Project Report

On

Car Price Prediction

ACKNOWLEDGEMENT

The following research papers helped me understand the problem of car prices, the various factors affecting the pricing of a particular car, fluctuations in car prices and their causes & finally, helped me in my model building & predictions.

The production of cars has been steadily increasing in the past decade, with over 70 million passenger cars being produced in the year 2016. This has given rise to the used car market, which on its own has become a booming industry. The recent advent of online portals has facilitated the need for both the customer and the seller to be better informed about the trends and patterns that determine the value of a used car in the market. Using Machine Learning Algorithms such as Lasso Regression, Multiple Regression and Regression trees, we will try to develop a statistical model which will be able to predict the price of a used car, based on previous consumer data and a given set of features.

INTRODUCTION

Business Problem Framing With the covid 19 impact in the market, we have seen a 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. Predicting the price of used cars is an important and interesting problem. Predicting the resale value of a car is not a simple task. It is trite knowledge that the value of used cars depends on a number of factors. The most important ones are usually the age of the car, its make (model), the origin of the car (location of the manufacturer), its mileage (the number of kilometers it has run) and its horsepower (amount of power that the engine produces). 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 a car price valuation model.

ANALYTICAL PROBLEM FRAMING

Mathematical/Analytical Modeling of the Problem

As a first step I have scrapped the required data from cardekho.com website. I have fetched data for different locations and saved it to excel format & csv format. In this particular problem I have “Price” as my target column and it was a continuous column. So clearly it is a regression problem and I have to use all regression algorithms while building the model.

- There were no null values in the dataset.
- Since we have scrapped the data from cardekho.com website the raw data was not in the format, so we have used feature engineering to extract the required feature format.
- To get better insight on the features I have used plotting like distribution plot, bar plot, reg plot, pie plot, scatter plot and count plot. With these plots I was able to understand the relation between the features in a 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 models, then tuned the best model and saved the best model. At last I have predicted the car-price using the saved model.

Data preprocessing

- As a first step I have scrapped the required data using selenium from cardekho.com website.
- And I have imported required libraries and I have imported the dataset which was saved in csv format.
- Then I did all the statistical analysis like checking shape, columns, data types, nunique, value counts, info etc.
- While checking the data types of the columns I found all the columns to be of object data type, so I changed the data types of the columns to the correct data types.
- I also extracted the car manufacturing year from the brand column using feature extraction techniques.
- While checking for null values I found no null values in the dataset.
- There were no empty values present in the dataset.
- Performed univariate, bivariate and multivariate analysis to visualize the data. Visualized each feature using seaborn and matplotlib libraries by plotting several categorical and numerical plots like pie plot, count plot, bar plot, distribution plot, box plots and pair plot.
- Next, I used Label Encoder to encode all the object data type columns to make it easier for model building and visualization.

● Identified outliers using box plots and removed outliers in columns using Z Score Method and stored the data frame after removing outliers as “df_usedcars”.
● Checked for skewness and removed skewness in numerical columns using power transformation method (yeo-johnson).
● Separated features and target variables and conducted feature scaling using Standard Scaler method to avoid any kind of data biasness.
● Checked Variance Inflation Factor (VIF) got rid of high multicollinearity issues if present.

Testing of Identified Approaches (Algorithms)

Since Price was my target and it was a continuous column so this particular problem was a regression problem. And I have used all regression algorithms to build my model. By looking into the r2 score and cross validation score I found ExtraTreesRegressor as a best model with high scores. Also to get the best model we have to run through multiple models and to avoid the confusion of overfitting we have to go through cross validation. Below is the list of regression algorithms I have used in my project.

