Final Exam

Vaishnavi Badame

2020-08-11

Question : How much should I expect to pay for a used Toyota Corolla?

```
library(knitr)
options(warn = -1)
opts_chunk$set(tidy.opts = list(width.cutoff = 50), tidy = TRUE)
library(MASS)
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
require(caTools)
```

```
## Loading required package: caTools
```

```
library(Metrics)
library(leaps)
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.0-2
```

```
library(DMwR)
```

```
## Loading required package: lattice
```

```
## Loading required package: grid
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
setwd("/Users/vaishnavibadame/downloads")
Toyota <- read.csv(file = 'toyotaCorolla.csv', sep = ',', header = TRUE,
                   na.strings = '?', stringsAsFactors = T, encoding = 'UTF-8')
```

Dataset Description and Data Cleaning

The data set uploaded has 37 attributes and 1436 numbers of records. The data set has 24 qualitative variables that may have been assigned characters or a numerical 1 and 0 for a 'Yes' and 'No' value respectively and has 13 quantitative variables. I have planned to split the data set in the 70:30 ratio for training data set and test data set respectively.

```
dim(Toyota)
```

```
## [1] 1436   37
```

```
summary(Toyota)
```

```
##        Id                                                    Model
##  Min.   :   1.0   TOYOTA Corolla 1.6 16V HATCHB LINEA TERRA 2/3-Doors: 107
##  1st Qu.: 361.8   TOYOTA Corolla 1.3 16V HATCHB LINEA TERRA 2/3-Doors:  83
##  Median : 721.5   TOYOTA Corolla 1.6 16V LIFTB LINEA LUNA 4/5-Doors  :  79
##  Mean   : 721.6   TOYOTA Corolla 1.6 16V LIFTB LINEA TERRA 4/5-Doors :  70
##  3rd Qu.:1081.2   TOYOTA Corolla 1.6 16V SEDAN LINEA TERRA 4/5-Doors :  43
##  Max.   :1442.0   TOYOTA Corolla 1.4 16V VVT I HATCHB TERRA 2/3-Doors:  42
##                   (Other)                                            :1012
##      Price        Age_08_04       Mfg_Month         Mfg_Year
##  8950   : 108   Min.   : 1.00   Min.   : 1.000   Min.   :1998
##  9950   :  83   1st Qu.:44.00   1st Qu.: 3.000   1st Qu.:1998
##  7950   :  63   Median :61.00   Median : 5.000   Median :1999
##  10950  :  62   Mean   :55.94   Mean   : 5.549   Mean   :2000
##  11950  :  47   3rd Qu.:70.00   3rd Qu.: 8.000   3rd Qu.:2001
##  8750   :  41   Max.   :80.00   Max.   :12.000   Max.   :2004
##  (Other):1032   NA's   :1
##        KM          Fuel_Type          HP          Met_Color
##  Min.   :     1   CNG   :  17   Min.   : 69.0   Min.   :0.0000
##  1st Qu.: 43000   Diesel: 155   1st Qu.: 90.0   1st Qu.:0.0000
##  Median : 63390   Petrol:1264   Median :110.0   Median :1.0000
##  Mean   : 68533                 Mean   :101.5   Mean   :0.6748
##  3rd Qu.: 87021                 3rd Qu.:110.0   3rd Qu.:1.0000
##  Max.   :243000                 Max.   :192.0   Max.   :1.0000
##
##    Automatic             cc            Doors         Cylinders     Gears
##  Min.   :0.00000   Min.   : 1300   Min.   :2.000   Min.   :4   Min.   :3.000
##  1st Qu.:0.00000   1st Qu.: 1400   1st Qu.:3.000   1st Qu.:4   1st Qu.:5.000
##  Median :0.00000   Median : 1600   Median :4.000   Median :4   Median :5.000
##  Mean   :0.05571   Mean   : 1577   Mean   :4.033   Mean   :4   Mean   :5.026
##  3rd Qu.:0.00000   3rd Qu.: 1600   3rd Qu.:5.000   3rd Qu.:4   3rd Qu.:5.000
##  Max.   :1.00000   Max.   :16000   Max.   :5.000   Max.   :4   Max.   :6.000
##
##  Quarterly_Tax        Weight     Mfr_Guarantee    BOVAG_Guarantee
##  Min.   : 19.00   Min.   :1000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.: 69.00   1st Qu.:1040   1st Qu.:0.0000   1st Qu.:1.0000
##  Median : 85.00   Median :1070   Median :0.0000   Median :1.0000
##  Mean   : 87.12   Mean   :1072   Mean   :0.4095   Mean   :0.8955
##  3rd Qu.: 85.00   3rd Qu.:1085   3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :283.00   Max.   :1615   Max.   :1.0000   Max.   :1.0000
```

```
##
##  Guarantee_Period      ABS            Airbag_1         Airbag_2
##  Min.   : 3.000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.: 3.000   1st Qu.:1.0000   1st Qu.:1.0000   1st Qu.:0.0000
##  Median : 3.000   Median :1.0000   Median :1.0000   Median :1.0000
##  Mean   : 3.815   Mean   :0.8134   Mean   :0.9708   Mean   :0.7228
##  3rd Qu.: 3.000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :36.000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##
##      Airco        Automatic_airco   Boardcomputer      CD_Player
##  Min.   :0.0000   Min.   :0.00000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.0000
##  Median :1.0000   Median :0.00000   Median :0.0000   Median :0.0000
##  Mean   :0.5084   Mean   :0.05641   Mean   :0.2946   Mean   :0.2187
##  3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :1.00000   Max.   :1.0000   Max.   :1.0000
##
##   Central_Lock    Powered_Windows Power_Steering       Radio
##  Min.   :0.0000   Min.   :0.000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:1.0000   1st Qu.:0.0000
##  Median :1.0000   Median :1.000   Median :1.0000   Median :0.0000
##  Mean   :0.5801   Mean   :0.562   Mean   :0.9777   Mean   :0.1462
##  3rd Qu.:1.0000   3rd Qu.:1.000   3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :1.000   Max.   :1.0000   Max.   :1.0000
##
##    Mistlamps      Sport_Model     Backseat_Divider  Metallic_Rim
##  Min.   :0.000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:1.0000   1st Qu.:0.0000
##  Median :0.000   Median :0.0000   Median :1.0000   Median :0.0000
##  Mean   :0.257   Mean   :0.3001   Mean   :0.7702   Mean   :0.2047
##  3rd Qu.:1.000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :1.000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##
##  Radio_cassette
##  Min.   :0.0000
##  1st Qu.:0.0000
##  Median :0.0000
##  Mean   :0.1455
##  3rd Qu.:0.0000
##  Max.   :1.0000
##
##             Tow_Bar
##  \U0001f6be \U0001f192 \U0001f193 \U0001f195 \U0001f196 \U0001f197 \U0001f199 \U0001f3e7:   1
##  0                                                                                      :1036
##  1                                                                                      : 399
##
##
##
##
```

