Project report on

CAR PRICE PREDICTION

Submitted By

Mr. Abhinandan Kadam

ACKNOWLEDGMENT

It is my sensual gratification to present this report on CAR PRICE PREDICTION project. Working on this project was an incredible experience that has given me a very informative knowledge.

I would like to express my sincere thanks to MR. SAJID CHOUDHARY for a regular follow up and valuable suggestions provided throughout.

And I am also thankful to FlipRobo Technologies Bangalore for their guidance and constant supervision as well as for providing necessary information regarding the project and also for their support in completing the project

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.

I have scraped the data from the well known e-commerce website cardekho.com; where I find some more features of cars to fetch than other sites. As per the requirement of our client we need to build the model to predict the prices of these used cars. In this project I will build various machine learning models sand will check the performance of each and every model. Based on our evaluations finally we will select the best machine learning model.

INTRODUCTION

Business Problem Framing

The Indian Automotive sector is struggling during the COVID-19 pandemic but however, that's not the case with the used car industry. COVID-19 has hit the used car segment too, but the impact is estimated to have been much less severe than the business of selling new cars.

The increased preference for personal mobility due to safety concerns amid the Covid-19 pandemic has led to supply constraints in the used car market, as people are holding on to their cars longer.

Meanwhile, demand for used cars has increased rapidly compared to pre-Covid times, and this mismatch between demand and supply has led to a 2-7 per cent increase in the price of used cars

"The demand for used cars has certainly increased from pre-pandemic times. The used vehicle market is facing supply constraints due to three factors: customers are holding on to their used vehicles and not selling them as they were doing before the pandemic, exchanges have been impacted due to continued challenges in the new car market, and repossessions have virtually stopped since April due to the ongoing loan moratorium. Due to supply constraints and unpredictable nature of the lockdown, sales were impacted.

Conceptual background of domain problem

The growing world of e-commerce is not just restricted to buying electronics and clothing but everything that you expect in a general store. Keeping the general store perspective aside and looking at the bigger picture, every day there are thousands or perhaps millions of deals happening in the digital marketplace. One of the most booming markets in the digital space is that of the automobile industry wherein the buying and selling of used cars take place. Sometimes we need to walk up to the dealer or individual sellers to get a used car price quote.

However, buyers and sellers face a major stumbling block when it comes to their used car valuation or say their second hand car valuation. Traditionally, you would go to a showroom and get your vehicle inspected before learning about the price. So instead of doing all these stuff we can build a machine learning model using different features of the used cars to predict the exact and valuable car price.

Review of Literature

Here in this project as per the requirement of our client, I have scraped the data from online retail site and on the bases of that data by analysing the different features we understood which features from our data are important to predict the price of the car and how they are affecting the price. So that the user can guess their car prices based on the features.

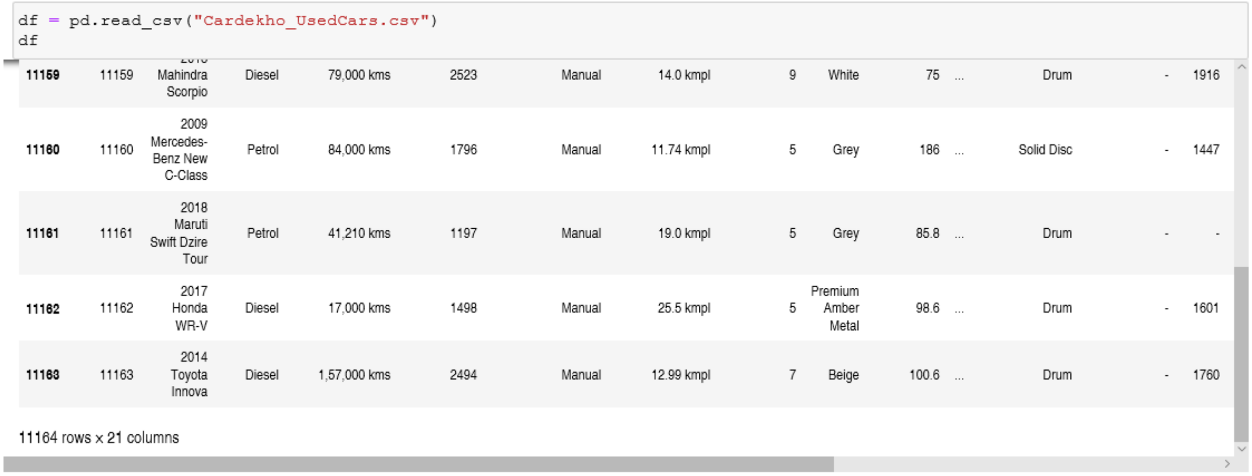
Motivation for the Problem Undertaken

Here for this project I have worked on the bases of client's requirement; and followed all the steps which have been instructed to me for building the model.

Analytical Problem Framing

- ### Mathematical/Analytical modeling of the problem

  I have scraped the required data from cardekho.com with different features and car price as a target variable. And loaded the data into python using jupyter notebook and did analysis.



```
df = pd.read_csv("Cardekho_UsedCars.csv")
df
```