1. LinearRegression
2. SVR
3. Ridge Regression
4. RandomForestRegressor
5. GradientBoostingRegressor
6. DecisionTreeRegressor

Linear Regression

```
from sklearn.linear_model import LinearRegression



LR=LinearRegression()
LR.fit(x_train,y_train)
print(LR.score(x_train,y_train))
LR_predict=LR.predict(x_test)
```

0.0761024892057085

In [48]:

```
print('MSE:',mean_squared_error(LR_predict,y_test))
print('MAE:',mean_absolute_error(LR_predict,y_test))
print('r2_score:',r2_score(LR_predict,y_test))
```

MSE: 819842.20784503
MAE: 745.610952005011
r2_score: -9.976092404043055

Ridge Regression

In [49]:

```
from sklearn.linear_model import Ridge

R=Ridge()
R.fit(x_train,y_train)
print(R.score(x_train,y_train))
R_predict=R.predict(x_test)
```

0.07609397122139971

In [50]:

```
print('MSE:',mean_squared_error(R_predict,y_test))
print('MAE:',mean_absolute_error(R_predict,y_test))
print('r2_score:',r2_score(R_predict,y_test))
```

MSE: 819961.9592567373
MAE: 745.982241612618
r2_score: -10.164583526816866

SVR(Support Vector Regression)

In [51]:

```
from sklearn.svm import SVR

svr=SVR(kernel='linear')
svr.fit(x_train,y_train)
print(svr.score(x_train,y_train))
svr_predict=svr.predict(x_test)
```

0.02761569815541165

In [52]:

```
print('MSE:',mean_squared_error(svr_predict,y_test))
print('MAE:',mean_absolute_error(svr_predict,y_test))
print('r2_score:',r2_score(svr_predict,y_test))
```

MSE: 875626.4835046877
MAE: 793.7301916363455
r2_score: -105.43499712347014

In [53]:

```
svr_p=SVR(kernel='poly')
svr_p.fit(x_train,y_train)
print(svr_p.score(x_train,y_train))
svrpred_p=svr_p.predict(x_test)
```

0.06930514596887327

In [54]:

```
print('MSE:',mean_squared_error(svrpred_p,y_test))
print('MAE:',mean_absolute_error(svrpred_p,y_test))
print('r2_score:',r2_score(svrpred_p,y_test))
```

MSE: 839816.4599106477
MAE: 734.6406688565291
r2_score: -5.94975398542537

In [55]:

```
svr_r=SVR(kernel='rbf')
svr_r.fit(x_train,y_train)
print(svr_r.score(x_train,y_train))
svrpred_r=svr_r.predict(x_test)
```

0.036085277482763356

Random Forest Regressor

In [56]:

```
from sklearn.ensemble import RandomForestRegressor

RF=RandomForestRegressor()
RF.fit(x_train,y_train)
print(RF.score(x_train,y_train))
RF_PRED=RF.predict(x_test)

print('MSE:',mean_squared_error(RF_PRED,y_test))
print('MAE:',mean_absolute_error(RF_PRED,y_test))
print('r2_score:',r2_score(RF_PRED,y_test))
```

0.9820970556486404
MSE: 128318.14848301114
MAE: 124.1008066914498
r2_score: 0.8349720309428137

Decision Tree Regressor

In [57]:

```
from sklearn.tree import DecisionTreeRegressor

DTR=DecisionTreeRegressor()
DTR.fit(x_train,y_train)
print(DTR.score(x_train,y_train))
DTR_PRED=DTR.predict(x_test)

print('MSE:',mean_squared_error(DTR_PRED,y_test))
print('MAE:',mean_absolute_error(DTR_PRED,y_test))
print('r2_score:',r2_score(DTR_PRED,y_test))
```