The data set has null values which have been omitted using na.omit() and the columns-'Price' and 'Tow_Bar' had special characters apart from digits for which, the specific records with such special character values are not considered. Apart from that, I had to convert the class of the 'Price' column back to numeric for further data analysis using the 'lapply' function and removed the extreme value(outlier) in the following manner

```
nrow(Toyota)
```

```
## [1] 1436
```

```
Toyota <- na.omit(Toyota)
Toyota <- subset(Toyota, grepl('[0-9]+', Toyota$Price))
Toyota <- subset(Toyota, grepl('[0-9]', Toyota$Tow_Bar))
Toyota[,c("Price")]<-lapply(c("Price"), function(fn)
  as.numeric(as.character(Toyota[,fn])))
Toyota <- subset(Toyota, Toyota$Price < 9999990)
nrow(Toyota)
```

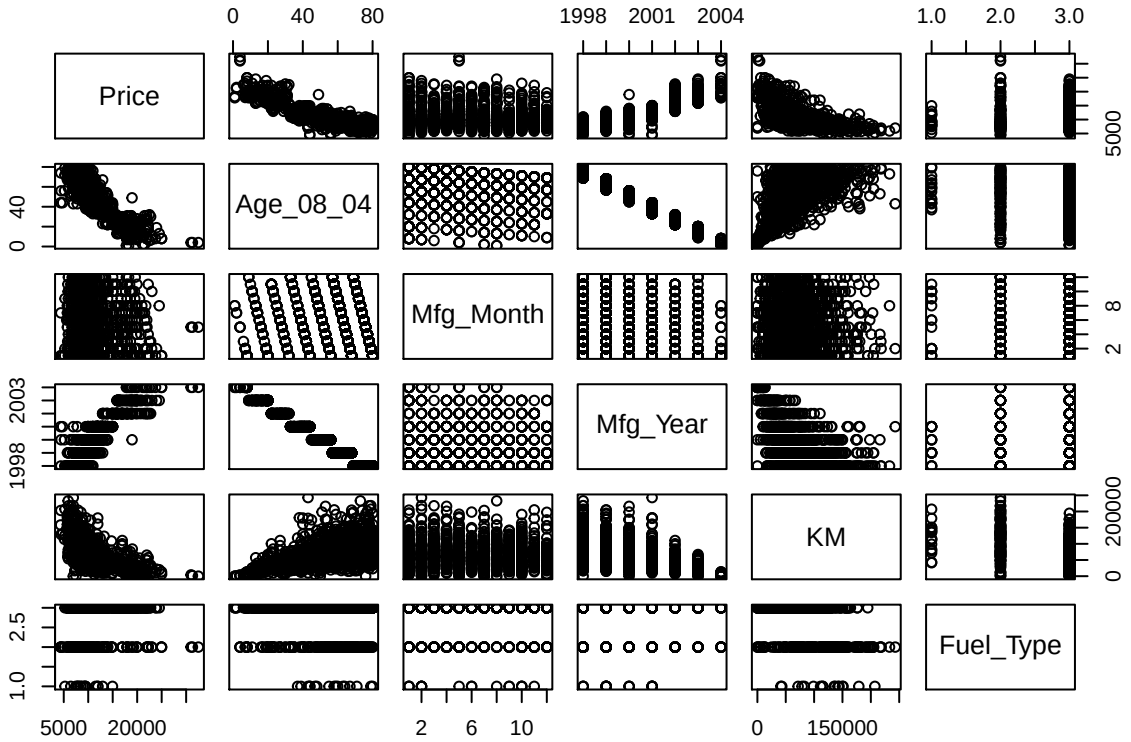
```
## [1] 1431
```

Four records were removed from the data set after data-cleaning.

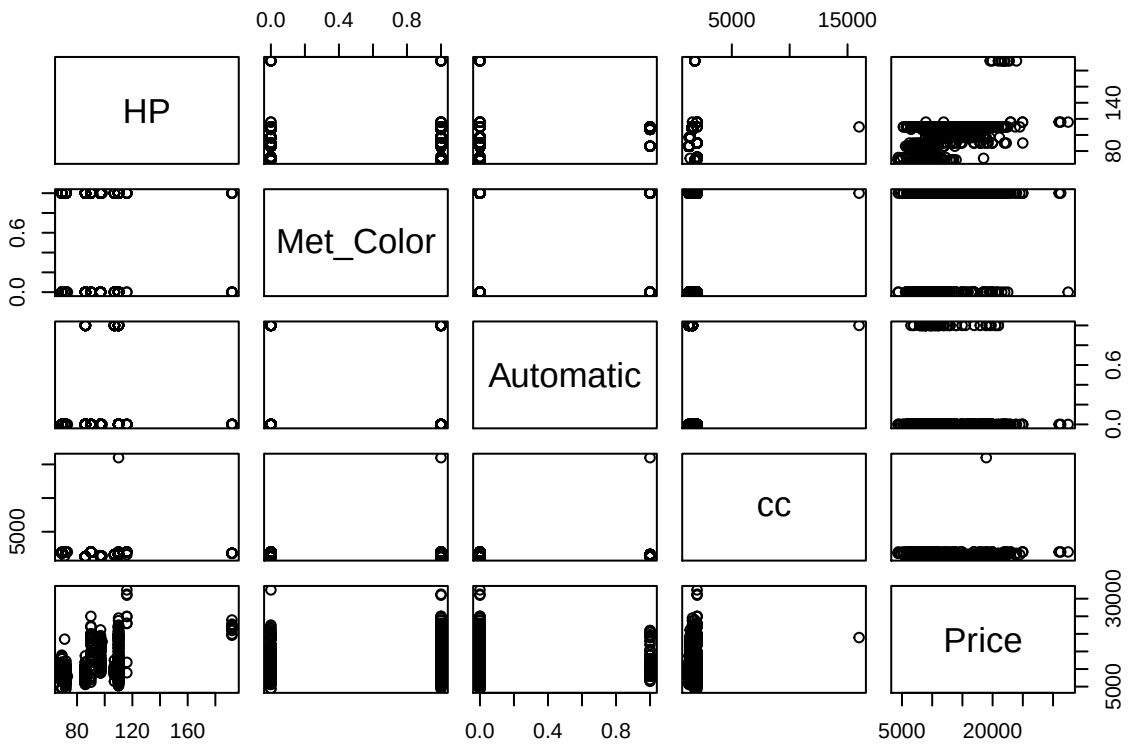
Data Exploration, Data Visualization and Feature Selection:

Pairs and correlation matrix for the quantitative variables od the data set:

