*Nanotechnology Perceptions*
*ISSN 1660-6795*
*www.nano-ntp.com*

# Car Price Prediction Using Machine Learning Techniques

**Amneh J Kenana[1], Iyas A Qaddara[1], Khawla M Al-Tarawneh[2], Bashar Mufeed Mohammed Ananbeh[1], Mu'ayyad Khalil Al-Asouli[1]**

*[1]Part time lecturer, Prince Abdullah Bin Ghazi Faculty for Information Technology, Department of Computer Science, Al- Balqa' Applied University, Amman, Jordan*
*[2]Researcher, King Abdullah, School of Information Technology, Department of Computer Science, The University of Jordan, Amman, Jordan*
*Email: amneh.kenana@bau.edu.jo*

In this paper , we suggest a new model to predict the car price by using different machine learning algorithms such as (RANDOM FOREST, Linear-regression, KNN, ADABOOST , SVM , XGboost, Decision tree) and based on 5 datasets , 4 of them from Kaggle and research gate(Ultimate Car Price Prediction Dataset , Car price prediction(used cars) , Used Car Price Prediction Dataset ,and Car Price Prediction) , and generate a new one from Opensooq website contain 8740 instance and 67 features. Several evaluation metrics have been used : Mean Squared Error (MSE) , and R-squared (R2) Score , the results yield that Random forest give the highest R-squared and lowest mean squared error(MSE).

**Keywords:** mean squared error(MSE), Car price prediction, Machine Learning.

## 1. Introduction

The globe is expanding continuously, and the expectations of individuals are also increasing. Among the various expectations, one of them is to purchase an automobile. However, a newcomer may not be aware of the current market value for their desired vehicle, whether it is brand new or pre-owned. In order to make trustworthy and precise predictions, it is necessary to possess specialist knowledge in the sector, since the price of automobiles is influenced by several significant elements. We require a platform that assists newcomers in predicting automobile prices.

In recent years, the automotive industry has undergone a significant transformation, with advancements in technology, design, and manufacturing processes. The increasing complexity of modern cars, coupled with diverse market dynamics, has led to a growing interest in leveraging machine learning techniques for predicting car prices. Accurate and reliable price predictions are crucial for various stakeholders, including manufacturers, dealerships, and

consumers, as they navigate the intricate landscape of buying and selling automobiles.

The traditional approach to pricing cars involves considering factors such as production costs, market demand, and economic conditions. However, the sheer volume and diversity of data available today, including historical sales data, features of the vehicles, and market trends, present an opportunity to enhance pricing models using machine learning algorithms. These algorithms can analyze vast datasets, identify patterns, and extract valuable insights to make more informed and precise predictions.

Machine learning models have demonstrated their efficacy in predicting prices across various industries, and their application to the automotive sector holds promise for optimizing pricing strategies. By incorporating features such as brand reputation, vehicle specifications, fuel efficiency, and even external factors like economic indicators, machine learning models can offer a holistic approach to car price prediction. This approach not only enhances the accuracy of predictions but also allows for dynamic adjustments to account for changing market conditions.

This paper aims to explore the application of machine learning techniques in predicting car prices, delving into the various factors influencing pricing models and the methodologies employed to enhance prediction accuracy. By leveraging the power of data-driven insights, we seek to contribute to the ongoing dialogue within the automotive industry regarding the adoption of innovative technologies for more effective decision-making and improved customer satisfaction. Through this research, we hope to pave the way for the integration of advanced machine learning models into the pricing strategies of the automotive sector, fostering a more dynamic and responsive pricing landscape.

## 2. Literature Review

Car price prediction has been a prominent and intellectually demanding study topic, necessitating substantial expertise and effort from specialists in the field. A significant number of diverse attributes are analyzed to provide dependable and precise prediction. In order to construct a predictive model for determining the cost of pre-owned vehicles in Bosnia and Herzegovina, we utilized three distinct machine learning methodologies: Artificial Neural Network, Support Vector Machine, and Random Forest. However, the aforementioned strategies were used to function collectively as a group. The forecast utilized data obtained from the online portal autopijaca.ba using a PHP-based web scraper. Subsequently, the performances of several algorithms were evaluated in order to identify the one that most effectively matches the given data set. The ultimate forecasting model was included into a Java program. Additionally, the model underwent evaluation utilizing test data, resulting in an accuracy of 87.38% [1].

