PRICE PREDICTION OF USED CARS

Milestone: Project Report

Group 12

Student 1: Amar Sai Kiran Poosarla

Student 2: Venkata Sai Krishna Paruchuri

857-230-1191

857-334-6587

poosarla.a@northeastern.edu

paruchuri.v@northeastern.edu

Percentage of Effort contributed by Student 1: 50%

Percentage of Effort contributed by Student 2: 50%

Signature of Student 1: Amar Sai Kiran Poosarla

Signature of Student 2: Venkata Sai Krishna Paruchuri

Submission Date: 04/25/2022

Table of contents

DATA EXPLORATION AND VISUALIZATION

IE 7275: DATA MINING IN ENGINEERING

I. Problem Setting:

According to a recent survey, the automotive industry has seen a decline in the production of cars due to chip shortages compared to previous years. This had a massive impact on car manufacturing and sales. But the demand from the customers has not decreased. In fact, the demand for the used cars has been inflated due to the global pandemic, also for safe and secured travel. This has raised the question of giving a reasonable car price to the customers.

II. Problem Definition:

The Dataset considered is about the reasonable prices of the past 10 years of various car brands, including luxury brands like AUDI and BMW and economy cars like Hyundai and Toyota with their models.

The purpose of this analysis is to predict the price of the used car based upon various specifications of the car like the model, year of manufacturing etc. The analysis we use will help customers to get the price predicted before they get a quote from the used cars companies.

III. Data Source:

The Dataset has been taken from Kaggle.com, an online platform for enthusiastic data scientists and machine learning professionals.

Kaggle.com is an online storehouse for a wide variety of datasets.

Topic of the Dataset: - **100,000 UK Used car dataset**

Dataset source link is https://www.kaggle.com/adityadesai13/used-car-dataset-ford-and-mercedes

IV. Data Description:

The Dataset consist of a total 108,540 instances and have 9 attributes with one target attribute "PRICE"

SNO	Variable	Description
1	Model	Car model
2	Year	Year of manufacturing
3	Price	Price of Used cars
4	Transmission	Type of transmission (Manual/automatic/ semi-auto)
5	Mileage	Total distance travelled
6	Fuel type	Type of fuel (Petrol/Diesel)
7	Tax	Sales tax
8	MPG (Miles per gallon)	Performance in terms of miles per gallon
9	Engine size	Capacity of the Engine

Table-1 Description of variables

V. Data Collection:

Data collected in initial phase is from different sources (car brands) which is later merged. All the data required for exploration, visualization and prediction has been gathered, collected, and cleaned.

Data collection is primarily required for measuring information available on variables used for prediction. Variables used for price prediction are analyzed in systematic way, enabling one to test hypotheses and evaluate outcomes.

Dataset Information:

