FLIP ROBO

CAR PRICE PREDICTION

SUBMITTED BY :

Saurav Kumar

ACKNOWLEDGMENT

I would like to thank Flip Robo Technologies for providing me with the opportunity to work on this project from which I have learned a lot. I am also grateful to Shwetank Mishra for his constant guidance and support .

INTRODUCTION

- **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 cheaper. 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. We have to make car price valuation model.

This project contains two phase :-

Data Collection Phase - You have to scrape at least 5000 used cars data. You can scrape more data as well, it's up to you. more the data better the model In this section You need to scrape the data of used cars from websites (Olx, cardekho, Cars24 etc.)

You need web scraping for this. You have to fetch data for different locations. The number of columns for data doesn't have limit, it's up to you and your creativity. Generally, these columns are Brand, model, variant, manufacturing year, driven kilometers, fuel, number of owners, location and at last target variable Price of the car.

This data is to give you a hint about important variables in used car model. You can make changes to it, you can add or you can remove some columns, it completely depends on the website from which you are fetching the data.

Try to include all types of cars in your data for example- SUV, Sedans, Coupe, minivan, Hatchback.

Model Building Phase :- After collecting the data, you need to build a machine learning model. Before model building do all data pre-processing steps. Try different models with different hyper parameters and select the best model.

Follow the complete life cycle of data science. Include all the steps like.

1. Data Cleaning

2. Exploratory Data Analysis

3. Data Pre-processing

4. Model Building

5. Model Evaluation

6. Selecting the best model

• Conceptual Background of the Domain Problem

So, In this problem Statement we have to Build a new Machine Learning Model of Second Hand Cars. We get data from OLX, Car24, CarDekho.com and etc. Due to covid 19 impact Some of the car in demand and make Costly and Some of the car are not in Demand so that car is not costly . Our client have old machine learning model which are predicting the car price . but after covid 19 in 2020 there are lots of changes in Second hand Car Maket . Because of less sell car in 2020 and 2021 the car company increases the price of the cars .

and all country facing the impact of Covid 19 . So the most of the person wants to buy a new car this reason for car company increases car prices and now we have scrap the data from Second Hand Car Selling website and build a machine learning model to predict the price car which are very high in demand.

• Review of Literature

This project is more about exploration, feature engineering and classification that can be done on this data. Since we scrape huge amount of data that includes more car related features, we can do better data exploration and derive some interesting features using the available columns.

The goal of this project is to build an application which can predict the car prices with the help of other features. In the long term, this would allow people to better explain and reviewing their purchase with each other in this increasingly digital world.

• Motivation for the Problem Undertaken

Based on the problem statement and the real time data scrapped from the Cars24 websites, I have understood how each independent feature helped me to understand the data as each feature provides a different kind of information. It is so interesting to work with different types of real time data in a single data set and perform root cause analysis to predict the price of the used car. Based on the analysis of the model of the Car Name,Model,Variant, Kilometres driven, Brand , Location, Number of Owner and fuel type etc. I would be able to model the price of used car as this model will then be used by the client to understand how exactly the prices vary with the variables. They can accordingly work on it and make some strategies to sell the used car and get some high returns. Furthermore, the model will be a good way for the client to understand the pricing dynamics of a used car.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

  - We have 8774 Rows and 16 Columns (6 Columns are Unncessory)

  - This dataset contain null values 1264 . We drop all null values . then we left with 7506 Rows and 10 Columns.

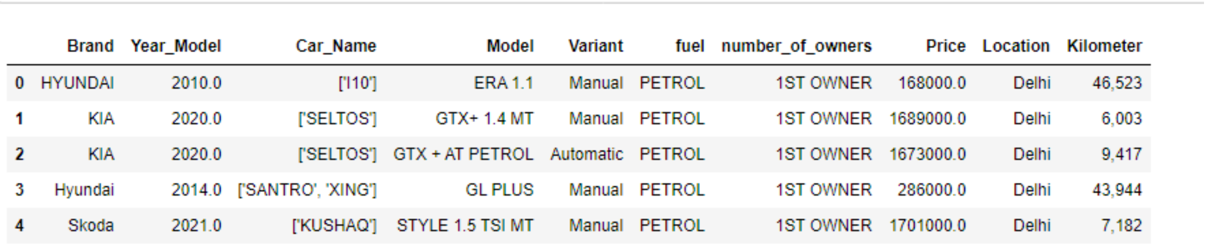
  - This Dataset Contain 4 Duplicated Value . We Drop the duplicates because we we think that for showing in top of the website many persons are update daily same to show in front of the website .

  - All the Columns has Object data type . And Year_Model is Float Data Type. And we have to convert Price and KM to a Float Data Type.

