USED CAR PRICE PREDICTION

A Step-by-Step guide to Build a Machine Learning Model to predict Car Prices

Submitted by

Razni Nazeem

Internship 26

Datatrained Education

AGENDA

- Overview.
- Problem Statement.
- Problem Understanding.
- Housing Price Prediction and importance
- Exploratory data analysis.
- Visualizations.
- Analysis.
- Data cleaning steps.
- Model Building.
- Hyper Parameter Tunning.
- Saving the model and predictions from saved best model.
- Conclusion.

OVERVIEW

Data Collection Phase

In this Phase, I collected the data from the website Cars24.com using Web scraping by Selenium.

Model Building Phase

1. EXPLORATORY DATA ANALYSIS

Analyzing the dataset using Exploratary Data Analysis and Visualization

2. BUILDING THE MODEL

Building the model taking the highest R2 score and CV Score and Minimum errors.

3. PREDICTION

Building the model taking the highest R2 score and CV Score andSaving the best model and predicted the values of the cars.

PROBLEM STATEMENT

In this Article, I will be guiding you to the step-by-step procedure in building a Machine Learning model in Python using popular machine learning libraries NumPy, Pandas & scikit- learn to predict used Car prices in India.

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.

www.cars24.com search

PROBLEM UNDERSTANDING

As per Times of India, Small-town India is cashing in on its luxury cars taking advantage of the soaring demand and a resultant rise in prices of pre-used luxe vehicles. In the first six months of Calendar 2022, Tier 2 luxury car listings were up 45% compared to 40% for Tier 1 listings on the OLX Platform. This project was scraped on 24th July,2022 from Cars24.com website for Used cars. I have taken 11 major cities in India including New Delhi, Chennai, Bangalore and the dataset includes 21 explanatory features and 9176 entries of Used cars. The aim is to predict the used car Prices for our client with respect to their budgets and priorities. By analyzing previous market trends and price ranges, and upcoming developments future prices will be predicted.

DATA SCIENCE

Data science comes as a very important tool to solve problems and to help the companies increase their overall revenue, improving their marketing strategies and focusing on changing trends in car prices. Used Car Price prediction is important to drive Economic efficiency. Traders can investigate the market for which Brands and Models the price goes up and for which its coming down.

IMPORTANCE

As Internet risen substantially, Websites and Applications for used Cars have grown tremendously. It is very Important to build a model for used Car price trend so that small traders can get a hold of the current market situations. Therefore, the used Car Price prediction model is very essential in filling the information gap and improving Economic efficiency.

EXPLORATORY DATA ANALYSIS

1

Imported Libraries and Loaded the dataset. Aldo did all the statistical Analysis of the dataset like shape, unique, value_counts, etc.

2

Dropped columns with 60% and more null values as it might cause bias and variance while model building.

3

Replaced NaN values with mean, median and mode using Imputation Technique.

