



USED CAR PRICE PREDICTION

Submitted by:

ASHUTOSH MISHRA

ACKNOWLEDGMENT

I'd like to extend my gratitude to my mentor Mr. Md. Kashif for giving me this opportunity to work upon this project. Below are all the details of the project which I consumed while preparing and drafting the project –

The references, research papers, data sources, professionals, etc are majorly referred from the websites- **Carwale**, **Cartrade**, **Cardekho** and **Quikr**. Also, some of the other resources like “Research Gate” were explored to gain a deep understanding on the assigned subject.

Above stated details stands correct to the best of my knowledge and I hereby acknowledge the same.

INTRODUCTION

The used car market has accelerated since the pandemic. Customers are displaying a considerable preference towards used cars and the gap between new cars and used car sales is reducing phenomenally. As per the some of the study, the organized used car market share is expected to increase from 20 per cent in FY 2021-2022 to 45 per cent in FY 2026-2027.

Some of the key factors contributing to this massive growth in the next five years are a rising middle class and India's young population. This is also fuelled by the fact that India has seen a steady growth in disposable incomes over the years. The report adds that factors like technology-driven transparency, convenience, simplicity of transactions, etc. will also lead to this projected growth. The average car age has reduced by 33 per cent from six years in FY 2010-2011 to four years in FY 2021-2022.

A passenger car is a road motor vehicle, other than a moped or a motor cycle, intended for the carriage of passengers and designed to seat no more than nine persons (including the driver). The Indian passenger car market has been segmented by Vehicle Type and Fuel Type. Based on the Vehicle Type, the market is segmented into Hatchbacks, Sedans, SUVs, and MUVs. Based on the Fuel Type, the market is segmented into Petrol, Diesel, and Other Fuel Type.

The top five reasons why Indian consumers opt for a used vehicle are a need for mobility for personal and business growth, budget constraints and macroeconomic uncertainty, progressive industry players offering refurbished, certified, high-quality cars with warranties, digital & AI-led transformation increase convenience, trust and transparency and value for money nature of used cars compared to new cars. The growth is also expected to be driven by the emergence of organized online and phygital used car platforms amid a surge in the demand for personal mobility and because of favorable government support.

Business Problem Framing

With the covid 19 impact in the market, we have seen lot of changes in the car market. Now some cars are in demand hence making them costly and some are not in demand hence making them cheaper is the next big thing for the traders to deploy it correctly.

One of our clients works with small traders, who sell used cars. With the change in market due to covid 19 impact, our client is facing problems with their previous car price valuation machine learning models. So, they are looking for new machine learning models from new data so we have to make car price valuation model.

This Project will be done in two parts. In the very first part we've to do web scraping in-order to extract at least 5000 used car data from different websites like OLX, Cardekho, CarTrade, Quikr, Carwale and etcetera. We've to scrape for different attributes like Brand, model, variant, manufacturing year, driven kilometers, fuel, number of owners, location and at last our Target variable; which is Price of the car.

After collecting the data, we need to build a machine learning model and have to predict our target variable Car Price.

Conceptual Background of the Domain Problem

Data science comes as a vital tool to solve business problems to help companies optimize the standards resulting in overall revenue and profits growth. Moreover, it also improves their marketing strategies and demands focus on changing trends.

Predictive modelling, Market mix modelling, recommendation systems are some of the machine learning techniques globally used for achieving the business goals for any entity or an organization.

Currently the assigned project utilizes **Predictive Modelling** algorithm to solve the business statement.

Review of Literature

As, we know that after Covid pandemic most of the industries are trying to overcome their business challenges; so does our client is. This research project was for a client who is working with some small traders of used cars and our client is facing problems with their previous car price valuation machine learning models. So, they are looking for new machine learning models in-order to take their decision w.r.t. increase or decrease in demand and supply.

Being a Data Scientist, I have used the **Regression Model** using multiple algorithms to design and optimize the results as the target variable is in the form of continuous value.

Some of the classes which I explored from Scikit-learn Libraries were – Statistics, Analytical Modelling, Hyper Tuning Method, CV Ratings, Predictive Modelling and Regression Model, etc.

In totality, my project comprises of eight different test cases of Regression models where the objective was to train the model and test for accuracy score and different classification reports using best fitted regression line and RMSE error to understand the accuracy of the different algorithms.

By using Selenium Web-Scraping with the help of python IDE I've extracted out 5037 used car data from four different websites and they are- CarWale, CarTrade, CarDekho and Quikr. I've scraped for 9 attributes and they are- Brand, Car Name, Manufacturing Year, Driven-Kilometers, Fuel Type, Car Price, Location, Total Owner's, URL"s.

Also, I have observed that there are 5 numeric columns whereas 4 categorical columns are there in the dataset; there are also 716 null values are present in the dataset.

As a result of the above analytical modelling, I have managed to achieve below top model, which are given below as following–

Regression Models	Testing Accuracy (in %)
Random Forest Regressor	85

Motivation for the Problem Undertaken






See, being an intern I've not worked for any such projects where no data is provided to me. Here in this project I've myself collected the data with the help of selenium web scraping and then I did all the data pre-processing, data cleansing and finally built a regression model that could be useful while predicting the used car price. Also, one more thing I would like to add is, that I was least aware about the car industry market and different structures of car accessories like- SUV, Sedans, Coupe, minivan, Hatchback and etcetera so interestingly I did explore lots of insights about the same by doing this project.

Hence, this project guided me to baseline each aspect carefully and be concrete on the decision making process regardless it's an individual or an entity like a company.



In order to cater to the above project, my current knowledge and skill set has aided me a lot which I'd explore on this exponentially further.

Analytical Problem Framing

Mathematical Modeling- I have used the below statistical models to find-out the corresponding mean, median, mode and relationship among the variables.

-  **Descriptive Statistics-** To find out the mean, median, mode, percentiles and IQR (Interquartile Range)
-  Statistical Modeling, Correlation- To find out the relationship among the variables i.e. finding out the positive, negative and zero correlated attributes.
-  **Multicollinearity-** To find out all those variables who are giving similar information to the target variable ('Car Price')
-  **Skewness-** To check whether the attributes are the skewing left hand side or right hand side (Threshold value is=0.5).
-  **Outliers-** To find out variables those are having the value greater than 3.

Analytical Modeling-

-  **Label Encoder-** To convert all the categorical columns into numeric categories
-  **Simple Imputer-** To replace all the null values present in the columns with mean or mode.

Data Sources and their formats

I've used selenium web scraping over the four websites, Carwale, Cartrade, CarDekho and Quikr, in-order to find out the required attributes data. Once I've scraped out 5037 car used data from these websites I've merged all the four excel data's in a consolidated single excel file. Below are the attached screenshot for your reference; so that you can check out the insights that were available in the excel file.

ut[310]:

	Brand	Car_Name	Manufacturing_Year	Driven_Kilometers	Fuel_Type	Car_Price	Location	Total_Owner's	URL's
0	Honda	Honda City V MT	2020	38906	0	1003000.0	Ahmedabad	NaN	https://www.cardekho.com/buy-used-car-details/..
1	Maruti Suzuki	Maruti Baleno 1.2 Alpha	2017	38203	0	643000.0	Ahmedabad	1.0	https://www.cardekho.com/buy-used-car-details/..
2	Volkswagen	Volkswagen Polo 1.2 MPI Highline	2016	69414	0	515000.0	Ahmedabad	NaN	https://www.cardekho.com/buy-used-car-details/..
3	Honda	Honda City ZX CVT	2020	21261	0	1299000.0	Ahmedabad	NaN	https://www.cardekho.com/buy-used-car-details/..
4	Honda	Honda Jazz V CVT	2016	39821	0	605000.0	Ahmedabad	NaN	https://www.cardekho.com/buy-used-car-details/..
...
5032	Hyundai	Hyundai i20 Asta 1.4 CRDI 6 Speed	2018	69000	1	750000.0	Vijayawada	NaN	https://www.quikr.com/cars/used-grey-2018-hyun..
5033	Hyundai	Hyundai Grand i10 1.2 CRDI Sportz O	2017	73600	0	420000.0	Srinagar	NaN	https://www.quikr.com/cars/used-white-2017-hyu..
5034	Audi	Audi Q3 2.0 TDI S Edition	2014	66027	1	1089449.0	Gurgaon	1.0	https://www.quikr.com/cars/used-white-2014-aud..
5035	Audi	Audi Q3 2.0 TDI S Edition	2014	93752	1	1037699.0	Noida	3.0	https://www.quikr.com/cars/used-white-2014-aud..
5036	Hyundai	Hyundai Elantra 1.6 SX MT	2013	50000	1	425000.0	Kolkata	3.0	https://www.quikr.com/cars/used-grey-2013-hyun..

5037 rows × 9 columns

```
In [227]: #categorical Dataframe
df_categorical=df.select_dtypes(include='object')
df_categorical
```

```
Out[227]:
```

	Brand	Car_Name	Location	URL"s
0	Honda	Honda City V MT	Ahmedabad	https://www.cardekho.com/buy-used-car-details/...
1	Maruti Suzuki	Maruti Baleno 1.2 Alpha	Ahmedabad	https://www.cardekho.com/buy-used-car-details/...
2	Volkswagen	Volkswagen Polo 1.2 MPI Highline	Ahmedabad	https://www.cardekho.com/buy-used-car-details/...
3	Honda	Honda City ZX CVT	Ahmedabad	https://www.cardekho.com/buy-used-car-details/...
4	Honda	Honda Jazz V CVT	Ahmedabad	https://www.cardekho.com/buy-used-car-details/...
...
5032	Hyundai	Hyundai i20 Asta 1.4 CRDI 6 Speed	Vijayawada	https://www.quikr.com/cars/used-grey-2018-hyun...
5033	Hyundai	Hyundai Grand i10 1.2 CRDI Sportz O	Srinagar	https://www.quikr.com/cars/used-white-2017-hyu...
5034	Audi	Audi Q3 2.0 TDI S Edition	Gurgaon	https://www.quikr.com/cars/used-white-2014-aud...
5035	Audi	Audi Q3 2.0 TDI S Edition	Noida	https://www.quikr.com/cars/used-white-2014-aud...
5036	Hyundai	Hyundai Elantra 1.6 SX MT	Kolkata	https://www.quikr.com/cars/used-grey-2013-hyun...

