## ISDS 574

**Data Mining for Business Applications**

# PRICE PREDICTION FOR USED CARS


**FALL 2019**

**SUBMITTED TO**

**Dr. YINFEI KONG**

**GROUP MEMBERS**

**ALEKHYA BOTTA**

**CHERISH REDDY**

**KINJAL PARIKH**

**MINESH BAROT**

**PRASHANT SHARMA**

**PRIYANKA KUNDER**

**SMRITI JAIN**

# INDEX

## Executive Summary

The used-car market in the United States is more than twice the size of the new-car segmented is outpacing it in growth. This industry is estimated to be around more than 350 Billion USD. For the past few years, this market has sailed over all the tides and still holding steady. This segment offers a safe harbor from the dramatic sales high and lows as seen in among the new vehicle segment. Historically, used-car sales have had a less volatile reaction to the market scenarios.

The purpose of this analysis is to develop a price predicting model for the used-car market. For this purpose, data fetched from Craigslist. This data consists of 22 variables with an estimated 500,000 observations. In order to determine variable significance and accurate predictions, three models were conducted.

The models implemented in the project are Multiple Linear Regression, K-Nearest Neighbors, and Classification and Regression Tree

The best model out of these was identified to be multiple linear regression. This model boasted the lowest RMSE value of 6047.269. This value is the lowest as compared to the RMSE for k-NN, which turns out to be 8220.9859 and 6564.735, for the Classification and Regression Tree. Thus, concluding the fact that multiple linear regression gives the best-predicted price for the used-cars in the dataset taken for the United States market.

## Introduction

In the project related to data mining, the team collectively decided to use a data set from the website Kaggle. The group selected a dataset that has information related to used cars. The dataset that is being used for the project has data retrieved from Craigslist which is a website that has the world's largest collection of used vehicles that are put up for sale.

The team undertook the task of analyzing the enormous dataset with several variables with different data types and perform prediction on it. Three techniques have been implemented in this project to carry out the task of prediction. Firstly, Multiple Linear Regression has been applied to achieve the goal. Secondly, Classification and Regression Tree algorithm has been employed to accomplish a successful prediction. Finally, KNN – K Nearest Neighbor technique has been implemented to execute prediction.

The application of the above-mentioned algorithms is a tedious task for the given dataset as it exceptionally big. A general assumption with a huge dataset is that it will have numerous issues. Therefore, the team carried out extensive data preprocessing that eventually helped in improving the accuracy of the models. In addition to cleaning the data, the team also carried out exploratory data analysis to gain a comprehensive overview of the dataset. This analysis involved the creation of multiple visualizations which reinforces the findings from the data preprocessing steps.

Once the data cleaning and exploratory data analysis were completed, the team proceeded to apply the prediction models. Each model was applied successfully after multiple iterations of modifying the code. The results for the same are shared below as the report progresses.

## Problem Description & Background

The dataset being has factors related to used cars. It has multiple variables; for example, odometer, color, and many more. These variables form different columns in the dataset and every column has either a numerical, or categorical value. Despite the dissimilar datatypes, all these variables have an impact on the resale value of the car.

Some of the variables have a greater effect on the price while the others have less. Depending upon the impact of the factors the cost of the used car fluctuates. Hence the aim of the project is to understand how the different factors increase or decrease the values of a car and more importantly develop appropriate models to predict the cost of used cars.

According to NYTimes the purchase of new cars steadily rose from 2009 to 2016. Currently, the trend has been shifting to buying reconditioned cars. Consequently, forecasting the cost of used cars is not only a significant task. Another reason for performing the prediction is that the cost of second-hand cars is not constant in the market. Furthermore, craigslist can utilize this model to recommend prices for customers selling their cars. Hence, a model that evaluates the prices of a car is immensely helpful for trading.

Python programming language was used to perform the data preprocessing steps. To understand the relation between the price & the different dimensions, exploratory data analysis was performed with the help of Tableau. Finally, the 3 models, MLR, CART and KNN were implemented by using the currently most popular programming language for statistical analysis i.e. R programming.

## Description of Variables

| ***Name*** | ***Type*** | ***Description*** |
|---|---|---|
| City | Nominal | Name of the City |
| URL | Nominal | Link to car on craigslist |
| City URL | Nominal | Craigslist link |
| Price | Numerical | Price of the vehicle |
| Year | Numerical | Release year of the car |
| Manufacturer | Categorical/Nominal | Brand |
| Make | Categorical/Nominal | Model |
| Condition | Categorical | Excellent/Good/Like New/Fair/New/Salvage |
| Cylinders | Numerical | Number of Cylinders |
| Title_Status | Categorical | Clean/Rebuilt/Salvage/Linen/Missing/Parts only |
| Transmission | Categorical | Automatic/Manual/Other |
| Drive | Categorical | 4wd/fwd/rwd |
| Size | Categorical | Full/Mid/Compact/Sub Compact |
| Type | Categorical | Type of the car(Sedan/SUV etc) |
| Paint Color | Categorical | Car Color |
| Odometer | Numerical | Number of miles run by the car |
| Desc | String | Description |
| Latitude | Numerical | Coordinates for location of car |

| | | |
|---|---|---|
| Longitude | Numerical | Coordinates for location of car |
| Fuel | Categorical | Gas/Diesel/hybrid/Electric etc |
| VIN | String | Unique Identification Number |
| Image URL | Text | Website link to the car photos |

**Table 1:** *Variables in the Data Set*

## Data Pre-Processing

The data set used for the analysis is massive and hence the probability for noisy data is high. Hence the team decided to perform data pre-processing to obtain clean data; which will result in better prediction accuracy.

| url | city | city_url | price | year | manufact | make | condition | cylinders | fuel | odometer | title_statu | transmissi | VIN | drive | size | type | paint_col | image_url | desc | lat | long |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| https://ab | abilene, T | https://ab | 9000 | 2009 | chevrolet | suburban | good | 8 cylinders | gas | 217743 | clean | automatic | 1GFN | rwd | full-size | SUV | white | https://im | 2WD 1/2 | 33.1301 | -100.234 |
| https://ab | abilene, T | https://ab | 31999 | 2012 | ram | 2500 | | | diesel | | clean | automatic | | | | | | https://im | www.GE | 30.6484 | -97.8629 |
| https://ab | abilene, T | https://ab | 16990 | 2003 | ram | 3500 | | | diesel | | clean | manual | | | | | | https://im | www.GE | 30.6485 | -97.8624 |
| https://ab | abilene, T | https://ab | 6000 | 2002 | gmc | sierra 150 | good | 8 cylinders | gas | 195000 | clean | automatic | | 4wd | | pickup | white | https://im | 2002 GMC | 32.4444 | -99.9924 |
| https://ab | abilene, T | https://ab | 37000 | 2012 | chevrolet | 3500 | excellent | 8 cylinders | diesel | 178000 | clean | automatic | | 4wd | full-size | pickup | silver | https://im | 2012 | 32.7817 | -98.9422 |
| https://ab | abilene, T | https://ab | 3700 | 2003 | | F150 | fair | 8 cylinders | gas | 269000 | clean | automatic | | 4wd | | pickup | silver | https://im | Silver 200 | 32.5796 | -99.6635 |
| https://ab | abilene, T | https://ab | 19950 | 2013 | ford | f-250 | | 8 cylinders | gas | 116792 | clean | automatic | 1FT7V | 4wd | full-size | pickup | white | https://im | DKR | 32.736 | -97.1336 |

**Figure 1:** *Snapshot of the Data Set*

Since our data contains many categorical variables, we decided that imputing the missing values simply with mode may make the data set biased. So, we chose to delete those records with missing values and proceed with the next steps of data preprocessing.

These columns which we found to be not significant in implementing our models were dropped – URL, City URL, Latitude, Longitude, Description, and Image URL.

We observed that the column "***Year***" has only 25% data less than 2006, and we decided to analyze more recent years (>2005) which can capture the trends and patterns in the automobile industry . Therefore, we set the limit of greater than 2005 for the column Year. In the next steps, it is noted that the variable year is not the same as the number of years a car is used. Upon plotting the below chart, it was seen that the price gradually rises as the year increases. However, the reason for the increase is that for the year 2006 and the years before that the number of cars produced were low compared to recent years. So, we decided to drop the variable 'Year'.

For the column "***Odometer"*** only values between 12000 and 120000 were utilized. This specific range was selected because based on the researched carried it was discovered that the average mileage run by car in the United States ranges from 12000 to 12000 miles. The column has numerical data and hence there was no need for creating dummy variables.

Similarly, a range was set for the column "***Price***" as well. The range selected for price is from 1000 to 100000. The reason for choosing this range was to avoid utilizing either extremely low or extremely high values. Furthermore, prices beyond the mentioned range could be in the dataset due to data entry errors.

