R Notebook

#load necessary libraries  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(readxl)  
library(ggplot2)  
library(ggpubr)  
library(qqplotr)

##   
## Attaching package: 'qqplotr'

## The following objects are masked from 'package:ggplot2':  
##   
## stat\_qq\_line, StatQqLine

#library(car)  
library(e1071)  
library(nortest)  
library(BSDA)

## Loading required package: lattice

##   
## Attaching package: 'BSDA'

## The following object is masked from 'package:datasets':  
##   
## Orange

library(psych)

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

library(caret)  
library(leaps)  
library(gvlma)  
# options("scipen"=100, "digits"=6)

# Import the csv into a dataframe  
file\_name = "auto-mpg.csv"  
df = read.csv(file\_name)  
head(df)

## mpg cylinder displacement horsepower weight acceleration model.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  
## car.name  
## 1 chevrolet chevelle malibu  
## 2 buick skylark 320  
## 3 plymouth satellite  
## 4 amc rebel sst  
## 5 ford torino  
## 6 ford galaxie 500

# Preliminary analysis

# structure of the dataframe  
str(df)

## 'data.frame': 398 obs. of 9 variables:  
## $ mpg : num 18 15 18 16 17 15 14 14 14 15 ...  
## $ cylinder : int 8 8 8 8 8 8 8 8 8 8 ...  
## $ displacement: num 307 350 318 304 302 429 454 440 455 390 ...  
## $ horsepower : chr "130" "165" "150" "150" ...  
## $ weight : int 3504 3693 3436 3433 3449 4341 4354 4312 4425 3850 ...  
## $ acceleration: num 12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...  
## $ model.year : int 70 70 70 70 70 70 70 70 70 70 ...  
## $ origin : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ car.name : chr "chevrolet chevelle malibu" "buick skylark 320" "plymouth satellite" "amc rebel sst" ...

# Converting the horsepower column to a numeric column  
df$horsepower = as.numeric(df$horsepower)

## Warning: NAs introduced by coercion

head(df)

## mpg cylinder displacement horsepower weight acceleration model.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  
## car.name  
## 1 chevrolet chevelle malibu  
## 2 buick skylark 320  
## 3 plymouth satellite  
## 4 amc rebel sst  
## 5 ford torino  
## 6 ford galaxie 500

# Summary of the dataframe  
summary(df)

## mpg cylinder displacement horsepower weight   
## Min. : 9.00 Min. :3.000 Min. : 68.0 Min. : 46.0 Min. :1613   
## 1st Qu.:17.50 1st Qu.:4.000 1st Qu.:104.2 1st Qu.: 75.0 1st Qu.:2224   
## Median :23.00 Median :4.000 Median :148.5 Median : 93.5 Median :2804   
## Mean :23.51 Mean :5.455 Mean :193.4 Mean :104.5 Mean :2970   
## 3rd Qu.:29.00 3rd Qu.:8.000 3rd Qu.:262.0 3rd Qu.:126.0 3rd Qu.:3608   
## Max. :46.60 Max. :8.000 Max. :455.0 Max. :230.0 Max. :5140   
## NA's :6   
## acceleration model.year origin car.name   
## Min. : 8.00 Min. :70.00 Min. :1.000 Length:398   
## 1st Qu.:13.82 1st Qu.:73.00 1st Qu.:1.000 Class :character   
## Median :15.50 Median :76.00 Median :1.000 Mode :character   
## Mean :15.57 Mean :76.01 Mean :1.573   
## 3rd Qu.:17.18 3rd Qu.:79.00 3rd Qu.:2.000   
## Max. :24.80 Max. :82.00 Max. :3.000   
##

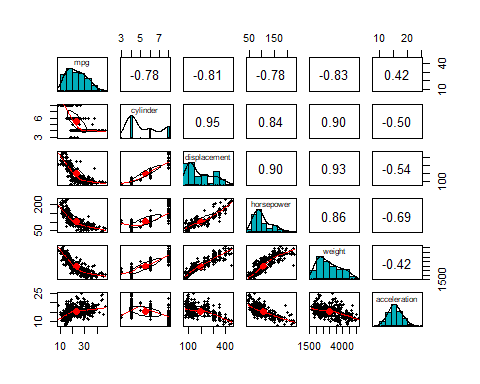
*There are 6 instances where horsepower is null. We need to remove those cases.*

# Removing all case with null value in any columns  
df <- na.omit(df)  
summary(df)

## mpg cylinder displacement horsepower weight   
## Min. : 9.00 Min. :3.000 Min. : 68.0 Min. : 46.0 Min. :1613   
## 1st Qu.:17.00 1st Qu.:4.000 1st Qu.:105.0 1st Qu.: 75.0 1st Qu.:2225   
## Median :22.75 Median :4.000 Median :151.0 Median : 93.5 Median :2804   
## Mean :23.45 Mean :5.472 Mean :194.4 Mean :104.5 Mean :2978   
## 3rd Qu.:29.00 3rd Qu.:8.000 3rd Qu.:275.8 3rd Qu.:126.0 3rd Qu.:3615   
## Max. :46.60 Max. :8.000 Max. :455.0 Max. :230.0 Max. :5140   
## acceleration model.year origin car.name   
## Min. : 8.00 Min. :70.00 Min. :1.000 Length:392   
## 1st Qu.:13.78 1st Qu.:73.00 1st Qu.:1.000 Class :character   
## Median :15.50 Median :76.00 Median :1.000 Mode :character   
## Mean :15.54 Mean :75.98 Mean :1.577   
## 3rd Qu.:17.02 3rd Qu.:79.00 3rd Qu.:2.000   
## Max. :24.80 Max. :82.00 Max. :3.000

*This summary does not show any major problems. The gap between the means and the median are not significant. This table is telling us that any data point greater than 1.5 times IQR might be an outlier and may cause an issue for our linear regression model.Please note that even though there outliers in our data, we choose to ignore them for now.*

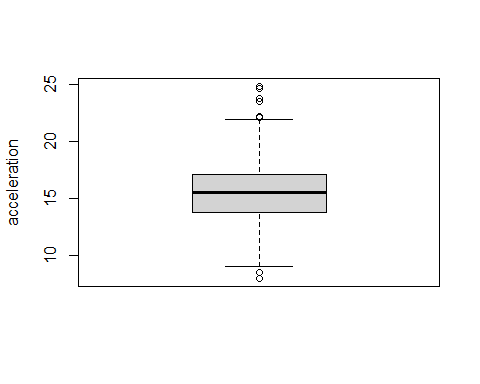
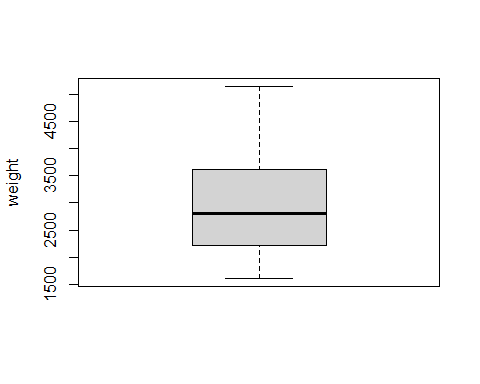
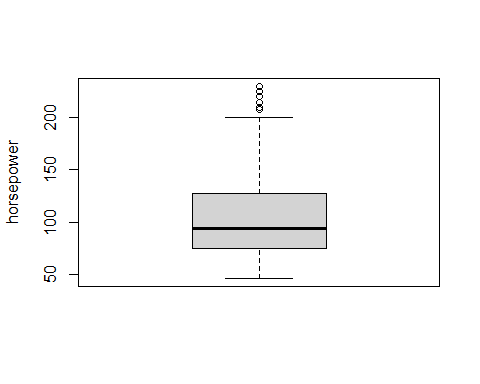
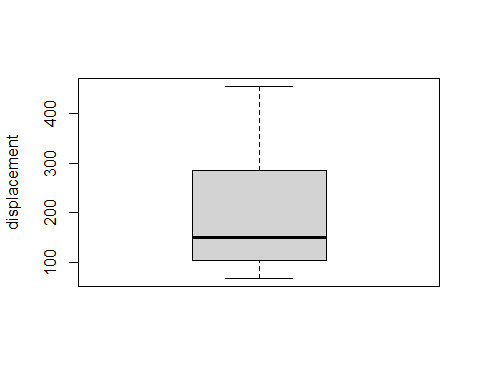
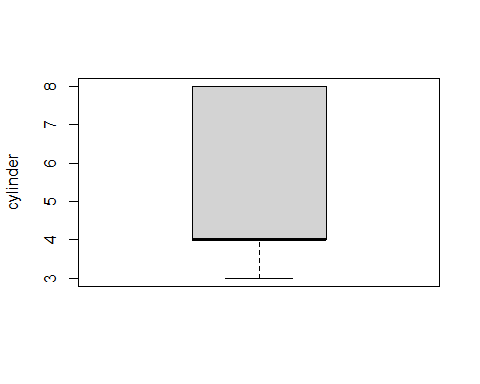
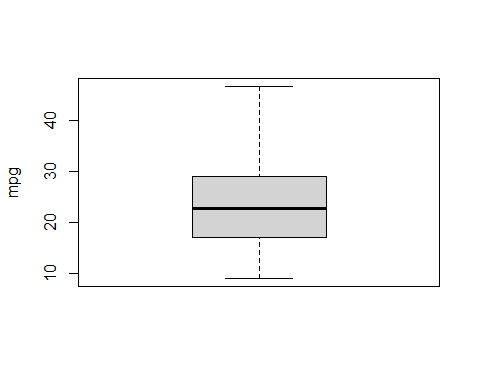
# Scatter plot of matrices  
pairs.panels(df[,1:6],method = "pearson",hist.col ="#00AFBB" ,density = TRUE,ellipses = TRUE)



