A Comparative Study of Machine Learning Models for Predicting Used Car Prices

Muktadirul Islam Nibir
Department of Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
Email: muktadirul.islam.nibir@g.bracu.ac.bd

***Abstract*—This paper presents a comprehensive analysis of multiple machine learning models used to predict used car prices based on vehicle attributes. We trained, tested, and evaluated Random Forest, Decision Tree, and Neural Network models on a cleaned and preprocessed dataset sourced from Kaggle. The performance of these models was compared using standard regression metrics. Our analysis reveals that the Random Forest model provides the best performance in terms of accuracy and reliability for this regression task.**

***Index Terms*—Car price prediction, machine learning, regression, model evaluation, feature engineering.**

I. Introduction

Machine learning has become an integral part of modern predictive analytics, particularly in sectors like automotive, where pricing strategies significantly impact revenue. Predicting used car prices is a valuable task that can benefit dealerships, sellers, buyers, and digital marketplaces. The complexity of the task arises from the interplay of various vehicle features, such as make, model, year, mileage, and condition, which jointly influence the final selling price. This paper explores and compares multiple machine learning approaches for solving this problem, aiming to identify which algorithm performs best for this real-world regression challenge.

II. Dataset Description

The dataset used for this study was sourced from Kaggle's "Used Car Auction Prices" dataset. It initially contained over 550,000 rows, representing a wide array of vehicles listed in online auctions. However, after filtering out incomplete, noisy, and redundant records, the dataset was reduced to 70,000 entries, and finally down to 68,314 after full preprocessing.

The dataset includes a total of 16 features, composed of both categorical and numerical attributes that influence vehicle pricing. These features can be grouped as follows:

- **Vehicle Identity:** Includes *year*, *make*, *model*, and *trim*, which describe the basic identity and configuration of the car.
- **Specifications:** Attributes such as *body type*, *transmission*, and *condition* provide details about the physical characteristics and operational state of the vehicle.
- **Usage Metrics:** The *odometer* reading gives an indication of how much the car has been used, which significantly affects depreciation and resale value.
- **Market Valuation:** The *mmr* (Manheim Market Report) price gives a standardized wholesale valuation of the car, serving as a benchmark in pricing predictions.

The target variable, *sellingprice*, is continuous in nature and represents the actual transaction price at which the car was sold. This makes the problem suitable for regression-based machine learning models.

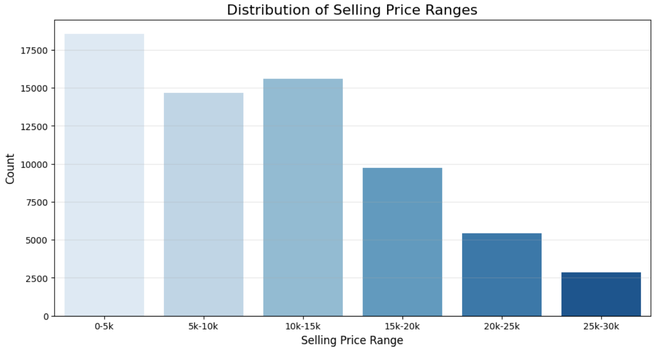

Fig. 1. Barchart showing the distribution of selling price ranges and the corresponding frequency of cars in each range.

The selling price shows a heavily right-skewed distribution, with a large number of cars priced in the low-to-mid range, and a long tail of high-value luxury vehicles. Understanding this distribution is critical for evaluating the models' performance across different price segments.

The correlation heatmap reveals interesting relationships among variables. Notably, *mmr* shows a strong positive correlation with *sellingprice*, while *odometer* has a moderate negative correlation. These insights guide both model feature selection and interpretation.

III. Preprocessing and Feature Engineering

Data preprocessing was crucial to improving model accuracy and ensuring fair comparisons. This stage involved several steps:

- **Missing Value Handling:** Missing numerical values were imputed using column-wise means. Categorical features with missing entries were filled with the mode or labeled as "Unknown".

