A Deep Learning Approach for Used Car Price Prediction

*By **Aravind Sasidharan Pillai***

Principal Architect, Data Engineering, Cox Automotive Inc, CA, USA

Abstract

Buying a used car can be a challenging experience. Like many other consumer goods, used car prices have risen rapidly in recent years. In addition, gasoline prices and rising interest rates have made the experience of owning a car even more painful. In this research, we propose an intelligent framework for estimating the cost of used cars using artificial neural network algorithms. The model was developed using a training dataset of 140,000 used vehicles from 30 popular US car brands. The model's predictions are validated against a test data set of 35,000 used cars. Numerous features are examined for reliable and accurate predictions. Artificial neural networks are built using the Keras regression algorithm, and their performance is compared to basic models such as linear regression, decision tree algorithms, gradient boosting, and random forests. Categorical variables were processed using embedding techniques to improve predictive performance. The results are consistent with actual values and significantly improved over the baseline model. Experimental results showed that an ANN model with a mean absolute percentage error of 11percent and an R2 value of 0.96 outperforms the random forest model with a MAPE of 14 percent and an R2 value of 0.94.

1. Introduction

Predicting used car prices is an exciting topic as more cars are being bought and sold than ever. According to Edmunds.com, an online car review site, about 40 million used cars are sold yearly in the United States. Buying a used car has its advantages[1]. On average, used cars are about 50% cheaper than new cars. Buyers can pay off used vehicles faster and save

on financing costs. Consumers often complain that new cars depreciate as soon as they leave the lot. A new car can lose 11% of its value on the way home. Vehicles lose value over weeks, months, and years. For used cars, most of the depreciation has already been incurred. Some used cars can even increase in value. With the increase in private vehicles and the promotion of the used car market, used cars are poised to become a priority for buyers.

In recent years, the growth of the Internet has dramatically increased the information available to potential buyers in multiple markets. Consumers now have access to a wide range of information about products, services, and alternatives to help them make purchasing decisions. Such data can help bridge the information gap between buyers and sellers and significantly reduce information asymmetry. For example, consumers and sellers typically negotiate the price of a car. However, consumers may need to be more consistent in their understanding and ability to deal effectively. Today, there are many sources of used car price information. Multiple price sources can confuse buyers and sellers, so developing a reliable method of predicting used car prices is crucial. In the automotive world, car valuation determines the market value of new and used cars. A car's appraisal varies based on the year, make, model, options, mileage, and rarity. The valuation also considers current market conditions and whether the market value of a particular vehicle is rising or falling. However, conventional evaluation methods make it challenging to select consistent indicators and rely too much on the subjective judgment of evaluators.

Artificial Neural Networks applications are becoming increasingly popular in various areas of human needs. Many organizations invest in neural networks to solve problems in multiple domains and business areas. This has traditionally been the responsibility of operations research. ANN is effective, efficient, and successful in providing advanced skills to deal with complex problems in many areas of life. ANN can solve problems in agriculture, science, medicine, education, finance, management, security, engineering, commodity trading, and art [2]–[5]. This study evaluates whether a neural network can accurately predict used car prices. The results are then compared with traditional regression techniques such as linear regression, decision trees, bagging, and boosting algorithms.

2. Related Works

Several studies and related work have been conducted to predict used car prices worldwide using various methods and approaches. In 2011, Gongqi et al.[6] used BP neural network and a nonlinear curve to fit the residual value of used private cars for various criteria such as make, mileage, and age. Monburinon et al.[7] performed a comparative study on regression performance based on supervised machine learning models using car market data collected from German e-commerce websites. Gradient-boosted regression trees showed the best performance, followed by random forest regression and multiple linear regression. Pal et al.[8] predicted used car prices using a random forest model. A random forest with 500 decision trees was constructed with a training accuracy of 95.82% and a testing accuracy of 83.63%. K. Samruddhi et al.[9] studied the used car market in India and proposed a model to estimate the cost of used cars using the K-Nearest Neighbor algorithm. KNN model achieved an accuracy of 85% compared to 71% with linear regression. Yu et al.[10] trained a BP neural network model to predict total car insurance claim amounts based on Shandong province, China, data. The results show that the prediction accuracy using the BP neural network model exceeds 95%. Cui et al. [11]developed used car price prediction based on the iterative framework of XGBoost and LightGBM. Purohit [12] established a correlation between the used car market and the new car market and found that the price of used cars decreases as the number of new cars increases, and its consumers are more inclined to buy models with slow depreciation.

3. Materials and Methods

3.1 Description of Data

The data used in this research project comes from the Kaggle dataset collection on used car sales. The original data set consisted of over 2 million vehicles. After cleaning and enrichment, 140,000 records were selected for training and 35,000 for testing. The dataset contains the following fields, as shown in Table 1.

Table 1. Data characteristics.

