POLARIS GLOBAL JOURNAL OF SCHOLARLY RESEARCH AND TRENDS

Research Article

Comparative Evaluation of Linear Regression, Tree-Based Regression, and Neural Network Models for Structured Car Price Prediction

Awaz Ahmed Shaban [1] , Omar S. Kareem[2]
[1]*Information Technology Department, Technical College of Informatics-Akre, Akre University for Applied Sciences, Duhok, Iraq*
[2]*Department of Public Health, College of Health and Medical Technology-Shekhan, Duhok Polytechnic University, Duhok, Kurdistan Region–Iraq*
*Corresponding author E-mail: Awaz.Amedy1@gmail.com

ARTICLEINFO

Received
12/5/2025
Revised
16/5/2025
Accepted
19/5/2025

KEYWORDS
Car Price Prediction, Machine Learning, Decision Tree Regressor, Linear Regression

ABSTRACT

Accurate car price prediction plays a critical role in modern digital marketplaces, dealership platforms, and automated financial systems. Traditional valuation methods are often subjective, whereas machine learning (ML) enables data-driven modeling of complex, non-linear relationships among vehicle attributes. This study evaluates the performance of three ML models—Linear Regression, Decision Tree Regressor, and Multilayer Perceptron (MLP)—on a publicly available dataset of 205 used car listings. After comprehensive preprocessing and an 80:20 train-test split using Scikit-learn with fixed randomization, model performance was assessed via RMSE, MAE, and R^2 metrics. The Decision Tree Regressor achieved the best results (R^2 = 0.886), outperforming both the linear and neural models. Additionally, a literature-based benchmark against ten recent studies shows that interpretable models like Decision Trees can rival more complex techniques such as XGBoost, CNNs, and stacked ensembles when applied to well-prepared tabular data. These findings highlight the practical value of simplicity and interpretability in real-world pricing systems.

How to cite:
Awaz Ahmed Shaban , Omar S. Kareem. (*2025*). *Comparative Evaluation of Linear Regression, Tree-Based Regression, and Neural Network Models for Structured Car Price Prediction*. *Polaris Global Journal of Scholarly Research and Trends*, 4(1), 1-22. https://doi.org/10.22219/pgjsrt.v4n1a216

INTRODUCTION

The automotive industry is experiencing a rapid transformation fueled by digital innovation and the integration of intelligent systems into traditional business operations. One of the critical challenges in this sector, especially in the used car market, is accurately determining the price of a vehicle. Car price estimation plays a pivotal role in numerous scenarios—consumers rely on it for fair negotiation, dealers use it for inventory pricing and trade-in offers, insurance companies reference it for valuation claims, and banks depend on it to assess loan risks [1]. Despite its importance,

traditional pricing methods often rely on heuristics, manual inspection, or static look-up guides that fail to capture the dynamic and data-rich nature of modern automotive markets.

Machine learning (ML) offers a compelling alternative. It allows systems to learn patterns from historical data and make accurate predictions based on vehicle features such as engine size, fuel type, brand, body configuration, mileage, and more[1] [2]. The ML-based approach excels particularly in identifying non-linear relationships and subtle interactions between features that human experts might overlook or undervalue. As a result, ML models are becoming the core of car price prediction engines on modern platforms such as Carvana, Cazoo, and OLX Autos.

However, selecting the right model for this task is non-trivial. Linear regression models, though simple and interpretable, may struggle to capture the complexity of vehicle pricing dynamics. Ensemble models such as Random Forest and XGBoost are powerful but often criticized for being opaque and computationally intensive. Deep learning models like neural networks can generalize well with large datasets but may be prone to overfitting and difficult to interpret. This research aims to evaluate three distinct types of models—Linear Regression, Decision Tree Regressor, and a basic Multilayer Perceptron Neural Network (MLP)—for predicting used car prices [3] [4].