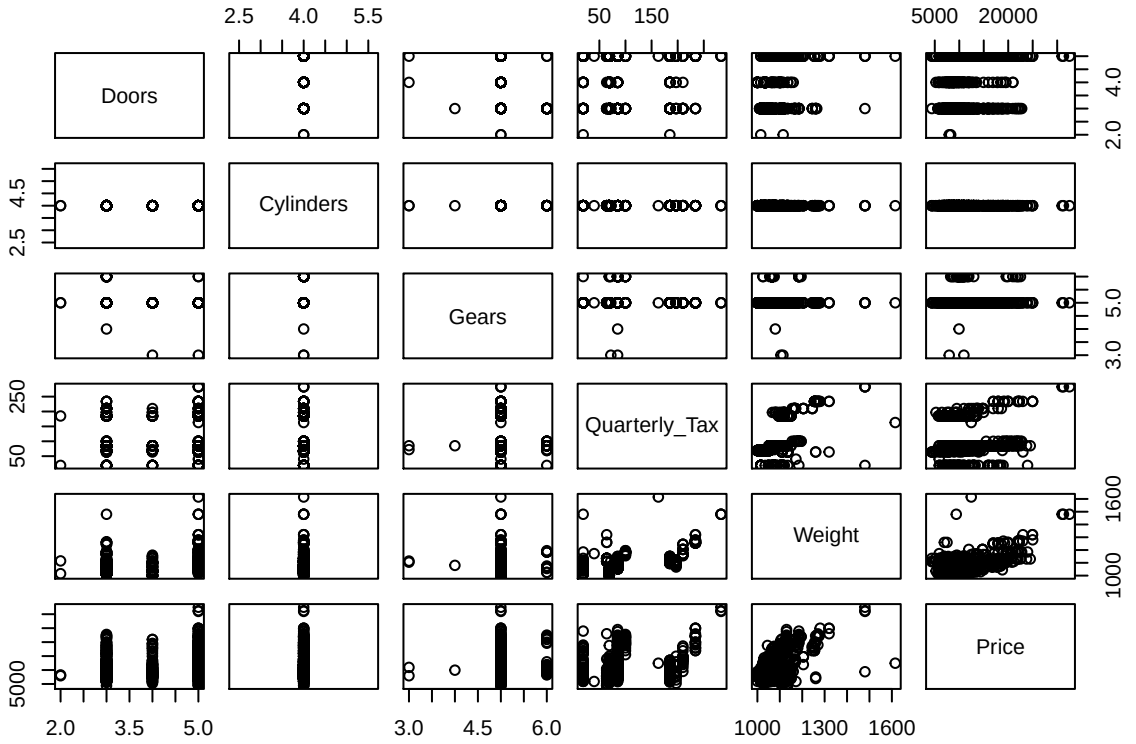
```
pairs(Toyota[3:8])
```



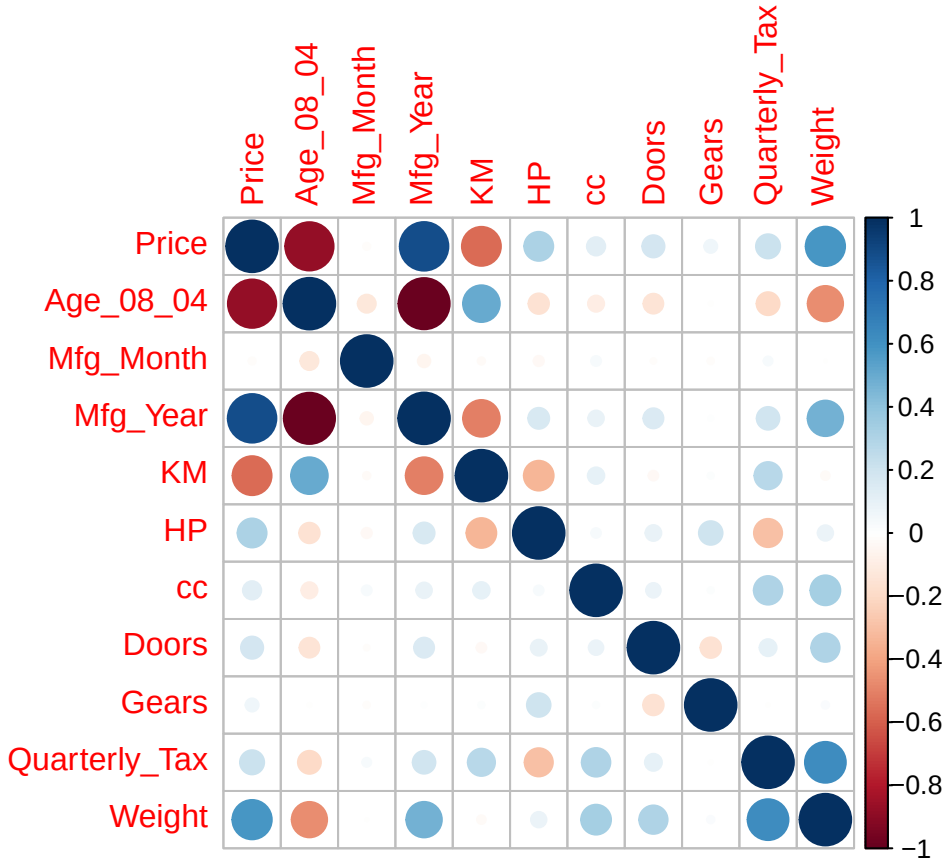
```
pairs(Toyota[c(9:12,3)])
```




```
pairs(Toyota[c(13:17,3)])
```




```
Toyota.cor=cor(Toyota[c(3:7,9,12,13,15:17)])
corrplot(Toyota.cor)
```