Sl	Feature	Description
1	Manufacturer	Car brand.
2	Model	The model of the car.
3	Model year	The sales year when the car model is officially launched and sales begin.
4	Condition	Indicate the condition of the car. Valid values are excellent, good, fair, and poor.
5	Title status	Vehicle titles contain a lot of important information regarding a vehicle's condition. Clear, clean, salvage, parts only, junk, etc., are valid values.
6	Fuel	The majority of motor vehicles worldwide are powered by gasoline or diesel. Electric and hybrid are the other fuel types.
7	Transmission	A medium that transmits power generated by the engine to the wheels via a mechanical system of gears and gear trains. Manual and automatic are valid values.
8	Cylinders	Chamber where fuel is combusted and power is generated. Most cars have a 4, 6, or 8-cylinder engine.
9	Drive	A car's drivetrain connects the engine to the wheels enabling the vehicle to move. There are four different types of drivetrains. All-wheel drive, four-wheel drive, front-wheel drive, and rear-wheel drive.
10	Size	Passenger car sizes are defined based on the combined passenger and cargo volume. Subcompact, compact, mid-size, and full-size are valid values.
11	Type	Indicates the body style. Basic types of cars ranging from coupes to full-size sedans to crossover SUVs.
12	Paint Color	The exterior color of the car
13	Vin WMI code	World Manufacturer Identifier derived from VIN information. The first digit of the VIN is the country of origin or final

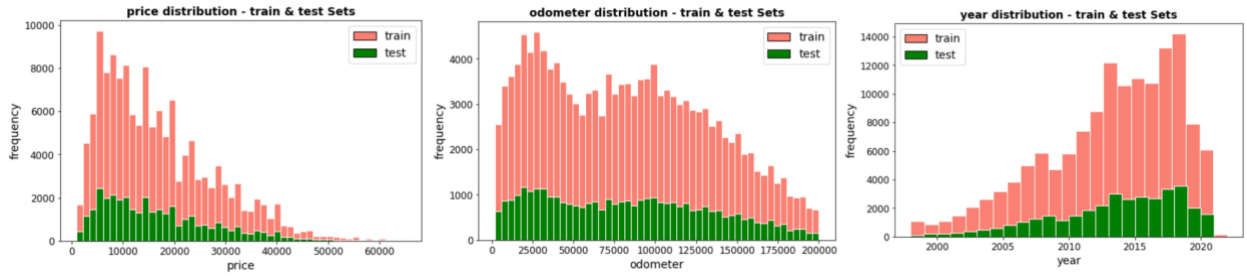
		processing plant. Next is the digit representing the manufacturer, and the third digit denotes the type of vehicle.
14	Vin VDS code	The vehicle descriptor section is derived from VIN information. Letters and numbers in the VDS provide information about the vehicle model, engine type, body style, and so forth
15	Region	The region where the car is being sold.
16	State	State where the car is sold.
17	Odometer	Distance traveled by a vehicle in miles
18	Price	Target variable. Selling price for the used car.

All major US car brands are part of the training and test sets. These include famous brands such as Toyota, Honda, Nissan, Chevrolet, Chrysler, Ford, BMW, and Mercedes. The distribution of brands on the test and train sets is shown in Figure 1.

Figure 1. Car Brands – Train and test data distribution.

Data exploration is integral to model building as it provides insight into the data and changes or modifications needed before designing the model. Histogram analysis was done for target price variables and numerical features such as mileage and year. All three of these variables exhibited skewed distributions. The price distribution was left-skewed on both the test and training data, with a peak of around 10,000 and a long tail up to 60,000 on the right. The odometer showed a bimodal distribution with two peaks at 25,000 and 100,000 miles. Model year values range from 1998 to 2021, are right-skewed, and peak after 2017. The distribution of numerical features is shown in Figure 2.

Figure 2. Numerical features – Train and test data distribution.



3.2 Modeling Approach

The primary goal of this study is to develop a deep-learning framework to predict car prices. We explored basic regression techniques like linear regression, tree-based regression, bagging, and boosting ensembles.

3.2.1 Base regression models

Linear regression:

Linear regression is undoubtedly one of the most frequently used statistical modeling methods. A linear regression line has an equation of the following form.

$$y = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \cdots + \beta_n * x_n + \varepsilon$$

In this formula, x is the explanatory variable, and y is the dependent variable. There should be a linear relationship between each explanatory variable and the output variable. For the regression model to be robust and to explain y as well as possible, it should include only independent variables that explain a large portion of the variance in y. The model equation produces a best-fit straight line that can be used to estimate the response based on the values of the predictors. The most common way to determine the best model is to choose the model that minimizes the squared difference between the model's actual and estimated values.

Tree based - Decision tree regression:

A decision tree builds a regression model in a tree structure. First, divide the dataset into smaller subsets while making relevant decision trees step-by-step. The splitting process starts with a root node, followed by a branching tree, and finally, a leaf node containing the predictions or results of the algorithm. The main components of a decision tree model are nodes and branches, and the most critical steps in building a model are splitting, stopping, and pruning [13]. Trees are constructed typically top-down, choosing the variable that best splits a set of elements at each step. Decision tree learning uses a divide-and-conquer strategy by performing a greedy search to identify the optimal split points in the tree. This splitting process is repeated from top to bottom until all or most records have been processed. Finally, pruning is typically used to remove branches split into less essential features to reduce complexity and prevent overfitting.