The user employed a total of five supervised machine learning models and obtained the following root mean squared error values: KNN Regressor: 7771.09, Linear Regression: 6846.23, XG Boost: 3980.77, Random Forest: 3702.34, and Decision Tree Regressor: 5590.43. Among all the models, the Random Forest model achieved the lowest Root Mean Square Error (RMSE) and performed exceptionally well with the greatest R-squared value of 0.93. The primary constraint of this study is the scarcity of historical automobile records. In

the future, if we receive a larger amount of data, we may retrain our models, perhaps resulting in a more precise and reliable model. This study employed several methods to forecast the pricing of pre-owned vehicles. Nevertheless, the dataset used for establishing a robust inference was limited due to the comparatively little number of observations, which was only 92386. Acquiring additional data can lead to more accurate and reliable forecasts. Additionally, there is the possibility of including other variables that have the potential to serve as strong predictors. Here are a few factors that might enhance the model: The factors to consider are the quantity of doors, the color, the duration of mechanical and cosmetic reconditioning, the ratio of used-to-new, and the ratio of appraisal-to-trade [2].

I utilized renowned machine learning techniques from Python packages to create algorithmic paradigms. Initially, we carry out pre-processing and data cleansing on our dataset. After doing our analysis, we discovered that 15% of the tuples included null values. Consequently, we removed such tuples from our dataset. The findings indicated a direct association between price and kilometers traveled, as well as between year of registration and kilometers traveled. Conversely, there was an inverse correlation between price and year of registration. Positive correlation refers to a direct proportionality between two variables, whereas negative correlation refers to an inverse proportionality. A total of 300,000 tuples were utilized for training the model. The year of registration had a somewhat greater level of dominance. The performance of K Nearest Neighbour (KNN) and Classification and Regression Trees (CART) is evaluated on two distinct car models. The root mean square error (RMSE) for K-nearest neighbors (KNN) with a value of k equal to 7 is 5581.96, whereas for the classification and regression tree (CART) algorithm, it is 4961.64 [3].

The suggested technique benefits both buyers and sellers in facilitating the acquisition and sale of vehicles. It enables them to accurately assess the value of their car and make informed decisions that are advantageous for both personal and business purposes. The performance of our suggested model demonstrates the productivity and efficiency of the study. The suggested study utilizes the machine learning algorithm Regression to get superior performance. In this analysis, we employ a Statistical test to determine the value of P and identify the ideal features. Additionally, we utilize linear regression. Initially, we identify the Residuals of the Fitted Equation (RFE) and subsequently employ the statistical test for Variance Inflation Factor (VIF) in the Ordinary Least Squares (OLS) Regression. The prediction findings indicate that the study is both efficient and effective [4].

This research employs four distinct machine learning algorithms to predict the prices of pre-owned vehicles in Mauritius. The average error in linear regression was around Rs51,000, whereas with kNN it was around Rs27,000 for Nissan automobiles and roughly Rs45,000 for Toyota cars. The accuracy of J48 and NaiveBayes fluctuated between 60-70% for various parameter combinations. Decision trees and naïve Bayes models have a primary limitation in that they are not capable of effectively handling output classes that include numeric values. Therefore, it was necessary to categorize the price characteristic into classes that encompassed a certain range of costs. However, this clearly resulted in more opportunities for mistakes. The primary constraint of this study is the limited quantity of records utilized [5].

I utilized several regression approaches rooted in supervised machine learning to forecast the resale value of pre-owned automobiles, taking into account multiple variables like mileage,

fuel type, fiscal power, brand, model, and the year of manufacture. The gradient boosting regressor consistently demonstrated a high R-squared score and a low root mean square error across all evaluated models [6].

