NAME OF THE PROJECT

Used Car Price Prediction

Submitted by: Ankur Kumar

ACKNOWLEDGMENT

I would like to thank My SME and fliprobo technologies ltd for giving me the opportunity to gather the information for the project.Websites like Analytics Vidya and Kaggle helped a lot in the process of making this project.

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. 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.

- Conceptual Background of the Domain Problem

We are about to deploy an ML model for car selling price prediction and analysis. This kind of system becomes handy for many people.

Imagine a situation where you have an old car and want to sell it. You may of course approach an agent for this and find the market price, but later may have to pay pocket money for his service in selling your car. But what if you can know your car selling price without the intervention of an agent. Or if you are an agent, definitely this will make your work easier. Yes, this system has already learned about previous selling prices over years of various cars.

So, to be clear, this deployed web application will provide you will the approximate selling price for your car based on the fuel type, years of service, showroom price, the number of previous owners, kilometres

driven, if dealer/individual, and finally if the transmission type is manual/automatic. And that's a brownie point.

Any kind of modifications can also be later inbuilt in this application. It is only possible to later make a facility to find out buyers. This a good idea for a great project you can try out. You can deploy this as an app like OLA or any e-commerce app. The applications of Machine Learning don't end here. Similarly, there are infinite possibilities that you can explore. But for the time being, let me help you with building the model for Car Price Prediction and its deployment process.

- Review of Literature

  The Car price Prediction project covers the scrapping of data through Cardekho.com which is data collection.

  Data Collection followed by converting the collected data into meaningful format which is a csv file or xlsx file.

  Further before moving forward in the project all the necessary library needs to be imported in the jupyter notebook.

  Importing the file for the prediction of car price.

  Cleaning the data,visualization and preprocessing of data.

  After the process the model is built with a machine learning algorithm.

  The algorithms used in preparation of predictive model is

  Random Forest Regressor

  Linear Regression

KNN Regressor

XGBoost Regressor

- Motivation for the Problem Undertaken

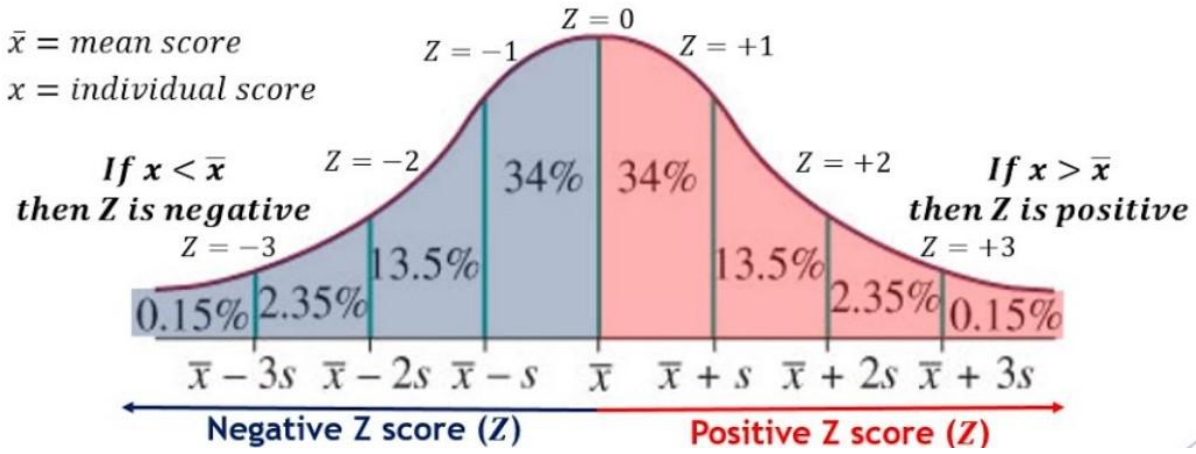
The objective behind building the model to create the predictive model which can predict the used car price through various inputs given by end user such as Brand, Model,Max Power, Gear type etc.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