- Data Sources and their formats

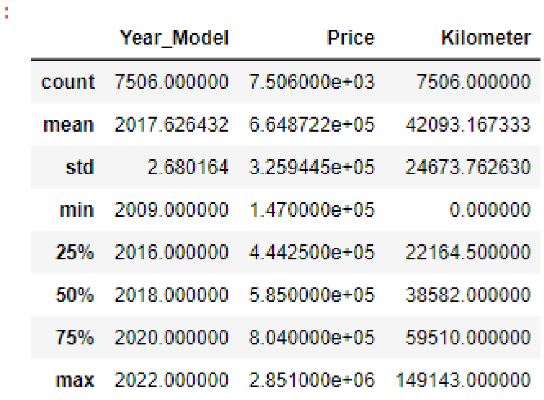
  - So, All the Data Scraped From Car24.com .by Using Selenium Web driver.
  - All the Data Save in CSV Formet.

Data Look Like->



	Brand	Year_Model	Car_Name	Model	Variant	fuel	number_of_owners	Price	Location	Kilometer
0	HYUNDAI	2010.0	['I10']	ERA 1.1	Manual	PETROL	1ST OWNER	168000.0	Delhi	46,523
1	KIA	2020.0	['SELTOS']	GTX+ 1.4 MT	Manual	PETROL	1ST OWNER	1689000.0	Delhi	6,003
2	KIA	2020.0	['SELTOS']	GTX + AT PETROL	Automatic	PETROL	1ST OWNER	1673000.0	Delhi	9,417
3	Hyundai	2014.0	['SANTRO', 'XING']	GL PLUS	Manual	PETROL	1ST OWNER	286000.0	Delhi	43,944
4	Skoda	2021.0	['KUSHAQ']	STYLE 1.5 TSI MT	Manual	PETROL	1ST OWNER	1701000.0	Delhi	7,182

Describe the Dataset.

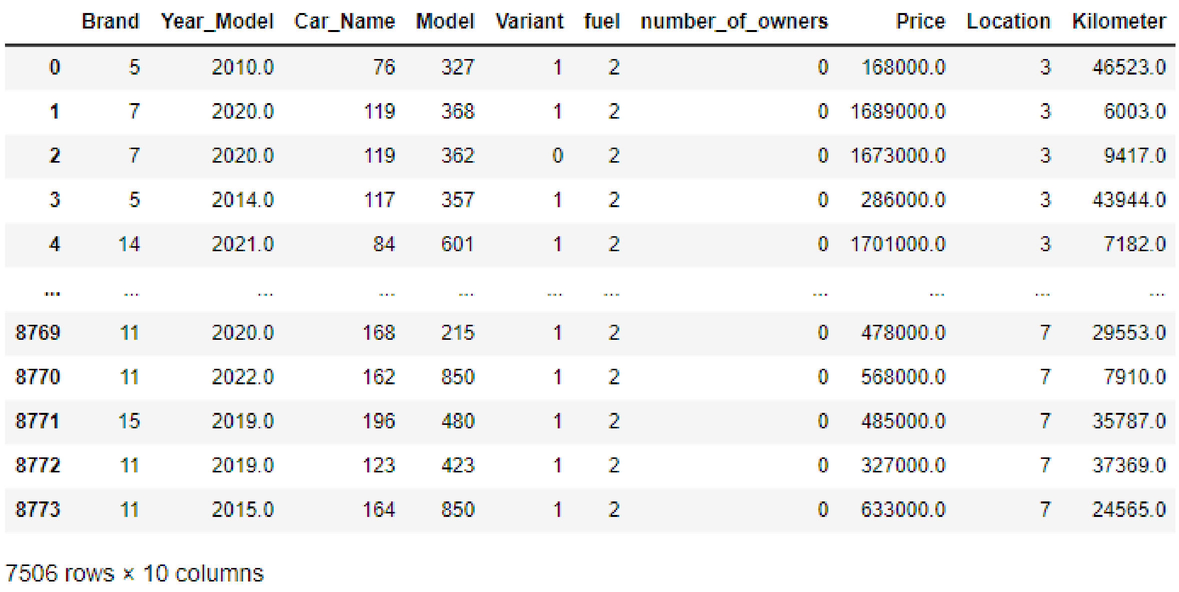


	Year_Model	Price	Kilometer
count	7506.000000	7.506000e+03	7506.000000
mean	2017.626432	6.648722e+05	42093.167333
std	2.680164	3.259445e+05	24673.762630
min	2009.000000	1.470000e+05	0.000000
25%	2016.000000	4.442500e+05	22164.500000
50%	2018.000000	5.850000e+05	38582.000000
75%	2020.000000	8.040000e+05	59510.000000
max	2022.000000	2.851000e+06	149143.000000

- Data Preprocessing Done

As , We all know our maximum data point are in object data type so we have encode first using label encoder.

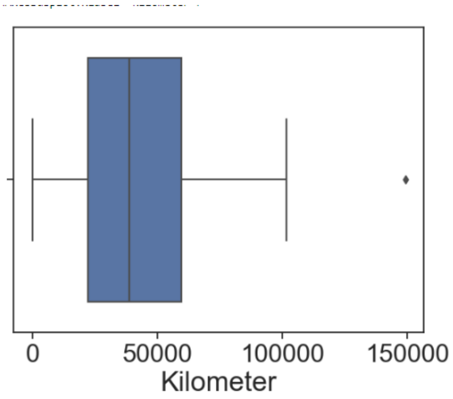
After Endoing we check the Statistically Summary of the Dataset.