Forecasting the value of pre-owned vehicles is a notable and captivating field of study. Due to a surge in demand in the second-hand automobile industry, the business for both buyers and sellers has seen growth. In order to make trustworthy and precise predictions, it is necessary to possess specialist knowledge in the sector, since the price of automobiles is influenced by several significant elements. This study presents a supervised machine learning model that utilizes the KNN (K Nearest Neighbor) regression approach to examine the prices of pre-owned vehicles. We trained our model using a dataset of pre-owned vehicles obtained from the Kaggle website. In this experiment, the data was analyzed using various training and test ratios. The suggested model achieves an accuracy of around 85% and is fitted as the optimum model [7].

This research showcases an operational prototype for accurately predicting the prices of pre-owned vehicles, achieving a minimal margin of error. Multiple diverse factors are analyzed to ensure dependable and precise forecasts. The acquired findings are consistent with theoretical expectations and have demonstrated enhancement compared to models that employ basic linear models. An Artificial Neural Network (ANN) is constructed using the Keras Regression method, namely the Keras Regressor. Additionally, various Machine Learning Algorithms such as Random Forest, Lasso, Ridge, and Linear regressions are employed. The automobile dataset is used to test these techniques. The experimental findings indicate that the Random Forest model has the lowest error compared to other methods, with a Mean Absolute Error value of 1.0970472 and an R2 error value of 0.772584. The research conducted in this study has demonstrated significant implications for future investigations on predicting the prices of used cars using the Random Forest method. It has the potential to contribute to the resolution of fraud-related issues with a high level of accuracy, perhaps reaching one hundred percent [8].

## 3. Problem Statement

Building seven machine learning algorithms and subsequently applying them to five different datasets, four of which are sourced from various references, and one constructed from scratch using the OPENSOOQ website as a reference and building it from the ground up. Then, performing data preprocessing operations on these diverse datasets. The challenges involved constructing the five new datasets, selecting suitable features, and excluding some types that did not fit the data. Subsequently, determining how to preprocess these datasets to make them suitable for each of the built algorithms.

## 4. Methodology

After the data collection process and studying the related work, the first step was the process of preparing the dataset to become suitable for the classification process for all algorithms and to give all the results and then define the algorithms with the parameters for each one and then come up with the results (Accuracy, Precision, Recall, F1-score) and find the best algorithm.

Figure1.1 Describes the methods employed in this study for forecasting automobile price statistics. The following sections illustrate the proposed methodology.

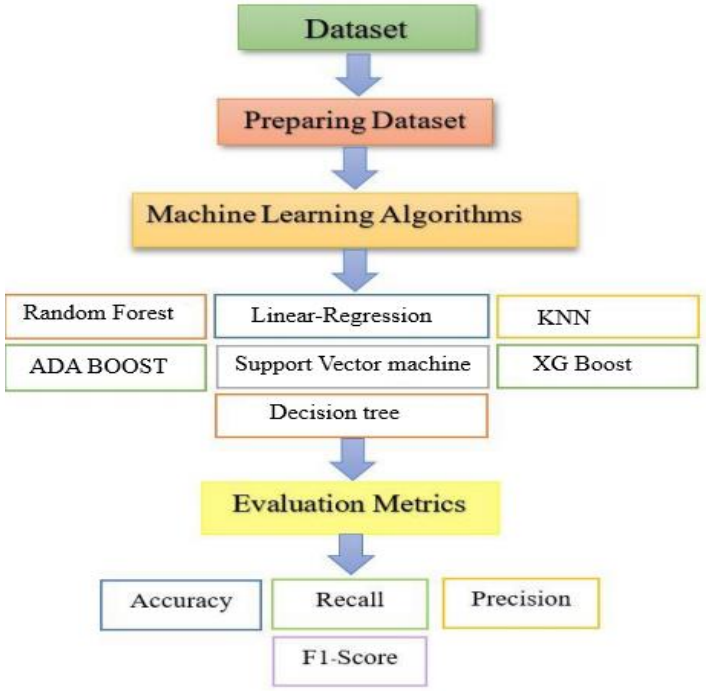


Figure 1: METHODOLOGY

## 5. Dataset Description

In this paper five datasets used as follow :

a new dataset collecting from scratch from opensooq website used , containing 8740 instance and 67 features , the output class is the price ( regression ) .

four dataset's from kaggle and research gate as follow :

- Ultimate Car Price Prediction Dataset , with 46,000 instance and 14 features
- Car price prediction(used cars) dataset , with 6829 instance and 9 features
- Used Car Price Prediction Dataset , with 4009 instance and 10 features
- Car Price Prediction Dataset , with 5000 instance , and 8 features .