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 108540 entries, 0 to 15156
Data columns (total 11 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   Brands        108540 non-null  object
 1   model         108540 non-null  object
 2   year          108540 non-null  int64
 3   price         108540 non-null  int64
 4   transmission  108540 non-null  object
 5   mileage       108540 non-null  int64
 6   fuelType      108540 non-null  object
 7   tax           94327 non-null   float64
 8   mpg           99187 non-null   float64
 9   engineSize    108540 non-null  float64
 10  tax(£)        4860 non-null    float64
dtypes: float64(4), int64(3), object(4)
memory usage: 9.9+ MB
```

FIG-1

VI. Data Mining Tasks (Processing):

Initially dataset consisted of 108,540 rows and 11 columns.

The first step towards processing of data is removing the duplicate values which decreases redundancy in dataset. There are around 2284 duplicate records which would have disturbed model's accuracy if not removed.

The tax (£) columns which contained more than 90% of missing values cannot be fitted by mean values hence entire column has been dropped.

All the missing values in individual columns have been replaced by the mean of their category.

After these two steps of cleaning, we still ended up with 4784 and 18 missing values in tax and mileage columns respectively.

These missing values are due to non-availability of non-zero values in entire group by category.

The category is defined as data grouped by 'model', 'year', 'transmission', 'Fuel-type', 'mpg', 'mileage'.

To remove inconsistency in dataset all the remaining null value records have been removed from the dataset.

The final cleaned and preprocessed dataset consists of data cleaning we finally ended up with 101,475 and 10 columns.

	Brands	model	year	price	transmission	mileage	fuelType	tax	mpg	engineSize
0	AUDI	A1	2017	12500	0	15735	0	150.0	55.4	1.4
1	AUDI	A6	2016	16500	1	36203	1	20.0	64.2	2.0
2	AUDI	A1	2016	11000	0	29946	0	30.0	55.4	1.4
3	AUDI	A4	2017	16800	1	25952	1	145.0	67.3	2.0
4	AUDI	A3	2019	17300	0	1998	0	145.0	49.6	1.0
...	...	...	...	...	...	...	...	...	...	...
15152	VW	Eos	2012	5990	0	74000	1	125.0	58.9	2.0
15153	VW	Fox	2008	1799	0	88102	0	145.0	46.3	1.2
15154	VW	Fox	2009	1590	0	70000	0	200.0	42.0	1.4