Boosting ensemble - Gradient boosting regression:

The goal of an ensemble algorithm is to combine the predictions of multiple basis estimators produced by a particular learning algorithm to improve the robustness of a single estimator. Gradient boosting is an ensemble technique that combines numerous weak models to improve overall performance. Boosting is sequential, in which each model attempts to correct the errors of the prior model, focusing on bias reduction. Ensembles are typically built from decision tree models. Trees are sequentially added one at a time and adjusted to correct prediction errors from previous models. Gradient boosting involves a loss function, weak learners that make predictions, and an additive model that combines these weak learners to minimize the loss function. LightGBM extends the gradient boosting algorithm by adding automatic function selection and focusing on boosting samples with more significant gradients. This significantly speeds up training and improves prediction performance. For example, LightGBM could speed up the training process of conventional GBDT by up to over 20 times while achieving almost the same accuracy [14].

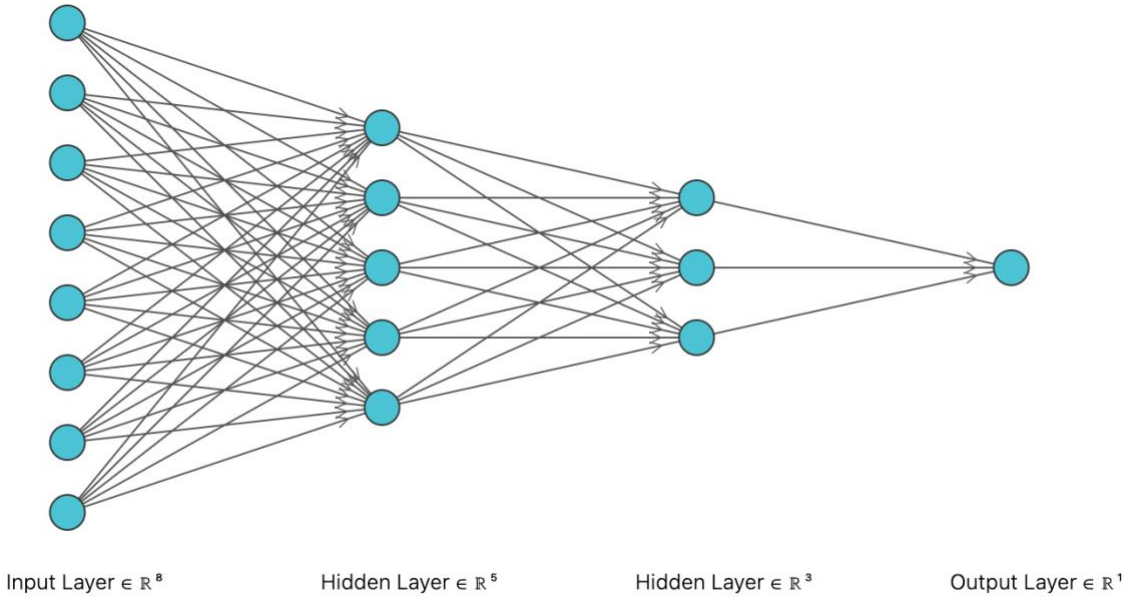
Bagging ensemble - Random Forest regression.

Bagging is an ensemble algorithm that combines multiple models with different subsets of the training dataset and combines predictions from all models. Bootstrapping is the first step in the bagging process flow, splitting the data into random samples. A regression algorithm is then applied to each sample, averaging all outcomes predicted by each learner. Bagging is especially effective when the learner is unstable and tends to over-fit. Random forest is an extension of the bagging algorithm in which only a subset of features is randomly selected from the whole set. The best split feature identified from the subset is used to split each node in the tree, as opposed to bagging, where all features are considered. Random forest algorithms combine ensemble learning methods with decision tree frameworks to build multiple randomly drawn decision trees from the data, then average the results to final output predictions. Each decision tree is fully trained on the selected dataset. Therefore, the output depends on multiple decision trees instead of a single decision tree, often leading to better predictions.

3.2.2 Artificial Neural Networks

ANNs are based on a collection of nodes called artificial neurons, which loosely model neurons in the biological brain[15]–[18]. Each connection can send signals to other neurons, like synapses in the biological brain. Artificial neurons can receive and process signals and send signals to connected neurons. The signals in the connection are real numbers, and a nonlinear function of the inputs computes each neuron's output. Connections are called edges. Neurons and edges typically have weights that adjust as learning progresses. Weights increase or decrease the signal strength of a link. A neuron can have a threshold, so a signal is only sent if the total signal exceeds that threshold. ANN consists of the following components, as given in Figure 3.

