A Case study of Housing Price Prediction

Submitted by:

Sharad Yadav

ACKNOWLEDGMENT

Data fetch by selenium web scrapping from car 24 website, analysis done by sharad Yadav under guidance of Mr. Keshav Bansal, articles content written by myself regarding used car price prediction

INTRODUCTION

- ## Business Problem Framing

  Metropolitan city are major user of car because of earning level in increased after covid most of the people prefer to use their own vehicle instead of public transport .So here I have taken data from Car 24 website with help of web scrapping techniques (selenium) to predict used car Price only Delhi NCR region

- ## Conceptual Background of the Domain Problem

- India's online (Car 24,Car dekho, Spinny etc)used car market is revving up, with companies and investors jostling to grab a larger piece of the pie. The potential of growth in the country of 1.3 billion people is huge as the percentage of car ownership is still in single digits.

- ## Review of Literature

  In this dataset, there are 700 observations with 12 explanatory variables describing (almost) every aspect of car brand, History, Owner, Kilometers, Fuel_type, Last_service, Transmission ,Registration, Insurance ,Year_of_manufacturing among explanatory variables. Descriptive analysis and quantitative analysis will use subsets of it depending on models.

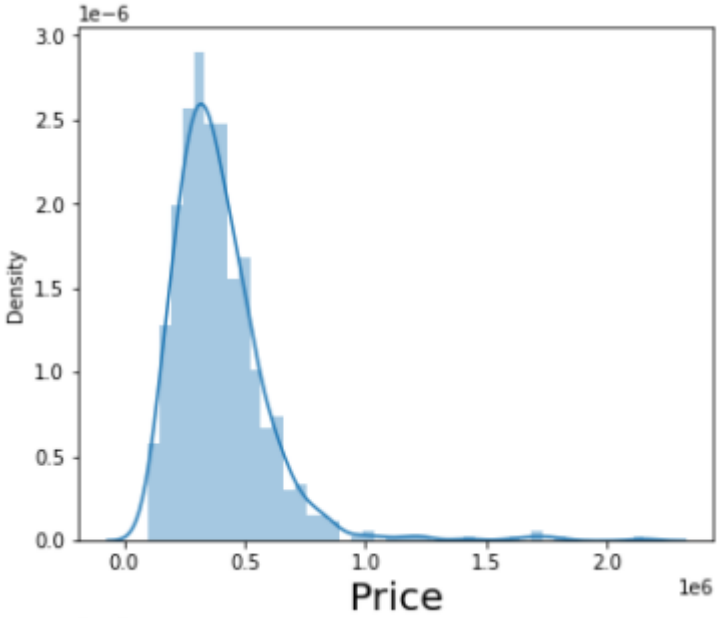
- ## Motivation for the Problem Undertaken

  Urbanization of city is key reason to people used to personal vehicle, after COVID-19 pandemic had a minimal impact on the industry. With the increased number of people preferring individual mobility and more finance options infused into the used car market, the market is set to grow considerably. Reduced cash inflow due to the pandemic has forced buyers to look for alternatives other than new cars, and the used car industry has great growth potential in these terms. With the sales and production of new vehicles hindered due to the pandemic, the immediate option for the buyers is the used car market.

Analytical Problem Framing

- Mathematical/ Analytical Modelling of the Problem

  In this project we have to predict used car price after plotting distribution plot of sales price data is right skewed which sold at higher price than average price, higher price of car depend upon various features



  But most importantly vehicle is how much year old, but here selenium code is time consuming which takes more than 1 days to fetch data. I am only able to fetch 700 car data

- **Data Sources and their formats**
- Dataset is in csv file, to read that data I have to used pandas library to read file further describe method to analysis get overview of data distribution, data info to identify object integer float data types

```
#import imp library
import pandas as pd
import re
import numpy as np
import seaborn as sns
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split ,GridSearchCV
from sklearn.metrics import r2_score,mean_squared_error,mean_absolute_error
import statsmodels as sm
from sklearn.model_selection import cross_val_score
import matplotlib.pyplot as plt
import scikitplot as skplt
import warnings
import math
warnings.filterwarnings('ignore')
```

```
data=pd.read_csv(r"C:\Users\INPshy\Desktop\DATA Science\Used car.csv")
data
```

:

	Unnamed: 0	Brand	Price	History	Owner	Kilometers	Fuel_Type	Last_service	Transmission	Registration	Insurance	Year Manufacturir
0	0	2020 Maruti Swift LXI MANUAL	₹ 5,66,399	Non-Accidental	1st Owner	1,755 km	Petrol	Petrol	DL-3C-x-xxxx	Valid upto Mar 2022\nZero_Dep	February 2020	
1	1	2015 Maruti Alto K10 LXI MANUAL	₹ 2,67,799	Non-Accidental	1st Owner	10,957 km	Petrol	Petrol	MANUAL	HR-51-x-xxxx	Valid upto Sep 2022\n3rd Party	
2	2	2020 Maruti Swift LXI MANUAL	₹ 5,59,499	Non-Accidental	1st Owner	3,570 km	Petrol	Petrol	MANUAL	DL-2C-x-xxxx	Valid upto Feb 2022\nZero_Dep	

