PROJECT REPORT ON:

"Used-Car Price Prediction Project"

SUBMITTED BY

RAHUL M

ACKNOWLEDGMENT

I would like to express my special gratitude to “Flip Robo” team, who has given me this opportunity to deal with a beautiful dataset and it has helped me to improve my analyzation skills. And I want to express my huge gratitude to Shwetank Mishra (SME Flip Robo) who encouraged me a lot with his valuable words and with his unconditional support I have ended up with a beautiful Project.

A huge thanks to my academic team “Data trained” who are the reason behind what I am today. Last but not least my parents who have been my backbone in every step of my life. And also thank you for many other persons who has helped me directly or indirectly to complete the project.

1.INTRODUCTION

1.1 Business Problem Framing:

Car price prediction is somehow interesting and popular problem. As per information that was gotten from the Agency for Statistics of BiH, 921.456 vehicles were registered in 2014 from which 84% of them are cars for personal usage. This number is increased by 2.7% since 2013 and it is likely that this trend will continue, and the number of cars will increase in future. This adds additional significance to the problem of the car price prediction. Accurate car price prediction involves expert knowledge, because price usually depends on many distinctive features and factors. Typically, most significant ones are brand and model, age, horsepower and mileage. The fuel type used in the car as well as fuel consumption per mile highly affect price of a car due to a frequent changes in the price of a fuel. Different features like exterior color, door number, type of transmission, dimensions, safety, air condition, interior, whether it has navigation or not will also influence the car price. In this report, we applied different methods and techniques in order to achieve higher precision of the used car price prediction.

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.

1.2 Conceptual Background of the Domain Problem

The prices of new cars in the industry is fixed by the manufacturer with some additional costs incurred by the Government in the form of taxes. So, customers buying a new car can be assured of the money they invest to be worthy. But due to the increased price of new cars and the incapability of customers to buy new cars due to the lack of funds, used cars sales are on a global increase. There is a need for a used car price prediction system to

effectively determine the worthiness of the car using a variety of features. Even though there are websites that offers this service, their prediction method may not be the best. Besides, different models and systems may contribute on predicting power for a used car's actual market value. It is important to know their actual market value while both buying and selling.

There are lots of individuals who are interested in the used car market at some points in their life because they wanted to sell their car or buy a used car. In this process, it's a big corner to pay too much or sell less than it's market value.

There are one of the biggest target group that can be interested in results of this study. If used car sellers better understand what makes a car desirable, what are the important features for a used car, then they may consider this knowledge and offer a better service.

1.3 Review of Literature

The second-hand car market has continued to expand even as the reduction in the market of new cars. According to the recent report on India's pre-owned car market by Indian Blue Book, nearly 4 million used cars were purchased and sold in 2018-19. The second-hand car market has created the business for both buyers and sellers. Most of the people prefer to buy the used cars because of the affordable price and they can resell that again after some years of usage which may get some profit. The price of used cars depends on many factors like fuel type, colour, model, mileage, transmission, engine, number of seats etc., The used cars price in the market will keep on changing. Thus the evaluation model to predict the price of the used cars is required.

1.4 Motivation for the Problem Undertaken

There are websites that offers an estimate value of a car. They may have a good prediction model. However, having a second model may help them to give a better prediction to their users. Therefore, the model developed in this study may help online web services that tells a used car's market value.

2.Analytical Problem Framing

2.1 Mathematical/ Analytical Modeling of the Problem

As a first step I have scrapped the required data from carsdekho website. I have fetched data for different locations and saved it to excel format.

In this perticular problem I have car_price as my target column and it was a continuous column. So clearly it is a regression problem and I have to use regression algorithms while building the model. There was null values in the dataset. Since we have scrapped the data from cardekho website the raw data was not in the format, so we have use feature engineering to extract the required feature format. To get better insight on the features I have used ploting like distribution plot, bar plot, reg plot, strip plot and count plot. With these ploting I was able to understand the relation between the features in better manner. Also, I found outliers and skewness in the dataset so I removed outliers using z-score method and I removed skewness using yeo-johnson method. I have used all the regression algorithms while building model then tuned the best model and saved the best model. At last I have predicted the car-price using saved model.

2.2 Data Sources and their formats

The data was collected from cardekho.com website in excel format. The data was scrapped using selenium. After scrapping required features the dataset is saved as excel file.

Also, my dataset was having 11281 rows and 11 columns including target. In this perticular datasets I have object type of data which has been changed as p er our analysis about the dataset. The information about features is as follows.

Features Information:

- Car_Name : Name of the car with Year
- City : Name of the particular location
- Manuf_yr : year of the car manufavtured
- Fuel_type : Type of fuel used for car engine
- Kms_driven : Car running in kms till the date
- ownerships : Number of pwnership passed till date

- Displacement_in_CC : Engine displacement/engine CC
- Transmission : Type of gear transmission used in car
- Mileage : Overall milage of car in Km/ltr
- seat_nos : Availability of number of seats in the car
- Car_price : Price of the car
- Brand : name of the car Brand

2.3 Data Preprocessing Done

- ✓ As a first step I have scrapped the required data using selenium from cardekho website.
- ✓ And I have imported required libraries and I have imported the dataset which was in excel format.
- ✓ Then I did all the statistical analysis like checking shape, nunique, value counts, info etc…..
- ✓ While checking for null values I found null values in the dataset and I replaced them using imputation technique.
- ✓ I have also droped Unnamed:0 column as I found they are useless.
- ✓ Next as a part of feature extraction I converted the data types of all the columns and I have extracted usefull information from the raw dataset. Thinking that this data will help us more than raw data.

2.4 Data Inputs- Logic- Output Relationships

- ✓ Since I had numerical columns I have plotted dist plot to see the distribution of skewness in each column data.
- ✓ I have used bar plot for each pair of categorical features that shows the relation between label and independent features.
- ✓ I have used reg plot and strip plot to see the relation between numerical columns with target column.
- ✓ I can notice there is a linear relationship between maximum columns and target.

2.5 Hardware and Software Requirements and Tools Used

While taking up the project we should be familiar with the Hardware and software required for the successful completion of the project. Here we need the following hardware and software.

Hardware required: -

1. Processor — core i5 and above
2. RAM — 8 GB or above
3. SSD — 250GB or above

Software/s required: -

1.Anaconda

Libraries required :-

To run the program and to build the model we need some basic libraries as follows:

```
In [1]: #importing required libraries
        import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import seaborn as sns
        import datetime as dt

        import warnings
        warnings.filterwarnings('ignore')
```

- ✓ **import pandas as pd**: **pandas** is a popular Python-based data analysis toolkit which can be imported using import pandas as pd. It presents a diverse range of utilities, ranging from parsing multiple file formats to converting an entire data table into a numpy matrix array. This makes pandas a trusted ally in data science and machine learning.
- ✓ **import numpy as np**: NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting,

selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

- ✓ **import seaborn as sns:** Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python. Visualization is the central part of Seaborn which helps in exploration and understanding of data.
- ✓ **Import matplotlib.pyplot as plt:** matplotlib.pyplot is a collection of functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.
- ✓ from sklearn.preprocessing import LabelEncoder
- ✓ from sklearn.preprocessing import StandardScaler
- ✓ from sklearn.tree import DecisionTreeRegressor
- ✓ from sklearn.ensemble import RandomForestRegressor
- ✓ from sklearn.ensemble import AdaBoostRegressor
- ✓ from sklearn.neighbors import KNeighborsRegressor
- ✓ from sklearn.ensemble import GradientBoostingRegressor
- ✓ from xgboost import XGBRegressor
- ✓ from sklearn.ensemble import ExtraTreesRegressor
- ✓ from sklearn.metrics import accuracy_score
- ✓ from sklearn.model_selection import cross_val_score

With this sufficient libraries we can go ahead with our model building.