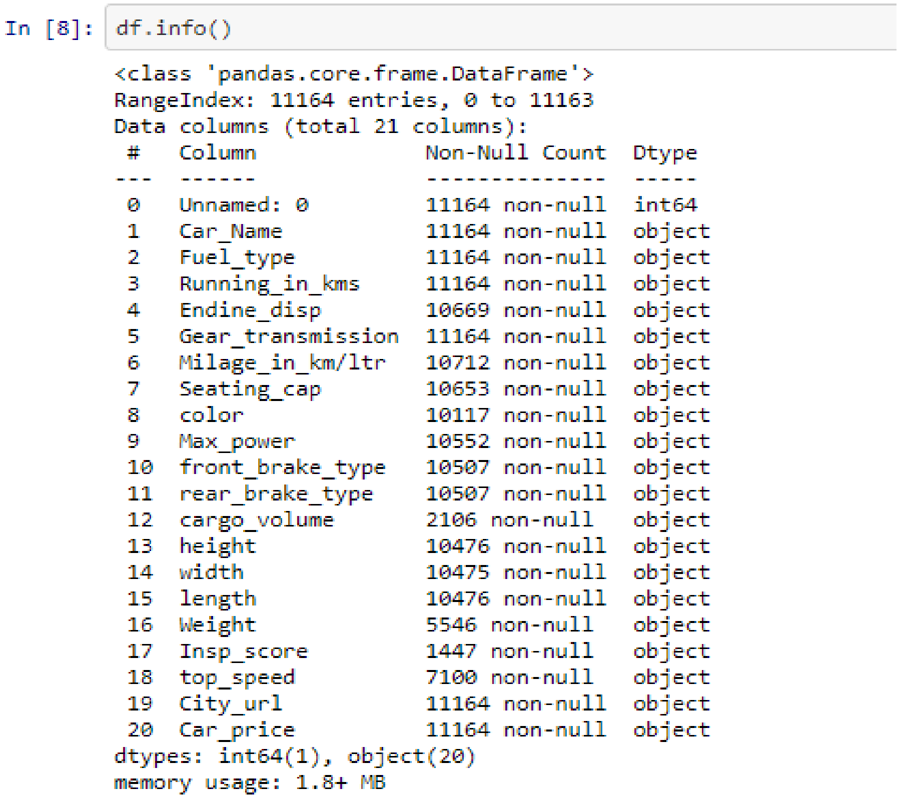
11159	11159	Mahindra Scorpio	Diesel	79,000 kms	2523	Manual	14.0 kmpl	9	White	75	...	Drum	-	1916
11160	11160	2009 Mercedes- Benz New C-Class	Petrol	84,000 kms	1796	Manual	11.74 kmpl	5	Grey	186	...	Solid Disc	-	1447
11161	11161	2018 Maruti Swift Dzire Tour	Petrol	41,210 kms	1197	Manual	19.0 kmpl	5	Grey	85.8	...	Drum	-	-
11162	11162	2017 Honda WR-V	Diesel	17,000 kms	1498	Manual	25.5 kmpl	5	Premium Amber Metal	98.6	...	Drum	-	1601
11163	11163	2014 Toyota Innova	Diesel	1,57,000 kms	2494	Manual	12.99 kmpl	7	Beige	100.6	...	Drum	-	1760

11164 rows x 21 columns

After loading the .csv file I have saved it into a data frame, we can see the data set is having 11164 rows and 21 different columns. Looking at the data we came to know that we need to go through various data processing, data cleaning as well as feature engineering steps.

Data Processing:

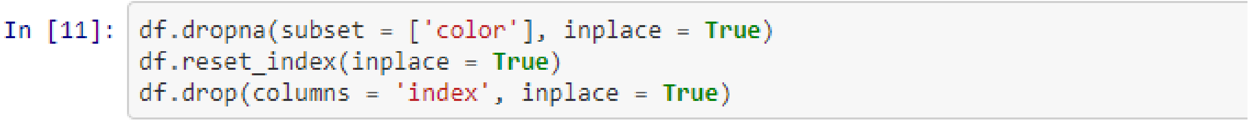
I have checked for duplicate rows and I didn’t found any duplicate row in our data. Many missing values were there and along with that columns have been entered with ‘-‘ and ‘null ‘ values which have been replaced by null values and treated as null values.



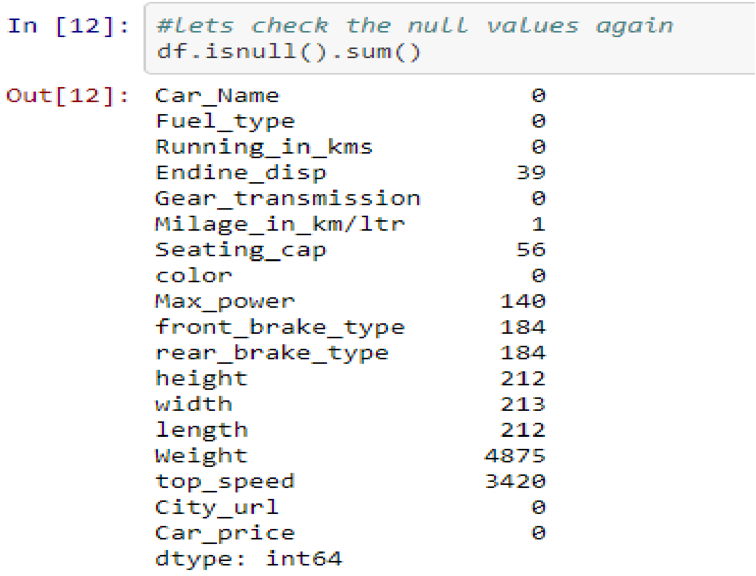
```
In [8]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11164 entries, 0 to 11163
Data columns (total 21 columns):
 #   Column             Non-Null Count  Dtype 
---  ------             --------------  ----- 
 0   Unnamed: 0         11164 non-null  int64 
 1   Car_Name           11164 non-null  object
 2   Fuel_type          11164 non-null  object
 3   Running_in_kms     11164 non-null  object
 4   Endine_disp        10669 non-null  object
 5   Gear_transmission  11164 non-null  object
 6   Milage_in_km/ltr   10712 non-null  object
 7   Seating_cap        10653 non-null  object
 8   color              10117 non-null  object
 9   Max_power          10552 non-null  object
 10  front_brake_type   10507 non-null  object
 11  rear_brake_type    10507 non-null  object
 12  cargo_volume       2106 non-null   object
 13  height             10476 non-null  object
 14  width              10475 non-null  object
 15  length             10476 non-null  object
 16  Weight             5546 non-null   object
 17  Insp_score         1447 non-null   object
 18  top_speed          7100 non-null   object
 19  City_url           11164 non-null  object
 20  Car_price          11164 non-null  object
dtypes: int64(1), object(20)
memory usage: 1.8+ MB
```

There are some columns with continuous data but here it is showing their data type as object type. It is may be because of the presence of some string values between them. That’s why we need to go through data processing.

