Used Car Price Prediction Using Data Mining and Machine Learning Techniques

Srija Venkata Sai Ravali Kothapalli
MSc in Computing
National College of Ireland
Dublin, Ireland
x21227454@student.ncirl.ie

***Abstract*— The used car market is expanding very fast all over the world. Price prediction for used cars becomes an essential topic and exploited by many researchers in recent years. where several factors influence the pricing of vehicles. In this research, we explore the use of data mining and machine learning techniques on predicting the prices of used cars in three different regions namely Europe, the US, and Germany, using Two popular algorithms, Random Forest and K-Nearest Neighbors (KNN) for classification, were implemented using two widely used libraries in the machine learning community they are Scikit-learn (SKlearn) and LightGBM, respectively. The datasets used in this study contains information on various attributes of used cars, such as make, model, year, mileage, and other technical specifications and various metrics such as accuracy, precision, recall, and F1-score were used to evaluate the models' performance. Our findings suggest that Random forest outperforms KNN in terms of accuracy and consistency across all three datasets when using SKLearn Library and KNN outperforms Random Forest when used LightGBM library. Our study contributes to the understanding of the used car market and provides insights into the potential use of data mining and machine learning techniques for predicting used car prices.**

***Keywords*—Used cars, Dataset, Data, Random Forest, KNN, Implementation, SKLearn, LightGBM, F1-score, Accuracy.**

I. INTRODUCTION

The price of a car drops right from the moment it has bought and the depreciation continues with each passing year. The used car market is very huge as many people were interested in buying the used ones as they were much cheaper compared to the new ones. Considering the demand for people who are willing to buy, used cars will create a good business for both buyers and sellers and customers who are willing to buy used cars struggle to find the car suitable for them within their budget. The used car does not have any particular price as it varies depending on several factors like make, model, and damages that happened to the car. Hence, the best model for price evaluation is required, which helps customers to choose the car.

The market for used cars is increasing even though there were many new cars on the market. Purchasing a new vehicle may not be financially feasible for many customers due to the amount they need to spend on the car. The insurance for the car and fuel prices are rising which were leading customers to buy used cars where they can save money on buying a car. After using a car for some time later, they will sell the car again for a price that the deprecation value for the car is not much that eventually leading the seller to save some amount. Before buying a new car the user to new to check many items like whether the entire interior is fine and all the things inside the car are functional like lights, air conditioner, and sensors as it will lead to extra cost if it has some problems. Therefore, the most important thing is the customer needs to invest in a car that is worthy of the price range and which will not lead to any trouble later on.

As the demand for used cars is increasing as used cars are more affordable and we can get them for much lower prices compared to a new one. The used car market is a vital part of the automotive industry, accounting for a significant share of global car sales. However, pricing used cars is a challenging task, as several factors influence their value, including age, mileage, condition, model, make, location, and demand. Traditional methods of pricing used cars rely on expert opinions, market trends, and historical data. In recent years, data mining and machine learning techniques have emerged as promising tools for predicting used car prices, leveraging the vast amounts of data available on the internet.

The prices may increase or decrease based on the market trends. We can buy a used car and use it for some days and if the market trend is increasing then we can resell the car for profit and buy another car. When we are buying or selling a used car, we should know the market trend and the assumed price for the particular car. As some sellers sell their cars for higher prices for their profit than the price they should be sold. Therefore, a Predicting model based on a used car's price is essential for both seller and buyer to sell or buy the used cars at a reasonable price. In this research, we would like to test and train the dataset using different algorithms like Random Forest, K-Nearest Neighbour (KNN), and obtain certain accuracy and compare with each other and then combine two or more algorithms to obtain a better result.

