Final Exam

Sahil Shah

8/11/2020

Importing the Toyotta Corolla Dataset.

```
dataset <- read.csv("toyotaCorolla.csv")
```

Analyzing the dataset

```
dim(dataset)
```

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

```
head(dataset)
```

```
##   Id                                      Model Price Age_08_04 Mfg_Month
## 1  1  TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors 13500        23         10
## 2  2  TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors 13750        23         10
## 3  3 ?TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors 13950        24          9
## 4  4  TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors 14950        26          7
## 5  5    TOYOTA Corolla 2.0 D4D HATCHB SOL 2/3-Doors 13750        30          3
## 6  6    TOYOTA Corolla 2.0 D4D HATCHB SOL 2/3-Doors 12950        32          1
##   Mfg_Year    KM Fuel_Type HP Met_Color Automatic   cc Doors Cylinders Gears
## 1     2002 46986    Diesel 90         1         0 2000     3         4     5
## 2     2002 72937    Diesel 90         1         0 2000     3         4     5
## 3     2002 41711    Diesel 90         1         0 2000     3         4     5
## 4     2002 48000    Diesel 90         0         0 2000     3         4     5
## 5     2002 38500    Diesel 90         0         0 2000     3         4     5
## 6     2002 61000    Diesel 90         0         0 2000     3         4     5
##   Quarterly_Tax Weight Mfr_Guarantee BOVAG_Guarantee Guarantee_Period ABS
## 1           210   1165             0               1               3   1
## 2           210   1165             0               1               3   1
## 3           210   1165             1               1               3   1
## 4           210   1165             1               1               3   1
## 5           210   1170             1               1               3   1
## 6           210   1170             0               1               3   1
##   Airbag_1 Airbag_2 Airco Automatic_airco Boardcomputer CD_Player Central_Lock
## 1        1        1     0              0             1         0            1
## 2        1        1     1              0             1         1            1
## 3        1        1     0              0             1         0            0
```

```
## 4        1        1     0              0            1        0           0
## 5        1        1     1              0            1        0           1
## 6        1        1     1              0            1        0           1
##   Powered_Windows Power_Steering Radio Mistlamps Sport_Model Backseat_Divider
## 1               1              1     0         0           0                1
## 2               0              1     0         0           0                1
## 3               0              1     0         0           0                1
## 4               0              1     0         0           0                1
## 5               1              1     0         1           0                1
## 6               1              1     0         1           0                1
##   Metallic_Rim Radio_cassette Tow_Bar
## 1            0              0       0
## 2            0              0       0
## 3            0              0       0
## 4            0              0       0
## 5            0              0       0
## 6            0              0       0
```

```
str(dataset)
```

```
## 'data.frame':    1436 obs. of  37 variables:
##  $ Id              : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Model           : chr  "TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors" "TOYOTA Corolla 2.0 D4D HA
##  $ Price           : chr  "13500" "13750" "13950" "14950" ...
##  $ Age_08_04       : int  23 23 24 26 30 32 27 30 27 23 ...
##  $ Mfg_Month       : int  10 10 9 7 3 1 6 3 6 10 ...
##  $ Mfg_Year        : int  2002 2002 2002 2002 2002 2002 2002 2002 2002 2002 ...
##  $ KM              : int  46986 72937 41711 48000 38500 61000 94612 75889 19700 71138 ...
##  $ Fuel_Type       : chr  "Diesel" "Diesel" "Diesel" "Diesel" ...
##  $ HP              : int  90 90 90 90 90 90 90 90 192 69 ...
##  $ Met_Color       : int  1 1 1 0 0 0 1 1 0 0 ...
##  $ Automatic       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ cc              : int  2000 2000 2000 2000 2000 2000 2000 2000 1800 1900 ...
##  $ Doors           : int  3 3 3 3 3 3 3 3 3 3 ...
##  $ Cylinders       : int  4 4 4 4 4 4 4 4 4 4 ...
##  $ Gears           : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ Quarterly_Tax   : int  210 210 210 210 210 210 210 210 100 185 ...
##  $ Weight          : int  1165 1165 1165 1165 1170 1170 1245 1245 1185 1105 ...
##  $ Mfr_Guarantee   : int  0 0 1 1 1 0 0 1 0 0 ...
##  $ BOVAG_Guarantee : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Guarantee_Period: int  3 3 3 3 3 3 3 3 3 3 ...
##  $ ABS             : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Airbag_1        : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Airbag_2        : int  1 1 1 1 1 1 1 1 0 1 ...
##  $ Airco           : int  0 1 0 0 1 1 1 1 1 1 ...
##  $ Automatic_airco : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Boardcomputer   : int  1 1 1 1 1 1 1 1 0 1 ...
##  $ CD_Player       : int  0 1 0 0 0 0 0 1 0 0 ...
##  $ Central_Lock    : int  1 1 0 0 1 1 1 1 1 0 ...
##  $ Powered_Windows : int  1 0 0 0 1 1 1 1 1 0 ...
##  $ Power_Steering  : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Radio           : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ Mistlamps       : int  0 0 0 0 1 1 0 0 0 0 ...
##  $ Sport_Model     : int  0 0 0 0 0 0 1 0 0 0 ...
```

```
##  $ Backseat_Divider: int  1 1 1 1 1 1 1 1 0 1 ...
##  $ Metallic_Rim    : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ Radio_cassette  : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ Tow_Bar         : chr  "0" "0" "0" "0" ...
```

The dataset consist of 37 Variables and 1436 Observations. From the `str()` function we can analyze the variable types that can be as follows : integer, character.

