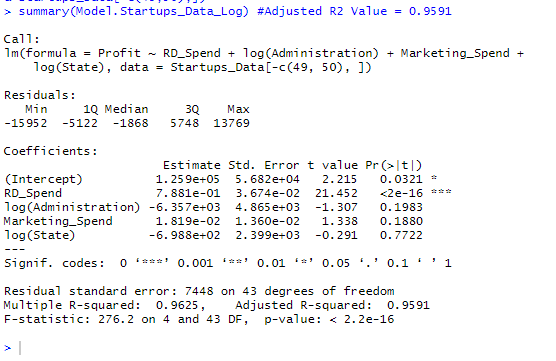
influencePlot(Model.Startups\_Data, id.n=3) # A user friendly representation of the above



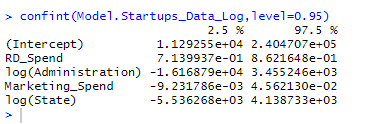
#############\*\*\*\*\*\* Logarthimic Transformation \*\*\*\*\*\*\*##############

Model.Startups\_Data\_Log<-lm(Profit~RD\_Spend+log(Administration)+Marketing\_Spend+log(State),data=Startups\_Data[-c(49,50),])

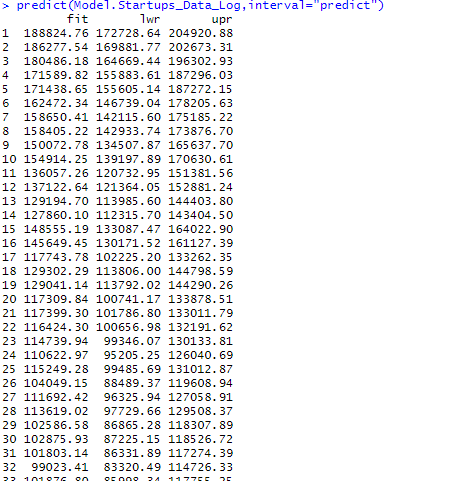
summary(Model.Startups\_Data\_Log) #Adjusted R2 Value = 0.9591 ,Multiple R-squared: 0.9625



confint(Model.Startups\_Data\_Log,level=0.95)

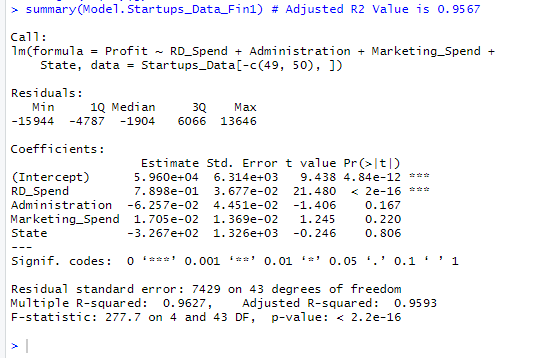


predict(Model.Startups\_Data\_Log,interval="predict")



Model.Startups\_Data\_Fin1<-lm(Profit~RD\_Spend+Administration+Marketing\_Spend+State,data=Startups\_Data[-c(49,50),])

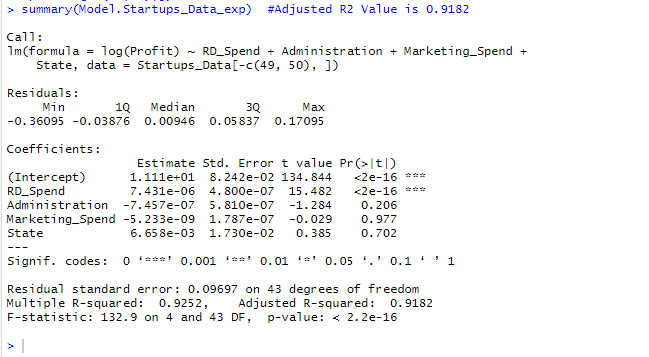
summary(Model.Startups\_Data\_Fin1) # Adjusted R2 Value is0.9593 ,Multiple R-squared: 0.9627



##########\*\*\*\*\*\*\*\*\*\*\* Exponential Transformation \*\*\*\*\*\*\*\*\*\*\*###########