15155	VW	Fox	2006	1250	0	82704	0	150.0	46.3	1.2
15156	VW	Fox	2007	2295	0	74000	0	145.0	46.3	1.2

101475 rows × 10 columns

FIG-2

The columns transmission and fuel type which are categorical variables are encoded numerically for statistical interpretation purpose.

Statistics (mean, median etc.) have been calculated after following numerical transformation.

	price	mileage	fuelType	tax
count	101475.000000	101475.000000	101475.000000	101475.000000
mean	17058.009204	23284.178251	0.485469	119.663571
std	9879.033578	21254.039236	0.567283	62.687141
min	450.000000	1.000000	0.000000	0.000000
25%	10299.000000	7669.000000	0.000000	125.000000
50%	14800.000000	17506.000000	0.000000	145.000000
75%	20998.000000	32523.000000	1.000000	145.000000
max	159999.000000	323000.000000	4.000000	580.000000

FIG-3

Pairplot has been plotted between the variables to observe the correlation.

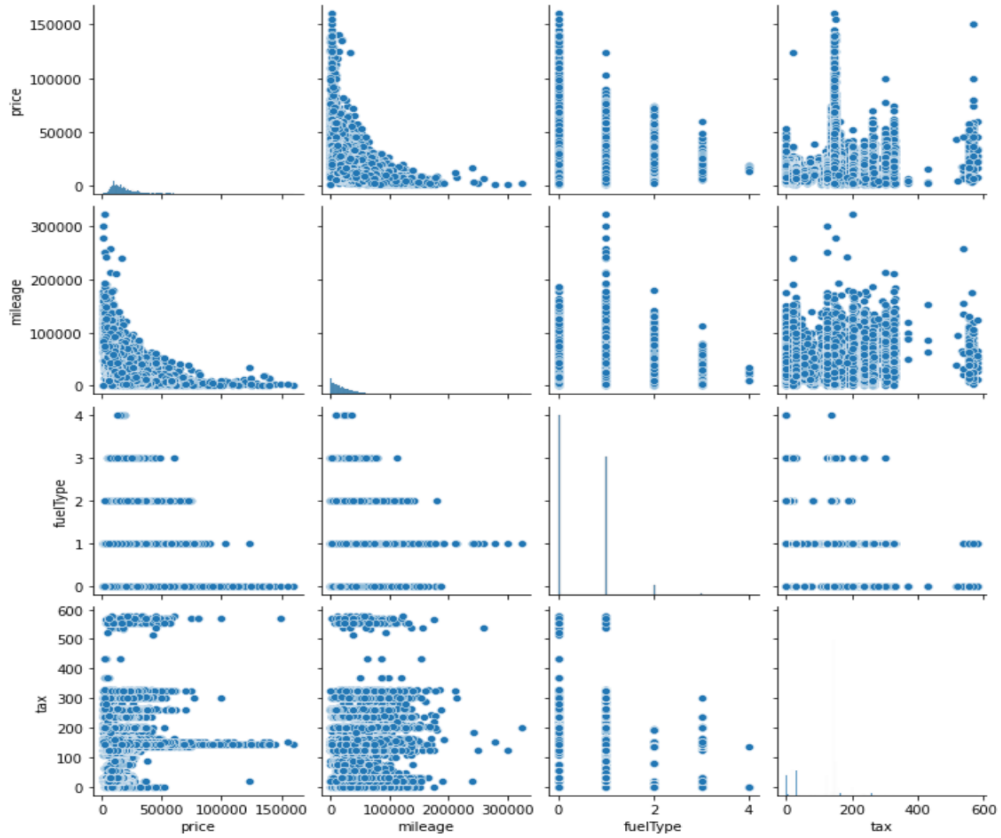

FIG-4

In support of above pairplot we calculated correlation between the numerical variable's mileage, price, fuel type, tax and found that no correlation exceeded 0.5 value.

	price	mileage	fuelType	tax
price	1.000000	-0.431020	0.194308	0.309647
mileage	-0.431020	1.000000	0.202709	-0.228625
fuelType	0.194308	0.202709	1.000000	-0.180989
tax	0.309647	-0.228625	-0.180989	1.000000

FIG-5

Boxplot for price with respect to transmission and differentiation based on fuel type have given insight regarding the quantity of outliers present in data.

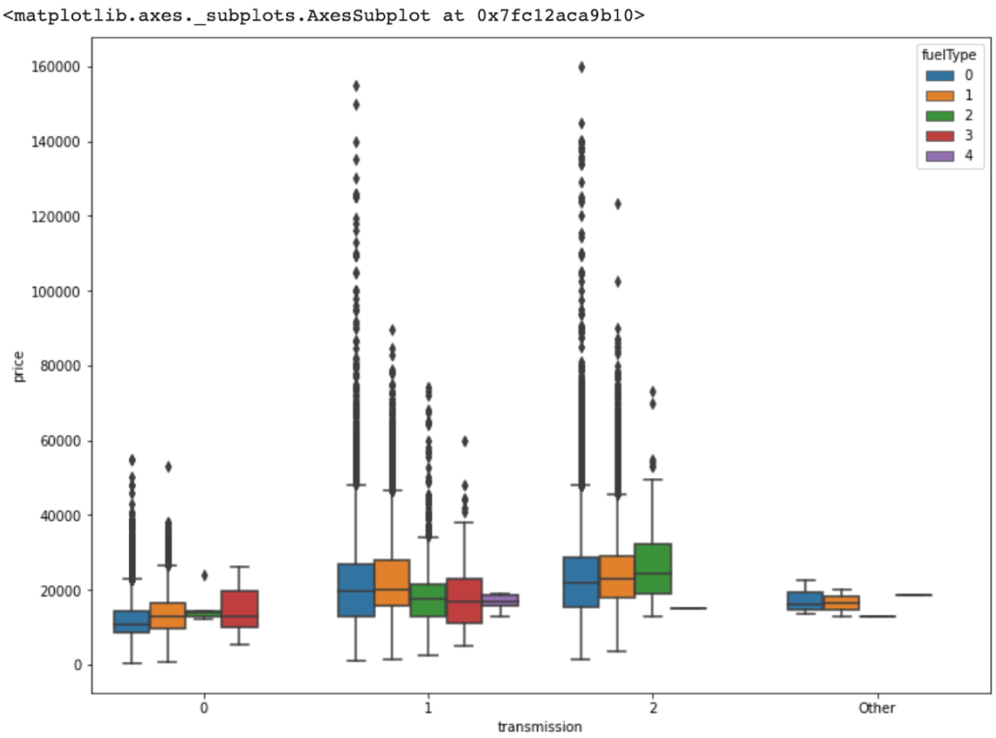

FIG-6

Since dimension reduction cannot be possible based on following two interpretation further data exploration is needed.

VII. Data Exploration and Visualization:

From the above figure **FIG-6** (previous page)

- We found that there are many outliers in our processed data set with the help of boxplot and decided to remove the outliers.
- Box plot has been plotted for price with respect to transmission and only for tax variable.
- After calculating interquartile region for tax distribution, it is observed that there are more than **6000 records** of outliers which is in turn significant percentage of dataset.
- Therefore, if the test score is not as expected for models, outliers will be substituted by their respective mean in every variable and re-test using new dataset.