5037 rows × 4 columns

```
In [218]: print('Row"s are',df.shape[0])
print('Columns are',df.shape[1])
print('Shape is',df.shape)
```

```
Row"s are 5037
Columns are 9
Shape is (5037, 9)
```

```
In [219]: #two dimensional dataframe
df.ndim
```

```
Out[219]: 2
```

```
In [220]: #Total datapoints in this dataframe
df.size
```

```
Out[220]: 45333
```

```
In [221]: #indexes are-
df.index
```

```
Out[221]: RangeIndex(start=0, stop=5037, step=1)
```

```
In [222]: #columns of the dataframes are-
df.columns
```

```
Out[222]: Index(['Brand', 'Car_Name', 'Manufacturing_Year', 'Driven_Kilometers',
                'Fuel_Type', 'Car_Price', 'Location', 'Total_Owner's', 'URL"s'],
                dtype='object')
```

In [223]: `#It shows top 5 Rows
df.head(5)`

Out[223]:

	Brand	Car_Name	Manufacturing_Year	Driven_Kilometers	Fuel_Type	Car_Price	Location	Total_Owner's	URL"s
0	Honda	Honda City V MT	2020	38906	0	1003000.0	Ahmedabad	NaN	https://www.cardekho.com/buy-used-car-details/...
1	Maruti Suzuki	Maruti Baleno 1.2 Alpha	2017	38203	0	643000.0	Ahmedabad	1.0	https://www.cardekho.com/buy-used-car-details/...
2	Volkswagen	Volkswagen Polo 1.2 MPI Highline	2016	69414	0	515000.0	Ahmedabad	NaN	https://www.cardekho.com/buy-used-car-details/...
3	Honda	Honda City ZX CVT	2020	21261	0	1299000.0	Ahmedabad	NaN	https://www.cardekho.com/buy-used-car-details/...
4	Honda	Honda Jazz V CVT	2016	39821	0	605000.0	Ahmedabad	NaN	https://www.cardekho.com/buy-used-car-details/...

In [224]: `#It shows bottom 7 Rows
df.tail(7)`

Out[224]:

	Brand	Car_Name	Manufacturing_Year	Driven_Kilometers	Fuel_Type	Car_Price	Location	Total_Owner's	URL"s
5030	Mahindra	Mahindra XUV 300 W8	2022	6800	0	1305000.0	Hyderabad	NaN	https://www.quikr.com/cars/used-2022-mahindra-...
5031	Hyundai	Hyundai i10 Magna 1.1 iRDE2	2008	85000	0	125000.0	Jamshedpur	NaN	https://www.quikr.com/cars/used-white-2008-hyu...
5032	Hyundai	Hyundai i20 Asta 1.4 CRDI 6 Speed	2018	69000	1	750000.0	Vijayawada	NaN	https://www.quikr.com/cars/used-grey-2018-hyun...
5033	Hyundai	Hyundai Grand i10 1.2 CRDI Sportz O	2017	73600	0	420000.0	Srinagar	NaN	https://www.quikr.com/cars/used-white-2017-hyu...

[225]: `#It shows any 2 random Rows
df.sample(2)`

Out[225]:

	Brand	Car_Name	Manufacturing_Year	Driven_Kilometers	Fuel_Type	Car_Price	Location	Total_Owner's	URL"s
3699	Maruti Suzuki	Maruti Suzuki Dzire Vxi	2018	12826	0	696599.0	Pune	2.0	https://www.quikr.com/cars/used-other-2018-mar...
3822	Renault	Renault Triber	2019	58887	0	681399.0	Chennai	2.0	https://www.quikr.com/cars/used-other-2019-ren...

[226]: `#numeric Dataframe
df_numeric=df.select_dtypes(exclude='object')
df_numeric`

Out[226]:

	Manufacturing_Year	Driven_Kilometers	Fuel_Type	Car_Price	Total_Owner's
0	2020	38906	0	1003000.0	NaN
1	2017	38203	0	643000.0	1.0
2	2016	69414	0	515000.0	NaN
3	2020	21261	0	1299000.0	NaN
4	2016	39821	0	605000.0	NaN
...
5032	2018	69000	1	750000.0	NaN
5033	2017	73600	0	420000.0	NaN
5034	2014	66027	1	1089449.0	1.0
5035	2014	93752	1	1037699.0	3.0
5036	2013	50000	1	425000.0	3.0

5037 rows × 5 columns

```
In [230]: categorical_columns=selector(dtype_include=object)(df)
print(categorical_columns)
print('\n Total string categorical columns are ',len(categorical_columns))

['Brand', 'Car_Name', 'Location', 'URL"s']

Total string categorical columns are 4
```

```
In [231]: df.dtypes

Out[231]: Brand                object
Car_Name                object
Manufacturing_Year      int64
Driven_Kilometers       int64
Fuel_Type               int64
Car_Price               float64
Location                object
Total_Owner's           float64
URL"s                  object
dtype: object
```

```
In [233]: df.isnull().any().any()
```


```
Out[233]: True
```

```
In [234]: df.isnull().sum()
```

```
Out[234]: Brand                7
Car_Name                0
Manufacturing_Year      0
Driven_Kilometers       0
Fuel_Type               0
Car_Price               8
Location                79
Total_Owner's           622
URL"s                  0
dtype: int64
```

```
In [235]: df.isnull().sum().sum()
```

```
Out[235]: 716
```

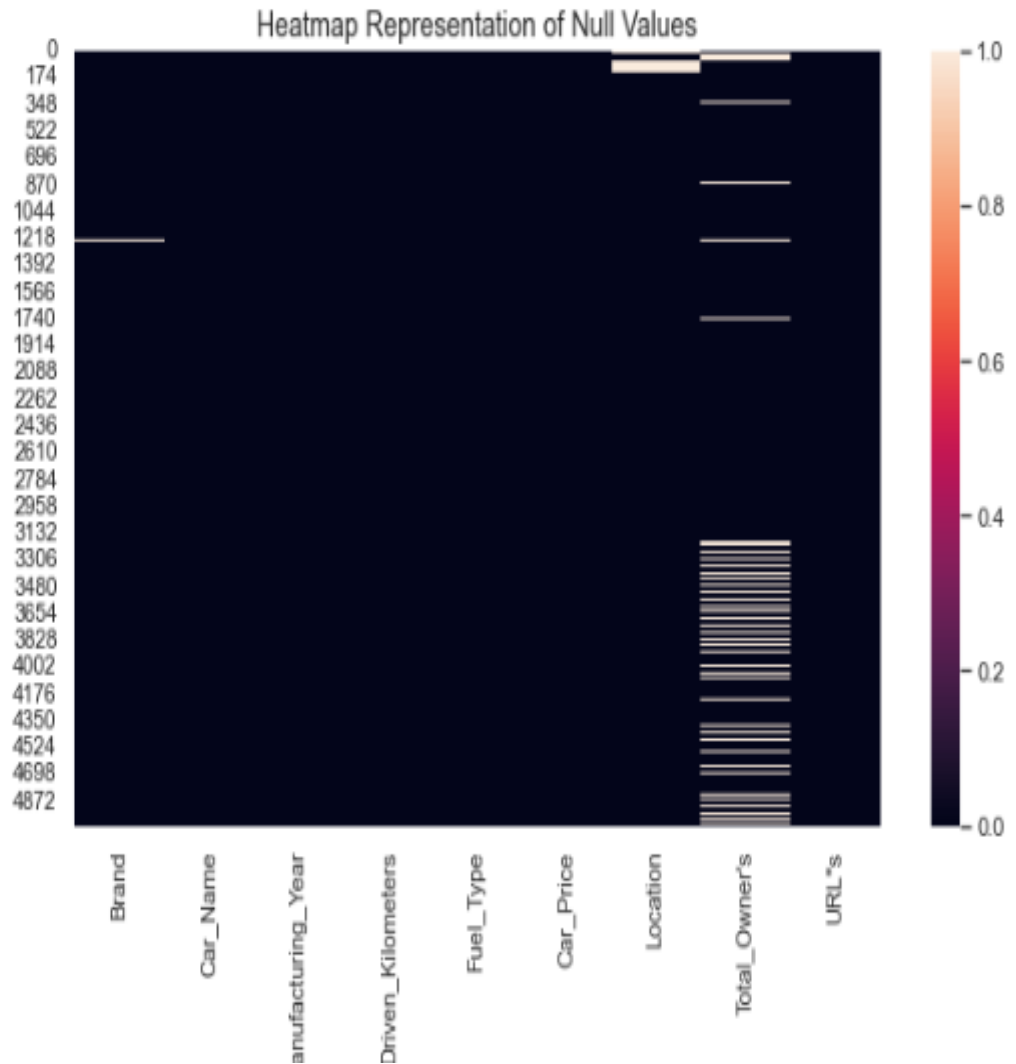
```
In [237]:  ##checking out the uniqueness of the columns  
df.nunique()
```

```
Out[237]: Brand          39  
Car_Name        1639  
Manufacturing_Year    23  
Driven_Kilometers  2266  
Fuel_Type         7  
Car_Price        1814  
Location         120  
Total_Owner's      5  
URL"s           3630  
dtype: int64
```

```
In [311]:  df['Location'].value_counts()
```

```
Out[311]: Mumbai        764  
Chennai        742  
Bangalore      732  
Pune           686  
Hyderabad      413  
...  
Rudrapur        1  
Buldhana        1  
Surat           1  
Indore          1  
Udaipur         1  
Name: Location, Length: 120, dtype: int64
```

```
In [236]: plt.figure(figsize=(10,6))
sns.heatmap(df.isnull())
plt.title('Heatmap Representation of Null Values',fontsize=14)
plt.show()
```



Data Preprocessing Done

See, there are three attributes having negative values and its showing that there is negative correlation b/w Car Price (which is our target variable) and other respective Negative attributes. Hence will remove, all the negative correlated columns which are very close to zero, later in data cleaning phase. Negative correlation means if input is +ve then output would be -ve and vice-versa whereas, Positive correlation means if input is +ve then output would also be +ve and vice-versa. Also we'll not remove any categorical columns and target variable in this process.

Also, I've used Label Encoder method to convert all the categorical variables value into numeric form and simple Imputer to replace null values of categorical and numeric columns w.r.t. mode and mean/median respectively.

The threshold value of Skewness is ± 0.5 . Attributes- Brand, Car Name, URL's, Location are only in the range while others are either skewed right or left hence will remove skewness from these columns later in data cleansing part, also will not touch target column and categorical columns for removing the skewness.

