Journal of Soft Computing Exploration

Homepage: htttps://shmpublisher.com/index.php/joscex

p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

Improving car price prediction performance using stacking ensemble learning based on ann and random forest

Yulizchia Malica Pinkan Tanga[1], Robert Panca R. Simanjuntak[2], Rofik[3], Much Aziz Muslim[4]
[1,2,3]Department of Computer Science, Universitas Negeri Semarang, Indonesia
[4]Faculty of Technology Management, Universiti Tun Hussein Onn Malaysia, Malaysia

Article Info

Article history:

Received August 27, 2024
Revised September 27, 2024
Accepted September 27, 2024

Keywords:

Car selling price prediction
Artificial neural networks
Random forest
Ensemble technique

ABSTRACT

Determining the right selling price for a car can be a challenge for car sales companies. The selling price of a car is highly influenced by car characteristics such as brand, type, year of production, fuel type, and mileage. Therefore, the research aims to develop a more accurate model of car price prediction model by using a stacking ensemble technique that combines Random Forest and ANN. Random Forest is effective in handling outliers and reducing the risk of overfitting, while ANN has the advantage of capturing complex nonlinear patterns. The results show that the stacking ensemble model combining ANN and Random Forest can predict car sales prices by achieving an R^2 value of 0.97. The results of this study can help distributors in selling cars make the right decisions regarding the sales price of cars. To improve the generalization of the model, future research is recommended to try a combination of different ensemble methods and the use of larger and more diverse datasets.

Corresponding Author:

Yulizchia Malica Pinkan Tanga,
Department of Computer Science,
Universitas Negeri Semarang, Indonesia
Email: yulizchiamalica@students.unnes.ac.id
https://doi.org/10.52465/joscex.v5i3.462

1. INTRODUCTION

Car sales are one of the most promising and strategic businesses in this modern era. The market for the sale of used cars is a growing industry, in recent years, it has almost quadrupled in market value [1]. Pricing products on the market involves both science and art and both require the use of statistical and experimental procedures. [2]. The correct price of the automobile is difficulty for this company to determine [3]. Because quality often depends on a number of different qualities and circumstances, accurate prediction of the price of vehicles requires unique experience [4]. Generally, the most important are the make and model name, year, miles driven, and mileage [5], [6]. Selling prices that are too low can cause losses for sellers; conversely, prices that are too high can reduce buyer interest. In the world of automobile business, one of the important things to consider is the prediction of car sales prices [7]. Consequently, a scientific value conversion procedure is necessary to accurately estimating the cost of second-hand automobiles [8]. By knowing the accurate selling price of a car, car business owners can determine the right car selling price in accordance with current market conditions.

Machine learning is a field of science that studies how to teach computers to operate autonomously without requiring detailed instructions [9]. This concept belongs to the branch of artificial intelligence which is one of the most popular terms in the 21st century. Machine learning has been applied in various industries, such as banking [10], [11] e-commerce [12], healthcare [13], and many others. Often, this technology is also used to predict the selling price of cars [14]–[16].

Mustapha Hankar et al. [17] in 2022 compared several regression techniques to predict the selling price of used cars considering many factors such as mileage, fuel type, fiscal power, make, model and year of car production. Regression models used such as the K-nearest neighbors regressor (KNN), random forest regressor (RFR), gradient bosting regressor (GBR), and artificial neural network (ANN). The results showed that the gradient-boasting regressor model achieved the highest R^2 value of 0.80 and RMSE reached 44516.20. Another study conducted by Muhammad Asghar et al. [18] in 2021 aims to predict used car prices using the Ordinary Least Squares (OLS) linear regression model. This study uses the Recursive Feature Elimination (RFE) method to identify optimal features that include fuel type, car body type, and horsepower. The results show that the OLS regression model achieves an R^2 value of 0.90, which indicates a fairly accurate prediction.