*This output show three different things which are the correlation between variables, the scatter plots that tell us how the variables are associated and, the histograms the inform us how skewed our data are. We noticed that cylinder, displacement, horsepower and weight are strongly correlated and negatively correlated with the MPG. There is a multicollinearity among the independent variables. This scatter plots denote that the relationships that could be linear are the one between weight and displacement and the one between weight and horsepower, all the other are clearly not linear as they have some curve. Based on the above plots of the independent variables vs mpg, our initial assessment is to consider either weight and horsepower or weight and displacement as possible candidates for our final multi linear regression model.*

# FInding outliers using boxplot

for (i in names(df[,1:6])) {  
 boxplot(df[,i], names = "names(df[,i])", ylab = i)  
}



*We see some outliers for horsepower and acceleration on the above boxplots.*

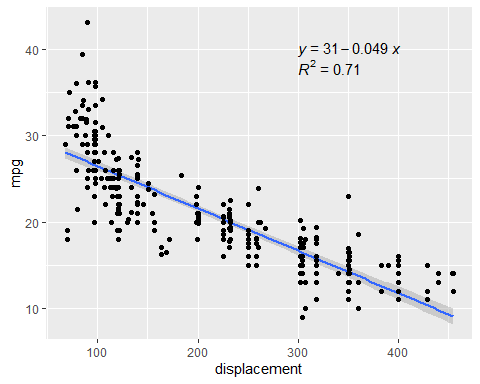
## Splitting the dataset for training and testing

#Using the rest 300 samples in the dataframe, run a simple linear regression to determine the relationship between mpg and a single variable  
df\_train <- df[1:300,1:6]  
df\_test <- df[301:nrow(df),1:6]

# Model with displacement as explanatory variable

# Linear regression plot  
ggplot(data=df\_train, aes(x=displacement, y=mpg)) +  
 geom\_smooth(method="lm") +  
 geom\_point() +  
 stat\_regline\_equation(label.x=300, label.y=40) +  
 stat\_cor(aes(label=..rr.label..), label.x=300, label.y=38)

## `geom\_smooth()` using formula 'y ~ x'



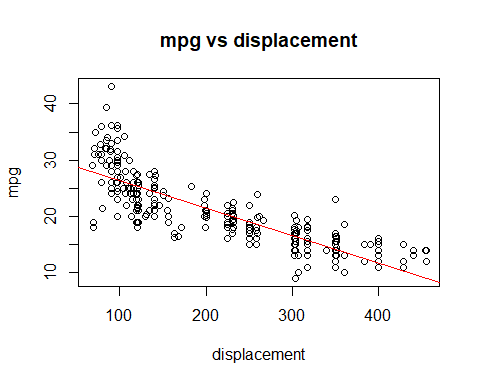
#performing regression  
dis\_model <- lm(mpg~displacement, data=df\_train)  
summary(dis\_model)

##   
## Call:  
## lm(formula = mpg ~ displacement, data = df\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.9282 -2.0043 -0.5401 1.9737 16.1501   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 31.352035 0.435875 71.93 <2e-16 \*\*\*  
## displacement -0.048913 0.001809 -27.04 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 3.412 on 298 degrees of freedom  
## Multiple R-squared: 0.7104, Adjusted R-squared: 0.7094   
## F-statistic: 731.1 on 1 and 298 DF, p-value: < 2.2e-16

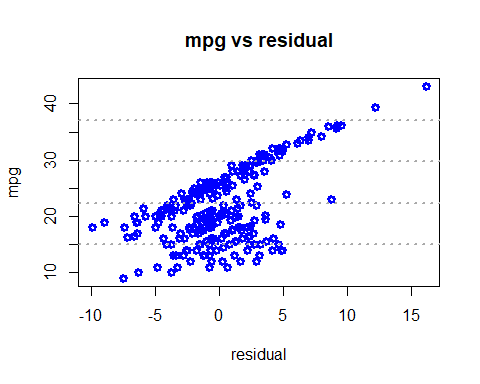
# mean of residuals  
mean(resid(dis\_model))

## [1] 3.035766e-16

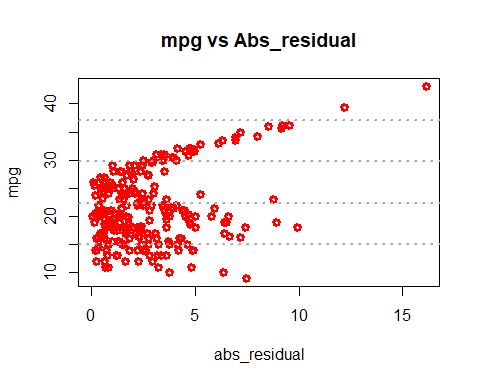
#plot the variable  
plot(df\_train$mpg~df\_train$displacement,main="mpg vs displacement",xlab="displacement",ylab = "mpg")  
abline(dis\_model,col="red")



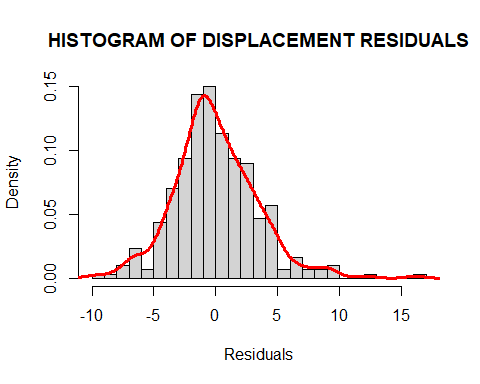
#residuals vs. the predictor variable  
residual <- dis\_model$residuals  
plot(df\_train$mpg~residual,lwd=3, col="blue",main="mpg vs residual", xlab="residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



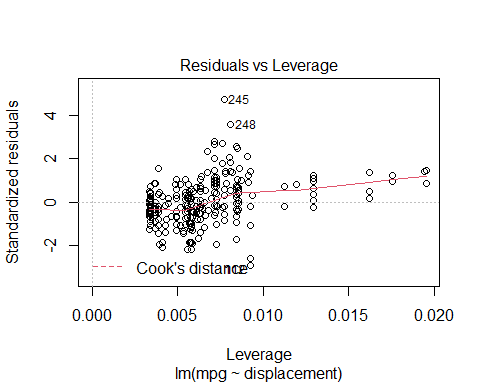
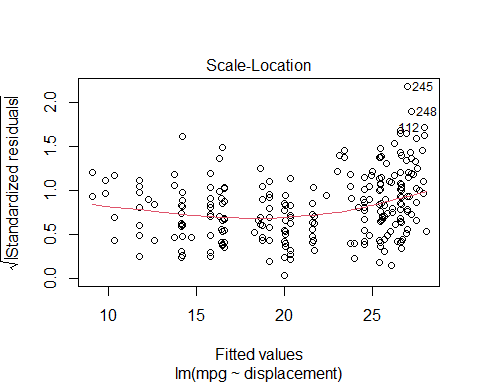
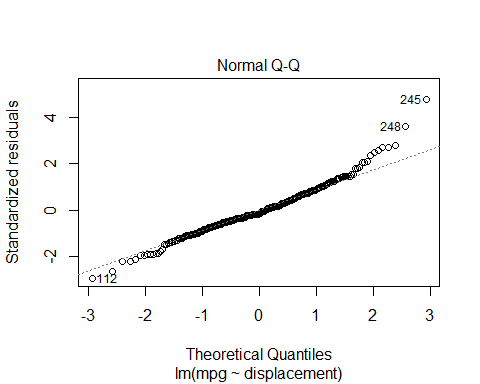
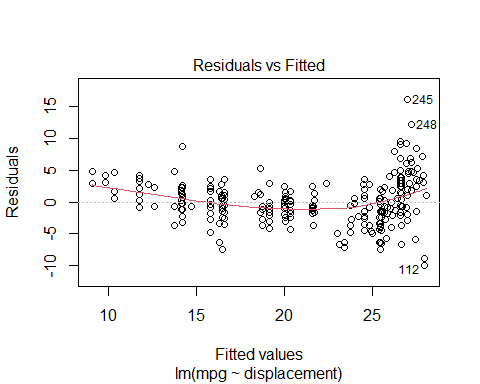
#absolute value of the residuals vs. the predictor variable  
abs\_residual <- abs(residual)  
plot(df\_train$mpg~abs\_residual,lwd=3, col="red",main="mpg vs Abs\_residual", xlab="abs\_residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



#histogram of the residuals  
hist(residual,prob=T,breaks=20,main="HISTOGRAM OF DISPLACEMENT RESIDUALS",xlab="Residuals")  
lines(density(residual),col="red",lwd=3)



plot(dis\_model)



# Make predictions and compute the R2, RMSE and MAE  
dis\_predict <- dis\_model %>% predict(df\_test)   
data.frame( R2 = R2(dis\_predict, df\_test$mpg),  
 RMSE = RMSE(dis\_predict, df\_test$mpg),  
 MAE = MAE(dis\_predict, df\_test$mpg))