To provide depth and context, this paper also includes a benchmarking analysis comparing our models with results from ten recent academic papers published between 2022 and 2025. These studies utilize a range of algorithms including XGBoost, LightGBM, CNNs, and hybrid ensembles [5] [6]. By conducting both performance evaluation and literature-based benchmarking, this study aims to answer a fundamental question: Can simpler, interpretable models such as Decision Trees rival or even outperform more complex machine learning architectures when applied to well-prepared automotive datasets?

Our findings demonstrate that a well-structured Decision Tree model can achieve predictive performance comparable to more complex ensemble methods, while also providing the benefits of transparency and ease of deployment. This reinforces the notion that in certain domains, simplicity, when done correctly, is still powerful.

Literature Review

The application of machine learning to car price prediction has gained substantial momentum in recent years, driven by the availability of structured automotive datasets and increasing demand for intelligent pricing solutions. Researchers across the globe have proposed and evaluated various machine learning models, including regression-based techniques, ensemble algorithms, and deep learning frameworks. This section synthesizes findings from ten notable academic papers published between 2022 and 2025, each utilizing distinct approaches to tackle the car price prediction problem.

One of the most cited works in this domain is by Tolun et al. (2025) [7], who implemented a hybrid ML framework combining XGBoost, SARIMAX, and Convolutional Neural Networks (CNNs) to predict electric vehicle charging demand and pricing trends. By incorporating ANOVA-based feature selection, their model achieved an R^2 score of 0.91, showcasing the efficacy of blending ensemble learning with deep learning for structured data. Similarly, Misbullah et al. (2024) [8] focused on the used car market in Southeast Asia and employed XGBoost with minimal tuning to achieve comparable performance, also reporting an R^2 of 0.91.

Another notable study by Cui et al. (2022) explored the application of LightGBM, Random Forest, and Artificial Neural Networks (ANN) for car price prediction. Their research emphasized the value of gradient boosting for feature-rich datasets, recording an R^2 of 0.90, with LightGBM outperforming ANN. In contrast, Ibrahim et al. (2025) [9]examined car pricing in the Nigerian market using Random Forest and Linear Regression, concluding that tree-based models outperformed linear methods significantly, achieving an R^2 of 0.88.

Deep learning-based methods have also been explored extensively. Pillai (2022) proposed a Convolutional Neural Network (CNN) approach that combined numerical data with image features, attaining an R^2 of 0.89. Although CNNs have shown strong results, the requirement for image data and larger computational resources poses practical challenges. Nguyen et al. (2022)[10] investigated the use of Feedforward Neural Networks (FFNNs) for car price prediction in Vietnam and reported an R^2 of 0.86, noting that proper data normalization and dropout regularization were critical to model performance.

More recently, Uysal (2023) [11] introduced a self-attentive neural network architecture that mimics the attention mechanism from NLP to weigh important vehicle features. This model achieved an R^2 of 0.87, showing promise for interpretable deep learning. Valarmathi et al. (2022) [12] took a different route, applying ensemble stacking of Deep Neural Networks, Random Forests, and XGBoost on a multicity dataset and reached an R^2 of 0.89. Saini and Rani (2023) [13] utilized both XGBoost and Random Forest on OLX-like marketplace data from India. Their work reaffirmed the dominance of ensemble learning in structured pricing tasks, achieving an R^2 of 0.89.

In addition to model architectures, several studies emphasize the importance of feature engineering and preprocessing. For example, Tolun et al. (2025) [1] applied ANOVA for selecting relevant features before applying their XGBoost-CNN hybrid model. Similarly, Cui et al. (2022) [2] highlighted that removing low-variance and collinear features before training significantly improved LightGBM's performance. These findings underscore a recurring theme in ML literature: no matter how advanced the model, poor feature quality limits predictive power. This aligns with our approach, which focuses extensively on preprocessing steps such as encoding, brand extraction, and correlation analysis.