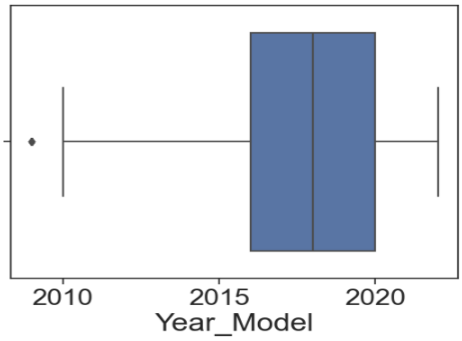


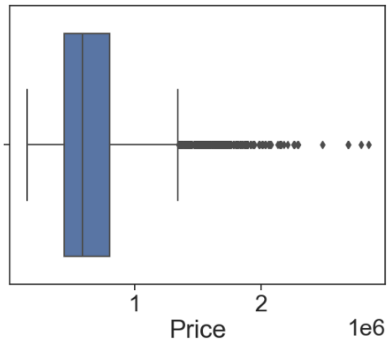
	Brand	Year_Model	Car_Name	Model	Variant	fuel	number_of_owners	Price	Location	Kilometer
0	5	2010.0	76	327	1	2	0	168000.0	3	46523.0
1	7	2020.0	119	368	1	2	0	1689000.0	3	6003.0
2	7	2020.0	119	362	0	2	0	1673000.0	3	9417.0
3	5	2014.0	117	357	1	2	0	286000.0	3	43944.0
4	14	2021.0	84	601	1	2	0	1701000.0	3	7182.0
...	...	...	...	...	...	...	...	...	...	...
8769	11	2020.0	168	215	1	2	0	478000.0	7	29553.0
8770	11	2022.0	162	850	1	2	0	568000.0	7	7910.0
8771	15	2019.0	196	480	1	2	0	485000.0	7	35787.0
8772	11	2019.0	123	423	1	2	0	327000.0	7	37369.0
8773	11	2015.0	164	850	1	2	0	633000.0	7	24565.0

7506 rows × 10 columns

Checking for Outliers -> Only on Continious Variable .





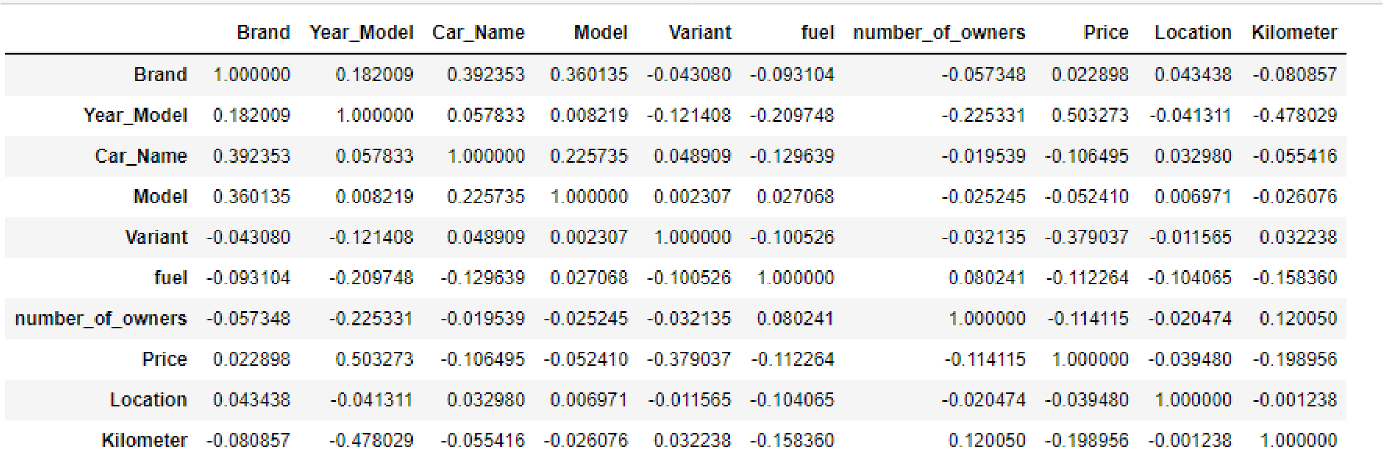


Kelometer -> We , Have outliers but we dont remove this because we consider 150000 KM any car can run

Year_Model -> Here , We have outlers . but we dont consider as a outlier beacuse beacuse 2010 model car are running in road . beacuse in india any motor vehicle life is for 15 Year.