The two pairs are showing high Multicollinearity among others and they are- (Brand, Car Name) and (URL's, Total Owner's). Since, (Brand and Car Name) have 95% collinearity and (URL's, Total Owner's) are having 45% of collinearity which is still less than 50% hence will remove only the attribute 'Car Name' from our dataset.

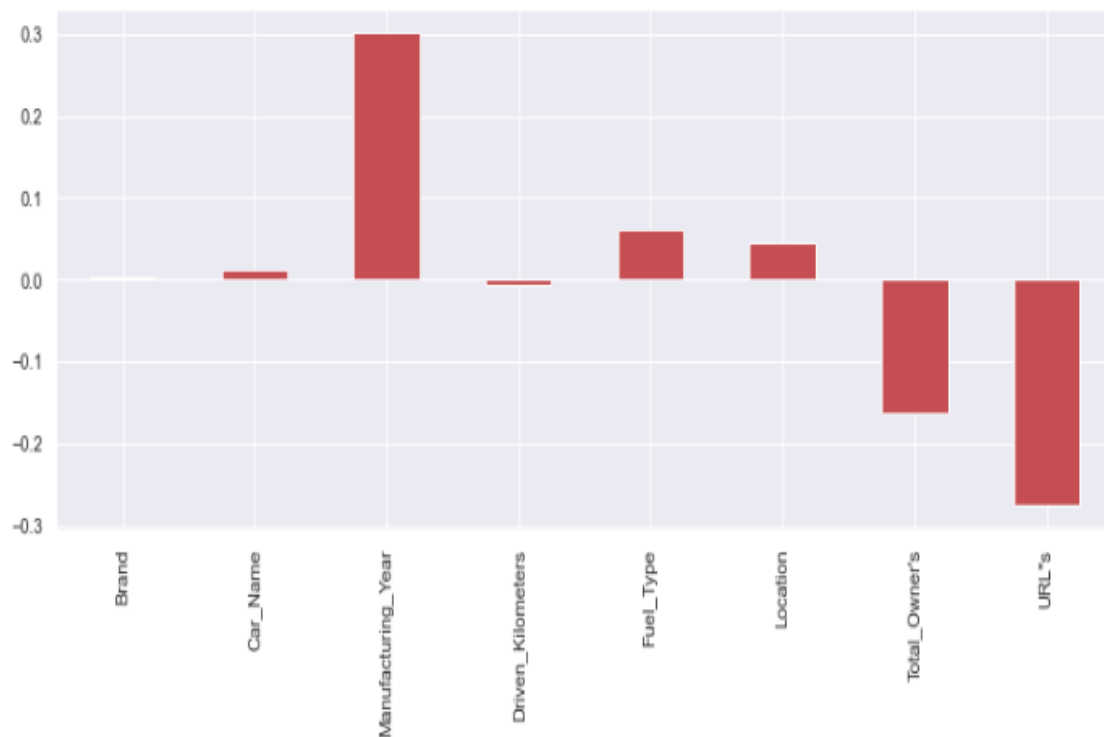
Since data is expensive and we can't lose more than 7-8% of the data but here after removing outliers I'm losing only 6% of the data and which is bearable hence will consider the outlier removal. I've used Power Transformer and Square-Root method to remove the skewness; while Standard Scaler technique to normalize the feature variable and later on I've used Principle component analysis and converted all the variables into six principle components which was seven earlier in the process.

Data Inputs- Logic- Output Relationships

Since, as mentioned in our problem statement we've to create used car price valuation model. So on the basis of output I have checked the correlational input data with my output, as shown in below figure.

Multicollinearity

```
[251]: ▶ plt.figure(figsize=(13,5))
df.corr()['Car_Price'].drop(['Car_Price']).plot(kind='bar',color='r')
plt.show()
```



As I checked the input and found that almost all the data is quite positively correlated with my output data, except few columns data. Attribute URL's and 'Total Owner' are highly negatively correlated with my output.

State the set of assumptions (if any) related to the problem under consideration

No.

Hardware and Software Requirements and Tools Used

I have used **Python IDE** (Integrated Development environment) as a dedicated software throughout solving this project.

```
import numpy as np
import pandas as pd
import scipy.stats
from scipy.stats import zscore, boxcox
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

Python Libraries that I've used throughout the process are-

- ✚ Numpy- It is use for linear algebra
- ✚ Pandas- data analysis/manipulation library o
- ✚ Scipy- Utility function for optimization o
- ✚ Matplotlib-Data visualization and plotting library o
- ✚ Seaborn- Data visualization and statistical plotting library
- ✚ Sklearn- Machine Learning Tool
- ✚ Imblearn- Deal with classification problems of Imbalanced classes o
- ✚ Statsmodels-Deal with advanced statistics

Classes-

- ✚ Label Encoder- Encoding the categorical variables into number category
- ✚ Simple Imputer- Replacing the null values with mean, median or mode
- ✚ Variance_Inflation_Factor- Calculate Multicollinearity
- ✚ Power Transformer- Remove skewness
- ✚ Standard Scaler- Normalize the feature variables
- ✚ Principle Component Analysis- Reduce the dimension of the data frame
- ✚ Cross_Val_Score- CV score
- ✚ Grid Search CV- Find out the best parameters for the model

Model/s Development and Evaluation

Identification of possible problem-solving approaches (methods)

Statistical Method-

When I used the describe function then I find out that some of the attributes has more median than its mean and so it indicates that there is the possibility of left skewed data in the dataset and the interquartile range for the variables **Driven Kilometers** and **URL"s** are varying too much hence it shows that datasets are skewing left hand side and it indicates that the variables are not normally distributed.

Also, I have used correlation method to check what are the variables that are giving strong correlation w.r.t Target variable Car Price.

Analytical Method-

I've uses Boxplot, Scatter Plots and Distribution Plots to check the outliers and skewness of the variables respectively through the plotting's.

Testing of Identified Approaches (Algorithms)

Listing down all the algorithms used for the training and testing-

```
from sklearn.linear_model import LinearRegression

from sklearn.model_selection import train_test_split

from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score

from sklearn.linear_model import Lasso, Ridge, ElasticNet, SGDRegressor

from sklearn.model_selection import cross_val_score

from sklearn.model_selection import GridSearchCV

from sklearn.ensemble import

    RandomForestRegressor, AdaBoostRegressor, GradientBoostingRegressor

lr=LinearRegression()

ls=Lasso()

rd=Ridge()

en=ElasticNet()

sgd=SGDRegressor()

rf=RandomForestRegressor()

ad=AdaBoostRegressor()

grd=GradientBoostingRegressor()
```

For Saving the Library-

```
import joblib

file='Car_Price.obj'