Figure 3. Artificial Neural Network components.



Input layer: Information from the outside world enters the artificial neural network from the input layer. Input nodes process the data, analyze or categorize it, and pass it on to the next layer. The input layer has as many neurons as there are sample features we want the network to learn.

Hidden layers get input from input layers or other hidden layers. Artificial neural networks can have many hidden layers. Each hidden layer analyzes the previous layer's output, processes it further, and passes it on to the next layer. In neural networks, hidden layers are determined according to the solution of the problem. In other words, there is no specific rule for the number of hidden layers, which varies based on requirements.

Output layer: The output layer is the final layer which outputs predictions. It can have one or more nodes. The output layer performs computations that classify or characterize the information from the input layer.

Activation functions: The activation function determines whether to activate a neuron by computing a weighted sum and adding another bias. The purpose of this is to introduce nonlinearity in the neuron's output. Commonly used activation functions in artificial neural networks are Step, Linear, Sigmoid, Hyperbolic Tangent, Rectified Linear Unit (ReLU), Swish, and Softmax.

In this study, we used a feedforward neural network. Feedforward neural networks are one of the simplest variants of neural networks. It routes the information in one direction through various input nodes until it reaches an output node. This network may or may not have a hidden node layer, making its operation more interpretable. Categorical embedding was used to handle variables for the car model, region, WMI code, and VDS code. Embedding is the encoding of categorical information properties into a fixed-length numeric format. One hot coding is used for all other categorical variables. The model network is shown in Table 2 and Figure 4.

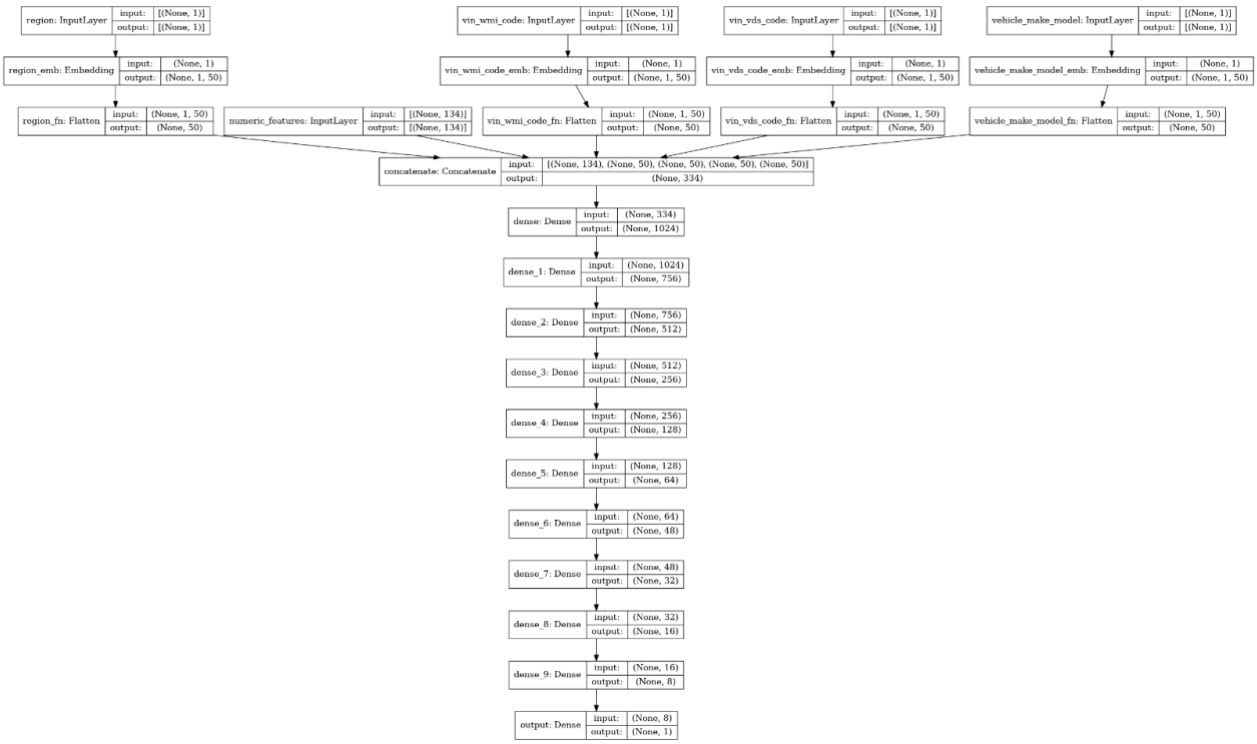
Table 2. ANN model summary

Layer Name	Layer Type	Output Shape	Parameters	Connected to
region	InputLayer	(None, 1)	0	
vin_wmi_code	InputLayer	(None, 1)	0	
vin_vds_code	InputLayer	(None, 1)	0	
vehicle_make_model	InputLayer	(None, 1)	0	
region_emb	Embedding	(None, 1, 50)	20200	region[0][0]
vin_wmi_code_emb	Embedding	(None, 1, 50)	16400	vin_wmi_code[0][0]
vin_vds_code_emb	Embedding	(None, 1, 50)	369450	vin_vds_code[0][0]
vehicle_make_model_emb	Embedding	(None, 1, 50)	639900	vehicle_make_model[0][0]

