CSE445

Project:

Price Prediction of Used Cars Using Machine Learning

Submitted by:

Name: Taufiqul Alam
ID: 1921606042
Email: taufiqul.alam@northsouth.edu

Name: Tamjid Hosain
ID: 2012952642
Email: tamjid.hosain@northsouth.edu

Name: Summiya Sunjida Kashpia
ID: 2012712042
Email: summiya.kashpia@northsouth.edu

Price Prediction of Used Cars Using Machine Learning

Taufiqul Alam
Department of ECE
North South University
Dhaka, Bangladesh

Md Tamjid Hosain
Department of ECE
North South University
Dhaka, Bangladesh

Summiya Sunjida Kashpia
Department of ECE
North South University
Dhaka, Bangladesh

***Abstract*— In our used car price prediction project, we are going to analyze a dataset of used car sales. Then, we will train a machine learning model using regression techniques to predict the sale price of a given used car based on its characteristics, such as brand name, model, years, mileage, etc.**

I. INTRODUCTION

It is quite apparent that for most of the population in the world, an automobile is vital in our daily lives. But in recent years, prices of brand-new cars have skyrocketed due to the economic crisis in the whole world. Due to this many people have deviated towards purchasing used cars. Unlike brand new cars which have a given price tag, it is very difficult to judge the fair value of a used car as it depends on many criteria like brand name, model, year, mileage, transmission, condition, etc. Customers who are not well informed about this necessary information are often taken advantage of and end up purchasing a vehicle that is not worth their price. To make the buyers and sellers make informed decisions regarding the fair value of a used car, we aim to make a system that will predict accurate prices of used cars based on different attributes of the car. The model will be trained on historical data of used car sales to identify patterns and relationships that can help it make accurate predictions. The ultimate goal of this project is to create a useful tool for buyers, and sellers that will provide valuable insights and improve the efficiency of the used car market.

II. LITERATURE REVIEW

III. METHODOLOGY

In our project, we are going to use Linear Regression and Random Forest Regression algorithms.

IV. DATASET

The data set that we are going to use is taken from Kaggle.
Link-
https://www.kaggle.com/datasets/nehalbirla/vehicle-dataset-from-cardekho?select=car+data.csv

V. RESULT AND ANALYSIS

In our car price prediction model, we aimed to predict the selling price of the cars. We used regression algorithms on the given data set above to achieve this. The dataset was downloaded from Kaggle, and we uploaded it on our google drive and mounted the drive in our Google Collaboratory file to access the dataset. In the dataset, there were initially 2059 rows but some of the rows had null values. There were 185 rows with null values. So those rows were removed from the dataset, and we ended up with 1874 rows. However, the dataset was found to have some sampling bias because the population of some of the car brands was very high compared to others. For example, the number of Maruti cars was 398 whereas the number of Rolls-Royce was only 3. This happened because luxury cars are bought in a very low number compared to regular cars. To address this issue stratified sampling was used during the train-test split.

The features of the cars that were given in our dataset were: Make, Model, Year, Kilometer Driven, Fuel type, Transmission, Location, Color, Number of Owners, Seller Type, Engine Capacity, Max Power, Max Torque, Length, Width, Height, Drivetrain, Seating Capacity, Fuel Tank Capacity, and Price.

However, some of these features were irrelevant and we removed these features using Recursive Feature Elimination. So, the features of the cars that were taken under consideration during training of our model were: Make, Model, Year, Kilometer Driven, Fuel type, Transmission, Number of Owners, Seller Type, Engine Capacity, Drivetrain, Seating Capacity, Fuel Tank Capacity, and Price. These were the relevant attributes of the car that were utilized to predict the selling price of cars.

Below is an example of our dataset which shows 10 rows of cars and all the relevant features as discussed.

Make	Model	Price	Year	Kilometer	Fuel Type	Transmiss	Owner	Seller Typ	Engine	Drivetrain	Seating Ca	Fuel Tank
Honda	Amaze 1.2	505000	2017	87150	Petrol	Manual	First	Corporate	1198 cc	FWD	5	35
Maruti Su	Swift DZir	450000	2014	75000	Diesel	Manual	Second	Individual	1248 cc	FWD	5	42
Hyundai	i10 Magna	220000	2011	67000	Petrol	Manual	First	Individual	1197 cc	FWD	5	35
Toyota	Glanza G	799000	2019	37500	Petrol	Manual	First	Individual	1197 cc	FWD	5	37
Toyota	Innova 2.4	1950000	2018	69000	Diesel	Manual	First	Individual	2393 cc	RWD	7	55
Maruti Su	Ciaz ZXi	675000	2017	73315	Petrol	Manual	First	Individual	1373 cc	FWD	5	43
Mercedes	CLA 200 P	1898999	2015	47000	Petrol	Automatic	Second	Individual	1991 cc	FWD	5	
BMW	X1 xDrive2	2650000	2017	75000	Diesel	Automatic	Second	Individual	1995 cc	AWD	5	51
Skoda	Octavia 1.	1390000	2017	56000	Petrol	Automatic	First	Individual	1798 cc	FWD	5	50
Nissan	Terrano XI	575000	2015	85000	Diesel	Manual	First	Individual	1461 cc	FWD	5	50

The pricing values for the cars listed in the dataset were excessively high (from thousands to crores). Due to this large range of target values, a scaling method called MinMaxScaler

was used to bring the values to a comparable range which is between 0 and 1.