- The best test score dataset and model will be finalized.

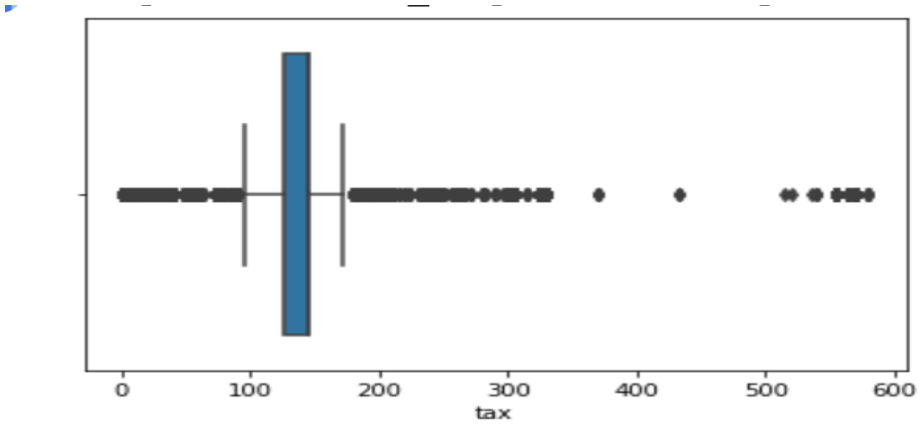

Fig – 7

In fig 7 it clearly shows that there are many outliers only with respect to TAX column

Visualized with and without outliers:

Insight: in the fig-8, bottom right boxplot, outliers are removed and that creates a huge data loss

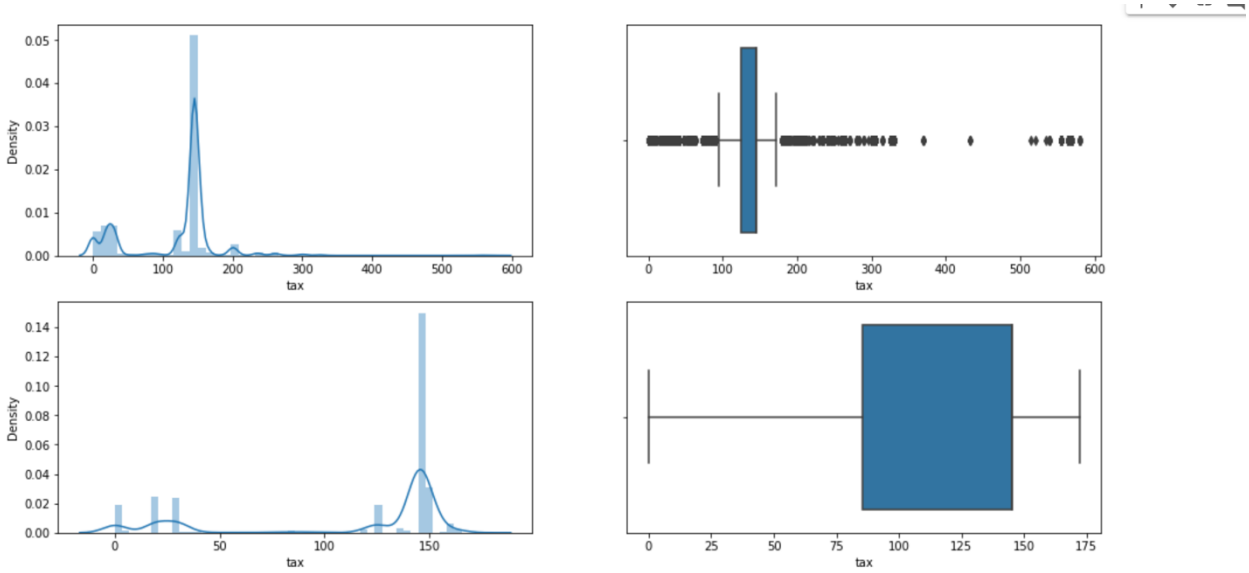

Fig-8

Visualization using heatmap:

Heatmap has been plotted with using the correlation values of four numerical response variables (transmission, fuel Type, mileage, tax).

Insight: It can be observed that there is no high correlation between the response variables Hence dimension reduction is possible.

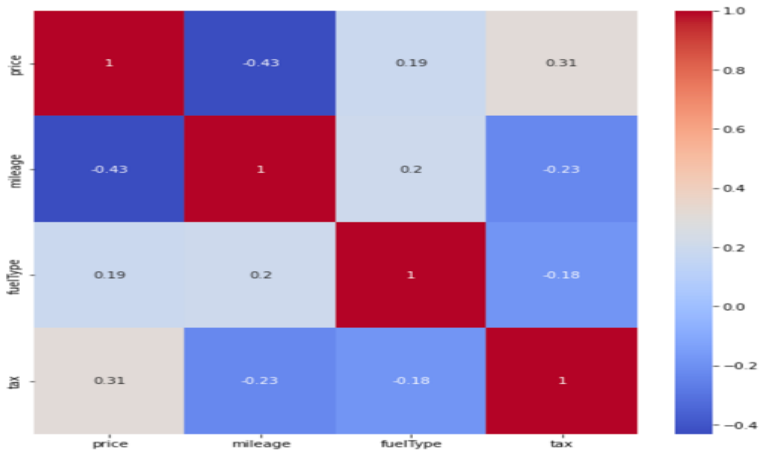

Fig -9

Visualization using Tree map:

A tree map has been plotted for price with respect to brands, models.

Insight: It has been observed that the price for C Class brand and C Class model is highest.

Respective model price to brand is visualized using.

Fig-10

Visualized variation of price (response variable) with respect to each predictor:

Insight for Price vs Transmission: Total price is highest for transmission ‘Manual’

Note: - ‘0’=Manual, ‘1’= Automatic, ‘2’= Semi-auto, ‘3’= Others

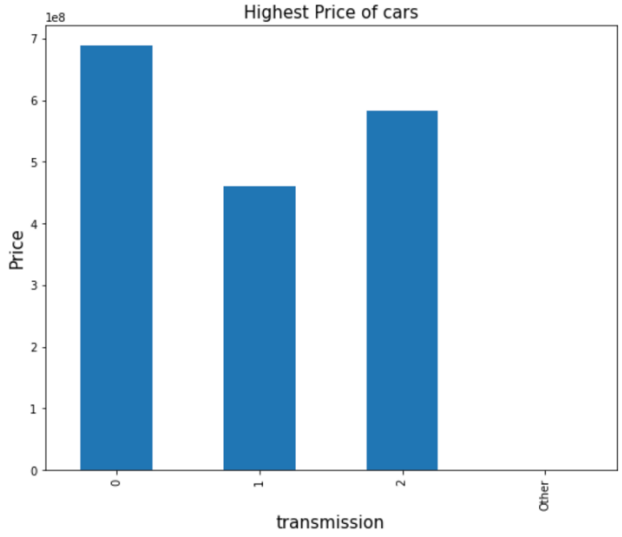

Fig-11

Insight for Price vs Brand: Total price is highest for Brand ‘merc’

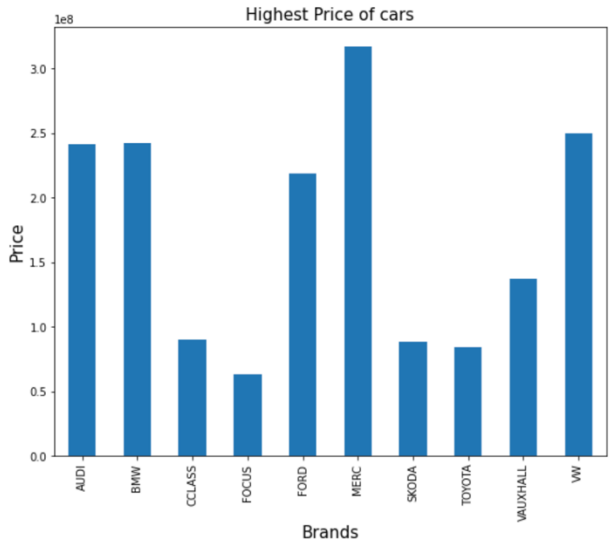

Fig – 12

Insight for Price vs fuel Type: Total price is highest for Petrol

Note: - ‘0’=Petrol, ‘1’= Diesel, ‘2’= Hybrid, ‘3’= Electric, ‘4’= Others

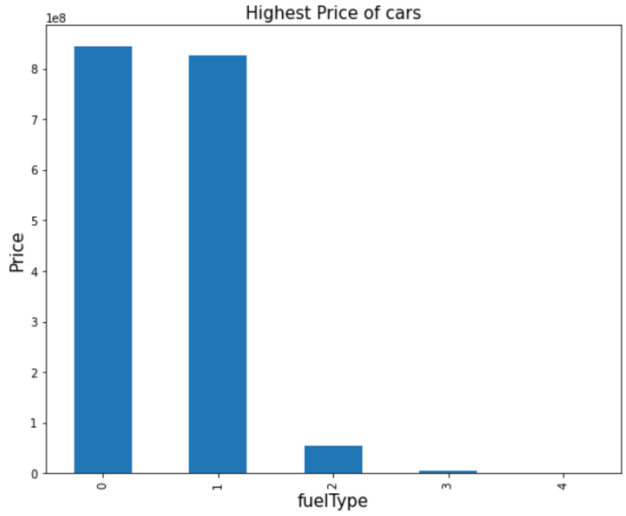

Fig-13

Insight for Price vs year: Total price is highest for year 2019 as the count for cars is high in 2019 In the

Fig 14-a shows the Sum of total price for all brands vs Year

Fig 14-b shows the Count of highest price in all brands vs Year

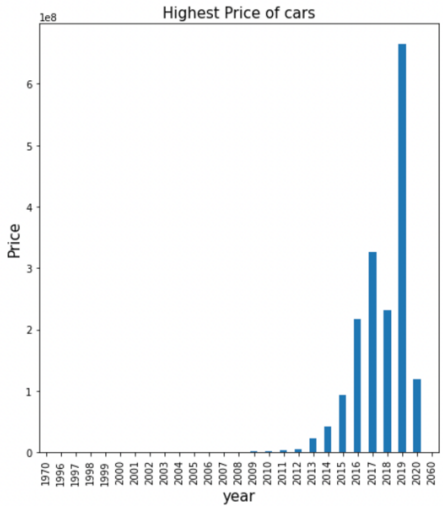

Fig = 14-a

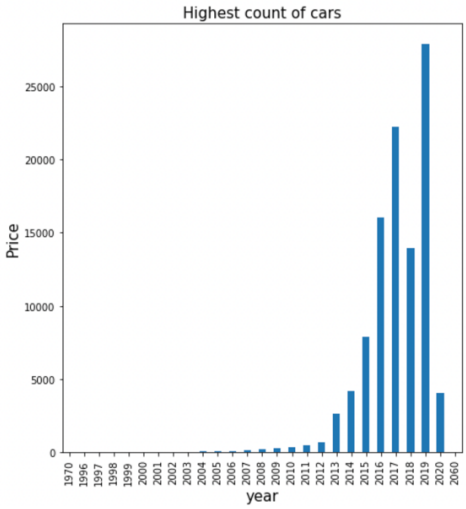

Fig = 14-b

Since our goal is price prediction, insights have been concentrated on price with respect to predictor variables rather than total count

VIII. Model selection and exploration:

Based on the visualization, correlation and heatmaps it has been decided not to eliminate the outliers. There are more than 4 numerical and categorical variables for the model to be built.
