ECE 9063 Data Analytics Foundation

Assignment 1: Forecasting

**Student Name:** Jianping Ye

**Student Number:** 250887769

**Instructor:** Katarina Grolinger

**Problem Statement**

The used car market is a perfect place for finding cars in decent conditions and with fair prices. It is also the reason that the market has been growing in recent years. However, it is difficult to decide the opportune moment to buy or sell as the price fluctuates constantly . And there are many factors contributing to the volatile prices. For instance, cars have diverse conditions and the market trend is not stationary all the time. It will be beneficial for both buyers and sellers if we could make a model to predict the value of cars such that they can make a more confident decision. With the help of a suitable model, buyers will be able to make sure the car is worthy of its price, and sellers can get a more accurate price estimation in accordance with other cars having similar conditions. In this report, the forecasting problem is defined as follow: predict the price of a used car in the current year given a set of relevant attributes.

**Dataset Description**

Link to the data: <https://www.kaggle.com/adityadesai13/used-car-dataset-ford-and-mercedes>

These datasets contain scraped data of used cars in the British market and are separated into files specific for each car manufacturer. In this report, the dataset selected is “Audi.csv”. It contains 9 attributes and 10668 samples. The dataset is suitable for this assignment as it has adequate attributes and samples. With over 10,000 samples, it is easier to strike a balance between computational time and reliability of the model . The attributes are listed below:

* Model: The model code of the car
* Year: registration year of the car
* Price: price on the market
* Transmission: type of gearbox, either manual, automatic, or semi-auto
* Mileage: distance used so far
* fuelType: type of fuel the engine uses, either diesel, petrol, hybrid, or other
* tax: road tax
* mpg: miles per gallon
* engineSize: size of engine in litres

Noticeably, model, transmission, and fuel type have nominal data that needs to be transformed into numerical values. All the attributes in the dataset are considered in the model as they are all important factors while estimating the price of cars in the real-world.

**Algorithms Overview**

The first algorithm I used is multivariate linear regression. In this hypothesis, multiple independent variables may influence the dependent variable linearly. In the context of our dataset, multiple factors are affecting the price of cars and there is a potential linear relationship between them, which makes it a straight-forward and promising model.

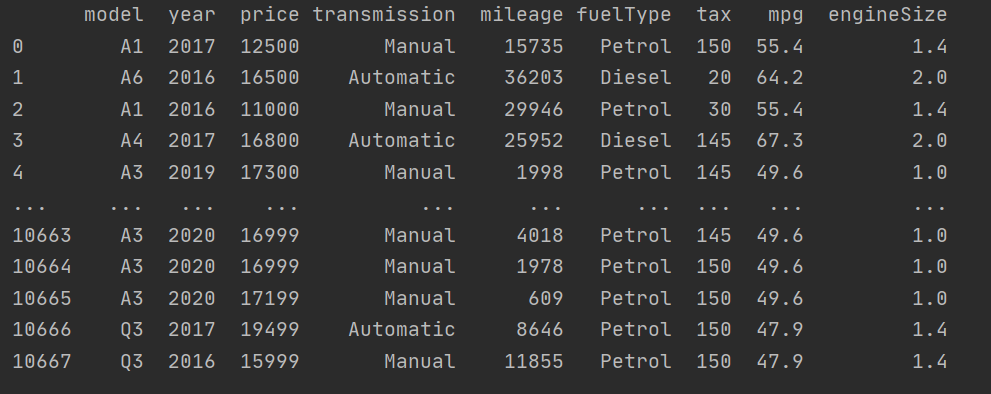
The second algorithm is support vector regression (SVR). Support vector regression adheres to the basic principle of support vector machine, which is the maximum margin characteristic, but it is used for regression instead. Linear kernel is used in the SVR.

The third algorithm is random forest regression. Random forest regression utilizes the idea of ensemble learning, which is a technique that can take advantages from multiple machine learning algorithms such that it can produce a more accurate prediction.

The python package used in this report to train models is sklearn. Before applying the above-mentioned algorithms, it is necessary to normalize our dataset as it is a common requirement of many machine learning algorithms and it is also considered good practice. The normalization technique used is standardization, which will make the data have zero mean and unit variance.