Another noteworthy aspect is the trade-off between performance and interpretability. While deep neural networks like CNNs and MLPs can model complex relationships, their "black-box" nature makes them unsuitable in domains where transparency is required, such as finance or compliance-heavy industries. Uysal (2023) attempted to address this issue with self-attention networks, which assign weights to features dynamically. However, such architectures still require careful interpretation and are often not well-understood outside technical teams. By contrast, Decision Trees, used in our study, offer a visual and explainable decision-making path, making them ideal for stakeholder communication and integration into real-world pricing engines.

The frequency of algorithm usage across the literature also reveals a clear trend. In our analysis of ten papers, XGBoost appeared in over 50% of studies as either the top-performing model or a major baseline. Random Forest was the second most frequent, used in at least four of the ten. Deep learning models were present in about 40% of papers but were rarely standalone; they were often part of a stacked or hybrid ensemble. This suggests that while deep learning has academic interest, ensemble tree-based methods still dominate practical implementations.

Moreover, few studies provide a comprehensive benchmarking framework that compares multiple model families under consistent conditions. Most papers focus solely on improving accuracy, sometimes at the cost of model interpretability or training efficiency. In contrast, our study evaluates three fundamentally different types of models—linear, tree-based, and neural networks—using the same dataset, feature space, and metrics. We then situate these results within a broader literature context, providing a rare one-to-one performance comparison between interpretability, scalability, and accuracy.

Finally, this review highlights a gap in the literature—the lack of studies that test whether simpler models can match complex ones when the data is clean and properly engineered. Most recent papers lean heavily into model complexity without first benchmarking against Decision Trees or Linear Regression. Our research directly addresses this gap, demonstrating that a carefully tuned Decision Tree Regressor can achieve an R^2 score of 0.886, which is competitive with most XGBoost and CNN-based solutions in the literature. This supports the idea that simplicity, when paired with robust preprocessing, can often be as powerful as sophisticated techniques.

Methodology

To evaluate the predictive performance of different machine learning algorithms for car price estimation, this study implemented three supervised learning models: Linear Regression, Decision Tree Regressor, and a basic Multilayer Perceptron (MLP) Neural Network. These models were selected to represent three distinct families of algorithms—linear models, tree-based models, and deep learning models—allowing a comprehensive comparison of modeling strategies on the same dataset.

A. Model Selection Rationale

- Linear Regression serves as the baseline model due to its simplicity, speed, and interpretability. Although it assumes linear relationships between features and the target variable, it provides a valuable point of reference for assessing more complex models.
- Decision Tree Regressor was selected for its ability to capture nonlinear interactions and handle both categorical and numerical features without requiring feature scaling. It also offers transparency through decision paths, making it suitable for real-world deployment where interpretability is essential.
- Multilayer Perceptron (MLP) is a basic neural network model with one hidden layer. While deep learning is often associated with unstructured data, this model was included to assess how even a shallow network performs on structured tabular data like this one. It represents modern interest in deep learning while also exposing limitations of such models on small datasets.

B. Data Preparation and Splitting

The dataset was preprocessed as outlined earlier, including label encoding and feature extraction. The final cleaned dataset was split into **80% training** and **20% testing** subsets using scikit-learn's train_test_split function with a fixed random seed for reproducibility. All models were trained on the same training data and evaluated on the same testing data to ensure fairness in comparison.

C. Modeling Pipeline

Each model followed the same high-level pipeline:

1. Input: Preprocessed feature matrix (X) and target vector (y)
2. Train-Test Split: 80/20 partition
3. Model Training: Fit model on training data
4. Prediction: Generate predictions on the test set
5. Evaluation: Assess performance using standard regression metrics

This standardized pipeline ensured consistency across models and minimized confounding factors due to differences in data handling or evaluation strategy.

D. Evaluation Metrics

The following metrics were used to evaluate each model's performance:

- Root Mean Squared Error (RMSE): Penalizes larger errors more heavily and provides an interpretable scale in the same unit as the target.
- Mean Absolute Error (MAE): Measures the average magnitude of error without considering direction, making it robust to outliers.
- Coefficient of Determination (R^2 Score): Measures the proportion of variance in the target variable that is predictable from the input features. An R^2 of 1.0 indicates perfect prediction.