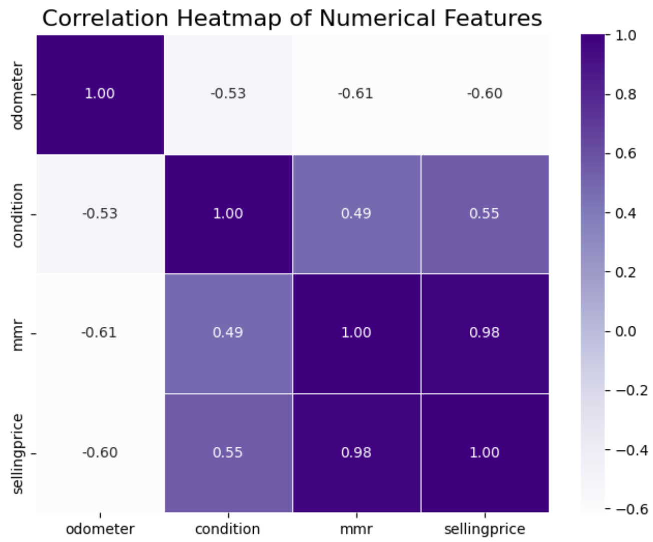

Fig. 2. Heatmap showing correlation among numerical features like condition, odometer, mmr, and selling price.

- **Outlier Detection and Removal:** Z-score based filtering was applied to remove data points with extreme values in *odometer* and *sellingprice*, helping stabilize the training process.

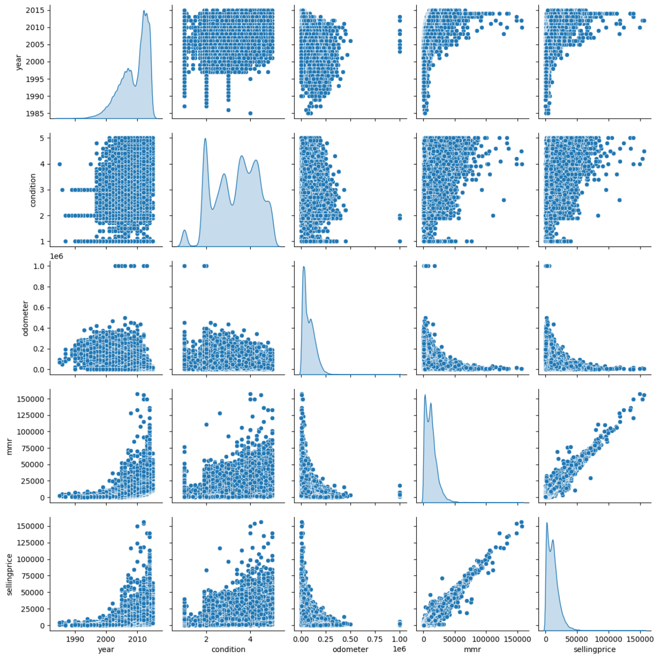

Fig. 3. The seaborn pairplot shows the correlation of variables before feature scaling.

- **Feature Scaling:** StandardScaler was used to normalize the values of numeric attributes such as *odometer*, *condition*, and *mmr*. This step is vital for algorithms sensitive to feature magnitudes, such as Neural Networks and KNN.

IV. Model Training and Evaluation

We split the dataset using a 70/30 train-test ratio to ensure a balanced evaluation. The following models were implemented and trained:

- **Random Forest:** Random Forest is an ensemble learning algorithm that constructs a multitude of decision trees during training and outputs the mean prediction of the individual trees for regression tasks. This model excels in handling non-linear relationships and avoids overfitting by averaging multiple models. It leverages bootstrapping and feature randomness to create diverse trees, making it highly robust to noise and capable of capturing complex patterns in tabular data. In this project, Random Forest demonstrated superior generalization performance and the highest prediction accuracy.
- **Decision Tree:** A Decision Tree is a flowchart-like structure where each internal node represents a test on a feature, each branch denotes the result of the test, and each leaf node holds a prediction. While easy to interpret and fast to train, decision trees tend to overfit the training data, especially when not pruned or regularized. In our experiment, the Decision Tree served as a useful baseline model, but it showed lower prediction accuracy and higher variance compared to the ensemble-based Random Forest.
- **Neural Network:** The Neural Network implemented in this study was a feedforward architecture built using TensorFlow/Keras. It consisted of multiple densely connected layers capable of learning complex, non-linear interactions among features. We used ReLU activations and optimized the model with backpropagation and an adaptive optimizer. While neural networks are powerful, their performance is heavily dependent on proper tuning, network depth, and data preprocessing. The model achieved competitive accuracy but exhibited slight inconsistency due to sensitivity to initial weights and hyperparameters.

The models were evaluated using Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R^2).

TABLE I
Model Performance Comparison

Model	MAE	MSE	RMSE	R^2 Score
Random Forest	974.75	2,343,037.62	1530.70	96.47%
Decision Tree	1050.44	3,070,751.01	1752.36	95.38%
Neural Network	1039.63	2,660,694.13	1631.16	96.01%

V. Results and Discussion

Our analysis shows that the Random Forest model outperforms the others, achieving the highest R^2 score of 96.47% and the lowest RMSE of 1530.70. This indicates that the model was able to explain a very high proportion of the variance in car prices, while maintaining minimal deviation from the actual values. The ensemble nature of Random Forest—by aggregating multiple decision trees—helps reduce overfitting and improves generalization, making it robust across various data splits and sample distributions.