And I have observed lot of null values from same rows in my data, which may give us trouble in model building, so I will drop some data where column color is having null values, by which we will lose arround 1047 entries but it will lead to a better model.



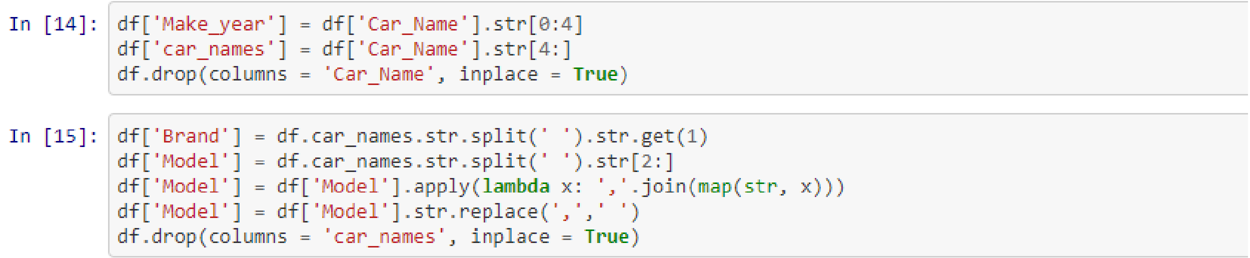
```
In [11]: df.dropna(subset = ['color'], inplace = True)
         df.reset_index(inplace = True)
         df.drop(columns = 'index', inplace = True)
```



```
In [12]: #Lets check the null values again
         df.isnull().sum()

Out[12]: Car_Name                0
         Fuel_type               0
         Running_in_kms          0
         Endine_disp            39
         Gear_transmission       0
         Milage_in_km/ltr        1
         Seating_cap            56
         color                   0
         Max_power             140
         front_brake_type      184
         rear_brake_type       184
         height                212
         width                 213
         length                212
         Weight               4875
         top_speed            3420
         City_url                0
         Car_price               0
         dtype: int64
```

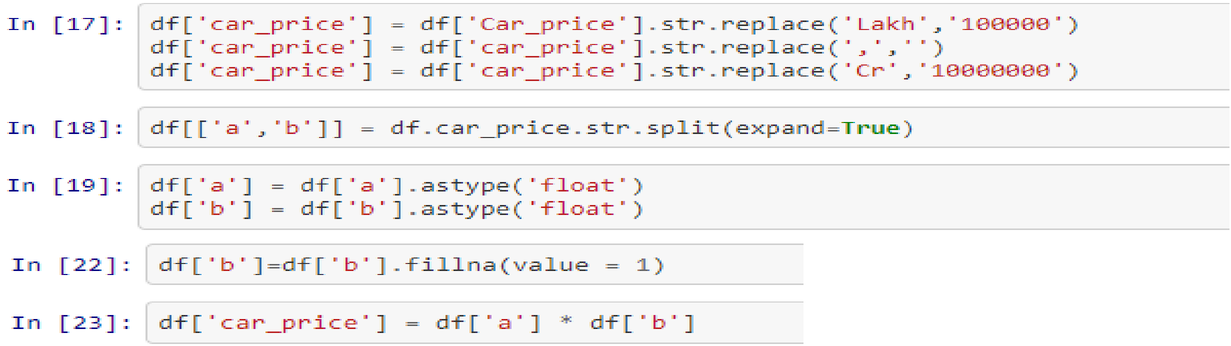
Great we have reduced percentage of null values to the better extent, still columns like Weight and top_spped are having huge number of missing values we will replace them by appropriate action.

As the column ‘Car_Name’ is containing the make year, Brand name as well as the model of the car; I will create three different columns for these respective features by using ‘Car_Name’



```
In [14]: df['Make_year'] = df['Car_Name'].str[0:4]
         df['car_names'] = df['Car_Name'].str[4:]
         df.drop(columns = 'Car_Name', inplace = True)

In [15]: df['Brand'] = df.car_names.str.split(' ').str.get(1)
         df['Model'] = df.car_names.str.split(' ').str[2:]
         df['Model'] = df['Model'].apply(lambda x: ','.join(map(str, x)))
         df['Model'] = df['Model'].str.replace(',',' ')
         df.drop(columns = 'car_names', inplace = True)
```

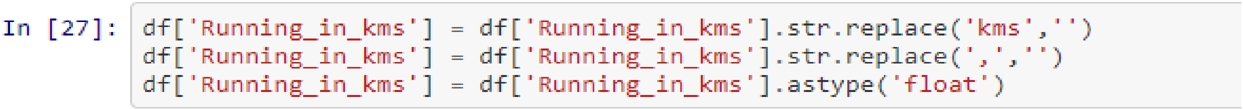
Our target variable that is car_price should cantain continuous data, but there are some string values like 'Lakh', 'Cr' and ','. I will replace 'Lakh' by 100000 and 'Cr' with 10000000 and comma by empty place. Then I will split it into two columns and after that multiply these two columns to get exact car price in numerical format



```
In [17]: df['car_price'] = df['Car_price'].str.replace('Lakh','100000')
         df['car_price'] = df['car_price'].str.replace(',','')
         df['car_price'] = df['car_price'].str.replace('Cr','10000000')

In [18]: df[['a','b']] = df.car_price.str.split(expand=True)

In [19]: df['a'] = df['a'].astype('float')
         df['b'] = df['b'].astype('float')

In [22]: df['b']=df['b'].fillna(value = 1)

In [23]: df['car_price'] = df['a'] * df['b']
```