## R2 RMSE MAE  
## 1 0.370789 8.371337 7.050203

prediction\_error = RMSE(dis\_predict, df\_test$mpg)/mean(df\_test$mpg)  
prediction\_error

## [1] 0.2620493

compare\_dis = as.data.frame(cbind(df\_test$mpg,dis\_predict),row=FALSE)  
names(compare\_dis) = c("observed","dis\_predict")  
head(compare\_dis)

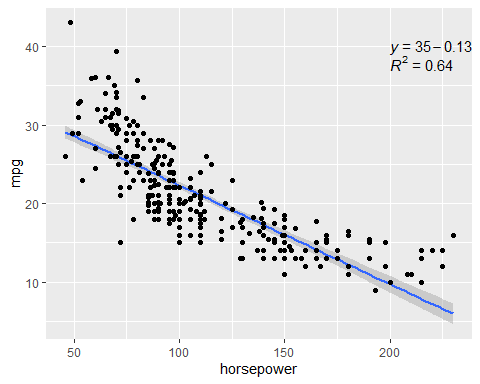
## observed dis\_predict  
## 1 34.5 26.21621  
## 2 31.8 27.19446  
## 3 37.3 26.90099  
## 4 28.4 23.96623  
## 5 28.8 22.89016  
## 6 26.8 22.89016

*In this model all the estimated values are statistically significant with a p-value less than 2e-16 . It is shown that the plot of MPG vs the Displacement is not linear and there is a sort of relationship between the variable and the residual. This model is definitely not the ideal one. The diagnostic plot reveal that these following data point 112,245,248 are outliers. The adjusted R-square state that 70.94% of displacement explain MPG.*

# Model with horsepower as explanatory variable

# Linear regression plot  
ggplot(data=df\_train, aes(x=horsepower, y=mpg)) +  
 geom\_smooth(method="lm") +  
 geom\_point() +  
 stat\_regline\_equation(label.x=200, label.y=40) +  
 stat\_cor(aes(label=..rr.label..), label.x=200, label.y=38)

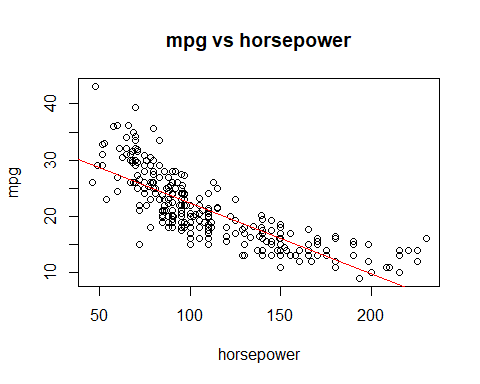
## `geom\_smooth()` using formula 'y ~ x'



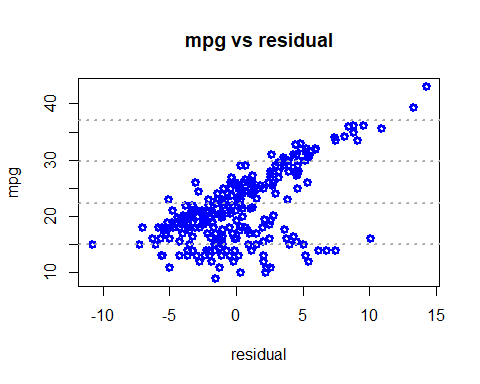
#performing regression  
hors\_model <- lm(mpg~horsepower, data=df\_train)  
summary(hors\_model)

##   
## Call:  
## lm(formula = mpg ~ horsepower, data = df\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -10.8442 -2.7816 -0.3376 2.4948 14.2360   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 34.903508 0.648037 53.86 <2e-16 \*\*\*  
## horsepower -0.125824 0.005455 -23.07 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 3.8 on 298 degrees of freedom  
## Multiple R-squared: 0.641, Adjusted R-squared: 0.6397   
## F-statistic: 532 on 1 and 298 DF, p-value: < 2.2e-16

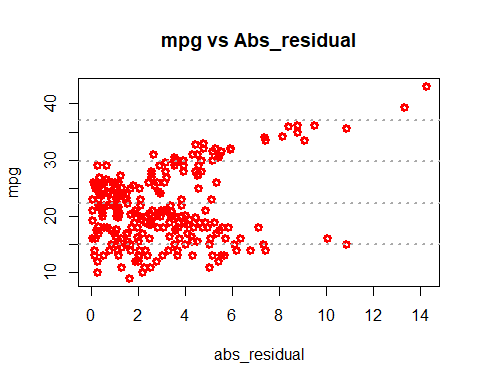
#plot the variable  
plot(df\_train$mpg~df\_train$horsepower,main="mpg vs horsepower",xlab="horsepower",ylab = "mpg")  
abline(hors\_model,col="red")



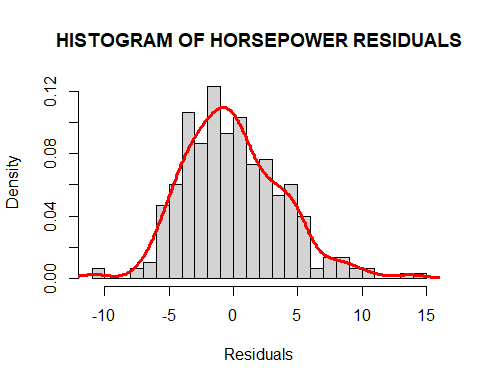
#residuals vs. the predictor variable  
residual <- hors\_model$residuals  
plot(df\_train$mpg~residual,lwd=3, col="blue",main="mpg vs residual", xlab="residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



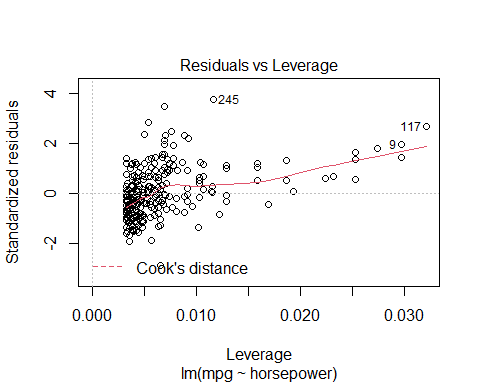
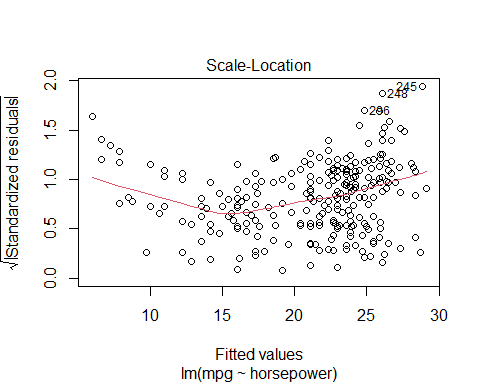
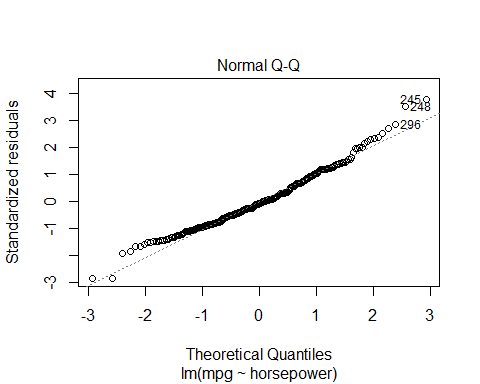
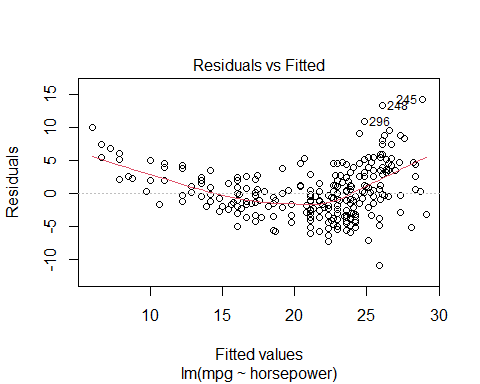
#absolute value of the residuals vs. the predictor variable  
abs\_residual <- abs(residual)  
plot(df\_train$mpg~abs\_residual,lwd=3, col="red",main="mpg vs Abs\_residual", xlab="abs\_residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



#histogram of the residuals  
hist(residual,prob=T,breaks=20,main="HISTOGRAM OF HORSEPOWER RESIDUALS",xlab="Residuals")  
lines(density(residual),col="red",lwd=3)



plot(hors\_model)



# mean of residuals  
mean(resid(hors\_model))

## [1] -8.257284e-17

# Make predictions and compute the R2, RMSE and MAE  
hors\_predict <- hors\_model %>% predict(df\_test)   
data.frame( R2 = R2(hors\_predict, df\_test$mpg),  
 RMSE = RMSE(hors\_predict, df\_test$mpg),  
 MAE = MAE(hors\_predict, df\_test$mpg))

## R2 RMSE MAE  
## 1 0.4483999 8.592932 7.508721

prediction\_error = RMSE(hors\_predict, df\_test$mpg)/mean(df\_test$mpg)  
prediction\_error