The column ***'Cylinders'*** has its value in the form of numbers as well as characters. For example, one of its values is '6 cylinders.' From this value only the number 6 is important for us. Hence the number has been extracted from the entire value.

The ***'Manufacturer'*** column has more than 30 unique values which is difficult to implement in machine learning models. So, it has been divided into three separate categories viz; Luxury, Budget, and Truck depending on the manufacturer. And, dummy variables were created for these three new levels.

The variables '***Condition'*** has six levels, viz; like new, like new, excellent, good, fair, salvage. Dummy variables have been created for these levels as well. Similar approach has been followed for the variables '***Fuel'***, '***Transmission'*** and ***'Drive'.***

For the column '***Size'*** a slightly different approach has been followed. We regrouped the categories compact and subcompact as compact. Hence, there are three different levels for the column size viz; full size, mid-size, compact and then converted into dummy variables.

The dataset has an eclectic mix of cars. Hence the column ***'Type'*** has several different values. These values are categorical as well. Therefore, Type has been divided into seven separate columns ranging from type 1 to type 7. Off road, pickup truck, and bus have similar price and hence have been grouped in type 1. Van and Mini-van have been clubbed together in type 2. Coupe and convertible have been combined and put into type 3. Type 4 includes hatchbacks and wagons. Type 5 and 6 have the values sedan and SUV respectively. Finally Type 7 has other types of cars. For every type a dummy variable has been created.

Finally, for the column '***Paint Color'***, we regrouped them based on our domain research. We regrouped the colors white, black, grey, black, blue, silver, and red into the group Color1 and the rest are put into Color2.

## EDA

After thorough cleaning, preprocessing of the dataset, we obtained the quality data which was ready for explanatory data analysis. With the help of Tableau, we were successfully able to extract several insights from the dataset, and which eventually helped in building the machine learning models. Aggregate functions and conditional calculations have been utilized to develop the visualizations in Tableau. Below is the explanatory data analysis depicting the relationship between various independent variables and dependent variable "Price":

### Condition:

The visualization shows us the median price of used cars depending on their condition:

- A used car in a fair condition has the median price of $ 4,795 and contributes 0.39 % of values to the entire dataset.
- A salvage conditioned car has a median price of $ 6,998 and forms only 0.18 % of data points.
- A good conditioned car has a median price of $ 14,500 and forms 15% of the whole data.
- An excellent conditioned car has a median price of $ 14,988. Therefore, the price difference between the good and excellent is not very high. However, excellent conditioned cars supply 59.21% of values i.e. more than half of the values in the dataset.
- Used cars in new condition have the same median price as excellent used cars, however it forms only 0.19% of the data which means very less no of cars have new condition.

- Finally, cars in like new condition have a median price of $16,999 and provides 14.27 % of data.

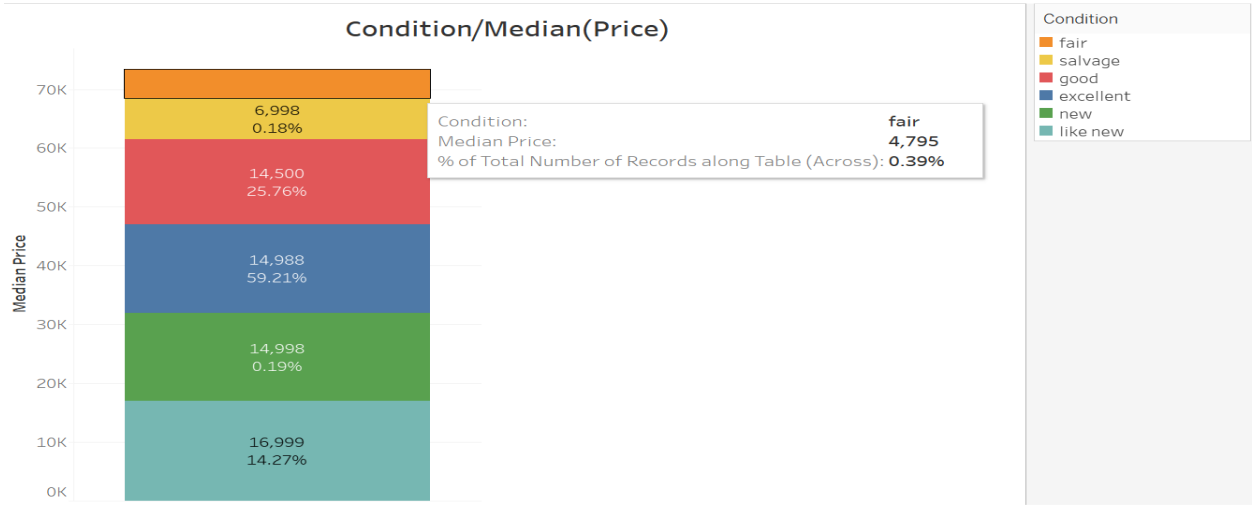


**Figure 2:** *Condition Vs Price*

**Here, we can see that as a condition of the car increases, the price of the car also increases. So, the price is directly dependent on the condition of the car.**

**Color:**

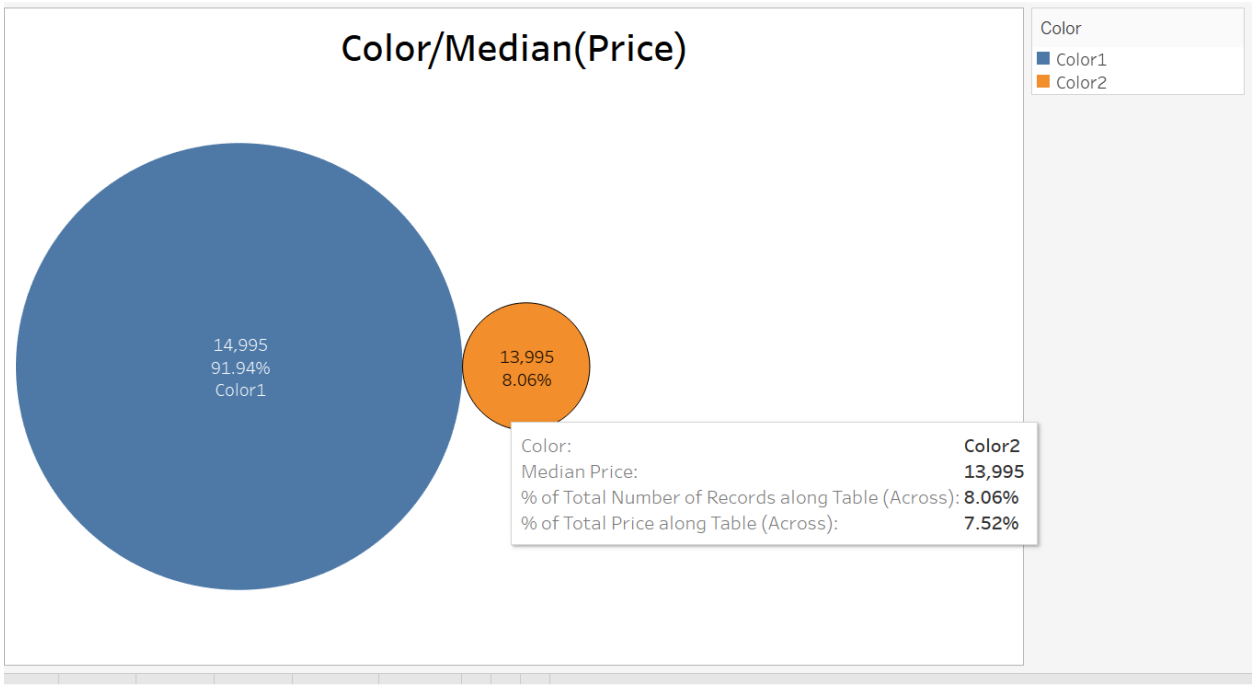


**Figure 3:** *Color Vs Price*

From the data preprocessing, it is known that the color category has been divided into Color1 and Color2. Color 1 consists of **white, black, grey, blue, silver, and red.** The remaining colors are grouped into Color2. Looking at the chart, it can be concluded that:

- Our dataset consists of 92% of cars which are either white, black, grey, black, blue or red in color i.e. Color1 and 8% of cars with other colors i.e. Color2. Additionally, we can see that cars having color1 has higher prices.