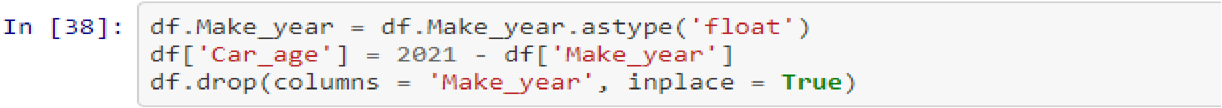
Running_in_kms

The column Running_in_kms is having some string values in it, and it should be continuous data, I will remove str values and commas between them and convert it to float data type.



```
In [27]: df['Running_in_kms'] = df['Running_in_kms'].str.replace('kms','')
         df['Running_in_kms'] = df['Running_in_kms'].str.replace(',','')
         df['Running_in_kms'] = df['Running_in_kms'].astype('float')
```

Make_year



```
In [38]: df.Make_year = df.Make_year.astype('float')
         df['Car_age'] = 2021 - df['Make_year']
         df.drop(columns = 'Make_year', inplace = True)
```

By using the car make_year I will create a new column named as Car_age which will tell us how old the car is.
Like this I have did some more data-processing which is required; and finally saved this processed data in .csv file. To check all the data processing steps please go through the github.

Data Pre-processing Done

Data pre-processing is a very important process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model. I have used some following pre-processing steps

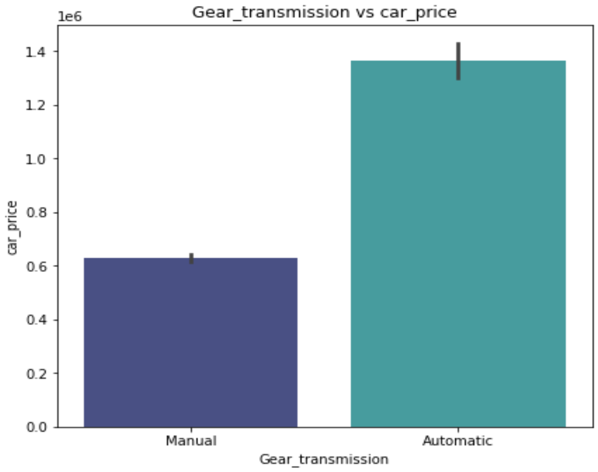
- Filling or Treating Missing values
- Outliers treatment
- Encoding
- scaling
- Skewness treatment

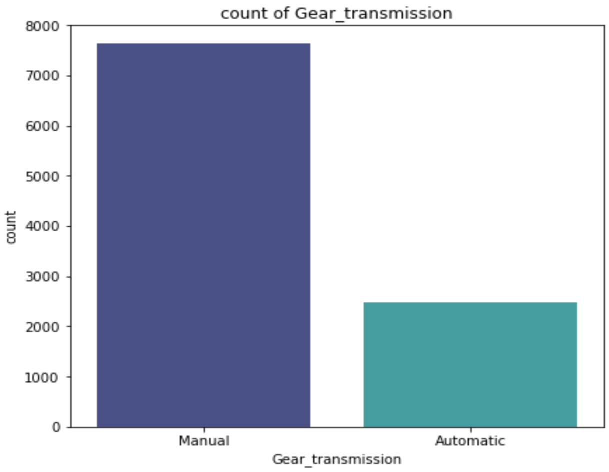
Data Inputs- Logic- Output Relationships

To analyse relationship between our features and the target variable I have did EDA to know the contribution of various features to the prediction. And got to know that which are the important features and which are not much. For EDA I have used different plots like distribution-plots, scatter-plots, box-plots, strip-plots, heat-map etc.

EDA

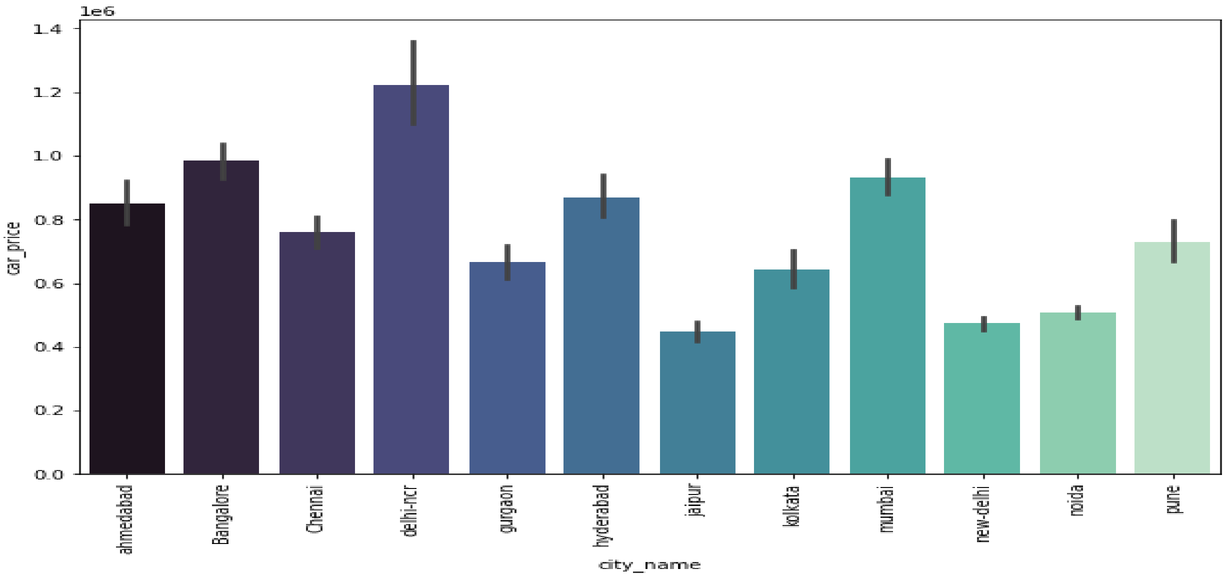
Here are some of the EDA examples which I have did