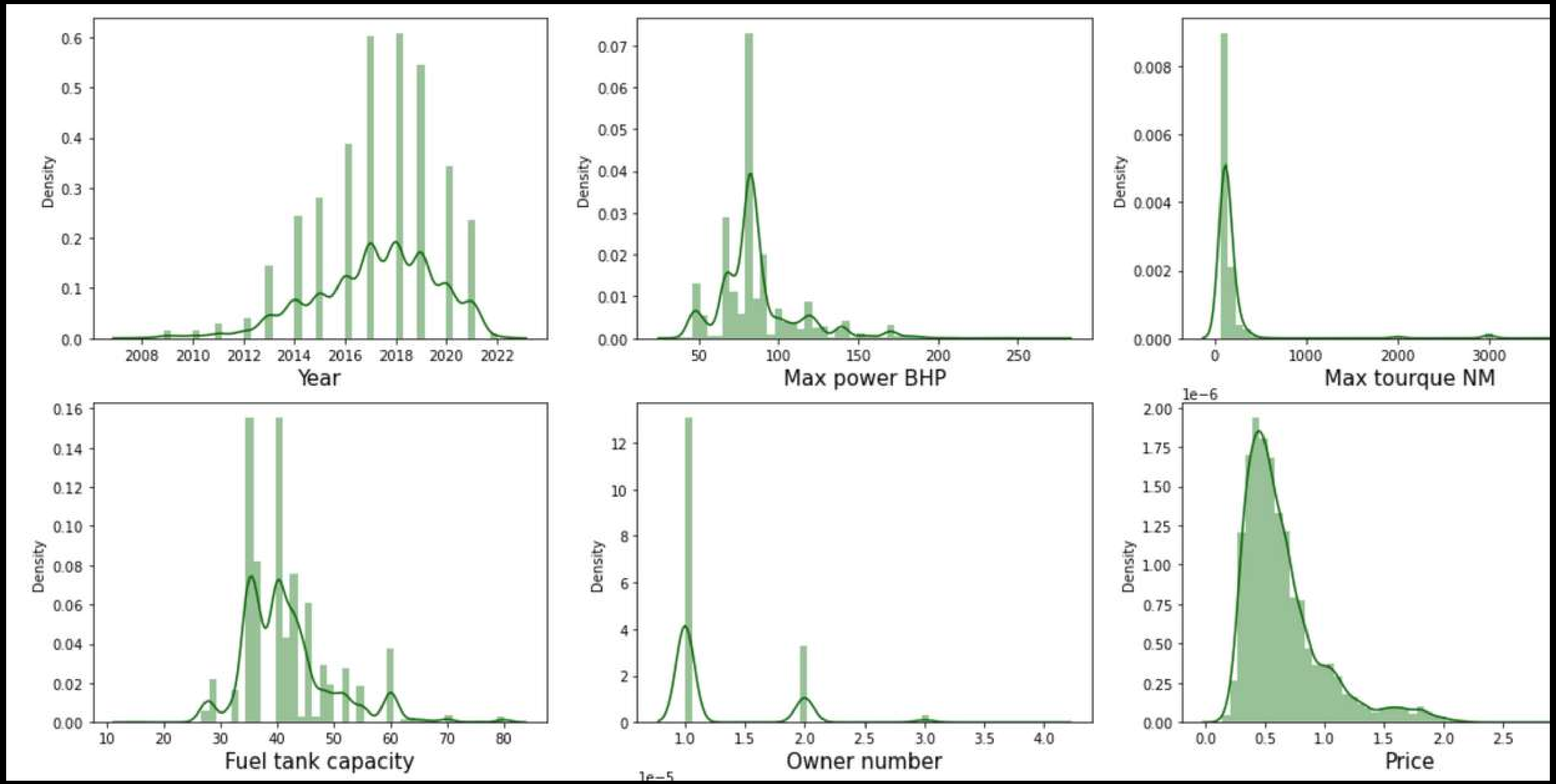
4

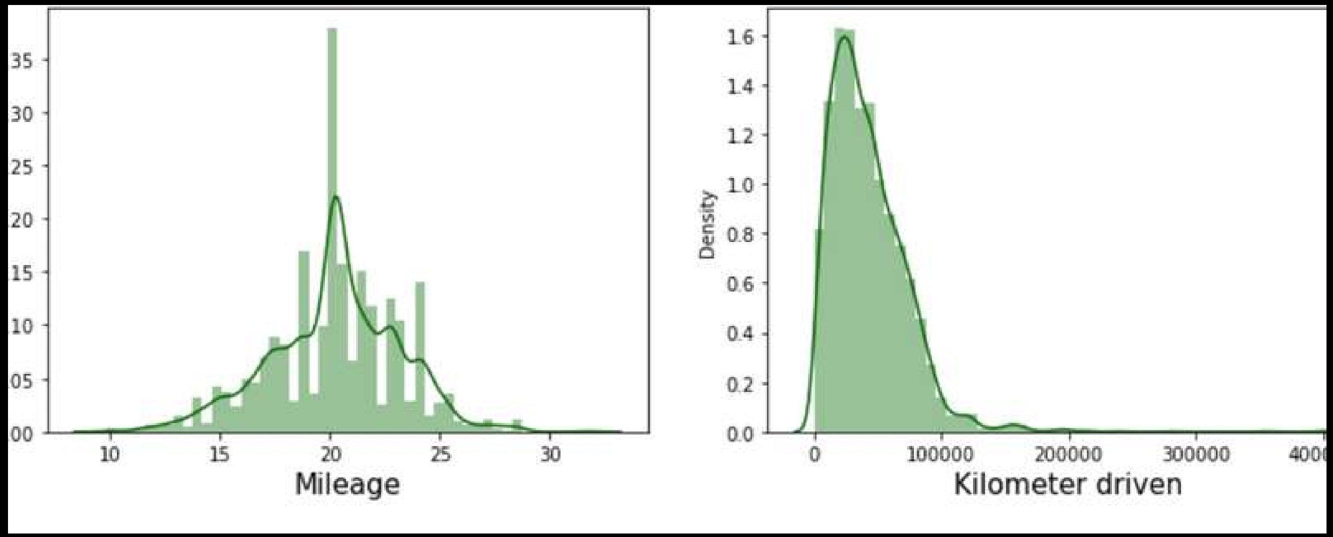
Dropped columns which have all unique values.