All the numerical values in the engine attribute ended with the string ‘cc’. To treat it as numerical data, the string ‘cc’ had to be removed and after removing it the values were converted to integers. So, in the end, the values in the engine attribute were all numerical data.
The categorical variables in the dataset were Make, Fuel Type, Transmission, Owner, Seller Type, Drivetrain, Seating Capacity, and Fuel Tank Capacity. All these categorical were encoded into a binary vector representation using one-hot encoding. All the other variables were numerical and were treated as quantitative data.

To determine if the prediction of the car prices were reliable, we relied on two evaluation metrics. R-squared score (R2) and Root Mean Squared Error (RMSE). R-squared is a statistical measure that shows how much of the target variable's variance can be accounted for by the model's independent variables. The R-squared score ranges from 0 to 1. Higher values suggest a better fit between the model and the data. RMSE, on the other hand, is a measure of the average distance between the predicted and actual values of the target variable. Lower values of RMSE indicate better performance of the regression model.

To apply the algorithms, we first split the dataset by a 90:10 ratio where 90 percent of the data was used for training and the remaining 10 percent for testing. This means 1686 rows were used for training (90% of 1874 rows) and the remaining 188 rows were used for testing (10% of 1874 rows).
For further evaluation, we used a 10-fold cross-validation technique.

Performance Measurement

Linear Regression Model

At first, we split the dataset where 90 percent of the data was used for training and the remaining 10 percent for testing. Applying this model on the training data, we got the following graph of actual price vs predicted price.

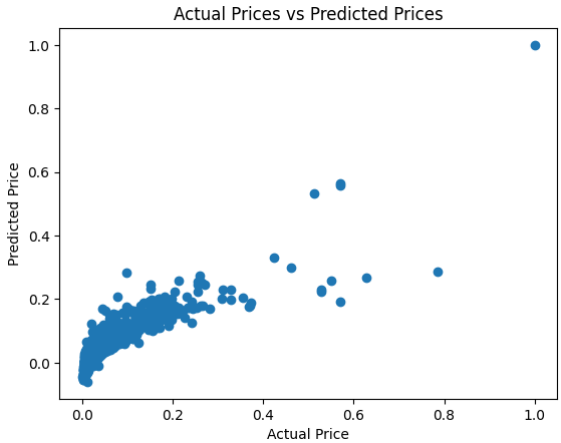

Figure 1: Graph of Predicted Price vs Actual Price on the training dataset with linear regression

For the training data, we got an R-squared score of 0.7607724473925945 which means that the model has a good fit.

Applying the model on the testing data, we got the following actual price vs. predicted price graph.

Figure 2: Graph of Predicted Price vs Actual Price on the testing dataset with linear regression

For the testing data, we got an R-squared score of 0.7459067963801611 which is a very good result. We got an RMSE value of 0.02685601244302204 which is very low. A lower RMSE indicates better performance of the model, as it means that the predicted values are closer to the true values.

For further validation of the linear regression model, we used a 10-fold cross-validation technique. By doing this, we got an R-squared score of 0.67000150076733. This value is smaller than the R-squared score we achieved using a 90% split but still, it is a good result. And we got an RMSE value of 0.03844546835472814. Again, the RMSE value is very low and a lower RMSE indicates better performance of the model, as it means that the predicted values are closer to the true values.

Random Forest Regression Model

At first, we split the dataset where 90 percent of the data was used for training and the remaining 10 percent for testing. Applying this model on the training data, we got the following graph of actual price vs predicted price.

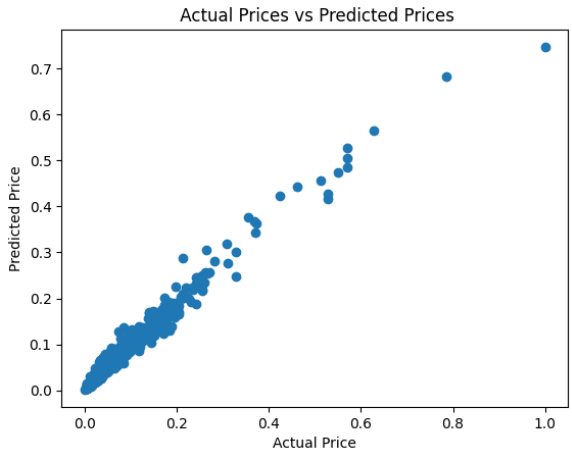

Figure 3: Graph of Predicted Price vs Actual Price on the training dataset with random forest regression

For the training data, we got an R-squared score of 0.9698186374464624 which means that the model has a very good fit.