joblib.dump(rf,file)
```

Run and Evaluate selected models

Best Model (Random Forest Regressor)

In [299]: `model(rf,x,y)`

Training Accuracy of model RandomForestRegressor(criterion='poisson', max_features='sqrt') is 0.935732944130608
Testing Accuracy of model RandomForestRegressor(criterion='poisson', max_features='sqrt') is 0.49086675412524783

Error in the model is calculated below-

The Mean Absolute Error is (MAE) 459965.00224071904
The Mean Squared Error is (MSE) 847040056975.8162
The Root Mean Squared Error is (RMSE) 920347.7913135969

Finding out the best K-Fold Value

At K-Fold 2 the CV Score of model RandomForestRegressor(criterion='poisson', max_features='sqrt') is -16.789953159792653 & std is 0.250990946327911

At K-Fold 3 the CV Score of model RandomForestRegressor(criterion='poisson', max_features='sqrt') is -52.273939942108605 & std is 0.5881007404690834

At K-Fold 4 the CV Score of model RandomForestRegressor(criterion='poisson', max_features='sqrt') is 19.274218059733077 & std is 0.4082792754179593



Key Metrics for success in solving problem under consideration

Below is the best **Regression Models** where I used below metrics -

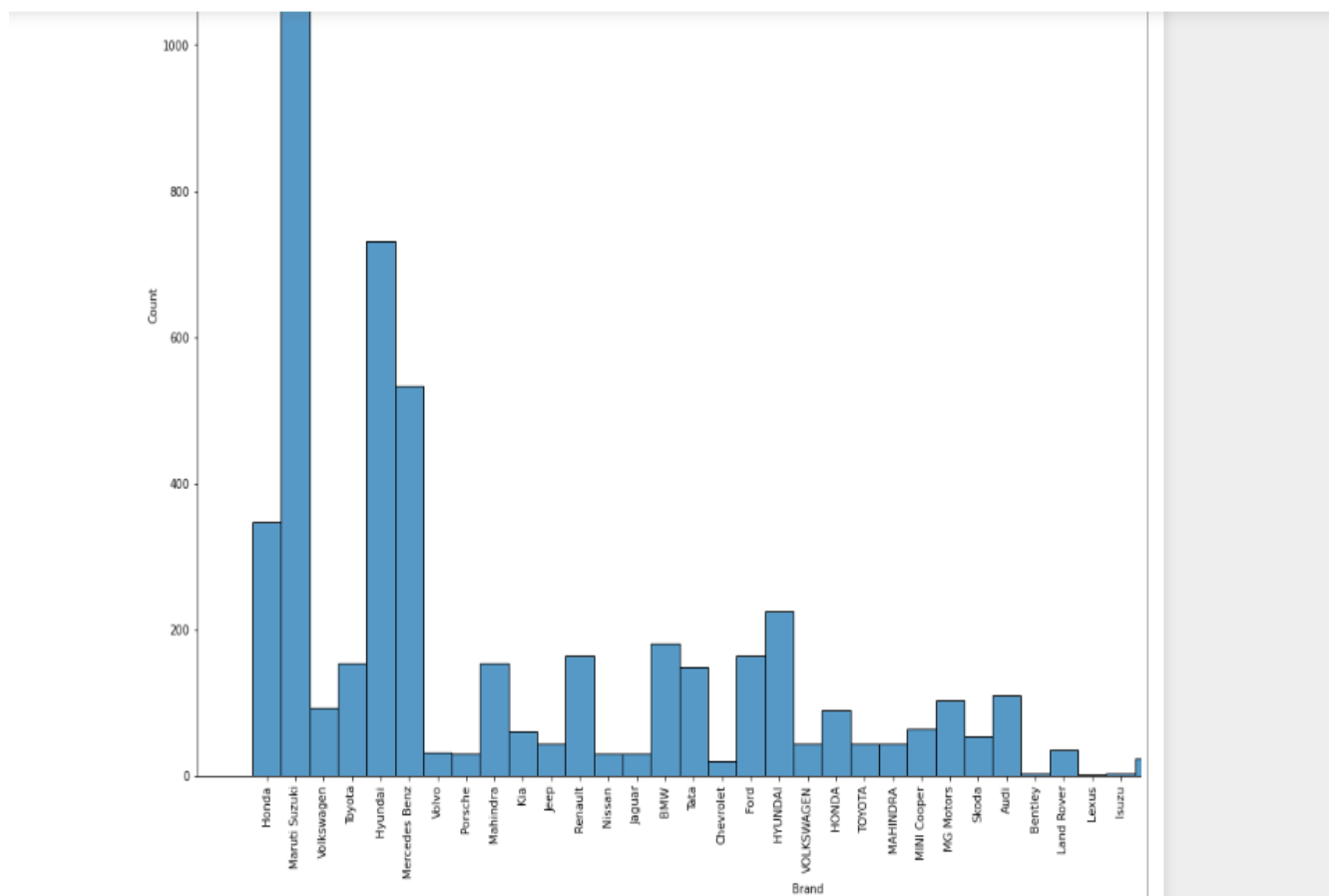
Regression Models	Testing Accuracy (in%)
Random Forest Regressor	85

- ✚ CV Score – Model testing Accuracy
- ✚ Hyper Tuning Method – Best parameter for the respective models
- ✚ Principle Component Analysis – Course of reduction of dimensions, etc

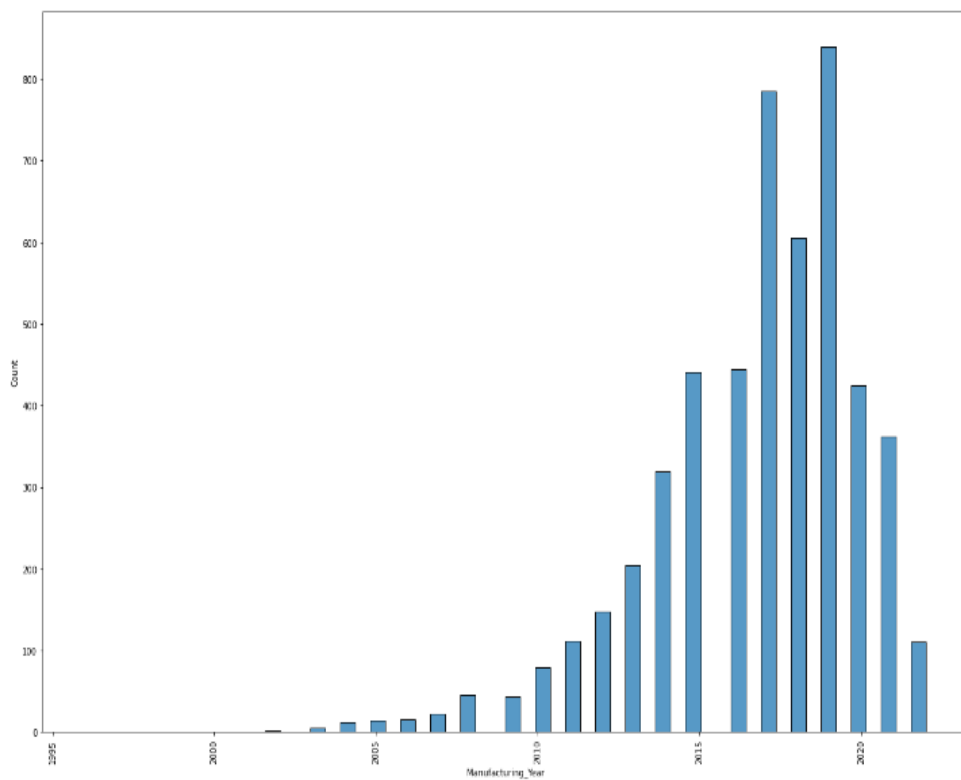
Visualizations

The plots are –

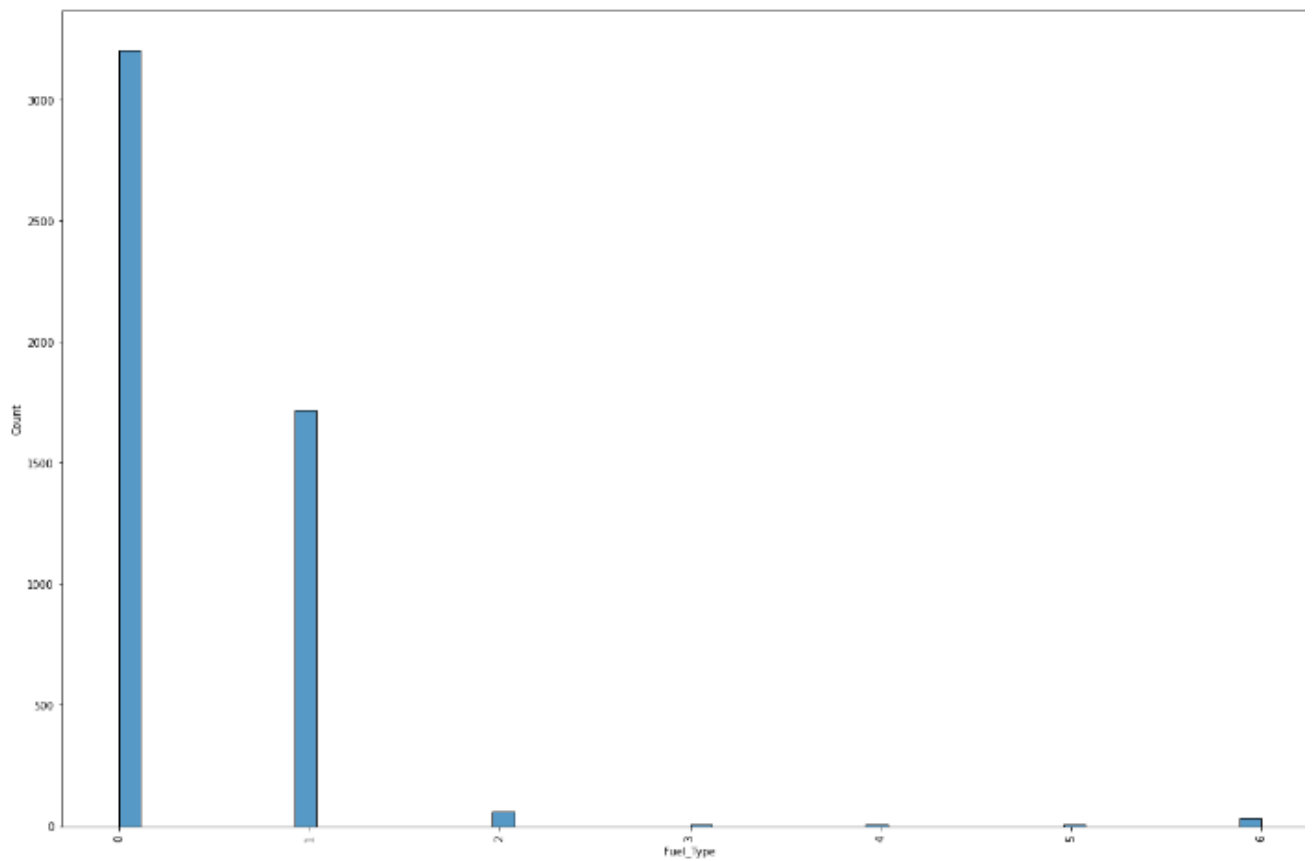
- Histograms
- Pearson Correlation Heat-map
- Scatter Plot
- Distribution plot, box plot, Violin Plot etc.



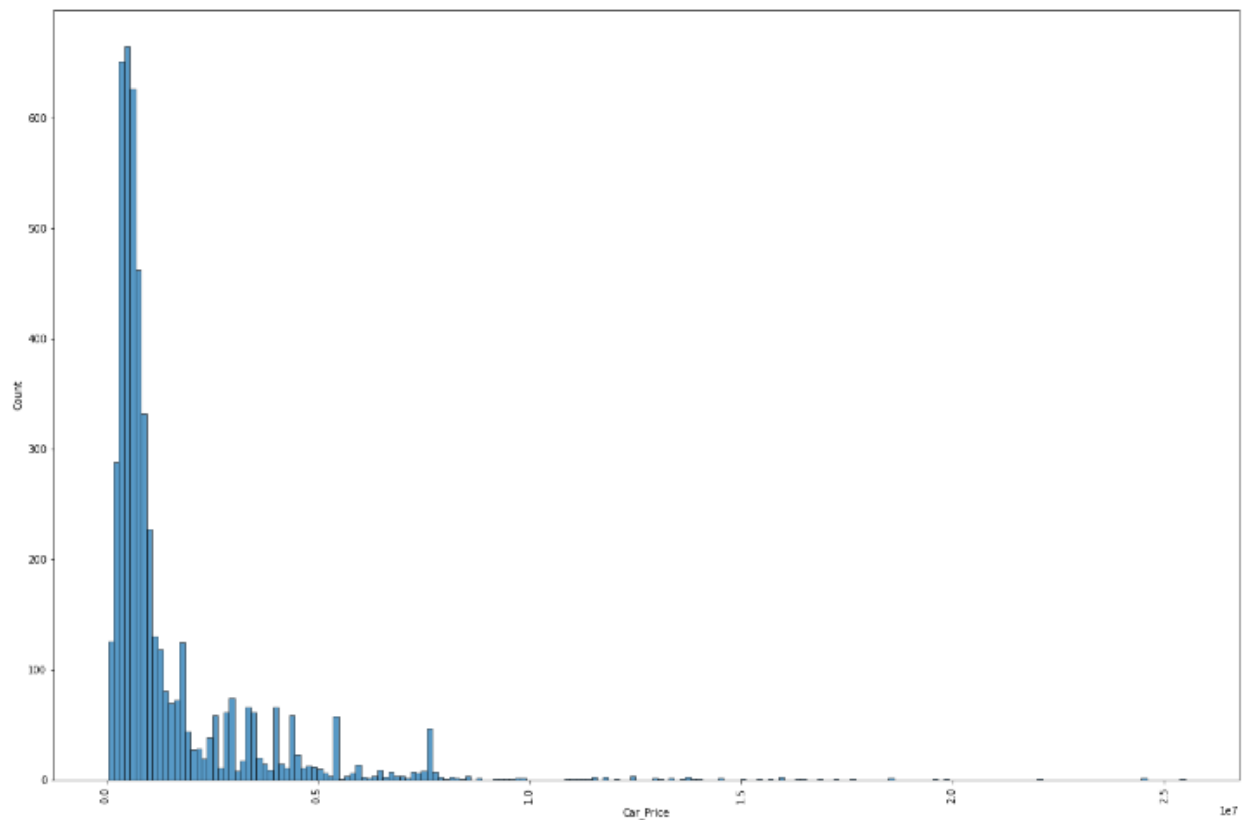
The Histtplot Diagram for the attribute "Manufacturing_Year" is
AxesSubplot(0.125,0.125;0.775x0.755)



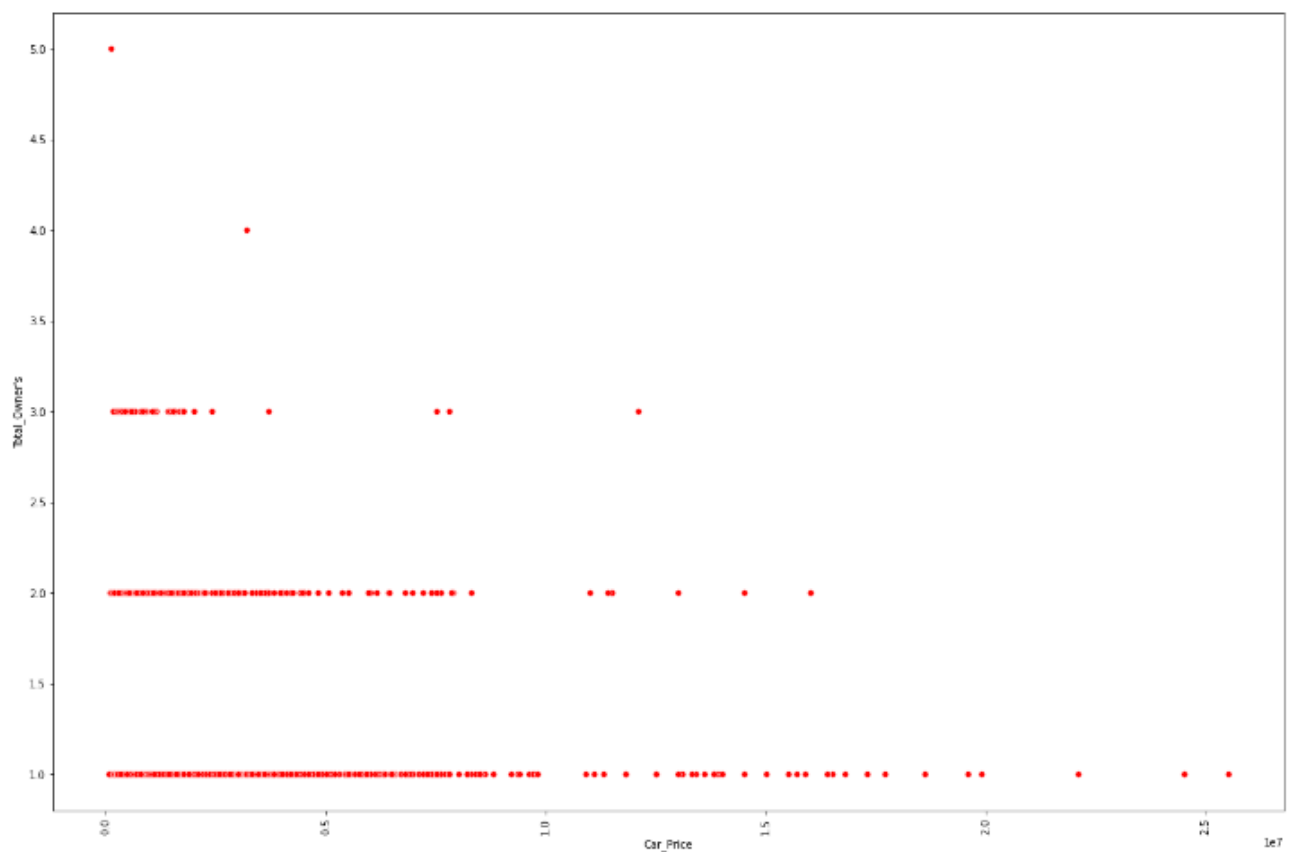
The Histtplot Diagram for the attribute "Fuel_Type" is
AxesSubplot(0.125,0.125;0.775x0.755)



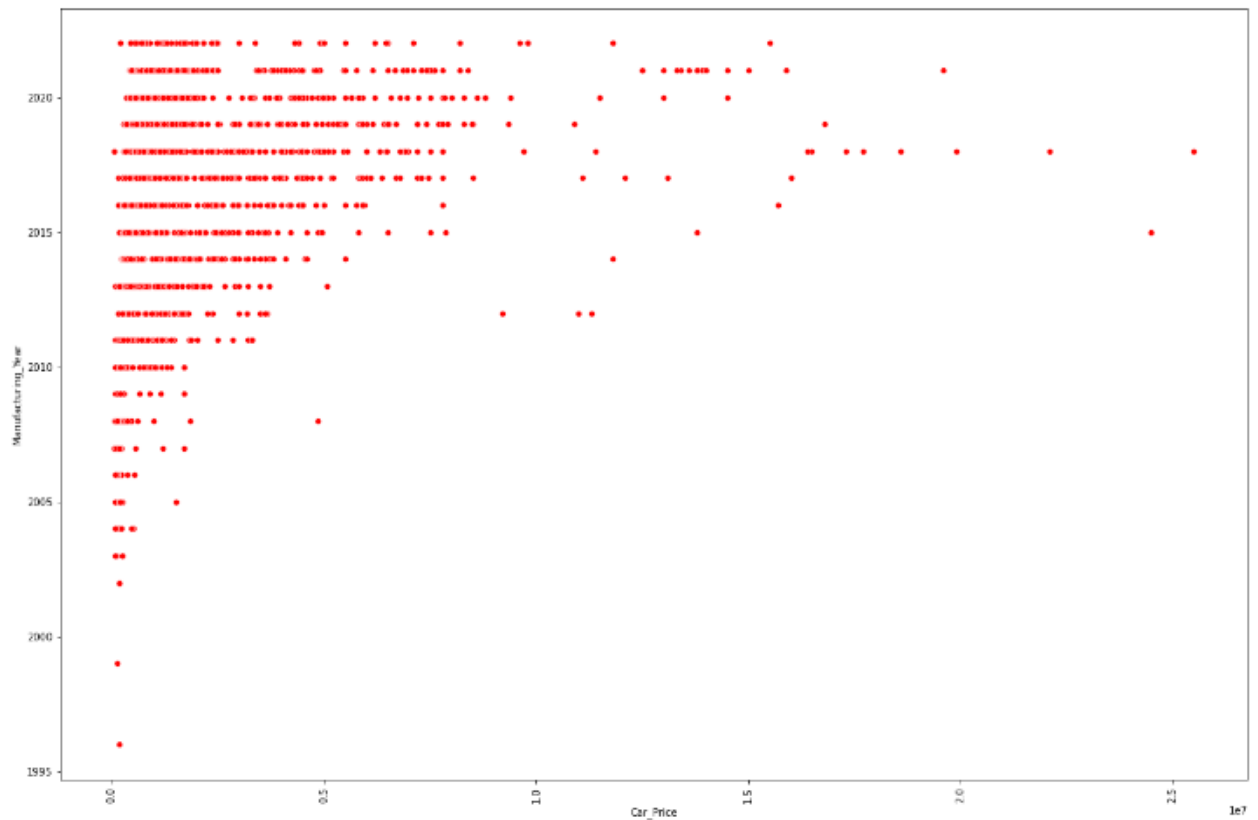
The Histtplot Diagram for the attribute "Car_Price" is
 AxesSubplot(0.125,0.125;0.775x0.755)



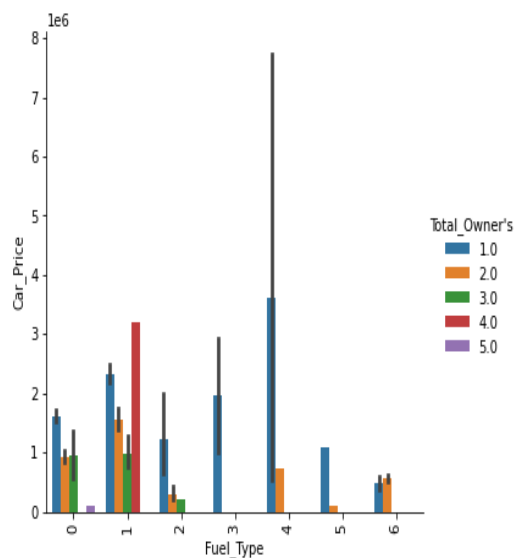
The Scatter Plot for the attribute "Car_Price" & "Total_Owner's" is-
 AxesSubplot(0.125,0.125;0.775x0.755)



The Scatter Plot for the attribute "Car_Price" & "Manufacturing_Year" is-
 AxesSubplot(0.125,0.125;0.775x0.755)

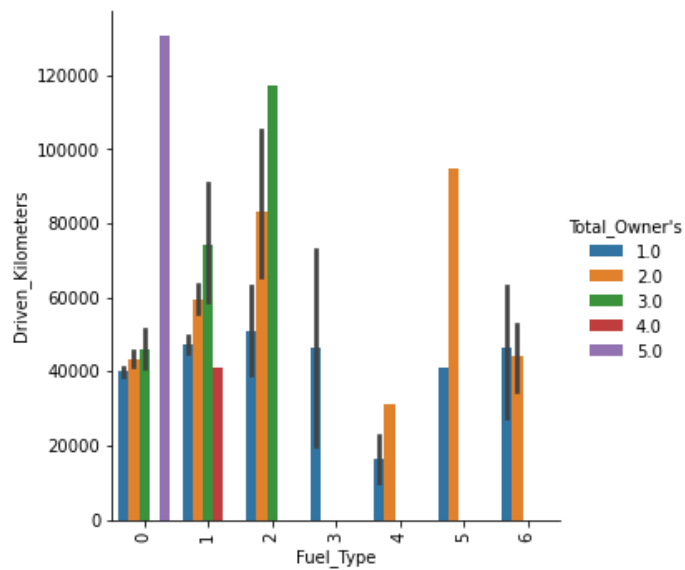