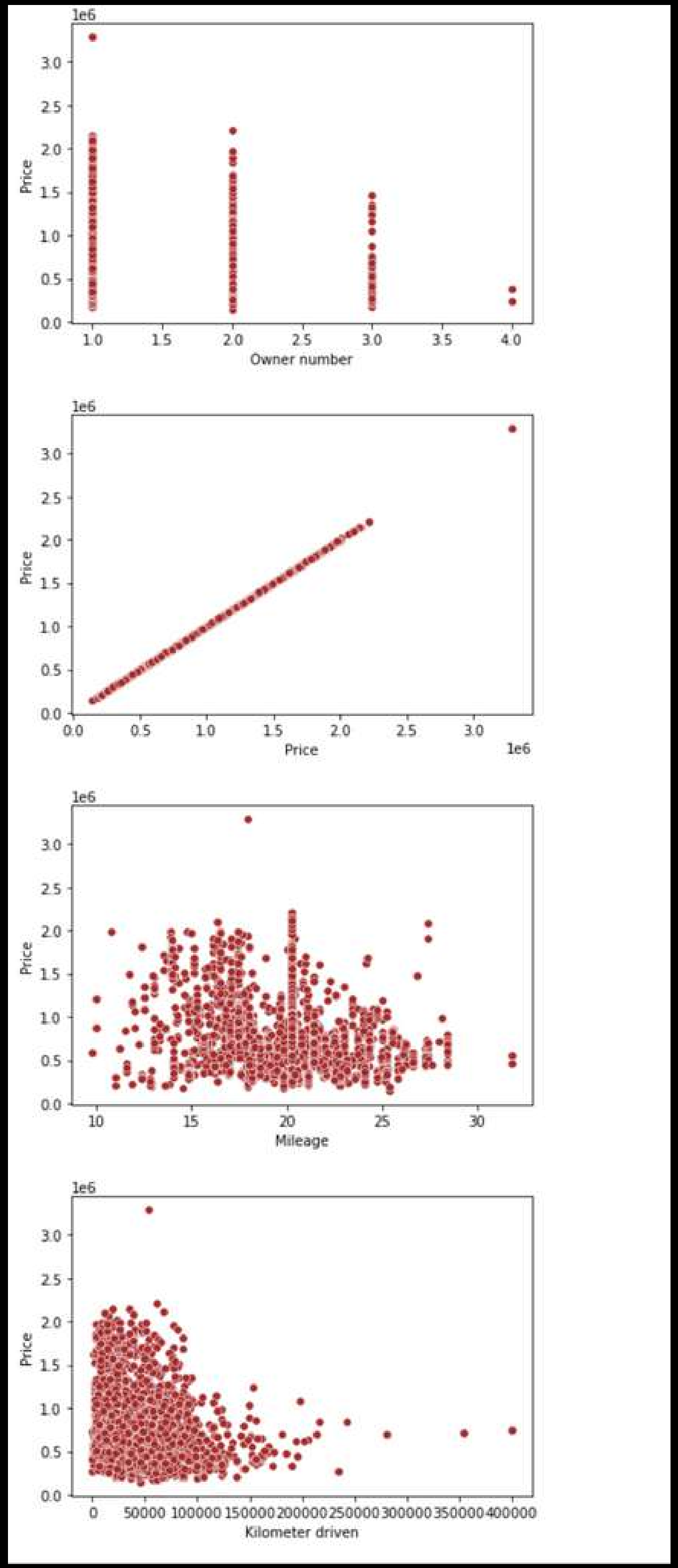
5

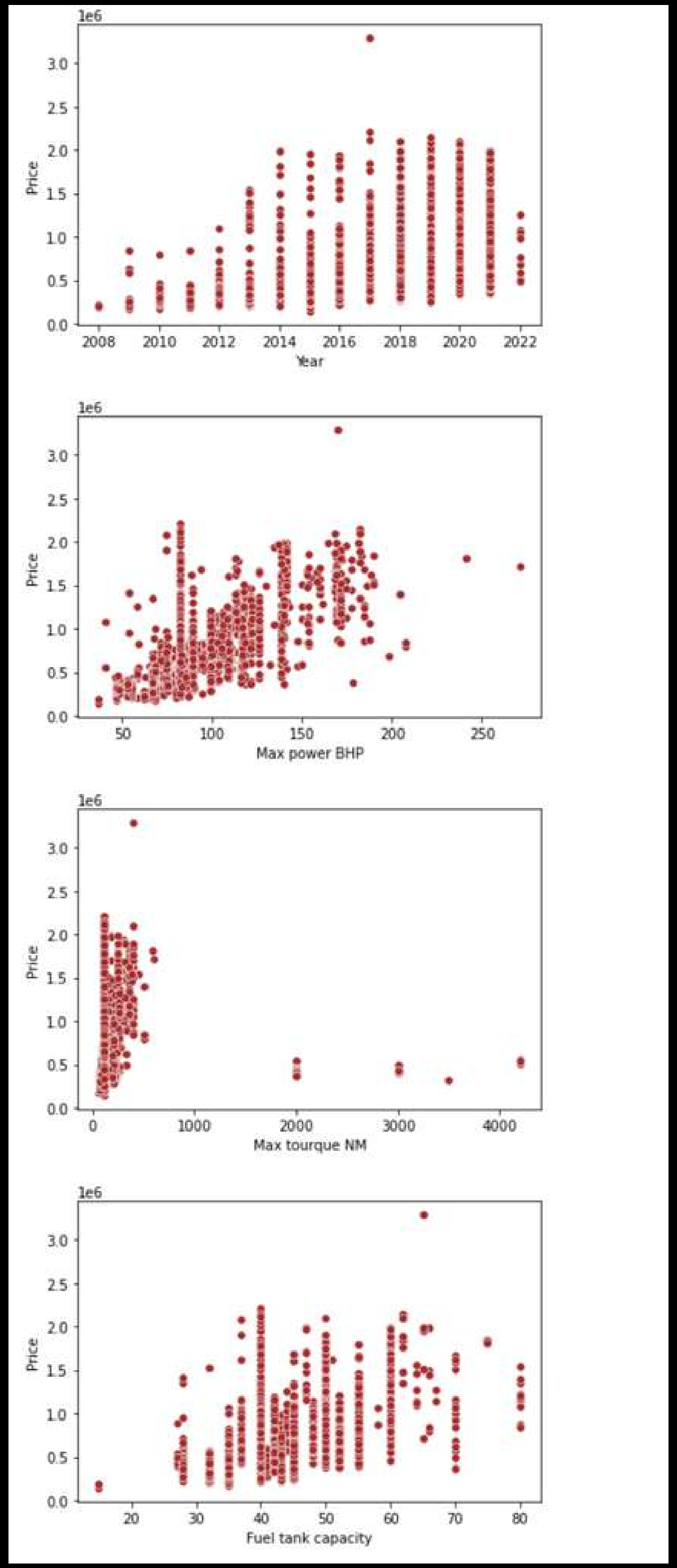
Data visualization tools like scatterplot, count-lot, boxenplot, etc have been used