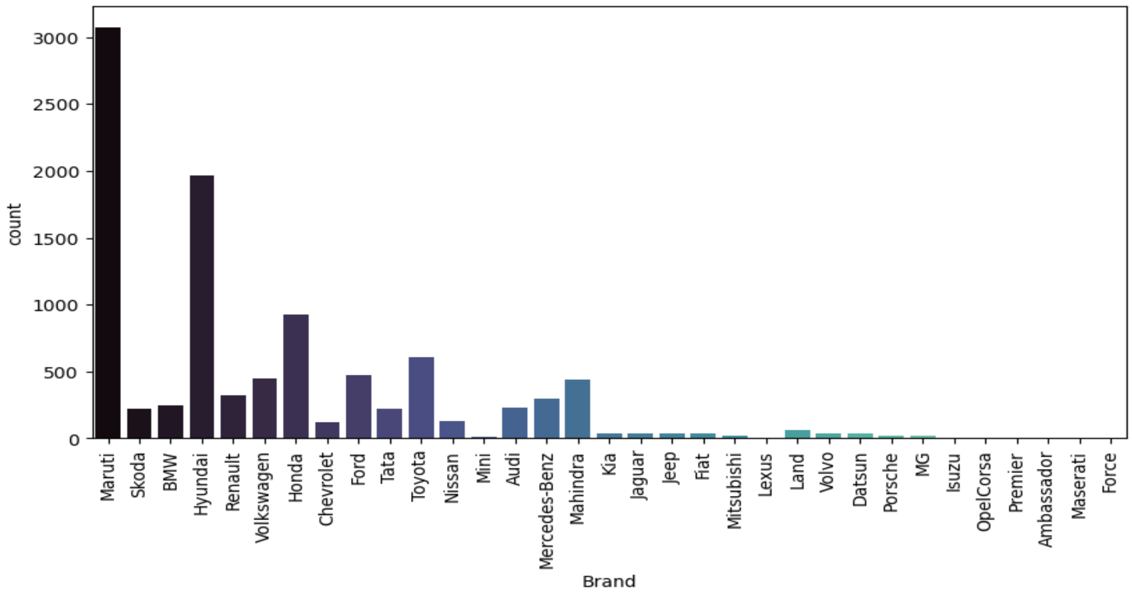


First plot is representing bar plot for Gear_transmission vs car_price, which will tell us that cars with manual gear transmission system are having less price compared to the cars which are with Automatic gear transmission.

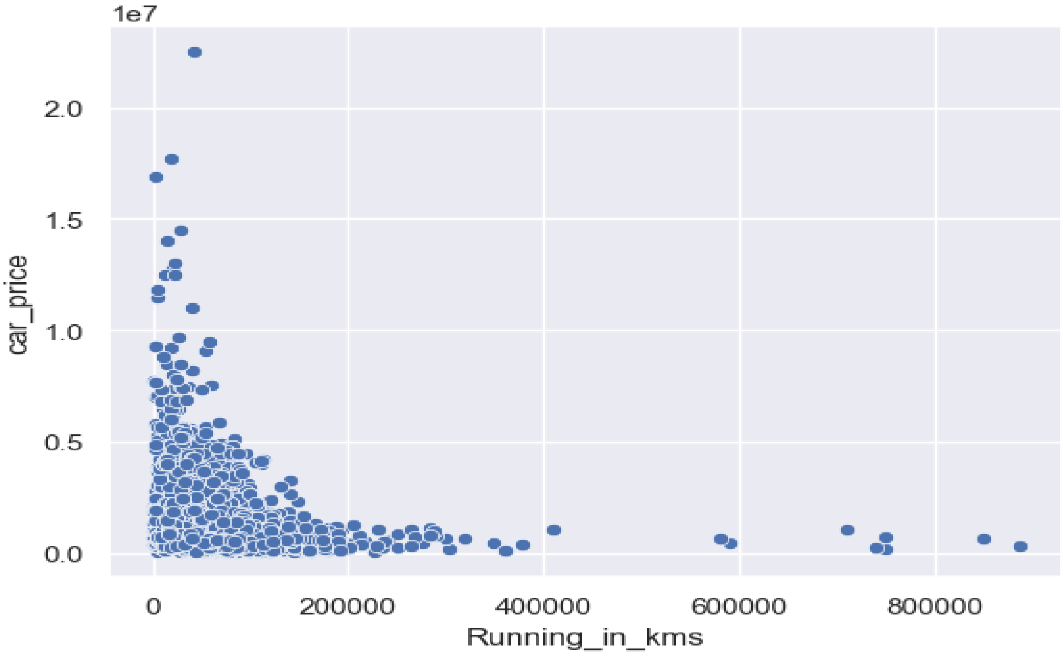
The second graph is count plot of Gear_transmission, by which we can conclude that arround 70% of the cars are with Manual gear transmission system.