1.0
MSE: 187989.59888475836
MAE: 118.79591078066915
r2_score: 0.791876189179938

Gradient Boosting Regressor

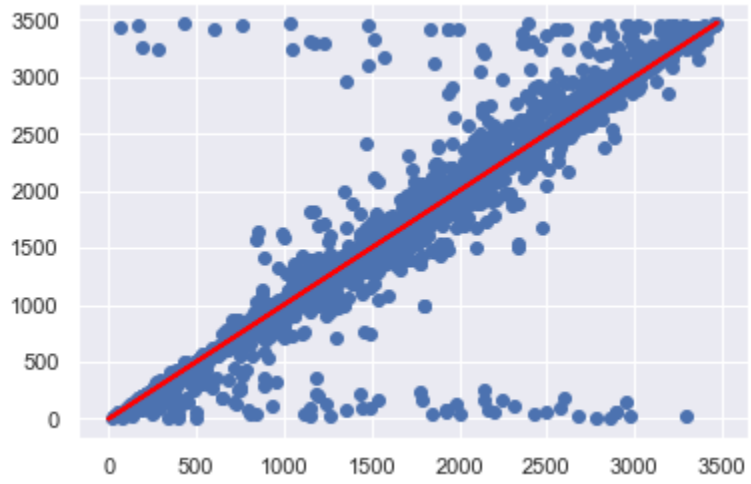
In [58]:

```
from sklearn.ensemble import GradientBoostingRegressor

GBR=GradientBoostingRegressor()
GBR.fit(x_train,y_train)
print(GBR.score(x_train,y_train))
GBR_PRED=GBR.predict(x_test)

print('MSE:',mean_squared_error(GBR_PRED,y_test))
print('MAE:',mean_absolute_error(GBR_PRED,y_test))
print('r2_score:',r2_score(GBR_PRED,y_test))
```

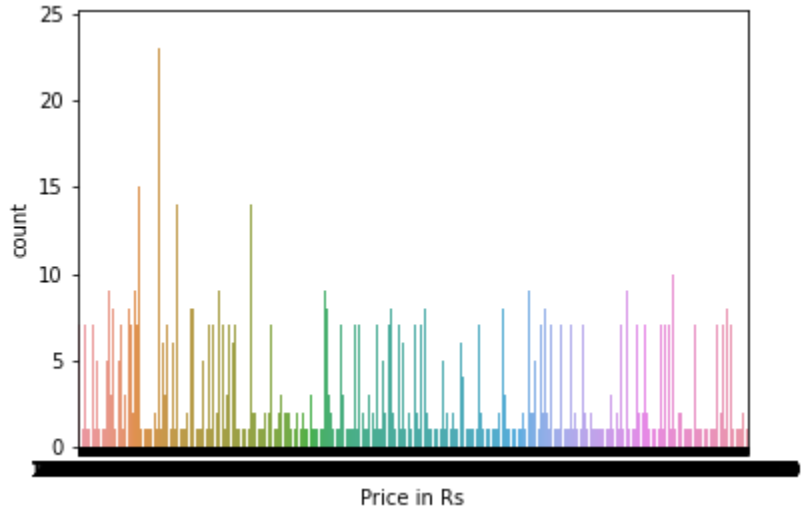
0.633936198390313
MSE: 357572.4820846948
MAE: 408.95878203223714
r2_score: 0.06346026005664829

Out[65]:

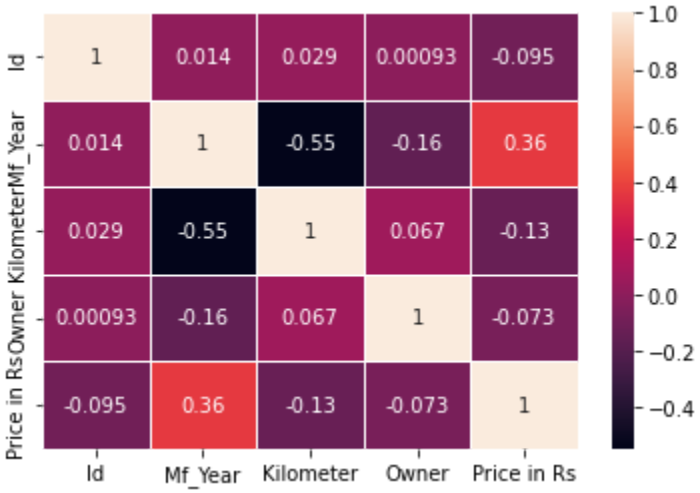
[<matplotlib.lines.Line2D at 0x7fd2f0e58850>]