Snehit Shaprapawad et al. [19] in 2023 conducted research by comparing several machine learning models, such as linear regression, Lasso Regression, Elastic Net Regression, Decision Tree, Random Forest Regressor, and support vector regression to predict the selling price of used cars. This research uses hyperparameter tuning techniques and model evaluation with metrics such as R^2, mean absolute error (MAE), Mean Squared Error (MSE), and mean squared error (RMSE). The results showed that the support vector regression model provided the best results by achieving an R^2 value of 95.27, 0.142 MAE, 0.047 MSE and 0.218 RMSE. Aravind Sasidharan Pillai [20] in 2022 used the Artificial Neural Network (ANN) to predict used car prices. The ANN model is compared with other models such as random forest, gradient boosting, and Linear Regression. The results achieved an R^2 value of 0.96, 1960.37 MAE, 0.11 MAPE, and 2104.13 RMSE. Purwa Hasan Putra et al. in 2023 [21] proposed random forest and decision tree methods to predict car prices. This study uses a dataset sourced from Kaggle about predicting car prices. The analysis employing decision trees and random forests produced various percentage findings. The random forest has a precision of 72.13 while the choice tree has an accuracy of 67.21%.

Previous research has proposed several models, such as KNN, ANN, Gradient Boosting, Random Forest, and Support Vector Regressor. These models provide good results but often face problems such as overfitting or limitations in handling complex data variations. For example, research by Aravind Sasidharan Pillai [20] found that the Artificial Neural Network (ANN) outperforms other methods but requires more computational time. Meanwhile, Random Forest has better performance handling outliers but is less accurate in handling complex features. Therefore, this research uses ensemble stacking by combining ANN and random forest regression. The stacking ensemble is used to improve prediction by using the advantages of each model. ANN and Random Forest have strong capabilities in handling complex prediction problems such as car selling prices. ANN can capture non-linear patterns [22] patterns that may be hidden in the data due to its multilayered structure and deep learning capabilities. On the other hand, Random Forest is a powerful ensemble model [23] model that can overcome overfitting and is capable of providing accurate predictions by combining many decision trees. The combination of these two models is expected to produce a more stable and reliable prediction model, considering that car price prediction depends on a variety of diverse and complex variables.

2. METHOD

This research uses stacking ensemble techniques that combine prediction results from Artificial Neural Networks (ANN) and Random Forest Regressor models. The stages in this research include data collection, data processing, ANN and Random Forest model training, and combining prediction results through ensemble stacking. Figure 1 shows the flow chart of the stages of this research.