```
In [65]: sns.catplot(x='Fuel_Type',y='Car_Price',hue="Total_Owner's",data=df,kind='bar')
plt.xticks(rotation=90)
plt.show()
```



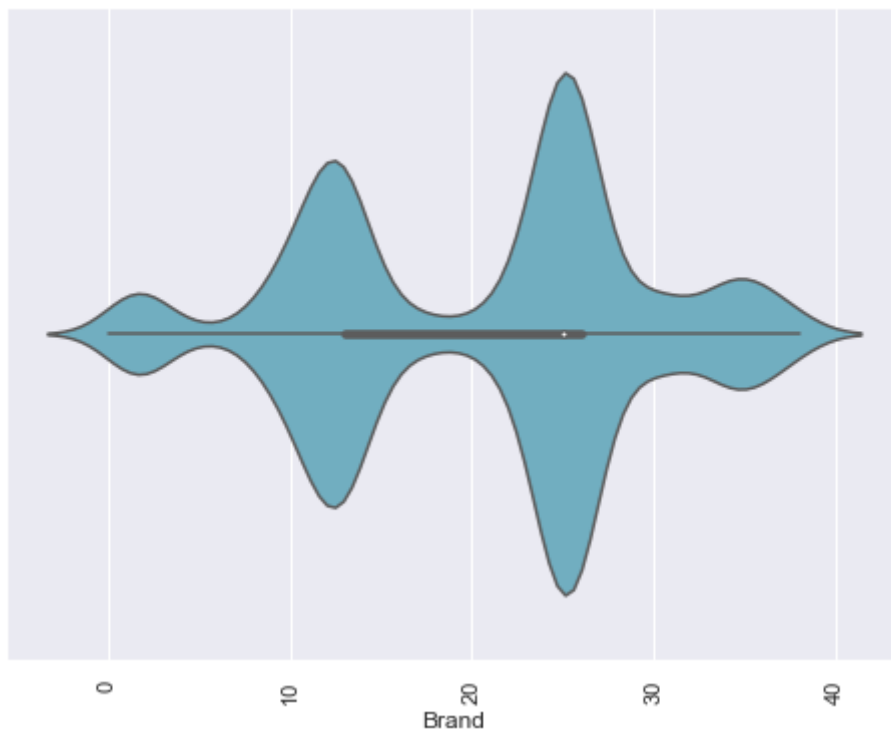
The used cars those have fuel type as Hybrid i.e. diesel/Petrol both are having maximum 2 owners only and the price is more than 30 Lakhs. Used car having fuel type Diesel and having 4 owners have price more than 30 Lakhs. the used car having fuel type Petrol+CNG have less car price and haing atleast two owners maximum.