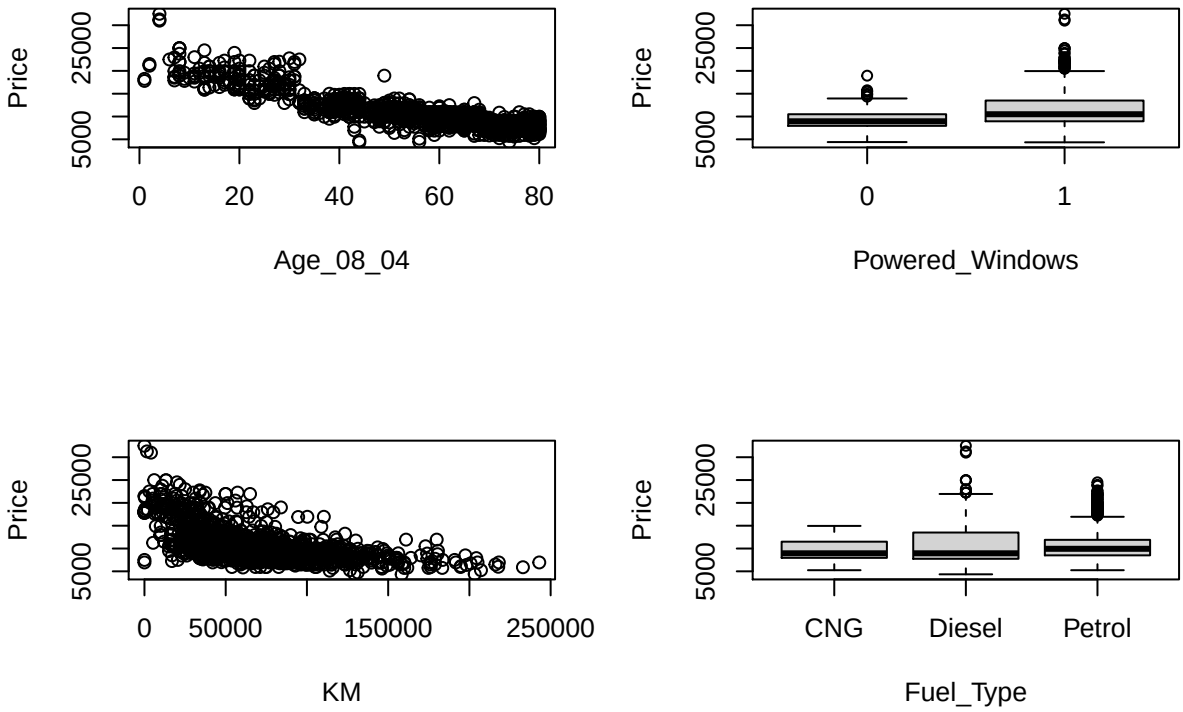


Hence we can infer from the above graphs and matrix that Age_08_04, Mfg_Year, KM, HP, and Weight have a fair significance with respect to the Price variable as their values in the correlation matrix are near to -1 or 1.

Further, I've generated some box plots and scatter plots to explore the data:

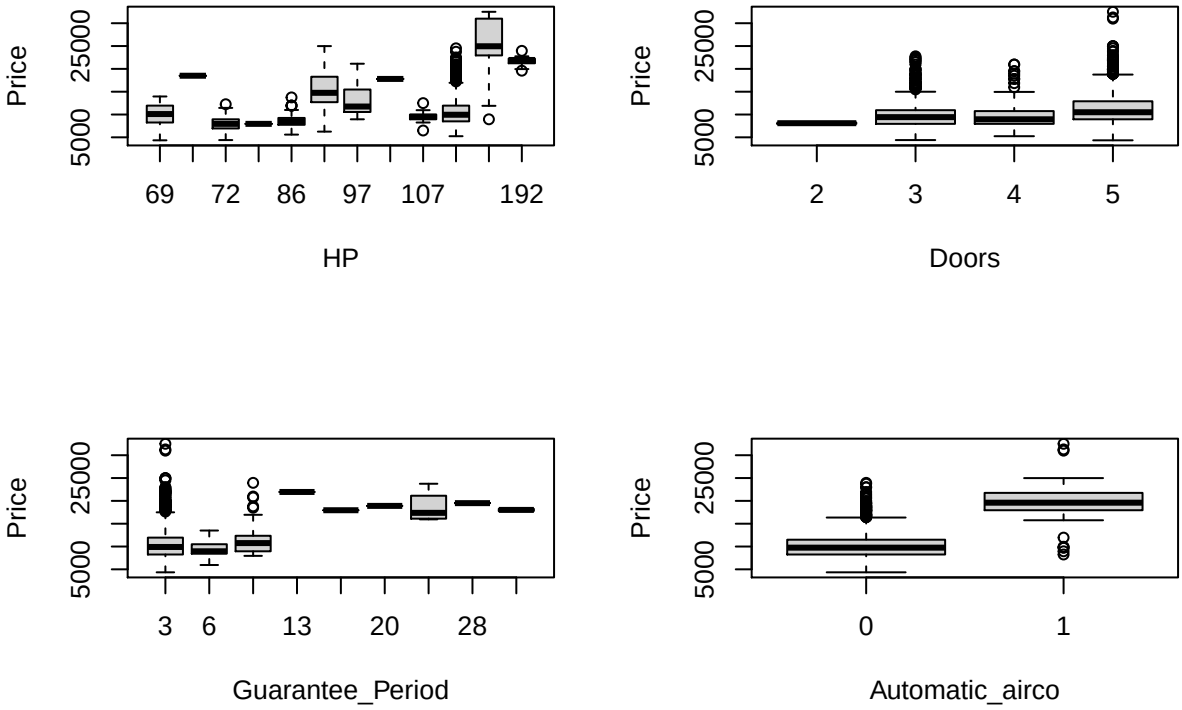
```
par(mfrow=c(2,2))
plot(Toyota$Price~Toyota$Age_08_04,
        xlab="Age_08_04", ylab="Price"

)
boxplot(Toyota$Price~Toyota$Powered_Windows,
        xlab="Powered_Windows", ylab="Price"
)
plot(Toyota$Price~Toyota$KM,
     xlab="KM", ylab="Price"
)
boxplot(Toyota$Price~Toyota$Fuel_Type,
        xlab="Fuel_Type", ylab="Price"
)
```



From the above plots, it can be said that all the above variables have a relationship with the 'Price' variable due to their pattern in plots and inter-quantile ranges in the box plots. Similarly, I've further analyzed more variables with respect to the Price using boxplots as below:

```
par(mfrow=c(2,2))
boxplot(Toyota$Price~Toyota$HP,
        xlab="HP", ylab="Price"
)
boxplot(Toyota$Price~Toyota$Doors,
        xlab="Doors", ylab="Price"
)
boxplot(Toyota$Price~Toyota$Guarantee_Period,
        xlab="Guarantee_Period", ylab="Price"
)
boxplot(Toyota$Price~Toyota$Automatic_airco,
        xlab="Automatic_airco", ylab="Price"
)
```