Model building have few steps:

1) Training
2) Validation
3) Testing

The given dataset has been split into 75% for training and 25% for validation. Using 75% of training dataset model has been built. The built model has been validated using rest 25% of dataset.
Two models are chosen for training and validation.
1)Linear Regression
2)K-NN Algorithm
3) Lasso
4) Bayesian
5) Decision tree
6) Random forest

Processes and Exploration involved:
1)Scaling is performed to maintain uniformity of various predictors and price to get accurate result
2)Uniformity in the data type of variables is necessary. Therefore, the transmission object type data type is converted into float to maintain necessary uniformity
3)It is observed that there is slight change due to the "other" in transmission therefore 7 records containing "other" are eliminate

Steps to build and execute for all models:
1)Predictors (X) and response (y) data frames are sliced from our dataset to find train and validating predictors and response variables
2)Our dataset consists of 101475 records after cleaning and preprocessing which is split into 75% and 25% using test split function
3)The train and valid X, y is scaled, fit, and transformed using standard scaler function imported from sklearn library
4)Scaled X and y are used to train and validate the data. The regression summary in linear regression module is used to find various statistics which determine the performance of module.
Example: RMSE, MSE
5)Where as KNN model performance is based on the k value (nearest neighbor value). The accuracy of k for a particular range of k determines the performance of the model
6)Necessary modules are imported from sklearn for respective models and the train and test datasets are fitted accordingly. Fitted predictors and outcomes are used to predict the outcomes which in turn used to calculate residuals and regression summaries

IX. Implementation of Selected models

Multiple Linear Regression:
It is a statistical method which is based on several predictors and an extension of linear (OLS) regression.

Training Result

	Predictor	coefficient
0	Brands-model	-0.090622
1	year	0.323148
2	transmission	0.166803
3	mileage	-0.213877
4	fuelType	-0.007154
5	mpg	-0.057723
6	engineSize	0.556149

Regression statistics