numeric_features	InputLayer	(None, 134)	0	
region_fn	Flatten	(None, 50)	0	region_emb[0][0]
vin_wmi_code_fn	Flatten	(None, 50)	0	vin_wmi_code_emb[0][0]
vin_vds_code_fn	Flatten	(None, 50)	0	vin_vds_code_emb[0][0]
vehicle_make_model_fn	Flatten	(None, 50)	0	vehicle_make_model_emb[0][0]
concatenate	Concatenate	(None, 334)	0	numeric_features[0][0]
dense	Dense	(None, 1024)	343040	concatenate [0][0]
dense_1	Dense	(None, 756)	774900	dense [0][0]
dense_2	Dense	(None, 512)	387584	dense_1[0][0]
dense_3	Dense	(None, 256)	131328	dense_2[0][0]
dense_4	Dense	(None, 128)	32896	dense_3[0][0]
dense_5	Dense	(None, 64)	8256	dense_4[0][0]
dense_6	Dense	(None, 48)	3120	dense_5[0][0]
dense_7	Dense	(None, 32)	1568	dense_6[0][0]
dense_8	Dense	(None, 16)	528	dense_7[0][0]

dense_9	Dense	(None, 8)	136	dense_8[0][0]
output	Dense	(None, 1)	9	dense_9[0][0]

Figure 4. Artificial Neural Network layers



3.3 **Evaluation Metrics**

The regression models were evaluated using the following metrics.

Mean Absolute Error

Mean Absolute Error gives the average absolute difference between the model prediction and the target value. MAE measures the average of the errors in a set of predictions without regard to direction. The following formula gives MAE: y is the prediction, x is the actual value, and n is the number of data points.

$$\text{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

Mean Absolute Percentage Error

Mean Absolute Percentage Error is the average of the absolute percentage errors of the predictions. To compute MAPE, the percent errors are summed regardless of sign. This scale is easy to understand as it shows the error as a percentage. The following formula gives MAPE: A is the actual value, F is the forecast, and n is the number of data points.

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n} \frac{A_t - F_t}{A_t}$$

Root Mean Squared Error

RMSE is the standard deviation of the residual forecast errors. Residuals measure how far the data points on the regression line are from the actual value. RMSE is a measure of how these residuals are distributed. In other words, it shows how much the data is clustered around the line of best fit. RMSE can be calculated using the following formula.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N}}$$

R Squared

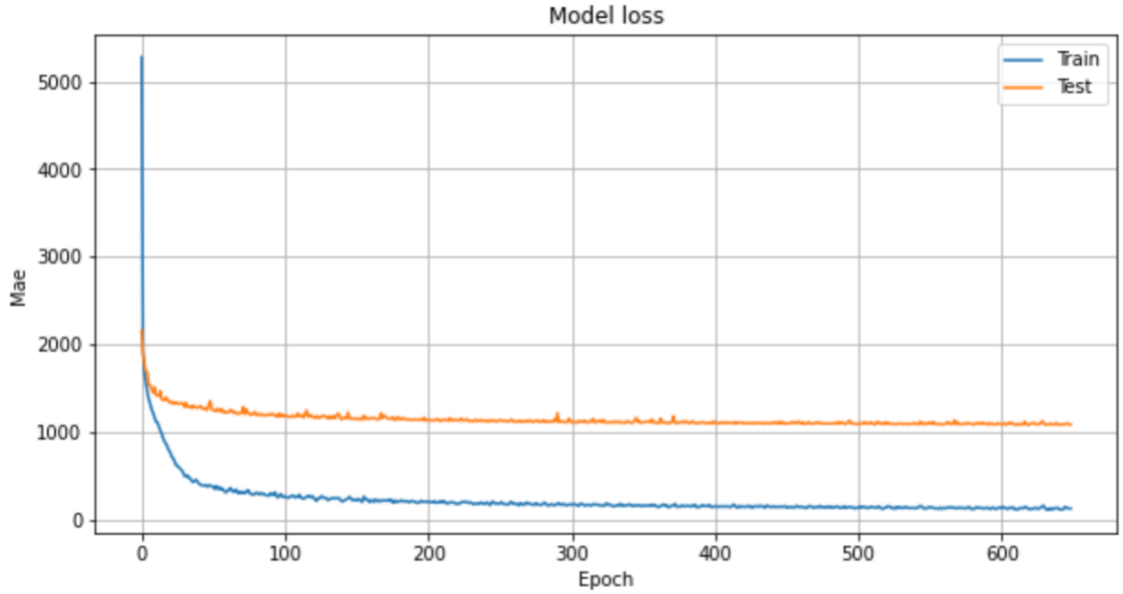
The coefficient of determination, commonly known as R Squared, is a statistical measure representing the amount of the variance of a dependent variable explained by a set of independent variables in a regression model. R-squared is a goodness-of-fit measure that describes how much variance in one variable explains variance in a second variable. R squared can be calculated using the following formula: RSS is the sum of squares of residuals, and TSS is the total sum of squares.

