Predicting Automobile Prices Using Multi-model Machine Learning

Md. Faisal Iftekhar
Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
md.faisal.iftekhar@g.bracu.ac.bd

***Abstract*—This report presents a comparative analysis of four machine learning regression models for predicting the price of automobiles. The models—Decision Tree Regressor, Random Forest Regressor, K-Neighbors Regressor, and Support Vector Regressor (SVR)—are trained and evaluated on the Automobile dataset from the UCI Machine Learning Repository. The study involves data preprocessing, exploratory data analysis, model training, and evaluation using Root Mean Squared Error (RMSE), R-squared (R^2), and Mean Absolute Percentage Error (MAPE) metrics. The results indicate that the Random Forest Regressor provides the best performance in predicting automobile prices among the models tested.**

***Index Terms*—machine learning, regression, automobile price prediction, data analysis, predictive modeling**

I. Introduction

The automobile industry is highly competitive, and pricing is a critical factor for both consumers and manufacturers. Predicting car prices accurately can help in various business decisions, including market analysis, used car valuation, and setting competitive prices for new models. Machine learning offers a powerful set of tools for building predictive models from historical data. This project aims to develop and evaluate several machine learning models to predict the price of automobiles based on their various features.

The overall objectives of this study are:

- To perform exploratory data analysis to understand the relationships between different features and the car price.
- To preprocess and prepare the Automobile dataset for modeling.
- To implement and train four different regression models: Decision Tree, Random Forest, K-Nearest Neighbors, and Support Vector Regression.
- To evaluate and compare the performance of these models using appropriate metrics.

II. Problem Statement

The problem addressed in this report is the prediction of automobile prices using a set of features. The goal is to build a regression model that can accurately estimate the price of a car given its specifications. This is an important problem because it can provide valuable insights for pricing strategies and help consumers make informed decisions. The scope of this project is limited to the dataset provided, and the focus is on comparing the performance of four popular regression algorithms.

III. Data Description

The dataset used in this project is the **Automobile Data Set** from the UCI Machine Learning Repository [1]. This dataset contains 205 instances with 25 attributes, including the target variable ”price”. The features include both categorical and numerical data, describing various aspects of a car such as make, model, engine size, horsepower, and fuel efficiency.

IV. Exploratory Data Analysis (EDA)

EDA was performed to gain insights into the data. Few attributes of the dataset had multiple missing values, that can be seen in the missing data heatmap.

Fig. 1. Missing Data Heatmap.

A correlation matrix of the numerical features was generated to understand the linear relationships between them.

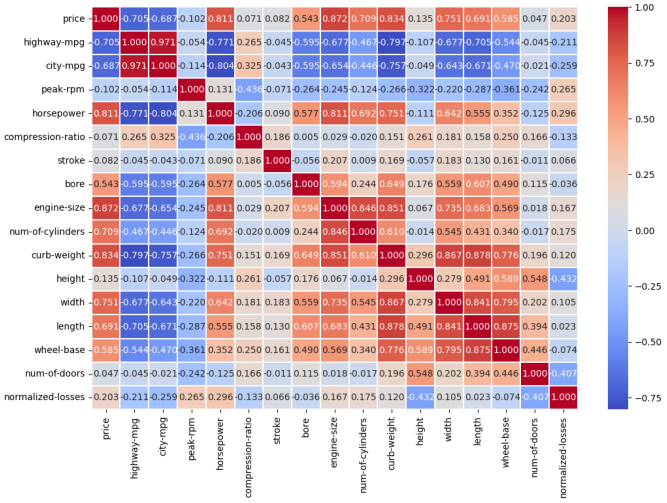

Fig. 2. Correlation Matrix of Numerical Features.

The distribution of the target variable 'price' was also visualized using a histogram and a boxplot, which showed a right-skewed distribution, indicating that most cars have a lower price, with a few being significantly more expensive.

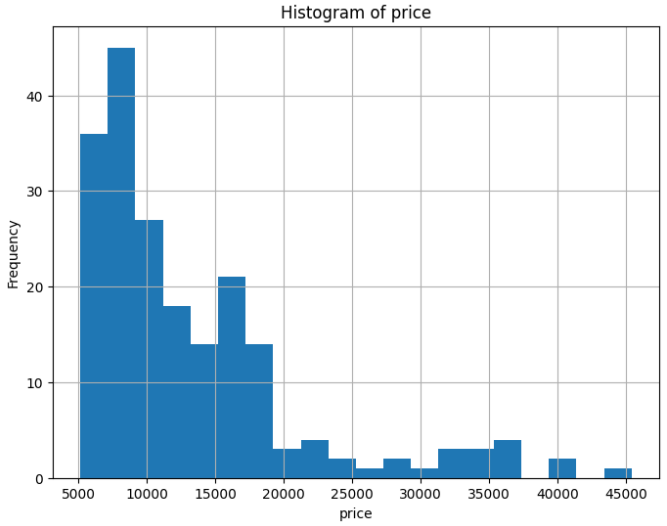

Fig. 3. Histogram of Price.