```
                        Mean Error (ME) : 0.0000
         Root Mean Squared Error (RMSE) : 0.5106
              Mean Absolute Error (MAE) : 0.3334
            Mean Percentage Error (MPE) : 26.6553
Mean Absolute Percentage Error (MAPE) : 188.4087
```

FIG - 15

Validation Result

	Predicted	Actual	Residual
3012	0.164989	0.095246	-0.069743
3227	0.837995	0.879615	0.041620
104	0.277510	0.545785	0.268275
3154	0.401836	0.238273	-0.163563
1607	-0.564404	-0.847029	-0.282625
2227	1.384726	1.005332	-0.379393
7252	-1.034300	-0.987728	0.046572
7118	0.961913	0.343240	-0.618673
16717	-0.738551	-0.684568	0.053983
11228	-0.693390	-0.775263	-0.081873
2758	0.952011	0.349009	-0.603002
4169	0.522964	-0.260954	-0.783918
1884	0.100053	0.043724	-0.056329
3344	-0.724675	-0.582941	0.141734
8136	-0.322070	-0.416836	-0.094766
3608	0.936747	1.511847	0.575100
3116	-0.294934	-0.070454	0.224479
1593	-1.357913	-1.119417	0.238495
7186	-0.981261	-0.756031	0.225230
9903	0.190796	0.226835	0.036039

Regression statistics

Mean Error (ME)	: 0.0053
Root Mean Squared Error (RMSE)	: 0.5214
Mean Absolute Error (MAE)	: 0.3375
Mean Percentage Error (MPE)	: 24.7412
Mean Absolute Percentage Error (MAPE)	: 173.1875

Fig - 16

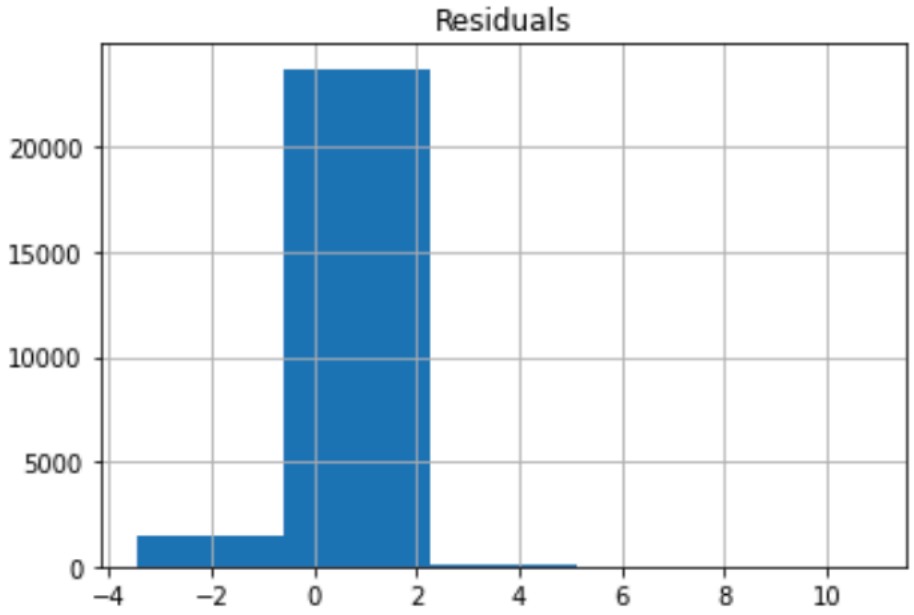

FIG – 17

K-NN:
It is a supervised algorithm which is used for both classification and prediction. The model based on the similarity of the nearest neighbors (predictor observations) of response variables

KNN Result