Summary

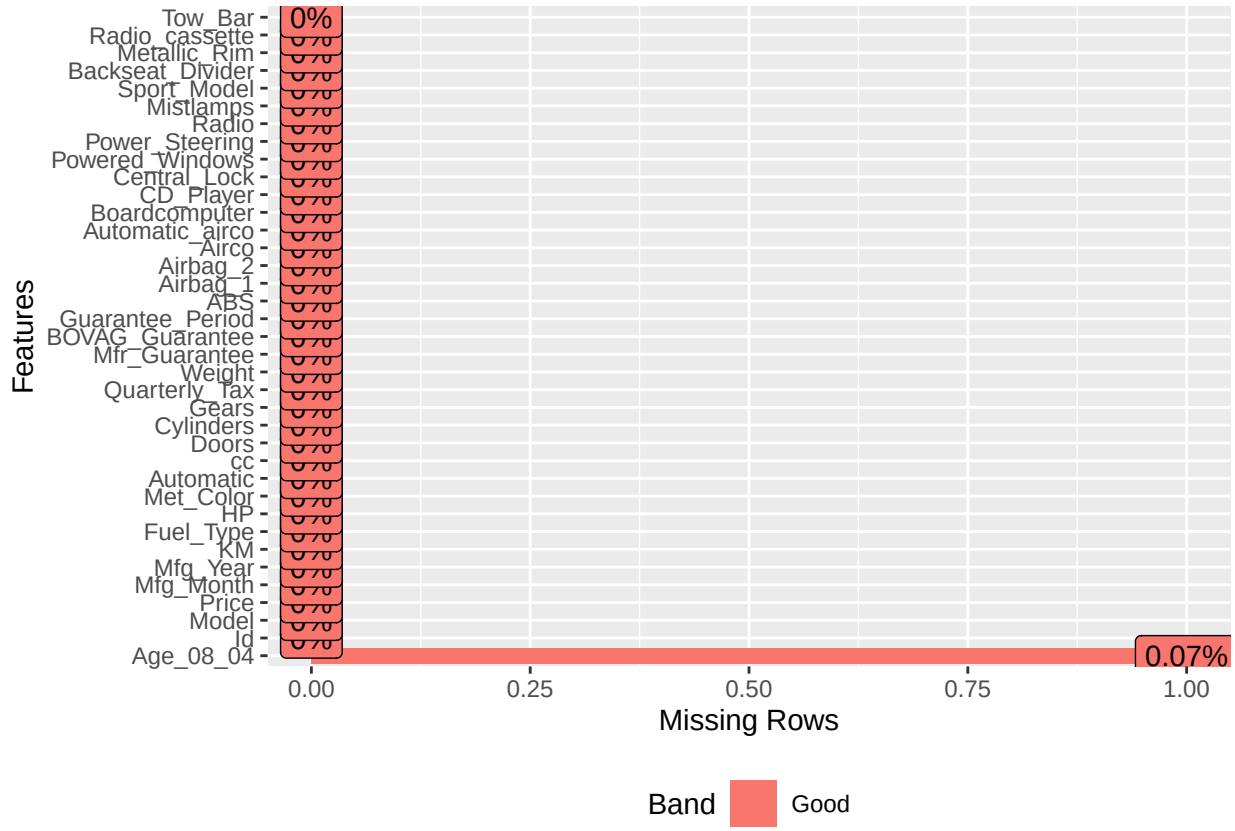
```
summary(dataset)
```

```
##        Id              Model               Price             Age_08_04
##  Min.   :   1.0   Length:1436        Length:1436        Min.   : 1.00
##  1st Qu.: 361.8   Class :character   Class :character   1st Qu.:44.00
##  Median : 721.5   Mode  :character   Mode  :character   Median :61.00
##  Mean   : 721.6                                        Mean   :55.94
##  3rd Qu.:1081.2                                        3rd Qu.:70.00
##  Max.   :1442.0                                        Max.   :80.00
##                                                        NA's   :1
##    Mfg_Month         Mfg_Year           KM          Fuel_Type
##  Min.   : 1.000   Min.   :1998   Min.   :     1   Length:1436
##  1st Qu.: 3.000   1st Qu.:1998   1st Qu.: 43000   Class :character
##  Median : 5.000   Median :1999   Median : 63390   Mode  :character
##  Mean   : 5.549   Mean   :2000   Mean   : 68533
##  3rd Qu.: 8.000   3rd Qu.:2001   3rd Qu.: 87021
##  Max.   :12.000   Max.   :2004   Max.   :243000
##
##        HP          Met_Color        Automatic             cc
##  Min.   : 69.0   Min.   :0.0000   Min.   :0.00000   Min.   : 1300
##  1st Qu.: 90.0   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.: 1400
##  Median :110.0   Median :1.0000   Median :0.00000   Median : 1600
##  Mean   :101.5   Mean   :0.6748   Mean   :0.05571   Mean   : 1577
##  3rd Qu.:110.0   3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.: 1600
##  Max.   :192.0   Max.   :1.0000   Max.   :1.00000   Max.   :16000
##
##      Doors         Cylinders     Gears       Quarterly_Tax        Weight
##  Min.   :2.000   Min.   :4   Min.   :3.000   Min.   : 19.00   Min.   :1000
##  1st Qu.:3.000   1st Qu.:4   1st Qu.:5.000   1st Qu.: 69.00   1st Qu.:1040
##  Median :4.000   Median :4   Median :5.000   Median : 85.00   Median :1070
##  Mean   :4.033   Mean   :4   Mean   :5.026   Mean   : 87.12   Mean   :1072
##  3rd Qu.:5.000   3rd Qu.:4   3rd Qu.:5.000   3rd Qu.: 85.00   3rd Qu.:1085
##  Max.   :5.000   Max.   :4   Max.   :6.000   Max.   :283.00   Max.   :1615
##
##  Mfr_Guarantee    BOVAG_Guarantee  Guarantee_Period      ABS
##  Min.   :0.0000   Min.   :0.0000   Min.   : 3.000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:1.0000   1st Qu.: 3.000   1st Qu.:1.0000
##  Median :0.0000   Median :1.0000   Median : 3.000   Median :1.0000
##  Mean   :0.4095   Mean   :0.8955   Mean   : 3.815   Mean   :0.8134
##  3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.: 3.000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :1.0000   Max.   :36.000   Max.   :1.0000
```

```
##
##     Airbag_1         Airbag_2          Airco        Automatic_airco
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.00000
##  1st Qu.:1.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000
##  Median :1.0000   Median :1.0000   Median :1.0000   Median :0.00000
##  Mean   :0.9708   Mean   :0.7228   Mean   :0.5084   Mean   :0.05641
##  3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0.00000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.00000
##
##  Boardcomputer      CD_Player       Central_Lock    Powered_Windows
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000
##  Median :0.0000   Median :0.0000   Median :1.0000   Median :1.000
##  Mean   :0.2946   Mean   :0.2187   Mean   :0.5801   Mean   :0.562
##  3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:1.000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.000
##
##  Power_Steering       Radio          Mistlamps      Sport_Model
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.000   Min.   :0.0000
##  1st Qu.:1.0000   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:0.0000
##  Median :1.0000   Median :0.0000   Median :0.000   Median :0.0000
##  Mean   :0.9777   Mean   :0.1462   Mean   :0.257   Mean   :0.3001
##  3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:1.000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.000   Max.   :1.0000
##
##  Backseat_Divider  Metallic_Rim    Radio_cassette     Tow_Bar
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Length:1436
##  1st Qu.:1.0000   1st Qu.:0.0000   1st Qu.:0.0000   Class :character
##  Median :1.0000   Median :0.0000   Median :0.0000   Mode  :character
##  Mean   :0.7702   Mean   :0.2047   Mean   :0.1455
##  3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##
```