**From this distribution, we can say that cars having color as white, black, grey, blue, silver and red have higher prices as compared to cars of any other color.**

## Drive:

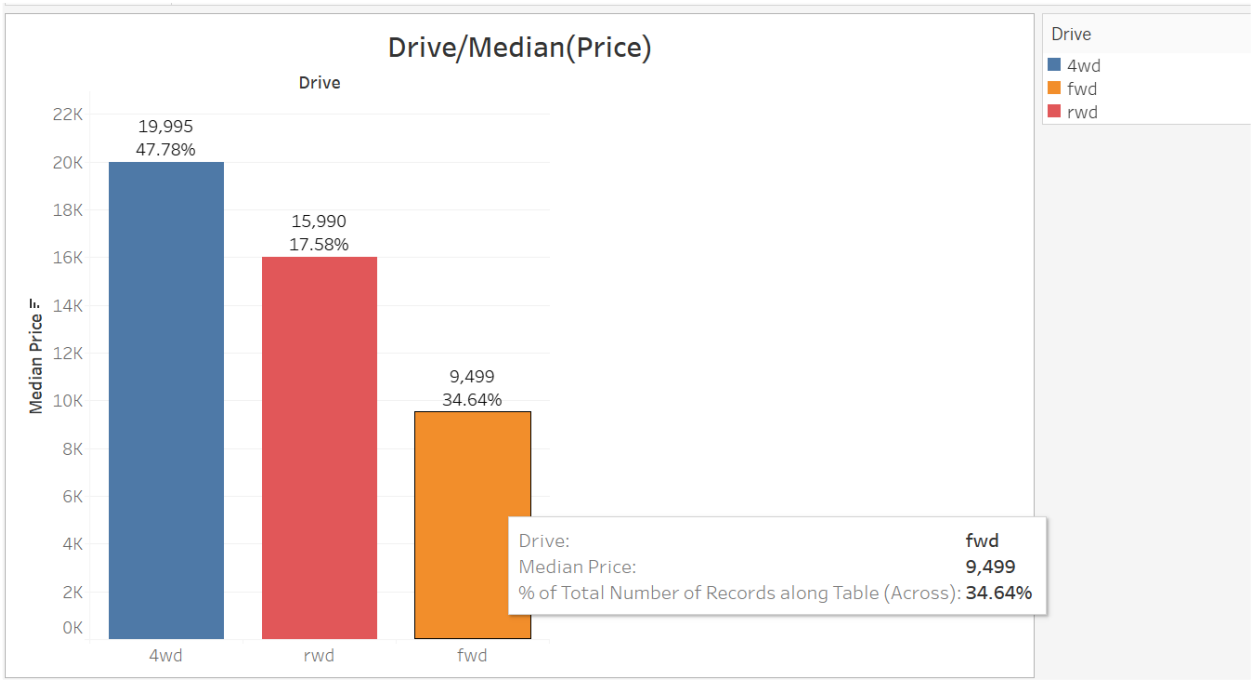


**Figure 4:** *Drive Vs Price*

There are three types of driving modes viz- 4wd means four-wheel drive, fwd indicating that the front wheels are pulling the car, and rwd indicates that the rear wheels are pushing the car. To study the impact of drive type on price the bar graph was plotted which depicts that:

- 4wd used cars are greater in number in the dataset. 4wd forms 47.78% of the data and has a median price of $19,995.
- Rwd has a median price of $15,000 and fwd has the median price of $9,499.

**So, from the domain knowledge we know that, 4wd are usually costlier than rwd and fwd and our data reflects the same. Then, 4wd drive has higher prices than fwd because the cost of manufacturing is higher for 4wd drive.**

## Fuel:

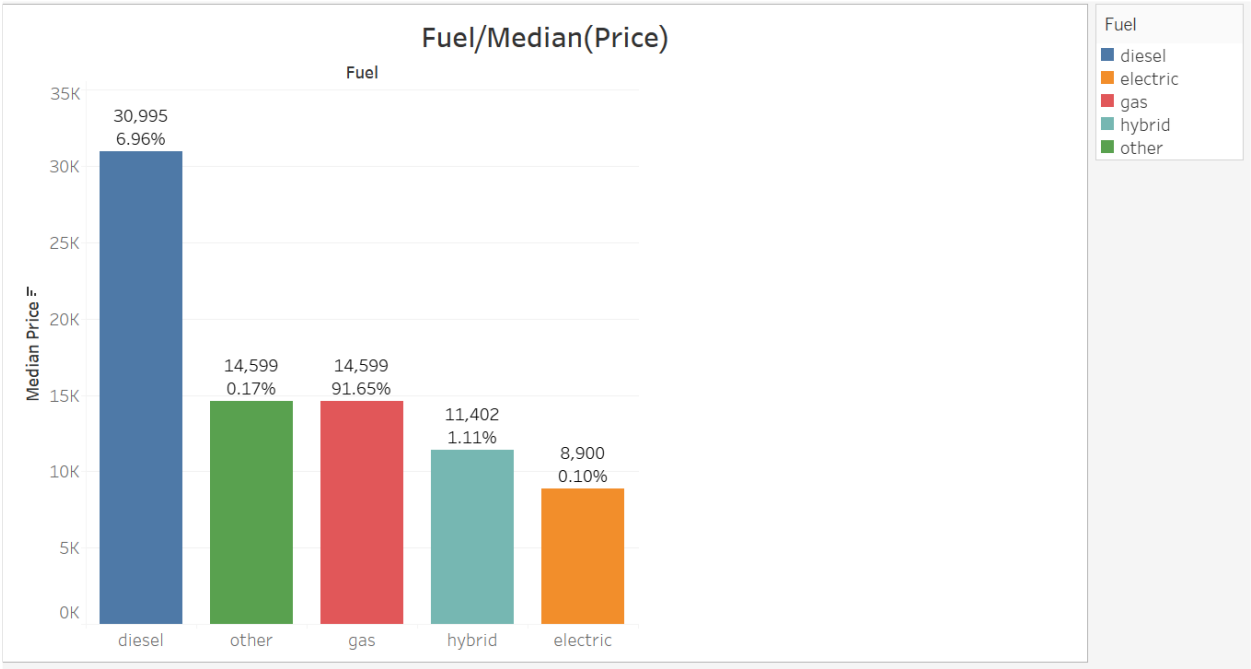


**Figure 5:** *Fuel Vs Price*

The impact of Fuel on the price of used cars is as follows:

- Upon plotting a bar graph, it was realized that diesel cars not only form the highest number of used in the data set but also have the highest median price i.e. $30,995.
- While diesel has the maximum resale value, electric cars have the lowest price i.e. $8,900, and their contribution to that dataset is only of 0.10%.

**From the other analysis, we can say that cars having fuel type as diesel, gas and others have higher prices than any other fuel type.**

## Manufacturer:

It was deemed that the effect of manufacturer on price is crucial. Hence, a bar graph of manufacturer vs price was developed which depicts:

- Trucks have the highest median price i.e. $24,170, and on the other hand Budget cars, true to its name, have the lowest value i.e. $13,999. Luxury cars has a median price of $14,900, slightly above Budget cars with a difference of approximately $ 2,000.