- **Data Preprocessing Done**

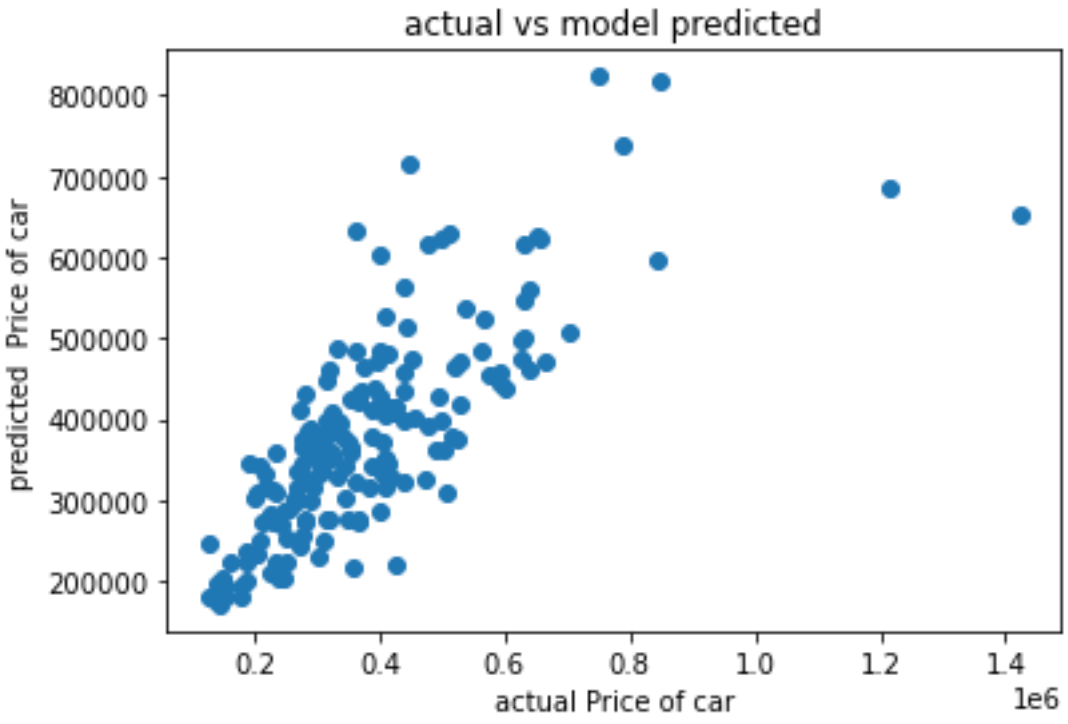
  First import important library then describe method to see data distribution, check null values, data types,data info, data.shape pie chart analysis of object data which factors have more effects on Price prediction i.e. noted down on notebook itself. Filling miss values, convert kilometres and price in to numeric values after removing text,encode object data with Ordinal encode after that distribution plot to check how data distributed ,box plot to check outliers in dataset, heatmap to check multicollnearity, scatter plot to check correlation between features ,correlation plot to check how features have correlation with Sale Price ,features selection ,train test split after that model building and tune the model and cross validation ,visualization of actual sale price vs predicted sale price with scatter plot on different model and XGB model found best

performer among all to save for future analysis of these dataset.

- **Data Inputs- Logic- Output Relationships**

  In this data set 12 columns including target variable i.e. Sale Price ,after cleaning and pre-processing ,visualization of which features have strong positive and negative relationship with target variable

  Predicting CAR Sale price is regression type problem ,building model I have predicted sale price and compare to actual sale price i.e. look like as shown in image below:-



- Hardware and Software Requirements and Tools Used

```
# lets start with importing necessary library
import numpy as np
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
import statsmodels as sm
from sklearn.metrics import r2_score,mean_absolute_error,mean_squared_error
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
import seaborn as sns
import pickle
import warnings
warnings.filterwarnings("ignore")
```

Screenshot of imported library used to build predictive model

Pandas library used to read csv file,pie chart analysis, distribution plot to check how data distributed, standardscaler to scale features traintestsplit dataset,sklearn.metrics used to check model accuracy and other parameters evaluation ,matplotlib used for visualization. Different model imported from sklearn libray to build model like Linear ,decision tree,GradientBoost Regression,RandomForest Regression,XGB model .

Model/s Development and Evaluation

- **Identification of possible problem-solving approaches (methods)**

  First step to read csv file, check data shape, missing values, describe method to view data mean, median, mode, std, max data, data types, all columns are object type then it converted to numerical value with help of pandas strip function and encoding techniques. After encoding plot distplot of each column to see how data distributed

```
data=pd.read_csv(r"C:\Users\INPshy\Desktop\DATA Science\Used car.csv")
data
```

	Unnamed: 0	Brand	Price	History	Owner	Kilometers	Fuel_Type	Last_service	Transmission	Registration	Insurance	Year Manufacturir
0	0	2020 Maruti Swift LXI MANUAL	₹ 5,66,399	Non-Accidental	1st Owner	1,755 km	Petrol	Petrol	DL-3C-x-xxxx	Valid upto Mar 2022\nZero_Dep	February 2020	
1	1	2015 Maruti Alto K10 LXI MANUAL	₹ 2,67,799	Non-Accidental	1st Owner	10,957 km	Petrol	Petrol	MANUAL	HR-51-x-xxxx	Valid upto Sep 2022\n3rd Party	
2	2	2020 Maruti Swift LXI MANUAL	₹ 5,59,499	Non-Accidental	1st Owner	3,570 km	Petrol	Petrol	MANUAL	DL-2C-x-xxxx	Valid upto Feb 2022\nZero_Dep	
3	3	2020 Maruti S PRESSO VXI	₹ 3,98,999	Non-Accidental	1st Owner	3,176 km	Petrol	Petrol	NaN	HR-51-x-xxxx	Valid upto Jan 2022\nZero_Dep	
4	4	2020 Renault TRIBER RXZ AT AUTOMATIC	₹ 6,41,499	Non-Accidental	1st Owner	6,460 km	Petrol	Petrol	AUTOMATIC	PB-10-x-xxxx	Valid upto Sep 2022\n3rd Party	