3.Data Analysis and Visualization

3.1 Identification of possible problem-solving approaches (methods)

- ✓ Since the data collected was not in the format we have to clean it and bring it to the proper format for our analysis. To remove outliers I have used z-score method. And to remove skewness I have used yeo-johnson method. We have dropped all the unnecessary columns in the dataset according to our understanding. Use of Pearson's correlation coefficient to check the correlation between dependent and independent features. Also I have used Standardisation to scale the data. After scaling we have to remove multicolinearity using VIF. Then followed by model building with Regression algorithms.

3.2 Testing of Identified Approaches (Algorithms)

Since price_in_lakhs was my target and it was a continuous column with improper format which has to be changed to continuous float datatype column, so this particular problem was Regression problem. And I have used Regression algorithms to build my model. By looking into the difference of r2 score and cross validation score I found ExtraTreesRegressor as a best model with least difference. Also to get the best model we have to run through multiple models and to avoid the confusion of overfitting we have go through cross validation. Below are the list of Regression algorithms I have used in my project.

- ➢ RandomForestRegressor
- ➢ XGBRegressor
- ➢ ExtraTreesRegressor
- ➢ GradientBoostingRegressor
- ➢ DecisionTreeRegressor
- ➢ KNeighborsRegressor
- ➢ AdaBoostRegressor

3.3 Key Metrics for success in solving problem under consideration

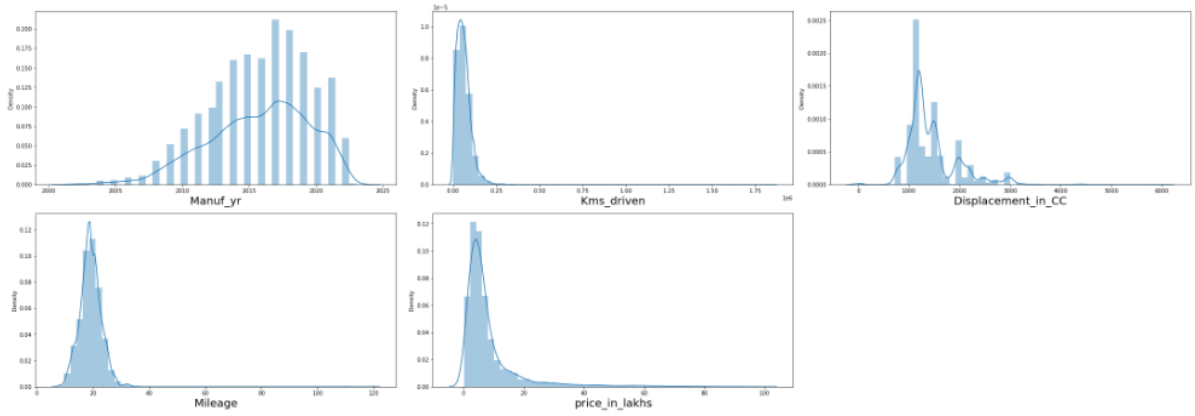
I have used the following metrics for evaluation:

- I have used mean absolute error which gives magnitude of difference between the prediction of an observation and the true value of that observation.
- I have used root mean square deviation is one of the most commonly used measures for evaluating the quality of predictions.
- I have used r2 score which tells us how accurate our model is.

3.4 Visualizations

I have used bar plots to see the relation of categorical feature with target and I have used 2 types of plots for numerical columns one is disp plot for univariate and reg plot, strip plot for bivariate analysis.

1. Univariate Analysis for numerical columns:



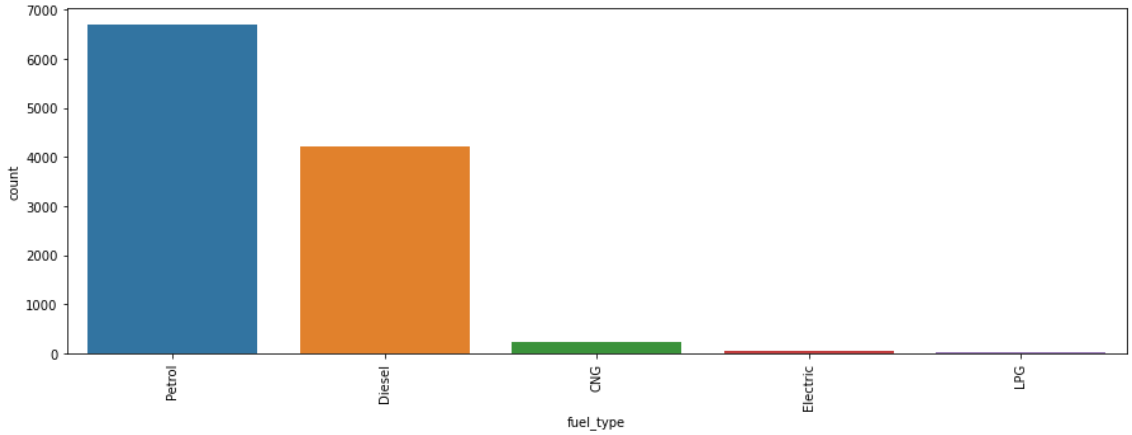
Observations:

- ✓ We can clearly see that there is skewness in most of the columns so we have to treat them using suitable methods.

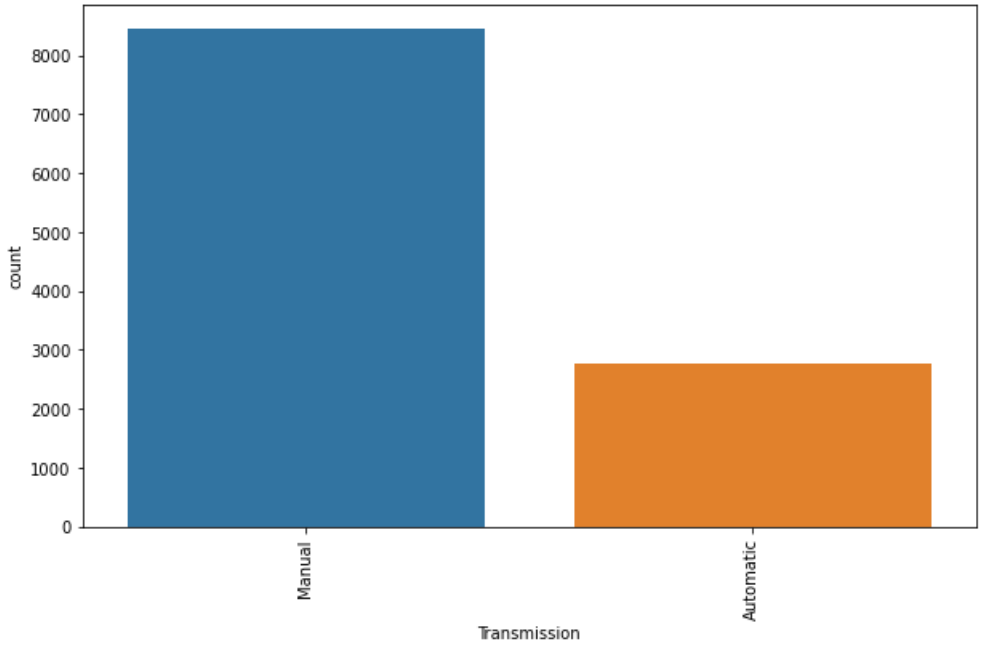
2. Univariate analysis for categorical column:

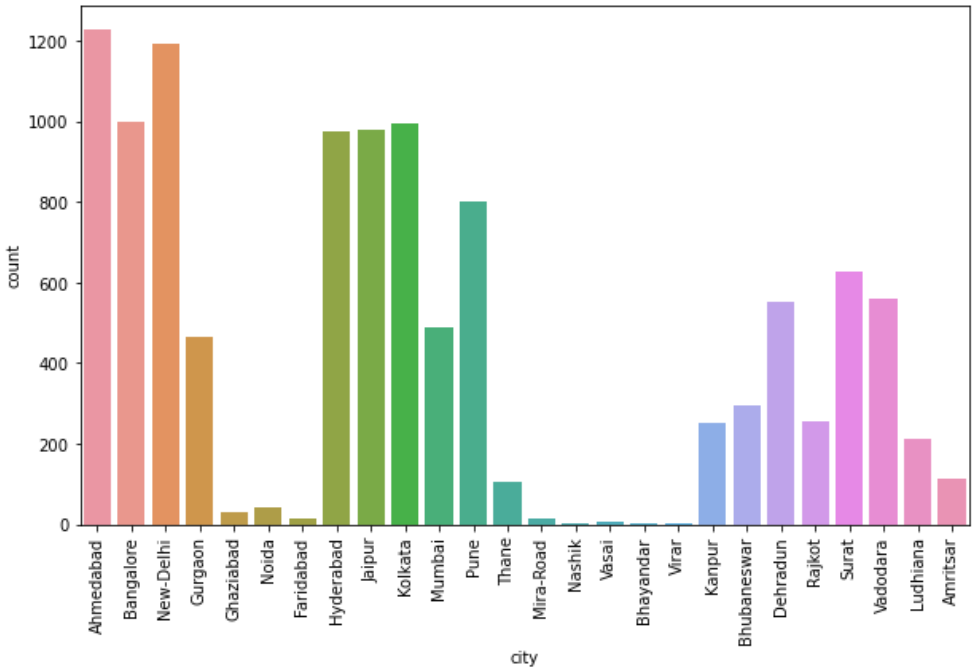
- ✓ We can see that most of the dataset consist of petrol cars with a count of 6701 and

- ✓ Then in second Diesel having 4210
- ✓ Cng having 227
- ✓ Electric and Lpg are less than 40 in the dataset



- ✓ We can see that most of the dataset consist of Manual gear transmission in cars with a count of 8450
- ✓ Automatic having count of 2757

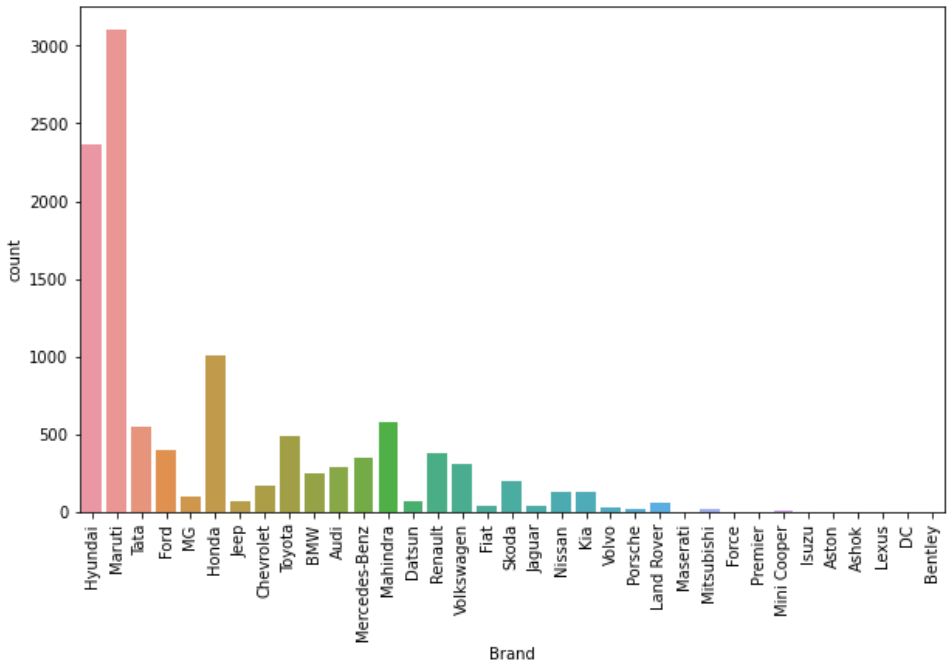




- Ahmedabad 1228
- New-Delhi 1193
- Bangalore 997
- Kolkata 996
- Jaipur 978
- Hyderabad 977
- Pune 800
- Surat 629
- Vadodara 561
- Dehradun 551
- Mumbai 487
- Gurgaon 467
- Bhubaneswar 295
- Rajkot 255
- Kanpur 252
- Ludhiana 211
- Amritsar 114
- Thane 104
- Noida 42
- Ghaziabad 31
- Mira-Road 15
- Faridabad 14
- Vasai 6
- Virar 2
- Bhayandar 1

- Nashik 1

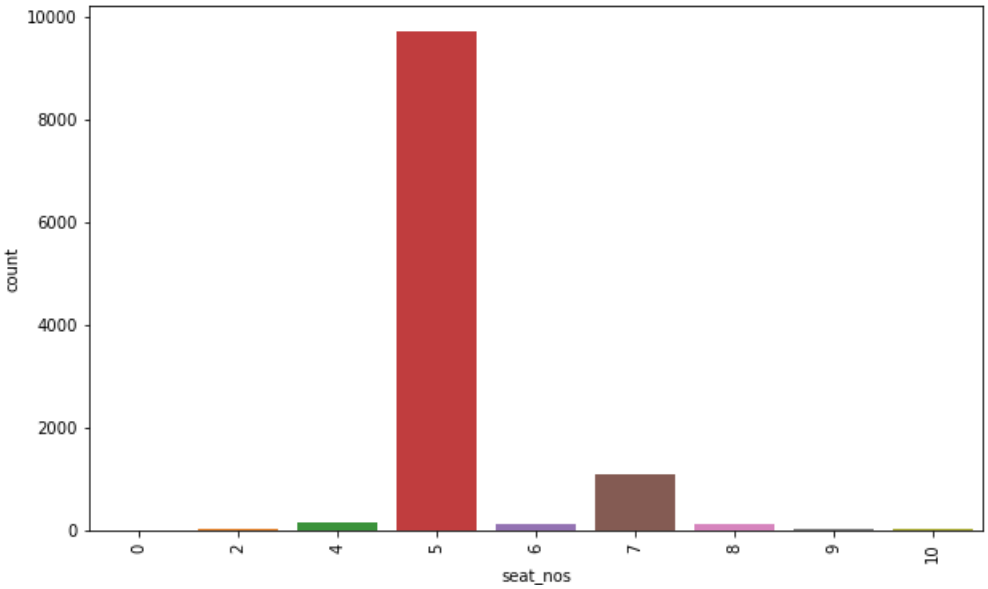
Most of the data is from Ahmedabad location with a count of 1228 in the dataset



- Maruti 3098
- Hyundai 2369
- Honda 1011
- Mahindra 581
- Tata 548
- Toyota 493
- Ford 401
- Renault 380
- Mercedes-Benz 354
- Volkswagen 314
- Audi 291
- BMW 254
- Skoda 197
- Chevrolet 169
- Kia 134
- Nissan 129
- MG 100
- Datsun 71
- Jeep 71
- Land Rover 64
- Jaguar 40
- Fiat 36