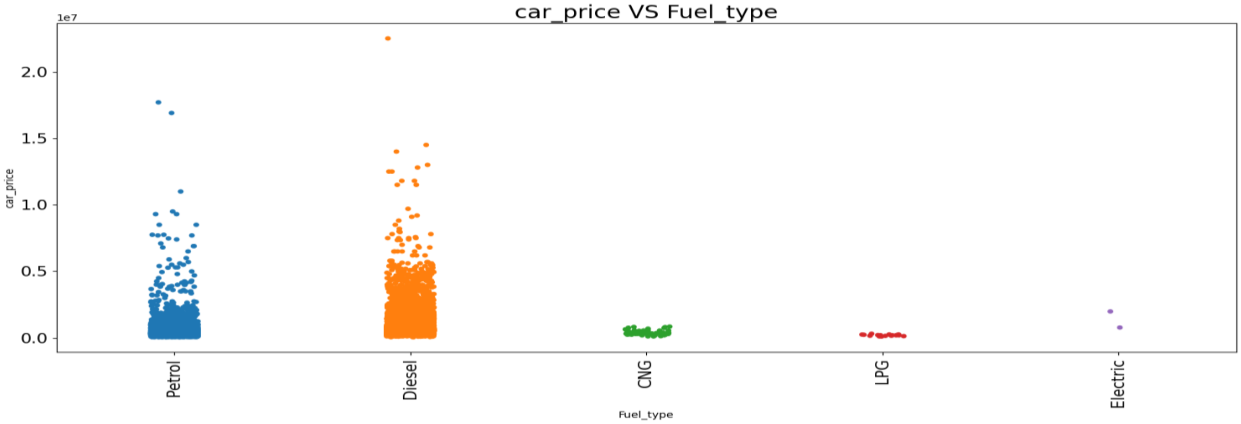
By looking at the above bar plot we came to know that the cars from the city delhi-ncr having higher prices and the cars from the city Jaipur are cheaper than other cities.

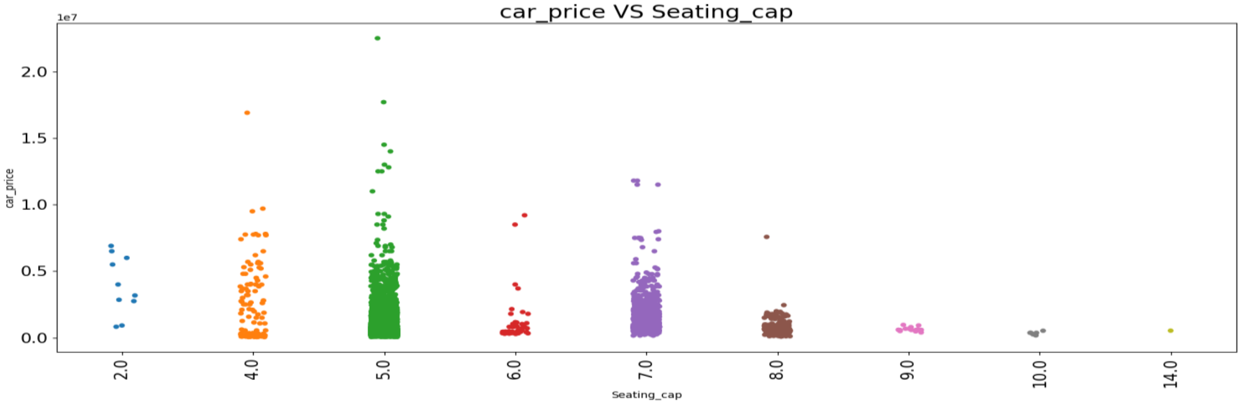


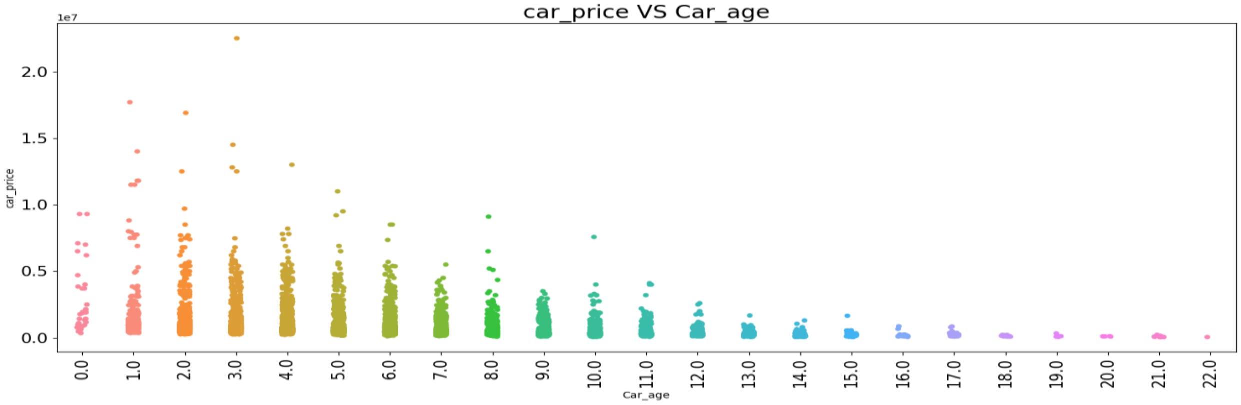
The above count plot is for car brand; looking at this plot we can conclude that we are having most of the cars from Maruti and Hyundai brand.



Looking at the above scatter plot we can say that the prices of cars are higher which are with less running.