## [1] 0.268986

compare\_hors = as.data.frame(cbind(df\_test$mpg,hors\_predict),row=FALSE)  
names(compare\_hors) = c("observed","hors\_predict")  
head(compare\_hors)

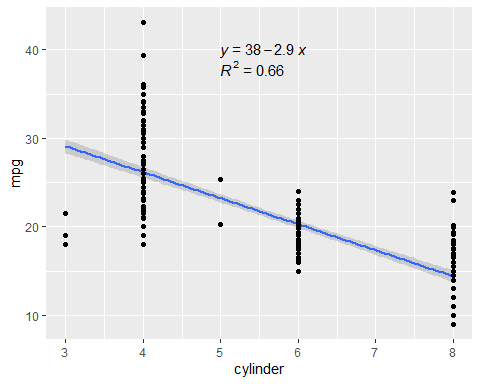
## observed hors\_predict  
## 1 34.5 26.09584  
## 2 31.8 26.72496  
## 3 37.3 26.22166  
## 4 28.4 23.57936  
## 5 28.8 20.43376  
## 6 26.8 20.43376

*In this model all the estimated values are statistically significant with a p-value less than 2e-16 . It is shown that the plot of MPG vs the horsepower is not linear and there is a sort of non-linear relationship between the variable and the residual. This model is definitely not the ideal one. The diagnostic plot reveal that these following data point 245,248,296 are outliers. The adjusted R-square state that 63.97 % of horsepower explain MPG.*

# Model with cylinder as explanatory variable

# Linear regression plot  
ggplot(data=df\_train, aes(x=cylinder, y=mpg)) +  
 geom\_smooth(method="lm") +  
 geom\_point() +  
 stat\_regline\_equation(label.x=5, label.y=40) +  
 stat\_cor(aes(label=..rr.label..), label.x=5, label.y=38)

## `geom\_smooth()` using formula 'y ~ x'



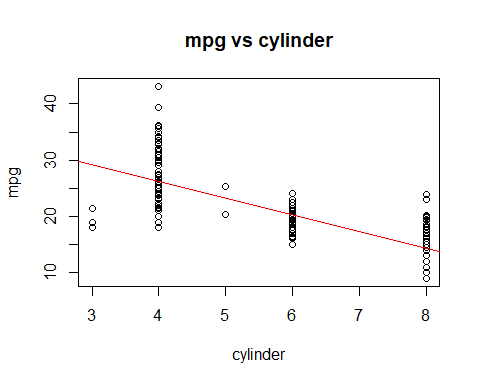
#performing regression  
cylinder\_model <- lm(mpg~cylinder, data=df\_train)  
summary(cylinder\_model)

##   
## Call:  
## lm(formula = mpg ~ cylinder, data = df\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -11.1071 -2.3012 -0.4306 1.8282 16.9282   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 37.9130 0.7356 51.54 <2e-16 \*\*\*  
## cylinder -2.9353 0.1211 -24.24 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 3.678 on 298 degrees of freedom  
## Multiple R-squared: 0.6636, Adjusted R-squared: 0.6624   
## F-statistic: 587.8 on 1 and 298 DF, p-value: < 2.2e-16

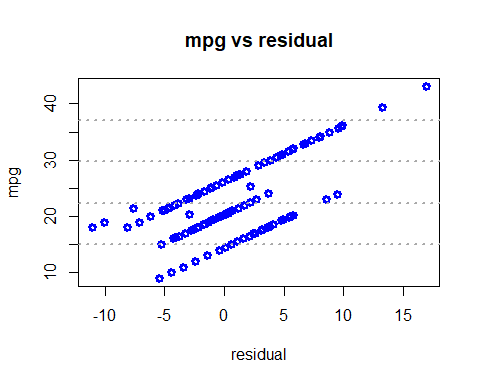
# mean of residuals  
mean(resid(cylinder\_model))

## [1] -4.930084e-17

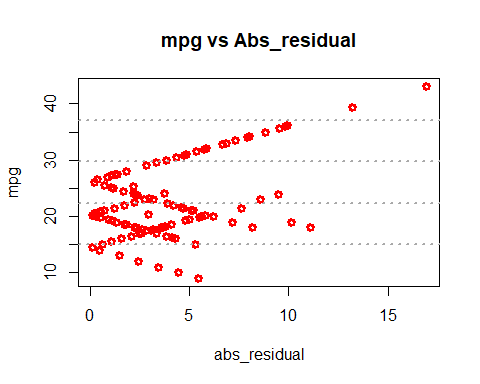
#plot the variable  
plot(df\_train$mpg~df\_train$cylinder,main="mpg vs cylinder",xlab="cylinder",ylab = "mpg")  
abline(cylinder\_model,col="red")



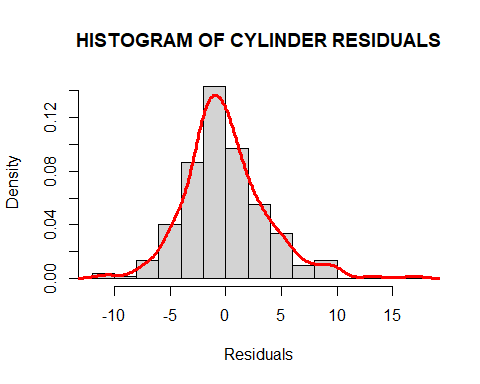
#residuals vs. the predictor variable  
residual <- cylinder\_model$residuals  
plot(df\_train$mpg~residual,lwd=3, col="blue",main="mpg vs residual", xlab="residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



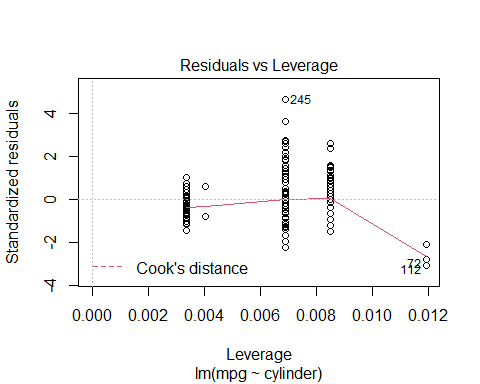
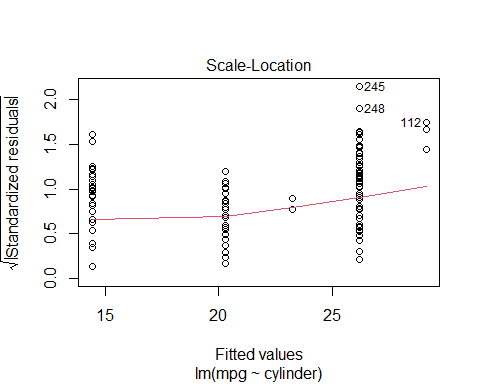
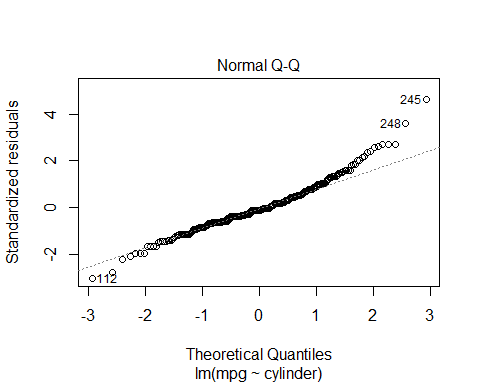
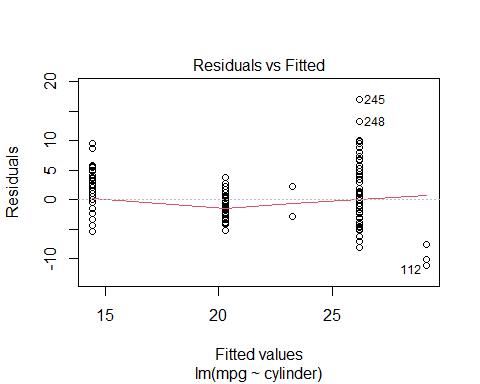
#absolute value of the residuals vs. the predictor variable  
abs\_residual <- abs(residual)  
plot(df\_train$mpg~abs\_residual,lwd=3, col="red",main="mpg vs Abs\_residual", xlab="abs\_residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



#histogram of the residuals  
hist(residual,prob=T,breaks=20,main="HISTOGRAM OF CYLINDER RESIDUALS",xlab="Residuals")  
lines(density(residual),col="red",lwd=3)



plot(cylinder\_model)



# Make predictions and compute the R2, RMSE and MAE  
cyl\_predict <- cylinder\_model %>% predict(df\_test)   
data.frame( R2 = R2(cyl\_predict, df\_test$mpg),  
 RMSE = RMSE(cyl\_predict, df\_test$mpg),  
 MAE = MAE(cyl\_predict, df\_test$mpg))

## R2 RMSE MAE  
## 1 0.1829322 8.611541 7.099133

prediction\_error = RMSE(cyl\_predict, df\_test$mpg)/mean(df\_test$mpg)  
prediction\_error

## [1] 0.2695685

compare\_cyl = as.data.frame(cbind(df\_test$mpg,cyl\_predict),row=FALSE)  
names(compare\_cyl) = c("observed","cyl\_predict")  
head(compare\_cyl)