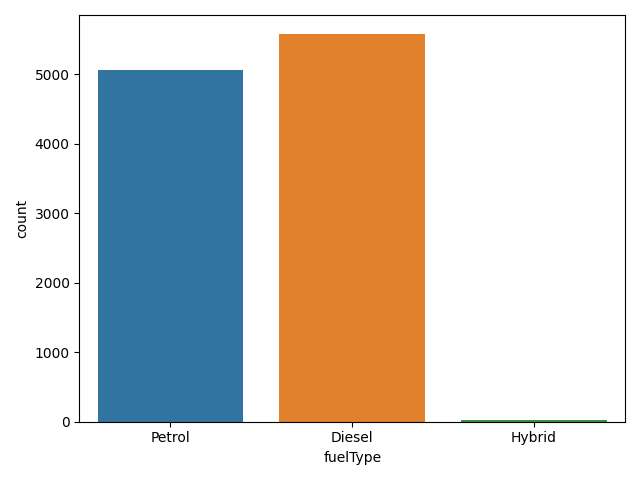
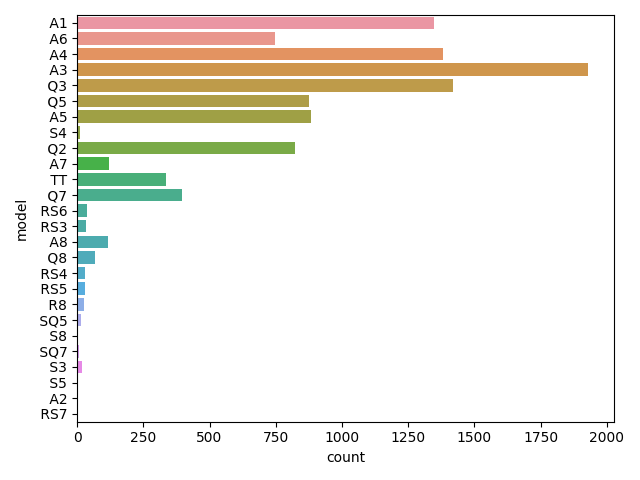
**Detailed Procedures**

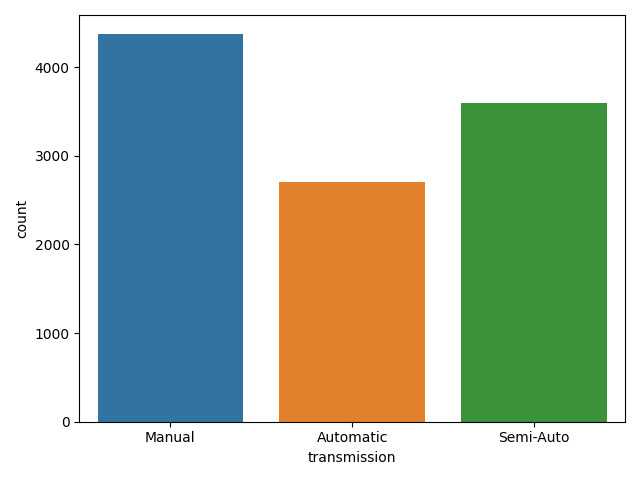
1. Exploratory Data Analysis



After importing the csv file, I printed the data to have an overview of its structure.

Clearly, model, transmission, and fuelType has nominal values. I plotted the counts of each type of observation versus that attribute.



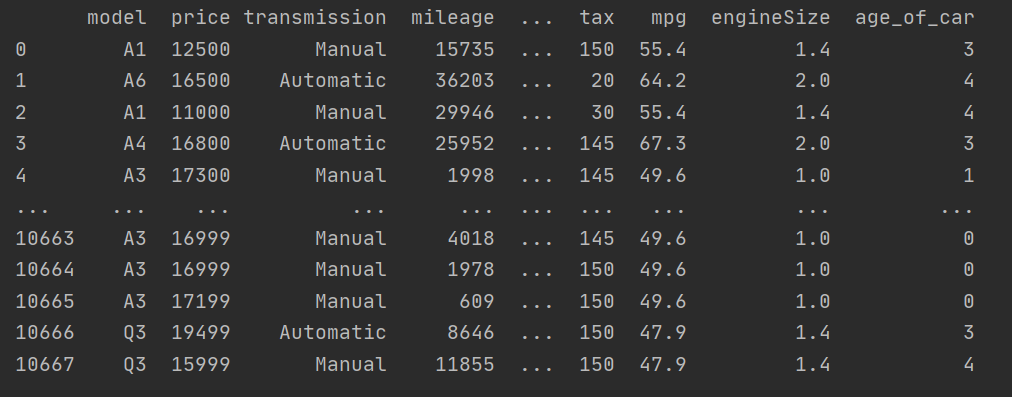


It is observed that there are finite number of possibilities in each attribute’s value, so one-hot encoding is selected as encoding scheme.