$$R^2 = 1 - \frac{RSS}{TSS}$$

4. Results

We randomly split the training data and allocated 20% as validation data. The base models were trained with default hyperparameters. The neural network has been trained up to 1000 epochs. Early stopping was achieved at 650 epochs, as given in Figure 5. MAE and MSE were used to determine early stop criteria.

Figure 5. Model training history plot.



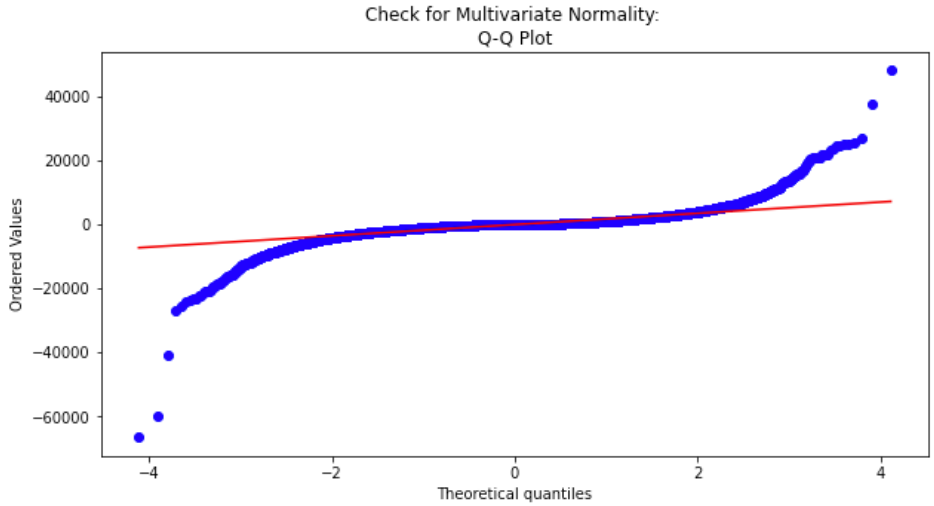
To evaluate the model's performance, we used unseen test data to predict used car prices. The test data consists of 35,000 records across 30 popular brands. Test results are given below in Table 3.

Table 3. Test results and performance metrics.

Model Name	MAE	MAPE	RMSE	R2
Neural Networks	1060.37	0.11	2104.13	0.9634
RandomForest Regressor	1382.87	0.14	2717.79	0.9389
Lightgbm	2437.12	0.21	3671.58	0.8885
Decision Tree	1703.58	0.17	3767.4	0.8826
Linear Regression	4005.88	0.38	5593.14	0.7412

Figure 6 shows the Q-Q plot, a plot used to test the normality of residual values. A quantile-quantile plot is a graphical technique for determining whether two data sets come from populations that share a common distribution. If the two distributions we are comparing are the same, the points on the plot will be perfectly straight.

Figure 6. Check for Normality Q-Q plot.



The predicted vs. actual scatter plot is given in figure 7. The predicted versus actual plot shows the effect of the model and compares it to the null model. For a good fit, points should be close to the fitted line, with narrow confidence interval bands.

Figure 7. Check for Linearity - Actual vs. Predictions.