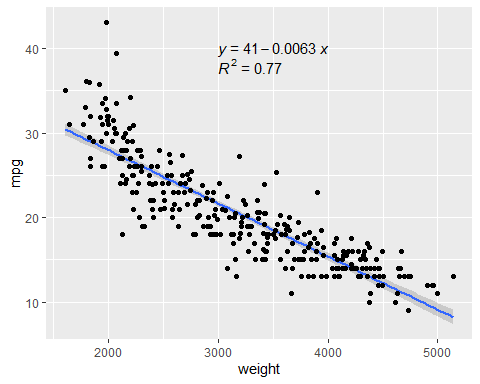
## observed cyl\_predict  
## 1 34.5 26.17179  
## 2 31.8 26.17179  
## 3 37.3 26.17179  
## 4 28.4 26.17179  
## 5 28.8 20.30120  
## 6 26.8 20.30120

*In this model all the estimated values are statistically significant with a p-value less than 2e-16 . It is shown that the plot of MPG vs the cylinder is not linear and there is a sort of non-linear relationship between the variable and the residual. This model is definitely not the ideal one. The diagnostic plot reveal that these following data point 245,248,112 are outliers. The adjusted R-square state that 66.24 % of cylinder explain MPG.*

# Model with weight as explanatory variable

# Linear regression plot  
ggplot(data=df\_train, aes(x=weight, y=mpg)) +  
 geom\_smooth(method="lm") +  
 geom\_point() +  
 stat\_regline\_equation(label.x=3000, label.y=40) +  
 stat\_cor(aes(label=..rr.label..), label.x=3000, label.y=38)

## `geom\_smooth()` using formula 'y ~ x'



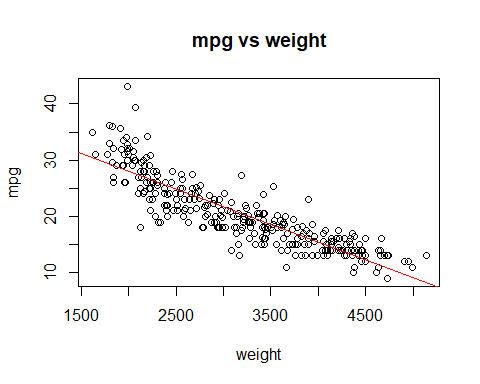
#performing regression  
weight\_model <- lm(mpg~weight, data=df\_train)  
summary(weight\_model)

##   
## Call:  
## lm(formula = mpg ~ weight, data = df\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.2011 -1.9157 -0.0812 1.7341 15.0246   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 40.5619792 0.6461532 62.77 <2e-16 \*\*\*  
## weight -0.0062905 0.0001984 -31.71 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 3.032 on 298 degrees of freedom  
## Multiple R-squared: 0.7714, Adjusted R-squared: 0.7706   
## F-statistic: 1005 on 1 and 298 DF, p-value: < 2.2e-16

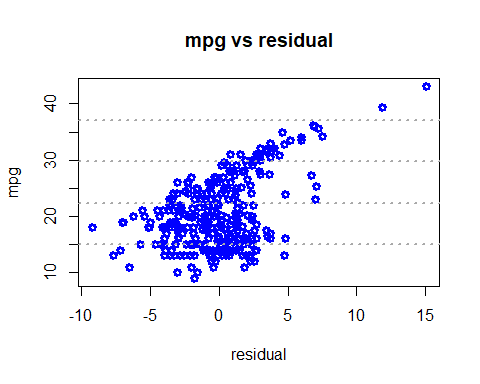
# mean of residuals  
mean(resid(weight\_model))

## [1] 2.543538e-16

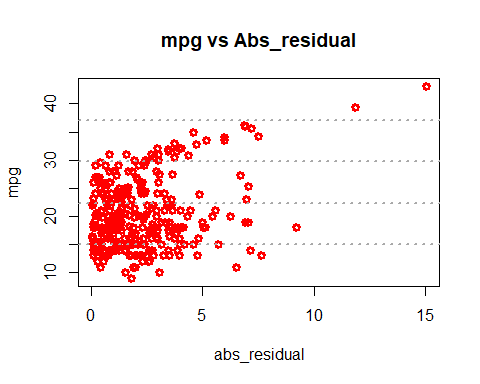
#plot the variable  
plot(df\_train$mpg~df\_train$weight,main="mpg vs weight",xlab="weight",ylab = "mpg")  
abline(weight\_model,col="red")



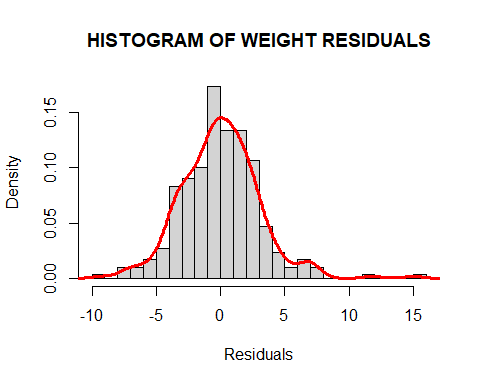
#residuals vs. the predictor variable  
residual <- weight\_model$residuals  
plot(df\_train$mpg~residual,lwd=3, col="blue",main="mpg vs residual", xlab="residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



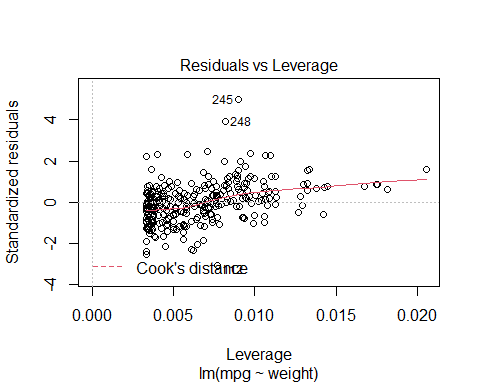
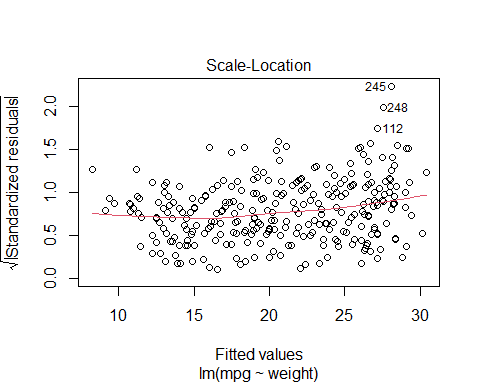
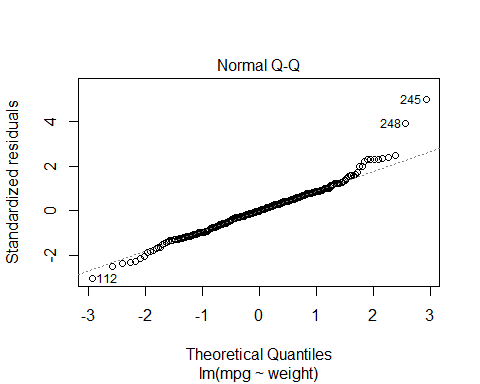
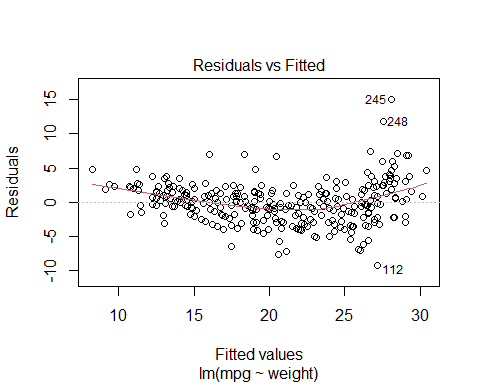
#absolute value of the residuals vs. the predictor variable  
abs\_residual <- abs(residual)  
plot(df\_train$mpg~abs\_residual,lwd=3, col="red",main="mpg vs Abs\_residual", xlab="abs\_residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



#histogram of the residuals  
hist(residual,prob=T,breaks=20,main="HISTOGRAM OF WEIGHT RESIDUALS",xlab="Residuals")  
lines(density(residual),col="red",lwd=3)



plot(weight\_model)



# Make predictions and compute the R2, RMSE and MAE  
weight\_predict <- weight\_model %>% predict(df\_test)   
data.frame( R2 = R2(weight\_predict, df\_test$mpg),  
 RMSE = RMSE(weight\_predict, df\_test$mpg),  
 MAE = MAE(weight\_predict, df\_test$mpg))

## R2 RMSE MAE  
## 1 0.5006516 8.157758 6.983514

prediction\_error = RMSE(weight\_predict, df\_test$mpg)/mean(df\_test$mpg)  
prediction\_error

## [1] 0.2553637

compare\_wght = as.data.frame(cbind(df\_test$mpg,weight\_predict),row=FALSE)  
names(compare\_wght) = c("observed","weight\_predict")  
head(compare\_wght)

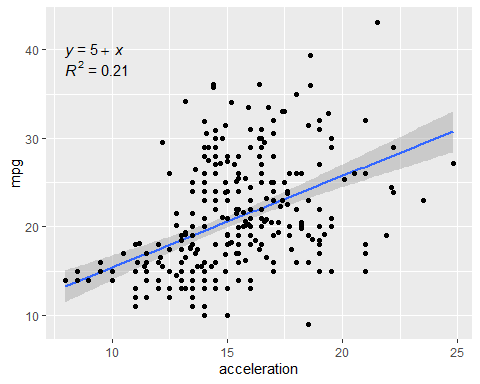
## observed weight\_predict  
## 1 34.5 27.03751  
## 2 31.8 27.85526  
## 3 37.3 27.16331  
## 4 28.4 23.76647  
## 5 28.8 24.23825  
## 6 26.8 23.57776