Reading of csv file with help of pandas library

- Testing of Identified Approaches (Algorithms)

1. from sklearn.linear_model import LinearRegression
2. from sklearn.ensemble import RandomForest Regression
3. from sklearn.tree import DecisionTreeRegressor
4. from sklearn.ensemble import GradientBoostingRegressor
5. import xgboost as xgb

- Run and Evaluate selected models

First model is Linear regression to predict Sale price of car ,linear regression model based on to find best fit line works on linear equation Y=mX+C

Linear Regression Model

```
]:  1 from sklearn.linear_model import LinearRegression

]:  1 Lr=LinearRegression()
    2 Lr.fit(x_train,y_train)

:[38]: LinearRegression()

]:  1 y_pred=Lr.predict(x_test)

]:  1 #linear model training score
    2 Lr.score(x_train,y_train)

:[40]: 0.4054881136101919

]:  1 #mean absolute erro
    2 mean_absolute_error(y_pred,y_test)

:[41]: 101437.98825391031
```

Mean absolute error is high so we need to tune the model with Ridge and Lasso after tuning the model

```
  1 ridge_l1.score(x_test,y_test)

50]: 0.3897922031934573
```

Cross validation of model

```
  1 cross_val_score(ridge_l1,X_scaler,Y,cv=5).mean()

51]: 0.29883380238852225

  1 cross_val_score(lasso_reg,X_scaler,Y,cv=5).mean()

52]: 0.2993518593605284
```

Linear model score after tuning and cross validation of model

Next model is Decision Tree which use import from sklearn.tree library it

```
#model building with Decesion tree
from sklearn.tree import DecisionTreeRegressor
```

```
dt=DecisionTreeRegressor()
dt.fit(x_train,y_train)
```

```
DecisionTreeRegressor()
```

```
dt.score(x_train,y_train)
```

```
1.0
```

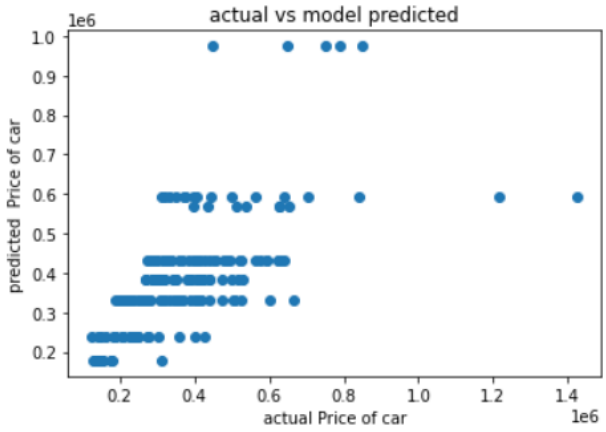
```
cross_val_score(dt,X_scaler,Y,cv=5).mean()
```

```
-1.187554995629677
```

DecisionTree regression Model

Cross val score is not good and when I plot scatter plot actual vs predicted model is not impressive

```
#Visualization of actual vs predicted price
plt.scatter(y_test,y_pred)
plt.xlabel('actual Price of car')
plt.ylabel('predicted  Price of car')
plt.title('actual vs model predicted')
plt.show()
```



Scatter plot of Decisiontree actual vs predicted model

DecesionTree score is 48% so I have to tune the model to achieve more accuracy of model with GridsearchCV

```
grid_search.best_params_
```

```
{'criterion': 'mse',
 'max_depth': 3,
 'min_samples_leaf': 1,
 'min_samples_split': 2,
 'min_weight_fraction_leaf': 0.02}
```

```
dt=DecisionTreeRegressor(criterion='mse',
                         max_depth=3,min_samples_leaf=1,min_samples_split=2,min_weight_fraction_leaf=0.02)
```

```
dt.fit(x_train,y_train)
```

```
DecisionTreeRegressor(max_depth=3, min_weight_fraction_leaf=0.02)
```

```
dt.score(x_train,y_train)
```

```
0.48335499391881853
```

```
y_pred=dt.predict(x_test)
```

```
print('**** accuracy post tuning****')
print(r2_score(y_test,y_pred))
```

```
**** accuracy post tuning****
0.3564318583966032
```

```
cross_val_score(dt,X_scaler,Y,cv=7).mean()
```

```
0.24660579510934985
```

DecesionTree model score and cross validation score

Similarly I have also build decision tree & GradientBoosting regression model

GradientBoost Regression model

```
from sklearn.ensemble import GradientBoostingRegressor
gbr=GradientBoostingRegressor()
```

```
gbr.fit(x_train,y_train)
```

```
70]: GradientBoostingRegressor()
```