Visualization of Numerical columns





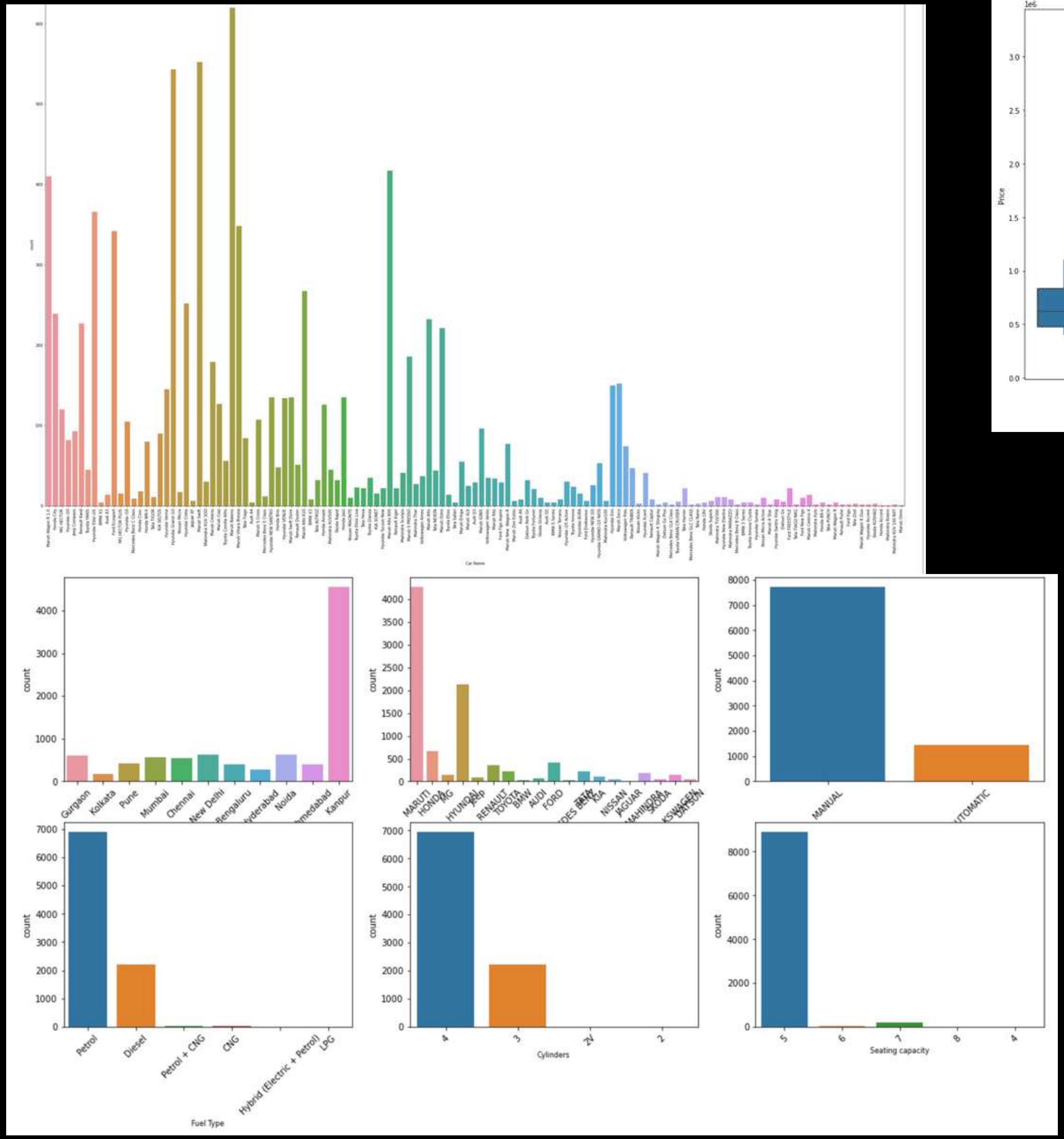


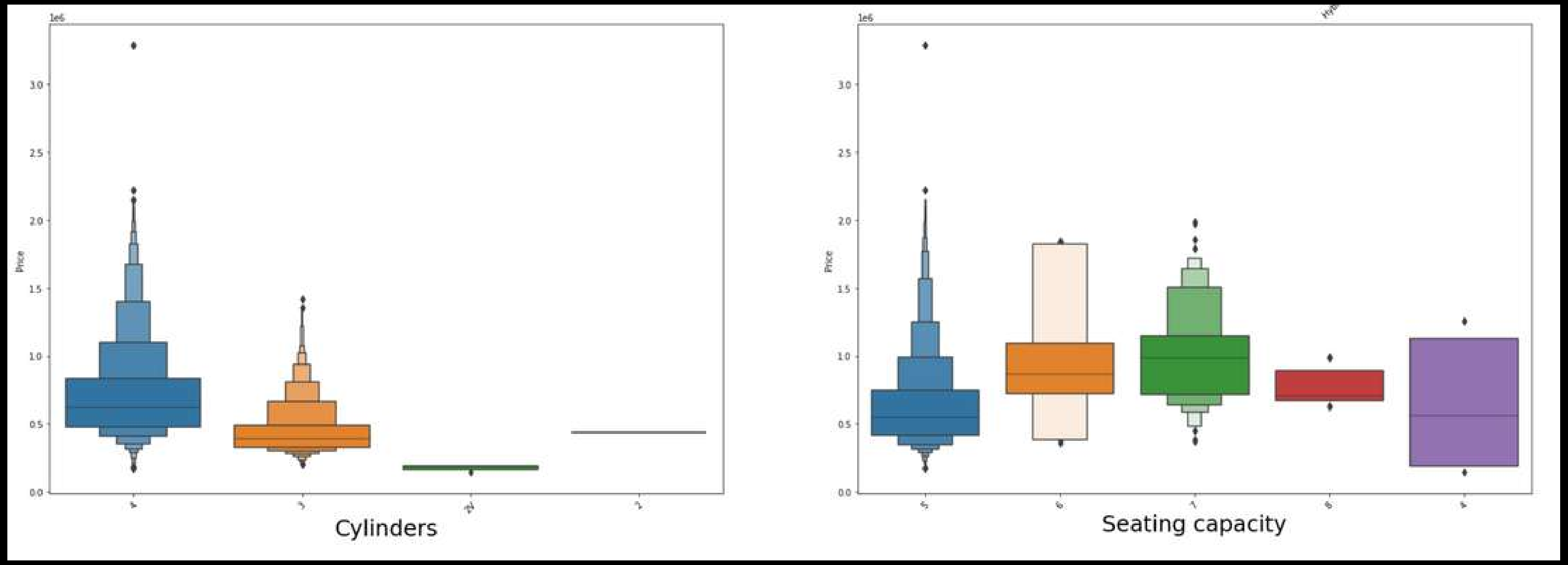


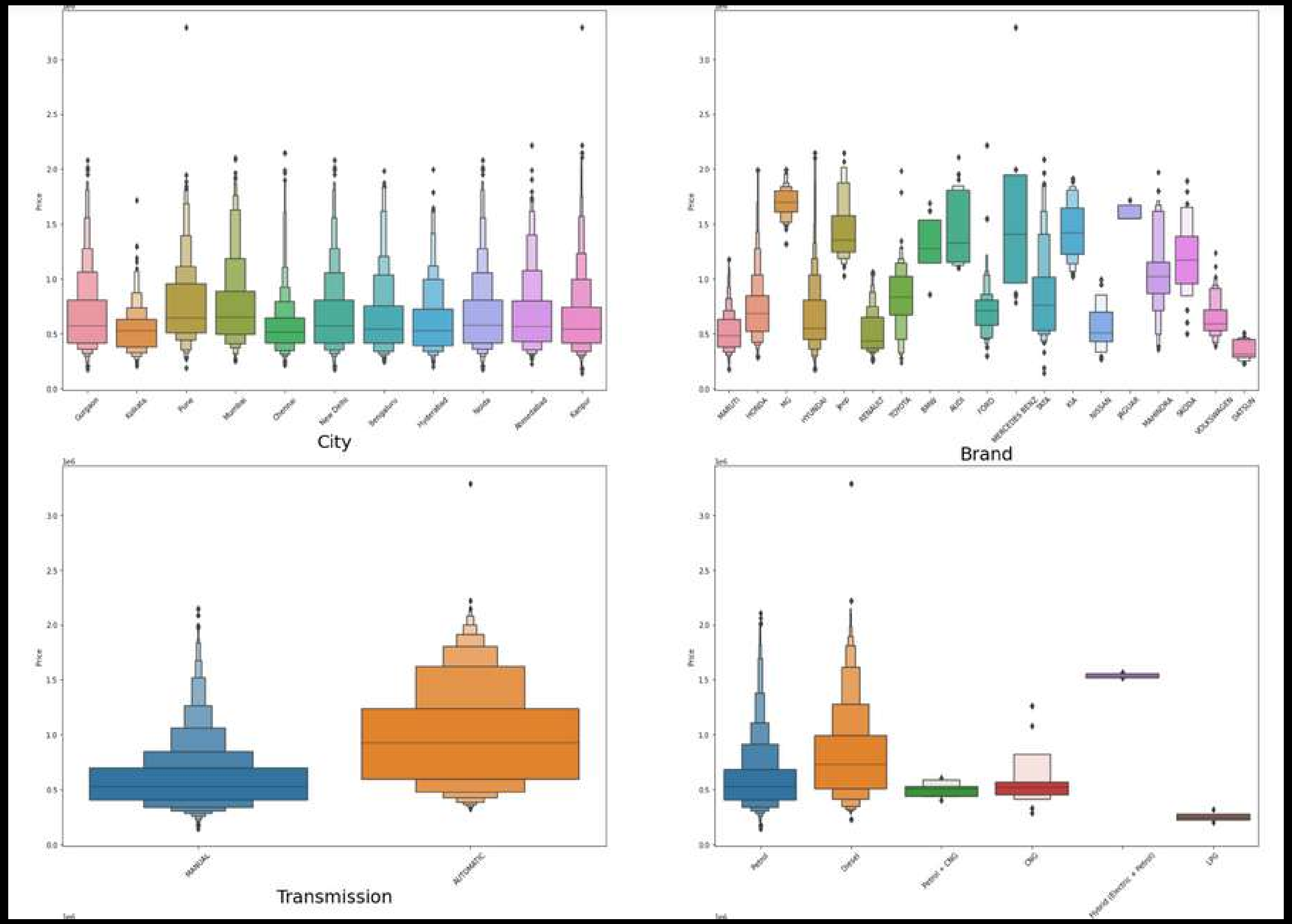
OBSERVATIONS

1. Max power BHP, Max torque NM, Price, Kilometer driven, Owner numbers are right skewed.
2. Year is left skewed.
3. Fuel tank capacity and Mileage are somewhat normally distributed
4. We can see some of the columns have direct relations and some do not have any relation with Target Price.
5. Year - As year increases, Price also increases.
6. Max power BHP - As max power BHP increases, Price also increases.
7. Max torque NM - It shows least the torque, most of the price lies in there. No specific relation.
8. Fuel Tank capacity - Most of the Price lies in between 35 to 60 litres capacity.
9. Owner Number - Price decreases as owner number increases.
10. Mileage - Mileage and Price do not have specific relationship.
11. Kilometers driven - Price is more when kilometers driven is less.

Visualization of Categorical Columns





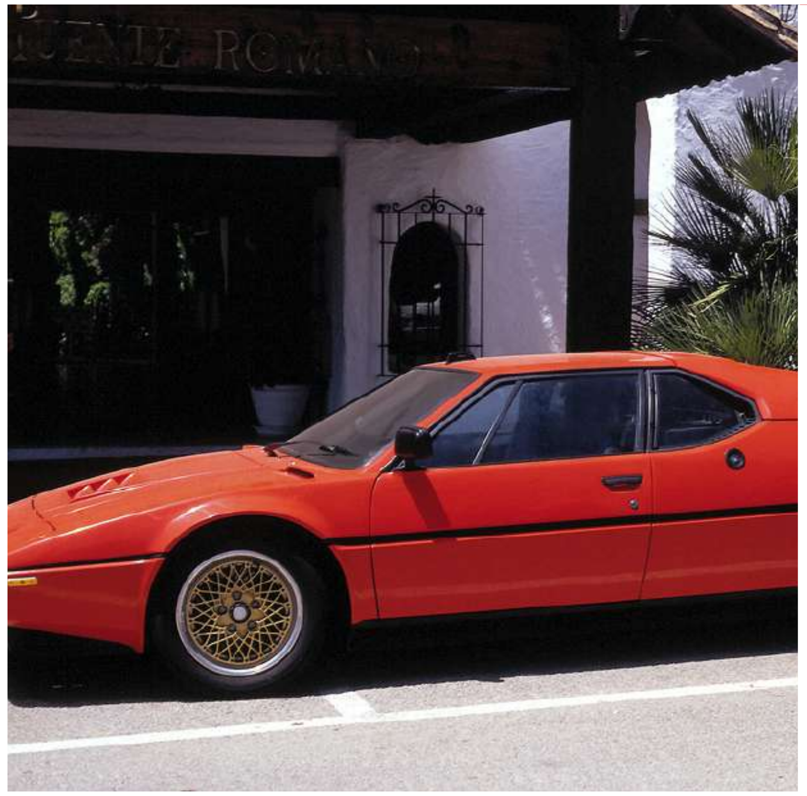


OBSERVATIONS

1. Most of the data of used cars are from City Kanpur and least from Kolkata.
2. Maruti is the most popular Brand in the dataset followed by hyundai. Least from Jaguar.
3. Most of the cars are Manual Transmission.
4. Most of the cars are Petrol as fuel type and i also have a very few number of Hybrid cars as well.
5. Most of the cars are 4 cylinders and a very few 2 cylinders inline and V shape.
6. Most cars have seating Capacity of 5 Seater.
7. Maruti baleno is the most popular in my dataset followed by Maruti Swift and
8. Hyundai grand i10.
9. I have Maruti Alto800 and Maruti wagonR1.0 also on the top 5 list.
10. We have only 1 data of Honda Accord, Maruti Omni and Mahindra Bolero.
11. As mentioned earlier, Maruti is the top Brand followed by Hyundai in my dataset.
12. City - City do not have specific relation with Price.
