CAR PRICE PREDICTION PROJECT

SUBMITTED BY : **SINDHU SHREE N**

INTERNSHIP BATCH : **19**

ACKNOWLEDGMENT

With great pleasure, I sincerely express my deep sense of gratitude and heartfelt thanks to several individuals from whom I received impetus motivation during the internship project work. I am very grateful to **SHUBHAM YADAV** sir internship 19th batch guide, for the help and I do express my deep gratitude to sir for his guidance, keen interest and constant encouragement throughout the internship project work. I am thanking him for personal concern, affinity and great spirit with which he has guided the work.

INTRODUCTION

Business Problem Framing

There are lots of individuals who are interested in the used car market at some points in their life because they wanted to sell their car or buy a used car. In this process, it's a big corner to pay too. To be able to predict used cars market value can help both buyers and sellers.
much or sell less then it's market value. Machine Learning is a field of technology developing with immense abilities and applications in automating tasks, where neither human intervention is needed nor explicit programming.

The power of ML is such great that we can see its applications trending almost everywhere in our day-to-day lives. ML has solved many problems that existed earlier and have made businesses in the world progress to a great extent.

Conceptual Background of the Domain Problem

The project contains car price dataset. And we are supposed to predict the selling price of the car based on multiple features. Here we have used regression technique for selling price prediction. 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.

Review of Literature

All possible information from all the available data tables more the information, more than for EDA and feature Engineering. its take more important to take the average during the aggregation of data from tans the table rather than taking the counts before the loan was applies no future information steps into the data to be used for modelling.

Motivation for the Problem Undertaken

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.

Analytical Problem Framing

Mathematical/ Analytical Modelling of the Problem

1. Included in 'Data-cleaning ipynb'.
2. Selecting relevant features.
3. Exploratory the data analysis and data cleaning.
4. Null value imputation
5. Handling outliers
6. Training a machine learning model
7. Hyperparameter tunning
8. Evaluate the model
9. Predictions of the model

Data Sources and their formats

We have scrapped the data using webscraping in the websites like olx, cars24 and stored them in a excel sheet.

Creating index on all the table to be joined this will speed up and processing here we are using EDA feature Engineering, Data visualization and statics approach perform data cleaning, outlier handling, missing values build model etc.

Data Pre-processing Done

Creating index on all the table to be joined this will speed up and processing here we are using EDA feature Engineering, Data visualization and statics approach perform data cleaning, outlier handling, missing values build model etc.

Partitioning and splitting that dataset account when credit loan default was applied as that dataset is skewed, stratification is used allocate the samples evenly based on sample classes so that training set and test set have similar ration of classes.

Data Inputs- Logic- Output Relationships

Since it is a regression model where the target feature is continuous data, there is no imbalance of the data so we used **correlation matrix** to find the relation between the input and output features.

Set of assumptions (if any) related to the problem under consideration

I am not taken any presumptions of this problem

Hardware and Software Requirements and Tools Used

Hardware requirements:

PROCESSOR: Intel(R) Core(TM) i3 CPU

MONITOR : Any display unit

HARD DISK : 240GB SSD

RAM : 8.00GB

Software requirements:

OPERATING SYSTEM: Windows 10 Pro

FRONT END : Jupyter Notebook (Anaconda3)

BACK END : Excel 2013

Tools Used:

1) Pandas Library
2) Numpy Library
3) Seaborn Library
4) Matplotlib
5) Scikit_learn

Model/s Development and Evaluation

Identification of possible problem-solving approaches (methods)

- Examining the data
- Checking the basic details (null value, D type, Shape etc)
- Identifying the target and independent features and perform EDA using Data visualization and Statistical approach accordingly
- Performing data cleaning, outliers handling, missing value imputation
- Feature engineering
- Performing hyperparameter tunning
- Evaluating the model again
- Making predictions

Testing of Identified Approaches (Algorithms)

Listing down all the algorithms used for the training and testing.

1) Linear Regression
2) Lasso Regression
3) Gradient Boosting Regression
4) Random Forest Regression
5) XGBoost Regression

Run and Evaluate selected models

1) **Linear Regression:** It is a basic and commonly used type of predictive analysis. Linear Regression fits a straight line or surface that minimizes the deprecancies between predicted and actual output values.

```
training_data_prediction=lr.predict(x_train)
```

```
from sklearn.metrics import r2_score
error=r2_score(y_train,training_data_prediction)
print("R squared error :",error)
```

```
R squared error : 0.7209171822160665
```

2) **Lasso Regression:** It is a type of linear model which is similar to the linear regression used for predictive models.

```
training_data_prediction=lasso.predict(x_train)
```

```
error=r2_score(y_train,training_data_prediction)
print("R Squared Error:",error)
```

```
R Squared Error: 0.7203470866688835
```

3) **Gradient Boosting Regression:** Gradient boosting is a machine learning technique for regression, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees.

```
training_data_prediction=gbr.predict(x_train)
```

```
error=r2_score(y_train,training_data_prediction)
print("R Squared Error:",error)
```

```
R Squared Error: 0.8377380199926404
```

4) **Random Forest Regression**: It is a type of ensemble learning method that uses numerous decision trees to achieve higher prediction accuracy and model stability. Every tree classifies a data instance based on attributes, and the forest chooses the classification that received most instances.

```
training_data_prediction=rf.predict(x_train)
```

```
error=r2_score(y_train,training_data_prediction)
print("R Squared Error:",error)
```

```
R Squared Error: 0.9834730297126856
```

5) **XGBoosting Regression:** Extreme Gradient Boosting regressor provides an efficient and effective implementation of gradient boosting algorithm which is used for regression predictive modelling.

```
training_data_prediction=xg.predict(x_train)
```

```
error=r2_score(y_train,training_data_prediction)
print("R Squared Error:",error)
```

```
R Squared Error: 0.9620008198398995
```

Interpretation of the Results

On comparing all the results of these five models mentioned above we have observed that the xgboost, random forest and gradient boost regressors are performing well but the training set and testing set score is high almost near in xgboost regression and also has a comparative cross validation score. Thus we have choosed this model for hyperparameter tuning and there also we got a improved result. Hence **XGBoosting Regressor** is the best fit for the present dataset.

CONCLUSION

Key Findings and Conclusions of the Study:

Most Regression problems in the real world are imbalanced. Also, almost always data sets have missing values. But in this case since the target feature price is a continuous data we don't have any data imbalance and here we covered strategies to deal with missing values in the datasets. We also explored different ways of building ensembles in sklearn.

Learning Outcomes of the Study in respect of Data Science:

There is no definitive guide of which algorithms to use given any situation. What may work on some data sets may not necessarily work on

others. Therefore, always evaluate methods using cross validation to get a reliable estimate.

Sometimes we may be willing to give up some improvement to the model if that would increase the complexity much more than the percentage change in the improvement to the evaluation metrics.

In some Regression problems, false negative is a lot more expensive that false positives. Therefore, we can reduce cut-off points to reduce the false negative.

Missing values sometimes add more information to the model than we might expect.one way of capturing it is to add binary features for each feature that has missing values.

Limitations of this work and Scope for Future Work

- Limited Access to information
- Time Limits
- Conflicts on biased views and personal issues
- How to structure my project research limitation correctly
- How to set my project research limitation.
- Formulation of my objectives and aims
- Implementation of my data collection methods
- Scope of discussions
- Finding my error on the codes

Concluding Thoughts

This study used different models in order to predict used car prices. However, there was a relatively small dataset for making a strong inference because number of observations was only 8128 rows. Gathering more data can yield more robust predictions. Secondly, there could be more features that can be good predictors.