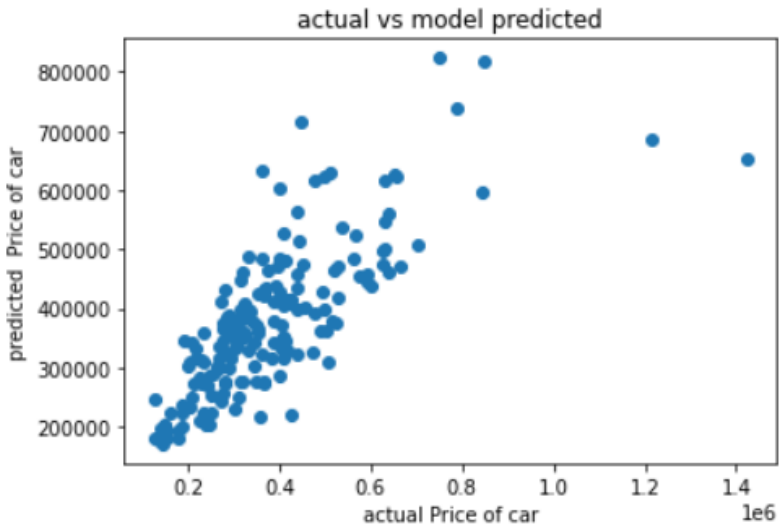
```
#check model score
gbr.score(x_train,y_train)
```

```
71]: 0.8701568709023249
```

GradientBoost regression model

Model performance is just ok, here is scatter plot of GradientBoost

```
#Visualization of actual vs predicted price
plt.scatter(y_test,y_pred)
plt.xlabel('actual Price of car')
plt.ylabel('predicted  Price of car')
plt.title('actual vs model predicted')
plt.show()
```



GradientBoost regression Model scatter plot actual vs predicted

```
from sklearn.ensemble import GradientBoostingRegressor
```

```
gbr= GradientBoostingRegressor()
gbr.fit(x_train,y_train)
```

```
GradientBoostingRegressor()
```

```
gbr.score(x_train,y_train)
```

```
0.9719801869518147
```

```
#cross validation of model
cross_val_score(gbr,X_scaler,Y,cv=5).mean()
```

```
0.8677098723510671
```

ıodel is slightly overfitted but good score so far

Gradientboost regression model score and cross validation score

Model score is 97% but cross val score tell that model have overfitting issue which we have to resolve by hyperparameter tuning

```
cross_val_score(gbr,X_scaler,Y,cv=5).mean()
```

```
0.3168074617254162
```

```
GridSearchCV(cv=5, estimator=GradientBoostingRegressor(),
             param_grid={'criterion': ['friedman_mse'],
                         'learning_rate': [0.1, 0.2, 0.3, 0.4], 'loss': ['ls'],
                         'n_estimators': [100, 150, 200, 250],
                         'subsample': [1.0, 2, 3, 4, 7]})
```

```
1 grid_search.best_params_
```

```
{'criterion': 'friedman_mse',
 'learning_rate': 0.2,
 'loss': 'ls',
 'n_estimators': 200,
 'subsample': 1.0}
```

```
1 gbr=GradientBoostingRegressor(criterion='friedman_mse',learning_rate=0.2, loss='ls',n_estimators=200,subsample=1)
```

```
1 gbr.fit(x_train,y_train)
```

```
GradientBoostingRegressor(learning_rate=0.2, n_estimators=200, subsample=1)
```

```
1 gbr.score(x_train,y_train)
```

```
0.9815908776391457
```

re improved after tuning the model

```
1 y_pred=gbr.predict(x_train)
```

```
1 cross_val_score(gbr,X_scaler,Y,cv=5).mean()
```