The summary of the dataset defines the statistical values including mean, median and Max. Here the Price variable is treated with Character datatype, which should be in the numeric form.

```
# install.packages('DataExplorer')
library('DataExplorer')
plot_missing(dataset)
```




```
sapply(dataset, function(x) sum(is.na(x)))
```

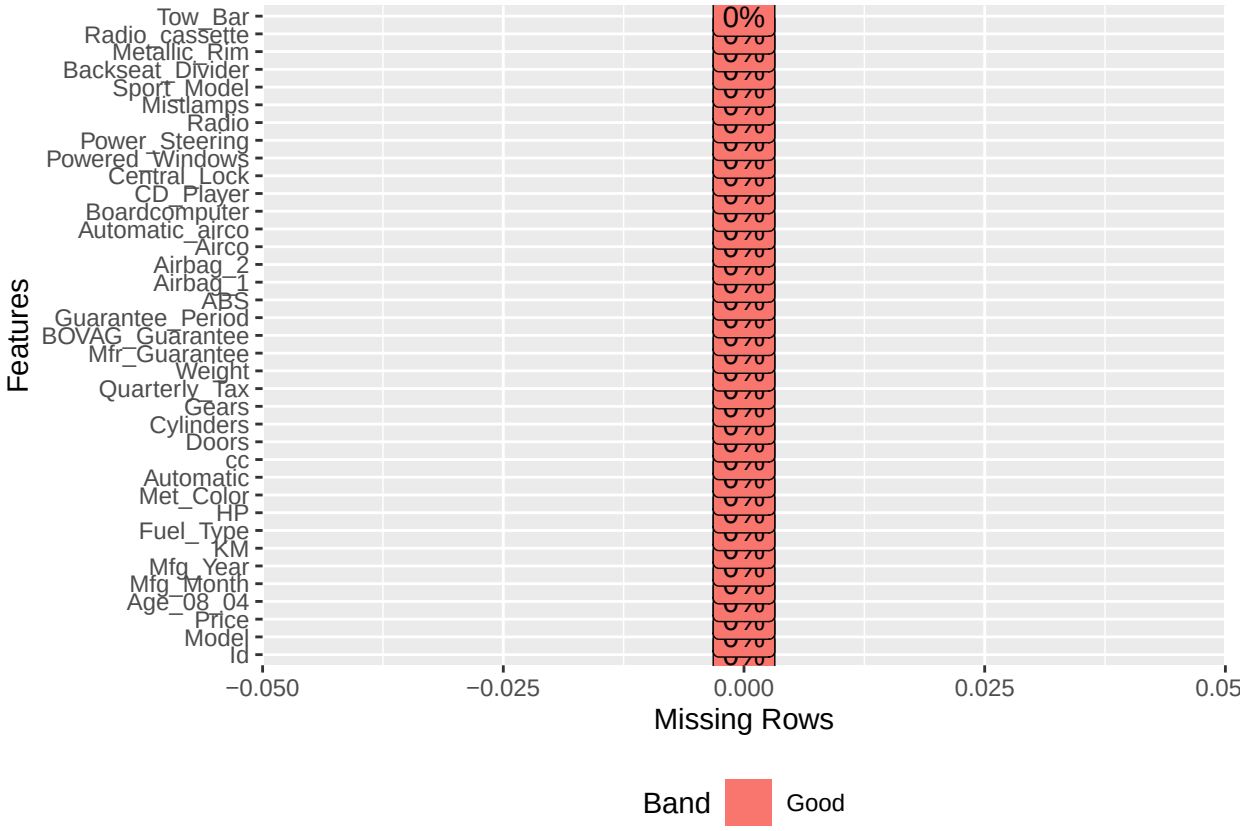
```
##               Id            Model            Price        Age_08_04
##                0                0                0                1
##        Mfg_Month         Mfg_Year               KM        Fuel_Type
##                0                0                0                0
##               HP        Met_Color        Automatic               cc
##                0                0                0                0
##            Doors        Cylinders            Gears    Quarterly_Tax
##                0                0                0                0
##           Weight    Mfr_Guarantee  BOVAG_Guarantee Guarantee_Period
##                0                0                0                0
##              ABS         Airbag_1         Airbag_2            Airco
##                0                0                0                0
##  Automatic_airco    Boardcomputer        CD_Player     Central_Lock
##                0                0                0                0
##  Powered_Windows   Power_Steering            Radio        Mistlamps
##                0                0                0                0
##      Sport_Model Backseat_Divider     Metallic_Rim   Radio_cassette
##                0                0                0                0
##          Tow_Bar
##                0
```

```
which(is.na(dataset$Age_08_04))
```

```
## [1] 798
```