```
For k value: 100 MSE value is 0.1187735358302568
For k value: 101 MSE value is 0.1191028051527691
For k value: 102 MSE value is 0.11931064863447507
For k value: 103 MSE value is 0.1195590887340954
For k value: 104 MSE value is 0.11983565080054746
For k value: 105 MSE value is 0.12013074179740037
For k value: 106 MSE value is 0.12045293622759912
For k value: 107 MSE value is 0.12077466274777886
For k value: 108 MSE value is 0.12097104806447545
For k value: 109 MSE value is 0.12122946939503663
```

```
k=109 has the best accuracy
```

FIG – 18

Regression statistics for KNN

```
Knn test dataset

Regression statistics

                     Mean Error (ME) : 0.0000
      Root Mean Squared Error (RMSE) : 0.3482
           Mean Absolute Error (MAE) : 0.2039
         Mean Percentage Error (MPE) : 2.3284
Mean Absolute Percentage Error (MAPE) : 107.9834
```

Fig 19

Lasso Result

```
Regression statistics

                     Mean Error (ME) : 0.0106
      Root Mean Squared Error (RMSE) : 1.0201
           Mean Absolute Error (MAE) : 0.7276
         Mean Percentage Error (MPE) : 99.4549
Mean Absolute Percentage Error (MAPE) : 99.4657
```

Fig - 20

Bayesian Model

```
Regression statistics

                     Mean Error (ME) : 0.0053
      Root Mean Squared Error (RMSE) : 0.5214
           Mean Absolute Error (MAE) : 0.3375
         Mean Percentage Error (MPE) : 24.7426
Mean Absolute Percentage Error (MAPE) : 173.1840
Bayesian ridge chosen regularization:  4.8112383129707474e-05
```

Fig - 21

Decision Tree:

Decision tree is a supervised learning model which can be implemented in both classification and regression problems.
It uses tree representation to separate differently classified outcomes in classification problems and range of numerical outcomes using visual color representation

```
Regression statistics

                    Mean Error (ME) : -0.0000
          Root Mean Squared Error (RMSE) : 0.2647
               Mean Absolute Error (MAE) : 0.1564
             Mean Percentage Error (MPE) : 0.5242
Mean Absolute Percentage Error (MAPE) : 92.1736

Regression statistics

                    Mean Error (ME) : 0.0012
          Root Mean Squared Error (RMSE) : 0.2862
               Mean Absolute Error (MAE) : 0.1686
             Mean Percentage Error (MPE) : -1.1998
Mean Absolute Percentage Error (MAPE) : 93.8719
```

Fig - 22

Random Forest:

A random forest may search for the best predictor to split among a random subset of predictors.

Random forest is an ensemble of decision trees.

Each decision tree is trained on a bootstrap sample.

Final predictions is made considering predictions of all individual trees.

```
Regression statistics

                    Mean Error (ME) : -0.0008
          Root Mean Squared Error (RMSE) : 0.2043
               Mean Absolute Error (MAE) : 0.1250
             Mean Percentage Error (MPE) : -3.6372
Mean Absolute Percentage Error (MAPE) : 68.4797
```

Fig - 23

R^2 Summary for all the models