II. Literature Review

K. Samruddhi, Dr. R. Ashok Kumar [1] used K- The nearest Neighbor-based model for predicting the used car prices. At first, they gathered data from Kaggle then they filtered and cleaned the data then they used the data to train using the K- Nearest Neighbor algorithm. Then they applied different K values for the K- Nearest Neighbor model, as it is a non-parametric method it can be used for both regressing and classification. They also tried different ratios for training and testing to obtain better accuracy. They used Root-Mean Squared Error (RMSE) rate Means Absolute Error (MAE) as metrics for evaluation. In this, they did cross-validation that has used to inspect the overfitting of the model. They did this with 5 folds and 10 folds that gave similar results. Finally, they reached an accuracy of 85% and declared it the best-fitted model.

Abhishek Pandey, Vanshika Rastogi, and Sanika Singh [2] used supervised learning to predict the price of the car by using previous selling car data. They used Random Forest and Extra tree regression using the sci-kit learn library to predict the price of the car where both of them are very accurate in both small and large datasets. They used an ensemble model to get better and more accurate results that give us a clear outline to propose a fusion model for our research.

Anu Yadav and Ela Kumar [3] have done their work on object detection and used car price prediction using machine learning. Firstly, they used object detection and fusion of linear regression and random forest to obtain accuracy by evaluating the model with MAE, MSE, and RMSE. Then they concluded that they achieved an acceptable performance increase rather than using a single technique. Using more than one technique increases the chances of getting better accuracy.

Fahad Rahman Amik [4] and his team are willing to find a good prediction model that helps customers to find used cars at reasonable prices. They wanted to estimate the price of the used car and applied various machine learning algorithms like lasso regression, linear regression, decision tree random forest, and extreme gradient boosting. Among all the algorithms, extreme gradient boosting performs better with higher accuracy in predicting the price of the used cars.

Nabarun Pal and his team [5] used a supervised learning method that is Random Forest to predict the prices of used cars. At first, they pre-processed their data and then they did Exploratory data analysis to find the relations between the data then They created 500 decision trees to train the data and the data split is done in a ratio of 70,20,10 for training, testing, and cross-validation respectively. Then they evaluated the model using R Square and obtained good accuracy.

Bhatia et al. (2020) [6] used the scikit-learn toolkit to evaluate the performance of KNN with various classification techniques. The study shown that KNN worked well on datasets with a small number of features, but that as the number of features rose, so did its performance. In order to get optimal performance, The study also found that tuning the value of k was critical in obtaining good performance.

In the research utilizing the scikit-learn toolkit, Shrivastava et al. (2020) [7] evaluated the effectiveness of Random Forest with different classification methods. Random Forest was proven to be effective across a range of datasets and to be especially helpful for datasets with a large number of characteristics. The study also discovered that the algorithm's performance was enhanced by adding more trees to the Random Forest.

Ke et al. (2017) [8] used a range of datasets to evaluate the performance of LightGBM with various gradient boosting frameworks. According to the study, LightGBM fared better than other frameworks in terms of speed and accuracy, especially when working with huge datasets that have a lot of characteristics. The study also discovered that LightGBM could manage missing values and imbalanced datasets and was resistant to overfitting.

The Wisconsin Breast Cancer dataset was used in this work by the authors [9] to assess the performance of KNN, Random Forest, and LightGBM in predicting breast cancer. According to the study, LightGBM performed better than KNN and Random Forest in terms of F1-score, accuracy, precision, recall, and KNN. Additionally, the study discovered that LightGBM required less time to train than Random Forest.

In this work, the authors [10] used the Chest X-Ray dataset to examine the classification accuracy of KNN, Random Forest, and LightGBM for pneumonia. According to the study, LightGBM performed better in terms of accuracy, precision, and recall than KNN and Random Forest. In addition, the study discovered that LightGBM required less time to train than KNN and Random Forest. Both of these investigations demonstrate that, in terms of accuracy and training time, LightGBM consistently beats KNN and Random Forest. However, the particular job and dataset at hand ultimately determine the method to use.

III. Methodology

A. Dataset

Three different locations (Europe, US and Germany) Used Cars Datasets were taken from the Kaggle website. Europe dataset size is 92 MB and It consists of 3552912 rows and 16 columns. US dataset size of 275 MB having 426880 rows and 26 columns of data, Germany dataset size is 19 MB with 371528 rows and 21 columns of data, this would be perfect for our research.