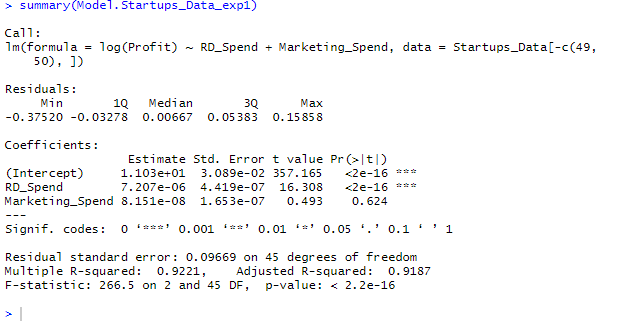
Model.Startups\_Data\_exp<-lm(log(Profit)~RD\_Spend+Administration+Marketing\_Spend+State,data=Startups\_Data[-c(49,50),])

summary(Model.Startups\_Data\_exp) #Adjusted R2 Value is 0.9182,Multiple R-squared: 0.9252,



Model.Startups\_Data\_exp1<-lm(log(Profit)~RD\_Spend+Marketing\_Spend,data=Startups\_Data[-c(49,50),])

summary(Model.Startups\_Data\_exp1) #Multiple R-squared: 0.9221, Adjusted R-squared: 0.9187

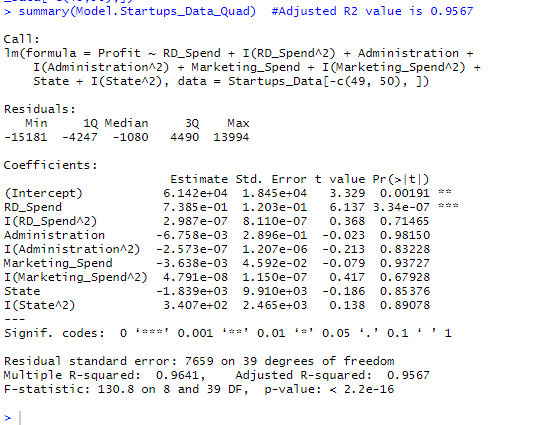


##################\*\*\*\*\*\*\*\*\*\* Quadratic Model \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*##############

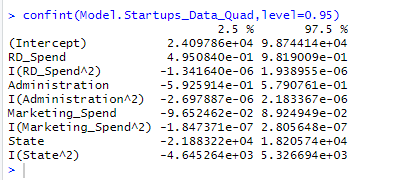
Model.Startups\_Data\_Quad <- lm(Profit~RD\_Spend+I(RD\_Spend^2)+Administration+I(Administration^2)

+Marketing\_Spend+I(Marketing\_Spend^2)+State+I(State^2),data=Startups\_Data[-c(49,50),])

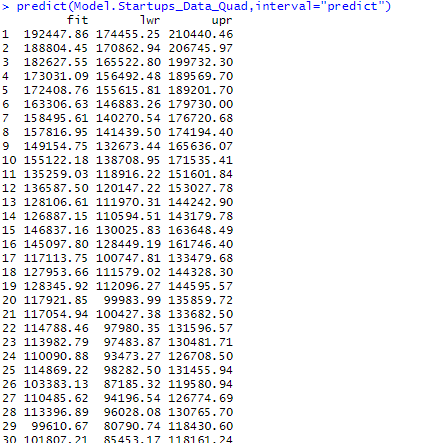
summary(Model.Startups\_Data\_Quad) #Multiple R-squared: 0.9641, Adjusted R-squared: 0.9567



confint(Model.Startups\_Data\_Quad,level=0.95)



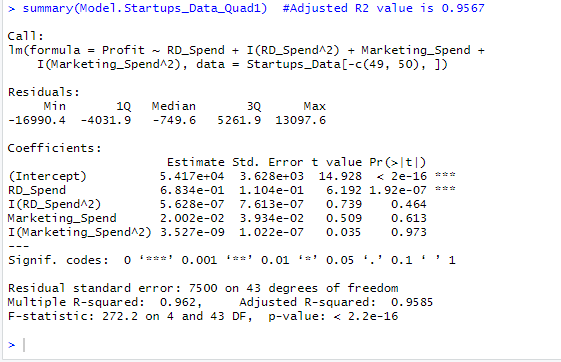
predict(Model.Startups\_Data\_Quad,interval="predict")



Model.Startups\_Data\_Quad1 <- lm(Profit~RD\_Spend+I(RD\_Spend^2)+Marketing\_Spend+I(Marketing\_Spend^2)

,data=Startups\_Data[-c(49,50),])

summary(Model.Startups\_Data\_Quad1) #Multiple R-squared: 0.962, Adjusted R-squared: 0.9585