图示

描述已自动生成From the pair plot graph, we can see the data is dense (No zeros). Therefore, standardization will be used to normalize the data.

1. Preprocessing



Firstly, compute a new attribute called “age\_of\_car” by subtracting 2020 from the ‘year’ attribute. The result is number of years that car has been used since its registration. I believe the ‘age of car’ will be a more informative and directly related attribute than its original.

In addition, use one-hot encoding to transform categorical features into numerical features. Moreover, separate the dataset into X (features) and Y (target) set. X set is independent variables including all the attributes except ‘price’. Y set is the target variable --‘price’ attribute. Lastly, apply standard scaling (Standardization) on X set to normalize the data. At this stage, we can split the normalized features set and target set into X\_train, X\_test, Y\_train, and Y\_test.

1. Modeling

By using sklearn, we can easily create regression model to fit the training data. When creating the regression model instance, all the optional arguments are set to default to keep simplicity. All three algorithms will expect two arguments to pass into the function, one being X\_train and the other being Y\_train. After fitting the model, we can predict the data in X\_test, then compare the predicted results with the actual target values, Y\_test.

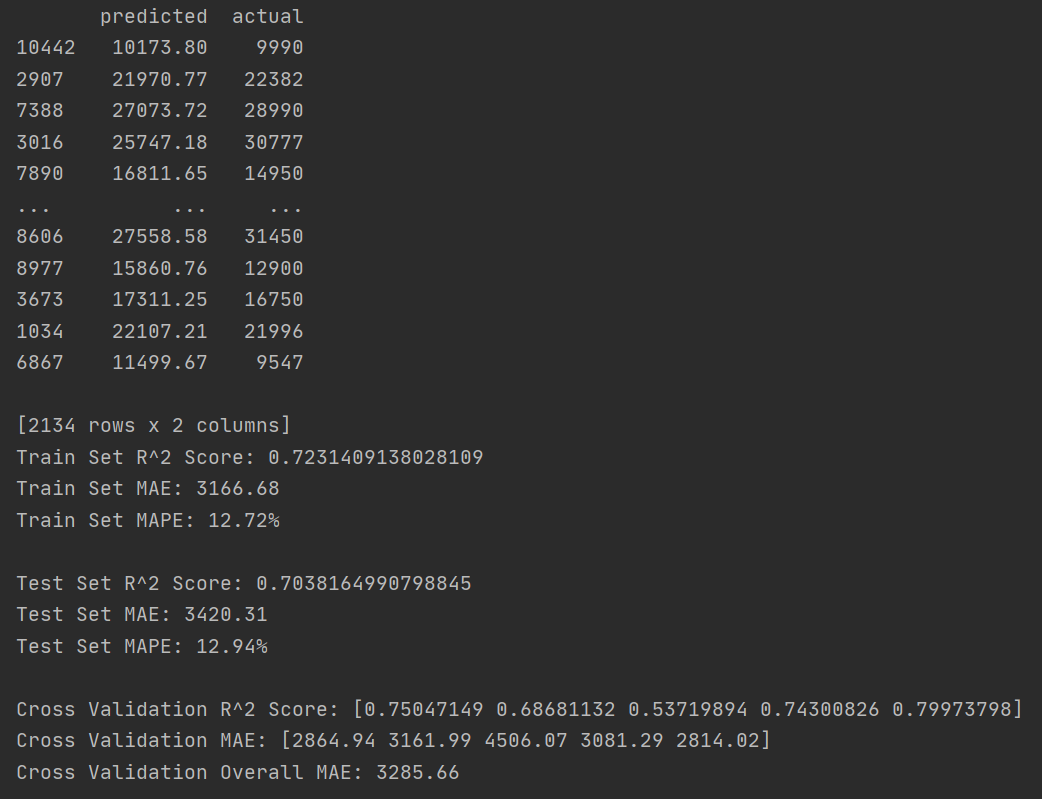
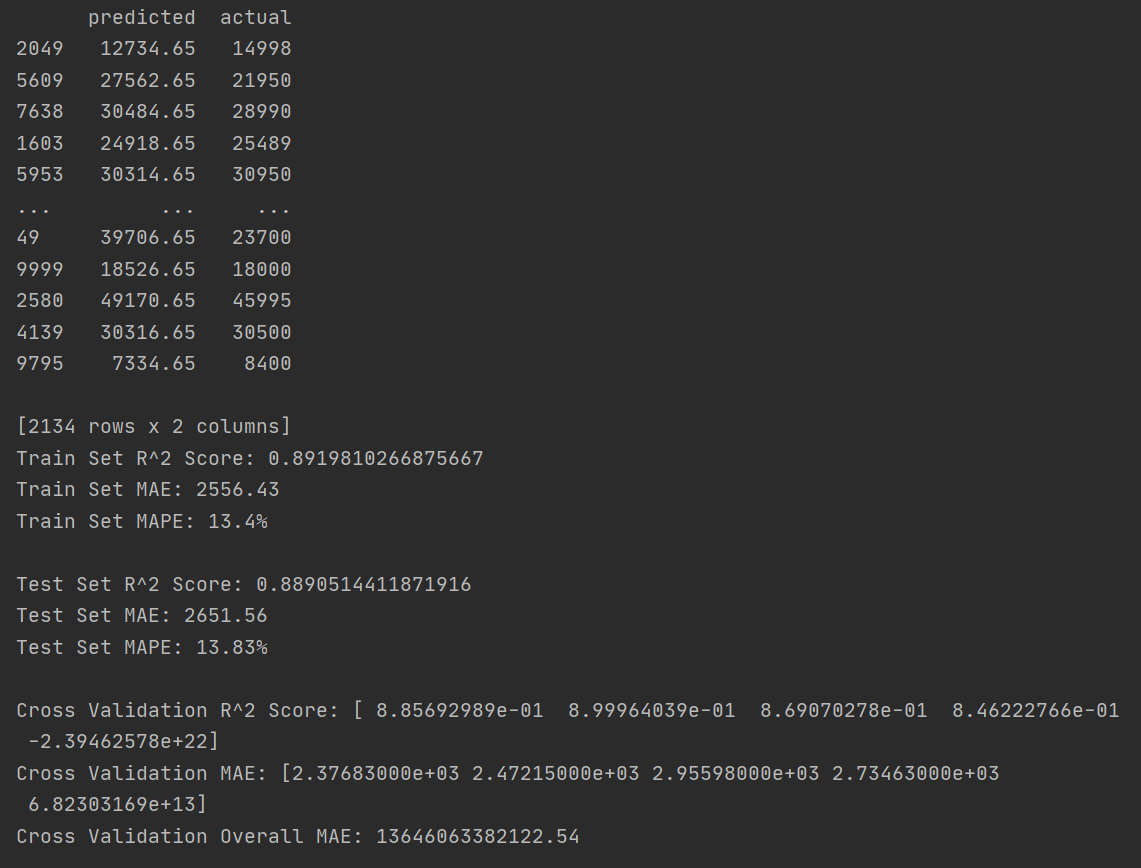
1. Accuracy & Evaluation

Different metrics are required to reveal how well the models fit the data. Firstly, we will compute the R^2 score, which is the coefficient of determination, as an indication of goodness of fit. It represents the proportion of variance in the dependent variable (Y) that has been explained by the independent variables (X’s) in the model. Therefore, it is also a measure of how well the model can perform on unseen data.

Then, I used the mean absolute error to measure the average discrepancy between predicted values and the actual values. To change the perspective, I also used the mean absolute percentage error to show the error as percentage values to the actual target. Noticeably, both the training error and test error are reported. Generally, it is anticipated that the test error will be larger than the training error.