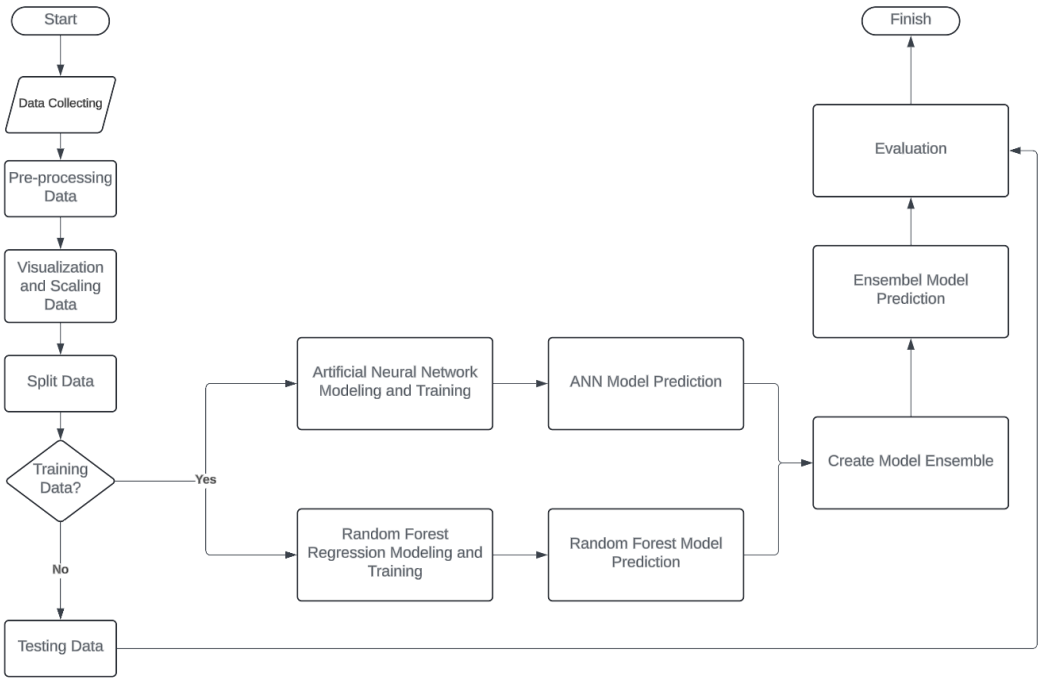

Figure 1. Research Methods

Data Collecting

The approach suggested in this research was created using the Python programming language, which is supported by Google Collaboratory. The dataset is taken from the official kaggle website and is in csv form. The data collection stage is performed by downloading it through a publicly accessible platform, Kaggle. The data set can be accessed via the URL: https://www.kaggle.com/datasets/yashpaloswal/ann-car-sales-price-prediction. The data set consists of 500 data with 9 features. The features in the dataset include customer name, customer email, country, gender, age, annual salary, credit card debt, net worth, number of car purchases. The data set was then saved to Google Drive so that it could be called into Google Collaboratory. In the early stages, the researchers connected Google Collaboratory with Google Drive containing the datasets used.

Pre-processing Data

Data preprocessing was carried out by deleting several columns that were not needed, such as the customer's name, customer e-mail, country and gender columns. The techniques used in this process are Data Clearning. Data cleaning is applied to remove irrelevant attributes, so that the dataset becomes simpler and is focused on important features.

Feature selection is performed to select the most relevant features for the model. This research uses Recursive Feature Elimination (RFE) with the Random Forest model. The results of the feature selection show that the age, annual salary, and net worth columns are the selected features, while the credit card debt column is still included to improve the model prediction. Subsequently, a separation was made between the features or inputs consisting of the age, annual salary, credit card debt, and net worth columns and the target or output consisting of the column of number of car purchases. After that, a separation is made between the features or inputs consisting of age, annual salary, credit card debt, and net worth columns, and the target or output consisting of the number of car purchases column.

The data are then scaled using MinMaxScaler to ensure that the feature values are within the same range to maintain model performance. MinMaxScaler is a data normalization method that converts data into a range of 0 and 1. The calculation in MinMaxScaler is shown in the Formula (1).

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

.

Explanation:
X : original value of the data
X_{min} : the minimum value of the feature
X_{max} : the maximum value of the feature
X_{scaled} : normalized data value

Visualization and Scaling of Data

To help to understand the features of the data being used, data visualization is done [24]. Each numerical feature in the data set receives a histogram from the program, which in this case employs the seaborn and matplotlib libraries. This histogram gives us a broad picture of the distribution and distribution of data for each attribute, enabling us to determine whether the data contain outliers, skewness, or specific patterns. We can utilize this representation to aid in our early understanding of the dataset that will be used. Additionally, data scaling is done to keep the scales consistent and uniform for all the dataset features. Data scaling is done using MinMaxScaler from sklearn. Scaling is done by converting each feature value into a range of 0 to 1, thereby ensuring that all features have a similar scale.

Split Train Data and Test Data

Split training data and test data using train_test_split [24]. Separated data was carried out with a ratio of 8: 2, where train data = 0.8 and the test data = 0.2. Dividing the data into two separate groups based on a specified proportion ensures that the original data is fairly represented in both groups.

Artificial Neural Network

Artificial neural network (ANN) is a machine learning algorithm that mimics the way the human brain processes information [3]. ANN forkflow can be seen in Figure 2.