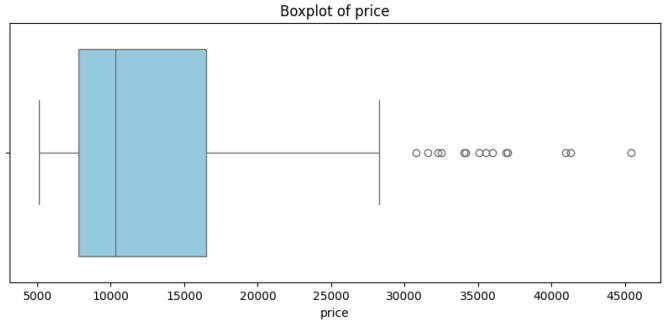

Fig. 4. Boxplot of Price.

The density plots of the numerical features were also explored.

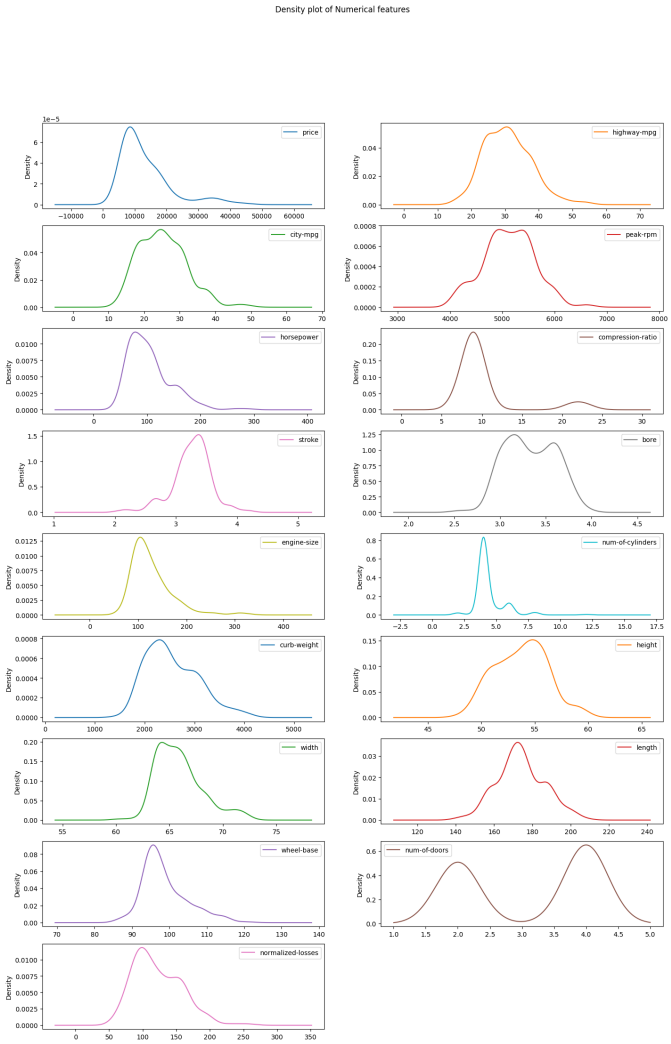

Fig. 5. Density Plot of Numerical.

V. Preprocessing

The raw dataset contained missing values, which were handled by filling numerical columns with a combination of means and modes. Entries without a price attribute were dropped. Categorical features did not have any missing values. They were converted into numerical format using one-hot encoding.

VI. Methodology

A. Feature Engineering and Selection

After handling missing values and encoding categorical variables, the features were scaled using the 'StandardScaler' from scikit-learn [2]. This ensures that all features have a mean of 0 and a standard deviation of 1, which is important for algorithms like SVR and KNN.

B. Model Selection

Four regression models were selected for this study:

- **Decision Tree Regressor:** A non-parametric supervised learning method used for regression tasks. It is simple to understand and interpret. [3]
- **Random Forest Regressor:** An ensemble learning method that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. [4]
- **K-Neighbors Regressor (KNN):** A non-parametric method that predicts the target value by averaging the values of its 'k' nearest neighbors. [5]
- **Support Vector Regressor (SVR):** A supervised learning model that uses support vector machines for regression. It is effective in high-dimensional spaces. [6]

C. Training Setup

The dataset was split into training and testing sets, with 80% of the data used for training the models and 20% for testing.

VII. Evaluation Metrics

The performance of the models was evaluated using the following metrics:

- **Mean Squared Error (RMSE):** Measures the root average of the squares of the errors, that is, the root of average squared difference between the estimated values and the actual value. [7]

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{1}$$

- **R-squared (R^2):** A statistical measure of how close the data are to the fitted regression line. It is also known as the coefficient of determination. [8]

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

- **Mean Absolute Percentage Error (MAPE):** Measures the average of the absolute percentage errors between predicted and actual values, that is, the average of the absolute differences between forecasted and actual values, expressed as a percentage of the actual values. [9]

$$\text{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{3}$$

These metrics are commonly used for evaluating regression models and provide a good measure of their predictive accuracy.

VIII. Results

The four models were trained on the training data and evaluated on the test data. The performance of each model is summarized in Table I.