#################\*\*\*\*\* Poly Modal \*\*\*\*\*\*\*\*\*\*################

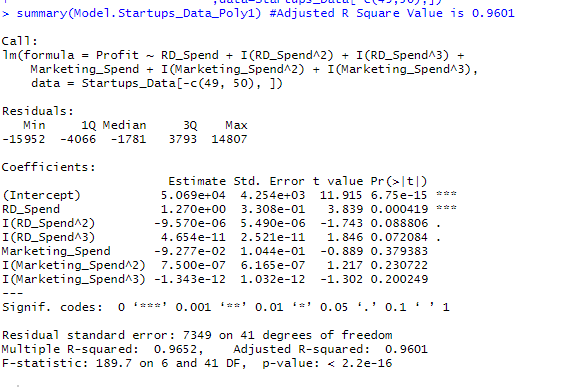
Model.Startups\_Data\_Poly <- lm(Profit~RD\_Spend+I(RD\_Spend^2)+I(RD\_Spend^3)+

Administration+I(Administration^2)+I(Administration^3)+

Marketing\_Spend+I(Marketing\_Spend^2)+I(Marketing\_Spend^3)+

State+I(State^2)+I(State^3),data=Startups\_Data[-c(49,50),])

summary(Model.Startups\_Data\_Poly) #Multiple R-squared: 0.967, Adjusted R-squared: 0.9569

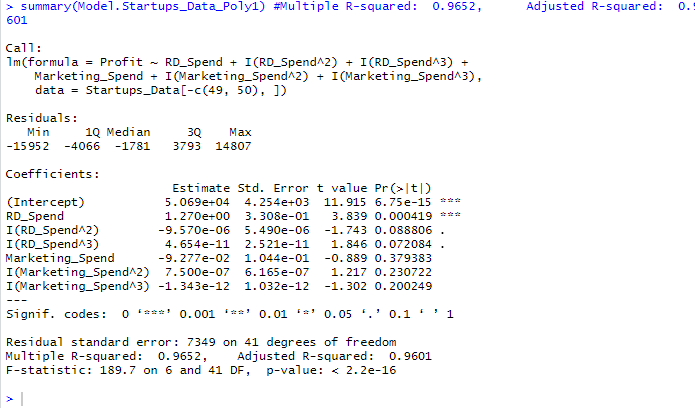


Model.Startups\_Data\_Poly1 <- lm(Profit~RD\_Spend+I(RD\_Spend^2)+I(RD\_Spend^3)+

Marketing\_Spend+I(Marketing\_Spend^2)+I(Marketing\_Spend^3)

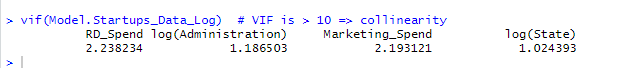
,data=Startups\_Data[-c(49,50),])

summary(Model.Startups\_Data\_Poly1) #Multiple R-squared: 0.9652, Adjusted R-squared: 0.9601