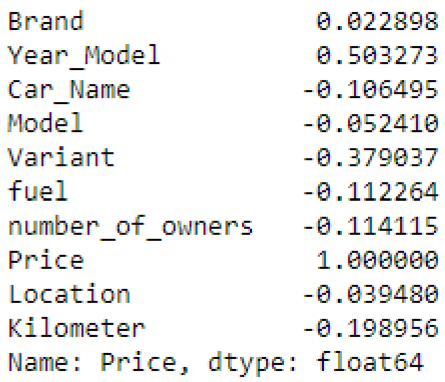
Price -> We, Have outlier but we dont correct them because we its a target column.

Checking the Correlation :



	Brand	Year_Model	Car_Name	Model	Variant	fuel	number_of_owners	Price	Location	Kilometer
Brand	1.000000	0.182009	0.392353	0.360135	-0.043080	-0.093104	-0.057348	0.022898	0.043438	-0.080857
Year_Model	0.182009	1.000000	0.057833	0.008219	-0.121408	-0.209748	-0.225331	0.503273	-0.041311	-0.478029
Car_Name	0.392353	0.057833	1.000000	0.225735	0.048909	-0.129639	-0.019539	-0.106495	0.032980	-0.055416
Model	0.360135	0.008219	0.225735	1.000000	0.002307	0.027068	-0.025245	-0.052410	0.006971	-0.026076
Variant	-0.043080	-0.121408	0.048909	0.002307	1.000000	-0.100526	-0.032135	-0.379037	-0.011565	0.032238
fuel	-0.093104	-0.209748	-0.129639	0.027068	-0.100526	1.000000	0.080241	-0.112264	-0.104065	-0.158360
number_of_owners	-0.057348	-0.225331	-0.019539	-0.025245	-0.032135	0.080241	1.000000	-0.114115	-0.020474	0.120050
Price	0.022898	0.503273	-0.106495	-0.052410	-0.379037	-0.112264	-0.114115	1.000000	-0.039480	-0.198956
Location	0.043438	-0.041311	0.032980	0.006971	-0.011565	-0.104065	-0.020474	-0.039480	1.000000	-0.001238
Kilometer	-0.080857	-0.478029	-0.055416	-0.026076	0.032238	-0.158360	0.120050	-0.198956	-0.001238	1.000000

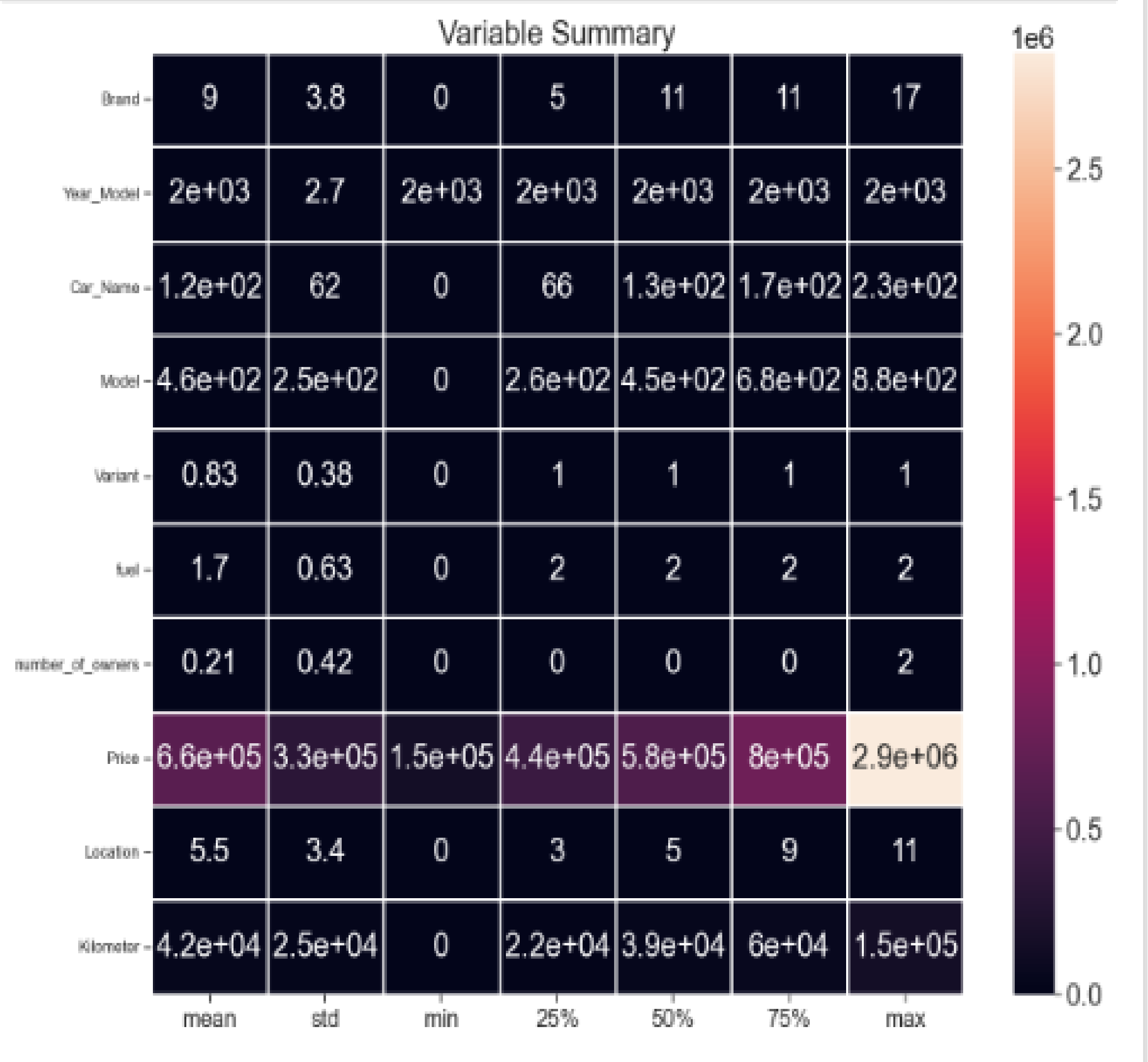
Lets Check Correlation With Target Variable (Price):



```
Brand               0.022898
Year_Model          0.503273
Car_Name           -0.106495
Model              -0.052410
Variant            -0.379037
fuel               -0.112264
number_of_owners   -0.114115
Price               1.000000
Location           -0.039480
Kilometer          -0.198956
Name: Price, dtype: float64
```