In order to select the best features for our models, I've performed Forward Step Wise Selection on all the variables except 'Model' and 'Id' as these are qualitative variables and have a large number of distinctive records.

```
attach(Toyota)
step.model<-regsubsets(Price ~ .-Model-Id, data = Toyota,
                       really.big = TRUE, method="forward")
```

```
## Reordering variables and trying again:
```

```
summary(step.model)
```

```
## Subset selection object
## Call: regsubsets.formula(Price ~ . - Model - Id, data = Toyota, really.big = TRUE,
##     method = "forward")
## 36 Variables  (and intercept)
##                   Forced in Forced out
## Age_08_04             FALSE      FALSE
## Mfg_Month             FALSE      FALSE
## KM                    FALSE      FALSE
## Fuel_TypeDiesel       FALSE      FALSE
## Fuel_TypePetrol       FALSE      FALSE
## HP                    FALSE      FALSE
## Met_Color             FALSE      FALSE
## Automatic             FALSE      FALSE
```

```
## cc                  FALSE      FALSE
## Doors               FALSE      FALSE
## Gears               FALSE      FALSE
## Quarterly_Tax       FALSE      FALSE
## Weight              FALSE      FALSE
## Mfr_Guarantee       FALSE      FALSE
## BOVAG_Guarantee     FALSE      FALSE
## Guarantee_Period    FALSE      FALSE
## ABS                 FALSE      FALSE
## Airbag_1            FALSE      FALSE
## Airbag_2            FALSE      FALSE
## Airco               FALSE      FALSE
## Automatic_airco     FALSE      FALSE
## Boardcomputer       FALSE      FALSE
## CD_Player           FALSE      FALSE
## Central_Lock        FALSE      FALSE
## Powered_Windows     FALSE      FALSE
## Power_Steering      FALSE      FALSE
## Radio               FALSE      FALSE
## Mistlamps           FALSE      FALSE
## Sport_Model         FALSE      FALSE
## Backseat_Divider    FALSE      FALSE
## Metallic_Rim        FALSE      FALSE
## Radio_cassette      FALSE      FALSE
## Tow_Bar0            FALSE      FALSE
## Mfg_Year            FALSE      FALSE
## Cylinders           FALSE      FALSE
## Tow_Bar1            FALSE      FALSE
## 1 subsets of each size up to 9
## Selection Algorithm: forward
##          Age_08_04 Mfg_Month Mfg_Year KM  Fuel_TypeDiesel Fuel_TypePetrol HP
## 1  ( 1 ) " "       " "       "*"      " " " "              " "              " "
## 2  ( 1 ) " "       " "       "*"      " " " "              " "              " "
## 3  ( 1 ) " "       " "       "*"      "*" " "              " "              " "
## 4  ( 1 ) " "       " "       "*"      "*" " "              " "              " "
## 5  ( 1 ) " "       " "       "*"      "*" " "              " "              "*"
## 6  ( 1 ) " "       " "       "*"      "*" " "              " "              "*"
## 7  ( 1 ) " "       " "       "*"      "*" " "              " "              "*"
## 8  ( 1 ) " "       " "       "*"      "*" " "              "*"              "*"
## 9  ( 1 ) " "       " "       "*"      "*" " "              "*"              "*"
##          Met_Color Automatic cc  Doors Cylinders Gears Quarterly_Tax Weight
## 1  ( 1 ) " "       " "       " " " "   " "       " "   " "           " "
## 2  ( 1 ) " "       " "       " " " "   " "       " "   " "           " "
## 3  ( 1 ) " "       " "       " " " "   " "       " "   " "           " "
## 4  ( 1 ) " "       " "       " " " "   " "       " "   " "           "*"
## 5  ( 1 ) " "       " "       " " " "   " "       " "   " "           "*"
## 6  ( 1 ) " "       " "       " " " "   " "       " "   " "           "*"
## 7  ( 1 ) " "       " "       " " " "   " "       " "   "*"           "*"
## 8  ( 1 ) " "       " "       " " " "   " "       " "   "*"           "*"
## 9  ( 1 ) " "       " "       " " " "   " "       " "   "*"           "*"
##          Mfr_Guarantee BOVAG_Guarantee Guarantee_Period ABS Airbag_1 Airbag_2
## 1  ( 1 ) " "           " "             " "              " " " " "      " "
## 2  ( 1 ) " "           " "             " "              " " " " "      " "
## 3  ( 1 ) " "           " "             " "              " " " " "      " "
```

```
## 4  ( 1 ) " "           " "             " "              " " " " "      " "
## 5  ( 1 ) " "           " "             " "              " " " " "      " "
## 6  ( 1 ) " "           " "             " "              " " " " "      " "
## 7  ( 1 ) " "           " "             " "              " " " " "      " "
## 8  ( 1 ) " "           " "             " "              " " " " "      " "
## 9  ( 1 ) " "           " "             "*"              " " " " "      " "
##          Airco Automatic_airco Boardcomputer CD_Player Central_Lock
## 1  ( 1 ) " "   " "              " "           " "       " "
## 2  ( 1 ) " "   "*"              " "           " "       " "
## 3  ( 1 ) " "   "*"              " "           " "       " "
## 4  ( 1 ) " "   "*"              " "           " "       " "
## 5  ( 1 ) " "   "*"              " "           " "       " "
## 6  ( 1 ) " "   "*"              " "           " "       " "
## 7  ( 1 ) " "   "*"              " "           " "       " "
## 8  ( 1 ) " "   "*"              " "           " "       " "
## 9  ( 1 ) " "   "*"              " "           " "       " "
##          Powered_Windows Power_Steering Radio Mistlamps Sport_Model
## 1  ( 1 ) " "             " "            " "   " "       " "
## 2  ( 1 ) " "             " "            " "   " "       " "
## 3  ( 1 ) " "             " "            " "   " "       " "
## 4  ( 1 ) " "             " "            " "   " "       " "
## 5  ( 1 ) " "             " "            " "   " "       " "
## 6  ( 1 ) "*"             " "            " "   " "       " "
## 7  ( 1 ) "*"             " "            " "   " "       " "
## 8  ( 1 ) "*"             " "            " "   " "       " "
## 9  ( 1 ) "*"             " "            " "   " "       " "
##          Backseat_Divider Metallic_Rim Radio_cassette Tow_Bar0 Tow_Bar1
## 1  ( 1 ) " "               " "          " "            " "      " "
## 2  ( 1 ) " "               " "          " "            " "      " "
## 3  ( 1 ) " "               " "          " "            " "      " "
## 4  ( 1 ) " "               " "          " "            " "      " "
## 5  ( 1 ) " "               " "          " "            " "      " "
## 6  ( 1 ) " "               " "          " "            " "      " "
## 7  ( 1 ) " "               " "          " "            " "      " "
## 8  ( 1 ) " "               " "          " "            " "      " "
## 9  ( 1 ) " "               " "          " "            " "      " "
```

This method gives us the variables which are significant for our target variables which can be recognized by the '*' in the respective variable columns. Thus I have further considered the following variables: Mfg_Year, Fuel_Type, HP, KM, Weight, Guarantee_Period, Quarterly_Tax, Powered_Windows and Automatic_airco

Regression Algorithms:

Since the problem is to predict the price for a used Toyota Corolla, it is a regression problem. I've implemented the following regression algorithms on the given dataset for the problem:

- Linear Regression
- Lasso Regression
- Ridge Regression.