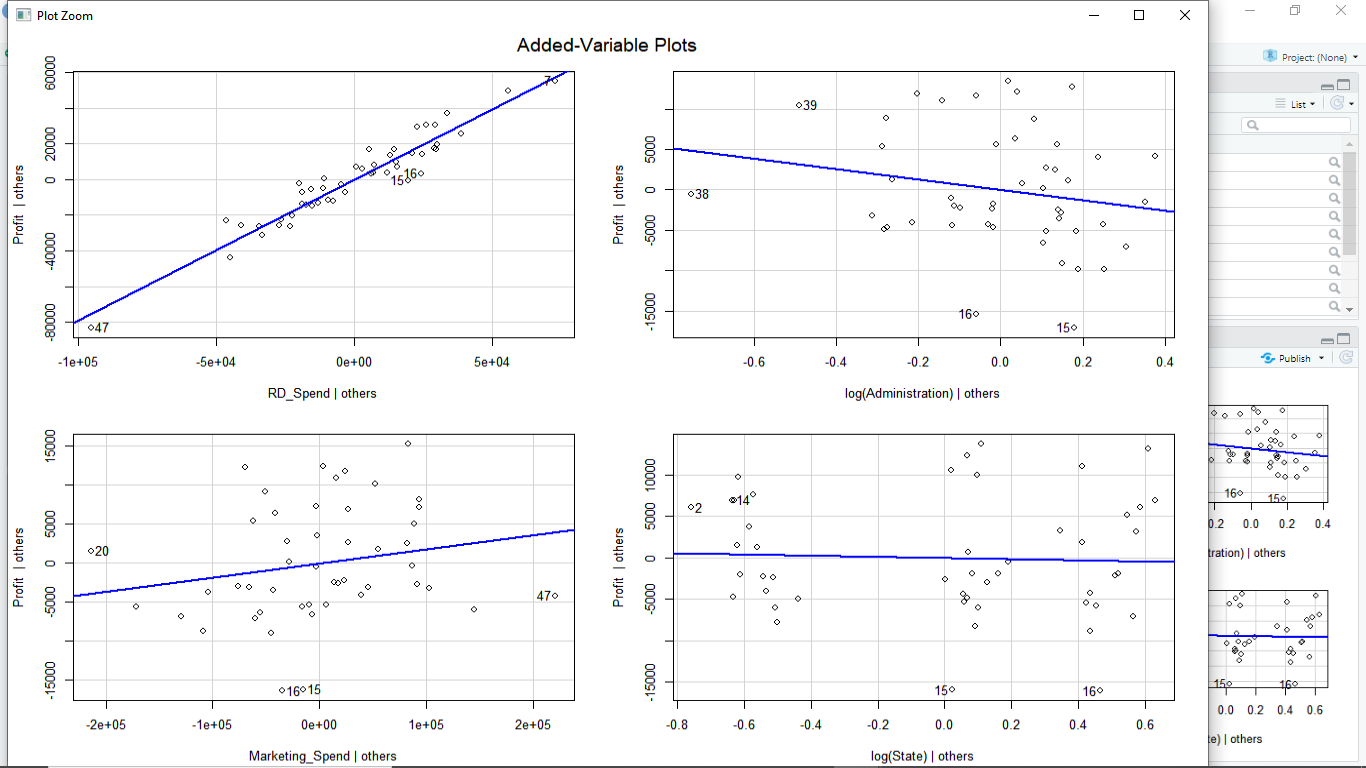


### Variance Inflation Factors is a formal way to check for collinearity

vif(Model.Startups\_Data\_Log) # VIF is > 10 => collinearity



avPlots(Model.Startups\_Data\_Log, id.n=2, id.cex=0.7) # Added Variable Plots

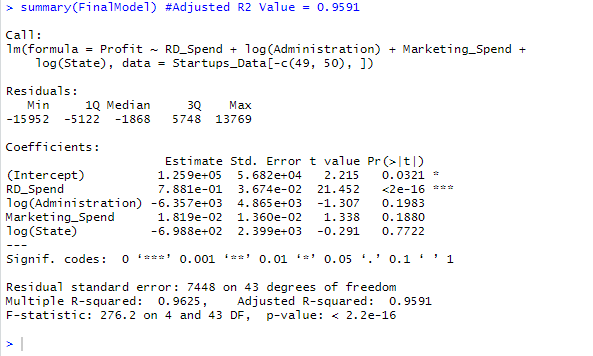


################### Final Model ####################

FinalModel<-lm(Profit~RD\_Spend+log(Administration)+Marketing\_Spend+

log(State),data=Startups\_Data[-c(49,50),])

summary(FinalModel) #Multiple R-squared: 0.9625, Adjusted R-squared: 0.9591

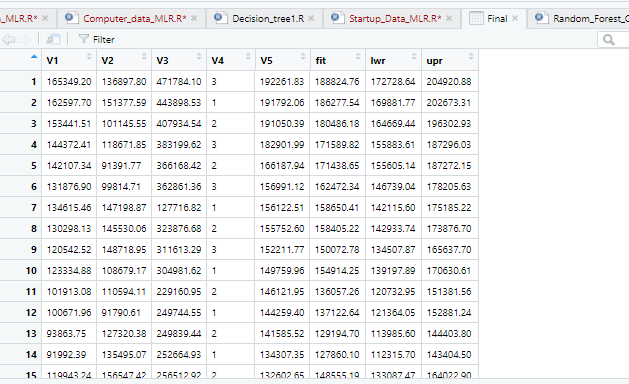


Profit\_Predict <- predict(FinalModel,interval="predict")

Final <- cbind(Startups\_Data$RD\_Spend,Startups\_Data$Administration,Startups\_Data$Marketing\_Spend,

