Simple Explanation

on

Car Price Prediction

Introduction

When we are to buy a car, we go to car showroom, or a car dealership. In there, a sale executive (or may be a sale person) will help us in every details of the car, like this car is this much, this care is the latest model, etc. And also, we can go to a second hand car dealership to buy a used car. In there too, the sale people will help us in every details of the car. However, we can substitute this work (or the help given by the sale people) with a Machine Learning model.

This project Car Price Prediction is a simple project to predict the price of a car. In this project, the quartile method (specifically using Interquartile Range, IQR) of removing outliers is used to remove the outliers. Then, the algorithms RandomForestRegressor, GradientBoostingRegressor and XGBRegressor are used and obtained the R^2 and RMSE as 0.6080 and 0.04, 0.6295 and 0.0413, and 0.6512 and 0.0401 respectively.

Note: (Here, with RandomForestRegressor and GradientBoostingRegressor, I used GridSearchCV to get optimum parameters (hyper parameter tuning). If you want to check, you can prefer the Github file.)

Dataset Overview

Details of the dataset are as follows:

- Source:
  The dataset used is from Kaggle platform. Here's the link to the dataset: https://www.kaggle.com/datasets/deepcontractor/car-price-prediction-challenge

- Number of Columns and Rows:
  The dataset is in csv format, and it has 19237 rows and 18 columns, Price column as target.

Tools

Tools that are used in this project are python programming language, sklearn, numy, pandas, and xgboost.

Methodology

Note: For real academic project, one should add figures (like conceptual diagrams or figures and others diagrams)

First, the dataset is downloaded from Kaggle. Now, the second step is the preprocessing. After dropping some columns (like ID, Levy, etc.), all the columns which have categories are converted into numerical forms using OneHotEncoder. And only one column that is Leather interior, which has only Yes and No is converted into numerical forms using LabelEncoder. The Mileage column has 'km', and this is removed, because we cannot send an entry like '100 km' to algorithm. After this, some columns (not the one hot encoded or label encoded ones) are scaled using StandardScaler from sklearn.

After scaling, we need to check about outliers. There are some ways to check and remove outliers but the quartile method is used in this project. After detecting and removing the outliers, we now have 7746 rows and 13 columns.

Now, next step is to check whether the data are linear related or not. Scatter plot is used to check this, and we know there is no linear relationship (we can perform log transformation, and check if the distribution follows normal distribution). So, we import RandomForestRegressor, GradientBoostingRegressor and XBGRegressor. Then, the dataset is split into train set as 80 percent and test set as 20 percent as conventional one to train the above algorithms.

To evaluate the trained algorithms, R-Squared and RMSE are used. R-Squared measures the proportion of variance in the dependent variable that is explained by the independent variable(s) in the model. It indicates how well the model fits the observe data. And RMSE is a measure of how well the model's predictions match the actual values.

Note: If you are writing an academic project's report, add references for anything you borrow from other sources.

The algorithms are trained using X_train and y_train. The R-Squared and RMSE of the algorithms are as follows as pair: 0.6080 and 0.04 for RandomForestRegressor, 0.6295 and 0.0413 for GradientBoostingRegressor, and 0.6512 and 0.0401 for XGBRegressor.

Conclusion

With the help of Machine Learning, we can build this kind of application to predict the car price in a car dealership. Using the idea behind this project, one can implement car price prediction application. For future use, you can use pickle to save the model, and integrate it with a website or web app.