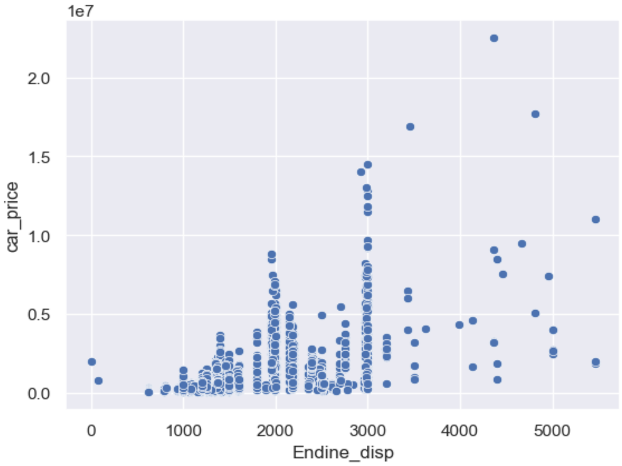


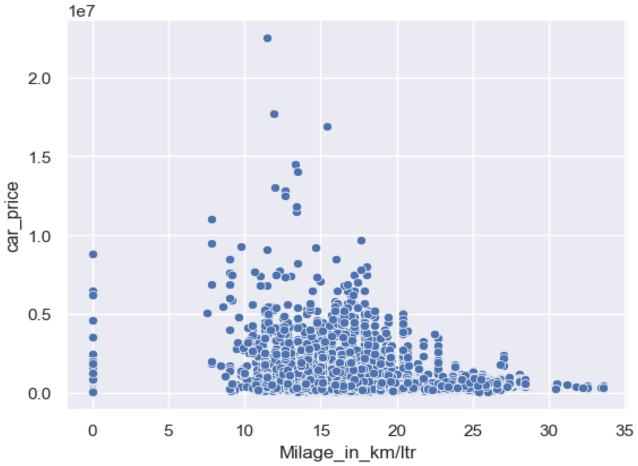




Observations

- More numbers of cars are using petrol and diesel as fuel; and these cars having wide range of price from minimum to maximum. Very few cars uses CNG, LPG, and Electricity as fuel type which are not much expensive when compared to that of the diesel and petrol cars.
- We can say that more number of cars having seating capacity of 5, 7 and 4 and these cars having higher prices than other cars as well. And only one car is observed with the seating capacity of 14.
- Looking at the graph for the car age we can conclude that the older cars are having very lower prices in market when compared to newer cars