Table 1 show the summarize of used datasets :

Table 1 : Datasets

| NAME | YEAR | INSTANCE | FEATURES | SOURCE |
|---|---|---|---|---|
| New dataset | 2023 | 8740 | 67 | Generate new dataset |
| Ultimate Car Price Prediction Dataset | 2023 | 46,000 | 10 | KAGGLE |

| | | | | |
|---|---|---|---|---|
| Car price prediction(used cars) | 2022 | 6829 | 9 | KAGGLE |
| Used Car Price Prediction Dataset | 2023 | 4009 | 10 | RESEARCH GATE |
| Car Price Prediction | 2022 | 5000 | 8 | KAGGLE |

## 6. PREPARING DATASESTS

In order to apply the machine learning techniques described in this study to the dataset, non-numerical characteristics will be turned into numerical features using a widely-used encoding technique called Ordinal Encoding. The technique referred to as "Label Encoder" is used to convert non-numerical input into a format that can be understood by machines. It accomplishes this by substituting each value with a unique integer, beginning from 0. This process is facilitated by Scikit-Learn's OrdinalEncoder.

Handle with the duplicated value bye delete the duplicate , and drop any record contain missing values .

Feature selection is an essential process in machine learning that entails selecting a subset of pertinent characteristics from the initial collection of features to optimize model performance, mitigate overfitting, and improve interpretability. Wrapper approaches are employed to choose subsets of features by evaluating the performance of a certain machine learning algorithm. The process entails training and assessing the model using several subsets of features, and then selecting the subset that produces the highest performance by employing the Recursive Feature Elimination (RFE) technique.

## 7. ALGORITHMS

I. Random Forest Algorithm (RF)

One kind of ensemble supervised learning technique used for regression and classification is called a random forest. It entails teaching a huge quantity of decision trees. The class that is selected by the majority of trees is used to forecast the outcome in classification tasks. Moreover, the mean or average prediction produced by each tree is the output in regression tasks. The problem of decision trees overfitting their training set is addressed by random decision forests. We use the random forest classifier to classify all cybersecurity datasets, which include multiple attacks. Because the label in our experiment is discrete, we used the random forest method with a classification type. The following are the precise RF settings that we used: 500 decision trees (n_estimators) are the number that is set. The square root of the total number of features is the maximum number of features that will be taken into account when determining the optimal split (max_features). There is no information provided about the decision trees' maximum depth (max depth). 42 is the value of the random state (random_state) [9].

II. The Decision Tree Algorithm (DT)

A decision tree is implemented by a machine learning algorithm that makes decisions based on a predefined set of rules, comparable to human decision-making processes [20]. Decision

tree learning, sometimes referred to as decision tree simulation, is a predictive modeling technique utilized in the domains of statistics, data mining, and machine learning [21]. The process involves transitioning from observations and a sample (shown as branches) to making inferences about the target value of the sample using a decision tree (represented as leaves corresponding to attack types) [22]. Classification trees are tree models that are used to forecast discrete target variables. In these tree structures, the leaves represent class labels, which are different sorts of attacks, while the branches represent attributes in the dataset. This allows us to predict the class labels accurately. [23, 24] Regression trees are decision trees that are used when the objective variable is continuous, often represented by real numbers [23]. While decision trees are valued for their comprehensibility and simplicity, they are also well recognized as one of the most prominent machine learning algorithms. In our experiment, we employed the Decision Tree algorithm with a classification type because the label we were dealing with is discrete. The specific parameters we utilized for the Decision Tree were criteria = entropy and random_state = 42 [10].

## III. Xtreme Gradient Boosting Algorithm (XGBoost)

Extreme Gradient Boosting, or XGBoost, is a machine learning technique that is often used for both regression and classification applications. The gradient boosting technique has undergone painstaking tuning and parallelization. By parallelizing the whole boosting procedure, the training time was significantly reduced. We train many models on different subsets of a training dataset and then use a vote mechanism to choose the best-performing model, as opposed to traditional approaches that try to produce the most optimum model utilizing the data. XGBoost frequently outperforms conventional gradient boosting techniques[10].

improving techniques. Many internal settings of the Python implementation are accessible and may be adjusted to improve accuracy and precision.