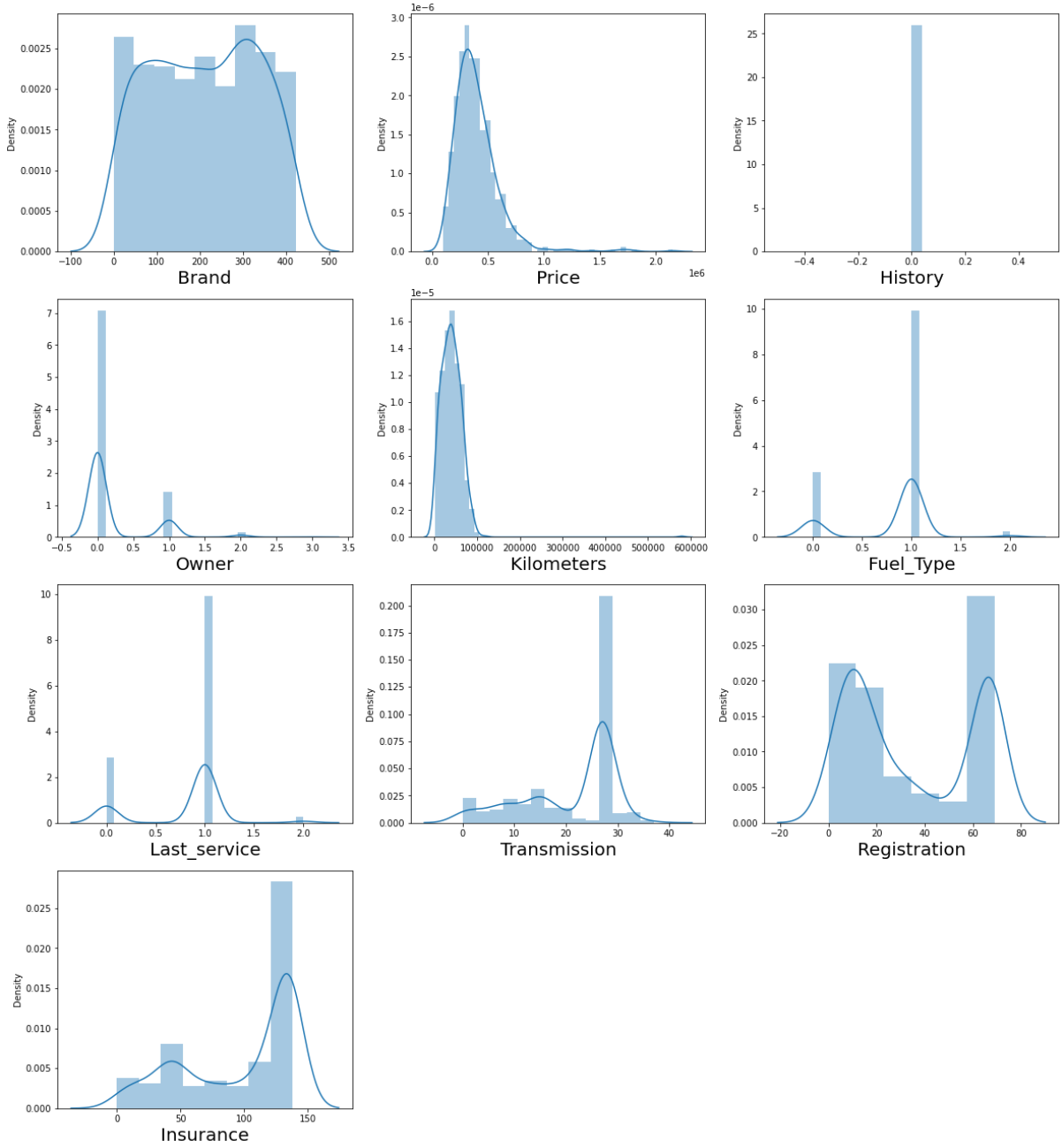
```
0.28645446841003375
```

Model score is improve i.e. 98% now ,but cross validation is still low score which

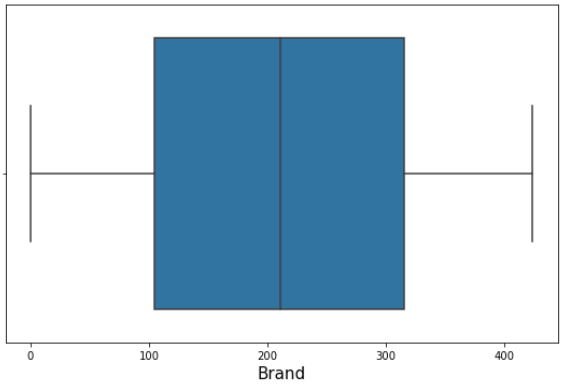
- **Key Metrics for success in solving problem under consideration**

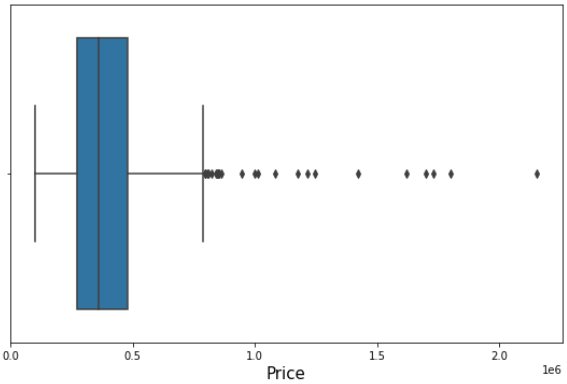
  R2 score ,mean squared error score , give idea about which model score performance is accurate ,cross validation score of each model test overfitting issue ,hyperparametr tuning boost model score ,scatter plot between actual vs predicted model.

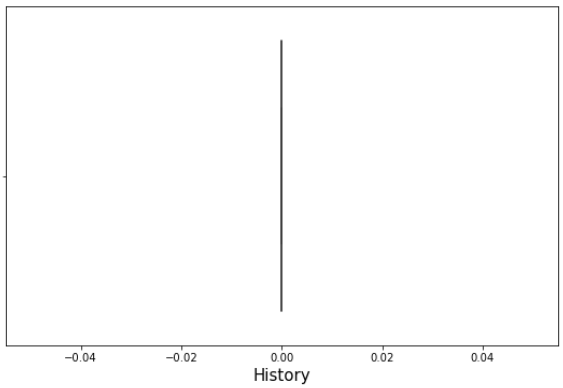
- **Visualizations**

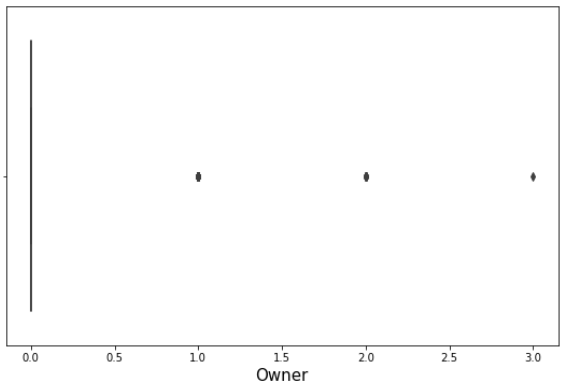


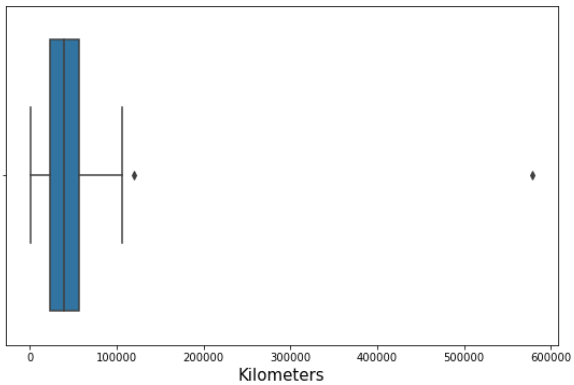
Distplot to visualization of data distribution