**As the name suggests we can say that budget cars have the lowest price and the trucks have the highest prices.**

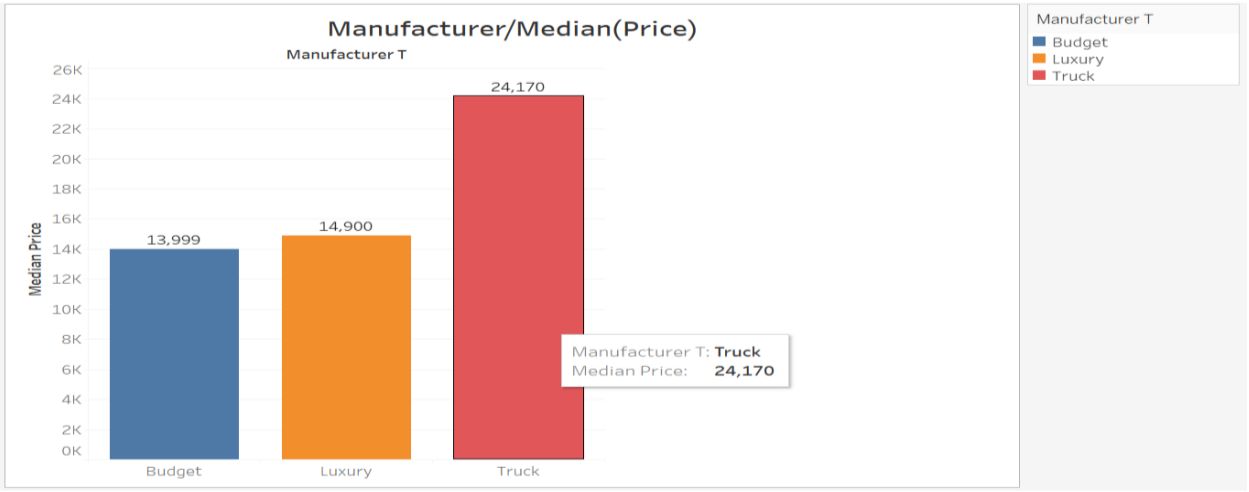


**Figure 6:** *Manufacturer Vs Price*

**Size:**

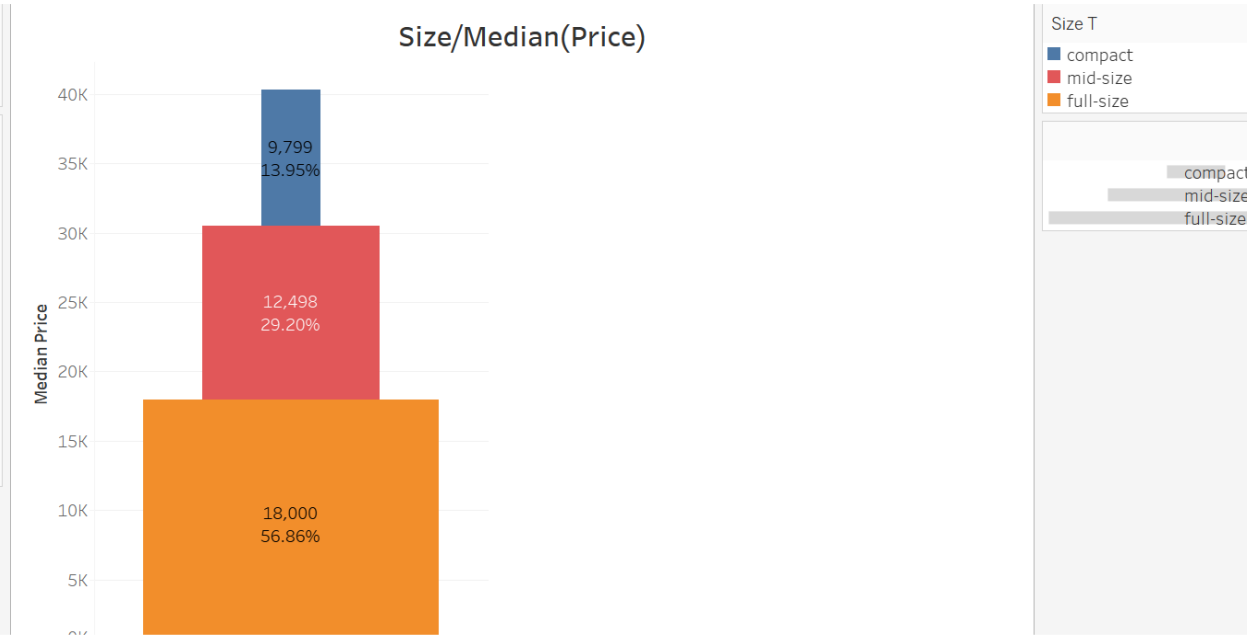


**Figure 7:** *Size Vs Price*

Size is another crucial factor that affects the price of a car. Hence an attempt to study the relation of size to price was perform as well. As mentioned in the data cleaning section, subcompact cars were combined with compact cars into one category. The visualization below depicts:

- The resale value of the car increases as the size increases. Therefore, full sized cars have the maximum value i.e. $ 18,000 whereas compact cars are priced almost half the rate of full-sized cars. Compact cars have a value of $9,799.

**So, if size increases, the prices also increases.**

## Transmission:

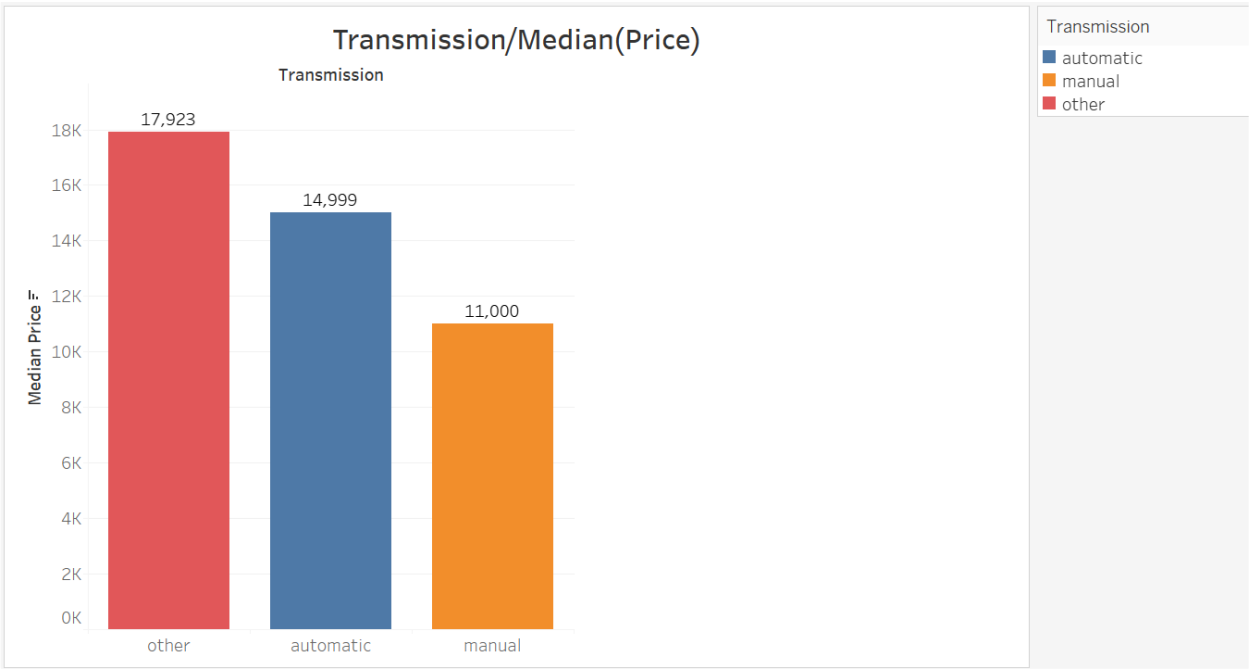


**Figure 8:** *Transmission Vs Price*

The transmission type of a car is also a factor that plays an important role in deciding the cost of a car. The dataset being used has automatic, manual and other as the values for the transmission column. Looking at the bar graph it can be asserted that:

- Cars with transmission type other have the highest median price i.e. $ 17,923 and manual cars have the lowest median price i.e. $ 11,000.

As we don't have much information about what others stands for, we can assume that those cars might have higher transmission values, so that's why the prices are high for hose cars. **Moreover, as we know. Automatic cars have higher prices as compared to manual cars.**

## Type:

Finally, the team looked at how the type affects the price. Data cleaning section informed that type has been divided into seven different columns. Furthermore, it also mentioned what values each column has. Please review the data cleaning section to learn again about the types of cars.

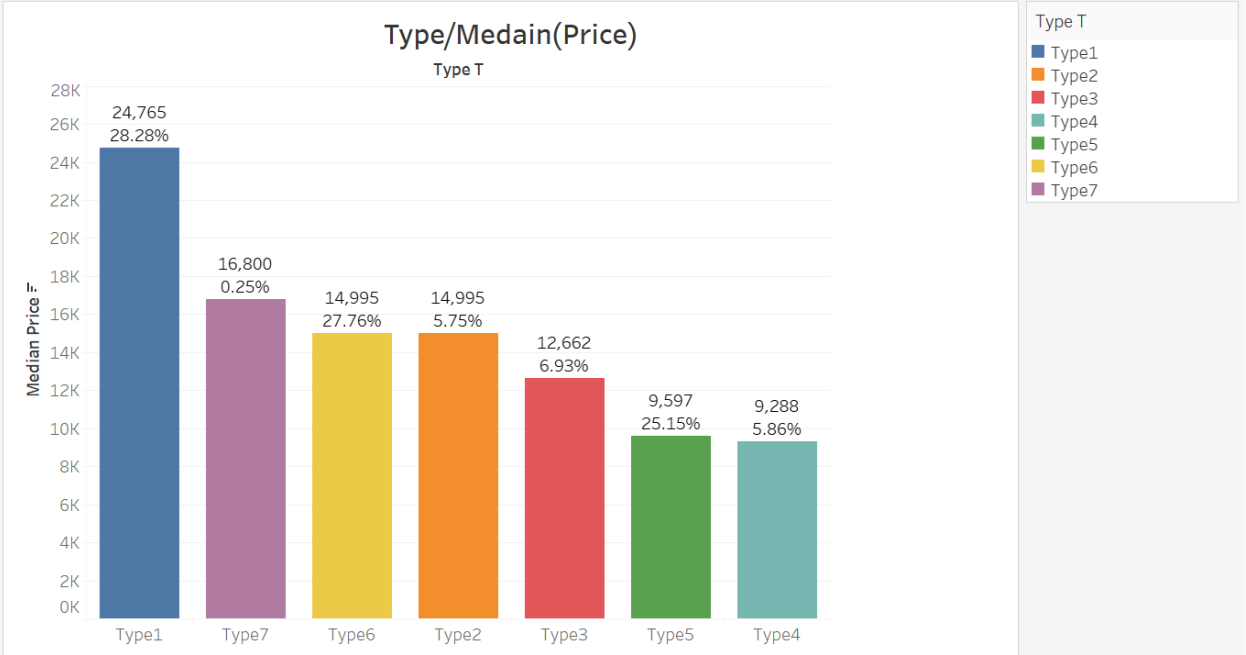


**Figure 9:** *Transmission Vs Price*

The visualization for type vs median price shows that:

- Type 1 has the highest rate which is $24,765. (off-road, pickup, trucks, buses)
- Type 4 has the lowest value which $9,288. (Sedans)
- Type 7 contributes the least amount of data; however, its median price is second highest with a price of $ 16,800.