*In this model all the estimated values are statistically significant with a p-value less than 2e-16 . It is shown that the plot of MPG vs the weight is not linear and there is a sort of non-linear relationship between the variable and the residual. This model is definitely not the ideal one. The diagnostic plot reveal that these following data point 245,248,112 are outliers. The adjusted R-square state that 77.06% of weight explain MPG.*

# Model with acceleration as explanatory variable

# Linear regression plot  
ggplot(data=df\_train, aes(x=acceleration, y=mpg)) +  
 geom\_smooth(method="lm") +  
 geom\_point() +  
 stat\_regline\_equation(label.x=8, label.y=40) +  
 stat\_cor(aes(label=..rr.label..), label.x=8, label.y=38)

## `geom\_smooth()` using formula 'y ~ x'



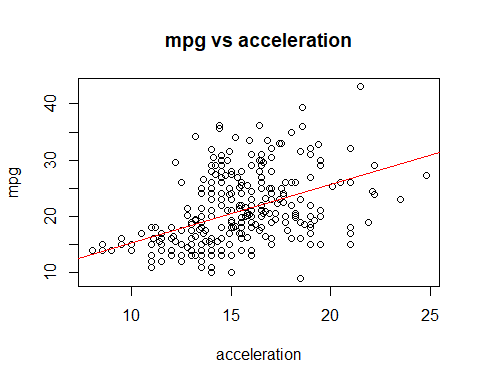
#performing regression  
acc\_model <- lm(mpg~acceleration, data=df\_train)  
summary(acc\_model)

##   
## Call:  
## lm(formula = mpg ~ acceleration, data = df\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -15.202 -4.126 -1.012 3.268 16.154   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.0012 1.8352 2.725 0.00681 \*\*   
## acceleration 1.0379 0.1183 8.770 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.654 on 298 degrees of freedom  
## Multiple R-squared: 0.2052, Adjusted R-squared: 0.2025   
## F-statistic: 76.91 on 1 and 298 DF, p-value: < 2.2e-16

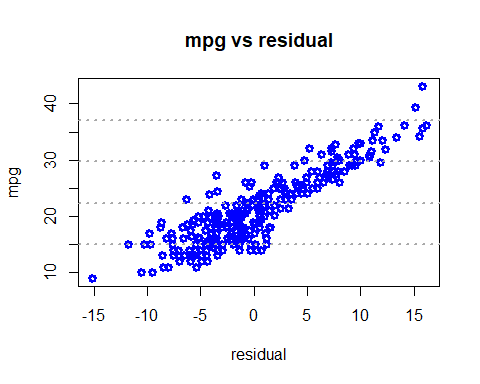
# mean of residuals  
mean(resid(acc\_model))

## [1] -6.274957e-16

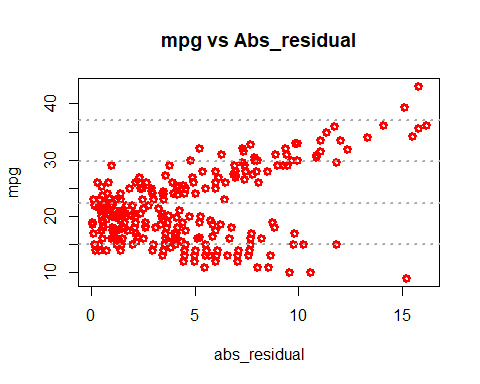
#plot the variable  
plot(df\_train$mpg~df\_train$acceleration,main="mpg vs acceleration",xlab="acceleration",ylab = "mpg")  
abline(acc\_model,col="red")



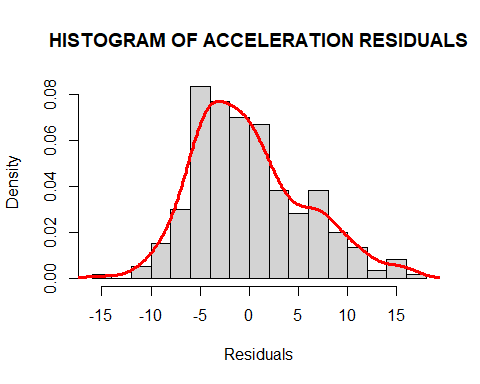
#residuals vs. the predictor variable  
residual <- acc\_model$residuals  
plot(df\_train$mpg~residual,lwd=3, col="blue",main="mpg vs residual", xlab="residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



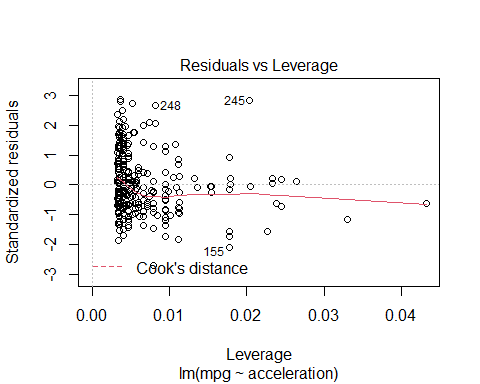
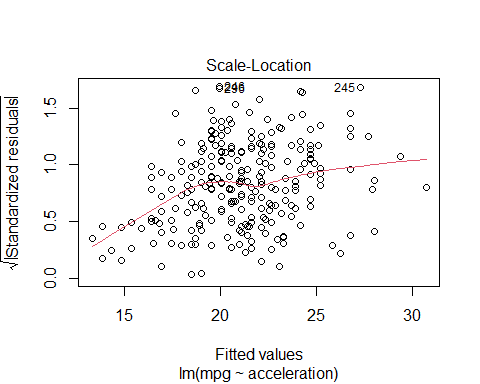
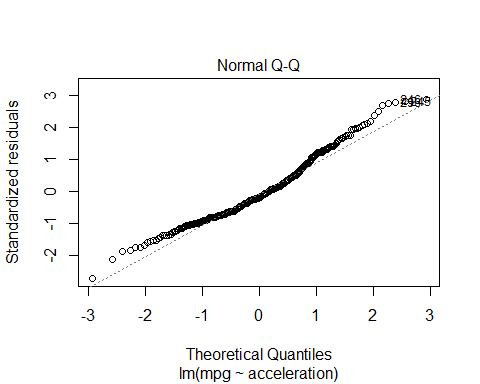
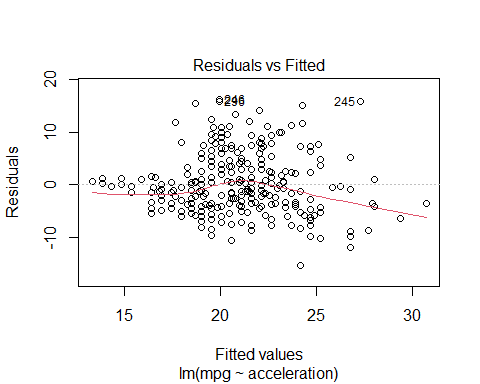
#absolute value of the residuals vs. the predictor variable  
abs\_residual <- abs(residual)  
plot(df\_train$mpg~abs\_residual,lwd=3, col="red",main="mpg vs Abs\_residual", xlab="abs\_residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



#histogram of the residuals  
hist(residual,prob=T,breaks=20,main="HISTOGRAM OF ACCELERATION RESIDUALS",xlab="Residuals")  
lines(density(residual),col="red",lwd=3)



plot(acc\_model)



# Make predictions and compute the R2, RMSE and MAE  
acc\_predict <- acc\_model %>% predict(df\_test)   
data.frame( R2 = R2(acc\_predict, df\_test$mpg),  
 RMSE = RMSE(acc\_predict, df\_test$mpg),  
 MAE = MAE(acc\_predict, df\_test$mpg))

## R2 RMSE MAE  
## 1 0.03597167 11.51665 10.16914

prediction\_error = RMSE(acc\_predict, df\_test$mpg)/mean(df\_test$mpg)  
prediction\_error

## [1] 0.3605077

compare\_acc = as.data.frame(cbind(df\_test$mpg,acc\_predict),row=FALSE)  
names(compare\_acc) = c("observed","acc\_predict")  
head(compare\_acc)

## observed acc\_predict  
## 1 34.5 20.46536  
## 2 31.8 24.92818  
## 3 37.3 20.25778  
## 4 28.4 21.60701  
## 5 28.8 16.72904  
## 6 26.8 18.38963

*In this model all the estimated values are statistically significant with a p-value less than 2e-16 . It is shown that the plot of MPG vs the acceleration is not linear and there is a sort of non-linear relationship between the variable and the residual. This model is probably the worst one. The diagnostic plot reveal that these following data point 245,248,155 are outliers. The adjusted R-square state that only 20.25% of acceleration explain MPG.*

# Multiple regression

## Feature selection

### Stepwise regression

# To find out which independent variable to use in our multiple regression we are going to use the step wise regression  
null=lm(mpg~1,data=df\_train)  
full=lm(mpg~.,data=df\_train)  
step(null,scope=list(upper=full),data=df\_train,direction="both")