Splitting the data set into train and test:

```r
set.seed(1)
Toyota.sample = sample.split(Toyota,SplitRatio = 0.70)
train =subset(Toyota,Toyota.sample ==TRUE)
test=subset(Toyota, Toyota.sample==FALSE)
x <- model.matrix(Price~Mfg_Year+KM+Fuel_Type+HP+Quarterly_Tax+
                    Weight+Guarantee_Period+Automatic_airco+
                    Powered_Windows)
y <- Price
x_train <- model.matrix(train$Price~train$Mfg_Year+train$KM+
                          train$Fuel_Type+train$HP+train$Quarterly_Tax+
                          train$Weight+train$Guarantee_Period+
                          train$Automatic_airco+train$Powered_Windows)[,-1]
y_train <- train$Price
x_test <-  model.matrix(test$Price~test$Mfg_Year+test$KM+test$Fuel_Type+test$HP+
                          test$Quarterly_Tax+test$Weight+test$Guarantee_Period+
                          test$Automatic_airco+test$Powered_Windows)[,-1]
y_test <- test$Price
```

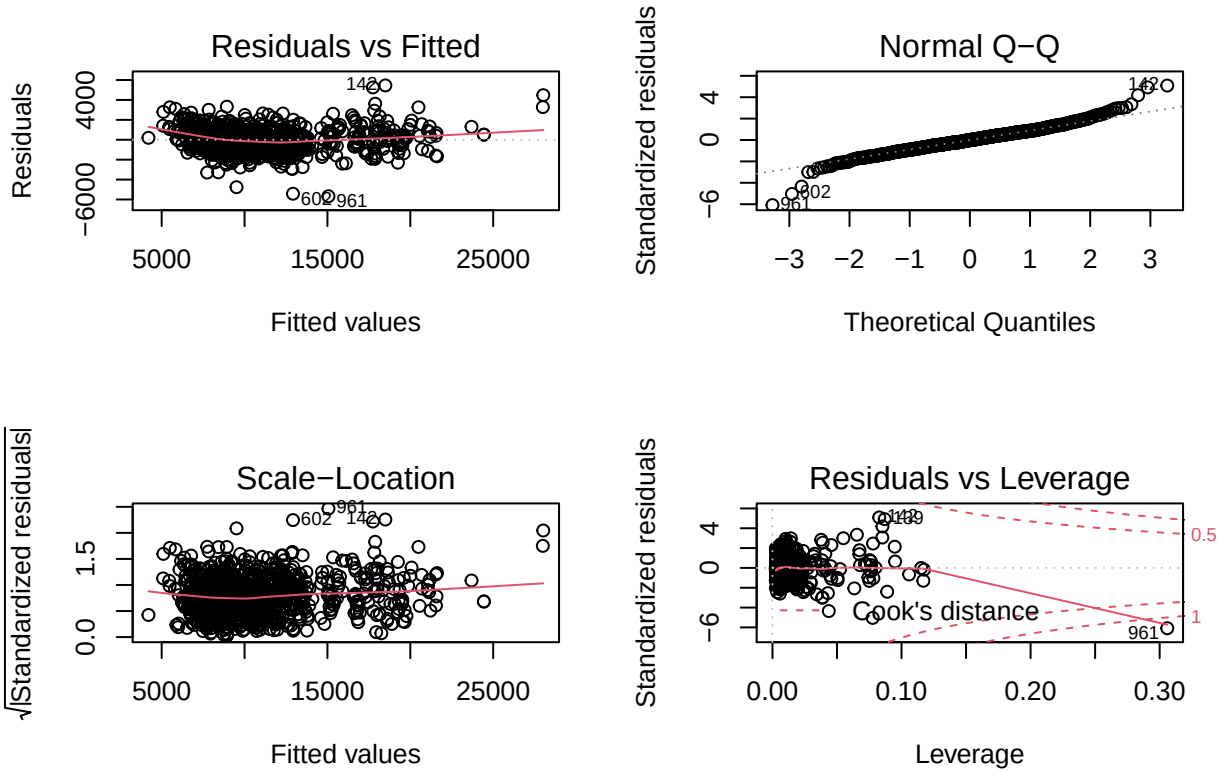
Linear Regression:

```r
lm.model <- lm(Price~Mfg_Year+KM+Fuel_Type+HP+Quarterly_Tax+
               Weight+Guarantee_Period+Automatic_airco+Powered_Windows,
             data=train)
summary(lm.model)
```

```
##
## Call:
## lm(formula = Price ~ Mfg_Year + KM + Fuel_Type + HP + Quarterly_Tax +
##     Weight + Guarantee_Period + Automatic_airco + Powered_Windows,
##     data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5671.2  -673.4    -7.3   662.4  5456.2
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -2.718e+06  6.710e+04 -40.510  < 2e-16 ***
## Mfg_Year          1.354e+03  3.368e+01  40.215  < 2e-16 ***
## KM               -1.689e-02  1.356e-03 -12.460  < 2e-16 ***
## Fuel_TypeDiesel   2.350e+02  3.618e+02   0.650    0.516
## Fuel_TypePetrol   2.416e+03  3.483e+02   6.936 7.42e-12 ***
## HP                9.313e+00  3.699e+00   2.517    0.012 *
## Quarterly_Tax     1.743e+01  1.697e+00  10.270  < 2e-16 ***
## Weight            1.566e+01  1.315e+00  11.907  < 2e-16 ***
## Guarantee_Period  7.818e+01  1.245e+01   6.279 5.16e-10 ***
## Automatic_airco   2.178e+03  1.858e+02  11.726  < 2e-16 ***
## Powered_Windows   4.517e+02  7.914e+01   5.707 1.53e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 1119 on 955 degrees of freedom
## Multiple R-squared:  0.9064, Adjusted R-squared:  0.9054
## F-statistic: 924.3 on 10 and 955 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(lm.model)
```