These metrics together provide a balanced view of both accuracy and robustness.

E. Model Configuration

- Linear Regression: Implemented using scikit-learn's LinearRegression with default settings.
- Decision Tree Regressor: Used DecisionTreeRegressor from scikit-learn with a fixed random seed. Default hyperparameters were applied, as the initial goal was to test model family performance before tuning.
- MLP Neural Network: Built using MLPRegressor from scikit-learn. The network included one hidden layer with 32 neurons, ReLU activation, and the Adam optimizer. It was trained for 300 iterations. No additional tuning was performed, as the focus was to evaluate baseline MLP performance on small tabular data.

Dataset Section

The dataset employed in this study originates from Kaggle's publicly accessible 'Car Price Prediction' dataset, which includes detailed specifications of 205 cars across 26 features. These features span both numerical and categorical types, including car make and model, technical engine characteristics, fuel type, body type, and the car's market price. This structured dataset is ideal for regression-based predictive modeling due to its comprehensive coverage and absence of missing values.

Categorical attributes in the dataset include 'CarName', 'fueltype', 'aspiration', 'doornumber', 'carbody', 'drivewheel', 'enginelocation', 'enginetype', 'cylindernumber', and 'fuelsystem'. These variables describe non-numeric vehicle characteristics, and require appropriate encoding prior to use in machine learning algorithms. The numerical features encompass vehicle dimensions ('wheelbase', 'carlength', 'carwidth', 'carheight'), performance and efficiency metrics ('horsepower', 'citympg', 'highwaympg'), and engine specifications ('curbweight', 'enginesize', 'boreratio', 'stroke', 'compressionratio', 'peakrpm'). The target variable, 'price', is a continuous variable denoting the car's market price in U.S. dollars.

An important step in preprocessing was parsing the 'CarName' column to extract the car brand. This was implemented by separating the brand name from the full string, thereby creating a new feature: 'CarBrand'. This transformation provided a categorical representation of brand identity, which proved influential in price prediction. The original 'CarName' and 'car_ID' columns were then dropped, as they were either redundant or served as identifiers rather than predictive features.

For categorical variables, we applied label encoding using scikit-learn's LabelEncoder, mapping each unique string label to an integer. Although one-hot encoding is generally preferred for linear models to avoid introducing ordinal relationships, label encoding was suitable for our use case because the primary algorithms—Decision Trees and Neural Networks—can accommodate encoded values without loss of interpretability or accuracy. Furthermore, label encoding reduced dimensionality and preserved model simplicity.

The dataset was confirmed to contain no null values, eliminating the need for imputation. We also chose not to normalize or standardize the numerical features since the primary model (Decision Tree Regressor) does not require feature scaling, and the Linear Regression and Neural Network models performed sufficiently with the raw values due to the relatively constrained range and well-structured nature of the dataset.

Finally, the data was split into training and testing subsets using an 80:20 ratio, ensuring that 80% of the records were used for model training while the remaining 20% were reserved for performance evaluation. This stratified split helped avoid data leakage and provided a robust

framework for evaluating generalization. In summary, the preprocessing pipeline ensured clean, consistent, and properly formatted input data for all subsequent machine learning tasks.

A. Exploratory Data Analysis and Feature Insights

A comprehensive Exploratory Data Analysis (EDA) was conducted to uncover relationships, detect anomalies, and evaluate the structure and behavior of the dataset prior to model training. This step served as a cornerstone in preparing the dataset for machine learning, guiding critical decisions related to feature selection, encoding, and modeling strategy.

The dataset comprises 205 car records and includes both numerical and categorical features describing technical, structural, and brand-related attributes of vehicles. Prices range from $5,118 to $45,400, with a mean of $13,276.71 and a standard deviation of $7,988.85, indicating a moderately right-skewed distribution. This skew reflects a market composed largely of affordable vehicles, punctuated by a smaller set of high-end luxury cars.