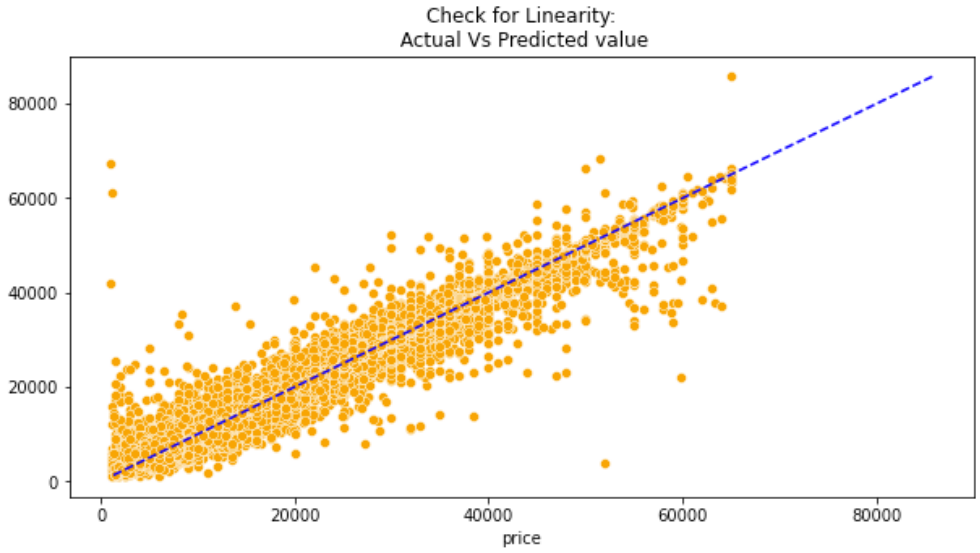
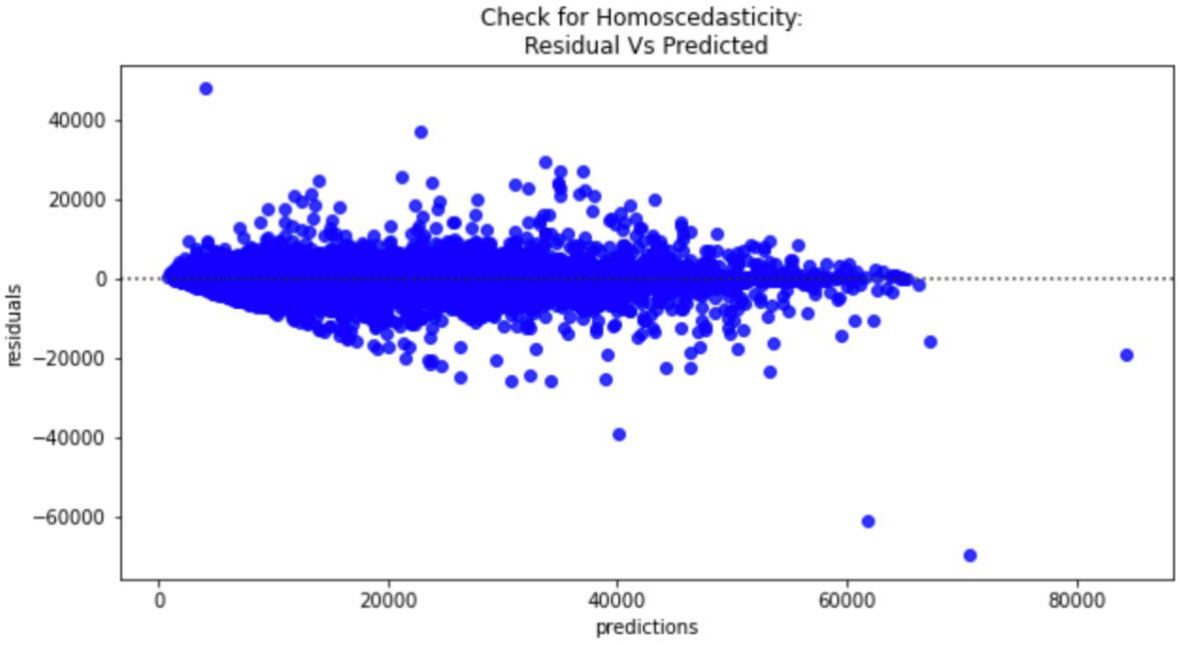


Figure 8 checks for homoscedasticity with a scatterplot of the residuals against the dependent variable. Homogeneous variances of the model mean that the error is constant along the values of the dependent variable. For the data set to have homoscedasticity, the points should be approximately the same distance from the line.

Figure 8. Check for Homoscedasticity - Residuals vs. Predictions.



5. Discussion

This research aimed to develop an artificial neural network model for predicting used car prices. Various regression models were created, and the results were compared with the ANN model. According to Table 3, the ANN model achieved the best performance with MAE of 1060, MAPE of 11%, RMSE of 2104, and R-square of 0.96. The Random Forest Regressor performed best among the baseline models with an MAE of 1382 and a MAPE of 14%. Linear regression showed the worst performance with an MAE of 4005 and a MAPE of 38%.

The ANN model performs well based on the evaluation metrics. An R Squared value close to 1 indicates that the model explains most of the variance in the outcome variable. The RMSE of the ANN model is significantly smaller than the other models, which also indicates the superior performance of his ANN model. Except for some outliers, the actual versus prediction plot and the Q-Q plot shows an excellent fit to the ANN model.

6. Conclusions

Online used car trading platforms have developed rapidly in recent years, but they still need help. In practice, institutions and people differ in how they check used car price parameters and how they predict used car prices. Under such circumstances, the market is likely to develop randomly, it is difficult to establish a unified rating system, and the sale of used cars will be complicated. In theory, classical used car appraisal methods rely too much on the subjective judgment of appraisers who can no longer meet the demands of online transactions in the used car market. Therefore, it is necessary to establish an efficient, rational, fair, and accurate used car price appraisal system. This paper aims to tackle the complex problem of predicting used car prices. We examined the performance of various regression algorithms. Artificial neural network algorithms outperform other models, such as random forests and gradient-boosted models. The ANN model was a good fit, achieving an R-squared of 0.96 and a MAPE of 11%.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

Data Availability

The dataset used for this study is publicly available.

https://github.com/aravindsp/Machine_Learning_DataSets/tree/main/ml_used_car_price_data

REFERENCES

[1] J. Moody, E. Farr, M. Papagelis, and D. R. Keith, "The value of car ownership and use in the United States," *Nat Sustain*, vol. 4, no. 9, pp. 769–774, Jun. 2021, doi: 10.1038/s41893-021-00731-5.