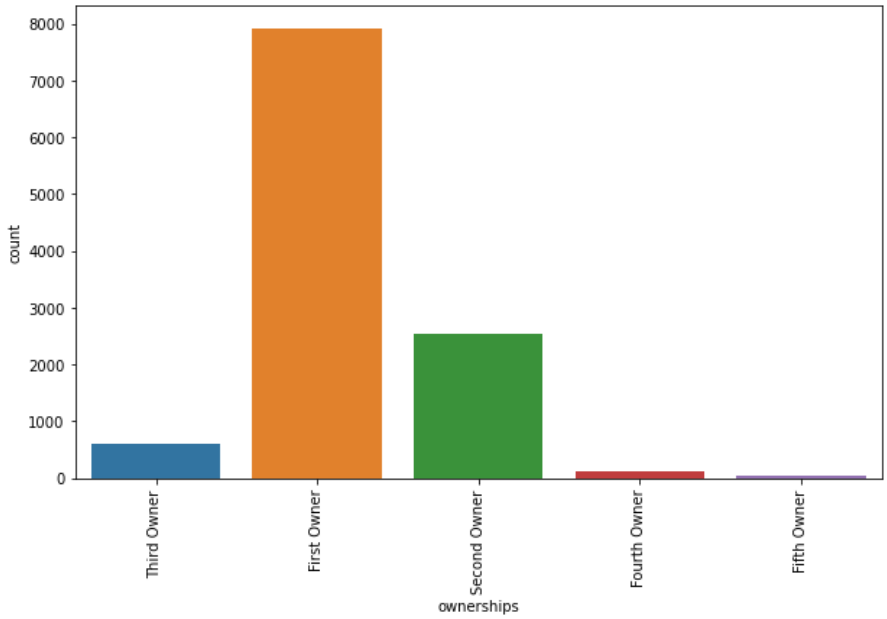
- Volvo 34
- Mitsubishi 20
- Porsche 16
- Mini Cooper 14
- Maserati 4
- Isuzu 3
- Lexus 3
- Bentley 3
- Premier 1
- Force 1
- Aston 1
- Ashok 1
- DC 1

✓ so the max data is for Maruti cars with a count of 3098 and least datacis for Premier,Force,Aston,Ashok,DC among the brands



✓ 5seater 9706
✓ 7seater 1092
✓ 4seater 144
✓ 6seater 121
✓ 8seater 112
✓ 9seater 15
✓ 2seater 11
✓ 10seater 5

we can see the majority of the cars are having 5seaters

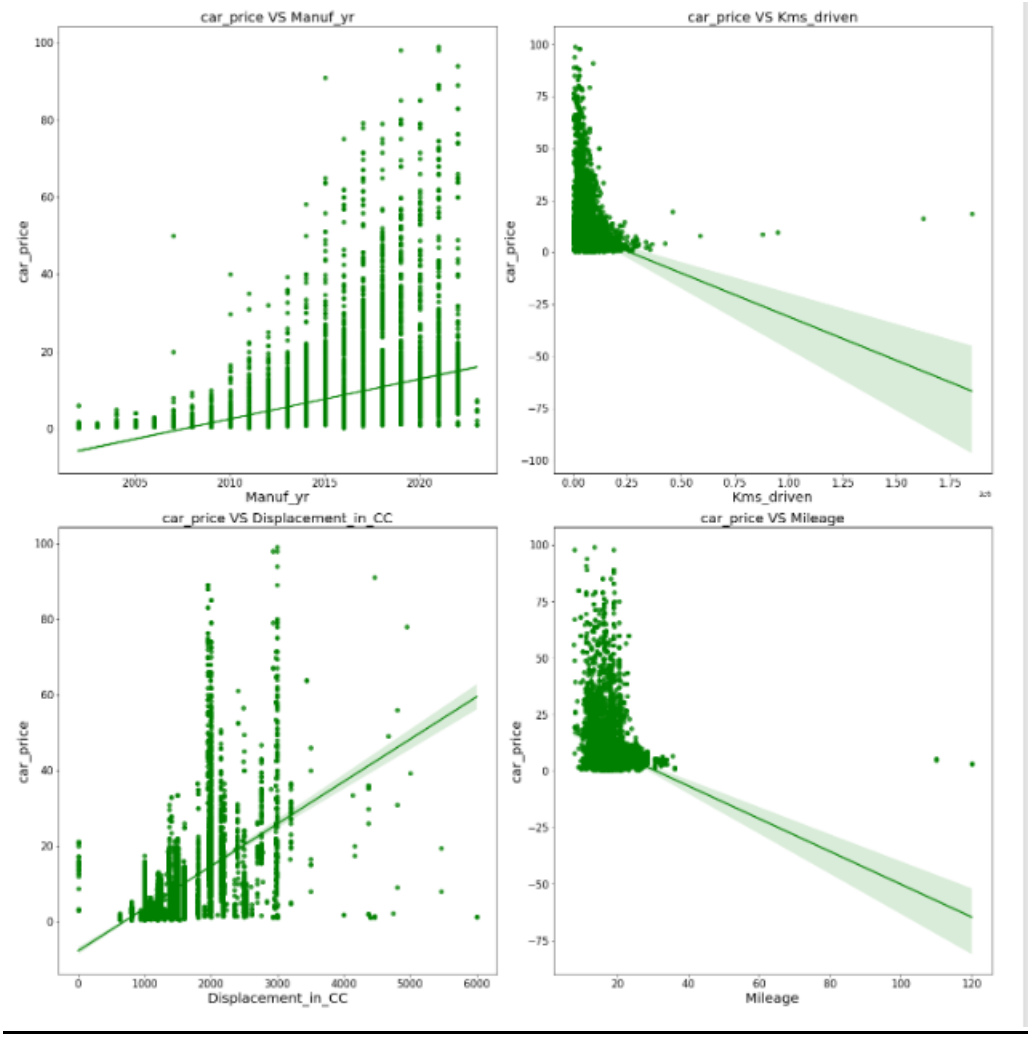


Observations:

- ✓ First Owner 7913
- ✓ -Second Owner 2527
- ✓ Third Owner 612
- ✓ Fourth Owner 128
- ✓ Fifth Owner 27

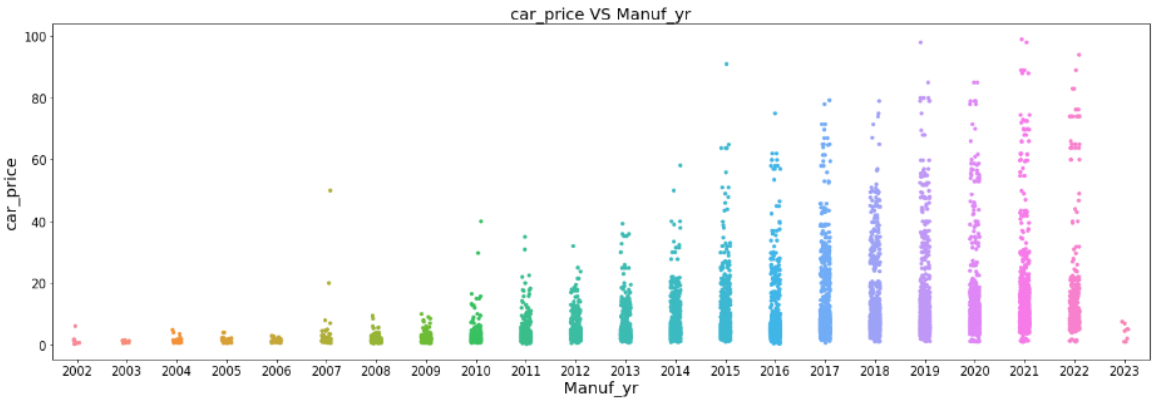
we can observe the max data is for First Owned cars with least data for Fifth Owned cars in the dataset