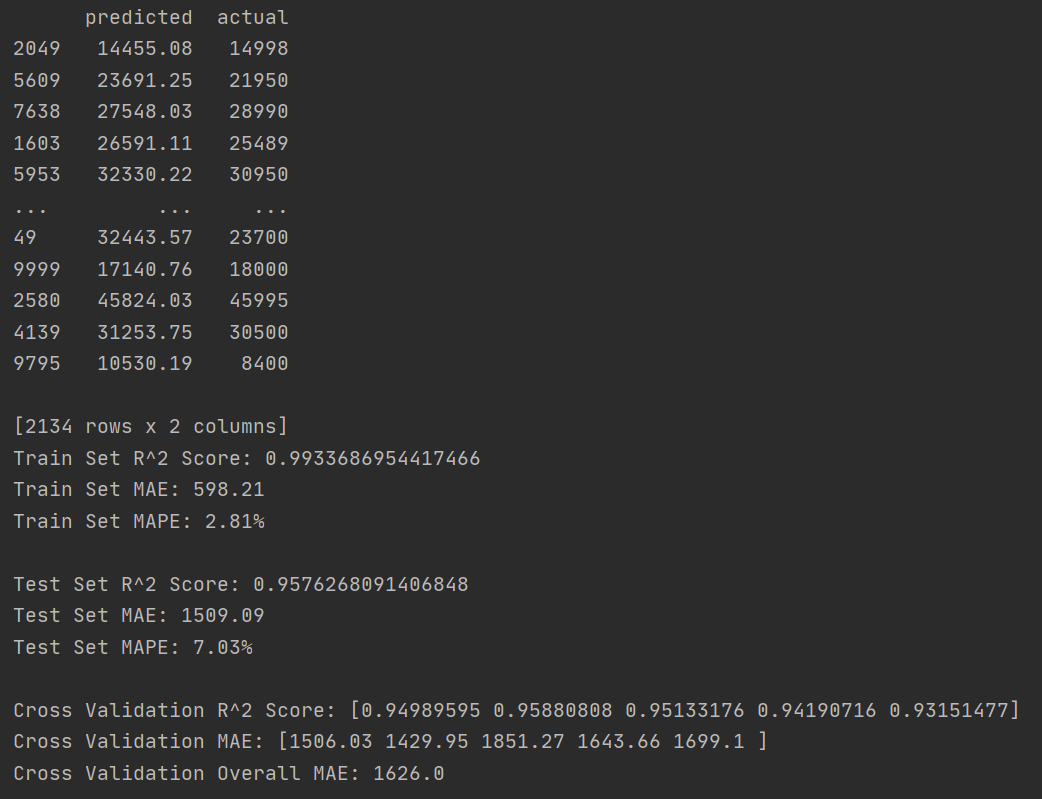
5-fold cross validation is also used to evaluate how well the model generalize on new data. The training set is split into 5 subsets and in each cross-validation iteration, one non-repeatable subset is selected as validation and the other 4 subsets is used for training. The validation error in each iteration is stored to allow computation of overall error for the 5 folds by simply taking average value.

**Comparison of Results**



Multivariate Linear Regression Support Vector Regression

Random Forest Regression



We can see random forest regression has the highest R^2 score in the test set, also in 5-fold cross validation. Correspondingly, it has the lowest mean absolute error (MAE) and mean absolute percentage error (MAPE) of 7.03%, which indicates that the average error is of 7.03% to the actual values. Random forest regression also has the lowest overall MAE in cross validation.

Support vector regression has a slightly better MAPE than multivariate linear regression. However, it has a much lower R^2 score which means the support vector regression may not generalize better than multivariate linear regression in new data.

Multivariate linear regression has a MAPE of 13.83%, which could be acceptable depends on the tolerance of customers. Generally, it has a 0.88 R^2 score which indicates that it is qualified to be used in future samples. It is worth noticing that in the 5th fold cross validation, the model has a negative R^2 score. This is because the model can be arbitrarily worse as explained in the sklearn documentation. Due to this, the MAE of the 5th fold is incredibly large which will also pull up the average error in cross validation significantly.

As anticipated, the MAE and MAPE of the test set are larger than those of the training set in all three algorithms. And the R^2 scores of test set are lower than the training set because the model is not fitted to the test data, it may not perform as good as it was in the training stage.