B. Exploratory Data Analysis

We have explored the datasets to understand the dataset better. The first step we have taken in this Europe dataset is finding unwanted columns, Null values and missing values which are not useful for our research. Therefore, we should drop those columns and also Mean/Mode/Median Imputation is followed to remove null values to avoid further complex problems. We should find a way to fix this or else the prediction we were going to make will be inaccurate and become misleading. Then we have gone through each column and checked for outliers for three datasets.

For the first dataset, Europe Used Cars Dataset. we have found that many outliers which are meaningless. For example, the mileage of the car having less than 300 km and more than 300000 km. In addition, we found outliers in the manufacture year column having the year greater than 2018 and less than 1980 has to be removed and the engine displacement less than 20 and more than 4000 and Engine power having more than 300 and less than 20 should be removed. Also, price more than 50000 and less than 900 has to be removed. We have used histogram to identify all these outliers to understand and remove the outliers successfully.

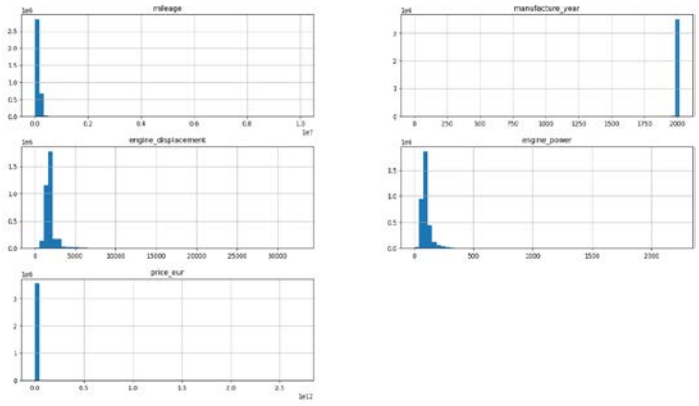

Fig 1. Before Removing Outliers for Europe Dataset

Similarly, The Second Dataset of US location has outliers such as Car Price more than 40000 and less than 300; Year greater than 2022 and less than 1980 and Also, Odometer less than 1000km and greater than 200000 km found to be outliers and these outliers has to be removed.

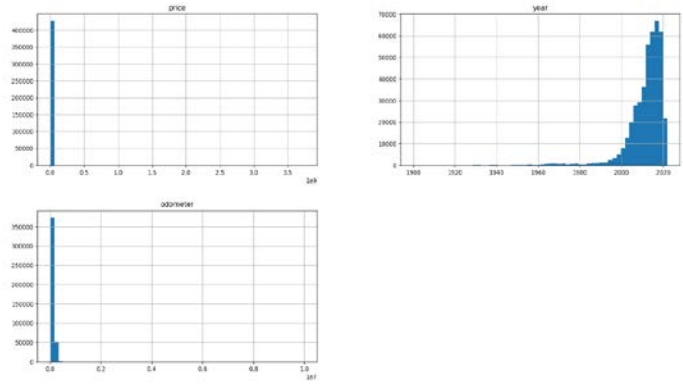

Fig 2. Before Removing Outliers for US Dataset

The third, Germany Dataset has some words in German language were changed to English. Similarly, some useless columns having null values and missing values were found. Also, some outliers were found with price greater than 100000 and less than 100; year of registration greater than 2017 and less than 1900 and Power PS less than 20 and greater

than 1500 has to be dropped. Whereas in Month of registration, we have found that some rows of data with 0 then checked for mode of this parameter.