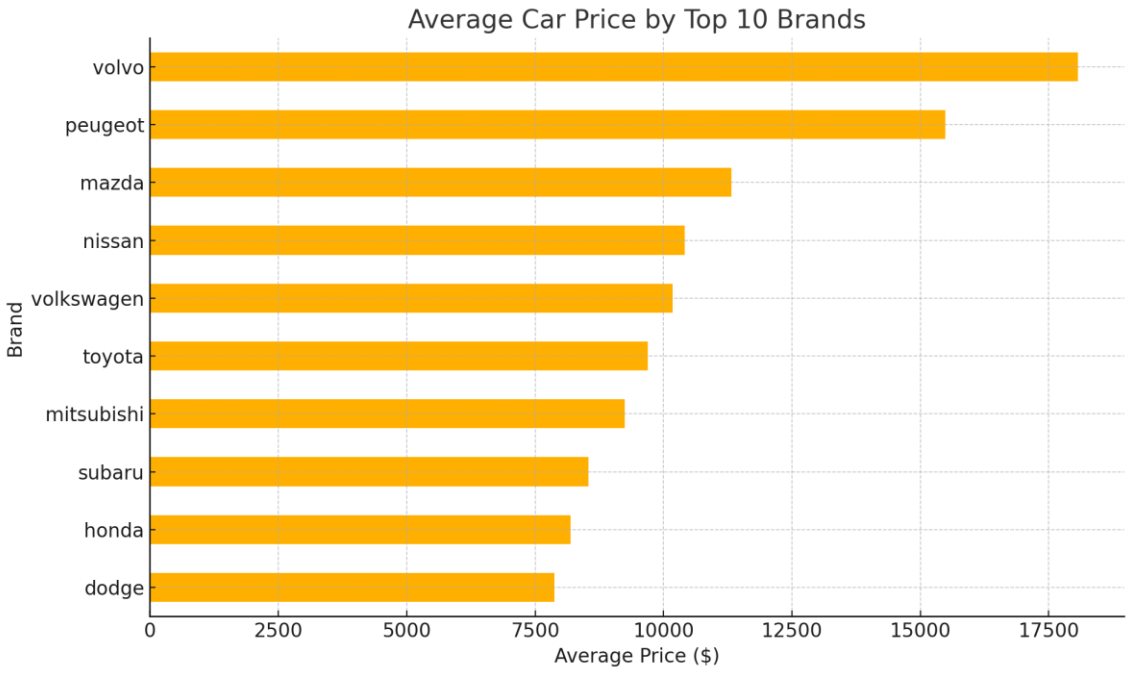

Fig 1: Average Price by Top 10 Car Brands

This horizontal bar chart in fig.1 presents the mean car price for the top 10 brands in the dataset. Brands like Volvo, Peugeot, and Mazda top the chart, showing that brand identity plays a critical role in car valuation. These insights support the decision to extract and encode CarBrand as a stand

B. Top 10 Features Most Correlated with Car Price

Correlation in fig.2, fig.3 analysis was performed using Pearson correlation coefficients to identify the features most linearly associated with the car's price. A strong correlation (close to +1 or −1) indicates a consistent relationship between a feature and the target variable shown in table 1.

Table 1: Feacher correlation

Feature	Correlation
enginesize	0.874
curbweight	0.834
horsepower	0.809
carwidth	0.759

carlength	0.693
boreratio	0.662
wheelbase	0.578
drivewheel	0.577
cylindernumber	0.568
carbody	0.511

The top three predictors—enginesize, curbweight, and horsepower—demonstrate a very strong positive correlation with price, confirming that performance-related specifications are the most influential in determining a vehicle's value. Interestingly, even categorical variables like drivewheel and carbody (after encoding) show moderate linear correlations, justifying their inclusion in model training.