The primary purpose of this approach is to enhance the performance of weak learners, namely decision trees, by transforming them into strong learners. A strong learner is capable of producing the final prediction label, which is computed as the average of the predictions made by each weak classifier. The XGBoost [28, 29, 30] possesses several notable characteristics: 1) Parallelization: The model is specifically built to execute concurrently on multiple CPU cores. 2) Regularization: XGBoost provides a range of regularization penalties to mitigate overfitting. Regularization techniques that incorporate penalties contribute to effective training, enabling the model to achieve generalization[11].

Triumphantly. 3) Non-linearity refers to the XG-Boost's capability to identify and acquire knowledge from non-linear patterns in data. 4) Cross-validation is integrated and readily accessible. 5) Scalability: XGBoost has the potential to operate in a distributed manner, enabling the management of large volumes of data through the use of distributed servers and clusters such as Spark and Hadoop. Several programming languages, such as C++, JAVA, Python, and Julia, are compatible. For our experiment, we employed the XGBoost method with a classification type since the label is discrete. The specific settings utilized for XGBoost were as follows: colsample_bylevel = 1, learning_rate = 0.1, gamma = 0, n_estimators = 100, and random_state = 42.

## IV. K Nearest Neighbor Algorithm (KNN)

The k-nearest neighbor approach (k-NN), created in 1951 by Evelyn Fix and Joseph Hodges, was further expanded upon by Thomas Cover. Where Regression and data classification both utilize it. A data set containing the k-closest training samples is the input in both scenarios [13]. The following outcomes were attained based on the application of k-NN for regression or classification: The k-NN classification determines class membership. The item is placed in the most frequent class among its k-closest neighbors (where k is a positive number, usually small) based on the majority vote of its neighbors. In such case It is evident that an item is only allocated to the class of its nearest neighbor if k = 1 [14]. The value of an object's attribute is the result of kNN regression. Where The average of the k nearest neighbors' values is this value. The following details clarify how KNN functions [15]: First, the cybersecurity dataset is loaded. We set K to the desired number of neighbors in the second step. In the third phase, each sample of data has: For every third-stage data sample: 1) Using our data, we compute the distance between the query example and the present example. 2) It augments the ordered collection with the example's index and distance. The ordered set of distances and indices was sorted by distance in the fourth phase, going from smallest to greatest (in ascending order). In addition, in the fifth step, we select the first K entries from the sorted collection. The mean of the K labels in the regression and the mode of the K labels in the classification (as the label in our datasets) are returned in the sixth step to obtain the labels of the chosen K entries. Because the label in our experiment is discrete, we employed a KNN with classification type. The settings of the KNN that we utilized are as follows: n_neighbors = 1 and random_state = 42.

## V. The Decision Tree Algorithm (DT)

A decision tree is executed by a machine learning algorithm that renders judgments based on a predefined set of criteria, akin to human decision-making. Decision tree learning, often referred to as decision tree induction, is a predictive modeling technique utilized in the domains of statistics, data mining, and machine learning. The process involves transitioning from observations and a sample (shown as branches) to making inferences about the goal value of the sample using a decision tree. Classification trees are tree-based models that are used to forecast discrete target variables. In these tree structures, the leaves correspond to class labels, while the branches represent characteristics in the dataset. This allows us to make predictions about the class labels. Regression trees are decision trees that are used when the objective variable is continuous, often represented by real numbers [16]. While decision trees are valued for their comprehensibility and simplicity, they are also well recognized as one of the most prominent machine learning algorithms. In our experiment, we employed the Decision Tree algorithm with a classification type because the label we were dealing with is discrete. The specific parameters we utilized for the Decision Tree were criteria = entropy and random_state = 42.