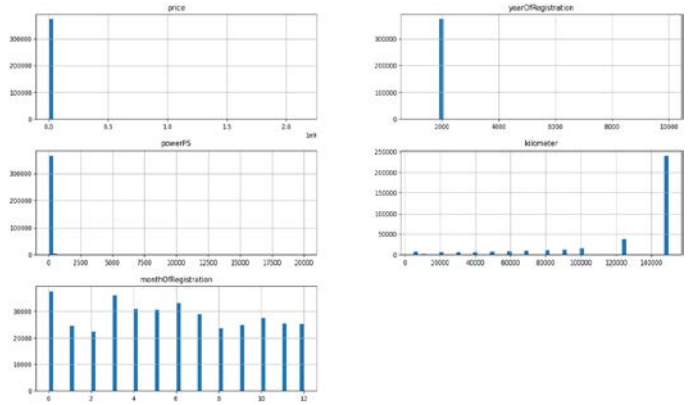

Fig 3. Before Removing Outliers for Germany Dataset

Thus, we have seen so many useful insights in the data exploration that we need to fix in the next step which is preprocessing of the data.

C. PreProcessing

In Preprocessing, the first thing to do is to remove unwanted columns. In our datasets, there are many unwanted columns.

Europe Dataset has date_created, last_seen, stk_year, color_slug columns which are useless and no need of them in the prediction and then we have checked for null values and missing values and removed accordingly. we have decided to group columns into three groups such as categorical columns, continuous columns and discrete columns. In these groups, in categorical columns such as maker, model, body_type, transmission, door_count, seat_count, fuel_type, we need to fill the missing values with mode as it is an appropriate way to do it. In continuous columns, we have the mileage, manufacture_year, engine_displacement, engine_power that has null values. We have filled the missing values with the median of the respective column. we don't have any null values in discrete columns. Therefore, there is no need to deal with it. Finally, we have filled all the null values in the entire dataset.

Then coming to outliers, we have removed them by checking with histogram plots they are the mileage of the car having less than 300 km and more than 300000 km is not useful, the manufacture year column having the year greater than 2018 and less than 1980 and the engine displacement less than 20 and more than 4000 and Engine power having more than 300 and less than 20 has been removed. Also, price more than 50000 and less than 900 has been removed.

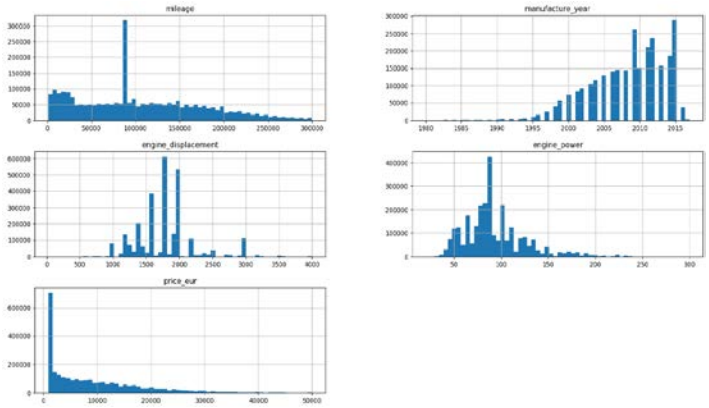

Fig 4. After Removing Outliers for Europe Dataset

Similarly, in second dataset from US, we have dropped unwanted columns such as id, url, region_url, VIN, size, image_url, description, lat, long and posting_date, county then checked for Null, Missing values and removed also filled accordingly by checking datatypes of the columns by running the created python data frame to read the data. We have checked for categorical, continuous, and discrete columns as it is important in data cleaning so that we can select the appropriate data cleaning and analysis techniques to ensure that the data is accurately represented and useful for our purposes. So, we have grouped the columns into three groups categorical columns, continuous columns, and discrete columns and found that manufacturer, model, condition, cylinders, fuel, title_status, transmission, drive, type, paint_colour were categorical columns having null values and we have to fill the missing values with mode. In continuous columns, we have the year and odometer that has null values and therefore these missing values is filled with the median of the respective column. we don't have any null values in discrete columns in this dataset. Then the outliers that has been found in this dataset such as Car Price more than 40000 and less than 300; Year greater than 2022 and less than 1980. Also, Odometer less than 1000km and greater than 200000 km were found be useless and these outliers has been removed.