Applying the model on the testing data, we got the following actual price vs. predicted price graph.

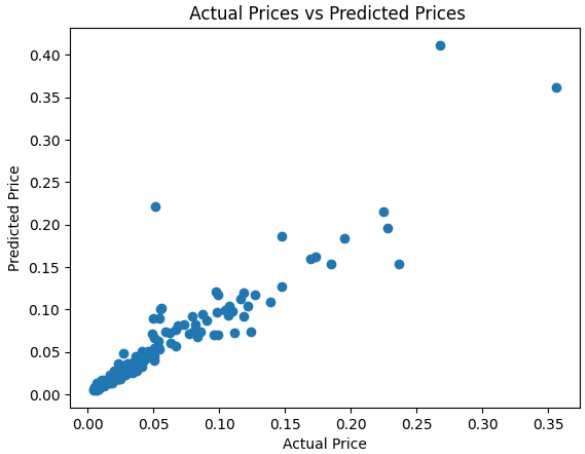

Figure 4: Graph of Predicted Price vs Actual Price on the testing dataset with random forest regression

For the testing data, we got an R-squared score of 0.848614052666839 which is a very good result. We got an RMSE value of 0.02072945033644719 which is very low. A lower RMSE indicates better performance of the model, as it means that the predicted values are closer to the true values.

For further validation of the random forest regression model, we used a 10-fold cross-validation technique. By doing this, we got an R-squared score of 0.8410094066439795. This value is smaller than the R-squared score we achieved using 90% split but still, it is a good result. And we got an RMSE value of 0.026802310926536167. Again, the RMSE value is very low and a lower RMSE indicates better performance of the model, as it means that the predicted values are closer to the true values.

Comparing both models

Based on our evaluation metric we can see that both the models performed with great accuracy in predicting car prices.

But if we compare both the linear regression model and the random forest regression model, we can clearly observe that the random forest model performed much better than the linear regression model. The result of the evaluation metrics of both models is displayed in the tables below.

Model Name	R-squared score	RMSE score
Linear Regression	0.745906	0.026856
Random Forest Regression	0.848614	0.020729

Table 1: Evaluation metric scores during a 90 percent split

Model Name	R-squared score	RMSE score
Linear Regression	0.670001	0.038445
Random Forest Regression	0.841009	0.026802

Table 2: Evaluation metric scores during 10-fold cross-validation

Based on the values of the above table we can observe that the R-squared score in the random forest regression model is higher than that of the linear regression model in both types of evaluation. In addition, the RMSE score in the random forest regression model is less than that of the linear regression model in both types of evaluation. This indicates that the random forest regression model performed with better accuracy than that of the linear regression model.

The minimum value of R-squared score that was expected was at least 0.60. Any regression model that would have scored below this mark would have been discarded. Fortunately, both models which we worked with exceeded our expectations and produced great R-squared scores as well as RMSE scores.

In a research paper [4], it is seen that they did the model for car price prediction using different models, including both random forest regression and linear regression models. They got the following R-squared scores shown in the table below.

Model Name	R-squared score
Linear Regression	0.625564
Random Forest Regression	0.911812

Table 3: R-squared score from research paper [4]

In another research paper [5], the following R-squared scores were obtained for random forest regression and linear regression.

Model Name	R-squared score
Linear Regression	0.764600
Random Forest Regression	0.931100

Table 4: R-squared score from research paper [5]

We can see that the results we achieved in our model are similar to the results they obtained in the research paper. So, the results we obtained are ideal.

In a perfect solution for a car price prediction problem, the goal is to accurately predict car prices with minimal error, consistently providing predictions that closely match the actual prices. It would perform well across different car brands, models, and features, without being overly sensitive or biased. It would generalize effectively to new, unseen car instances and would identify and explain the most significant features and their impact on the pricing of cars.

VI. CONCLUSION

VII. REFERENCES

[1] https://www.webology.org/data-cms/articles/20220523121909pmwebology%2018%20(6)%20-%20443%20pdf.pdf

[2] https://scikit-learn.org/stable/index.html

[3] https://www.kaggle.com/datasets/nehalbirla/vehicle-dataset-from-cardekho?select=car+data.csv

[4] https://www.ijraset.com/research-paper/car-price-prediction-using-machine-learning-algorithms

[5] https://www.irjmets.com/uploadedfiles/paper/volume3/issue_3_march_2021/6681/1628083284.pdf