## VI. Support Vector Machine (SVM)

The Support Vector Machine (SVM) serves as a supervised machine learning algorithm applicable to both classification and regression tasks, with a primary emphasis on classification. Its fundamental objective involves identifying the optimal hyperplane within an N-dimensional space, facilitating the segregation of data points into distinct classes within the feature space[17]. The hyperplane endeavors to maximize the margin between the closest

points of different classes. The dimensionality of the hyperplane is contingent on the number of features; for instance, with two input features, the hyperplane manifests as a line, while with three features, it transforms into a 2-D plane. Visualizing the hyperplane becomes challenging as the number of features surpasses three [18].

VII. ADABOOST

AdaBoost, also known as Adaptive Boosting, is a statistical classification meta-algorithm developed by Yoav Freund and Robert Schapire in 1995. Their work on AdaBoost was recognized with the Gödel Prize in 2003. It may be utilized in combination with several other learning algorithms to enhance performance. The boosted classifier aggregates the output of the weak learners, which are different learning algorithms, through a weighted sum to get the final result. Typically, AdaBoost is introduced for binary classification, but it may also be extended to handle multiple classes or limited intervals on the real line. AdaBoost is considered adaptive since it adjusts successive weak learners to focus on examples that were misclassified by earlier classifiers[19]. Some issues may exhibit a lower susceptibility to overfitting compared to other learning techniques. While individual learners may exhibit weakness, if each learner's performance above that of random guessing, the total model may be demonstrated to converge towards a robust learner. AdaBoost is commonly employed to aggregate weak base learners, such decision stumps. However, research has demonstrated that it can also successfully aggregate strong base learners, such as deep decision trees, resulting in a more precise model.Each learning algorithm exhibits varying degrees of suitability for different issue types and often requires the adjustment of several parameters and configurations in order to get optimal performance on a given dataset. AdaBoost, which uses decision trees as weak learners, is commonly regarded as the most effective pre-built classifier. When AdaBoost method is combined with decision tree learning, the information collected at each step regarding the difficulty level of each training sample is incorporated into the tree growth algorithm. This ensures that later trees primarily concentrate on instances that are more challenging to categorize.

## 8. Evaluation Metrics

There are several assessment measures to examine the machine learning algorithms that were utilized, including

**I.** Mean Squared Error (MSE): Represents the average of the squared differences between predicted and true values.

Formula : MSE = $\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ [21].

**II.** R-squared (R2) Score:

Measures the proportion of the variance in the dependent variable that is predictable from the independent variables. Ranges from 0 to 1, where 1 indicates a perfect prediction.

Formula :

$$\frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2},$$

[22].

## 9. Results

In this experiment, machine learning algorithms were applied to predict car price , figure 2 show the opensooq dataset Root mean Squared Error (RMSE) :

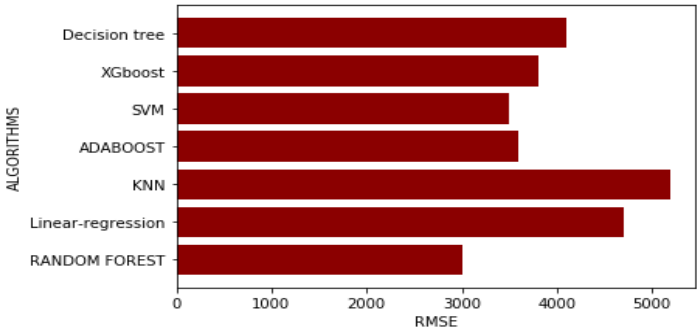


Figure 2: RSME RESULT ON OPENSOOQ DATASE

Figure 2 demonstrates that the Random Forest regression technique has yielded the lowest Root Mean Squared Error (RMSE) on the test dataset, outperforming all other algorithms. It indicates that the Random Forest algorithm has achieved good performance.