13. Brand - Mercedes is the most expensive Car in my dataset and Datsun Brand cars are mostly cheap.
14. There are many outliers in the Brands column as the prices are sometimes very high for few of the models.
15. Transmission - As we can see that Automatic cars are very expensive than Manual ones.
16. Fuel type - Hybrid and Diesel cars are expensive than others.
17. Cylinders - 4 cylinder cars are very expensive and 2 and 2V cylinders are cheap.
18. Seating capacity - 6 and 7 seaters are most expensive than 5 and 4 seater.

ANALYSIS

- I have used bar plots to visualize the count of Categorical.
- I have used distplot to analyze distribution of numerical.
- I have used a boxen plot to find the relation between categorical columns and target.
- I have used swarmplot, strip plot and scatter plot for visualization of numerical columns with target.



DATA CLEANING STEPS

1. In my datasets I found null values, outliers and skewness and removed them.
2. I have used imputation method to replace null values. To remove outliers I have used Zscore method. And to remove skewness I have used Yeo-Johnson method.
3. To encode the categorical columns I have use Ordinal Encoding.
4. I have used Pearson's correlation coefficient to check the correlation between dependent and independent features.
5. Also I have used standardization and also checked Multicollinearity and dropped columns.
6. Next step was model building with all regression algorithms.

MODEL BUILDING

Since our Target is Price, which is continuous, I have a Regression Problem. I have used 7 different algorithms to build the models and found the R2 score and CV Score of each one of them. I have finally decided to select the model which has the highest r2 and CV Score and least MSE, RMSE and MAE and that model is Random Forest Regressor model.

1. Linear Regression
2. Ridge Regressor
3. Random Forest Regressor
4. KNN Regressor
5. XGB Regressor
6. SGD Regressor
7. Gradient Boosting Regressor