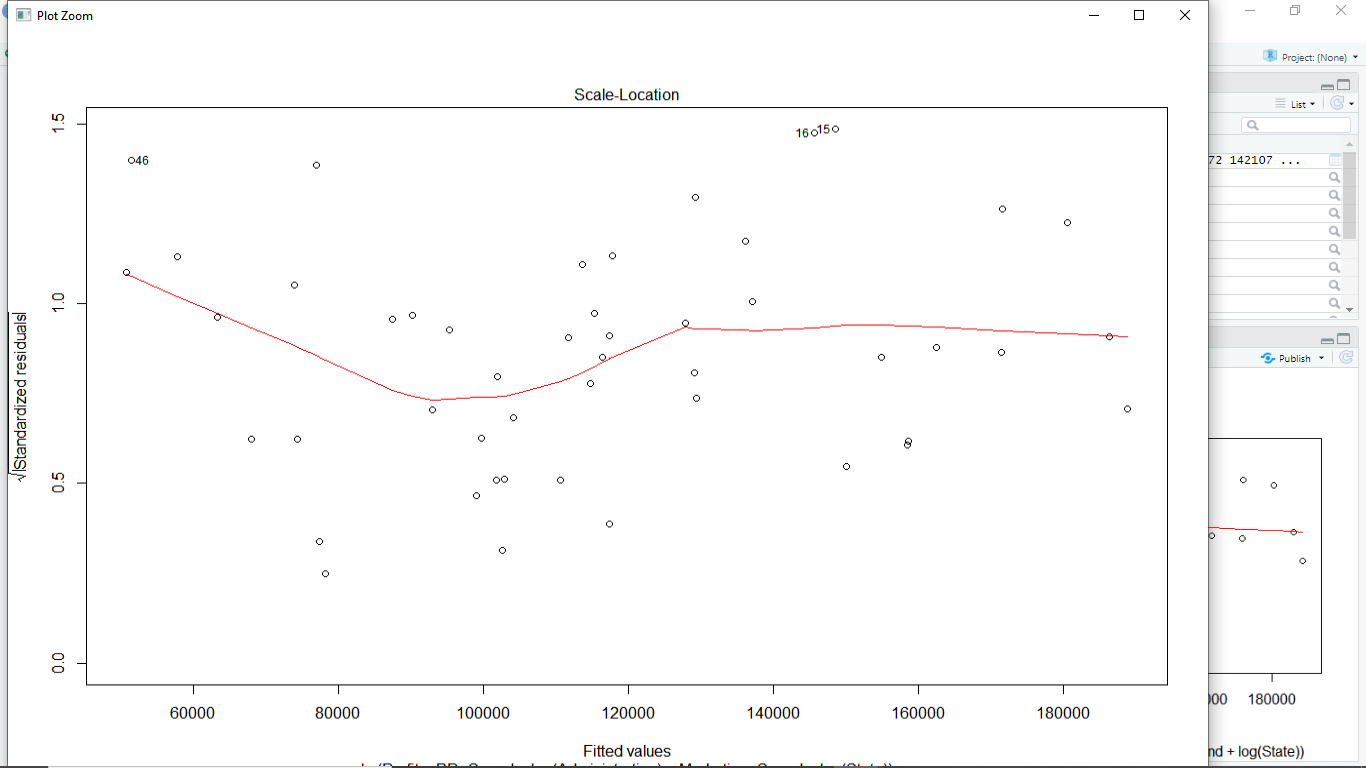
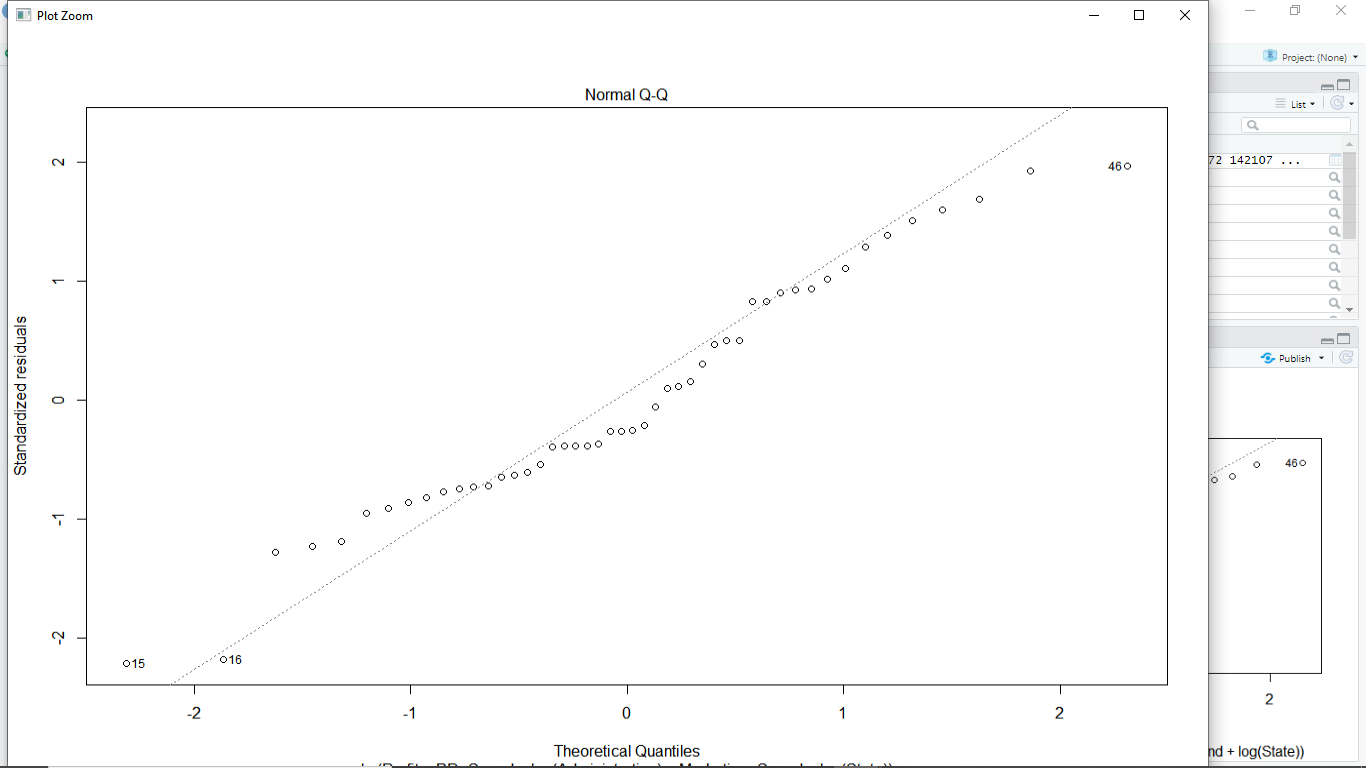
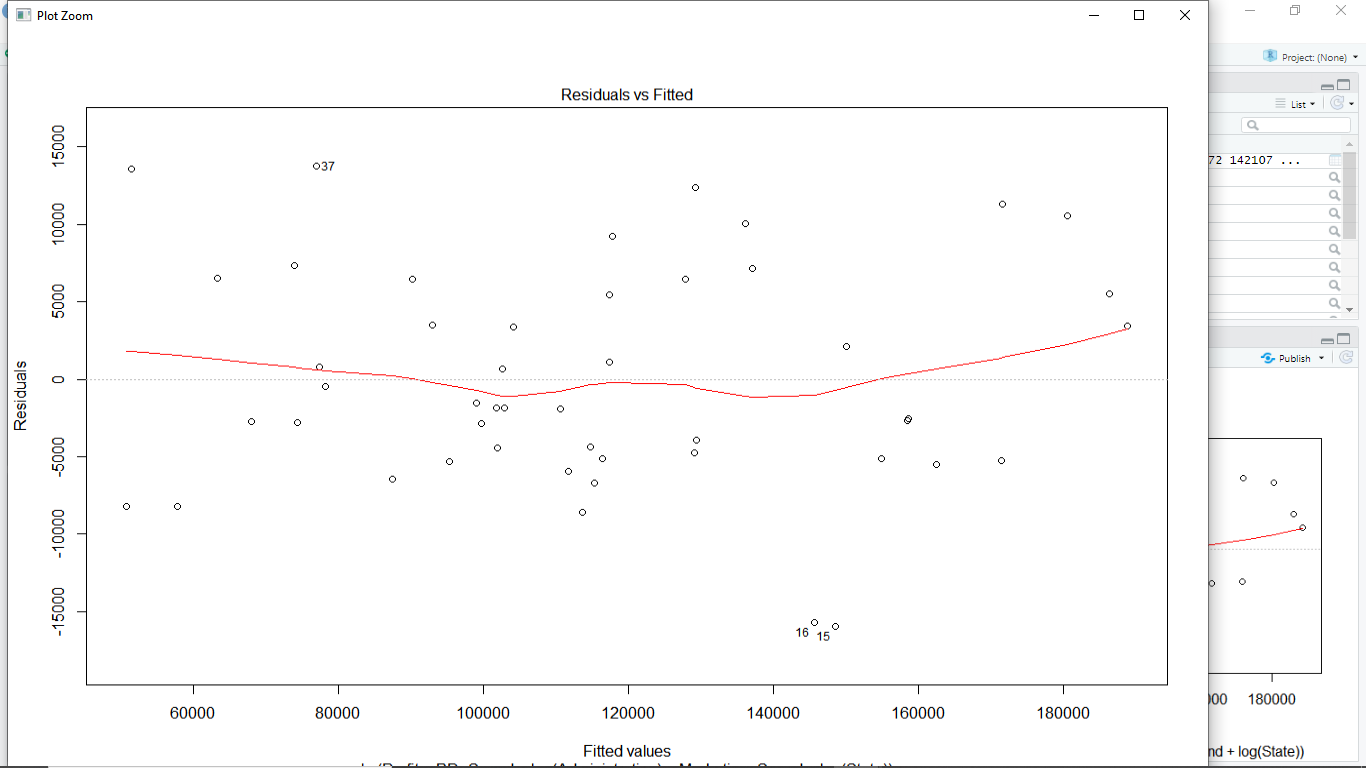
Startups\_Data$State,Startups\_Data$Profit,Profit\_Predict)