Here , We can see that Year_Model ,Car_Name, Variant , Number_of_Owners and Kelometer have relation with Price

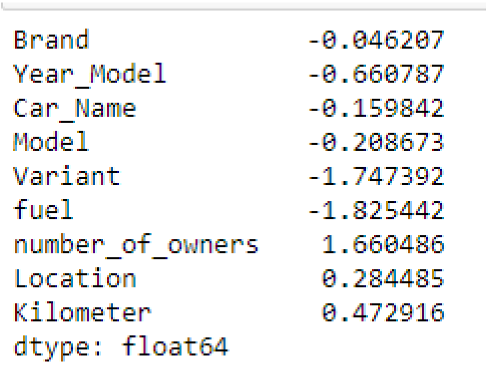
Describing the Dataset after Encoding



Variable Summary

	mean	std	min	25%	50%	75%	max
Brand	9	3.8	0	5	11	11	17
Year_Model	2e+03	2.7	2e+03	2e+03	2e+03	2e+03	2e+03
Car_Name	1.2e+02	62	0	66	1.3e+02	1.7e+02	2.3e+02
Model	4.6e+02	2.5e+02	0	2.6e+02	4.5e+02	6.8e+02	8.8e+02
Variant	0.83	0.38	0	1	1	1	1
fuel	1.7	0.63	0	2	2	2	2
number_of_owners	0.21	0.42	0	0	0	0	2
Price	6.6e+05	3.3e+05	1.5e+05	4.4e+05	5.8e+05	8e+05	2.9e+06
Location	5.5	3.4	0	3	5	9	11
Kilometer	4.2e+04	2.5e+04	0	2.2e+04	3.9e+04	6e+04	1.5e+05

Then we Separate Features and Label .and then Check For skewness



```
Brand                -0.046207
Year_Model           -0.660787
Car_Name             -0.159842
Model                -0.208673
Variant              -1.747392
fuel                 -1.825442
number_of_owners      1.660486
Location              0.284485
Kilometer             0.472916
dtype: float64
```

Here we set threshold for +/- 0.66. and we can see that our all continous variable comes in threshold range. So, We dont need no remove skewness. Lets move foreword.

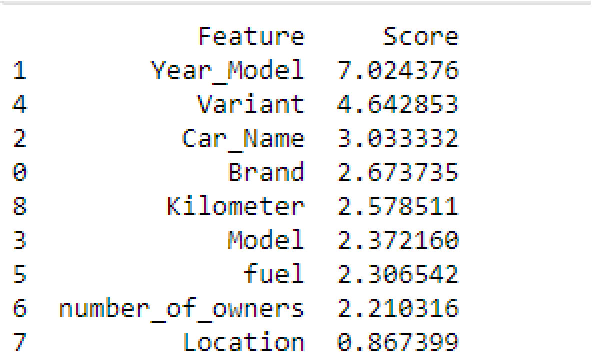
- **Hardware and Software Requirements and Tool Used**
  - Anaconda Navigator -> Jupyter Notebook
  - Windows 11 - > Home Version
  - Hardware -> AMD Ryzen 3 Processor with Vega Graphics 2200 U.
  - Using Selenium Web Driver to Scrape the data from car24.com.

Tool Used -> Python , Pandas , Numpy , Seaborn , Matplotlib , Machine Learning Algorithms .

- **Model/s Development and Evaluation**

  Here I am Separating the Feature and Label in X and Y . X is use for feature and Y is used for label . Then we check for skewness and we dont find any skewness .

then we use Standard Scaler .Standard Scaler Basically Normalize the data .
After that we use SelectKBest for feature selection . and we can see that all feature contributiong for predicting the model.