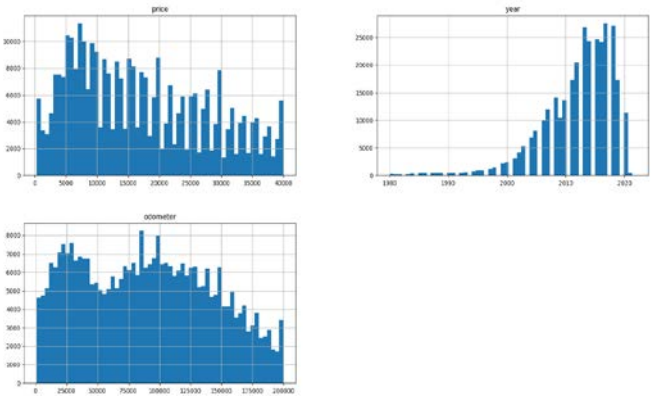

Fig 5. After Removing Outliers for US Dataset

In third dataset related to Germany after reading the data, we found the German words. we have translated the German words into English words. For example, privat to private, gewerblich to dealer, angebot to services, gesuch to wanted, kleinwagen to smallcar,cabrio to convertible, andere to other, kombi to stationwagon,manuell to manual, automatik to automatic, benzin to petrol, andere to other, elektro to electric, ja to yes, nein to no. So, now things get clearer as our dataset has been translated to English. Later we have checked for unwanted columns and found that index, dateCrawled, dateCreated, nrOfPictures, postalCode, lastseen and seller were dropped. After checking for datatypes, we found that OfferType column is also useless for both seller and buyer and removed respectively. Then later we checked for null values and missing values by verifying with categorical, continuous, and discrete columns and then replaced with median and mode imputation method such as the categorical columns such as vehicletype, gearbox, model, fueltype, and notrepaireddamage have null values. we have filled the missing and null values with mode as it is an appropriate way to do it. In continuous columns, we have the price, year of registration, and powerPS that have null values. We have filled the missing values with the median of the respective column. we don't have any null values in discrete columns. Therefore, there is no need to deal with it. Finally, we have filled all the null values in the entire dataset. hence removed all the null values or the missing values that make our data stronger.

In this dataset, price greater than 100000 and less than 100; year of registration greater than 2017 and less than 1900 and Power PS less than 20 and greater than 1500 has been found as outliers and were dropped accordingly. In the month of registration column, we have some incorrect values. As we have 12 months, the value should lie between 1 and 12 but some of the values were not in the range so the rows that are not in the range were replaced with the mode of the column that is 3. To make the column useful. Finally, preprocessing of the dataset is completed. Now it is ready to use for applying algorithms. Therefore, all the outliers and the irrelevant columns for the three datasets were removed and the three cleaned datasets were exported to excel files successfully.

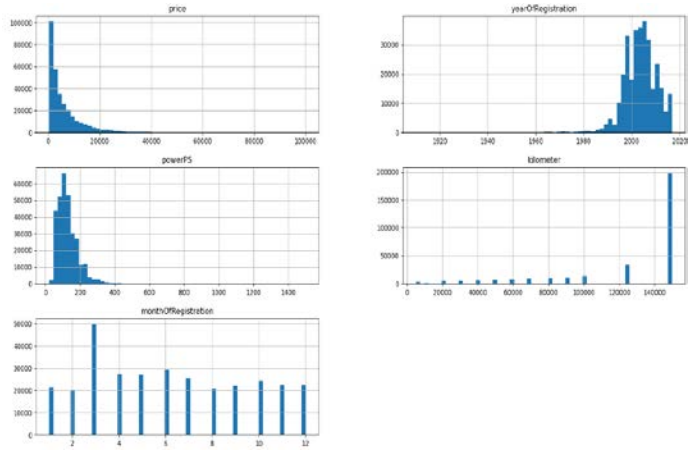

Fig 6. After Removing Outliers for Germany Dataset

D. Creating a new Class

A new class called price class is created based on the price column for three datasets, where the price is lower than 2500 is classified as low and when the price is higher than 10000 is classified as high and the price between 2500 and 10000 is classified as mid in the first Dataset.