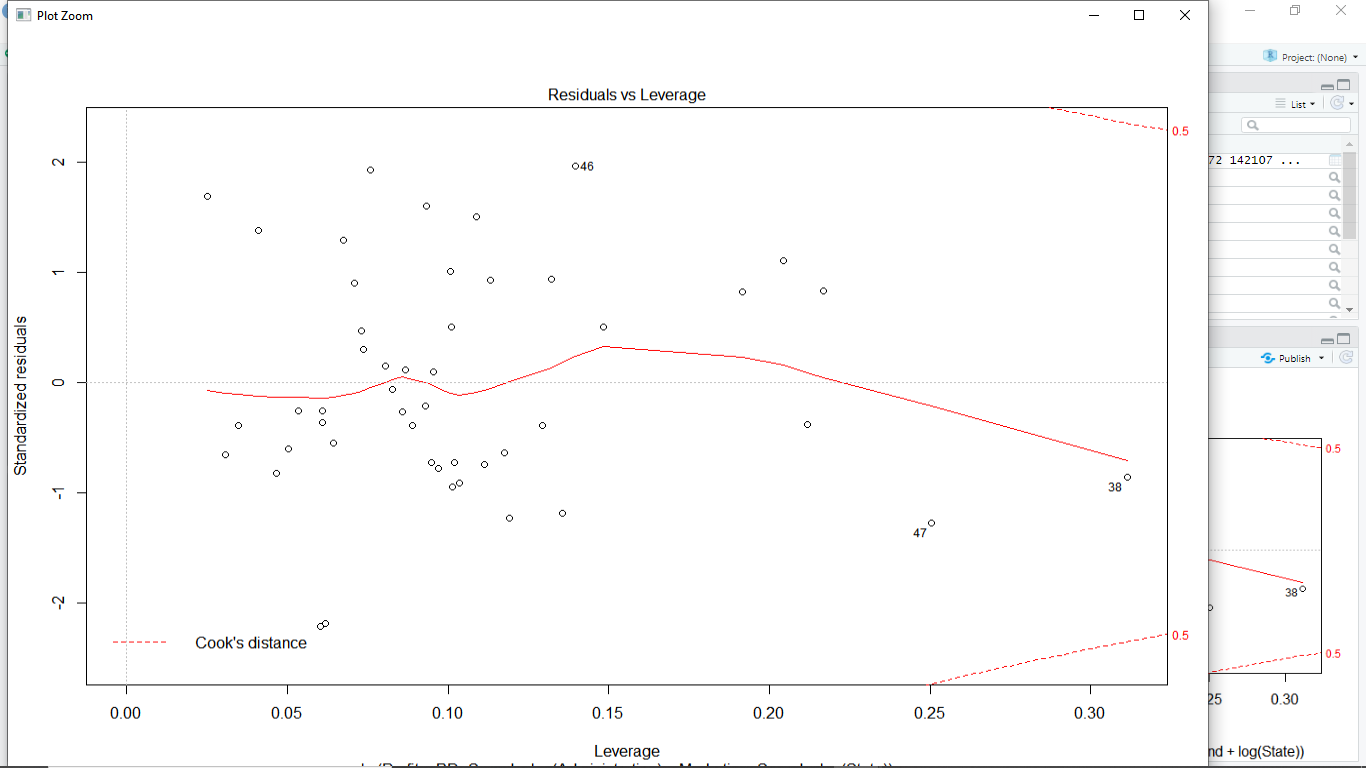
View(Final)