LINEAR REGRESSION

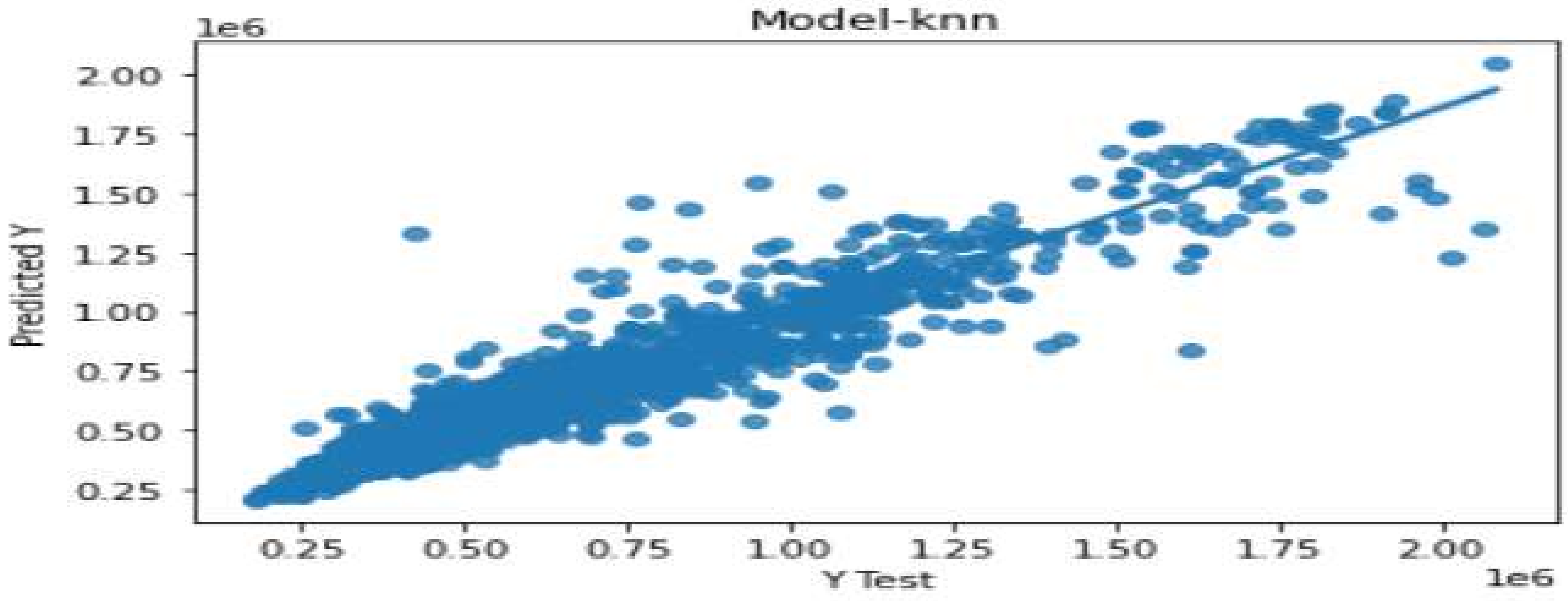
At first, I found the best Random state for which I got the best score and performed a train- test-split to fit the model. My score for Linear regression model is 73.33% and CV Score of 69.68%. I have tuned with the best parameters, but score remained the same.

```
pred = Rd.predict(x_test)
print('The r2 score is:', r2_score(y_test, pred))
print('The mean absolute error', mean_absolute_error(y_test, pred))
print('The mean squared error', mean_squared_error(y_test, pred))
cv = cross_val_score(Rd, x,y,cv=5)
print('The cross validation score', cv.mean())
```

```
The r2 score is: 0.7333770327426984
The mean absolute error 107775.91304121495
The mean squared error 24377502674.3793
```

KNN REGRESSOR

I found the best random state which yields the best score and then fit the model. R2 score for KNN Regressor was 91.09%. I have tuned with different parameters and the score has improved to 95.45% and CV Score of 94.64%. The regplot of actual and predicted values using KNN is:



RANDOM FOREST REGRESSOR

I first found the best random state and fit the model. I got a score of 98.94% and a CV Score of 97.04%. Hence, I selected the random forest as the best model and saved the model. While performing the fitting of the final model, my score was improved, and it became 98.98%. Hyper parameter tuning of the model did not increase my score.

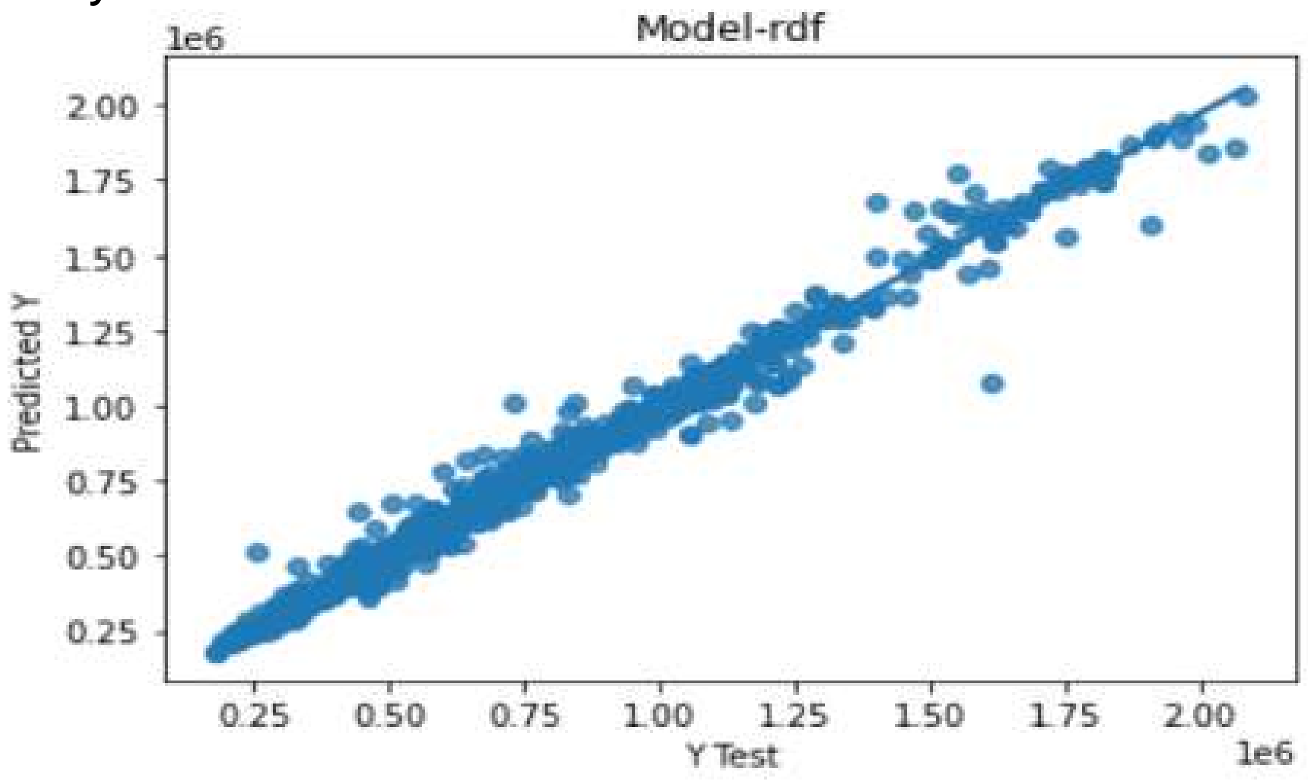
```
pred_rdf = rdf.predict(x_test)

print('The r2 score is:', r2_score(y_test, pred_rdf))
print('The mean absolute error', mean_absolute_error(y_test, pred_rdf))
print('The mean squared error', mean_squared_error(y_test, pred_rdf))
print('root_mean_squared_error:',np.sqrt(mean_squared_error(y_test,pred_rdf)))
```

```
The r2 score is: 0.9894158276247104
The mean absolute error 15711.477481567714
The mean squared error 1004202483.1612341
```

```
cv = cross_val_score(rdf, x,y,cv=5)
print('The cross validation score', cv.mean())
```

```
The cross validation score 0.9704991901546685
```