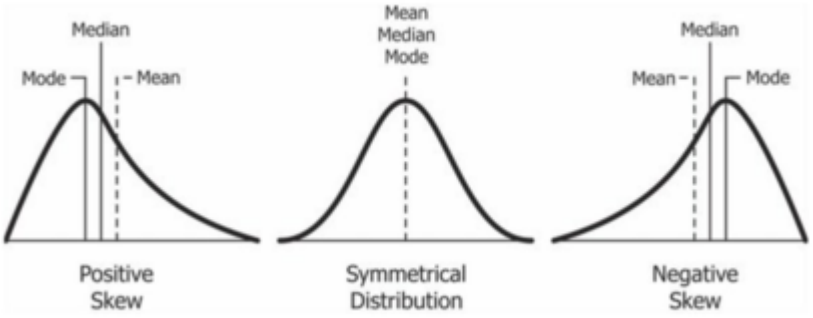
Various mathematical tools required to build the model such as

Zscore-



Through this method we can remove outliers present in data so that it should be normally distributed which is essential for the model building.

2. SKewness-



- Data Sources and their formats

The data source is taken from cardekho.com where data is scrapped extensively through web scraping using selenium tool and further it is converted into csv file for further prediction.

- Data Preprocessing Done

Various steps involved in data preprocessing

- Acquire the dataset. ...
- Import all the crucial libraries. ...

- Import the dataset. ...
- Identifying and handling the missing values. ...
- Encoding the categorical data. ...
- Splitting the dataset. ...
- Feature scaling.

- Data Inputs- Logic- Output Relationships

  Various independent Features like brand, model which shows the change in car price which change of the input features

- Hardware and Software Requirements and Tools Used

  Hardware required:

  • Processor: core i5 or above

  • RAM: 8 GB or above

  • ROM/SSD: 250 GB or above

  Software Required:

  Anaconda
  Python Programming Language
  Selenium
  Chrome

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

Not every problem which has numbers involved in it is a machine learning problem. There's a great saying, if the only tool you have is a hammer, you tend to see every problem as a nail.

Machine Learning can only be used in the following problems:

1. **Learning from the data is required.**
2. **Prediction of an outcome is asked for.**
3. **Automation is involved.**
4. **Understanding the pattern is required like that in the case of user sentiments.**
5. **Same as point d for building recommendation systems.**
6. **Identification/Detection of an entity/object is required.**

There are many other bullets to it too but the fundamentals are the ones mentioned above. A use case may have more than one bullet. There may be things where one might simply not need to have machine learning practice for the same in such a case he should go with one because simplicity is what is valued everywhere.

Now coming up with how to solve a machine learning problem. A following stepwise approach would help you solve almost any machine learning problem.

Step 1(a). How to solve a Machine Learning problem?

Stepwise approach

1. **Read the data (from csv, json etc)**
2. **Identify the dependent and independent variables.**
3. **Check if the data has missing values or the data is categorical or not.**
4. **If yes, apply basic data preprocessing operations to bring the data in a go to go format.**

5. **Now split the data into the groups of training and testing for the respective purpose.**
6. **After splitting data, fit it to a most suitable model. (How to find a suitable model is answered below)**
7. **Validate the model. If satisfactory, then go with it, else tune the parameters and keep testing. In a few cases, you can also try different algorithms for the same problem to understand the difference between the accuracies.**
8. **From step 7 one can also learn about accuracy paradox.**
9. **Visualize the data.**

Visualising the data is important because we need to understand where our data is heading and also it looks more representative while storytelling about the data.