## Start: AIC=1108.25  
## mpg ~ 1  
##   
## Df Sum of Sq RSS AIC  
## + weight 1 9243.7 2739.6 667.54  
## + displacement 1 8513.2 3470.1 738.45  
## + cylinder 1 7951.8 4031.5 783.43  
## + horsepower 1 7680.7 4302.6 802.96  
## + acceleration 1 2458.4 9524.9 1041.36  
## <none> 11983.3 1108.25  
##   
## Step: AIC=667.54  
## mpg ~ weight  
##   
## Df Sum of Sq RSS AIC  
## + horsepower 1 98.2 2641.5 658.59  
## + displacement 1 59.9 2679.7 662.91  
## + acceleration 1 41.1 2698.6 665.01  
## + cylinder 1 34.9 2704.8 665.70  
## <none> 2739.6 667.54  
## - weight 1 9243.7 11983.3 1108.25  
##   
## Step: AIC=658.59  
## mpg ~ weight + horsepower  
##   
## Df Sum of Sq RSS AIC  
## <none> 2641.5 658.59  
## + cylinder 1 10.65 2630.8 659.38  
## + displacement 1 9.81 2631.7 659.48  
## + acceleration 1 1.00 2640.5 660.48  
## - horsepower 1 98.16 2739.6 667.54  
## - weight 1 1661.08 4302.6 802.96

##   
## Call:  
## lm(formula = mpg ~ weight + horsepower, data = df\_train)  
##   
## Coefficients:  
## (Intercept) weight horsepower   
## 40.258743 -0.005204 -0.027759

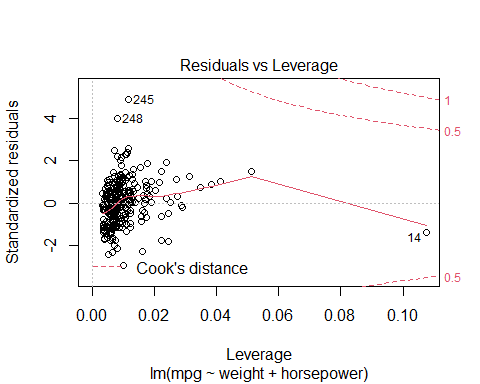
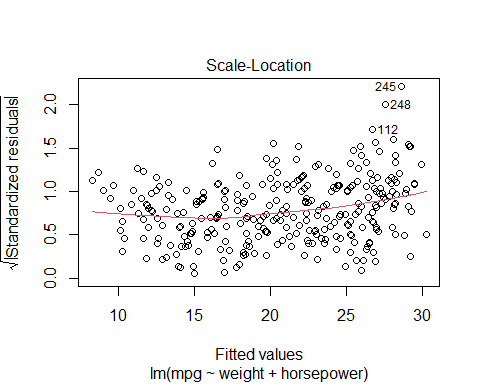
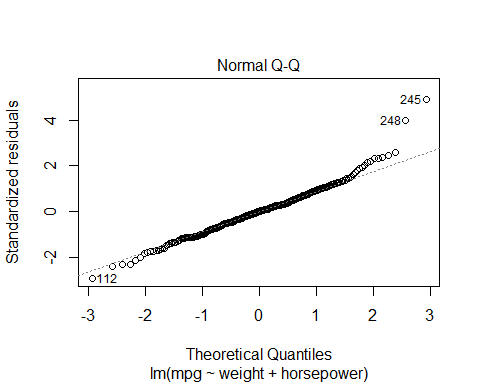
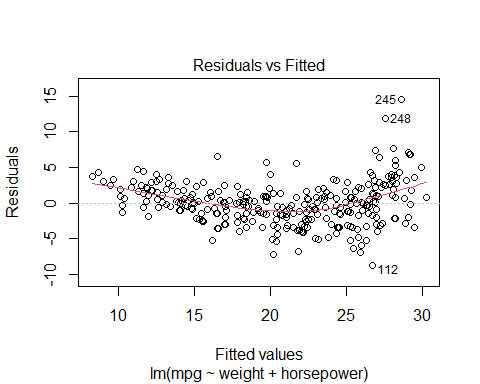
*As shown in the R results above the step function produced weight and horsepower as the optimum variables to produce the desired linear regression model . This step function use AIC as criterion. It selects the combination of variables with which we can have the lower AIC statistic.This would be our final model.*

# Building the final model

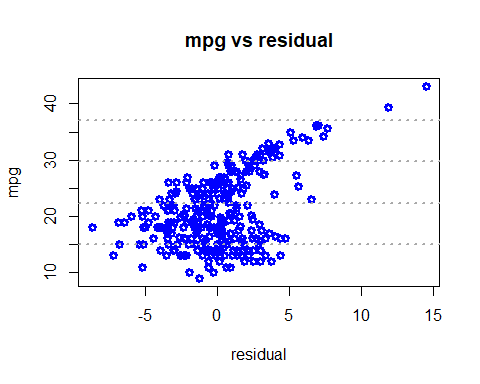
final\_model <- lm(mpg ~ weight + horsepower, data = df\_train)  
summary(final\_model)

##   
## Call:  
## lm(formula = mpg ~ weight + horsepower, data = df\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -8.7069 -1.8380 0.0207 1.6877 14.5038   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 40.2587429 0.6420610 62.702 < 2e-16 \*\*\*  
## weight -0.0052041 0.0003808 -13.666 < 2e-16 \*\*\*  
## horsepower -0.0277594 0.0083560 -3.322 0.00101 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.982 on 297 degrees of freedom  
## Multiple R-squared: 0.7796, Adjusted R-squared: 0.7781   
## F-statistic: 525.2 on 2 and 297 DF, p-value: < 2.2e-16

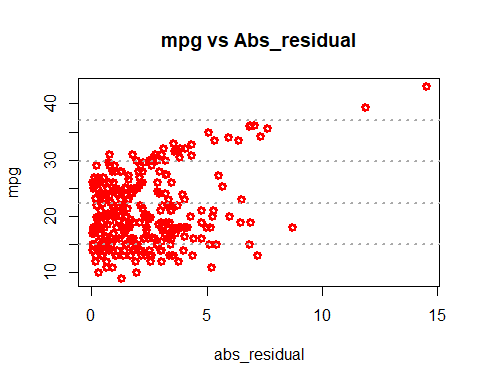
plot(final\_model)



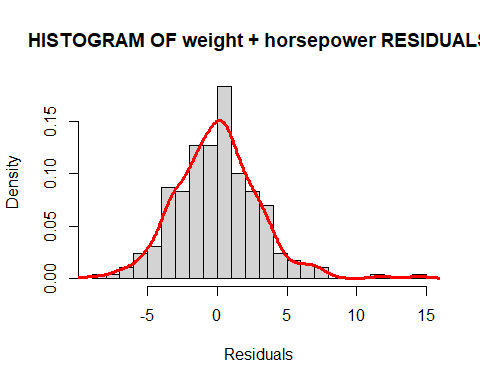
#residuals vs. the predictor variable  
residual <- final\_model$residuals  
plot(df\_train$mpg~residual,lwd=3, col="blue",main="mpg vs residual", xlab="residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



#absolute value of the residuals vs. the predictor variable  
abs\_residual <- abs(residual)  
plot(df\_train$mpg~abs\_residual,lwd=3, col="red",main="mpg vs Abs\_residual", xlab="abs\_residual",ylab = "mpg")  
grid(NA, 5, lwd = 2,col = "darkgray")



#histogram of the residuals  
hist(residual,prob=T,breaks=20,main="HISTOGRAM OF weight + horsepower RESIDUALS",xlab="Residuals")  
lines(density(residual),col="red",lwd=3)



*From the model summary,the model p value and predictor’s p value are less than the significance level (5%).We can state that our model is statistically significant.The R-Squared and the Adjusted R-Squared are respectively 0.7796 and 0.7781 which mean that more than 77% of the model are explained by the independent variables. We decided to plot test weight against test mpg and found that the linear regression line through this data confirm our suspicion that the linear regression model is not truly representing the data. From the plot (residual vs fitted), the red line is pretty flat, with no increasing or decreasing trend. The Normal Q-Q plot indicate that the data are normally distributed.These plot are showing some outliers that we decided to keep in the previous models.*