TABLE I
Model Performance

Model	RMSE	R^2	MAPE
Decision Tree Regressor	2844.6545	0.9339	10.9909%
Random Forest Regressor	2720.7662	0.9395	9.8057%
K-Neighbors Regressor	6533.2159	0.6511	18.2942%
Support Vector Regressor	12216.4871	-0.2198	41.1640%

The Random Forest Regressor achieved the lowest RMSE and MAPE, and the highest R^2 score, indicating that it was the best-performing model for this dataset. The results are also visualized in Fig. 6, Fig. 7, and Fig. 8.

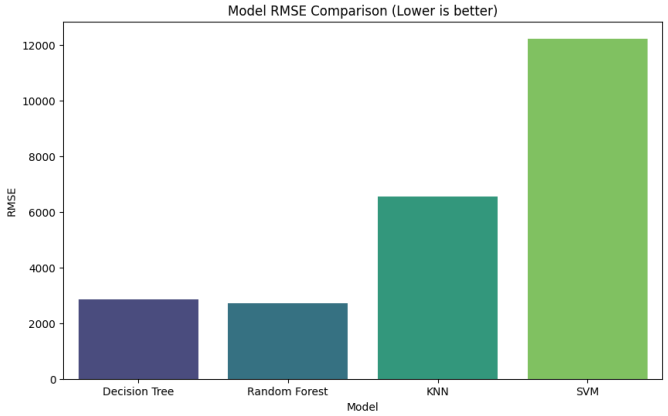

Fig. 6. Comparison of RMSE Scores of the Models.

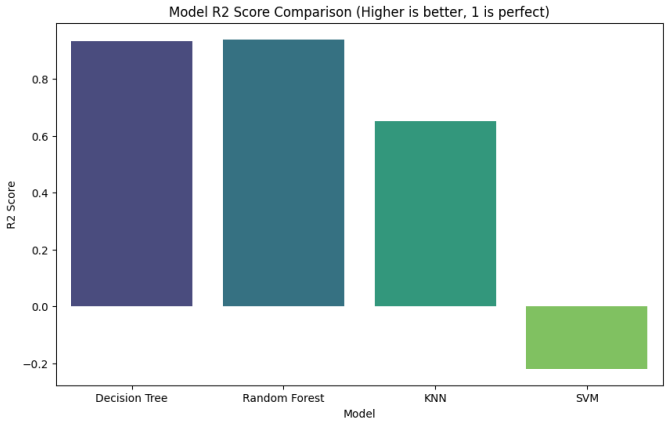

Fig. 7. Comparison of R-squared Scores of the Models.

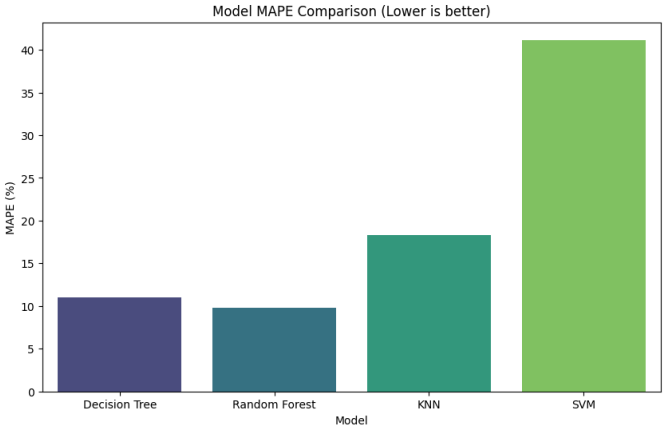

Fig. 8. Comparison of MAPE Scores of the Models.

IX. Discussion

The results show that the ensemble method, Random Forest, outperformed the other models. This is likely due to its ability to reduce variance by averaging the predictions of multiple decision trees, making it less prone to overfitting than a single Decision Tree. The K-Neighbors Regressor and SVR models did not perform as well, which might be due to the nature of the dataset.

One limitation of this study is the relatively small size of the dataset. With more data, the models' performance could potentially be improved. Additionally, more advanced feature engineering and hyperparameter tuning could be performed to further enhance the results.

X. Conclusion

This report presented a comparative study of four machine learning models for predicting automobile prices. The Random Forest Regressor was found to be the most accurate model, with an R^2 score of 0.9395. This demonstrates the effectiveness of ensemble methods for this type of regression problem. Future work could involve exploring other advanced regression models like Gradient Boosting and Neural Networks, as well as collecting more data to improve the model's robustness and accuracy.

References

[1] D. Dua and C. Graff, "UCI Machine Learning Repository," University of California, Irvine, School of Information and Computer Sciences, 2017. https://archive.ics.uci.edu/dataset/10/automobile

[2] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.

[3] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*, Wadsworth, 1984.

[4] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[5] T. Cover and P. Hart, "Nearest Neighbor Pattern Classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[6] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, Aug. 2004.

[7] J. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, 2nd ed., Springer, 2021.

[8] Wikipedia, "Coefficient of determination," Wikipedia. https://en.wikipedia.org/wiki/Coefficient_of_determination

[9] S. Makridakis, S. C. Wheelwright, and R. J. Hyndman, *Forecasting: Methods and Applications*, 3rd ed., Wiley, 1998.