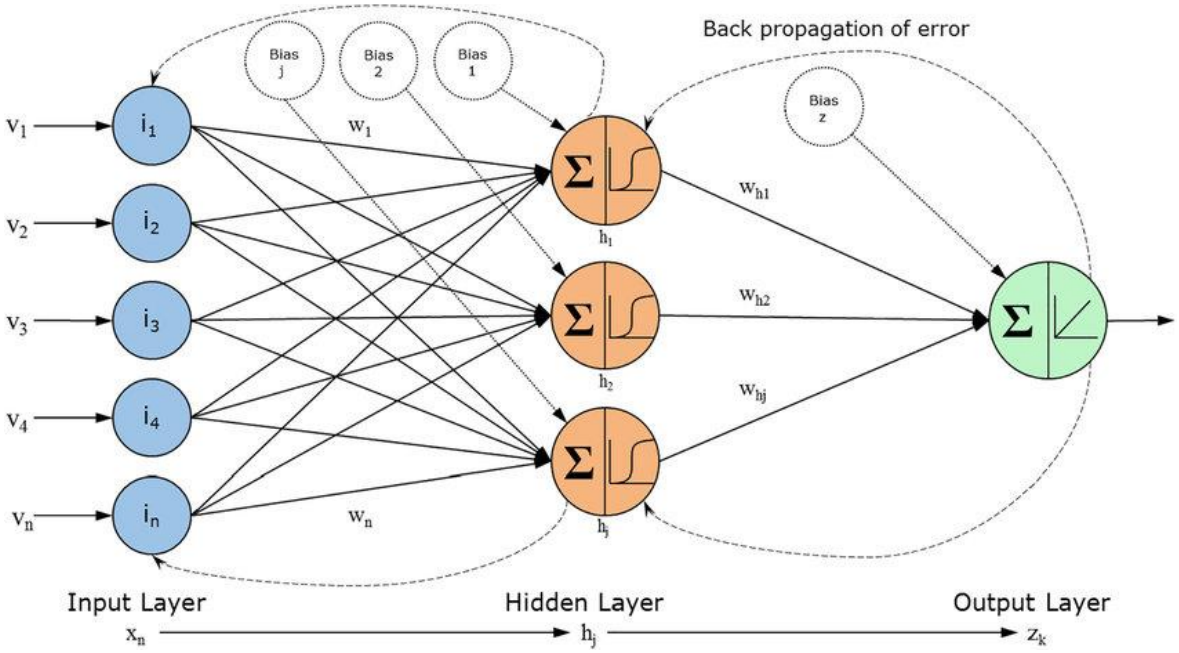

Figure 2. ANN work flow [25]

ANN consists of several layers of neurons, namely the input layer, the hidden layer, and the output layer. To obtain a non-linear pattern, the input for each neuron is multiplied by its weight, summed with the bias, and processed through an activation function such as ReLU, sigmoid, or tanh in the hidden layer. The output of each neuron is calculated using formula (2).

$$y = f = \sum_{i=i}^{n} w_i x_i + b \tag{2}$$

Explanation:
x_i : input to the neuron (features of the data)
w_i : weight for each input

b : bias added to shift the activation
f : activation function such as ReLU, sigmoid, or tanh
y : output of the neuron

The model is trained using the backpropagation technique to update the weights by gradient descent optimization to reduce the mean squared error (MSE) loss function. The weights are updated every iteration until convergence, which is indicated by the minimization of the error between the prediction and the actual target.

The training of the ANN model was carried out for 50 iterations (epoch), 20% of the training data serving as validation data. The training optimizes the model weights based on the loss function mean_squared_error and updates them iteratively. After training the ANN model, history_ann contains the training history, performance metrics, and loss at each epoch. This can be used to analyze and visualize changes in model performance during training.

Random Forest

Random Forest is an ensemble model algorithm that combines the results of multiple decision trees to improve prediction and reduce overfitting. RF can be used to classify regression problems and classify sets of methods [26]. Random forest workflow can be seen in Figure 3.

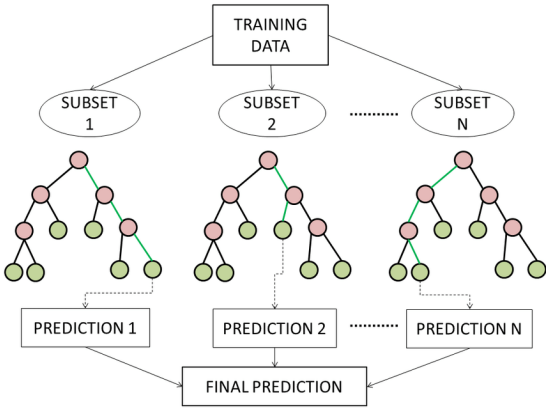

Figure 3. Random forest workflow [27]