Similarly, In Second dataset we have created price class the price lower than 10000 is classified as low and higher than 25000 is classified as high and the price between 10000 and 25000 is classified as mid

In Third Dataset the price lower than 2000 is classified as low and higher than 8000 is classified as high and the price between 2000 and 8000 is classified as mid.

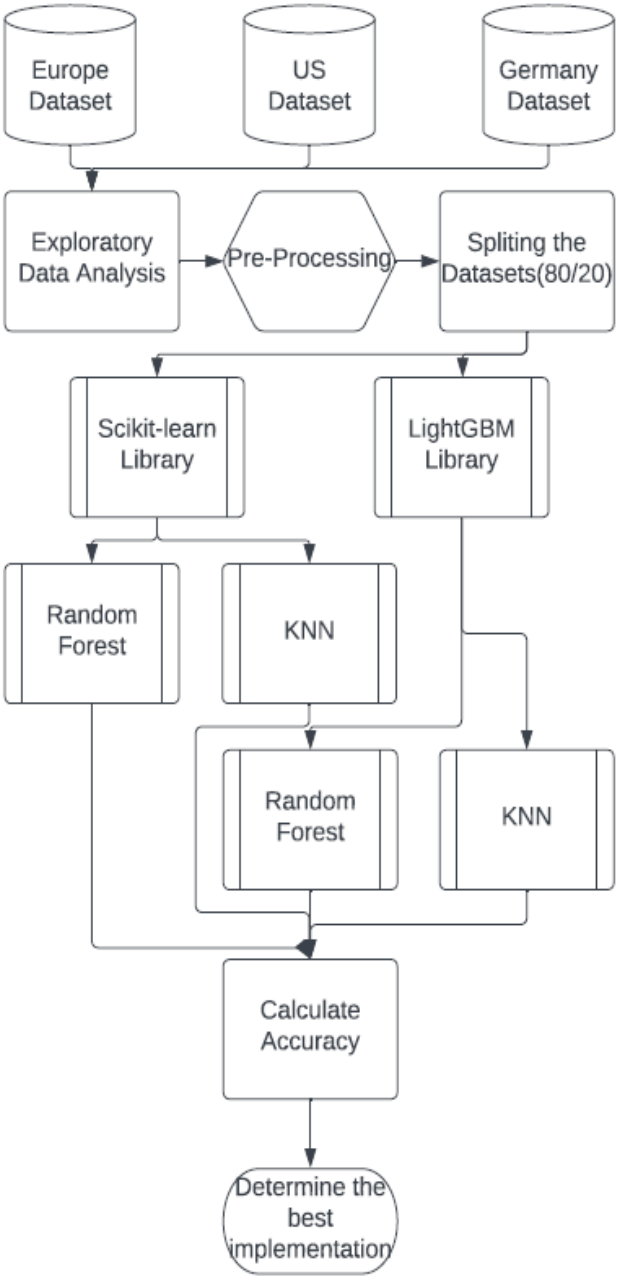

Fig.7 Architecture

E. Encoding the dataset

In Europe dataset, we have encoded feature variables using one hot encoding and target variable using label encoder and get the dataset ready for training.

Same procedure is followed in US dataset as well, we have used one hot encoding for feature variables and label encoder was used for target variables.

Finally, In Germany Dataset same encoding steps were followed for feature variables and target variables and dataset is made ready for testing and training.

F. Splitting the dataset

From the Europe dataset, we took a sample of 150000 at random and from the US dataset we took the sample of 75000 at random. Similarly, for Germany dataset we took sample of 150000 at random. As we have very huge datasets with more than 300000 rows which results in memory errors while processing them.

Therefore, we took different samples sizes for each dataset as required to avoid memory error. Then we had split our sample data for each dataset into training and testing data with the ratios of 80 and 20 respectively.