The Neural Network also performed well, achieving an R^2 score of 96.01% and an RMSE of 1631.16. While slightly less accurate than the Random Forest model, the Neural Network showed potential for capturing complex, non-linear relationships within the data. However, it was found to be more sensitive to hyperparameters such as the number of hidden layers, learning rate, and batch size. Additionally, training time and resource consumption were higher for this model, which may be a consideration in production environments.

The Decision Tree model, while offering the advantage of clear interpretability, lagged behind in predictive performance. It recorded the highest RMSE of 1752.36 and the lowest R^2 score of 95.38%. This can be attributed to its tendency to overfit on training data without the regularization benefits of ensemble methods. Nevertheless, its simplicity and speed of training make it a viable option for applications where quick interpretability is essential and minor performance trade-offs are acceptable.

The metrics used in Table I—Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R^2 Score—collectively provide a comprehensive view of model performance. While MAE provides a straightforward interpretation of average error magnitude, RMSE penalizes larger errors more strongly, and the R^2 Score gives an overall measure of goodness-of-fit. Across all these indicators, Random Forest consistently emerged as the top-performing model.

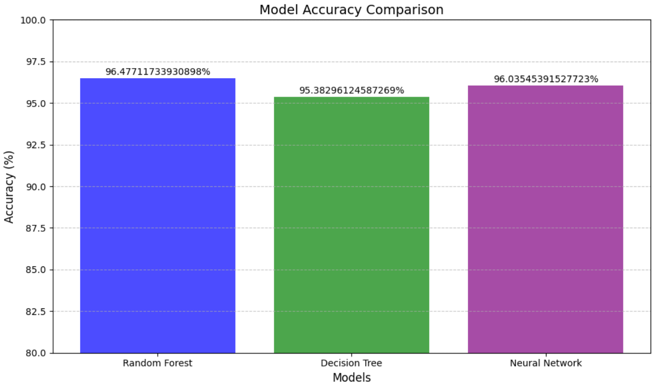

Fig. 4. Bar chart comparing R2 scores or the Accuaracy of the three models.

Figure 4 visually reinforces these findings, with a bar chart that compares the R^2 scores (or predictive accuracies) of the three models. The clear lead held by Random Forest further supports its efficacy for this task.

This demonstrates the trade-offs between simplicity, interpretability, and predictive power. Random Forest provides an excellent balance, making it well-suited for deployment in real-world applications. Its ability to handle high-dimensional data, combined with its resilience to overfitting and superior predictive accuracy, makes it a strong candidate for any regression task involving structured datasets.

VI. Conclusion and Future Work

In this paper, we conducted a comprehensive comparative analysis of three machine learning models—Random Forest, Decision Tree, and Neural Network—for predicting used car prices based on multiple vehicle attributes. Our findings reveal that the Random Forest model consistently outperforms the others in terms of accuracy, stability, and generalization capability, achieving the lowest prediction error and the highest R^2 score.

This study illustrates the potential of ensemble learning techniques for real-world regression problems involving structured data. By incorporating vehicle specifications, market valuation benchmarks like MMR, and usage metrics such as odometer readings, our models were able to capture essential trends and relationships governing car prices. The preprocessing pipeline, which included handling missing values, outlier removal, and feature scaling, proved critical in enhancing model performance.

For future work, several promising directions can be explored:

- **Advanced Models:** Incorporating more sophisticated algorithms such as XGBoost, LightGBM, or CatBoost, which often outperform traditional ensemble methods in regression tasks.
- **Hyperparameter Optimization:** Applying automated tuning techniques like Grid Search or Bayesian Optimization to fine-tune neural network architectures and Random Forest parameters for even better performance.
- **Model Interpretability:** Leveraging SHAP (SHapley Additive exPlanations) or LIME (Local Interpretable Model-Agnostic Explanations) to better understand feature importance and model decisions.
- **Integration of External Data:** Enhancing the feature space by incorporating external market indicators, fuel price trends, geographic location data, or seasonal effects to improve predictive power.
- **Deployment and Monitoring:** Building a lightweight, web-based deployment pipeline that integrates the trained model with a user-friendly interface for real-time price prediction and monitoring model drift over time.

In conclusion, machine learning holds significant promise for transforming how used car prices are estimated, and this study contributes a strong foundation for continued exploration in this domain.

Acknowledgments

Thanks to BRAC University for the guidance provided through out the semester.