- Testing of Identified Approaches (Algorithms)

  The List of algorithm used are

  Random forest Regressor with R2_Score is 90%

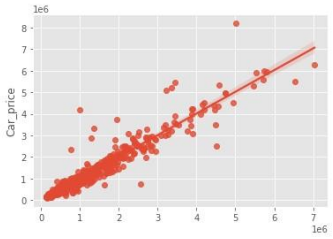
  XGBoost Regressor with R2_Score is 93%

  KNN Regressor with R2_score is 88 %

- Run and Evaluate selected models

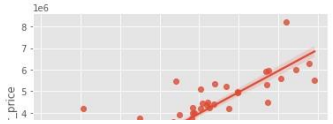
Random Forest

```
In [157…
rf=RandomForestRegressor()
rf.fit(x_train,y_train)

Out[157…
RandomForestRegressor()

In [158…
model(rf,x_train,x_test,y_train,y_test,train = True)

Traning r2_score:  98.83794250007168

In [159…
model(rf,x_train,x_test,y_train,y_test,train = False)

The testing Score- 92.04296156006917
MSE:  63057705667.0097
MAE:  106195.81154771452
RMSE  251112.9340894445
```



XGBoost

```
In [161…
xgb=XGBRegressor()
xgb.fit(x_train,y_train)

Out[161…
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, ...)

In [162…
model(xgb,x_train,x_test,y_train,y_test,train = True)

Traning r2_score:  99.68163402561375

In [163…
model(xgb,x_train,x_test,y_train,y_test,train = False)

The testing Score- 93.27566100609464
MSE:  53288845125.46947
MAE:  101866.94486768244
RMSE  230843.76778563778
```

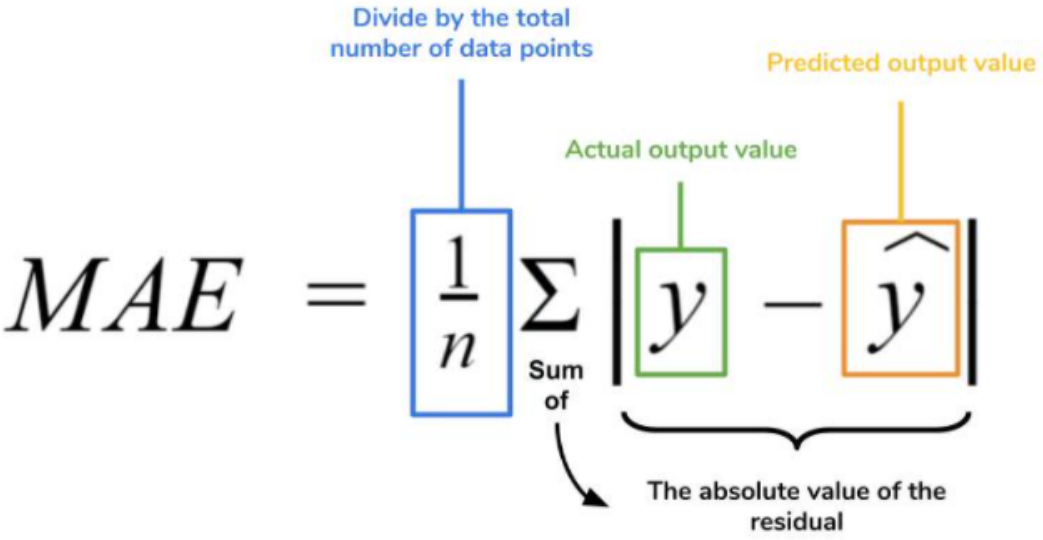


- Key Metrics for success in solving problem under consideration

  Mean Squared Error- It means the sum of all predicted value difference with actual value which should be lesser to get accurate model.

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2$$

  Mean Absolute error-



- Visualizations

Univariate Analysis:-

Uni means one and variate means variable, so in univariate analysis, there is only one dependable variable. The objective of univariate analysis is to derive the data, define and summarize it, and analyze the pattern present in it. In a dataset, it explores each variable separately. It is possible for two kinds of variables- Categorical and Numerical.

Bivariate Analysis:-

Bi means two and variate means variable, so here there are two variables. The analysis is related to cause and the relationship between the two variables. There are three types of bivariate analysis.

CONCLUSION

- Key Findings and Conclusions of the Study

In this article, we tried predicting the car price using the various parameters that were provided in the data about the car. We build machine learning and deep learning models to predict car prices and saw that machine learning-based models performed well at this data than deep learning-based models.

- Learning Outcomes of the Study in respect of Data Science

  Through this prediction the outcome shows that we can predict the price of a car by giving various inputs.