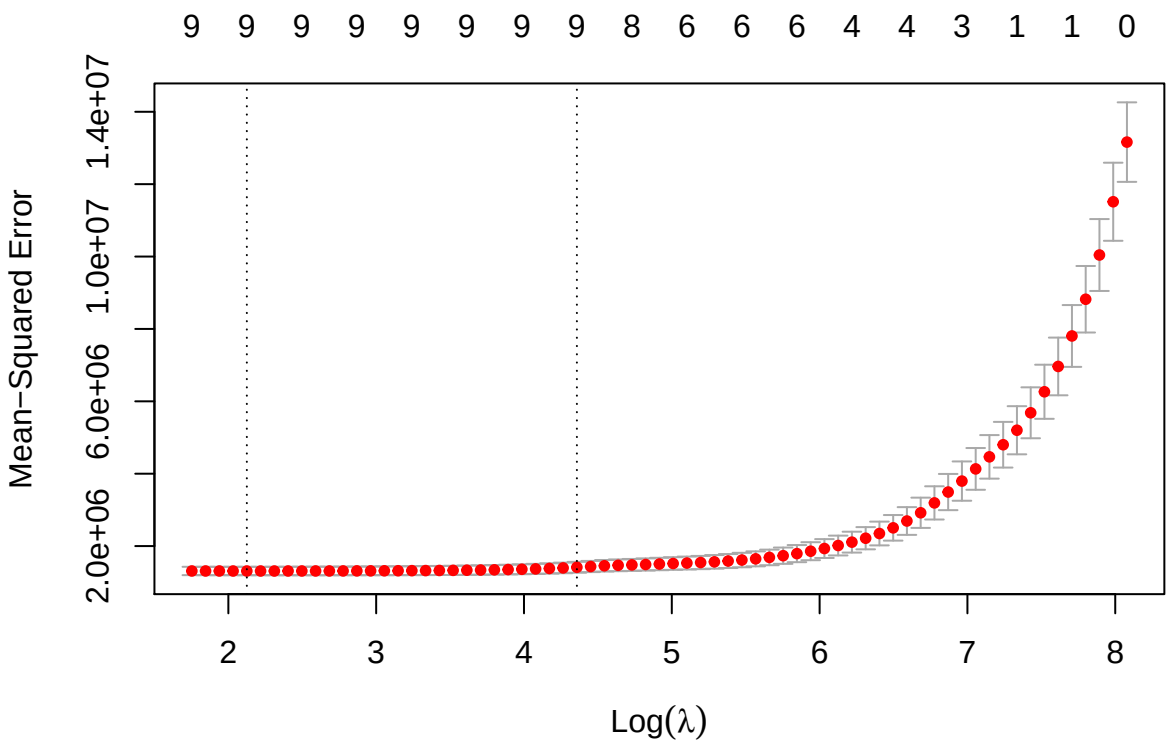

```
test_pred <- predict(lm.model,data = test)
rmse(test$Price,test_pred)
```

```
## [1] 4406.904
```

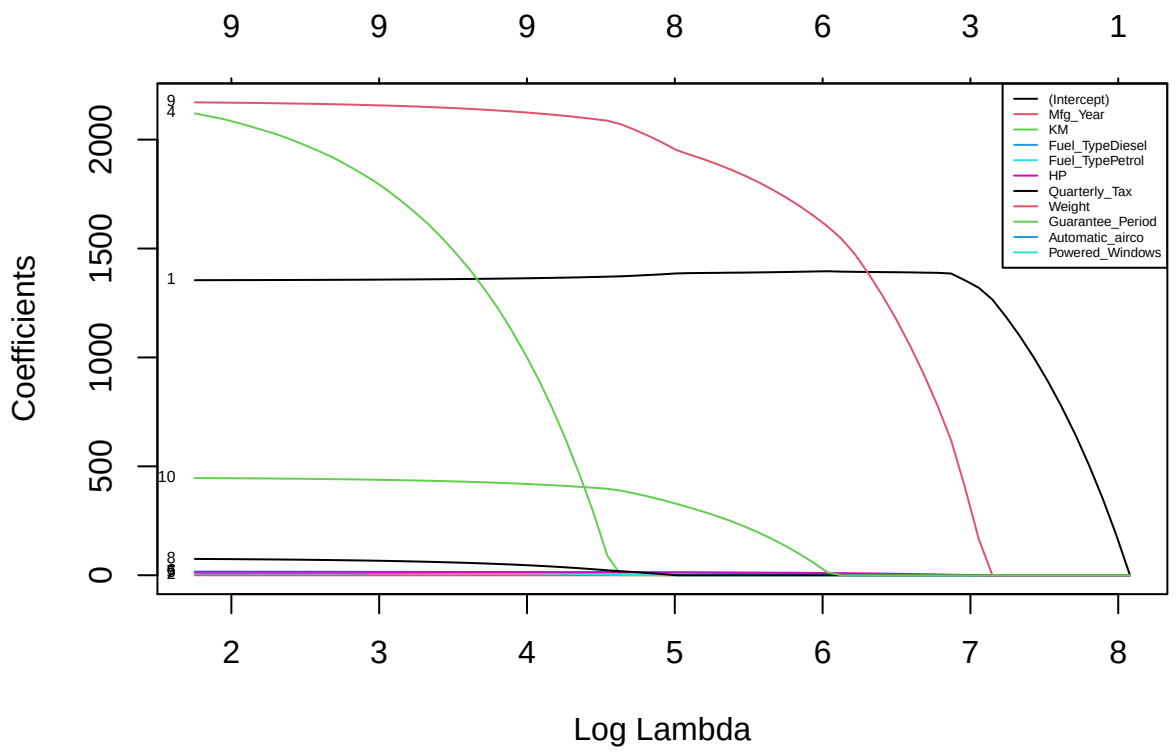
For the residual vs fitted graph, we can say that the error terms are showing sufficient pattern and is a bit non-linear. We have probably left out something in the model. Data point 961 in 'Residual vs Leverage' plot has high leverage with a small residual magnitude. Since the Test RMSE is greater than Train RMSE I have further opted for Lasso Regression and Ridge Regression.

Lasso Regression:

```
lasso.mod <- glmnet(x_train, y_train, alpha = 1, thresh = 1e-12)
cv.out <- cv.glmnet(x_train, y_train, alpha = 1)
plot(cv.out)
```




```
plot(cv.out$glmnet.fit, xvar = "lambda", label = TRUE)
legend("topright", lwd = 1, col = 1:6, legend = colnames(x),
       cex = 0.4)
```




```
bestlam <- cv.out$lambda.min
bestlam
```

```
## [1] 8.374123
```

```
lasso.pred <- predict(lasso.mod, s = bestlam, newx = x_test)
sst <- sum((y_test-mean(y_test))^2)
sse <-sum((lasso.pred-y_test)^2)
rsq <- 1-(sse/sst)
rsq
```

```
## [1] 0.884734
```

```
rmse(y_test,lasso.pred)
```

```
## [1] 1228.7
```