**So, we can say that trucks, off road, buses and pickups cars have higher prices as compared to sedans, hatch bags, wagons and SUV's.**

# DATA MINING TECHNIQUES

## Multiple Linear Regression

Multiple Linear Regression is a statistical method that helps one determine the relationship between two or more variables by identifying the independent variables and dependent variables. The independent factors affect the dependent variable; which is also known as the target variable. Implementing this model helped the team to understand how each variable contributes either positively or negatively to the Price dimension. Furthermore, it also provided information regarding the variables that have maximum contribution in determining the resale value of a used car.

The process to apply a multiple linear regression model was initiated after the completion of data partitioning. As mentioned in the previous paragraph, this model has dependent and independent variables. For this specific model the target variable is 'price' and every variable except price is considered as independent variable.

As the group proceeded to implement the model it was realized that it extremely critical to check the validity of the data. Thus, several ways to validate the data were discovered and finally Root Mean Squared Error was selected. It is abbreviated as RMSE. RMSE can be defined as the standard deviation of residuals. Standard deviation refers to the dispersion between the values of the variables. Loosely speaking, residuals can be thought of as errors in the result. Residual informs one about the distance between the data points and the regression line. To interpret the results, it is important know that residuals that fall above the regression are line are positive. Conversely, residuals are negative for the points that fall below the regression line.

There are numerous applications for RMSE. For example, it can be used for forecasting, and regression analysis. For this project we will be using it to predict the error for our model. Therefore, it will eventually assist in examining the validity of the model. To calculate the RMSE in R programming the team used the 'hydroGOF' package. This package provides statistical as well graphical goodness-of-fit measures between the observed and the simulated values.

Th graph below has been plotted to determine the relationship between the variables being used for the model. The plotted values on the graph are for residuals and fitted; wherein the x-axis is residuals and the y-axis are fitted. Fitted is nothing but our predicted values. The graph depicts that a linear relationship does exist between the variables of interest.

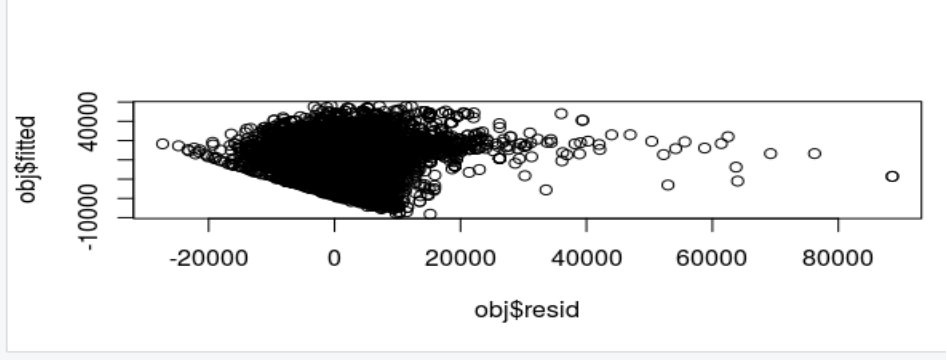


**Figure 10:** *Fitted Vs Residuals*

To implement multiple linear regression effectively it was critical to select the correct variables. To choose the appropriate variables three methods were employed viz; Forward Selection, Backward Selection, and Stepwise Selection. These approaches have been explained further in detail below.

**Forward Selection:**

This approach of variable selection begins with an empty model. Next it adds variables one after the other. The addition of a variable to the model depends on how that variable improves the model. Each time a

variable is added it is tested against a certain criterion to find out if it leads to a better model. Once the model discovers that it is no longer improving by including more variables, the process stops.

```
> summary(obj1) ## ending up with a model with 16 variables

Call:
lm(formula = price ~ cylinders + odometer + diesel_fuel + X4wd +
    Type1 + title_status + good_condition + Type5 + Budget_Type +
    Type4 + fair_condition + full.size + hybrid_fuel + Color1 +
    Type7 + salvage_condition + excellent_condition + fwd + electric_fuel +
    Luxury_Type + Type3 + gas_fuel, data = dat[id.train, ])

Residuals:
   Min     1Q Median     3Q    Max
-28552  -3694   -288   3135  90710

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)         1.160e+04  1.146e+03  10.124  < 2e-16 ***
cylinders           1.719e+03  3.873e+01  44.380  < 2e-16 ***
odometer           -1.443e-01  1.552e-03 -92.932  < 2e-16 ***
diesel_fuel         1.195e+04  1.080e+03  11.064  < 2e-16 ***
X4wd                3.869e+03  1.341e+02  28.862  < 2e-16 ***
Type1               2.616e+03  1.411e+02  18.547  < 2e-16 ***
title_status        2.176e+03  1.893e+02  11.497  < 2e-16 ***
good_condition     -2.207e+03  1.488e+02 -14.828  < 2e-16 ***
Type5              -2.271e+03  1.354e+02 -16.776  < 2e-16 ***
Budget_Type        -1.010e+03  1.598e+02  -6.323 2.61e-10 ***
Type4              -2.264e+03  2.151e+02 -10.525  < 2e-16 ***
fair_condition     -7.166e+03  7.120e+02 -10.064  < 2e-16 ***
full.size           7.883e+02  1.094e+02   7.205 6.02e-13 ***
hybrid_fuel         5.545e+03  1.147e+03   4.837 1.33e-06 ***
Color1              1.115e+03  1.624e+02   6.865 6.86e-12 ***
Type7               5.553e+03  8.790e+02   6.317 2.71e-10 ***
salvage_condition  -7.370e+03  1.187e+03  -6.210 5.39e-10 ***
excellent_condition -6.764e+02  1.299e+02  -5.206 1.95e-07 ***
fwd                -6.489e+02  1.563e+02  -4.151 3.33e-05 ***
electric_fuel       5.927e+03  1.732e+03   3.423 0.000621 ***
Luxury_Type         5.637e+02  2.068e+02   2.726 0.006417 **
Type3              -5.070e+02  2.033e+02  -2.494 0.012650 *
gas_fuel            2.113e+03  1.067e+03   1.980 0.047691 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6302 on 20642 degrees of freedom
Multiple R-squared:  0.6227,    Adjusted R-squared:  0.6223
F-statistic:  1549 on 22 and 20642 DF,  p-value: < 2.2e-16
```

**Figure 11:** *MLR Results – Forward Selection*

The summary function is used for getting the output of residuals and coefficients.

The lm function in the figure above displays the dependent as well as all the independent variables sued in the regression model.

For the variables from cylinders to all the way down to the variable electric_fuel there are three asterisk marks on the extreme right-hand side of the table. This is an indication that the p value for these variables is less than 0.05 and that they contribute significantly to the regression model.

Also, the R squared value is 0.6227 i.e. 62.27 %. When the value of R squared is compared to our large data set that has approximately 500,000 rows, it can be said that is a good value.