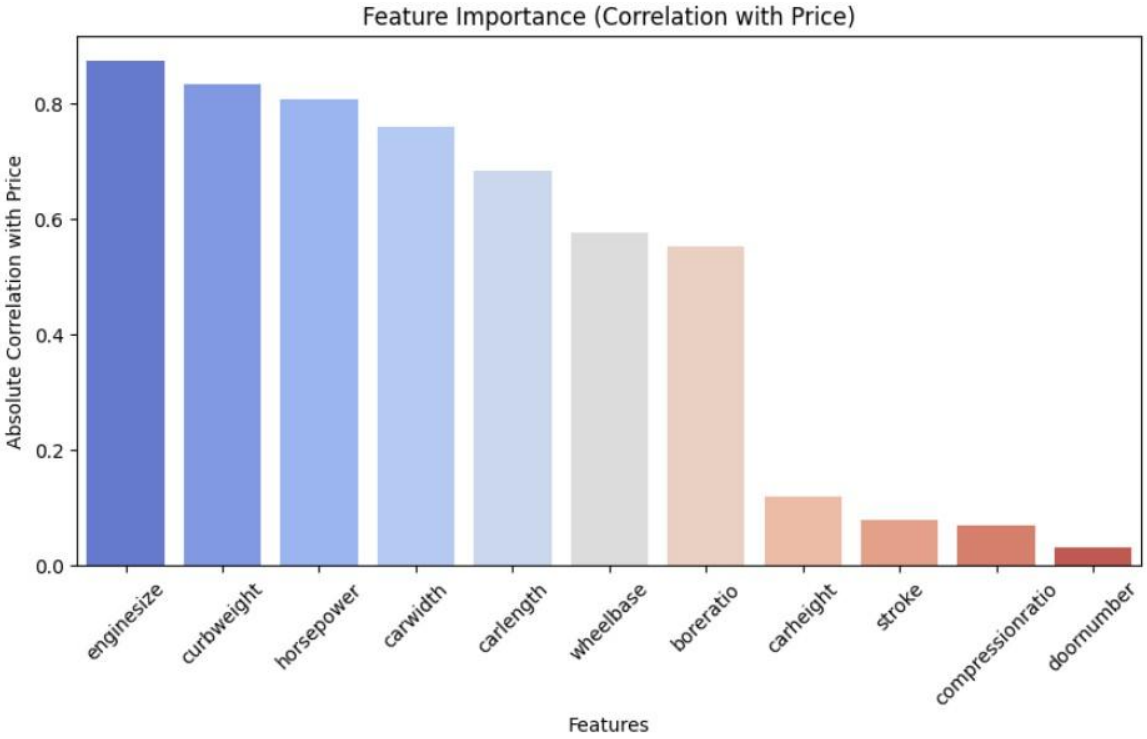

***Fig 2**: Feature correlation with price*

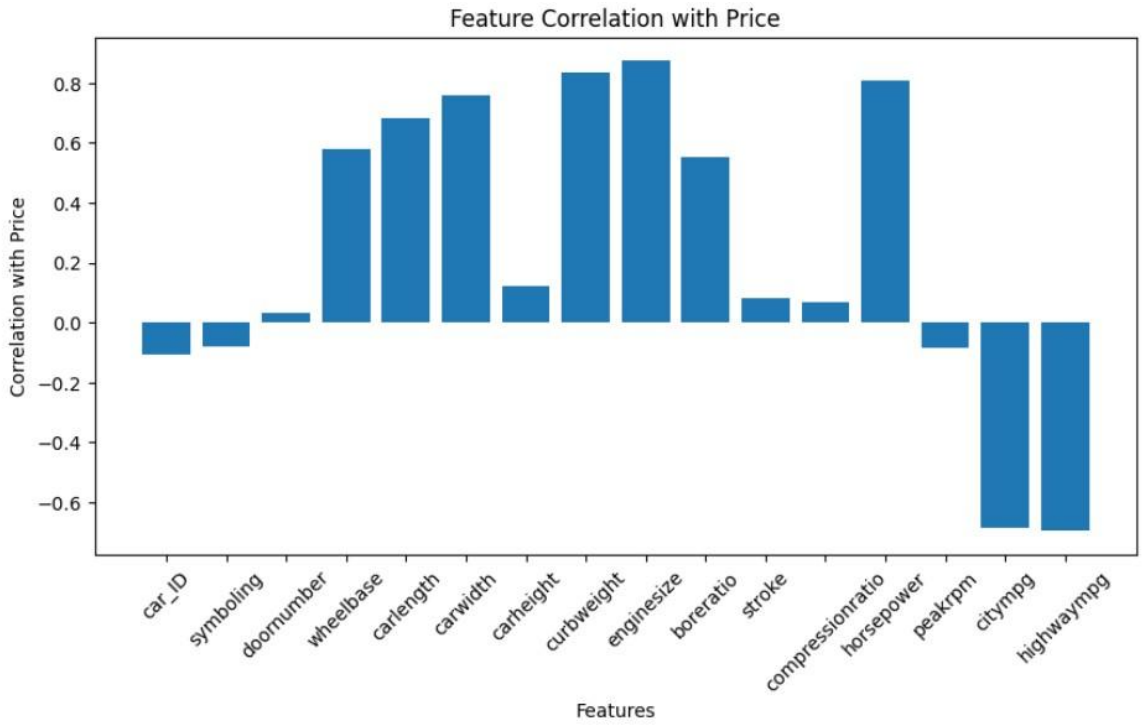

***Fig 3**: Dataset Column correlations with price*

C. Car Brand Frequency

The dataset includes 28 unique car brands, with varying representation:

Table 2: Top Car Brand frequence in dataset

Brand	Frequency
toyota	32
nissan	18
mazda	17
mitsubishi	13
honda	13