# Global Validation of Linear Models Assumptions  
gvlma(final\_model)

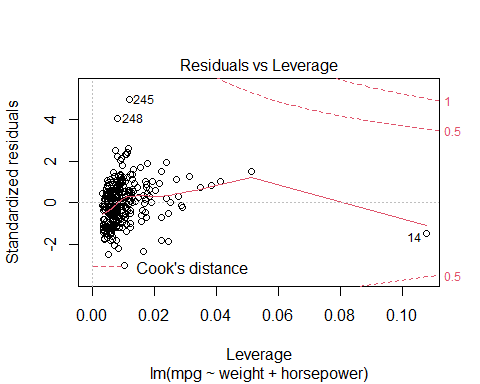
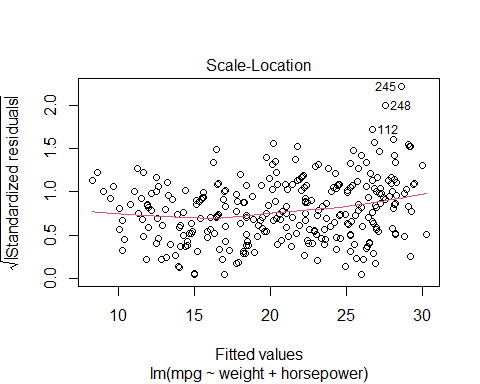
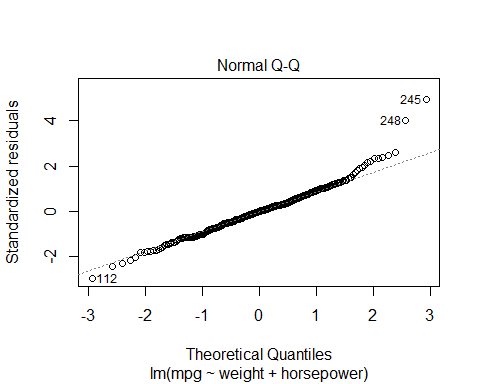
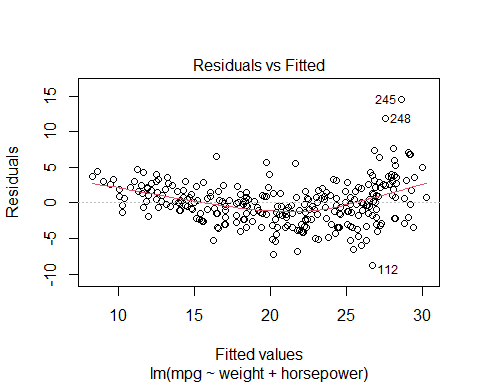
##   
## Call:  
## lm(formula = mpg ~ weight + horsepower, data = df\_train)  
##   
## Coefficients:  
## (Intercept) weight horsepower   
## 40.258743 -0.005204 -0.027759   
##   
##   
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS  
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:  
## Level of Significance = 0.05   
##   
## Call:  
## gvlma(x = final\_model)   
##   
## Value p-value Decision  
## Global Stat 150.774 0.000e+00 Assumptions NOT satisfied!  
## Skewness 16.365 5.225e-05 Assumptions NOT satisfied!  
## Kurtosis 59.193 1.432e-14 Assumptions NOT satisfied!  
## Link Function 65.957 4.441e-16 Assumptions NOT satisfied!  
## Heteroscedasticity 9.259 2.343e-03 Assumptions NOT satisfied!

*None of the assumptions are satisfied. This could be due to the fact that we have some outliers in the data that can impact the quality of the model. Let’s remove them from the data then see what happen.*

final\_model2 <- lm(mpg ~ weight + horsepower, data = df\_train[-c(112,245,248),])  
summary(final\_model2)

##   
## Call:  
## lm(formula = mpg ~ weight + horsepower, data = df\_train[-c(112,   
## 245, 248), ])  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -8.7327 -1.7911 0.0047 1.6783 14.5074   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 40.2859314 0.6408855 62.860 < 2e-16 \*\*\*  
## weight -0.0052394 0.0003785 -13.844 < 2e-16 \*\*\*  
## horsepower -0.0269430 0.0083047 -3.244 0.00131 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.96 on 294 degrees of freedom  
## Multiple R-squared: 0.7824, Adjusted R-squared: 0.7809   
## F-statistic: 528.6 on 2 and 294 DF, p-value: < 2.2e-16

plot(final\_model2)



# Global Validation of Linear Models Assumptions  
gvlma(final\_model2)

##   
## Call:  
## lm(formula = mpg ~ weight + horsepower, data = df\_train[-c(112,   
## 245, 248), ])  
##   
## Coefficients:  
## (Intercept) weight horsepower   
## 40.285931 -0.005239 -0.026943   
##   
##   
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS  
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:  
## Level of Significance = 0.05   
##   
## Call:  
## gvlma(x = final\_model2)   
##   
## Value p-value Decision  
## Global Stat 158.37 0.000e+00 Assumptions NOT satisfied!  
## Skewness 18.69 1.537e-05 Assumptions NOT satisfied!  
## Kurtosis 64.02 1.221e-15 Assumptions NOT satisfied!  
## Link Function 65.56 5.551e-16 Assumptions NOT satisfied!  
## Heteroscedasticity 10.09 1.490e-03 Assumptions NOT satisfied!

*We notice that even if we remove the outlier the linear model assumptions are not satisfied. R-squared and adjusted R-squared values did not improve much even after removing the outliers. We suspect that a non linear model would perform better on this car data.*

# Prediction

# Make predictions and compute the R2, RMSE and MAE  
predict\_final <- final\_model2 %>% predict(df\_test)   
data.frame( R2 = R2(predict\_final, df\_test$mpg),  
 RMSE = RMSE(predict\_final, df\_test$mpg),  
 MAE = MAE(predict\_final, df\_test$mpg))

## R2 RMSE MAE  
## 1 0.5427488 8.0093 6.881424

predictions\_error <- RMSE(predict\_final, df\_test$mpg)/mean(df\_test$mpg)  
predictions\_error

## [1] 0.2507164

compare\_final <- as.data.frame(cbind(df\_test$mpg,predict\_final),row=FALSE)  
names(compare\_final) <- c("observed","predict\_final")  
head(compare\_final)

## observed predict\_final  
## 1 34.5 27.13531  
## 2 31.8 27.95114  
## 3 37.3 27.26704  
## 4 28.4 23.87199  
## 5 28.8 23.59136  
## 6 26.8 23.04123

cor(compare\_final)

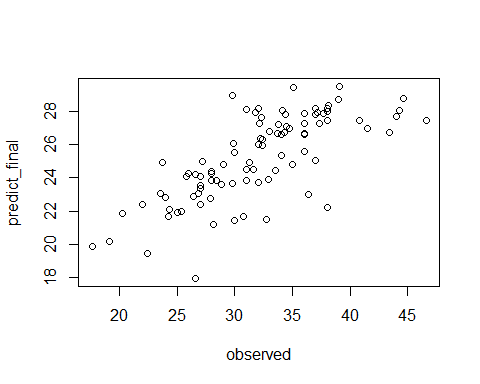
## observed predict\_final  
## observed 1.0000000 0.7367148  
## predict\_final 0.7367148 1.0000000

*The correlation between The actual values and the predicted values is 73.74% .*

summary(compare\_final)

## observed predict\_final   
## Min. :17.60 Min. :17.94   
## 1st Qu.:27.73 1st Qu.:23.49   
## Median :32.05 Median :25.21   
## Mean :31.95 Mean :25.21   
## 3rd Qu.:36.00 3rd Qu.:27.46   
## Max. :46.60 Max. :29.53

plot(compare\_final)



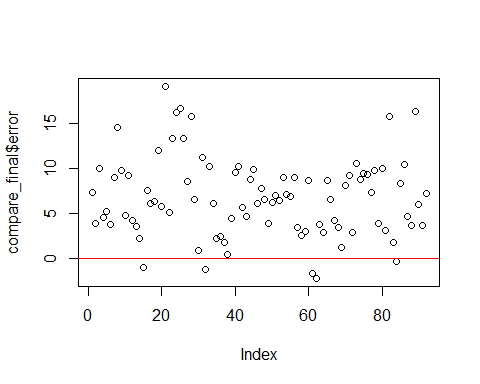
compare\_final$error = compare\_final$observed - compare\_final$predict\_final  
# compare\_final$residuals = final\_model$residuals  
head(compare\_final)

## observed predict\_final error  
## 1 34.5 27.13531 7.364688  
## 2 31.8 27.95114 3.848857  
## 3 37.3 27.26704 10.032958  
## 4 28.4 23.87199 4.528012  
## 5 28.8 23.59136 5.208635  
## 6 26.8 23.04123 3.758767

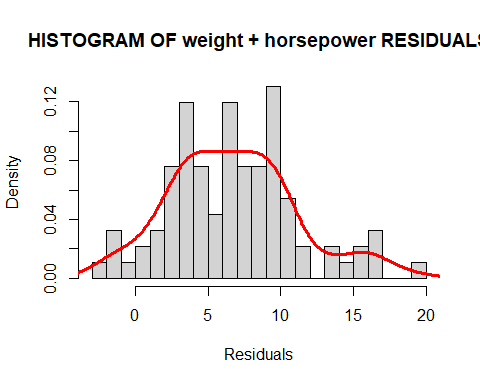
summary(compare\_final)

## observed predict\_final error   
## Min. :17.60 Min. :17.94 Min. :-2.241   
## 1st Qu.:27.73 1st Qu.:23.49 1st Qu.: 3.712   
## Median :32.05 Median :25.21 Median : 6.471   
## Mean :31.95 Mean :25.21 Mean : 6.738   
## 3rd Qu.:36.00 3rd Qu.:27.46 3rd Qu.: 9.255   
## Max. :46.60 Max. :29.53 Max. :19.120

# Residuals plot  
plot(compare\_final$error)  
abline(h = 0, col= 'red')



#histogram of the residuals  
hist(compare\_final$error,prob=T,breaks=20,main="HISTOGRAM OF weight + horsepower RESIDUALS",xlab="Residuals")  
lines(density(compare\_final$error),col="red",lwd=3)



*So it seems like our regression model is predicting more than the actual value most of the time.*

# So the final formula is mpg = 40.2859314 - 0.0052394 \* weight - 0.0269430 \* horsepower