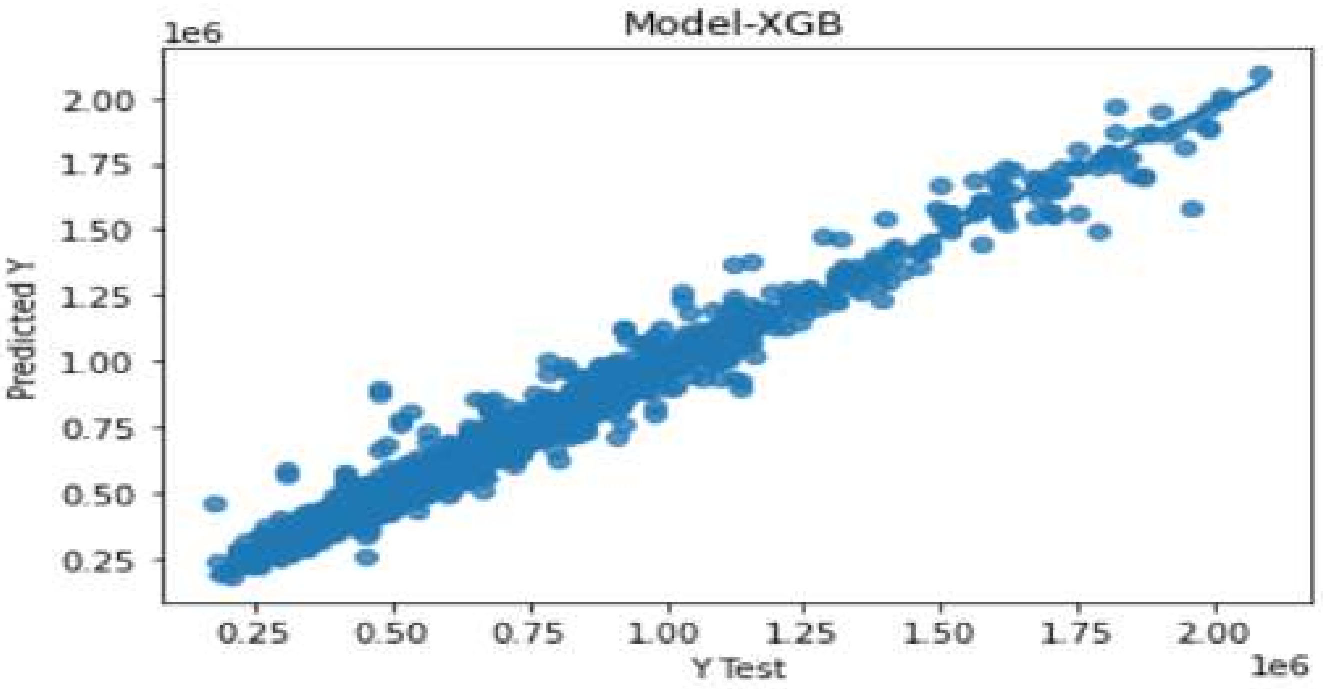


XGB REGRESSOR

For XGB Model, the r2 score was initially 97.76% NS after Hyper parameter tuning score Improved to 98.04%. The CV Score for the XGB hyper parameter tuned model was 97.52%. The regplot of XGB model is also very good and linear.

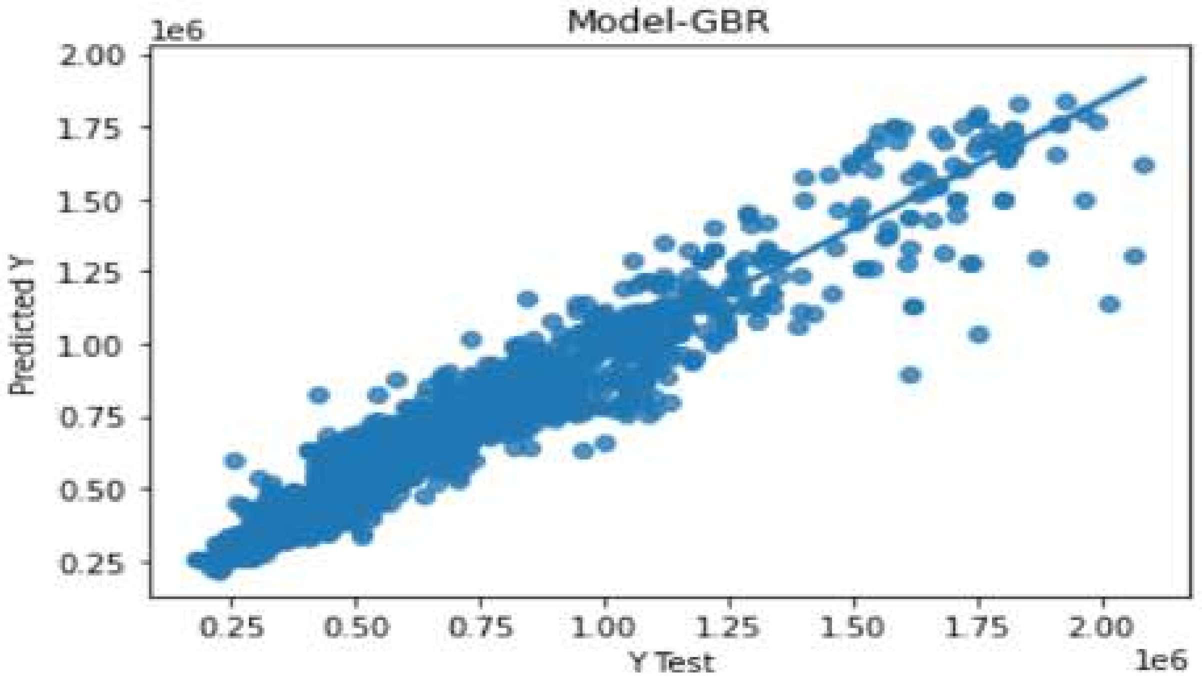
```
XGB_H=XGBRegressor(learning_rate=0.1,max_depth=7,n_estimators=300,
                   reg_alpha=0.5,reg_lambda=1,gamma=0.05)
XGB_H.fit(x_train,y_train)
xgbpred=XGB_H.predict(x_test)
print('The r2 score is:', r2_score(y_test,xgbpred))
print('The mean squared error', mean_squared_error(y_test,xgbpred))
print('The mean absolute error', mean_absolute_error(y_test,xgbpred))
print('root_mean_squared_error:',np.sqrt(mean_squared_error(y_test,xgbpred
```

```
The r2 score is: 0.9804031840462478
The mean squared error 1938634330.2744417
```



GRADIENT BOOSTING REGRESSOR

The R2 score of Gradient Boosting regressor model was 91.82%. But after Hyper parameter tuning, R2 score improved to 95.32% and CV Score of 93.53% Regplot of this model is also lineaR.