3. Bivariate analysis for numerical columns:



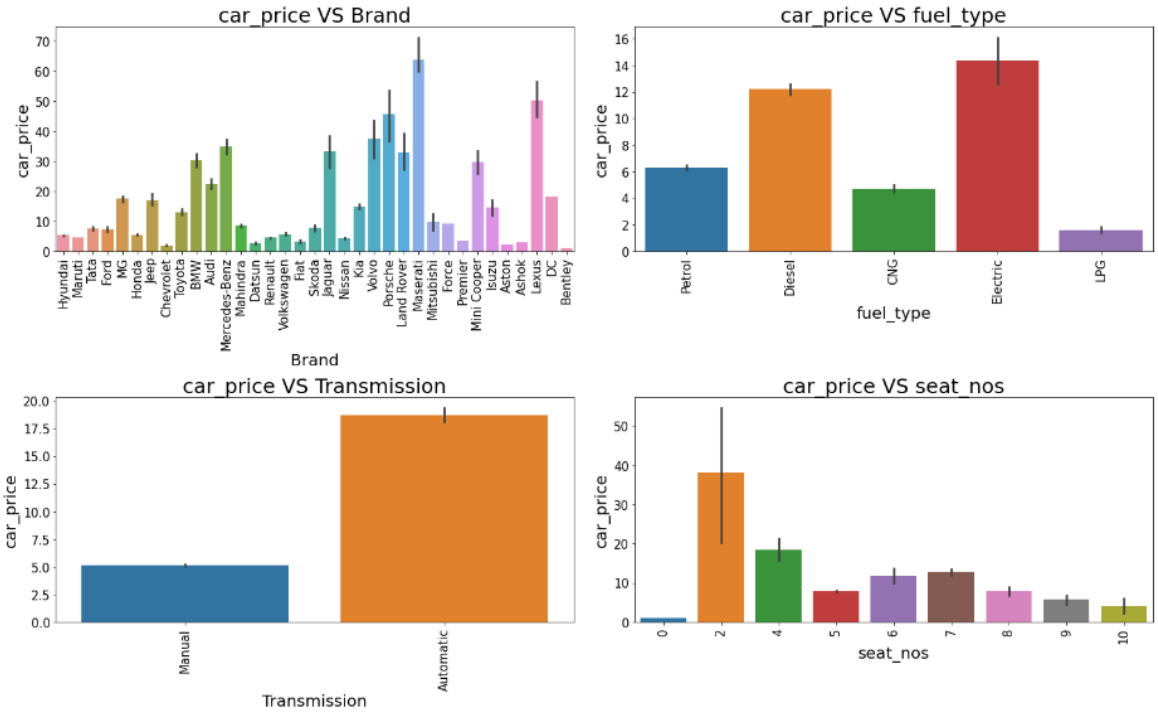
Observations:

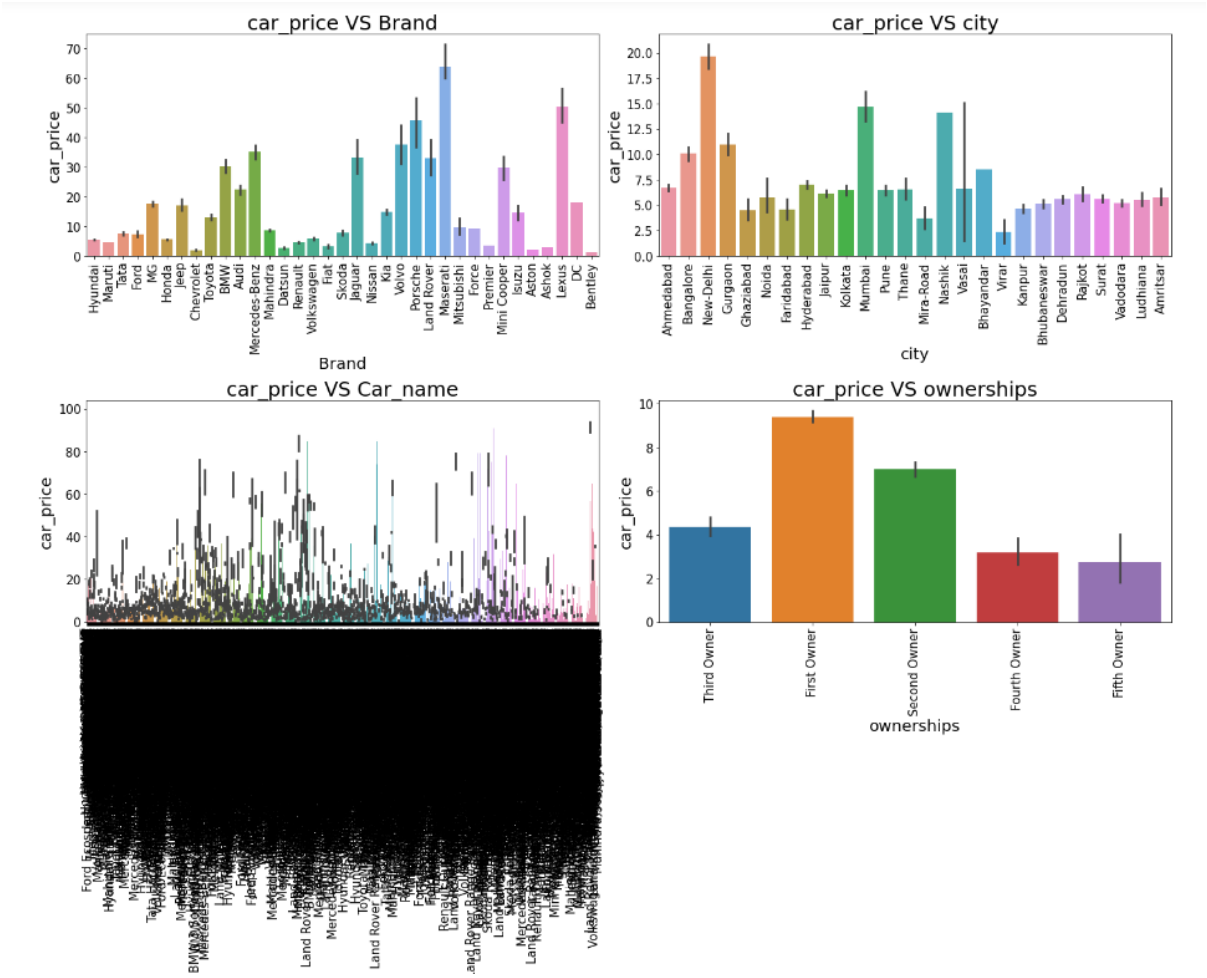
- ✓ we can see increase in price with the manufacturing years increases in the dataset, We can see a linear relationship among the dataset.We can see high increase in the numbser of cars manufactured from 2015 to 2020 in the dataset.
- ✓ As the car price increases the mileage of the cars are decreasing .We can see high amount of the cars are having min mileage in the dataset with less than 20.
- ✓ we can see increase in price with the Displacement in cc increases in the dataset, We can see a linear relationship among the dataset.Max dataset is in cars with cc of 2000 in teh dataset.
- ✓ As the car price increases the kms driven by the cars are decreasing .We can see least mk driven for high cost cars in the dataset.



- ✓ we can see increase in price with the manufacturing years increases in the dataset, We can see a linear relationship among the dataset.
- ✓ We can see high increase in the numbser of cars manufactured from 2015 to 2020 in the dataset.