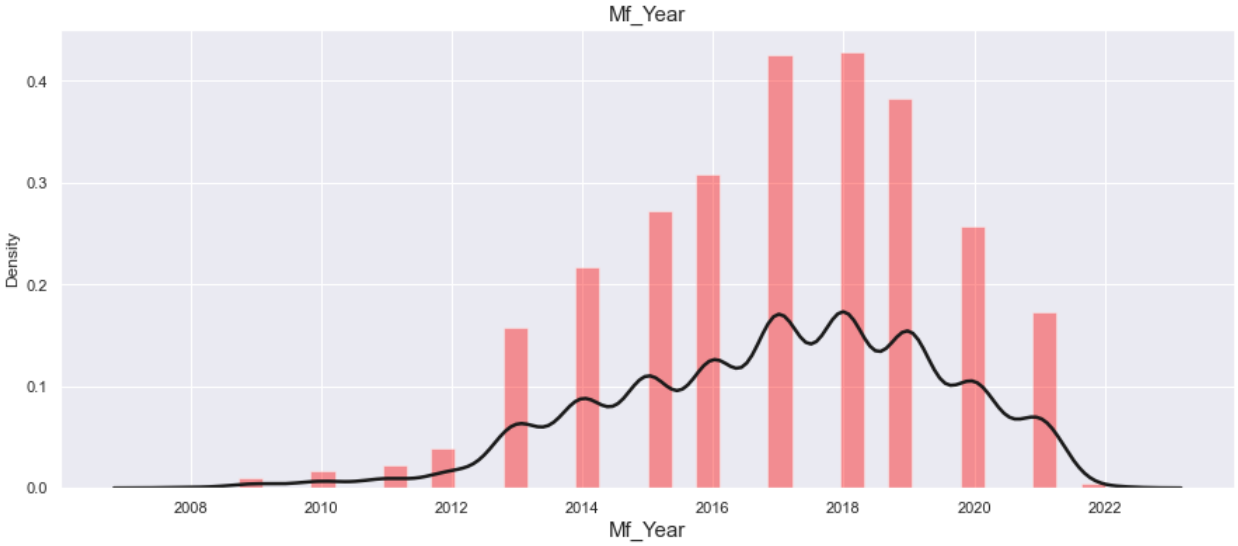


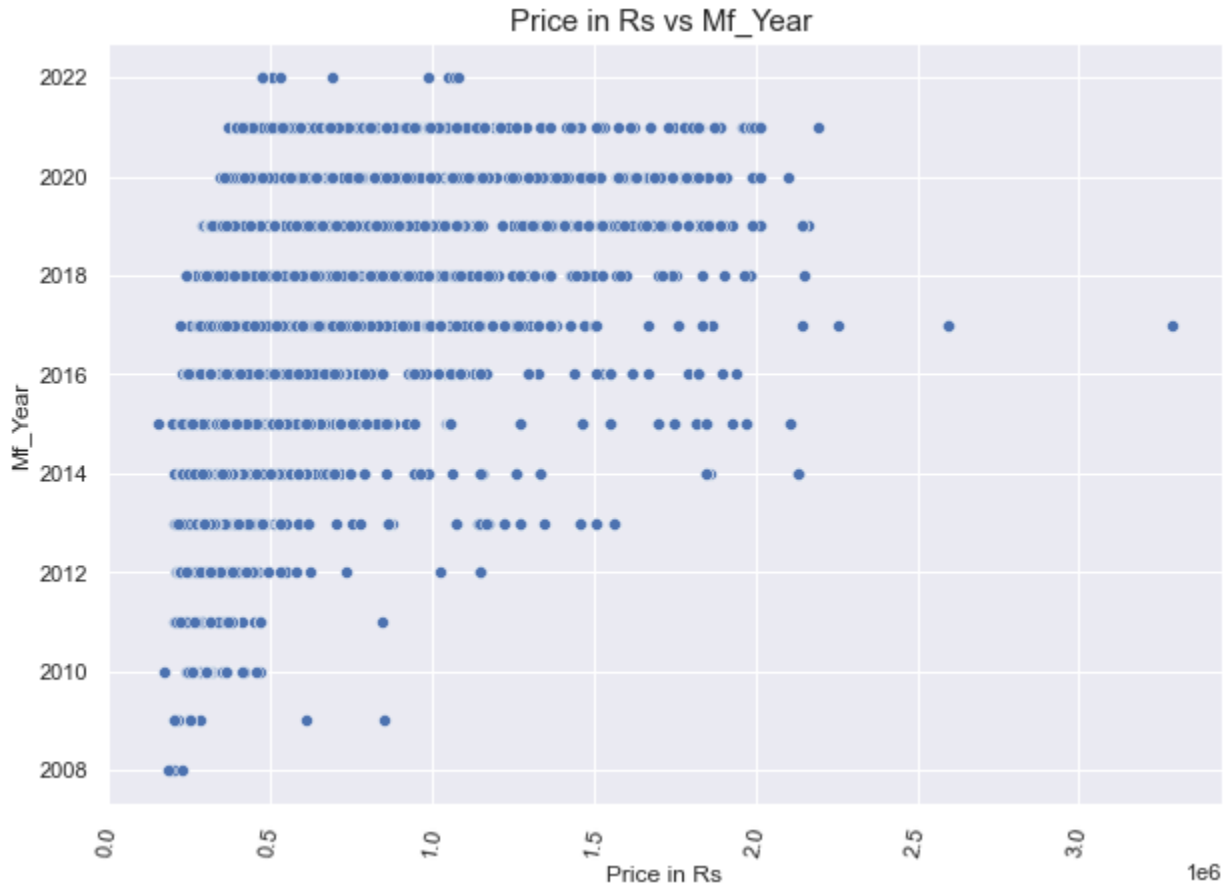
Visualization



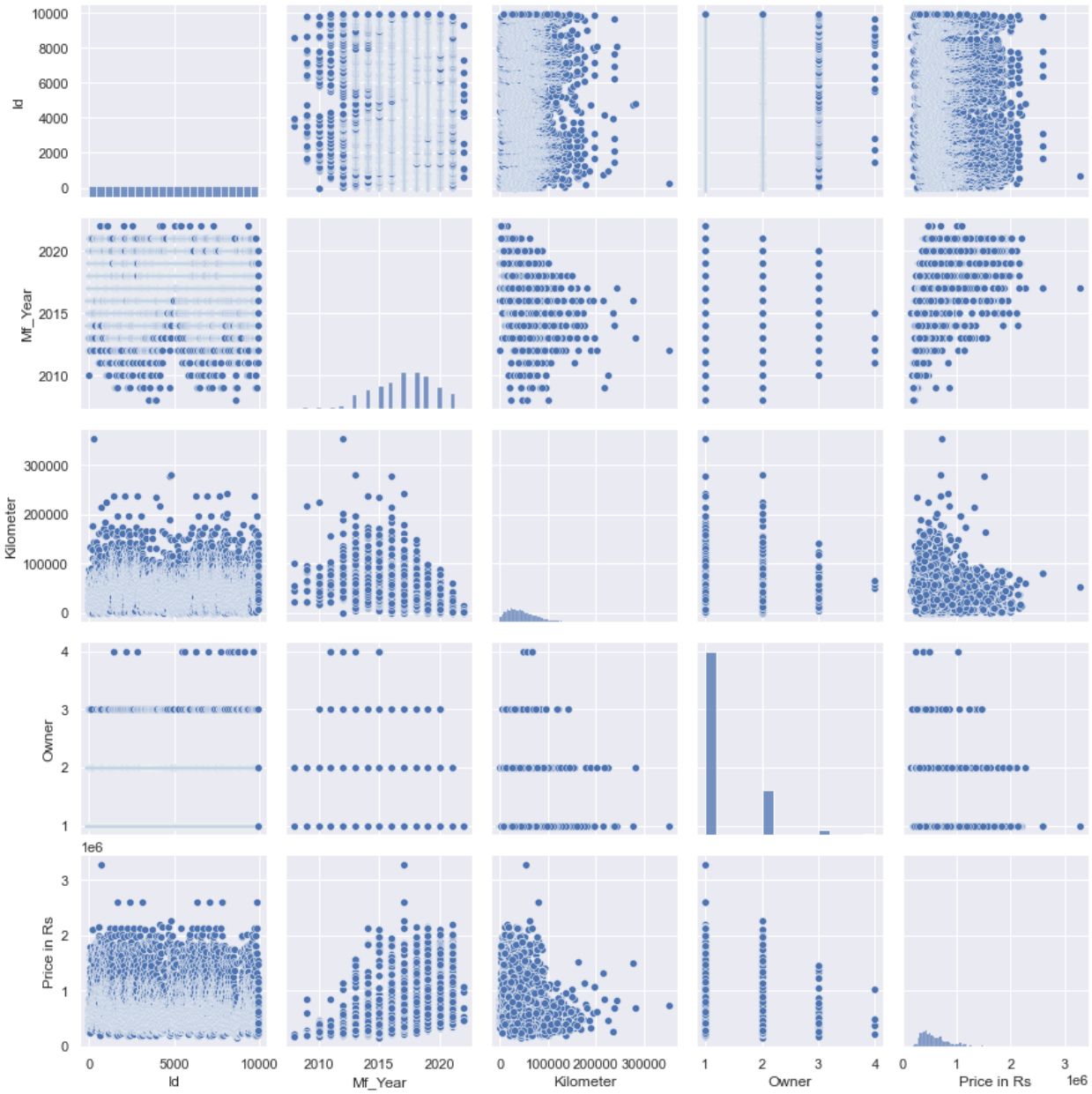
Heatmap




Mf_Year
Density
0.0
0.1
0.2
0.3
0.4
2008
2010
2012
2014
2016
2018
2020
2022
Mf_Year


Price in Rs vs Mf_Year
Mf_Year
2022
2020
2018
2016
2014
2012
2010
2008
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Price in Rs
1e6

Multivariate analysis



CONCLUSION

Key Findings and Conclusions of the Study In this project report, we have used machine learning algorithms to predict the prices of used cars. After the completion of this project, we got an insight of how to collect data, pre-processing of the data, analyzing the data, cleaning the data and building a model. In this study, we have used multiple machine learning models to predict the sale price of the used cars. We have gone through the data analysis by performing feature engineering, finding the relation between features and target through visualizations. And got the important features and we used these features to predict the car price by building ML models. After training the model we checked CV score to overcome the overfitting issue. Performed hyper parameter tuning on the best model and the best model's R2 score increased and was giving R2 score as 99.97 %. We have also got good prediction results of car prices. Learning Outcomes of the Study in respect of Data Science While working on this project I learned many things about the features of cars and about the various car selling platforms.

By building the machine learning models have helped to predict the price of used cars which provides greater understanding into the many advantages & disadvantages of selling old cars. I found this problem to be quite interesting to handle as it contains all types of data in it and that the data had to be scraped from cardekho.com website using selenium. The data visualization has helped us in understanding the data by graphical representation. It has made me understand what data is trying to say. This study is an exploratory attempt to use 8 machine learning algorithms in estimating housing prices, and then compare their results. Finally, our aim was achieved by predicting the sale price of used cars and building a car price prediction model that could help clients to understand the future price of used cars. New analytical techniques of machine learning can be used in used car price research. Limitations of this work and Scope for Future Work