The algorithm works by building a number of decision trees on a random subset of the training data and then taking the average of each tree's predictions to produce the final prediction in the regression. A bagging technique (Booststrap Aggregating) is used to train each tree in the random forest. Generate the final prediction in the random forest regression model is shown in Formula (3).

$$\hat{y}RF = \frac{1}{T}\sum_{t=1}^{T}\hat{y}^{(t)} \quad (3)$$

Explanation:
$\hat{y}RF$: the final prediction of random forest
T : total number of trees (decision tree)
$\hat{y}^{(t)}$: prediction of the decision tree to t

The Random Forest model uses the RandomForestRegressor function, this model was trained with 100 decision trees. The random_state parameter is set to 0 to ensure consistent results every time the code is run. This model is trained using train data. Random forest builds a regression model by examining the correlation between the target value y_train and the feature x_train during the training phase. Once the model is trained, it is used to predict the values in the test data.

Ensemble Model

Stacking is one of the ensemble learning techniques that aims to improve prediction performance by combining the results of several different models. In this study, the stacking ensemble combines the prediction results of the ANN and RF models using the RandomForestRegressor. When the prediction results of the two models are merged, it increases the variety and complexity of the features used by the ensemble model. Then test using the train data.

Predictions from the ensemble model using the random forest regression model. The ensemble model can improve overall performance by utilizing the predictive power of both models with these additional features. After combining the prediction results from the ANN and RF models as additional features, the next step is to make predictions using the RF ensemble model that has been created.

Evaluation

Calculate the R^2 score from the prediction results of the ensemble model by calling the r2_score method from sklearn. The R2 score is used to evaluate the performance of the ensemble model. The formula for the R^2 score is shown in Formula (4).

$$R^2 = 1 - \left(\frac{SSR}{SST}\right) = 1 - \frac{\sum_i (yi - \hat{y}i)^2}{\sum_i (yi - \bar{y})^2} \quad (4)$$

SSR (Sum of squared residuals) is the sum of squares between the true value (yi) and the predicted value i for each sample. The SSR indicates how much variability the model cannot explain. The smaller the SSR, the better the regression model explains the variation in the data. SST (Total Sum of Squares) is the sum of the squares of the difference between the true value (yi) and the mean value ($\bar{y}$) for each sample. SST measures the total variation in the data without considering the model. A higher level of SST indicates that the variation in the data that needs to be explained by the model is greater. R^2 score i.e., the R^2 value, also referred to as the coefficient of Determination, is the percentage of variation in the data that can be explained by the model. The R^2 score is calculated by dividing SSR (the sum of the squares of the difference between the true values) by SST (the total sum of the squares of the errors of the average model) and then subtracting 1. A value of 0 indicates that the model does not explain the variation in the data well, while a value indicates that the model does explain the variation in the data well.

3. RESULTS AND DISCUSSIONS

Data collection, data analysis, and problem identification are all necessary components of this study to locate useful information in the data set and to better comprehend the data that will be used. The dataset used in this research comes from Kaggle, namely ANN-Car Sales Price Prediction in CSV format. The data contains 500 rows and 9 columns consisting of customer name, customer email, age, annual salary, credit card debt, and net worth.

Data analysis was carried out to gain insight from the data by visualizing four numerical columns: age, annual salary, credit card debt, and net worth. The age distribution appears normal, with the majority of customers around 40 years old. The annual salary also has a symmetrical distribution, with most customers in the $60,000 to $80,000 salary range. Credit card debt shows a range of $10,000 to $20,000 and most customers have a net worth of around $400,000. The required columns are age, annual salary, credit card debt, and net worth, which can be seen in Figure 4.