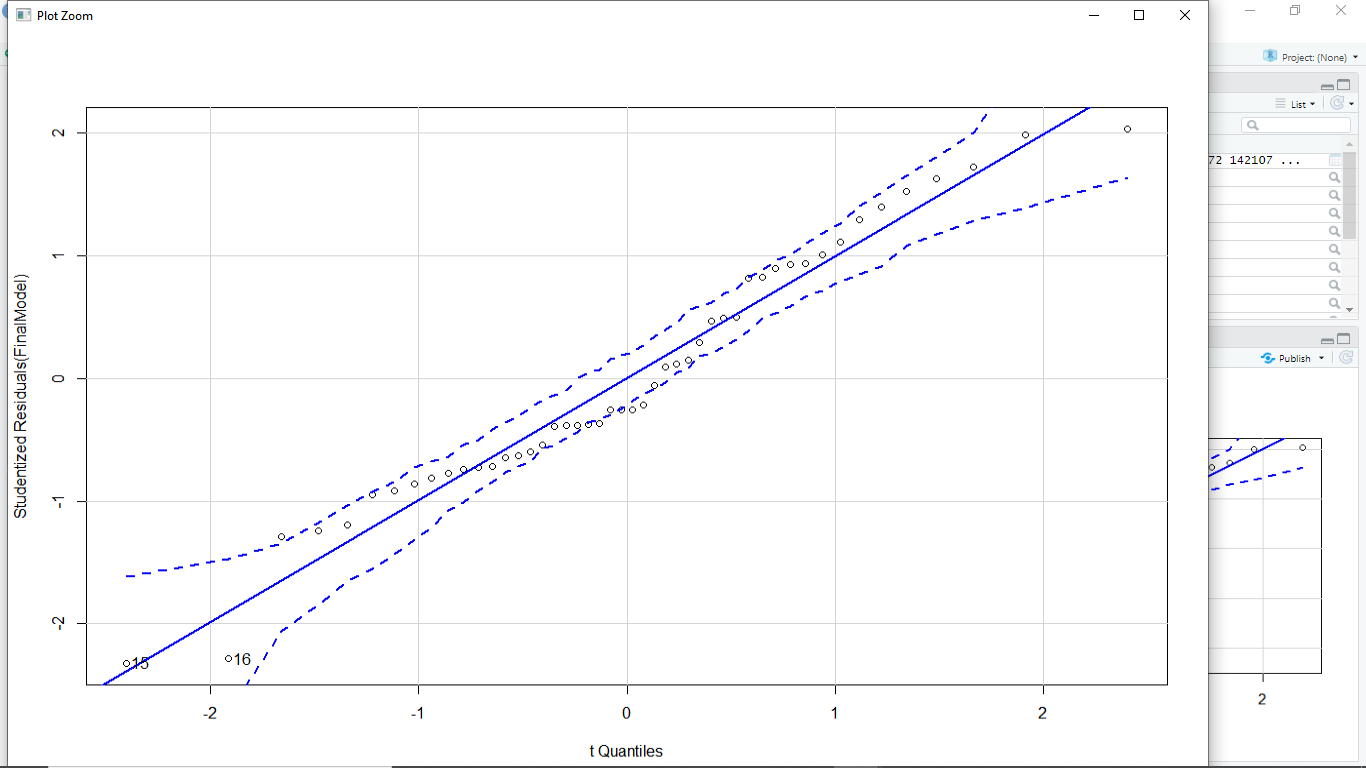
################# Evaluate model LINE assumptions ###################