G. Applying Algorithms

Now, we have applied Random Forest, K-Nearest Neighbor (KNN), classification algorithms by using Scikit-learn and LightGBM libraries in python to understand how algorithms perform in different libraries for our datasets and obtain the best implementation. Therefore 12 accuracies, for every individual algorithm such as Random Forest and KNN having two implementations each using Scikit-Learn and LightGBM libraries. Therefore, each dataset has 4 implementations. overall, 12 implementations for three datasets

H. Evaluate the results using the Confusion matrix

In classification, the evaluation is done using a confusion matrix. It consists of True Positives, True Negatives, False Positives, and False Negatives. Based on that we can calculate the Precision, Recall, F1-Score, Support, and Accuracy for every model. Below is the figure that explains all the values required for confusion matrix.

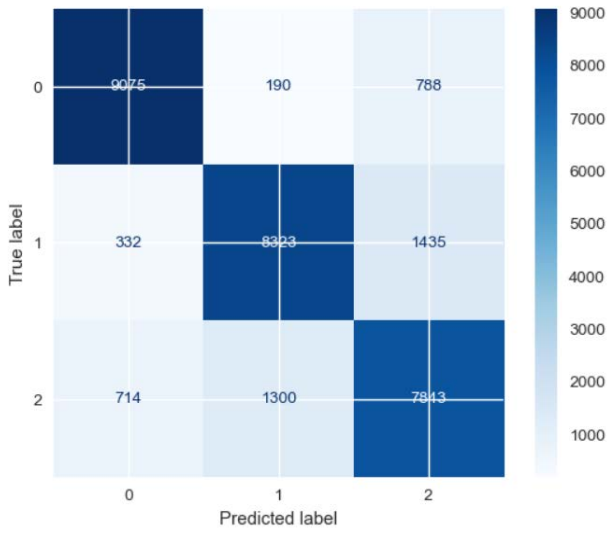

Fig 8. Confusion Matrix

IV. Evaluation and Results

We have obtained 12 accuracies, for every individual algorithm such as Random Forest and KNN having two implementations each using Scikit-Learn and LightGBM libraries. Therefore, each dataset has 4 implementations. overall, 12 implementations for three datasets which can be seen in table 1.

Datasets	Libraries	Accuracy of Algorithms	
		Random Forest	KNN
Europe Dataset	SK Learn	0.84	0.64
	LightGBM	0.79	0.84
US Dataset	SK Learn	0.86	0.65
	LightGBM	0.74	0.82
Germany Dataset	SK Learn	0.85	0.82
	LightGBM	0.81	0.86

Table.1 Accuracy of the Models

Specifically, for the first dataset Europe using the Random Forest algorithm, the implementation with scikit-learn library gave accuracy around 0.84. while the LightGBM implementation achieved about 0.79. On the other hand, for the KNN algorithm with LightGBM implementation achieved accuracy around 0.84, while using the scikit-learn, we have achieved around 0.64.

For the second dataset US Random forest using scikit-learn library implementation achieved more accuracy around 0.86 compared to Random Forest using LightGBM library implementation with 0.74. Then KNN algorithm using scikit-learn achieved accuracy around 0.65 and KNN using LightGBM with 0.82.

In the Third dataset Germany we have accuracies in same range between (0.80-0.86) for both algorithms with four implementations such as for Random forest using scikit-learn library achieved accuracy around 0.85, while using LightGBM library accuracy is 0.81. Similarly, KNN with scikit-learn library achieved accuracy almost near to Random forest implementations around 0.82 and KNN with Light GBM implementation gave high accuracy around 0.86 compared to all implementations.

The algorithms that we used in our experiments were performing well with more than 0.84 accuracies except for some combinations of algorithm and library such as Random Forest with LightGBM having accuracies for first two datasets Europe and US around 0.79 and 0.74. Whereas KNN with scikit-learn having accuracies of 0.64 and 0.66, which were lower than 0.80.