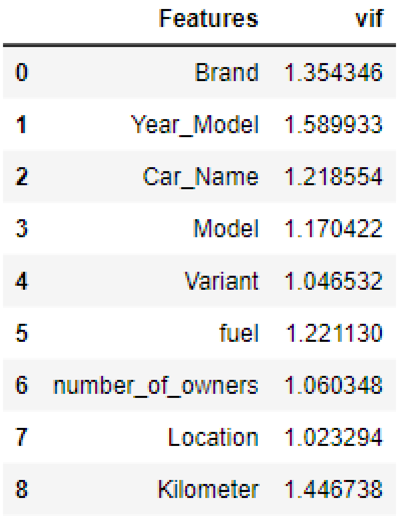


```
             Feature     Score
1         Year_Model  7.024376
4            Variant  4.642853
2           Car_Name  3.033332
0              Brand  2.673735
8          Kilometer  2.578511
3              Model  2.372160
5               fuel  2.306542
6   number_of_owners  2.210316
7           Location  0.867399
```

<u>**Here, We can find all the columns contributing in tha model building**</u>

Check For VIF

The, We check for multi collinearity problem using VIF (Variane Inflation Factor). And we set threshold for +/- 10 .



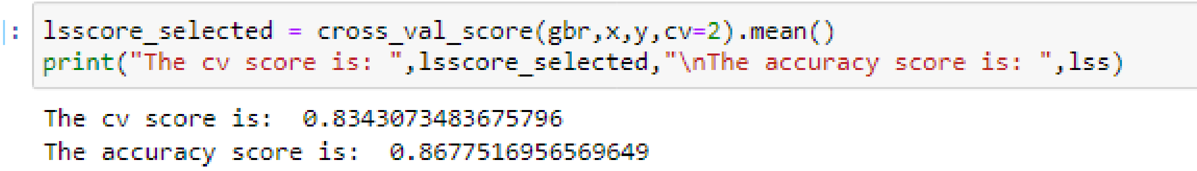
	Features	vif
0	Brand	1.354346
1	Year_Model	1.589933
2	Car_Name	1.218554
3	Model	1.170422
4	Variant	1.046532
5	fuel	1.221130
6	number_of_owners	1.060348
7	Location	1.023294
8	Kilometer	1.446738

Here we can set threshold for 10 . and all the column comes in our range. and we dont have multicollieanrity problem.. we are good to proceed.

Lets Build a Model : -

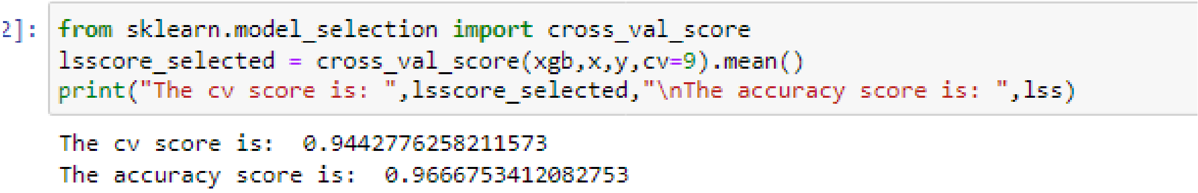
Here , We are using 4 Machine Learning Algorithm. And all four gives us good accuracy for confirming that we use also cross validation score .

Gradient Bossting Regressor ->



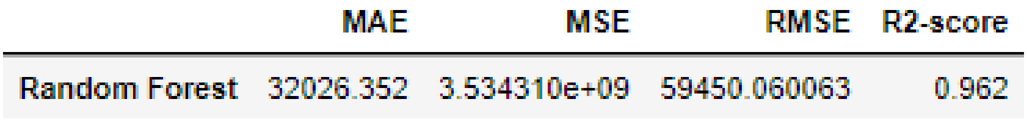
```
|: lsscore_selected = cross_val_score(gbr,x,y,cv=2).mean()
   print("The cv score is: ",lsscore_selected,"\nThe accuracy score is: ",lss)

   The cv score is:  0.8343073483675796
   The accuracy score is:  0.8677516956569649
```

Xtreme Gradient Boost ->



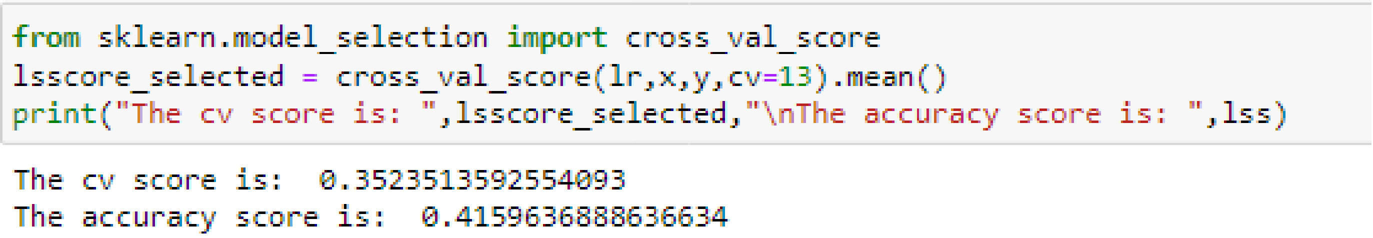
```
2]: from sklearn.model_selection import cross_val_score
    lsscore_selected = cross_val_score(xgb,x,y,cv=9).mean()
    print("The cv score is: ",lsscore_selected,"\nThe accuracy score is: ",lss)

    The cv score is:  0.9442776258211573
    The accuracy score is:  0.9666753412082753
```