4. Bivariate Analysis for categorical columns:

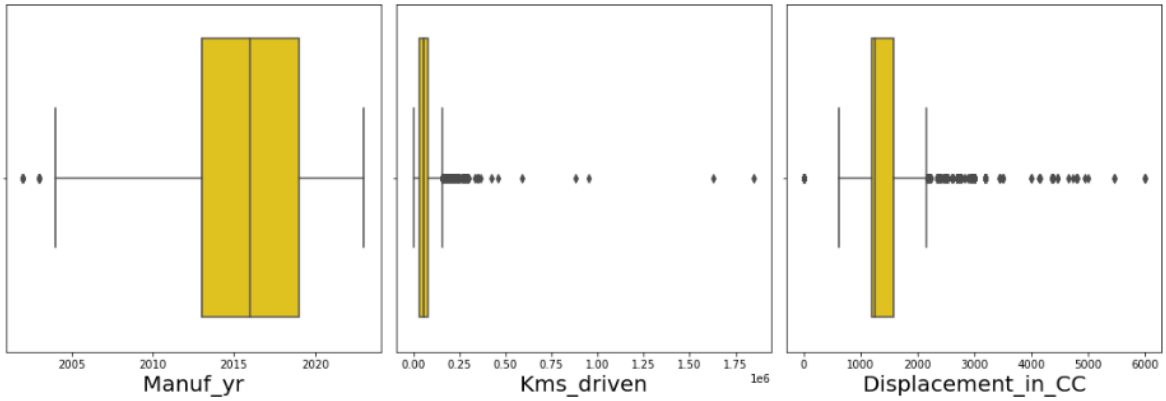




Observations:

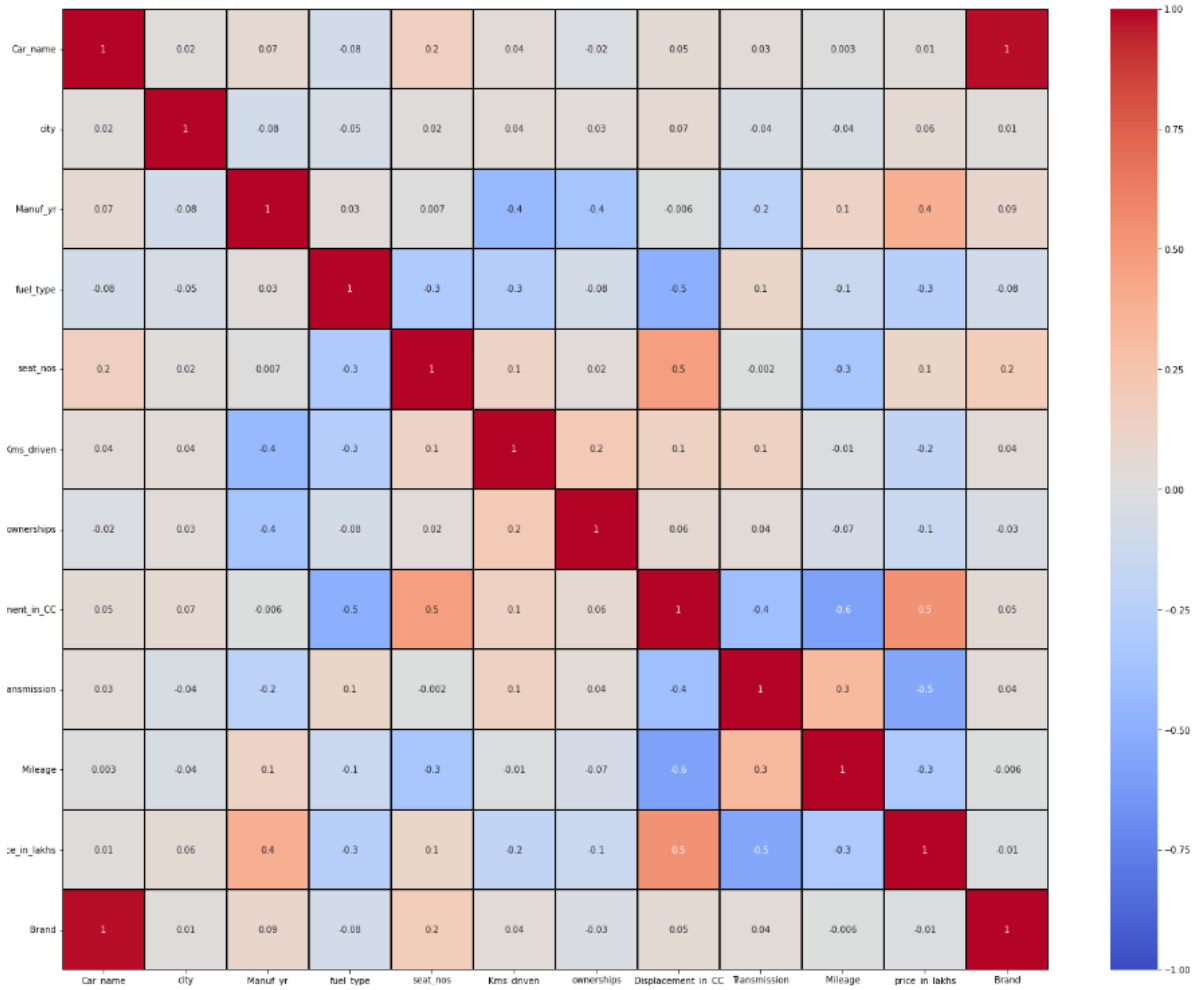
- ✓ We can see highest car price for the maserati brand cars in the dataset and second highest car price for the Lexus brand and least for Bentley among the dataset
- ✓ We can see highest car price is for Electric cars among the dataset second highest price for the Diesel Fuel and least is for Lpg Fuel types
- ✓ We can see highest car price is for Automatic transmission in cars among the dataset
- ✓ We can see highest car price is for twoseater cars among the dataset and second highest car price is for four seater cars
- ✓ We can see highest car price is for new delhi location cars among the dataset and second highest car price is for mumabi cars and least is for virar location in the dataset
- ✓ We can see that the as the Number of owner ship increases the price decreases and the highest price is for First owned cars

Checking for outliers



We can see the outliers in manufacturing years and Mileage and Displacement_in_CC columns in the dataset

Heatmap

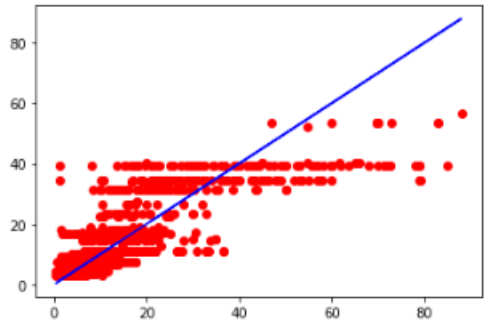


Run and Evaluate selected models

1. Model Building:

1) AdaBoostRegressor

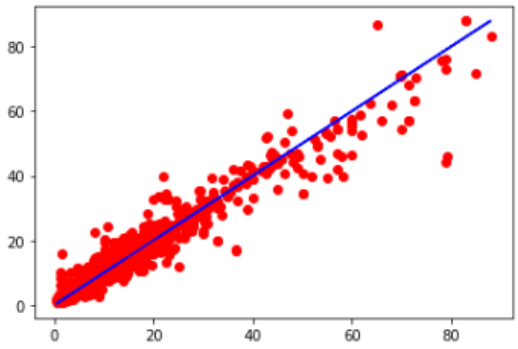
```
AdaBoostRegressor()
r2_score for train data is 71.24%


r2_score for  test data is  72.22%


Error
mean absolute error :  2.9425130565931368
mean squared error :  29.45562465071421
mean squared error is:  5.427303626177018

 AdaBoostRegressor() Cross val score is  [0.68916246 0.24936528 0.65184113 0.67735693 0.74079508]
mean is  60.17041781045904


difference b/w accuracy and crossval score is 12.04902606786527
```



AdaBoostRegressor has given me 71.24% r2_score and the difference between r2_score and cross validation score is 12.04

2) **RandomForestRegressor:**

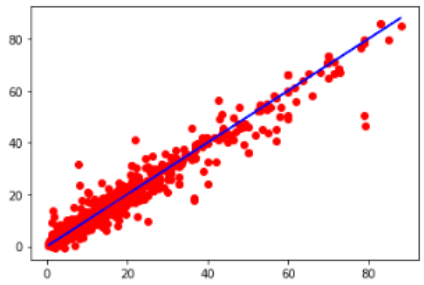
```
RandomForestRegressor()
r2_score for train data is 98.89%


r2_score for  test data is  95.19%


Error
mean absolute error :  1.072961601942451
mean squared error :  5.100166386661839
mean squared error is:  2.2583547964529043

 RandomForestRegressor() Cross val score is  [0.91825538 0.90382843 0.90538572 0.94232147 0.95088591]
mean is  92.41353836842426


difference b/w accuracy and crossval score is 2.776329094423531
```



- RandomForestRegressor has given me 95.19% r2_score and the difference between r2_score and cross validation score is 2.77%, but still we have to look into multiple models.