As from the plot and other function we can see there is one null values present in **Age_08_04** the 798th row is null, we will add mean to the data.

```
dataset$Age_08_04[is.na(dataset$Age_08_04)] <- mean(dataset$Age_08_04,na.rm = TRUE)
plot_missing(dataset)
```




```
dataset$Price <- as.numeric(dataset$Price)
```

```
## Warning: NAs introduced by coercion
```

```
mean(dataset$Price,na.rm=TRUE)
```

```
## [1] 17705.21
```

```
dataset$Price[is.na(dataset$Price)] <- mean(dataset$Price,na.rm = TRUE)
```

After removing the null or dirty values in the Price column there is still Looking up the missing values and plotting it. As Price is the dependent Variable, the cleaning of the data involves the removal of outliers.

```
summary(dataset$Price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    4350    8450    9900   17705   11950 9999995
```

```
uv <- 3*quantile(dataset$Price, 0.99)
dataset$Price[dataset$Price>uv] <- uv
summary(dataset$Price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    4350    8450    9900   10788   11950   67238
```

In the above chunk there were some obvious errors, so I used the capping method for Prices, which helps the removal of the outliers. After the cleaning of the column the values changes to max as 67238, and mean and median comes closer.

Now there is no missing values and no dirty values in the dataset.

Removal of Columns

```
dataset_toyota <- dataset[c("Price", "Age_08_04", "Mfg_Month",
                            "Mfg_Year", "KM", "Fuel_Type", "HP",
                            "Automatic", "Doors", "Gears", "Quarterly_Tax")]
```

We are only selecting the Price, Age_08_04, Mfg_Month, Mfg_Year, KM, Fuel_Type, HP, Automatic, Doors, Gears and Quarterly_Tax.

```
attach(dataset_toyota)
summary(dataset_toyota)
```

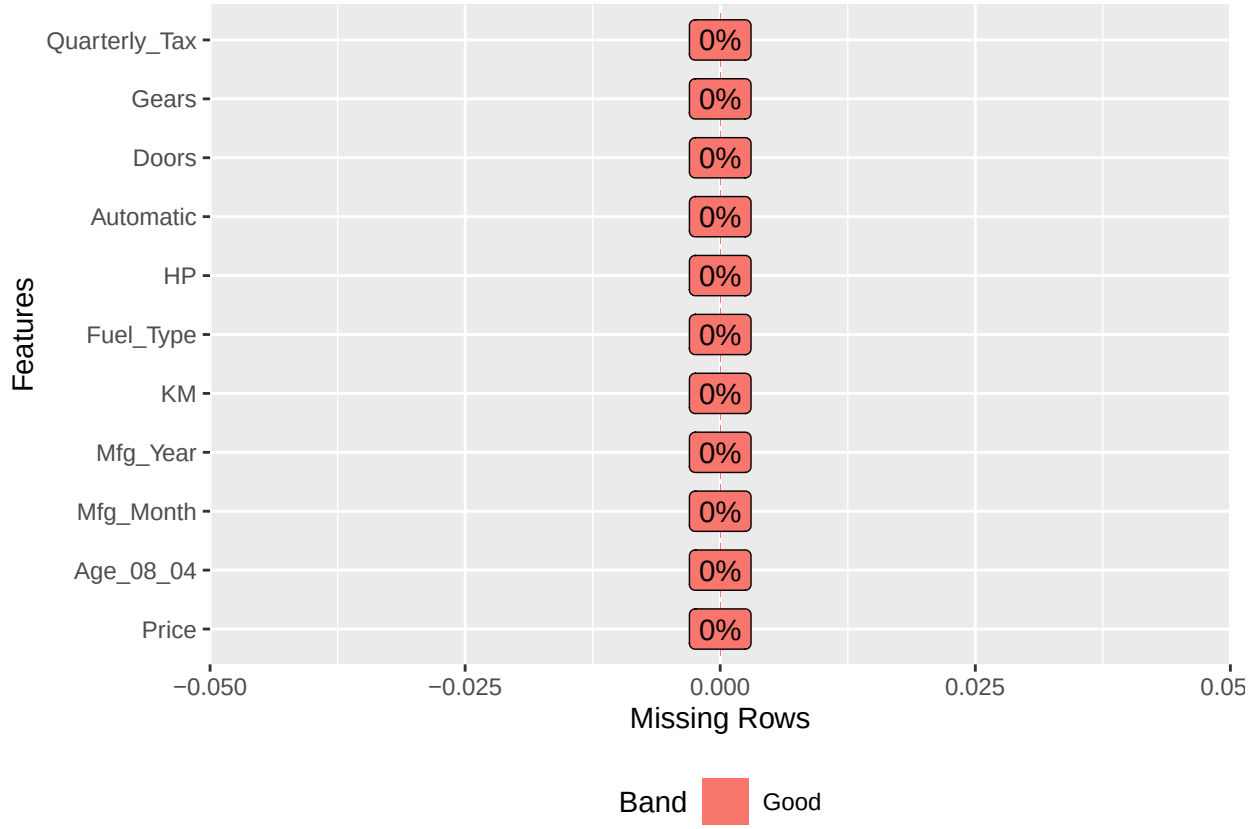
```
##      Price         Age_08_04       Mfg_Month         Mfg_Year
##  Min.   : 4350   Min.   : 1.00   Min.   : 1.000   Min.   :1998
##  1st Qu.: 8450   1st Qu.:44.00   1st Qu.: 3.000   1st Qu.:1998
##  Median : 9900   Median :61.00   Median : 5.000   Median :1999
##  Mean   :10788   Mean   :55.94   Mean   : 5.549   Mean   :2000
##  3rd Qu.:11950   3rd Qu.:70.00   3rd Qu.: 8.000   3rd Qu.:2001
##  Max.   :67238   Max.   :80.00   Max.   :12.000   Max.   :2004
##        KM          Fuel_Type              HP          Automatic
##  Min.   :     1   Length:1436        Min.   : 69.0   Min.   :0.00000
##  1st Qu.: 43000   Class :character   1st Qu.: 90.0   1st Qu.:0.00000
##  Median : 63390   Mode  :character   Median :110.0   Median :0.00000
##  Mean   : 68533                      Mean   :101.5   Mean   :0.05571
##  3rd Qu.: 87021                      3rd Qu.:110.0   3rd Qu.:0.00000
##  Max.   :243000                      Max.   :192.0   Max.   :1.00000
##      Doors           Gears       Quarterly_Tax
##  Min.   :2.000   Min.   :3.000   Min.   : 19.00
##  1st Qu.:3.000   1st Qu.:5.000   1st Qu.: 69.00
##  Median :4.000   Median :5.000   Median : 85.00
##  Mean   :4.033   Mean   :5.026   Mean   : 87.12
##  3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.: 85.00
##  Max.   :5.000   Max.   :6.000   Max.   :283.00
```

```
str(dataset_toyota)
```

```
## 'data.frame':    1436 obs. of  11 variables:
##  $ Price        : num  13500 13750 13950 14950 13750 ...
##  $ Age_08_04    : num  23 23 24 26 30 32 27 30 27 23 ...
##  $ Mfg_Month    : int  10 10 9 7 3 1 6 3 6 10 ...
##  $ Mfg_Year     : int  2002 2002 2002 2002 2002 2002 2002 2002 2002 2002 ...
##  $ KM           : int  46986 72937 41711 48000 38500 61000 94612 75889 19700 71138 ...
##  $ Fuel_Type    : chr  "Diesel" "Diesel" "Diesel" "Diesel" ...
##  $ HP           : int  90 90 90 90 90 90 90 90 192 69 ...
##  $ Automatic    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Doors        : int  3 3 3 3 3 3 3 3 3 3 ...
##  $ Gears        : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ Quarterly_Tax: int  210 210 210 210 210 210 210 210 100 185 ...
```