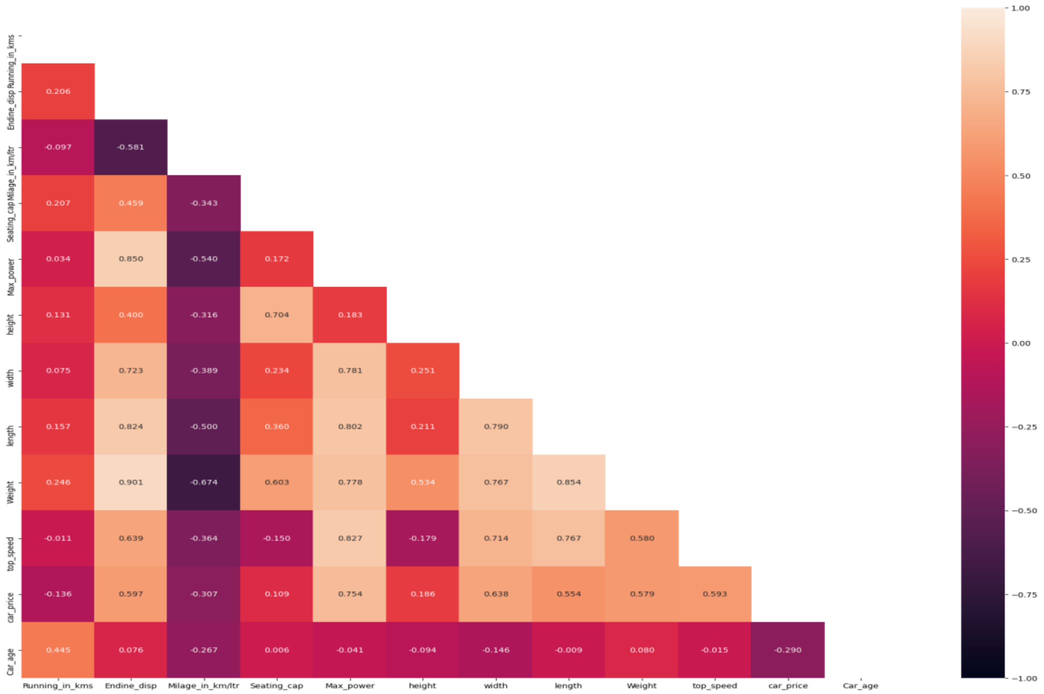


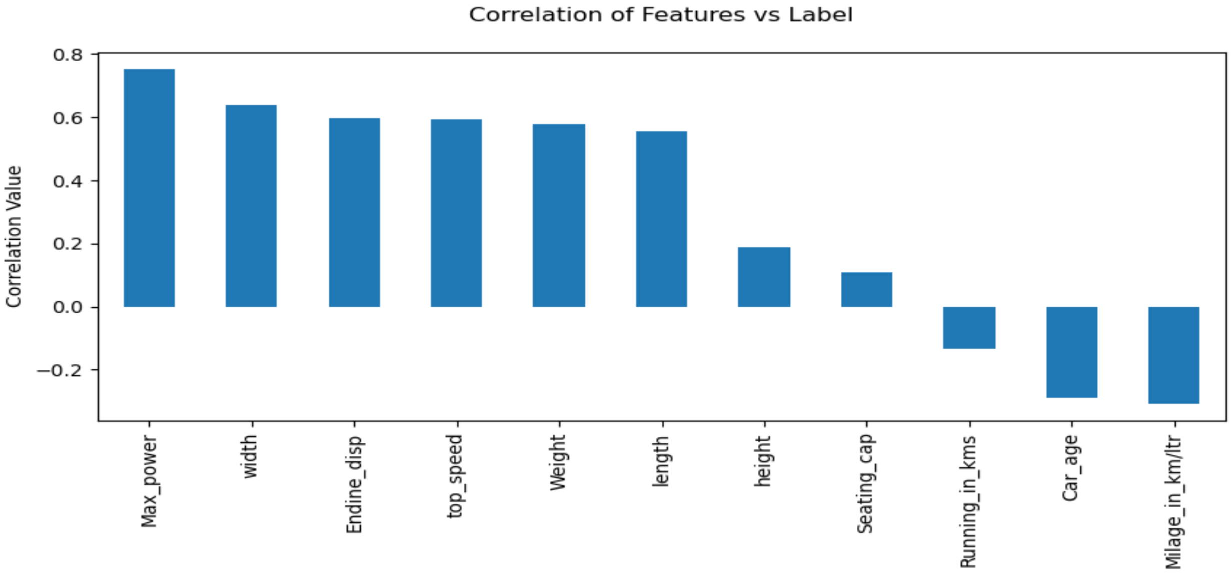


The first scatter plot is showing relation between Engine_disp and car_price. Looking at this plot we can say that as engine disp or engine cc increases the price of the car increases. And there are less number of cars which are having engine cc more than 3000.

Second plot is Milage vs car_price. We can see the car prices are higher some cars are with 0 milage, which is not realistic. And the cars having milage in the range 8 to 20 km/ltr are having higher prices.

Correlation Heat-map





The above correlation heat-map will tell us that all the features are in good relation with our target variable. Among all these features Max_power and car width are having higher correlation with the target variable.
Other features like Engine_disp, length, weight and top_speed are showing nearly equal relation with target variable. Columns Running_in_kms, car_age and milage_in_kms are negatively related with target variable.It seems like the column Engine_disp and Weight are having maximum coefficient of correlation between each other.

Hardware and Software Requirements and Tools Used

- For hardware I have used my laptop that have i5 processor and 8gb ram
- For software I have used Jupyter notebook
- For Tools I have use this following library-
  Numpy
  Pandas
  Seaborn
  Matplotlib
  Sklearn

Model/s Development and Evaluation

For this project we need to predict the prices of used cars, means our target column is continuous so this is a regression problem. I have used various regression algorithms and tested for the prediction. By doing various evaluations I have selected ExtraTreeRegressor as best suitable algorithm for our final model as it is giving good r2-score and least difference in r2-score and CV-score among all the algorithms used, other algorithms are also giving me better accuracy but those are differing more between r2-score and cv-scores.
For getting good performance as well as accuracy and to check my model for over-fitting and under-fitting I have used K-Fold cross validation.

I have used following algorithms and evaluated them

- ExtraTreeRegressor
- RandomForest Regressor
- DecisionTree Regressor
- LightGBM
- LinearRegression
- XGBRegressor

From all of these above models ExtraTreeRegressor was giving me good performance.

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.

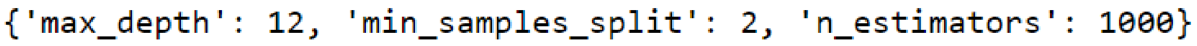
Testing and Evaluation of algorithms used

algorithm	R2-score	CV-score	R2_score – CV_score	Mae	Rmse
Linear regression	83.50	53.70	29.79	0.22	0.30
DecisionTreeRegressor	90.91	76.74	14.16	0.14	0.22
RandomForestRegressor	95.30	90.07	5.23	.108	0.163
XgboostRegressor	96.35	90.69	5.65	0.099	0.14
ExtraTreeRegressor	95.46	91.80	3.66	0.10	0.16
LGBMRegressor	96.39	90.48	5.91	0.10	0.14

Based on above observations we can see that for this problem; the tree related algorithms are giving us higher accuracy (r2-scores). Among all these algorithms ExtraTReeRegressor is giving good performance hence I have selected this as best suitable algorithm for our final model.

Hyperparameter Tuning

I have did hyperparameter tuning for ExtraTReeRegressor for the parameters like 'n_estimators', 'max_depth', 'min_samples_split' using GridSearchCV



```
{'max_depth': 12, 'min_samples_split': 2, 'n_estimators': 1000}
```

After running the code for above mentioned parameters I got the values which are indicated in the above figure as best parametric values for our final model.

Using these parametric values I trained our final model and got good r2-score of about 94.86 %. This means now we are able to predict the car price around 95% accurately.

Conclusion

Key findings of the study

According to this study and analysis of different parameters we got to know that the features like the number of kms that car has driven, Car brands, age of the car, Engine performance, mileage, braking system used are performing important role in predicting the car price.

The manufacturers like Land Rover, BMW, Benz cars are having costliest used cars in the market than other cars. New cars which are having less running are also getting more prices in the market compared to older cars. I have observed that there are huge numbers of cars from Maruti brand selling at lower prices.

We are having cars with different fuel types like petrol, diesel, CNG, LPG and some of the electric cars. Very few cars having CNG, LPG and electricity as the fuel and these are having lower prices in the market when compared to petrol and diesel variant. Also cars with automatic transmission variant are getting more prices in the market.

For this project I have also scraped the car dimensions which are also contributing to the price prediction. Cars with more height and weights are getting more prices than that of with lower height and weights.

Limitations of this work and scope for the future work

As we have scraped this data during the COVID-19 crises; during these days the automotive market is already struggling and it affected the used car prices as well. In future the conditions may be different compared to now, so our model may fail to predict the prices with the same accuracy in the future

And this data is collected from some of the well known cities of India; that means for different cities other than which we have considered the car's price may vary according to the local conditions.

The car price may affect due to any accident occurred or any kind of external damage which we have not considered here.

So we can say we have to take data on the bases of the respective current situations and feed to this model which will give the results accordingly to that time.