[2] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, and H. Arshad, "State-of-the-art in artificial neural network applications: A survey," *Heliyon*, vol. 4, no. 11, p. e00938, Nov. 2018, doi: 10.1016/j.heliyon.2018.e00938.

[3] K. Li, X. Xie, W. Xue, X. Dai, X. Chen, and X. Yang, "A hybrid teaching-learning artificial neural network for building electrical energy consumption prediction," *Energy Build*, vol. 174, pp. 323–334, Sep. 2018, doi: 10.1016/j.enbuild.2018.06.017.

[4] A. S. Pillai, "Cardiac disease prediction with tabular neural network." 2022. doi: 10.13140/RG.2.2.29633.22883.

[5] F. Amato, A. López, E. M. Peña-Méndez, P. Vaňhara, A. Hampl, and J. Havel, "Artificial neural networks in medical diagnosis," *J Appl Biomed*, vol. 11, no. 2, pp. 47–58, Jul. 2013, doi: 10.2478/v10136-012-0031-x.

[6] Shen Gongqi, Wang Yansong, and Zhu Qiang, "New Model for Residual Value Prediction of the Used Car Based on BP Neural Network and Nonlinear Curve Fit," in

2011 Third International Conference on Measuring Technology and Mechatronics Automation, Jan. 2011, pp. 682–685. doi: 10.1109/ICMTMA.2011.455.

[7] N. Monburinon, P. Chertchom, T. Kaewkiriya, S. Rungpheung, S. Buya, and P. Boonpou, "Prediction of prices for used car by using regression models," in *2018 5th International Conference on Business and Industrial Research (ICBIR)*, May 2018, pp. 115–119. doi: 10.1109/ICBIR.2018.8391177.

[8] N. Pal, P. Arora, P. Kohli, D. Sundararaman, and S. S. Palakurthy, "How Much Is My Car Worth? A Methodology for Predicting Used Cars' Prices Using Random Forest," 2019, pp. 413–422. doi: 10.1007/978-3-030-03402-3_28.

[9] K. Samruddhi and R. Ashok Kumar, "Used Car Price Prediction using K-Nearest Neighbor Based Model," *International Journal of Innovative Research in Applied Sciences and Engineering*, vol. 4, no. 3, pp. 686–689, Sep. 2020, doi: 10.29027/IJIRASE.v4.i3.2020.686-689.

[10] W. Yu *et al.*, "Claim Amount Forecasting and Pricing of Automobile Insurance Based on the BP Neural Network," *complexity*, vol. 2021, pp. 1–17, Jan. 2021, doi: 10.1155/2021/6616121.

[11] 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," *Electronics (Basel)*, vol. 11, no. 18, p. 2932, Sep. 2022, doi: 10.3390/electronics11182932.

[12] D. Purohit, "Exploring the Relationship Between the Markets for New and Used Durable Goods: The Case of Automobiles," *Marketing Science*, vol. 11, no. 2, pp. 154–167, May 1992, doi: 10.1287/mksc.11.2.154.

[13] Y.-Y. Song and Y. Lu, "Decision tree methods: applications for classification and prediction.," *Shanghai Arch Psychiatry*, vol. 27, no. 2, pp. 130–5, Apr. 2015, doi: 10.11919/j.issn.1002-0829.215044.

[14] G. Ke *et al.*, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Advances in Neural Information Processing Systems 30 (NIP 2017)*, Dec. 2017. [Online].

Available: https://www.microsoft.com/en-us/research/publication/lightgbm-a-highly-efficient-gradient-boosting-decision-tree/

[15] S. Walczak and N. Cerpa, "Artificial Neural Networks," in *Encyclopedia of Physical Science and Technology*, Elsevier, 2003, pp. 631–645. doi: 10.1016/B0-12-227410-5/00837-1.

[16] A. S. Pillai, "Multi-Label Chest X-Ray Classification via Deep Learning," *Journal of Intelligent Learning Systems and Applications*, vol. 14, no. 04, pp. 43–56, 2022, doi: 10.4236/jilsa.2022.144004.

[17] I. Kononenko and M. Kukar, "Artificial Neural Networks," in *Machine Learning and Data Mining*, Elsevier, 2007, pp. 275–320. doi: 10.1533/9780857099440.275.

[18] S. Sieniutycz, "Complex systems of neural networks," in *Complexity and Complex Thermo-Economic Systems*, Elsevier, 2020, pp. 51–84. doi: 10.1016/B978-0-12-818594-0.00004-0.