conversion of integer data into numeric datatype.

```
dataset_toyota$Mfg_Month <- as.numeric(dataset_toyota$Mfg_Month)
dataset_toyota$Mfg_Year <- as.numeric(dataset_toyota$Mfg_Year)
dataset_toyota$KM <- as.numeric(dataset_toyota$KM)
dataset_toyota$HP <- as.numeric(dataset_toyota$HP)
dataset_toyota$Automatic <- as.numeric(dataset_toyota$Automatic)
dataset_toyota$Doors <- as.numeric(dataset_toyota$Doors)
dataset_toyota$Gears <- as.numeric(dataset_toyota$Gears)
dataset_toyota$Quarterly_Tax <- as.numeric(dataset_toyota$Quarterly_Tax)
plot_missing(dataset_toyota)
```

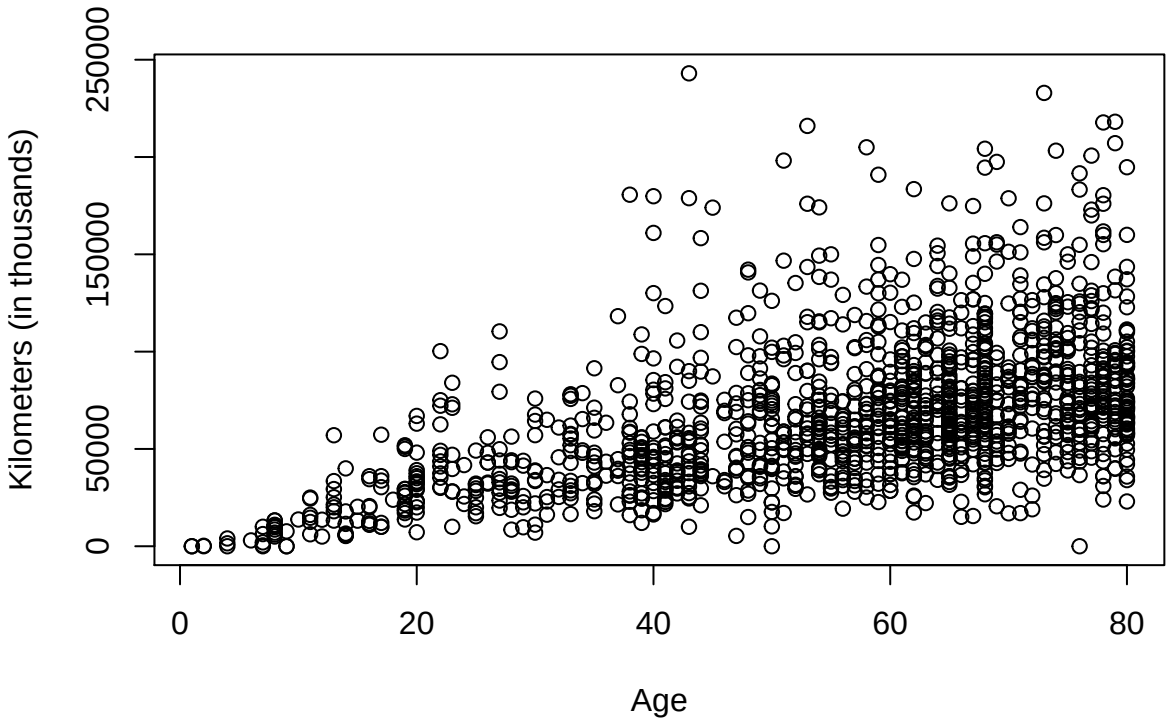


So in this preprocessed dataset there are no missing values.