Random Forest ->



	MAE	MSE	RMSE	R2-score
Random Forest	32026.352	3.534310e+09	59450.060063	0.962

Linear Regression –

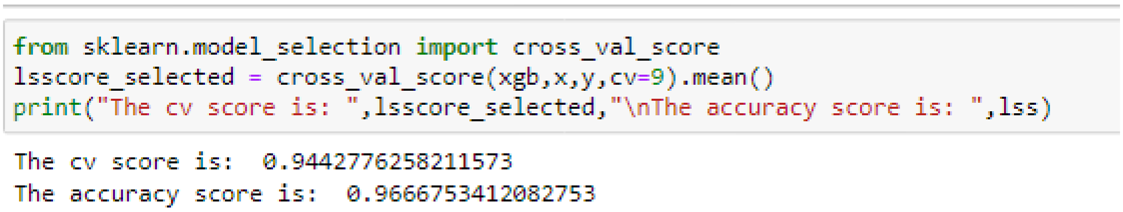


```
from sklearn.model_selection import cross_val_score
lsscore_selected = cross_val_score(lr,x,y,cv=13).mean()
print("The cv score is: ",lsscore_selected,"\nThe accuracy score is: ",lss)

The cv score is:  0.3523513592554093
The accuracy score is:  0.4159636888636634
```

And , In Model Building time I am using almost all algorithm which I was learned . But I can show only three algorithm . and I am selecting XGB for model building because its gives a best accuracy .

Lets Do Hyperparameter Tuning -> Using (Grid Search CV)

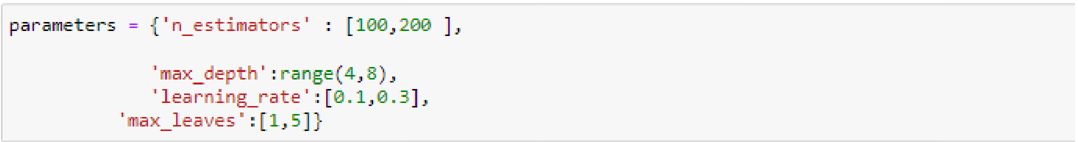
XGB ->



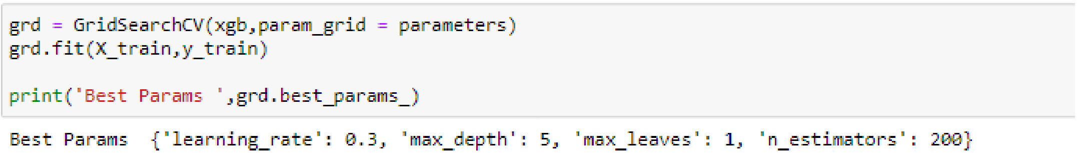
```
from sklearn.model_selection import cross_val_score
lsscore_selected = cross_val_score(xgb,x,y,cv=9).mean()
print("The cv score is: ",lsscore_selected,"\nThe accuracy score is: ",lss)
```

```
The cv score is:  0.9442776258211573
The accuracy score is:  0.9666753412082753
```

Lets Tune the Parameter ->

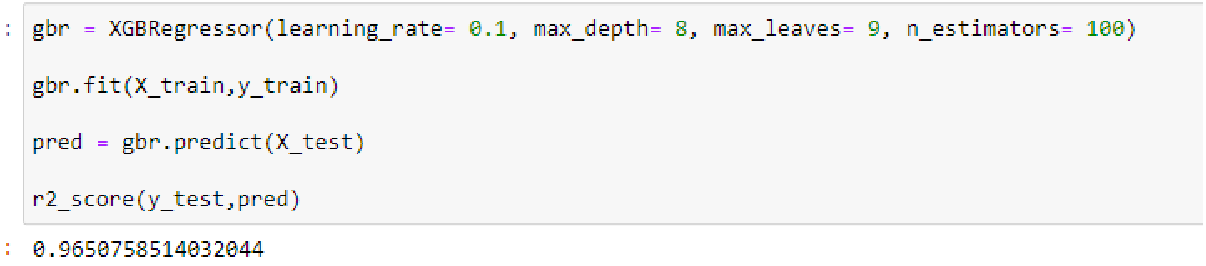


```
parameters = {'n_estimators' : [100,200 ],

             'max_depth':range(4,8),
             'learning_rate':[0.1,0.3],
          'max_leaves':[1,5]}
```



```
grd = GridSearchCV(xgb,param_grid = parameters)
grd.fit(X_train,y_train)

print('Best Params ',grd.best_params_)
```

```
Best Params  {'learning_rate': 0.3, 'max_depth': 5, 'max_leaves': 1, 'n_estimators': 200}
```