RIDGE REGRESSOR

The R2 score of Ridge regressor was 73.37% and CV Score of 69.68%. Same as Linear regression.

SGD REGRESSOR

The R2 score for SGD Model is 73.10% and CV Score of 69.61% similar to Linear regression and Ridge.

HYPER PARAMETER TUNING

Hyper parameter tuning of the Final model did not increase my score.

```
The best parameters across ALL searched params:
{'n_estimators': 200, 'max_features': 'sqrt', 'max_depth': 30, 'criterion': 'squared_error', 'bootstrap': True}
```

```
rdf_H = RandomForestRegressor(max_depth=30,max_features='sqrt',bootstrap= True,
                            criterion='squared_error',n_estimators=200)
rdf_H.fit(x_train,y_train)
predrdf = rdf_H.predict(x_test)
print('The r2 score is:', r2_score(y_test, predrdf))
print('The mean absolute error', mean_absolute_error(y_test, predrdf))
print('The mean squared error', mean_squared_error(y_test, predrdf))
print('root_mean_squared_error:',np.sqrt(mean_squared_error(y_test,predrdf)))
```

```
The r2 score is: 0.9753328028795336
The mean absolute error 29756.904531915796
The mean squared error 2340368214.224409
```

SAVING THE MODEL

I have saved my best model using .pkl as follows.
After saving the best model, loading my saved model and predicting the test values.

```
# Saving the model using .pkl

import joblib
joblib.dump(Final_model,"Used_Car_Price.pkl")
```

```
['Used_Car_Price.pkl']
```

CONCLUSIONS

In this project report, I have used machine learning algorithms to predict the Used car prices. I have mentioned the step-by-step procedure to analyze the dataset and find the correlation between the features. Hence, we calculated the performance of each model using different performance metrics and compared them based on these metrics. Then we have also saved the data frame of predicted prices and Actual Price.

I have observed that certain features like Max power BHP, Year of the Vehicle, etc. contribute the most to the Price of the Car. Also, conditions like Kilometers driven negatively affect the price. As years passed the value decreased.

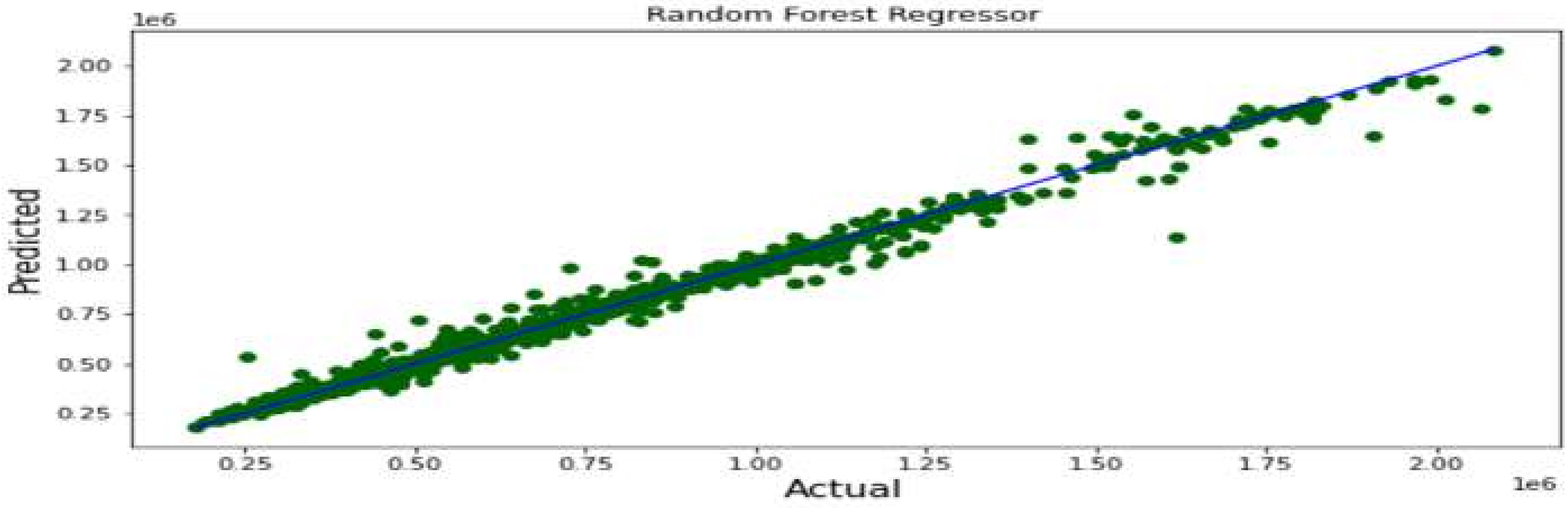
Improvement in computing technology has made it possible to examine social information that cannot previously be captured, processed, and analyzed. The power of visualization has helped us in understanding the data by graphical representation. Data cleaning is one of the most important steps to remove missing values and to replace them with respective mean, median or mode. This study is an exploratory attempt to use seven machine learning algorithms in estimating Car prices, and then compare their results.

The data was collected from a website, and I found that many of the ads are posted in different cities. This causes duplication of data. The MSE, MAE and RMSE is very high for the dataset. There was a lot of skewness present in the dataset which will again affect the model as we must transform it.

Even after all these Limitations and drawbacks, my model tends to perform well with an accuracy of 98.98% with Random Forest model and a CV Score of 97.05%.

Actual Vs Predicted Plot ¶

```
plt.figure(figsize=(10,5))
plt.scatter(y_test, prediction, c='darkgreen')
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("Random Forest Regressor")
plt.show()
```



THANK YOU

We want to provide the best models to our costumers that are high in quality, trusted, and have up the maximum Accuracy