Visualizations

```
plot(dataset_toyota$Price, dataset_toyota$Age_08_04,
     xlab = "Prices of the Car",
     ylab = "Age of the Car", main = "Age vs Price")
```


```
plot(dataset_toyota$Age_08_04,dataset_toyota$KM,xlab="Age",
     ylab="Kilometers (in thousands)")
```



From plot of *Age vs Price* the behaviour in negative correlation, further it can be said the price of the car increases when the age decreases and vise versa. The second plot defines the Km with respect to Age of the car. As the age increases the km is also increasing.

The boxplot visualizations are important to analyze the cars with the price and variables dependent on it are Automatic, Doors, and Fuel Type. The price of the Diesal cars are high compared to the Petrol and CNG. Also the price of the 5 doors cars are high.

The trend of the graph is related to the manufacturing year of the graph, that shows the price of the car in 2004 and 2003 are high as the age of the car is less.

Algorithms

We will be applying regression techniques. 1. Simple Linear Regression 2. Multiple Linear Regression 3. Decision Tree Regression

Simple Linear Regression

```
SLR_data <- dataset_toyota[,-3:-11]
```

Now splitting the data into testing and training set, which consist of 2/3 as training and 1/3 for testing

```
library(caTools)
set.seed(1)
slr_split = sample.split(SLR_data$Price, SplitRatio = 2/3)
slr_training_set = subset(SLR_data,slr_split == TRUE)
slr_test_set = subset(SLR_data,slr_split == FALSE)
```

Now building the slr regressor with Price as the dependent variable

```
slr_regressor = lm(formula = Price ~ Age_08_04,
                   data = slr_training_set)
y_pred = predict(slr_regressor, newdata = slr_test_set)
summary(slr_regressor)
```

```
##
## Call:
## lm(formula = Price ~ Age_08_04, data = slr_training_set)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -8545  -1065    -92    815  57275
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 20485.930    259.523   78.94   <2e-16 ***
## Age_08_04    -172.517      4.398  -39.23   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2597 on 981 degrees of freedom
## Multiple R-squared:  0.6107, Adjusted R-squared:  0.6103
## F-statistic:  1539 on 1 and 981 DF,  p-value: < 2.2e-16
```

Here the R^2 = 61.07 which not that good with respect to the SLR.

```
library("forecast")
```

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

```
accuracy(y_pred, slr_test_set$Price)
```

```
##                 ME     RMSE      MAE       MPE     MAPE
## Test set -149.8981 1657.969 1246.885 -2.910371 12.06603
```

```
slr.residuals <- slr_test_set$Price[1:20] - y_pred[1:20]
data.frame("Predicted" = y_pred[1:20], "Actual" = slr_test_set$Price[1:20], "Residual" = slr.residuals)
```

```
##    Predicted Actual   Residual
## 6   14965.37  12950 -2015.3744
```

```
## 11  16173.00  20950  4777.0040
## 12  16690.55  19950  3259.4519
## 13  16173.00  19600  3427.0040
## 14  15137.89  21500  6362.1082
## 21  15310.41  15950   639.5909
## 24  15655.44  16950  1294.5561
## 25  15482.93  16250   767.0735
## 34  16000.48  14950 -1050.4786
## 35  16690.55  15500 -1190.5481
## 39  14965.37  15750   784.6256
## 41  16690.55  13950 -2740.5481
## 42  15827.96  16750   922.0387
## 45  16690.55  16950   259.4519
## 47  15827.96  17950  2122.0387
## 53  16000.48  20500  4499.5214
## 57  15655.44  15250  -405.4439
## 63  15137.89  18750  3612.1082
## 66  16000.48  16950   949.5214
## 67  15655.44  18950  3294.5561
```

The RSME on the test set for the simple linear regression is 1657. The accuracy of the Simple Linear Regression is kind of low. Now building the plotting the graph test results.

Here we can see there are some outliers, and some predicted values are good based on our model.

Mulitple Linear Regression

We have to determine the numerical data to the Fuel_type as it contains the categorical values in the dataset.

```
mlr_data = dataset_toyota
mlr_data$Fuel_Type = factor(mlr_data$Fuel_Type,
                        levels = c('CNG', 'Diesal', 'Petrol'),
                        labels = c(1, 2, 3))
```

Splitting the dataset as same as the Simple model, training as 2/3 and testing as 1/3

```
mlr_split = sample.split(mlr_data$Price, SplitRatio = 2/3)
mlr_training_set = subset(mlr_data, mlr_split == TRUE)
mlr_test_set = subset(mlr_data, mlr_split == FALSE)

mlr_regressor = lm(formula = Price ~ .,
               data = mlr_training_set)
mlr_y_pred = predict(mlr_regressor, newdata = mlr_test_set)
summary(mlr_regressor)
```