**Backward Elimination:**

In contrast to Forward Selection, backward eliminations begin by including all the independent variables in to the equation. The next step, on the one hand is like forward selection, whereas on the other hand it is different. Backward elimination calculates how much each variable contributes towards improving the model. Subsequently, if it contributes significantly to the regression equation it is kept in the model, else

it is deleted from the model. The elimination of the variables stops when no further scope of improve is left.

```
> summary(obj1) ## ending up with a model with 16 variables

Call:
lm(formula = price ~ cylinders + odometer + diesel_fuel + X4wd +
    Type1 + title_status + good_condition + Type5 + Budget_Type +
    Type4 + fair_condition + full.size + hybrid_fuel + Color1 +
    Type7 + salvage_condition + excellent_condition + fwd + electric_fuel +
    Luxury_Type + Type3 + gas_fuel, data = dat[id.train, ])

Residuals:
   Min     1Q Median     3Q    Max
-28552  -3694   -288   3135  90710

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)          1.160e+04  1.146e+03  10.124  < 2e-16 ***
cylinders            1.719e+03  3.873e+01  44.380  < 2e-16 ***
odometer            -1.443e-01  1.552e-03 -92.932  < 2e-16 ***
diesel_fuel          1.195e+04  1.080e+03  11.064  < 2e-16 ***
X4wd                 3.869e+03  1.341e+02  28.862  < 2e-16 ***
Type1                2.616e+03  1.411e+02  18.547  < 2e-16 ***
title_status         2.176e+03  1.893e+02  11.497  < 2e-16 ***
good_condition      -2.207e+03  1.488e+02 -14.828  < 2e-16 ***
Type5               -2.271e+03  1.354e+02 -16.776  < 2e-16 ***
Budget_Type         -1.010e+03  1.598e+02  -6.323 2.61e-10 ***
Type4               -2.264e+03  2.151e+02 -10.525  < 2e-16 ***
fair_condition      -7.166e+03  7.120e+02 -10.064  < 2e-16 ***
full.size            7.883e+02  1.094e+02   7.205 6.02e-13 ***
hybrid_fuel          5.545e+03  1.147e+03   4.837 1.33e-06 ***
Color1               1.115e+03  1.624e+02   6.865 6.86e-12 ***
Type7                5.553e+03  8.790e+02   6.317 2.71e-10 ***
salvage_condition   -7.370e+03  1.187e+03  -6.210 5.39e-10 ***
excellent_condition -6.764e+02  1.299e+02  -5.206 1.95e-07 ***
fwd                 -6.489e+02  1.563e+02  -4.151 3.33e-05 ***
electric_fuel        5.927e+03  1.732e+03   3.423 0.000621 ***
Luxury_Type          5.637e+02  2.068e+02   2.726 0.006417 **
Type3               -5.070e+02  2.033e+02  -2.494 0.012650 *
gas_fuel             2.113e+03  1.067e+03   1.980 0.047691 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6302 on 20642 degrees of freedom
Multiple R-squared:  0.6227,    Adjusted R-squared:  0.6223
F-statistic:  1549 on 22 and 20642 DF,  p-value: < 2.2e-16
```

**Figure 12:** *MLR Results – Backward Selection*

Just like forward selection, the backward elimination also applied the summary function to obtain a similar output. As mentioned earlier the three asterisk marks on the extreme right-hand side in the output mean significant contribution. In the case of backward elimination, the variables that impact the model do not change. Therefore, variables from cylinders to electric fuel remain the best variables.

Next, the output displays the value of R-squared. Like the coefficients of the variables the value of R-squared does not change as well. Thus, the R-squared value for backward elimination is 62.27%.

**Stepwise Selection:**

This selection approach is a combination of the forward as well as the backward selection methods. It can begin either with all the available predictor variables or with none. Therefore, stepwise selection adds as well as eliminates variables from the model. An analysis is performed at each step. Subsequently, whenever a variable is added to the model, all the candidate variables are examined for their significance. Depending on the significance of the variable it is kept or removed. Hence, if it is below the specified threshold value, it is nonsignificant and is deleted from the model.

```
> summary(obj3) # end up with a model with 14 variables, final model same as backward

Call:
lm(formula = price ~ cylinders + odometer + diesel_fuel + X4wd +
    Type1 + title_status + good_condition + Type5 + Budget_Type +
    Type4 + fair_condition + full.size + hybrid_fuel + Color1 +
    Type7 + salvage_condition + excellent_condition + fwd + electric_fuel +
    Luxury_Type + Type3 + gas_fuel, data = dat[id.train, ])

Residuals:
   Min     1Q Median     3Q    Max
-28552  -3694   -288   3135  90710

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)          1.160e+04  1.146e+03  10.124  < 2e-16 ***
cylinders            1.719e+03  3.873e+01  44.380  < 2e-16 ***
odometer            -1.443e-01  1.552e-03 -92.932  < 2e-16 ***
diesel_fuel          1.195e+04  1.080e+03  11.064  < 2e-16 ***
X4wd                 3.869e+03  1.341e+02  28.862  < 2e-16 ***
Type1                2.616e+03  1.411e+02  18.547  < 2e-16 ***
title_status         2.176e+03  1.893e+02  11.497  < 2e-16 ***
good_condition      -2.207e+03  1.488e+02 -14.828  < 2e-16 ***
Type5               -2.271e+03  1.354e+02 -16.776  < 2e-16 ***
Budget_Type         -1.010e+03  1.598e+02  -6.323 2.61e-10 ***
Type4               -2.264e+03  2.151e+02 -10.525  < 2e-16 ***
fair_condition      -7.166e+03  7.120e+02 -10.064  < 2e-16 ***
full.size            7.883e+02  1.094e+02   7.205 6.02e-13 ***
hybrid_fuel          5.545e+03  1.147e+03   4.837 1.33e-06 ***
Color1               1.115e+03  1.624e+02   6.865 6.86e-12 ***
Type7                5.553e+03  8.790e+02   6.317 2.71e-10 ***
salvage_condition   -7.370e+03  1.187e+03  -6.210 5.39e-10 ***
excellent_condition -6.764e+02  1.299e+02  -5.206 1.95e-07 ***
fwd                 -6.489e+02  1.563e+02  -4.151 3.33e-05 ***
electric_fuel        5.927e+03  1.732e+03   3.423 0.000621 ***
Luxury_Type          5.637e+02  2.068e+02   2.726 0.006417 **
Type3               -5.070e+02  2.033e+02  -2.494 0.012650 *
gas_fuel             2.113e+03  1.067e+03   1.980 0.047691 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6302 on 20642 degrees of freedom
Multiple R-squared:  0.6227,    Adjusted R-squared:  0.6223
F-statistic:  1549 on 22 and 20642 DF,  p-value: < 2.2e-16

>
```

**Figure 13:** *MLR Results –Stepwise Selection*

The same process that was utilized for forward selection and backward elimination was applied for stepwise selection. Hence the summary function resulted in the output of residuals and coefficients.

Again, it is seen that the variables that contribution to the regression model equation don’ change. Furthermore, not only do the variables remain the same but also the value for R-squared remains the same. Therefore, for stepwise selection the R-squared value is 62.27%. After completing the execution of all the variable selection methods, the Root Mean Square Error was calculated for all three approaches. Below is the output received –

```
> ###################################################
> # compare prediction results of forward, backward, stepwise
>
> # forward
> yhat1 = predict(obj1, newdata = dat[id.test, ])
> rmse(dat[id.test, 'price'], yhat1) ## RMSE for test data
[1] 6047.269
>
> # backward
> yhat2 = predict(obj2, newdata = dat[id.test, ])
> rmse(dat[id.test, 'price'], yhat2)
[1] 6047.472
>
> # stepwise
> yhat3 = predict(obj3, newdata = dat[id.test, ])
> rmse(dat[id.test, 'price'], yhat3)
[1] 6047.269
>
```

**Figure 14:** *Feature Selection Results*

The output of the RMSE values has been summarized in the following table.

| Variable Selection Methods | Forward Selection | Backward Elimination | Stepwise Selection |
|---|---|---|---|
| RMSE | 6047.269 | 6047.472 | 6047.269 |

**Table 2:** *Feature Selection Comparison*

One can observe that the RMSE values for each method is almost identical with only a very minimal difference. Forward selection and stepwise selection have the same RMSE value of 6047.269. However, backward elimination differs from the other two approach with a very insignificant difference of 0.203. The RMSE value for backward elimination is 6047.472. With these values being very similar it can be concluded that any of the three methods can be selected. However, for the project the team decided to proceed with forward selection.

## KNN

The K-nearest neighbor method is the simplest prediction method that predicts based on the distance measures. It calculates the distance of a new data point to all the other training data. It then selects the K-nearest data points, where K can be any integer. The value of k is directly proportional to the smoothness of the curve of separation, i.e., larger the k, smoother is the curve of separation which results in a less complicated model whereas, on the other hand, small k value tends to overfit the data which results in a complex model.

**NOTE:** It is essential to have the right k-value when analyzing the dataset to avoid overfitting and underfitting of the dataset.

KNN is a non-parametric learning algorithm, which means that it does not assume anything about the underlying data. This is an essential feature, as most of the real-world data does not follow any theoretical assumption.

The following steps are performed in the above example:

- The k-nearest algorithm is imported from the sklearn package.
- Predictors and target variables are created.
- Moving further, the data is split into training and test data in the ratio 70:30.
- A k-NN model is generated using the neighbor's value.
- Data is trained and then fit into the model.
- Prospect future value predicted.