Based on our research, we found that the Random Forest algorithm implemented with scikit-learn library has provided the best results in used car price prediction models compared to Random Forest with LightGBM library. KNN algorithm implemented with LightGBM library has outperformed the implementation of KNN using scikit-learn library.

Random Forest is an ensemble learning approach that mixes the outputs of several decision trees to produce predictions. It is well-suited for dealing with high-dimensional data and has several features.

KNN, on the other hand, is a non-parametric approach that generates predictions based on the feature space's nearest k-neighbors. It is useful for handling non-linear relationships between features.

Scikit-learn is a popular Python machine learning package that offers a variety of methods for classification,

regression, and clustering issues. It also contains tools for preparing data, selecting features, and evaluating models.

LightGBM is a novel gradient boosting library for effective training of large-scale datasets with high-dimensional features. It employs a tree-based method similar to Random Forest, but with more advanced optimization techniques for better speed.

V. Conclusion

In this research, we have used three datasets, we have applied random Forest and KNN using scikit-Learn and LightGBM libraries in python. Random Forest has performed best when used with Scikit-learn Library compared to KNN in scikit-learn implementation. Whereas KNN has performed best when used with LightGBM implementation compared to Random Forest in LightGBM implementation.

In Europe dataset, Random Forest with scikit-Learn implementation and KNN in LightGBM implementation achieved same and best accuracies that is 0.84. In US Dataset, Random Forest with scikit-Learn implementation achieved highest accuracy i.e., 0.86 compared to other implementations. Where as in Germany Dataset, KNN in LightGBM implementation achieved highest accuracy that is 0.86 compared to other implementations.

In the future, we were going to experiment on more algorithms with various libraries that may lead us in achieving greater accuracy and best suitable models for Used cars price prediction.

REFERENCES

[1] K. Samruddhi, Dr. R.Ashok Kumar, “Used Car Price Prediction using K-Nearest Neighbor Based Model”, International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE) Volume 4, Issue 2, DOI: 10.29027/IJIRASE.v4.i2.2020.629-632, August 2020.

[2] Pandey, Abhishek, and Rastogi, Vanshika and Singh, Sanika, “Car’s Selling Price Prediction Using Random Forest Machine Learning Algorithm” (March 1, 2020). 5th International Conference on Next Generation Computing Technologies (Ngct-2019).

[3] [Yadav, A., Kumar, E., & Yadav, P. K. (2021).” Object detection and used car price predicting analysis system (UCPAS) using machine learning technique”. Linguistics and Culture Review, 5(S2), 1131-1147.

[4] Amik, F.R.; Lanard, A.; Ismat, A.; Momen, S. Application of Machine Learning Techniques to Predict the Price of Pre-Owned Cars in Bangladesh. Information 2021, 12, 514.

[5] Nabarun Pal, Priya Arora, Dhanasekar Sundararaman, Puneet Kohli, Sai Sumanth Palakurthy, "How much is my car worth? A methodology for predicting used cars prices using Random Forest”, Future of Information and Communications Conference (FICC),2018.

[6] Bhatia, N., Kumar, V., & Dabas, H. (2020). A Comparative Study of Classification Algorithms Using Scikit-Learn. 2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON), pp. 237-242.

[7] Shrivastava, S., Singh, A., & Verma, A. (2020). Comparative Analysis of Classification Techniques Using Scikit-Learn. 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), pp. 263-267.

[8] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., & Ye, Q. (2017). LightGBM: A Highly Efficient Gradient Boosting Decision Tree. Advances in Neural Information Processing Systems, vol. 30, pp. 3149-3157.

[9] Thapliyal, H., & Rathi, P. (2020). Comparative Analysis of Machine Learning Algorithms for Breast Cancer Prediction. International Journal of Advanced Computer Science and Applications, vol. 11, no. 11, pp. 349-356.

[10] Cholleti, S. R., & Tadi, R. (2020). A Comparative Study of Machine Learning Algorithms for Pneumonia Classification. 2020 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE), pp. 194-197.