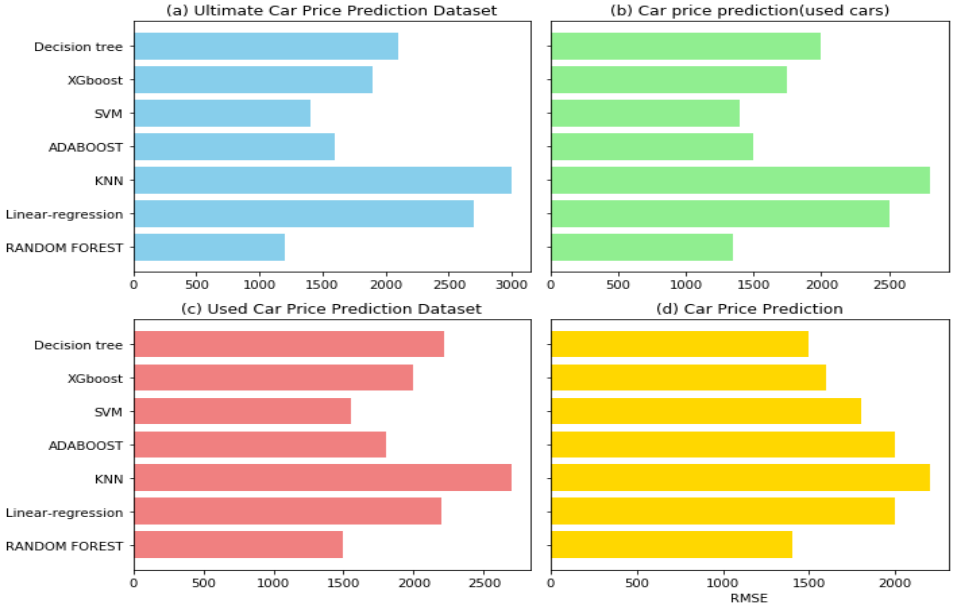


Figure 3: the four datasets results

Figure 3 demonstrates that the Random Forest regression technique has yielded the lowest Root Mean Squared Error (RMSE) across all datasets, outperforming all other algorithms. It indicates that the Random Forest algorithm has achieved a high level of performance.

According to R-squared (R2) Score, figure 4 show the result on opensooq dataset , shown in figure 4 as follow :

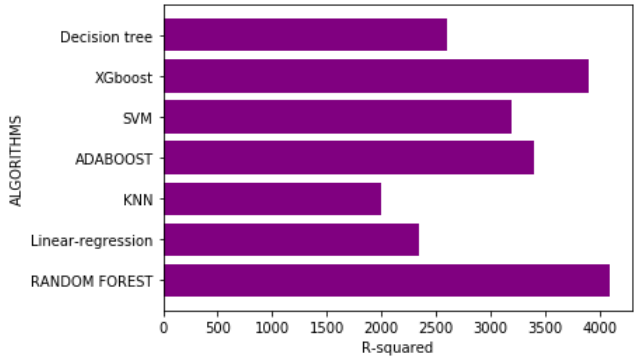


Figure 4: opensooq dataset results

Figure 4 is utilized to assess the R-Squared value of several methods. Both the random forest and XG boost algorithms exhibit similar and greater R2 scores compared to the other five techniques. Random forest is the algorithm with the greatest R2 score. The variance between the best fitted line and the mean line of the data set is the lowest among other methods, as determined by the R2 Score calculation.

After that , the result of R-squared on the four dataset shown in figure 5 as follow :


Figure 5

Figure 5 is utilized to examine the R-Squared value of methods across all datasets. Both the random forest and XG boost algorithms have consistently achieved superior R2 scores compared to the other five techniques. Random forest is the algorithm with the greatest R2 score. The variance between the best fitted line and the mean line of the data set is the lowest among other methods, as determined by the R2 Score calculation.

## 10. Conclousin and Future Works

In this paper , 7 machine learning algorithms ('RANDOM FOREST', 'Linear-regression', 'KNN', 'ADABOOST' , 'SVM' , 'XGboost','Decision tree') were implemented to predict the car price , using 5 datasets , 4 of them from kaggle and research gate(Ultimate Car Price Prediction Dataset , Car price prediction(used cars) , Used Car Price Prediction Dataset ,and Car Price Prediction) , and generate a new one from opensooq website contain 8740 instance and 67 features . using the evaluation metrics :

Mean Squared Error (MSE) , and R-squared (R2) Score , the results yield that Random forest give the highest R-squared and lowest mean squared error(MSE).

FUTURE WORK

Build a new model using an artificial neural network and try to reach better results and make the new dataset bigger by adding new instance to it to make it more reliable .