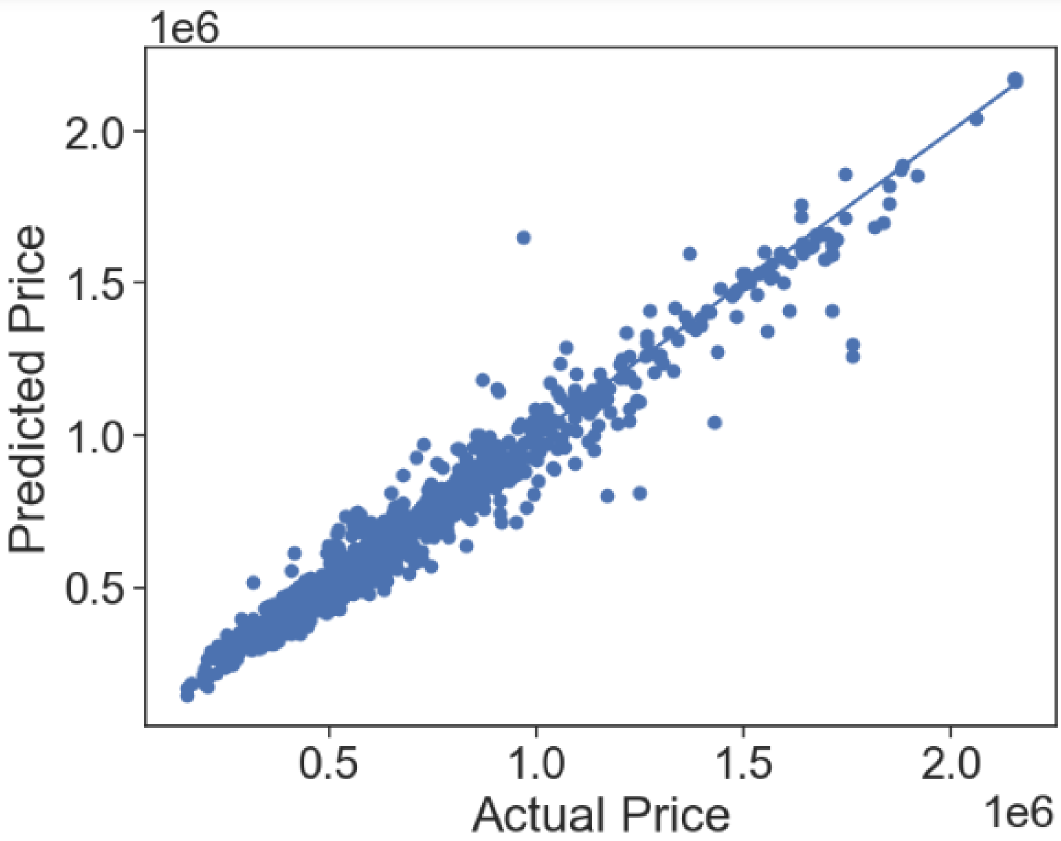


```
: gbr = XGBRegressor(learning_rate= 0.1, max_depth= 8, max_leaves= 9, n_estimators= 100)

  gbr.fit(X_train,y_train)

  pred = gbr.predict(X_test)

  r2_score(y_test,pred)
```

```
: 0.9650758514032044
```

After , Tune the Parameter we get accuracy 96 % and our Cross Val Score is – 94 . So its means is our model is not overfitted .

And Now we are good to plot Best Fit Line .



Here , We can see that Our most of The poins come very close to the Best fit line. Its meaning is our model Is good .

CONCLUSION

- We getiing the info if the seller has to focus on Maruti , Hyundai , Tata ,Mahindra and Honda Because Most of the Indian Family Is Middile Class and Those Company I Mentioned they are Making Mid Ranges Car So its easy to capture market .

- Most of the second hand car sell in price between 1.5 Lakh to 6 Lakhs Rupees. So we have to more focus on this price range car.
- Costly Car Sell From Toyota (Fortuner, Innova Crysta,) , Audi (Q3). And etc . Follow the 15 , 16 Slide .in PPT.

Learning Outcomes of the Study in respect of Data Science .

Visualization part helped me to understand the data as it provides graphical representation of huge data. It assisted me to understand the feature importance, outliers/skewness detection and to compare the independent-dependent features.Data cleaning is the most important part of model building and therefore before model building,I made sure the data is cleaned and scaled.I have generated multiple regression machine learning models to get the best model wherein I found Extra XGB Model being the best based on the metrics I have used.

Limitations of this work and Scope for Future Work

The limitations we faced during this project were : Presence of Outliers in Scraped data from car 24 , and the major problem is the separate the Car Name and the model in car24.com.

Scope of Future Work -> Current model is limited to used car data but this can further be improved for other sectors of automobiles by training the model accordingly. The overall score can also be improved further by training the model with more specific data.

Thankyou ..