3) XGBRegressor:

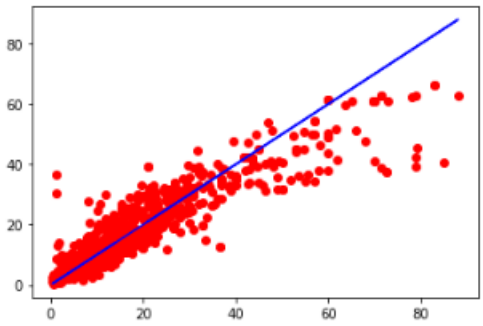
```
XGBRegressor(base_score=0.5, booster='gbtree', callbacks=None,
             colsample_bylevel=1, colsample_bynode=1, colsample_bytree=1,
             early_stopping_rounds=None, enable_categorical=False,
             eval_metric=None, gamma=0, gpu_id=-1, grow_policy='depthwise',
             importance_type=None, interaction_constraints='',
             learning_rate=0.300000012, max_bin=256, max_cat_to_onehot=4,
             max_delta_step=0, max_depth=6, max_leaves=0, min_child_weight=1,
             missing=nan, monotone_constraints='()', n_estimators=100, n_jobs=0,
             num_parallel_tree=1, predictor='auto', random_state=0, reg_alpha=0,
             reg_lambda=1, ...)
r2_score for train data is 99.37%


r2_score for  test data is  96.08%


Error
mean absolute error :  0.9910865602792218
mean squared error :  4.156682288976304
mean squared error is:  2.03879432238181

 XGBRegressor(base_score=0.5, booster='gbtree', callbacks=None,
             colsample_bylevel=1, colsample_bynode=1, colsample_bytree=1,
             early_stopping_rounds=None, enable_categorical=False,
             eval_metric=None, gamma=0, gpu_id=-1, grow_policy='depthwise',
             importance_type=None, interaction_constraints='',
             learning_rate=0.300000012, max_bin=256, max_cat_to_onehot=4,
             max_delta_step=0, max_depth=6, max_leaves=0, min_child_weight=1,
             missing=nan, monotone_constraints='()', n_estimators=100, n_jobs=0,
             num_parallel_tree=1, predictor='auto', random_state=0, reg_alpha=0,
             reg_lambda=1, ...) Cross val score is  [0.92872389 0.89330311 0.91479389 0.95071731 0.94659882]
mean is  92.68274018104853


difference b/w accuracy and crossval score is 3.3969577721204303
```



- XGBRegressor is giving me 96.08% r2_score and the difference between r2_score and cross validation score is 3.39%.

4) GradientBoostingRegressor:

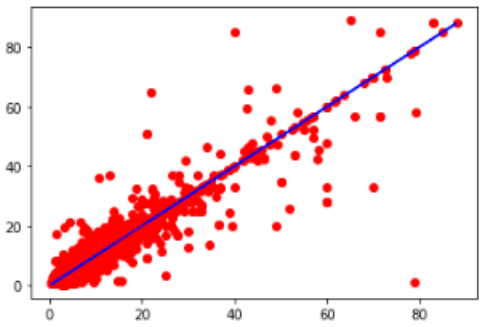
```
GradientBoostingRegressor()
r2_score for train data is 89.60%


r2_score for  test data is  88.19%


Error
mean absolute error :  1.6909663274544375
mean squared error :  12.522199094999957
mean squared error is:  3.5386719394428128

 GradientBoostingRegressor() Cross val score is  [0.85971646 0.82592866 0.8162221  0.86248214 0.89010786]
mean is  85.08914461841918


difference b/w accuracy and crossval score is 3.1007627812526977
```



- GradientBoostingRegressor is giving me 94.53% r2_score and the difference between r2_score and cross validation score is 3.10%.

5) DecisionTreeRegressor:

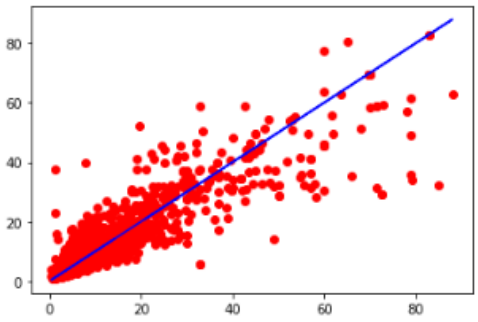
```
DecisionTreeRegressor()
r2_score for train data is 100.00%


r2_score for  test data is  88.88%


Error
mean absolute error :  1.3486306443429732
mean squared error :  11.786524460510737
mean squared error is:  3.4331508065493916

 DecisionTreeRegressor() Cross val score is  [0.82940767 0.87733358 0.77853136 0.89604632 0.92485817]
mean is  86.1235420478738


difference b/w accuracy and crossval score is 2.7602039891744283
```



- DecisionTreeRegressor is giving me 88.88% r2_score and the difference between r2_score and cross validation score is 2.76%.

6) KNeighborsRegressor:

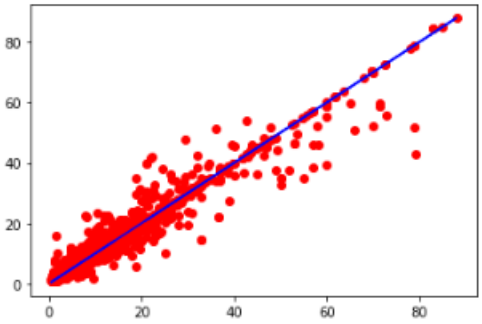
```
KNeighborsRegressor()
r2_score for train data is 88.43%


r2_score for  test data is  83.04%


Error
mean absolute error :  1.9086581430745813
mean squared error :  17.978684416438355
mean squared error is:  4.240127877368601

 KNeighborsRegressor() Cross val score is  [0.78549507 0.76775581 0.76695523 0.83654417 0.85127129]
mean is  80.16043133726957


difference b/w accuracy and crossval score is 2.8832875671043894
```



- KNeighborsRegressor is giving me 83.84% r2_score and the difference between r2_score and cross validation score is 2.88%.

ExtraTreesRegressor:

```
ExtraTreesRegressor()
r2_score for train data is 100.00%


r2_score for  test data is  94.53%


Error
mean absolute error :  1.0731404718417055
mean squared error :  5.804211470447067
mean squared error is:  2.409193116055055

 ExtraTreesRegressor() Cross val score is  [0.91432096 0.89665965 0.90863689 0.94928563 0.94965632]
mean is  92.37118908870679


difference b/w accuracy and crossval score is 2.154670593141148
```



- ✓ **By looking into the difference of r2_score and cross validation score i found** ExtraTreesRegressor **as the best model with 94.53% r2_score and the difference between r2_score and cross validation score is 2.15**

3. Hyper Parameter Tunning:

```
from sklearn.model_selection import GridSearchCV

params = {'n_estimators': [100,150,200,300],
          'criterion':['squared_error','absolute_error','friedman_mse','poisson'],
          'max_depth': [1,3, 5, 7],
          'min_samples_split': [1, 3, 5,7],
          'max_features' : ['sqrt', 'log2', 'None']
           }
```

```