## References

1. Gegic, E., Isakovic, B., Keco, D., Masetic, Z., & Kevric, J. (2019). Car price prediction using machine learning techniques. TEM Journal, 8(1), 113.
2. Gajera, P., Gondaliya, A., & Kavathiya, J. (2021). Old car price prediction with machine learning. Int. Res. J. Mod. Eng. Technol. Sci, 3, 284-290.
3. Chandak, A., Ganorkar, P., Sharma, S., Bagmar, A., & Tiwari, S. (2019). Car Price Prediction Using Machine Learning. International Journal of Computer Sciences and Engineering, 7(5), 444-450.
4. Asghar, M., Mehmood, K., Yasin, S., & Khan, Z. M. (2021). Used cars price prediction using machine learning with optimal features. Pakistan Journal of Engineering and Technology, 4(2), 113-119.
5. Pudaruth, S. (2014). Predicting the price of used cars using machine learning techniques. Int. J. Inf. Comput. Technol, 4(7), 753-764.
6. Hankar, M., Birjali, M., & Beni-Hssane, A. (2022, May). Used car price prediction using machine learning: A case study. In 2022 11th International Symposium on Signal, Image, Video and Communications (ISIVC) (pp. 1-4). IEEE.
7. Samruddhi, K., & Kumar, R. A. (2020). Used car price prediction using k-nearest neighbor based model. Int. J. Innov. Res. Appl. Sci. Eng.(IJIRASE), 4, 629-632
8. Varshitha, J., Jahnavi, K., & Lakshmi, C. (2022, January). Prediction of used car prices using artificial neural networks and machine learning. In 2022 International Conference on Computer Communication and Informatics (ICCCI) (pp. 1-4). IEEE.
9. Akar, Ö., & Güngör, O. (2012). Classification of multispectral images using Random Forest algorithm. Journal of Geodesy and Geoinformation, 1(2), 105-112.
10. Charbuty, B., & Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. Journal of Applied Science and Technology Trends, 2(01), 20-28.

11. Bentéjac, C., Csörgő, A., & Martínez-Muñoz, G. (2021). A comparative analysis of gradient boosting algorithms. Artificial Intelligence Review, 54, 1937-1967.
12. Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., ... & Zhou, T. (2015). Xgboost: extreme gradient boosting. R package version 0.4-2, 1(4), 1-4.
13. Sun, S., & Huang, R. (2010, August). An adaptive k-nearest neighbor algorithm. In 2010 seventh international conference on fuzzy systems and knowledge discovery (Vol. 1, pp. 91-94). IEEE.
14. Kozma, L. (2008). k Nearest Neighbors algorithm (kNN). Helsinki University of Technology, 32.
15. Peterson, L. E. (2009). K-nearest neighbor. Scholarpedia, 4(2), 1883.
16. Charbuty, B., & Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. Journal of Applied Science and Technology Trends, 2(01), 20-28.
17. Jakkula, V. (2006). Tutorial on support vector machine (svm). School of EECS, Washington State University, 37(2.5), 3.
18. Huang, S., Cai, N., Pacheco, P. P., Narrandes, S., Wang, Y., & Xu, W. (2018). Applications of support vector machine (SVM) learning in cancer genomics. Cancer genomics & proteomics, 15(1), 41-51.
19. Qaddara, I. A. A. R. (2022). APPLYING MACHINE LEARNING TECHNIQUES ON CYBER SECURITY DATASETS: DETECTING CYBER ATTACKS. Harbin Gongye Daxue Xuebao/Journal of Harbin Institute of Technology, 54(7), 95-110.
20. Wang, Y., & Feng, L. (2020). Improved Adaboost algorithm for classification based on noise confidence degree and weighted feature selection. IEEE Access, 8, 153011-153026.
21. Das, K., Jiang, J., & Rao, J. N. K. (2004). Mean squared error of empirical predictor.
22. Onyutha, C. (2020). From R-squared to coefficient of model accuracy for assessing" goodness-of-fits". Geoscientific Model Development Discussions, 2020, 1-25.
23. Qaddara, I. A., Kenana, A. J., Al-Tarawneh, K. M., & Sarhan, S. (2024). Evaluation of CPU Scheduling and Synchronization Algorithms in Distributed Systems. Nanotechnology Perceptions, 698-719.