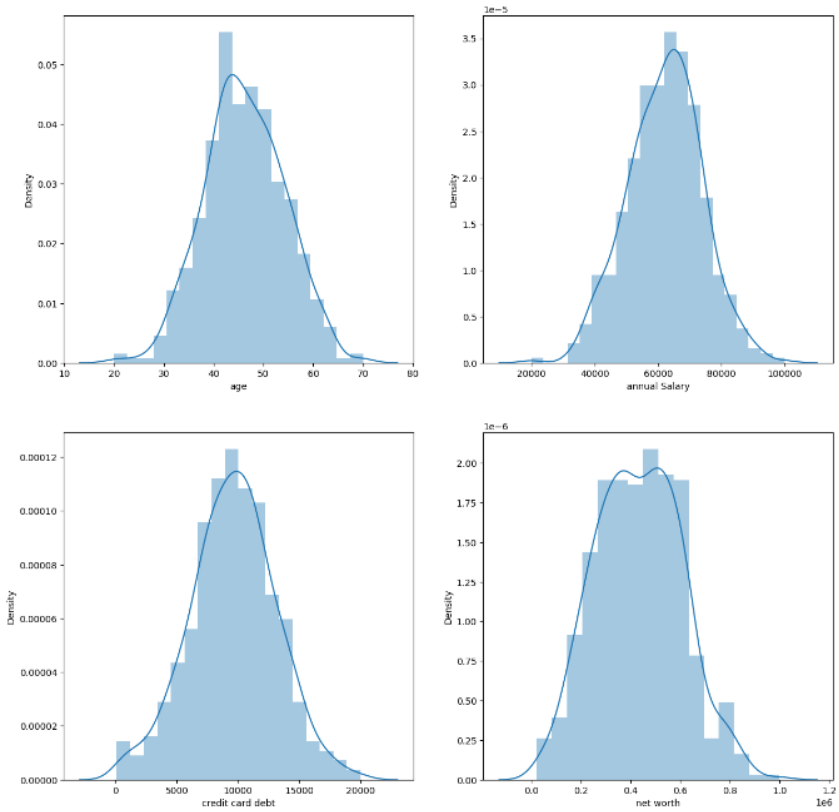

Figure 4. Required features

The proposed ANN model has 171 parameters that can be examined. The predefined model structure is associated with this number of parameters. In this model, there are three dense (fully connected) layers, namely dense_23, dense_24, and dense_25. Each layer contains a number of neurons. In the ANN model, there are 50 epochs that have been run. The model examines the training data and changes its weights each time to reduce the loss function. When model training is performed on the training data (x_train and y_train), the loss indicates how well the model predicts the target value so that it matches the actual value at each iteration. In the first epoch, the training loss is 2028900736.0000 which continues to decrease each subsequent period. Each epoch when model validation are performed on data that is 20% of the training data. This is done to ensure the performance of the model on data that are not used during training. The validation loss is 2161087232.0000 in the first epoch and will decrease in every subsequent epoch. The layers of the Artificial Neural Network (ANN) model for predicting car sales prices using TensorFlow-Hard are shown in Table 1.

Table 1. ANN layers

Layer (type)	Output Shape	Param #
dense_23 (Dense)	(None, 10)	50
dropout_10 (Dropout)	(None, 10)	0
dense_24 (Dense)	(None, 10)	110
Dropout_11 (Dropout)	(None, 10)	0
dense_25 (Dense)	(None, 1)	11
Total params: 171		
Trainable params: 171		
Non-trainable params:0		

In this study, the results obtained by the research model of combining ANN and the random forest regressor using a stacking ensemble will be compared with existing research models. The model comparison can be seen in Table 2.

Table 2. Comparison with previous research

Author	Year	Method	R^2
Mustapha Hankar [17]	2020	Gradien Boosting	0.80
Muhammad Asghar [18]	2021	Linear Regression	0.90
Snehit Shaprapawad [19]	2023	Support Vector Regressor	0.95
Aravind Sasidharan Pillai [20]	2022	Artificial Neural Network	0.96
Proposed Method	**2024**	**Stacking Ensemble**	**0.97**

In this research, the stacking ensemble is used to combine the prediction results of the ANN model with the prediction results of Random Forest. The results show information about the model evaluation process. The test data was processed with 13 iterations, each taking about 2 milliseconds per iteration. This research used the ensemble stacking technique to combine the predictions of ANN and Random Forest, which resulted in an R^2 value of 0.97. The results of the R^2 score obtained indicate that this ensemble stacking model has better prediction capabilities compared to models in previous studies.