```
R-Square Error associated with linear regression is 0.7387150430658079
R-Square Error associated with lasso model is -0.00010754762869558121
R-Square Error associated with bayesian ridge is 0.7387135097606286
R-Square Error associated with Knn is 0.8834966223161251
R-Square Error associated with Decision tree regressor is 0.9213048693962476
R-Square Error associated with Random Forest regressor is 0.959892257870169
```

Fig - 24

X. Visualization for all the models:

Actual vs predicted

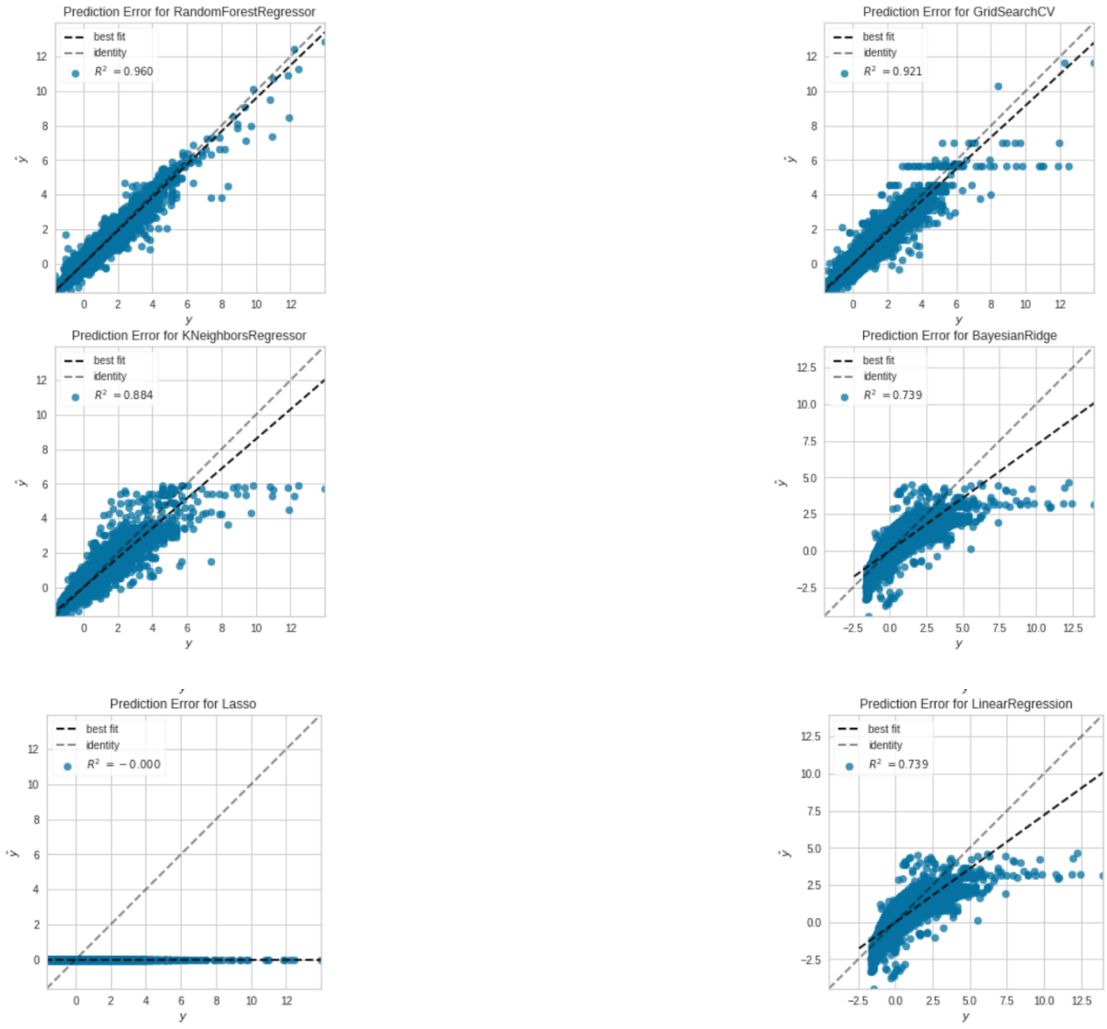

Fig - 25

Visualization Insight:
We can find that the plots are clumsy due to large number of records and similar residual values for various predictions however the model is accurate if the best fit line is identical to identity line. The best fit line is identical to identity line for Random Forest model

Distribution

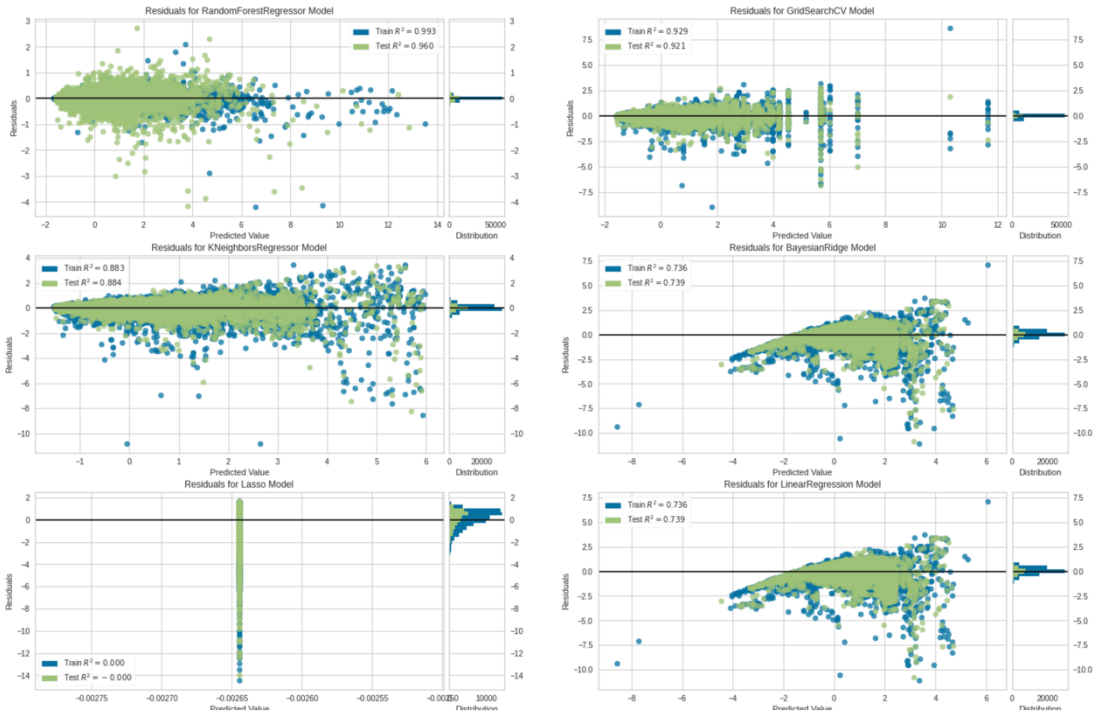

Fig - 26

Visualization Insight:
The residual error distribution is spread around zero more for random forest model than any other model

Visual representation in tabular form: -

	Linear_Regression	lasso	Bayesian	Knn	Decision_Tree	Random_Forest
ME	0.0053	0.0106	0.0053	0.0000	0.0012	-0.0008
RMSE	0.5214	1.0201	0.5214	0.3482	0.2860	0.2038
MAE	0.3375	0.7276	0.3375	0.2039	0.1685	0.1249
MPE	24.7412	99.4549	24.7412	2.3292	-1.2035	-3.5241
MAPE	173.1875	99.4657	173.1875	107.9830	93.8687	68.4430

Fig - 27

Result:

All the regression summary statistics are minimum for Random Forest regression model
Thus, making it the most suitable for regression model for this dataset.