```

```
#importing gridsearch
from sklearn.model_selection import GridSearchCV

Grid_sear= GridSearchCV(estimator=ExtraTreesRegressor(),param_grid=params,cv= 5,n_jobs=-1)
```

```
#training the model with paramters
Grid_sear.fit(x_train,y_train)
```

```
GridSearchCV(cv=5, estimator=ExtraTreesRegressor(), n_jobs=-1,
             param_grid={'criterion': ['squared_error', 'absolute_error',
                                       'friedman_mse', 'poisson'],
                         'max_depth': [1, 3, 5, 7],
                         'max_features': ['sqrt', 'log2', 'None'],
                         'min_samples_split': [1, 3, 5, 7],
                         'n_estimators': [100, 150, 200, 300]})
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

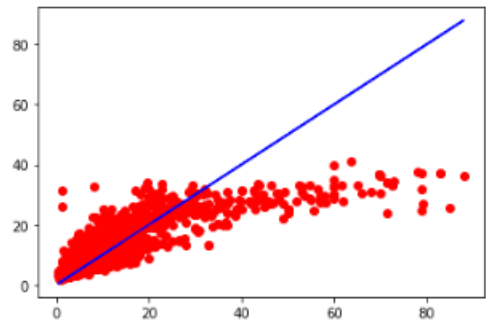
```
#Best parameteters in the hypertuned model
Grid_sear.best_params_
```

```
{'criterion': 'friedman_mse',
 'max_depth': 7,
 'max_features': 'log2',
 'min_samples_split': 5,
 'n_estimators': 150}
```

```
#training the model with gridsearch parameters

Best_mod = ExtraTreesRegressor(n_estimators=150 , criterion='friedman_mse', max_depth= 7, min_samples_split= 5,
                               max_features='log2')

```

```
r2_score for train data is 72.27%


r2_score for  test data is  69.15%


Error
mean absolute error :  2.7665044227702005
mean squared error :  32.712870386030374
mean squared error is:  5.719516621711172
```

```
[<matplotlib.lines.Line2D at 0x2241989fd60>]
```



I have choosed all parameters of DecisionTreeRegressor, after tunning the model with best parameters model accuracy decreases . the final model is selected as ExtraTreesRegressor with out hypertuning

5. Saving the model and Predictions:

- I have saved my best model as follows.

```
# Saving the model using joblib
import joblib
joblib.dump(etr,"Car_Price.obj")
```

```
['Car_Price.obj']
```

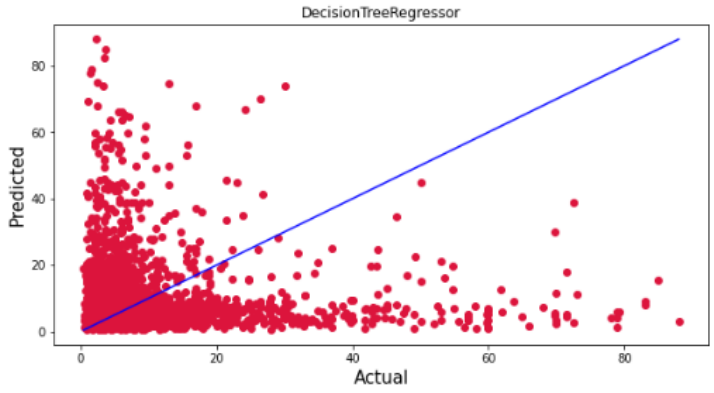
Predictions:

```
#Lets Check loading the file

Car_Price_prediction=joblib.load("Car_Price.obj")
```

```
pred=etr.predict(x_test)
Conclusion=pd.DataFrame([Car_Price_prediction.predict(x_test)[:],pred[:]],index=["Predicted","Original"])
```

- Now loading my saved model and predicting the price values.

```
plt.figure(figsize=(10,5))
plt.scatter(y_test, prediction, c='crimson')
p1 = max(max(prediction), max(y_test))
p2 = min(min(prediction), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('Actual', fontsize=15)
plt.ylabel('Predicted', fontsize=15)
plt.title("DecisionTreeRegressor")
plt.show()
```




```
Conclusion
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Predicted	12.6768	11.1517	3.0839	1.63	2.1003	13.643	24.0135	9.0153	2.6134	6.4172	1.1073	2.5995	4.7967	9.6244	6.5795	4.9305	1.1896	5.0908	1.5411
Original	12.6768	11.1517	3.0839	1.63	2.1003	13.643	24.0135	9.0153	2.6134	6.4172	1.1073	2.5995	4.7967	9.6244	6.5795	4.9305	1.1896	5.0908	1.5411

- Plotted Actual vs Predicted, To get better insight. Bule line is the actual line and red dots are the predicted values.

3.6 Interpretation of the Results

- ✓ The dataset was scrapped from cardekho website.
- ✓ The dataset was very challenging to handle it had 11 features with 11281 samples.
- ✓ Firstly, the datasets were having any null values, so I have used imputation method to replace the nan values.
- ✓ And there was number of unnecessary entries in all the features so I have used feature extraction to get the required format of variables.
- ✓ And proper ploting for proper type of features will help us to get better insight on the data. I found both numerical columns and categorical columns in the dataset so I have choosen reg plot, strip plot and bar plot to see the relation between target and features.
- ✓ I notice a huge amount of outliers and skewness in the data so we have choose proper methods to deal with the outliers and skewness. If we ignore this outliers and skewness we may end up with a bad model which has less accuracy.
- ✓ Then scaling dataset has a good impact like it will help the model not to get biased. Since we have removed outliers and skewness from the dataset so we have to choose Standardisation.
- ✓ We have to use multiple models while building model using dataset as to get the best model out of it.
- ✓ And we have to use multiple metrics like mse, mae, rmse and r2_score which will help us to decide the best model.
- ✓ I found ExtraTreeRegressor as the best model with 94.53% r2_score. Also I have improved the accuracy of the best model by running hyper parameter tunning.
- ✓ At last I have predicted the used car price using saved model. It was good!! that I was able to get the predictions near to actual values.

4.CONCLUSION

4.1 Key Findings and Conclusions of the Study

In this project report, we have used machine learning algorithms to predict the used car prices. We have mentioned the step by step procedure to analyze the dataset and finding the correlation between the features. Thus we can select the features which are correlated to each other and are independent in nature. These feature set were then given as an input to five algorithms and a hyper parameter tunning was done to the best model and the accuracy has been improved. Hence we calculated the performance of each model using different performance metrics and compared them based on those metrics. Then we have also saved the best model and predicted the used car price. It was good the the predicted and actual values were almost same.

4.2 Learning Outcomes of the Study in respect of Data Science

I found that the dataset was quite interesting to handle as it contains all types of data in it and it was self scrapped from cardekho website using selenium. Improvement in computing technology has made it possible to examine social information that cannot previously be captured, processed and analysed. New analytical techniques of machine learning can be used in used car price research. The power of visualization has helped us in understanding the data by graphical representation it has made me to understand what data is trying to say. Data cleaning is one of the most important steps to remove unrealistic values and null values. This study is an exploratory attempt to use five machine learning algorithms in estimating used car price prediction, and then compare their results.

To conclude, the application of machine learning in predicting used car price is still at an early stage. We hope this study has moved a small step ahead in providing some methodological and empirical contributions to crediting online platforms, and presenting an alternative approach to the valuation of used car price. Future direction of research may consider incorporating additional used car data from a larger economical background with more features.

4.3 Limitations of this work and Scope for Future Work

- ✓ First draw back is scrapping the data as it is fluctuating process.
- ✓ Followed by more number of outliers and skewness these two will reduce our model accuracy.
- ✓ Also, we have tried best to deal with outliers, skewness and null values. So it looks quite good that we have achieved a accuracy of 94.53% even after dealing all these drawbacks.
- ✓ Also, this study will not cover all Regression algorithms instead, it is focused on the chosen algorithm, starting from the basic ensembling techniques to the advanced ones.