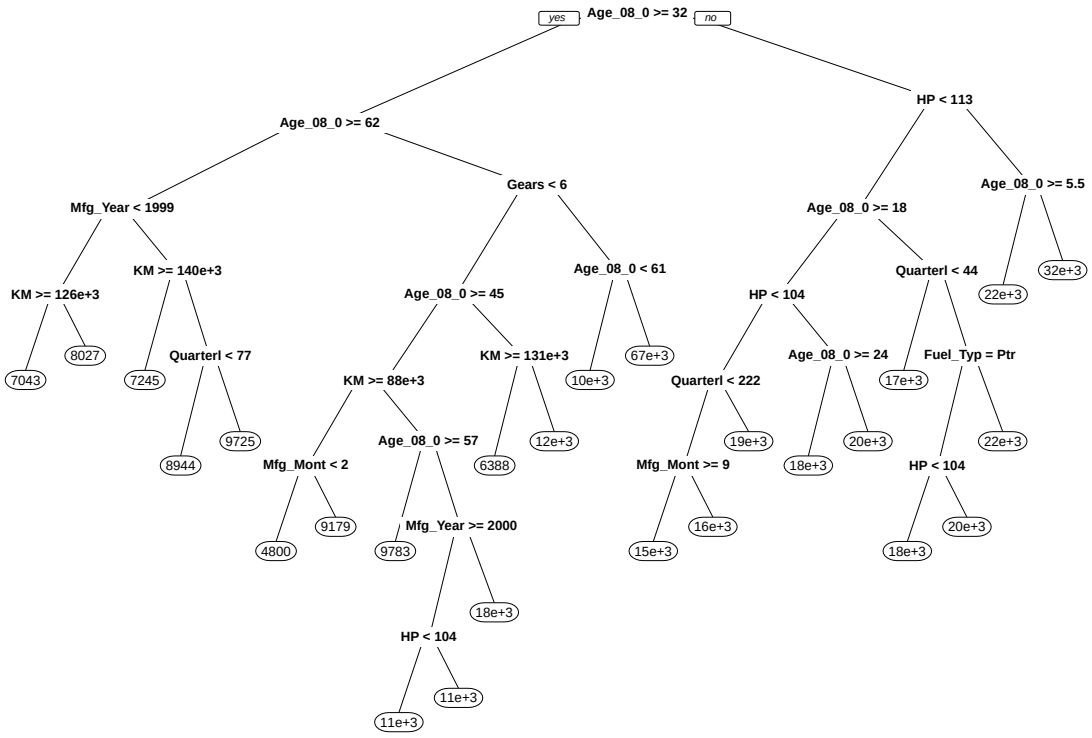
```
##
## Call:
## lm(formula = Price ~ ., data = mlr_training_set)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -3635   -892   -134    698  55056
##
```

```
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.323e+07  4.548e+06   2.909 0.003720 **
## Age_08_04     -6.923e+02  1.890e+02  -3.662 0.000265 ***
## Mfg_Month     -6.637e+02  1.894e+02  -3.503 0.000483 ***
## Mfg_Year      -6.599e+03  2.269e+03  -2.909 0.003722 **
## KM            -1.426e-02  3.060e-03  -4.659 3.67e-06 ***
## Fuel_Type3     1.270e+03  8.122e+02   1.563 0.118333
## HP             4.047e+01  6.203e+00   6.523 1.16e-10 ***
## Automatic      9.807e+02  3.360e+02   2.919 0.003607 **
## Doors          1.736e+02  8.652e+01   2.006 0.045152 *
## Gears          2.551e+03  4.137e+02   6.166 1.07e-09 ***
## Quarterly_Tax  4.322e+00  4.960e+00   0.871 0.383789
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2274 on 873 degrees of freedom
##   (99 observations deleted due to missingness)
## Multiple R-squared:  0.6561, Adjusted R-squared:  0.6521
## F-statistic: 166.5 on 10 and 873 DF,  p-value: < 2.2e-16
```

In Multiple linear regression, most of the variables like Age, Manufacturing Month, KM travelled, Horse Power and Gears are significantly responsible for the Price of the car.

Decision Tree Regression

```r
library("rpart")
library("rpart.plot")
reg_tree_data <- dataset_toyota
set.seed(12)
reg_split = sample.split(reg_tree_data$Price, SplitRatio = 2/3)
reg_training_set = subset(reg_tree_data, reg_split == TRUE)
reg_test_set = subset(reg_tree_data, reg_split == FALSE)
tr <- rpart(Price ~ ., data = reg_training_set, method = "anova",
            minbucket = 1, maxdepth = 30, cp = 0.001)
prp(tr)
```




```
t(t(tr$variable.importance))
```

```
##                      [,1]
## Age_08_04     14289951176
## Mfg_Year      10743466211
## KM             3745648424
## HP             1643190545
## Quarterly_Tax  1525125705
## Mfg_Month       552983393
## Gears           528466290
## Doors            76095714
## Fuel_Type        57331918
## Automatic        13066883
```

Here based on the decision tree vector, the Age, Mfg_Year, KM, HP and Quarterly Tax are important predictors for the Price of the Car.