```
In [66]: sns.catplot(x='Fuel_Type',y='Driven_Kilometers',hue="Total_Owner's",data=df,kind='bar')
plt.xticks(rotation=90)
plt.show()
```

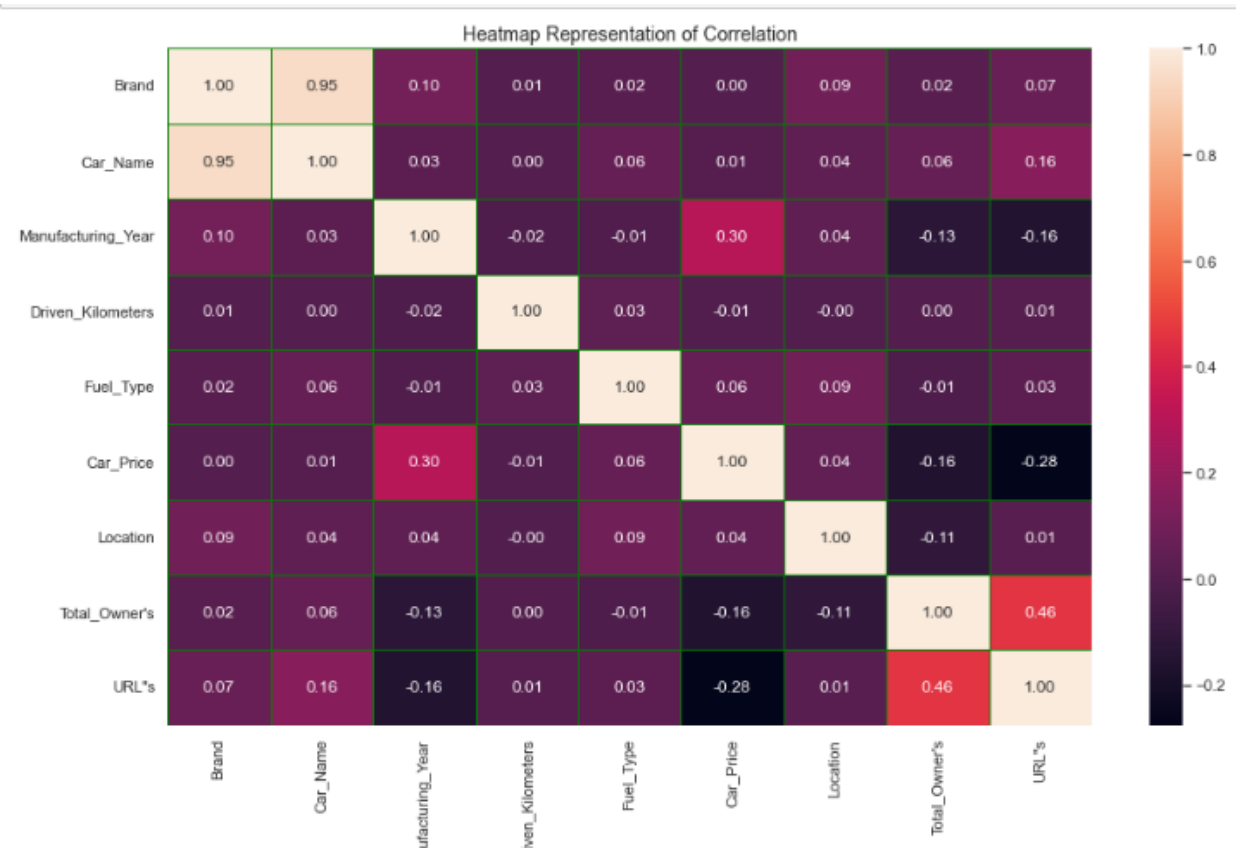
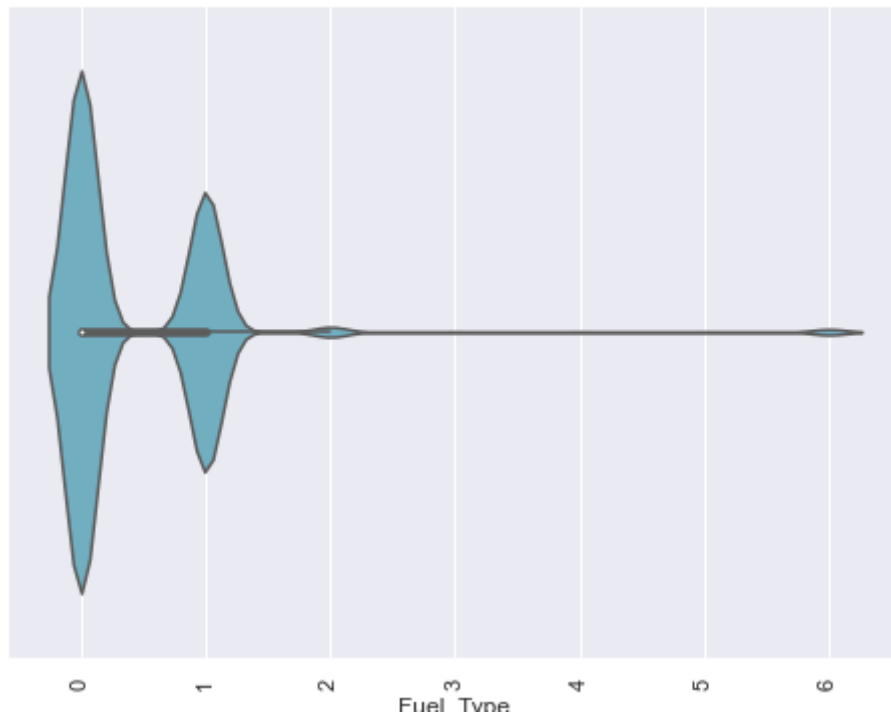


The used cars having fuel type as Petrol are already droven more than 1.2L kms.

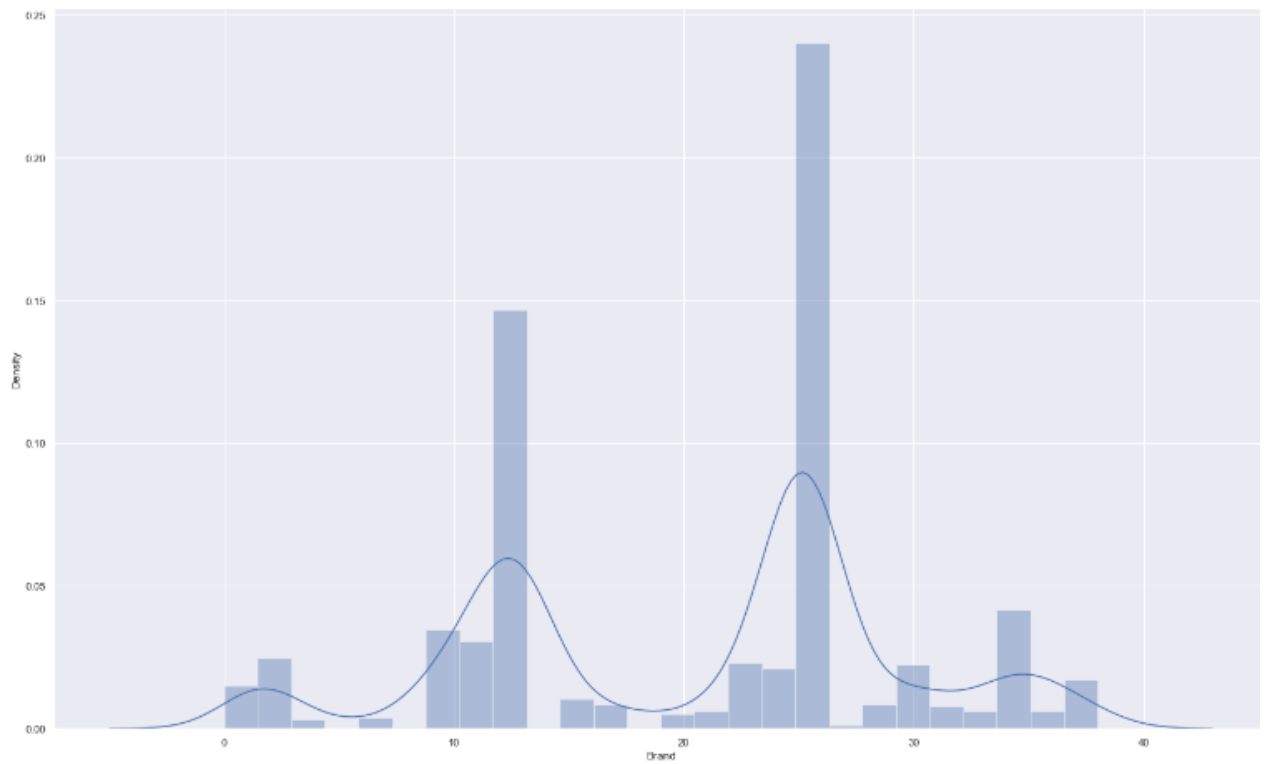
The Violin-Plot for the attribute "Brand" is-
 AxesSubplot(0.125,0.125;0.775x0.755)



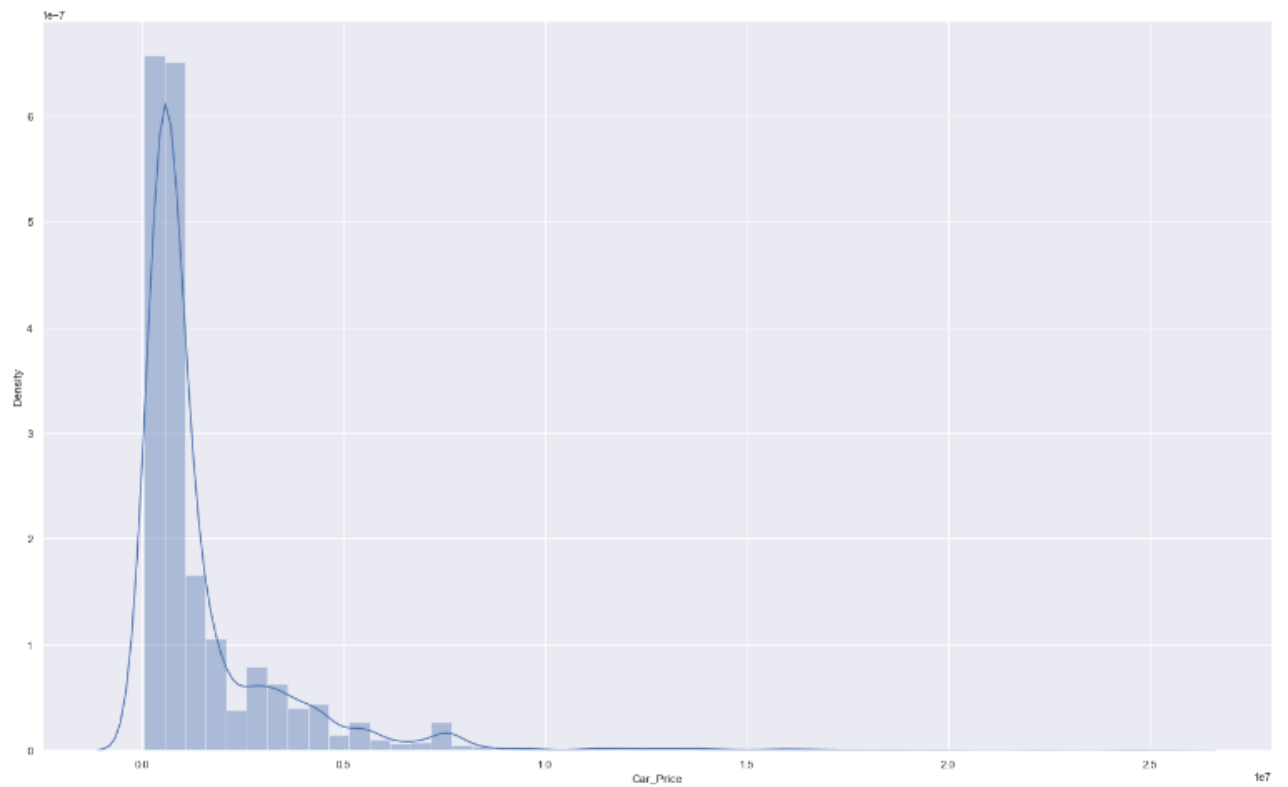
The Violin-Plot for the attribute "Fuel_Type" is-
 AxesSubplot(0.125,0.125;0.775x0.755)



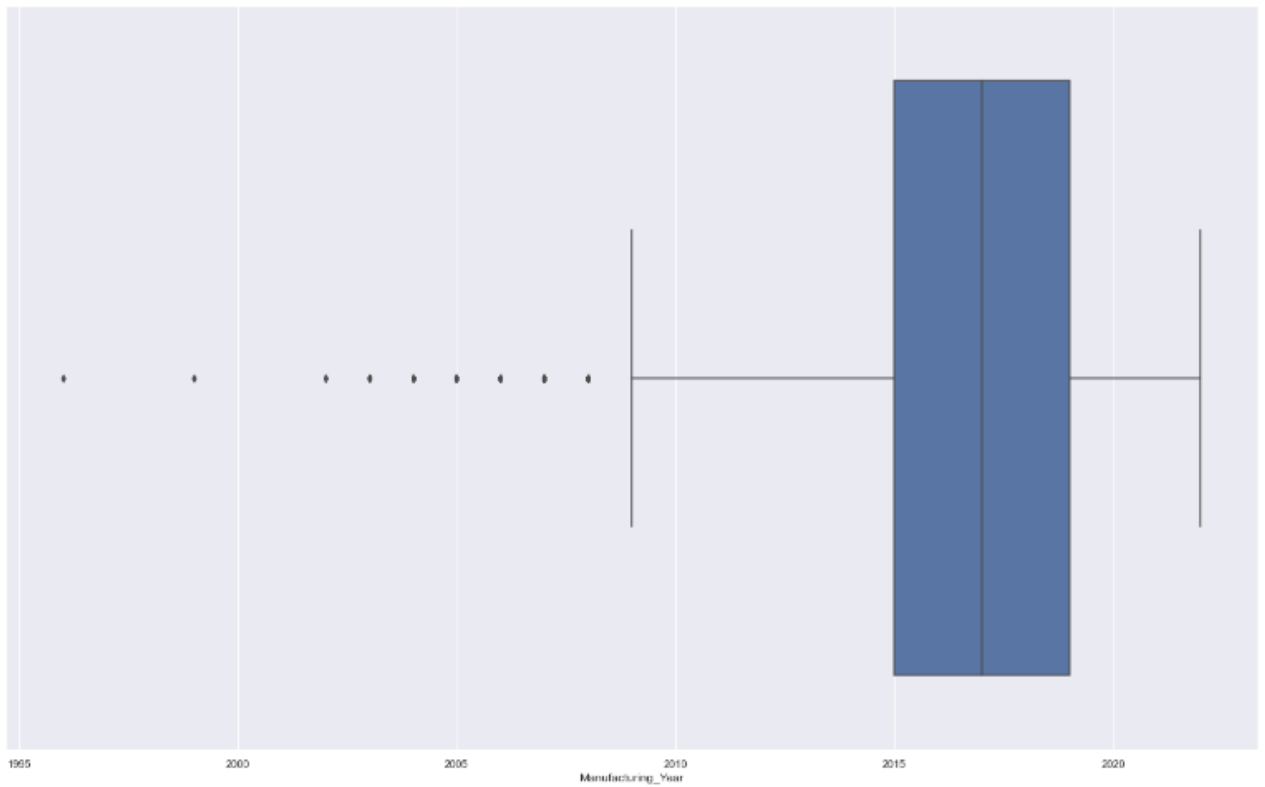
The Distribution Plot for attribute "Brand" is-
AxesSubplot(0.125,0.125;0.775x0.755)



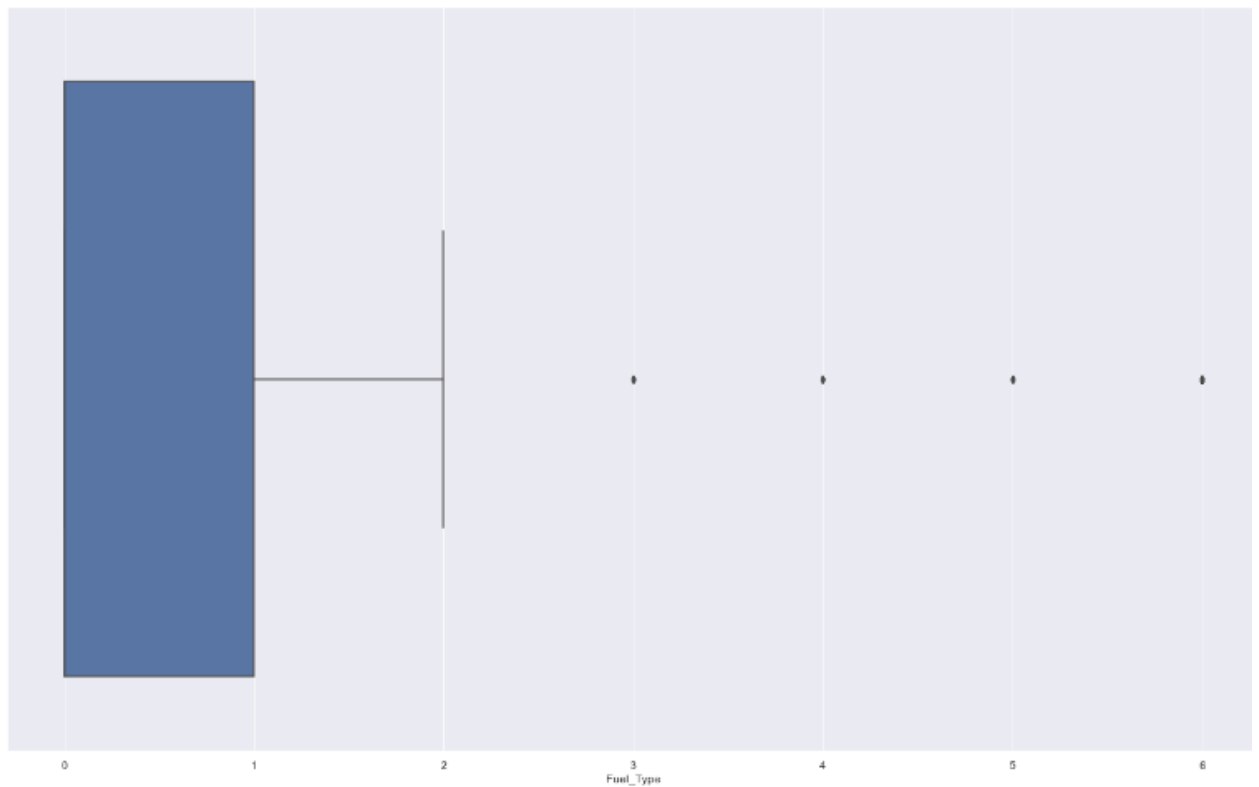
The Distribution Plot for attribute "Car_Price" is-
AxesSubplot(0.125,0.125;0.775x0.755)



The Box-Plot for attribute "Manufacturing_Year" is-
AxesSubplot(0.125,0.125;0.775x0.755)



The Box-Plot for attribute "Fuel_Type" is-
AxesSubplot(0.125,0.125;0.775x0.755)



Interpretation of the Results

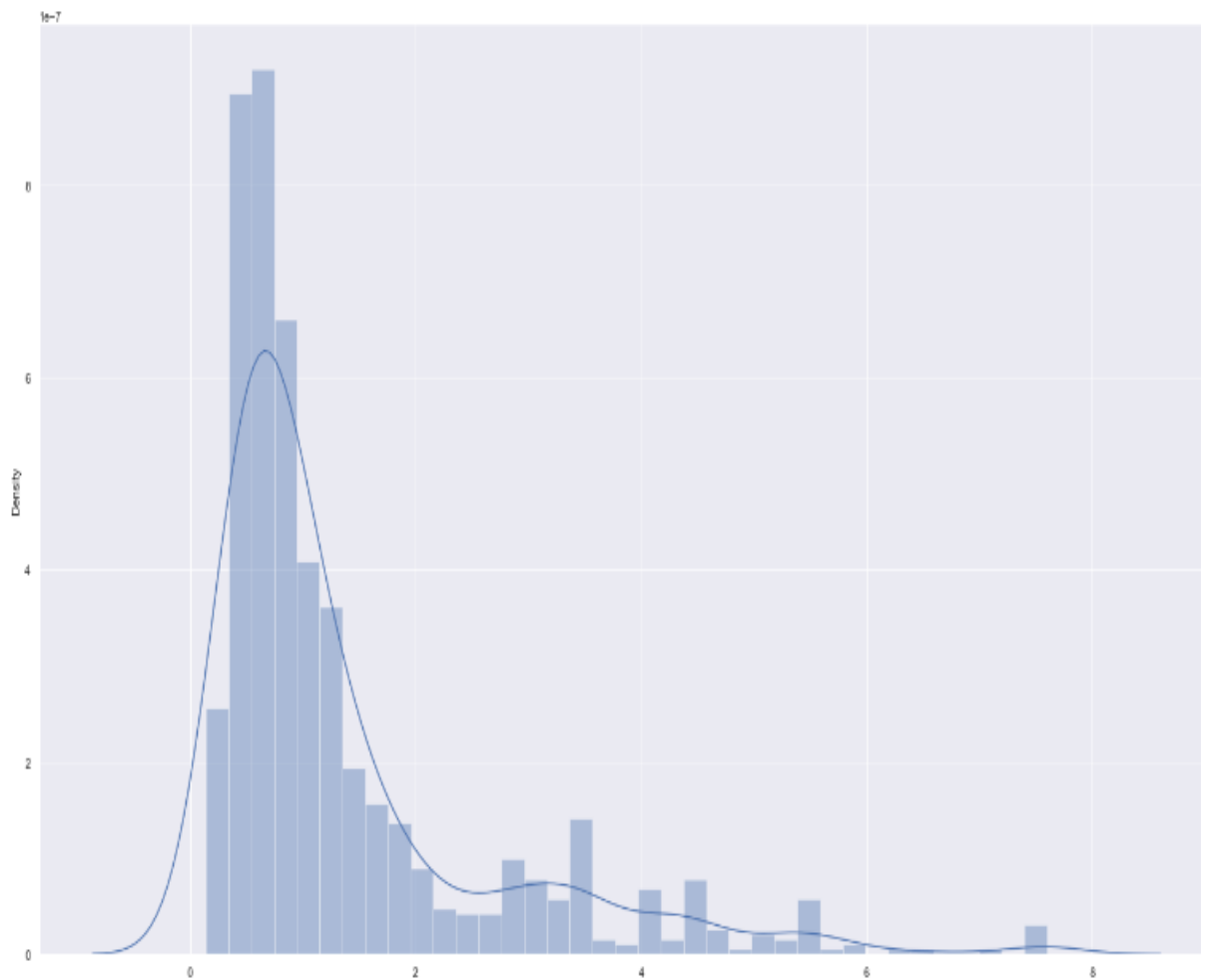
Below are some of the point's w.r.t. visualization, pre-processing presented above -

- ✚ Most of the used cars are of the brand "Maruti Suzuki" which is 1227 in numbers and sports cars are in less numbers.
- ✚ Most of the used cars have the Manufacturing Year as 2019 and 2017
- ✚ Most of the used cars are having the fuel type as Petrol which is 3206 in numbers then comes diesel as 1719.
- ✚ Minimum Used car price is 48k and maximum is 2.5cr
- ✚ Minimum drove Kms. for used car is 12km and maximum is 9cr km
- ✚ 3071 used cars have one owner; 1243 used cars have two owner.
- ✚ 764 used cars are from Mumbai and 742 are from Chennai
- ✚ The used cars those have fuel type as Hybrid i.e. diesel/Petrol both are having maximum 2 owners only and the price is more than 30 Lakhs. Used car having fuel type Diesel and having 4 owners have price more than 30Lakhs. The used car having fuel type (Petrol + CNG) have less car price and having at least two owner's maximum.
- ✚ The used cars having fuel type as Petrol are already driven more than 1.2L Kms.

CONCLUSION

As we can see in the above models '**RandomForestRegressor**' model seems best among other. With the help of this model the testing accuracy is coming as approx 85% which is a great score and also the RMSE error of this model is less as compare to other models. Also, as we can see in the graph that the best fitted Straight line of '**RandomForestRegressor**' model is containing more data-points as compare to that of others.