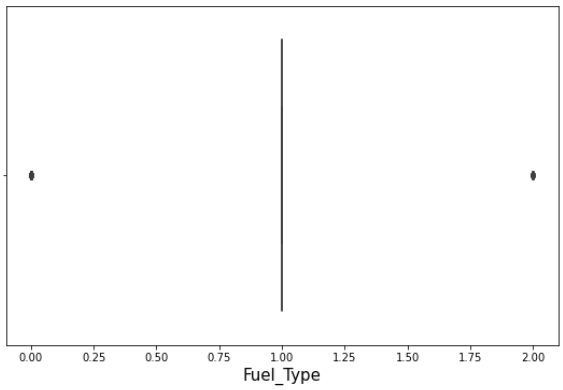

0
100
200
300
400
Brand

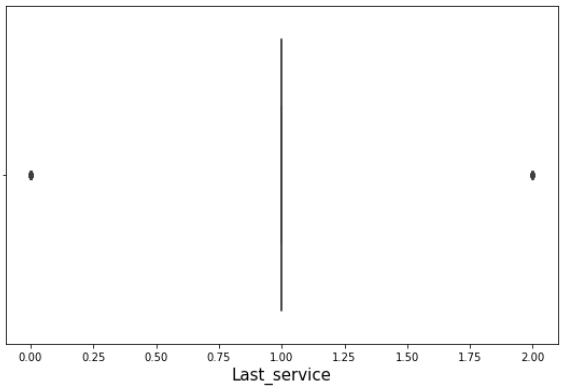

0.0
0.5
1.0
1.5
2.0
1e6
Price

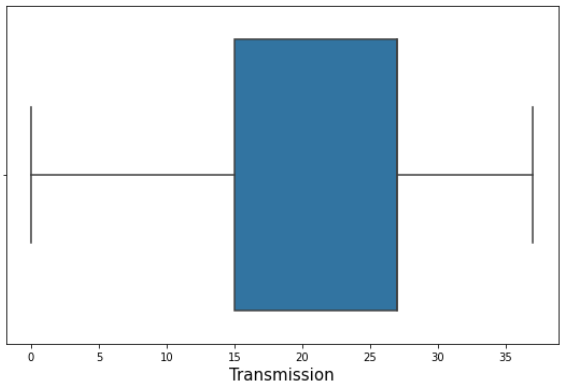

−0.04
−0.02
0.00
0.02
0.04
History

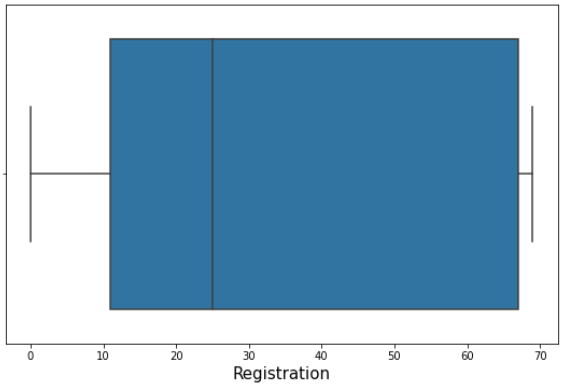

0.0
0.5
1.0
1.5
2.0
2.5
3.0
Owner

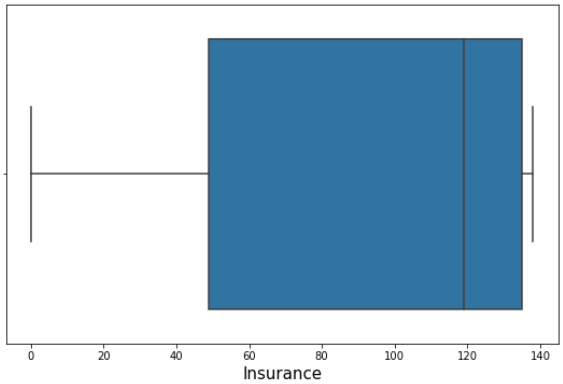

0
100000
200000
300000
400000
500000
600000
Kilometers


0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
Fuel_Type

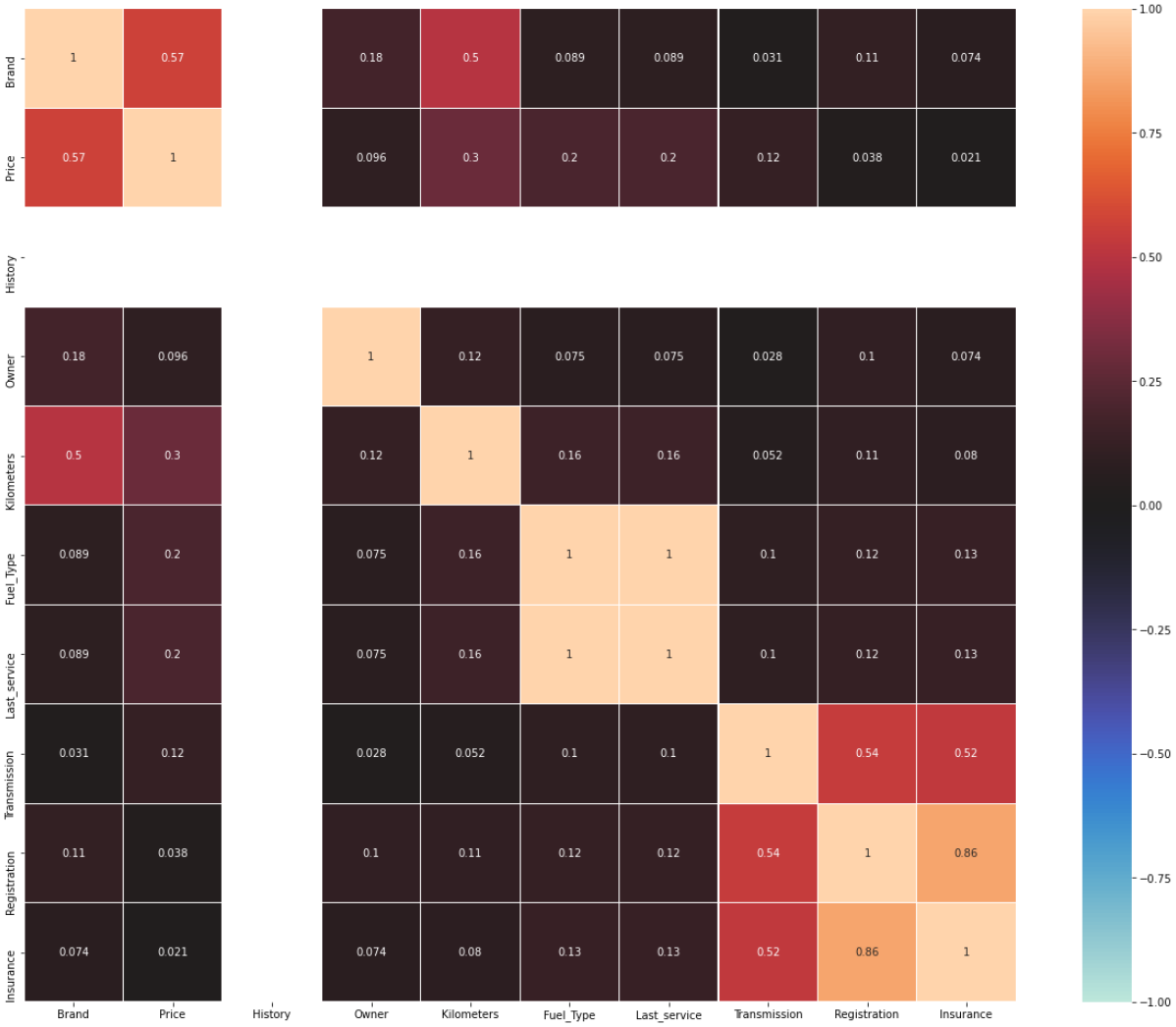

0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
Last_service


0
5
10
15
20
25
30
35
Transmission

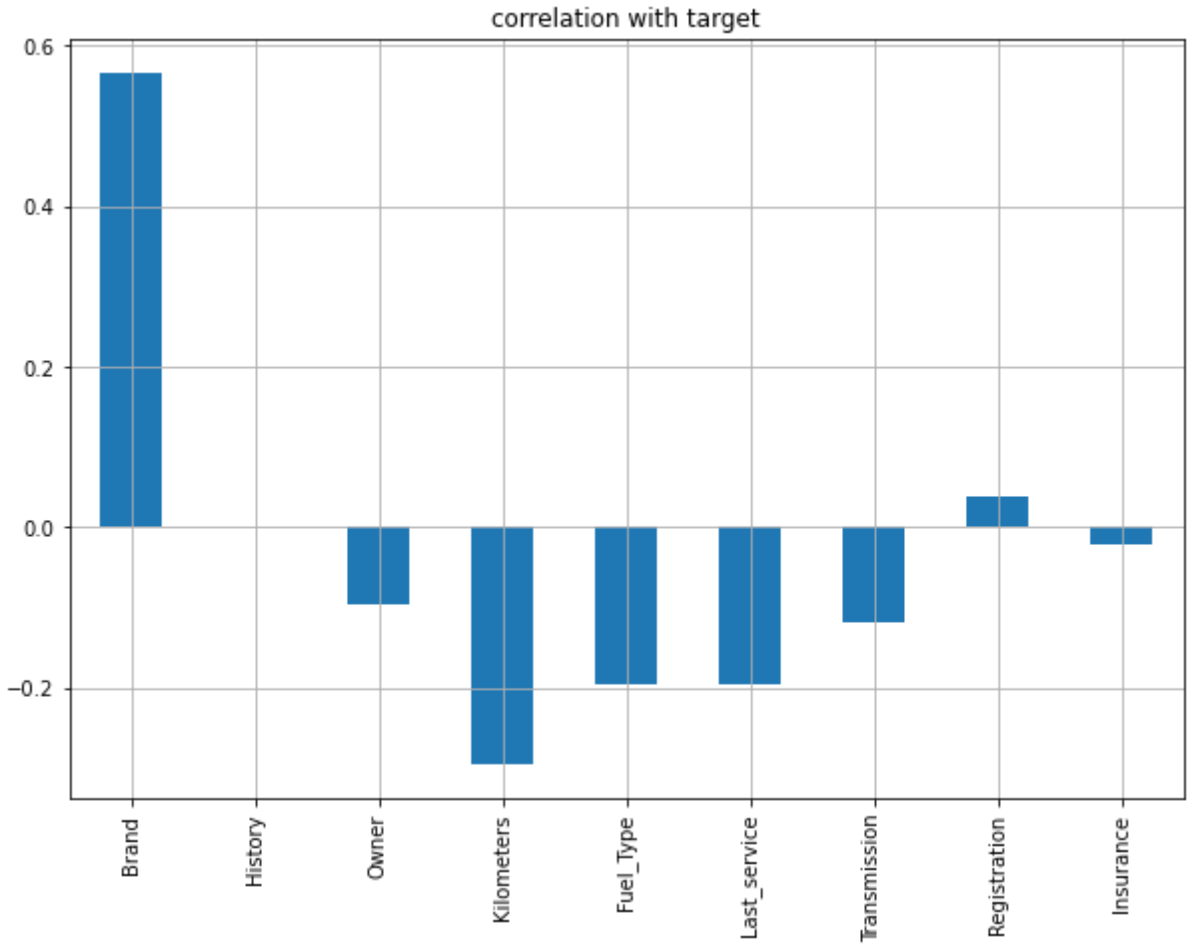

0
10
20
30
40
50
60
70
Registration


0
20
40
60
80
100
120
140
Insurance

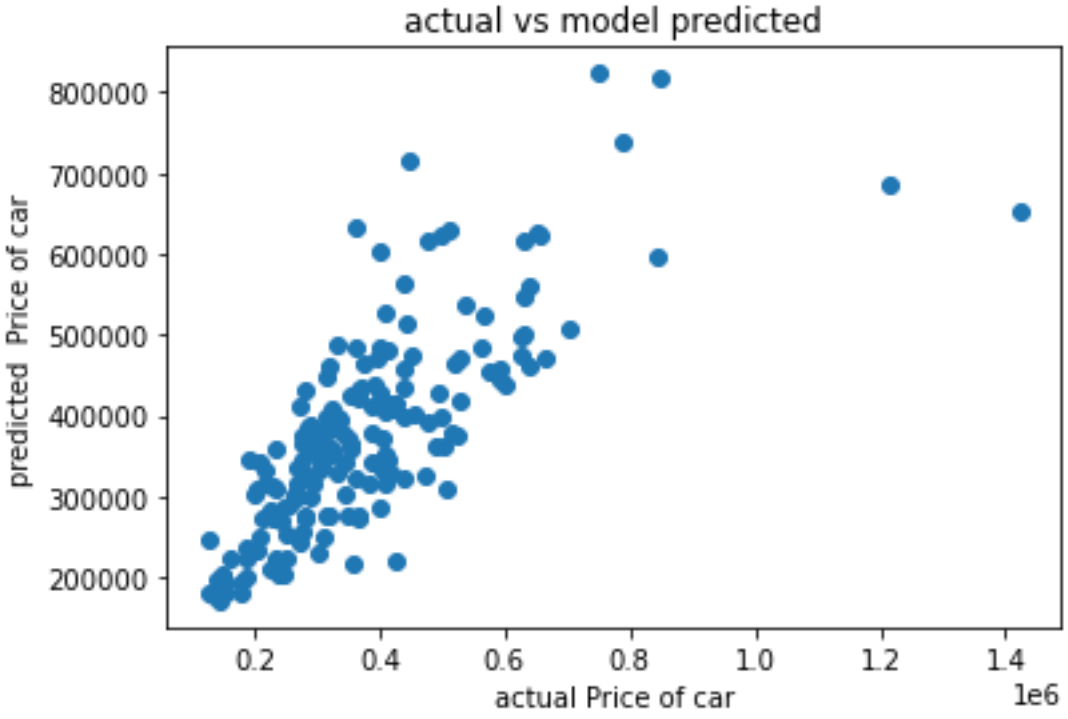
Boxplot to detect outliers in dataset



Heatmap to check multicollinearity present in dataset



Visualization the correlation with class



Scatter plot check actual price vs predicted price

- **Interpretation of the Results**