plot(FinalModel)# Residual Plots, QQ-Plos, Std. Residuals vs Fitted, Cook's distance



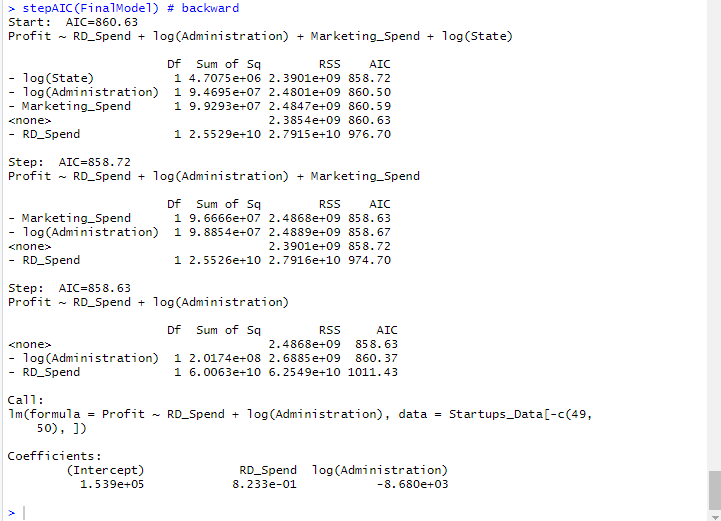


qqPlot(FinalModel, id.n=5) # QQ plots of studentized residuals, helps identify outliers



stepAIC(FinalModel) # backward

?stepAIC



# Lower the AIC value better is the model. AIC is used only if you build multiple models.