```
Out[306]: <AxesSubplot:ylabel='Density'>
```



```
In [305]: #Accuracy of RandomForestRegressor
accuracy_score=r2_score(pred,y_test)
accuracy_score
```

```
Out[305]: 0.8443847102053991
```

Now, getting the testing accuracy as almost 85% which is excellent accuracy

```
In [307]: #Plotting Best-Fitted Line
plt.figure(figsize=(8,6))
plt.scatter(y_test,pred,color='GREEN')
plt.plot(y_test,y_test,color='black')
plt.xlabel('Original Car_Price',fontsize=14)
plt.ylabel('Predicted Car_Price',fontsize=14)
plt.title('Best Fitted Line of the Model',fontsize=16)
```

```
Out[307]: Text(0.5, 1.0, 'Best Fitted Line of the Model')
```



```
In [308]: conclusion=pd.DataFrame(data=([pred,y_test]),index=['Predicted Car_Price','Original Car_Price'])
conclusion
```

```
Out[308]:
```

	0	1	2	3	4	5	6	7	8	9	...	938	939	...
Predicted Car_Price	655604.52	1038785.99	576463.0	1500619.98	970000.0	838800.02	2336016.98	855961.5	615912.95	324689.96	...	588180.01	1618090.0	586731
Original Car_Price	653199.00	1009000.00	580299.0	1136000.00	970000.0	784519.00	2625000.00	881899.0	425000.00	215000.00	...	631199.00	1549000.0	542999

2 rows x 948 columns

When I deployed our Random Forest Regressor model to the y_{test} data what I found is that the testing accuracy of the model went to 85% which is excellent accuracy for predicting any target variable correctly. As, we can see that the distance b/w the data-points of the predicted car price and original car price is not that high which indicates RMSE to be on lower side. Also, as we can see in the above figure the original and predicted car price data points are almost same and that's why Random Forest Regressor model is performing well in this scenario.

Learning Outcomes of the Study in respect of Data Science

I have used the Regression Model using multiple algorithms to design and optimize the results. Some of the classes which I explored from Scikit-learn libraries were – Statistics, Analytical Modelling, Hyper Tuning Method, CV Score, Predictive Modelling and different regression Model, etc.

I condensed to 6 columns keeping in mind the Principle Component Analysis. This resulted in condensed data being trained in a much shorter span of time with utmost accuracy. The visualizations helped in better and quick understanding of the outliers and skewness present in the data sets

With Mathematical Modeling helped me to find-out the corresponding mean, median, mode and relationship among the variables. Whereas Statistical Modeling helped in Correlation for understanding the relationship among the variables

One of the key challenges which I faced was importing the excel sheet to the Jupyter notebook because sorting of the data was not done initially as I've merged the data from four different websites and it's very obvious that each websites are having different attributes ergo I've sorted the excel-sheet first while launching it into Jupyter notebook. Also in the Fuel Column I've converted the categorical string classes into that of numerical form because when I was printing the unique values of this column then it was giving 17 classes rather than 7 classes and due to this I've converted the Owner column into respective numeric one too.

Fuel_Type	Description
0	Petrol
1	Diesel
2	CNG
3	Electric
4	Hybrid(Petrol/CNG)
5	LPG
6	Petrol + CNG

Limitations of this work and Scope for Future Work

Since I've extracted the dataset from different websites and made a consolidated one, so what I feel is that it would be more unbiased and productive information. But since some of the websites don't have the same attributes hence I've dropped few columns and have selected only those attributes which are important for this project.

The same analysis which was done for this project can be used for another Used Car related price prediction projects only if they provide the same characteristic that I've included in the Data Frame.