4. CONCLUSION

In this study, the use of an ensemble stacking model that combines Random Forest Regression and Artificial Neural Network (ANN) to predict used car price was successfully demonstrated in this study with an R^2 score of 0.97. Compared to the models in previous studies, this combination of ensemble stacking models shows better performance. The ANN model excels in handling nonlinear relationships, and Random Forest is able to handle data variations and outliers well. This shows the ability to improve R^2 scores by using ensemble stacking techniques. Although these results show the high effectiveness of the stacking ensemble model, it is recommended for future research to explore other combinations of ensemble techniques and perform evaluations using larger and more diverse datasets. This can help improve the predictability and generalizability of the model, as well as ensure its reliability in wider practical applications.

REFERENCES

[1] P. Venkatasubbu and M. Ganesh, "Used Cars Price Prediction using Supervised Learning Techniques," *Int. J. Eng. Adv. Technol.*, vol. 9, no. 1S3, pp. 216–223, Dec. 2019, doi: 10.35940/ijeat.a1042.1291s319.

[2] A. Fathalla, A. Salah, K. Li, K. Li, and P. Francesco, "Deep end-to-end learning for price prediction of second-hand items," *Knowl. Inf. Syst.*, vol. 62, no. 12, pp. 4541–4568, 2020, doi: 10.1007/s10115-020-01495-8.

[3] A. Varol, M. Mehdi Karakoç, and G. Çelik, "Car Price Prediction Using An Artificial Neural Network," 2020.

[4] C. Bo and H. Mammadov, "Car Price Prediction in the Usa By," vol. 11, no. January, pp. 99–108, 2021.

[5] S. MUTİ and K. YILDIZ, "Using Linear Regression For Used Car Price Prediction," *Int. J. Comput. Exp. Sci. Eng.*, vol. 9, no. 1, pp. 11–16, 2023, doi: 10.22399/ijcesen.1070505.

[6] N. Patel, "Car Price Prediction," no. 04, pp. 4856–4860, 2023.

[7] B. Kriswantara, Kurniawati, and H. F. Pardede, "Prediksi Harga Mobil Bekas dengan Machine Learning," vol. 6, no. March, pp. 1–19, 2021.

[8] B. Cui, Z. Ye, H. Zhao, Z. Renqing, L. Meng, and Y. Yang, "Used Car Price Prediction Based on the Iterative Framework of XGBoost+LightGBM," *Electron.*, vol. 11, no. 18, 2022, doi: 10.3390/electronics11182932.

[9] B. Saireddy, A. Vamshikrishna, G. Abhilash, and D. Vinith Srinivas, "CAR PRICE PREDICTION USING MACHINE LEARNING," *www.irjmets.com @International Res. J. Mod. Eng.*, 1655.

[10] R. Rofik, R. Aulia, K. Musaadah, S. S. F. Ardyani, and A. A. Hakim, "Optimization of Credit Scoring Model Using Stacking Ensemble Learning and Oversampling Techniques," *J. Inf. Syst. Explor. Res.*, vol. 2, no. 1, Dec. 2023, doi: 10.52465/joiser.v2i1.203.

[11] D. A. A. Pertiwi, K. Ahmad, S. N. Salahudin, A. M. Annegrat, and M. A. Muslim, "Using Genetic Algorithm Feature Selection to Optimize XGBoost Performance in Australian Credit," *J. Soft Comput. Explor.*, vol. 5, no. 1, pp. 92–98, 2024, doi: 10.52465/joscex.v5i1.302.

[12] R. A. Putra and E. Nurmawati, "Prediction-based Stock Portfolio Optimization Using Bidirectional Long Short-Term Memory (BiLSTM) and LSTM," vol. 11, no. 3, pp. 609–620, 2024, doi: 10.15294/sji.v11i3.5941.

[13] A. Nurizki, A. Fitrianto, and A. M. Soleh, "Performance of Ensemble Learning in Diabetic Retinopathy Disease Classification Performance of Ensemble Learning in Diabetic Retinopathy Disease Classification," vol. 11, no. May, pp. 375–386, 2024, doi: 10.15294/sji.v11i2.4725.