**RMSE (Root Mean Squared Error)**

RMSE is a quadratic scoring rule that also measure the average magnitude of error. It is the square root of average of squared differences between prediction and actual observation.

In the following process:

**Determining the RMSE value ranging from k=1 to k=20 nearest neighbors by making predictions**

```
RMSE value for k=  1 is: 8795.326395495327
RMSE value for k=  2 is: 8325.33376971175
RMSE value for k=  3 is: 8228.913658264388
RMSE value for k=  4 is: 8220.985868130188
RMSE value for k=  5 is: 8261.4072602948
RMSE value for k=  6 is: 8309.897167843043
RMSE value for k=  7 is: 8358.106074247937
RMSE value for k=  8 is: 8390.3452272024
RMSE value for k=  9 is: 8424.333969310754
RMSE value for k=  10 is: 8453.662211324045
RMSE value for k=  11 is: 8487.78334915354
RMSE value for k=  12 is: 8531.398123537965
RMSE value for k=  13 is: 8561.90314033848
RMSE value for k=  14 is: 8593.070234248813
RMSE value for k=  15 is: 8620.054988447106
RMSE value for k=  16 is: 8649.129040911346
RMSE value for k=  17 is: 8657.942840056576
RMSE value for k=  18 is: 8680.567322484005
RMSE value for k=  19 is: 8701.499611096426
RMSE value for k=  20 is: 8715.996429141873
```

**Figure 14:** *RMSR for k = 1 to k = 20*

**Hyper tuning model parameters using Grid Search CV**

Hyper tuning parameter is a process in which the value of metrics is set before the onset of the learning algorithm. This is done to improve the RMSE value and keep it low.

For this project, Grid Search CV is used to find the optimal value for 'neighbors.'

Grid Search is a traditional way to perform hyperparameter optimization. This works by searching exhaustively through a specified subset of hyperparameters. The data trained multiple times, and the model is tested with each parameter to get the optimal values to get the best RMSE value.

For the model, we specified a range of 'neighbors' to determine which value works best for our model. In order to achieve this, a dictionary is created setting 'neighbors' as the key and using NumPy; an array created from 2 to 20.

The new model using grid search will take in a new k-NN classifier, furthermore, parameters and cross-validation values of 5 is used in order to determine the optimal value of 'k-neighbors'.

After training, we can check the values which performed the best for the 'neighbors.' To achieve this, 'best_params_' is called upon the model.

We derived that 4 is the optimal value for 'n_neighbors.'

Hence, we conclude:

**Best optimal value k=4**

**Pertaining RMSE = 8220.9859**

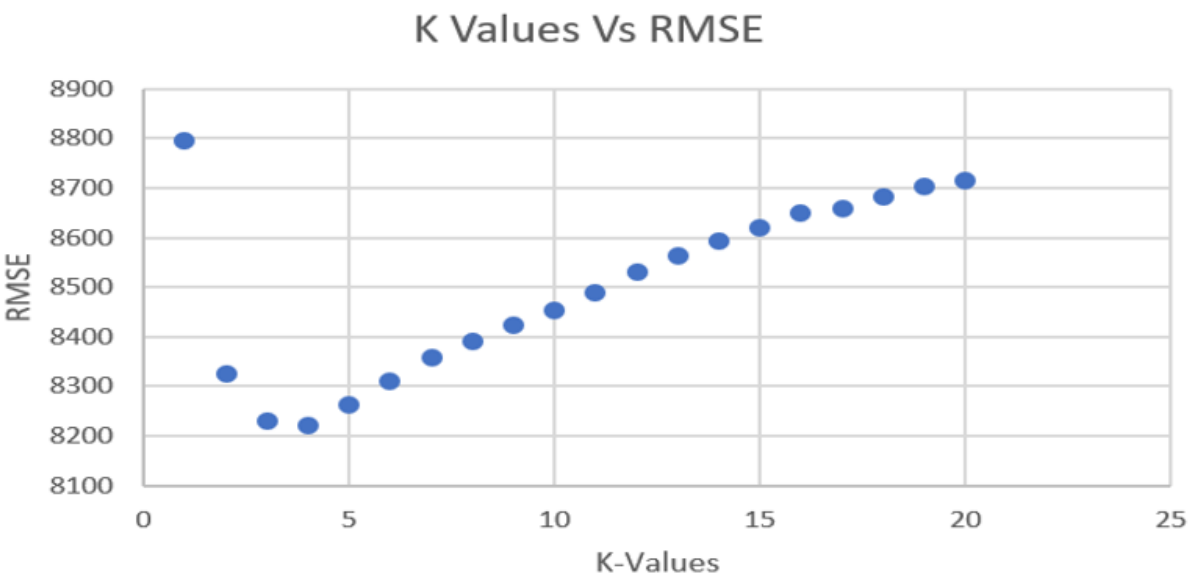


**Figure 15:** *RMSR Vs K-Values*

## Regression Modelling

Basic regression tree partitions a data set into smaller groups and then fit a simple constant for each sub-group. As single tree model is highly unstable and is a poor indicator, the partition is achieved by successive binary partitions based on different predictors. The constant to predict is based on the average response value for all the observations that fall in that sub-group. This technique can be quite effective and powerful.

Performing the following on the dataset:

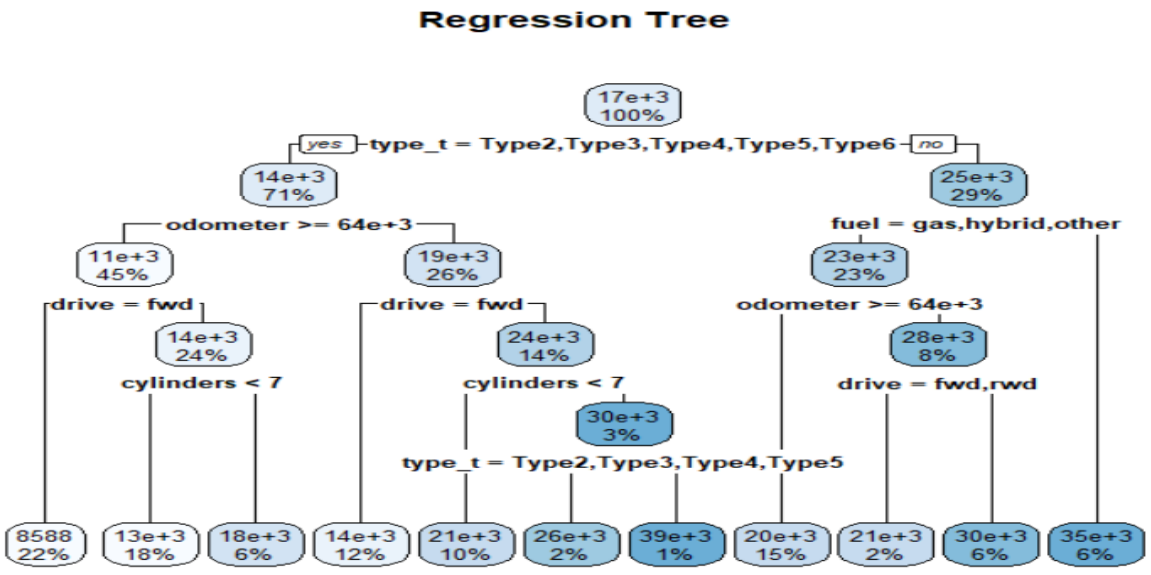


**Figure 16:** *Regression Tree*

In the above chart, it can be seen that the regression tree has 10 splits and 11 terminal nodes. Thus, interpreting from tree that if type is 2,3,4,5,6 then it gets split into left node stating that 71% of car data are either type 2,3,4,5,6; whereas the right node depicts that 29% of the car data is of type 1.

**Minimum Error Tree:**

For this process, using 'r part' package and prune function from r, the regression tree is pruned based on the lowest complexity parameter which yields the lowest error.

| | CP | nsplit | rel error | xerror | xstd |
|---|---|---|---|---|---|
| 1 | 0.234435 | 0 | 1.00000 | 1.00003 | 0.0147441 |
| 2 | 0.093248 | 1 | 0.76557 | 0.76573 | 0.0137664 |
| 3 | 0.063516 | 2 | 0.67232 | 0.67259 | 0.0123574 |
| 4 | 0.055253 | 3 | 0.60880 | 0.60903 | 0.0116271 |
| 5 | 0.033694 | 4 | 0.55355 | 0.55609 | 0.0105218 |
| 6 | 0.030671 | 5 | 0.51985 | 0.52263 | 0.0102871 |
| 7 | 0.019710 | 6 | 0.48918 | 0.49198 | 0.0101618 |
| 8 | 0.011231 | 7 | 0.46947 | 0.47658 | 0.0095482 |
| 9 | 0.010834 | 8 | 0.45824 | 0.46334 | 0.0095968 |
| 10 | 0.010681 | 9 | 0.44741 | 0.45508 | 0.0094779 |
| 11 | 0.010000 | 10 | 0.43673 | 0.44206 | 0.0093296 |

**Figure 17:** *Complexity Parameter Table*

Based on the complexity parameter table, the minimum error tree contains 10 splits, has a complexity parameter of 0.010000 and yields an error of 0.43673. Over here the complexity parameter table is used to select the total number of splits for which the complexity parameter is the lowest. It is also for which the sum of the tree's relative error along with standard error is less than the x-value relative error.