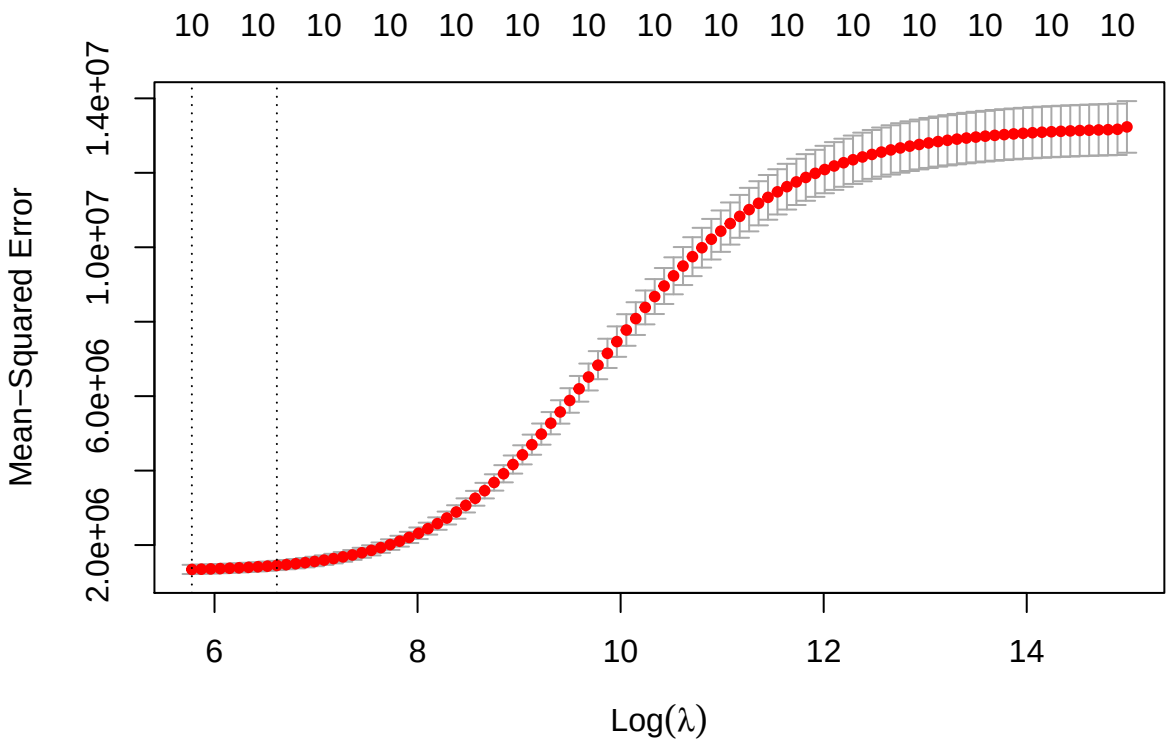
```
regr.eval(trues = y_test, preds = lasso.pred)
```

```
##          mae          mse         rmse         mape
## 8.461827e+02 1.509703e+06 1.228700e+03 8.315221e-02
```

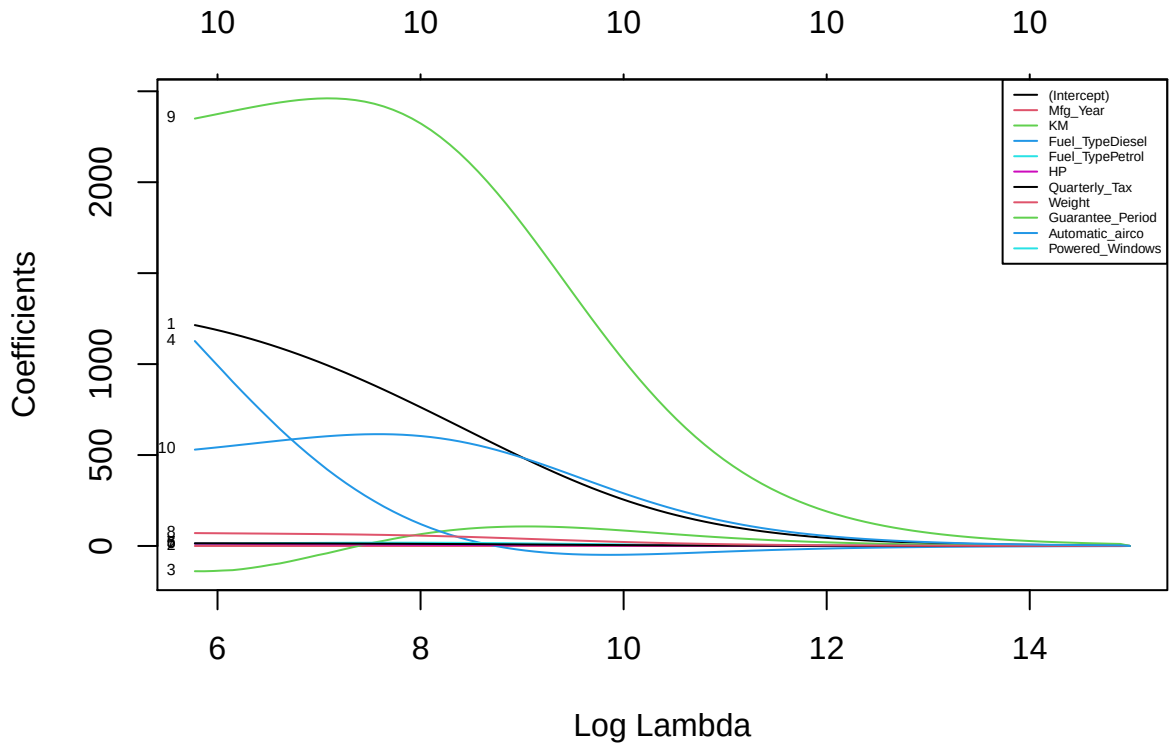
The Lasso regression greatly lowers the RMSE value as compared to Linear Regression of Test variable.

Ridge Regression:

```
ridge.mod <- glmnet(x_train, y_train, alpha = 0, thresh = 1e-12)
cv.out <- cv.glmnet(x_train, y_train, alpha = 0)
plot(cv.out)
```




```
plot(cv.out$glmnet.fit, xvar = "lambda", label = TRUE)
legend("topright", lwd = 1, col = 1:6, legend = colnames(x),
       cex = 0.4)
```




```
bestlam <- cv.out$lambda.min
bestlam
```

```
## [1] 322.6992
```

```
ridge.pred <- predict(ridge.mod, s = bestlam, newx = x_test)
sst <- sum((y_test-mean(y_test))^2)
sse <- sum((ridge.pred-y_test)^2)
rsq <- 1-(sse/sst)
rsq
```

```
## [1] 0.8860436
```

```
rmse(test$Price,ridge.pred)
```

```
## [1] 1221.7
```

```
regr.eval(trues = y_test, preds = ridge.pred)
```

```
##          mae          mse         rmse         mape
## 8.476521e+02 1.492550e+06 1.221700e+03 8.301377e-02
```

Among all the above regression algorithms Ridge regression has the least RMSE value.

Performance of Algorithms:

RMSE Values for implemented algorithms:

- Linear Regression: 4406.9
- Lasso Regression: 1228.7
- Ridge Regression: 1221.7

Since Linear Regression has the worst performance I've further compared Lasso and Regression to find the best algorithm for our solution.

R-Square values:

- Lasso Regression: 0.884
- Ridge Regression: 0.886

The higher value for R square indicates slightly greater accuracy in the Ridge Regression.

Conclusion

Thus considering the performances of the algorithms we can conclude that Ridge Regression is the best performing algorithm for our problem and the algorithms can be ranked as :

- Ridge Regression > Lasso Regression > Linear Regression.

Hence, we can use the generated Ridge Regression model in order to predict the price of a used Toyota Corolla more accurately as compared to other models generated above.