[14] A. S. J. Alexstan, K. M. Monesh, M. Poonkodi, and V. Raj, "Used Car Price Prediction Using Machine Learning," *IoT, Cloud Data Sci.*, vol. 124, no. April, pp. 512–517, 2023, doi: 10.4028/p-9x4ue8.

[15] A. Ospanova, V. C. Sanap, M. M. Rangila, S. Rahi, S. Badgujar, and Y. Gupta, "Car Price Prediction using Linear Regression Technique of Machine Learning Education 005 View project Infomatics View projaect Car Price Prediction using Linear Regression Technique of Machine Learning," *Artic. Int. J. Innov. Res. Sci. Eng. Technol.*, vol. 9001, no. April, 2008, doi: 10.15680/IJIRSET.2022.110405.

[16] A. Chandak, P. Ganorkar, S. Sharma, A. Bagmar, and S. Tiwari, "Car Price Prediction Using Machine Learning," *Int. J. Comput. Sci. Eng.*, vol. 7, no. 5, pp. 444–450, 2019, doi: 10.26438/ijcse/v7i5.444450.

[17] M. Hankar, M. Birjali, and A. Beni-Hssane, "Used Car Price Prediction using Machine Learning: A Case Study," *11th Int. Symp. Signal, Image, Video Commun. ISIVC 2022 - Conf. Proc.*, pp. 1–4, 2022, doi: 10.1109/ISIVC54825.2022.9800719.

[18] M. Asghar, K. Mehmood, S. Yasin, and Z. Mehboob Khan, "Used Cars Price Prediction using Machine Learning with Optimal Features," 2021.

[19] S. Shaprapawad, P. Borugadda, and N. Koshika, "Car Price Prediction:An Application of Machine Learning," *6th Int. Conf.*

Inven. Comput. Technol. ICICT 2023 - Proc., no. Icict, pp. 242–248, 2023, doi: 10.1109/ICICT57646.2023.10134142.
[20] A. S. Pillai, "A Deep Learning Approach for Used Car Price Prediction," vol. 3, no. 3, pp. 31–51, 2022.
[21] P. H. Putra, B. Purba, and Y. A. Dalimunthe, "Random forest and decision tree algorithms for car price prediction," vol. 1, no. 2, pp. 81–89, 2023.
[22] R. Rosita, D. Ananda Agustina Pertiwi, and O. Gina Khoirunnisa, "Prediction of Hospital Intesive Patients Using Neural Network Algorithm," *J. Soft Comput. Explor.*, vol. 3, no. 1, pp. 8–11, 2022, doi: 10.52465/joscex.v3i1.61.
[23] A. D. Goenawan and S. Hartati, "The Comparison of K-Nearest Neighbors and Random Forest Algorithm to Recognize Indonesian Sign Language in a Real-Time," *Sci. J. Informatics*, vol. 11, no. 1, pp. 237–244, 2024, doi: 10.15294/sji.v11i1.48475.
[24] A. Alhakamy, A. Alhowaity, A. A. Alatawi, and H. Alsaadi, "Are Used Cars More Sustainable? Price Prediction Based on Linear Regression," *Sustainability*, vol. 15, no. 2, p. 911, 2023, doi: 10.3390/su15020911.
[25] D. Zafeiris, S. Rutella, and G. R. Ball, "An Artificial Neural Network Integrated Pipeline for Biomarker Discovery Using Alzheimer's Disease as a Case Study," *Comput. Struct. Biotechnol. J.*, vol. 16, pp. 77–87, 2018, doi: 10.1016/j.csbj.2018.02.001.
[26] E. Gegic, B. Isakovic, D. Keco, Z. Masetic, and J. Kevric, "Car price prediction using machine learning techniques," *TEM J.*, vol. 8, no. 1, pp. 113–118, Feb. 2019, doi: 10.18421/TEM81-16.
[27] G. Laudato *et al.*, "Identification of R-peak occurrences in compressed ECG signals," in *2020 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, IEEE, Jun. 2020, pp. 1–6. doi: 10.1109/MeMeA49120.2020.9137207.

.