With reference to the table:
Minimum error tree = 0.43673
Standard Error Tree = 0.0093296
Now, taking on the sum of minimum error tree and standard error tree equals to .04460596 which is slightly greater than the x-value relative error of 0.446206. Thus, concluding the fact that minimum error tree contains 10 splits and 11 terminal nodes.

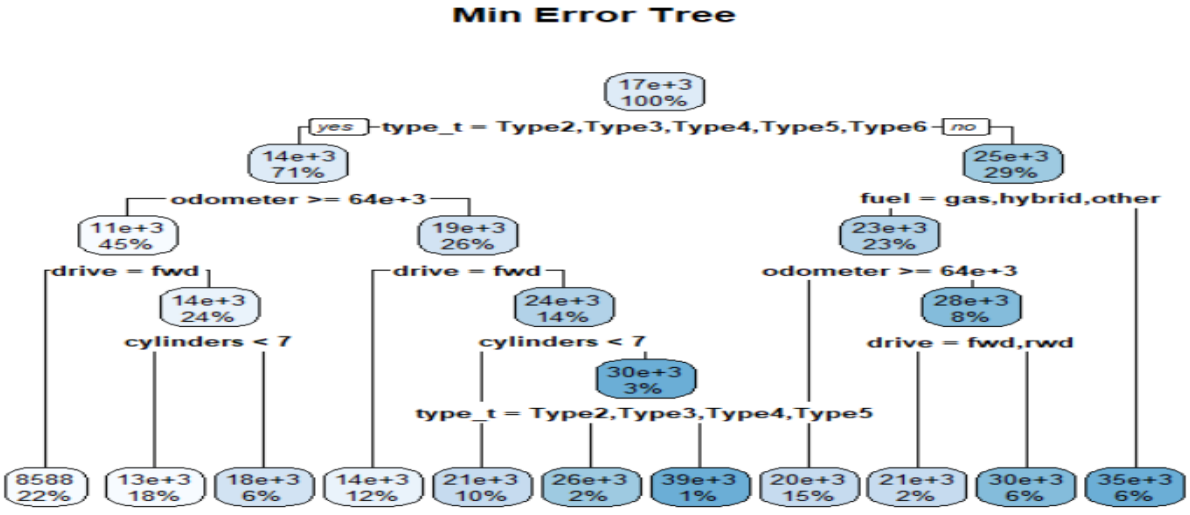


**Figure 18:** *Minimum Error Tree*

The minimum error tree contains similar decision nodes and terminal nodes when compared to the regression tree.

**Complexity Parameter Plot:**

This plot is used to control the size of the decision tree and to select the optimal tree size.

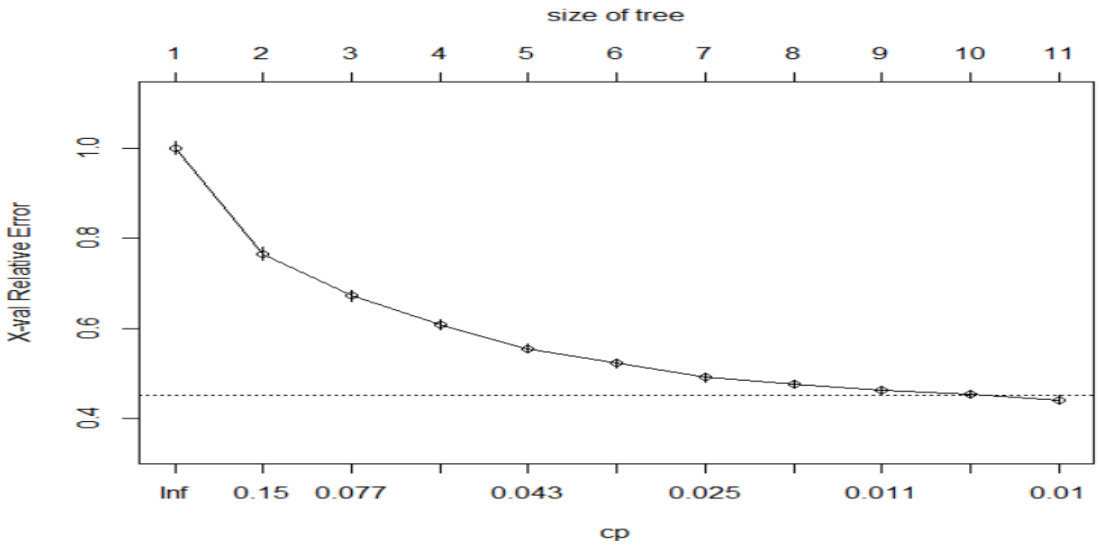


**Figure 19:** *Size of Tree Vs Error*

The plot above determines that the tree containing 10 splits has the lowest error which is within the one standard error of the minimum error tree.

## Result:

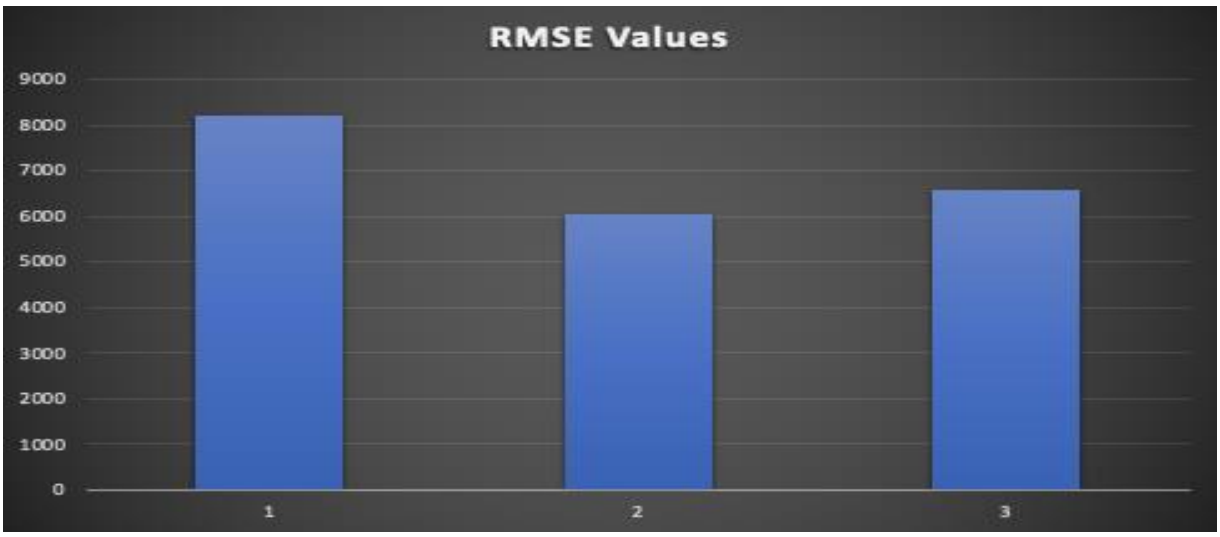


**Figure 20:** *Comparison of RMSE Values*

In the above figure, the bars represent the following models depicted in the project:

1) K-Nearest Neighbor **(8220.9859)**
2) Multiple Linear Regression **(6047.269)**
3) Classification and Regression Analysis **(6564.735)**

Out of the above three models, the best model is based on the RMSE values. When comparing the result of these models, the smaller value of the RMSE indicates a better prediction for the model.

Thus, looking at the above graph, RMSE values for all the models can be compared and therefore, the Multiple linear Regression turns out to be the one with the lowest RMSE value at 6047.269 and thus proves to be the best model for this prediction.

## Conclusion:

The analysis performed in this project helps in validating the future prospect of the data. As the market segment for the used cars in the United States has been increasing substantially with every passing year, this model helps in predicting the future prices for the same.

In the project, the RMSE depicted by all the three models was compared and the lowest is considered the best. Multiple linear regression fulfilled this criterion among the three and thus stands out to be the best prediction model among them.

The practical importance of this tool can be utilized by the car dealers; who in turn can implement this to better manage their stock, have a better share in the market and at the same time provide customer with the best possible price thus, building a long-lasting relationship and a strong foothold in the market.