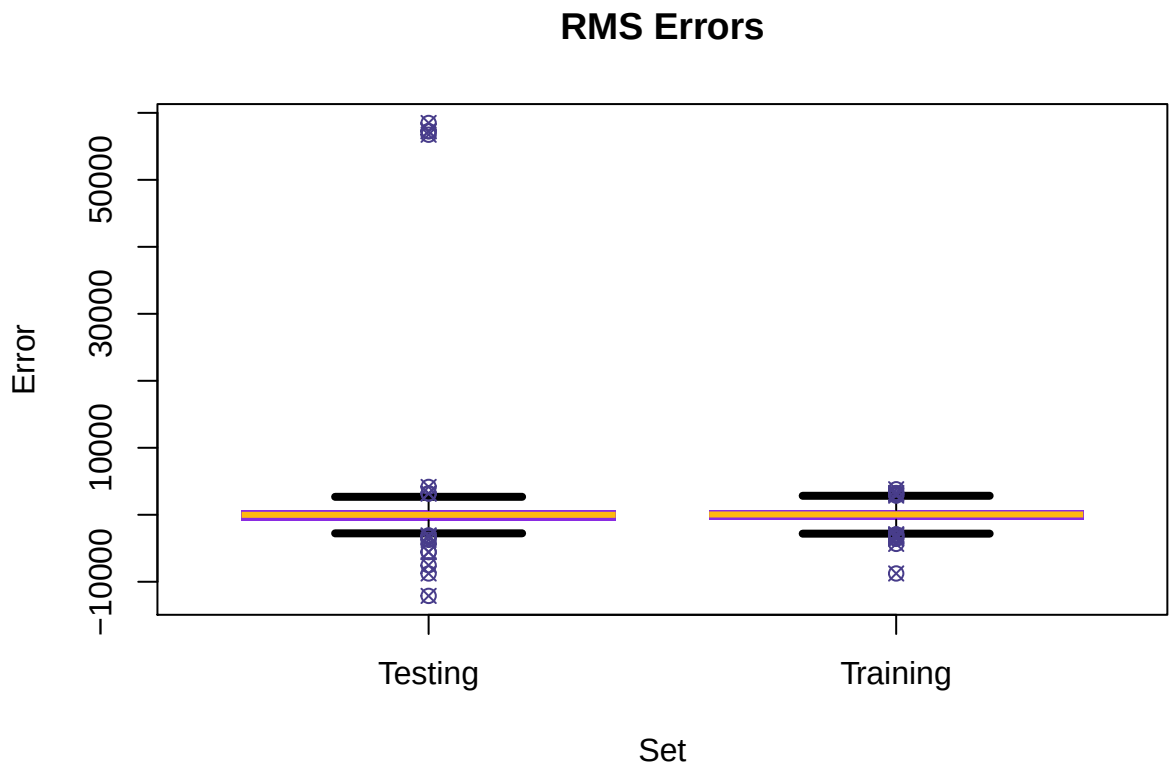
```
accuracy(predict(tr, reg_training_set), reg_training_set$Price)
```

```
##                     ME    RMSE      MAE       MPE    MAPE
## Test set 1.214316e-13 1110.35 853.7229 -1.229954 8.56044
```

```r
accuracy(predict(tr, reg_test_set), reg_test_set$Price)
```

```
##                  ME     RMSE      MAE       MPE     MAPE
## Test set -401.1735 5583.259 1468.738 -5.300753 14.35078
```

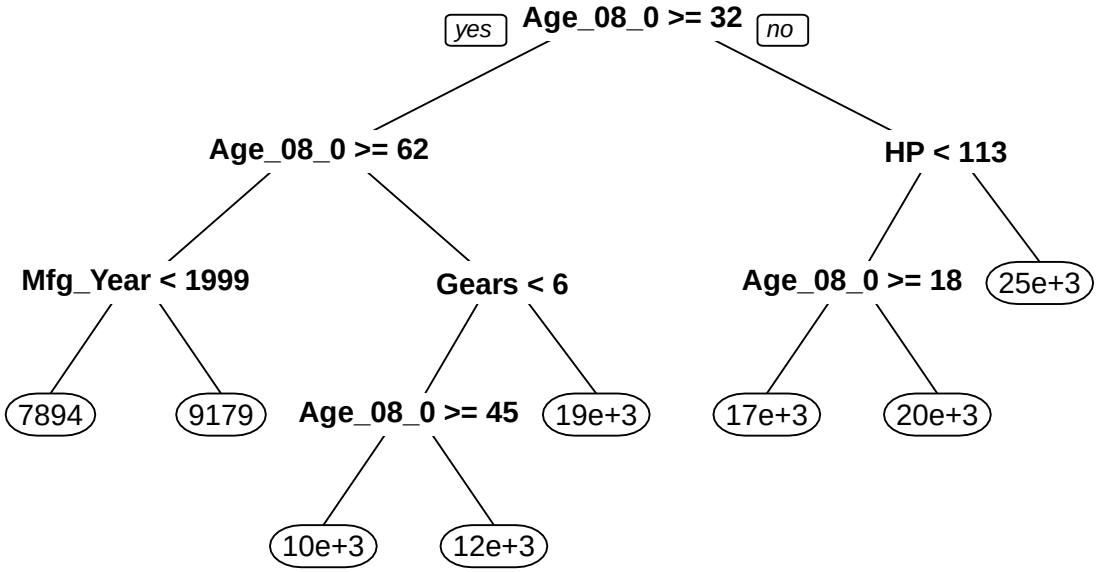
```r
train.err <-predict(tr, reg_training_set) -reg_training_set$Price
valid.err <-predict(tr, reg_test_set) -reg_test_set$Price

err <-data.frame(Error =c(train.err, valid.err),
                 Set =c(rep("Training", length(train.err)),
                        rep("Testing", length(valid.err))))

boxplot(Error~Set, data=err, main="RMS Errors",
        xlab = "Set", ylab = "Error",
        col="blueviolet",medcol="darkgoldenrod1",boxlty=0,border="black",
        whisklty=1,staplelwd=4,outpch=13,outcex=1,outcol="darkslateblue")
```



The testing data has fewer errors compared to the training data as they .

Will try to prune the tree using the cross-validationtion error.

```r
tr.shallow <- rpart(Price ~ ., data = reg_training_set)
prp(tr.shallow)
```




```
accuracy(predict(tr.shallow, reg_training_set), reg_training_set$Price)
```

```
##                       ME     RMSE      MAE       MPE     MAPE
## Test set -1.389615e-13 2238.204 1168.431 -2.439118 11.22922
```

```
accuracy(predict(tr.shallow, reg_test_set), reg_test_set$Price)
```

```
##                  ME     RMSE      MAE       MPE    MAPE
## Test set -28.86196 1805.635 1170.447 -1.840501 11.2726
```

According to analysis pruned tree performs worse on training set(RSME = 2238.20 compared to 1110.35) for full tree. The testing set performs worse better with RSME = 1805.63 compared 5583.25. The pruned testing set performes better than pruned training set.This results into underfitting of the model

Conclusion

The Question - How much should I expect to pay for a used Toyota Corolla? Based on the analysis of the three regression models, the best parameters buying the corolla is dependent on Age, HP, KM travelled, how many Gears they have and as the age of the car is more the price of the car will be less, also if the KM travelled is more the price of the car is less. You keep these parameters in mind while purchasing the car.