  Data is taken from car 24 website with selenium scrapping technique, one features that is year of manufacturing data is missed to fetch, which effects model building, due to time constraint. Gradient boost, Random forest, XGB model works quite okay, for used car price prediction. R2 score and cross val score is low model score is high means my model having overfitting issue even though hyperparameter tuning done but improvement in result is not up to mark.

CONCLUSION

- **Key Findings and Conclusions of the Study**

  Car sale price data is right skewed as per distplot
  In order to have a clear view of how the key variables relate to SalePrice, Brand and Kilometres Fuel Type, Last Service shows good relationship with class i.e. Price identified with scatter plot
  I have used Correlation Heatmap to plot correlation coefficients, and outcomes is
  Brand have strong positive relation ,Registration show weak positive relation other features shows negative relationship with target variable ,but history have no relation with target variable

- **Learning Outcomes of the Study in respect of Data Science**

  First model Linear reg what I attempted ad it's give me poor result laso disappointed with R2 score after tuning model score not improved and cross validation score is low ,moving forward when I build GradientBoost regression model score ,cross validation and scatter plot R2 score all reflect model performing better compare to decision tree .Then Random Forest from ensemble techniques also perform quite well like Gradient Boost. Price prediction of used car model accuracy R2 score ,squared error and other matrices can

perform better ,its effected because of year of manufacturing data not fetched with selenium web scrapping,

- ## Limitations of this work and Scope for Future Work

  You may build model with different approach more EDA techniques to identify insights, due to time constraint you may extend different approach to build more model and find best accuracy and R2 score.

  Used Car sale price project data taken from car 24 website ,you may fetch data other website like car dekho ,spinney ,etc and concatenate in this data set.

  This will definitely help to give idea about used car price prediction model building from scratch. I will extend this work in near term to improvise my model and add more features ,bring more insights.