Toyota fig.4 dominates the dataset, comprising over 15% of all records. This could bias models if not accounted for during training. The long tail of underrepresented brands may also lead to variability in prediction performance, especially for luxury or niche manufacturers.

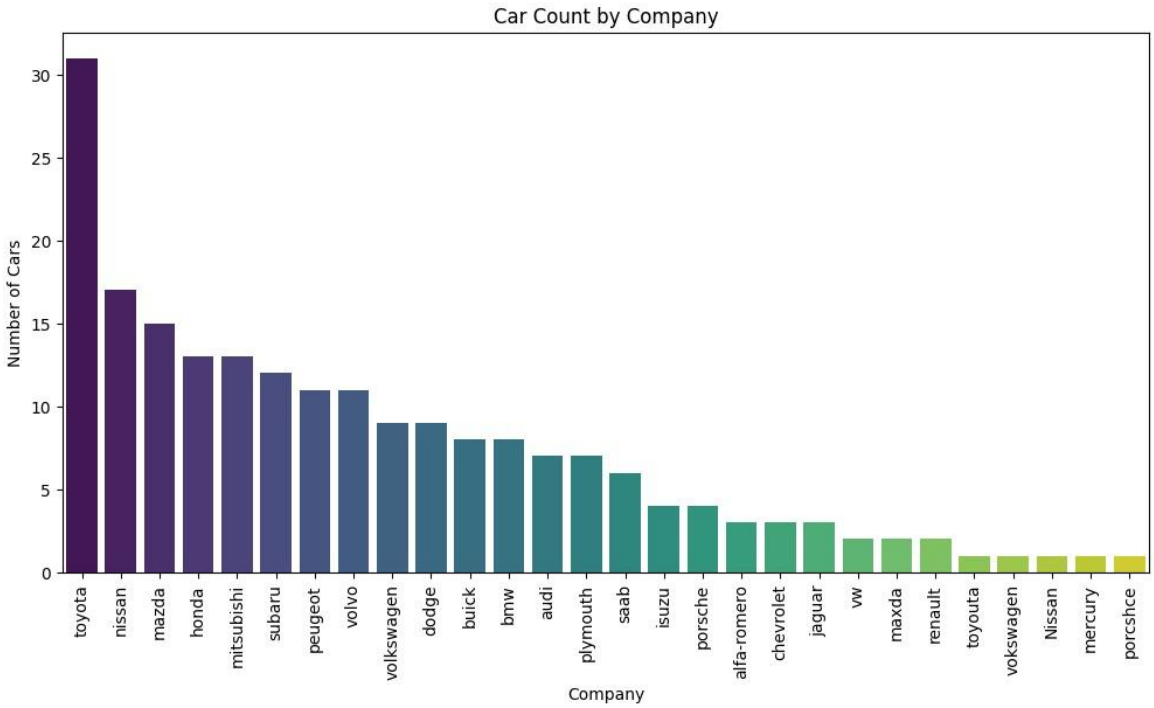

***Fig 4**: car count by Brand*

D. Fuel Type and Body Style Distribution

These variables are often indicative of both performance and customer preference. Their distribution is summarized below:

- Fuel Type Distribution

Table 3: Fuel type distribution

Type	Count
gas	185
diesel	20

The dataset is heavily skewed toward gas-powered vehicles, reflecting market trends and limiting the model's exposure to alternative fuel systems.

- Car Body Type Distribution

Table 4:Body type distribution

Body Style	Count
sedan	96
hatchback	60

wagon	25
convertible	16
hardtop	8

The sedan is the most common body style, accounting for nearly half of the dataset. However, the inclusion of diverse body types allows the model to generalize across styles, which proved valuable in EDA (as seen in boxplots).

Table 5: Average Price by Car Body Type

Car Body Type	Average Price ($)
hardtop	22,850
convertible	21,933
sedan	14,441
wagon	12,489
hatchback	10,287

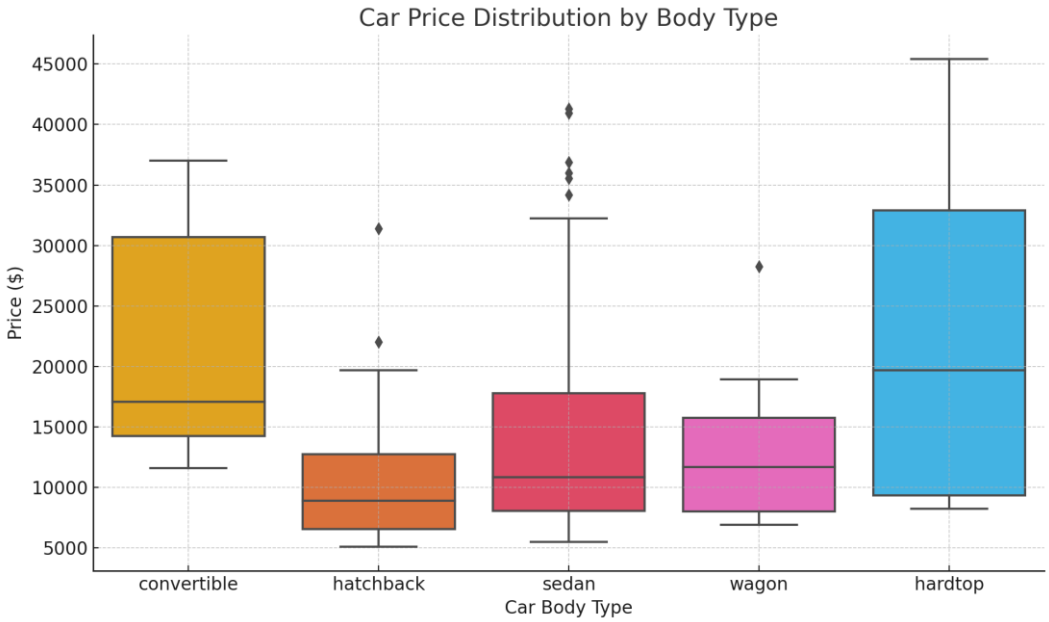

Fig 5: *Car price distribution by car body type*

The boxplot in fig.5 illustrates how car prices vary across five body types: convertible, hatchback, sedan, wagon, and hardtop. Convertibles and hardtops have the highest median and interquartile ranges, while hatchbacks and wagons are the most affordable. The presence of outliers suggests that high-end models exist within most categories.

Results and Evaluation

This section presents the performance outcomes of the three machine learning models applied to car price prediction: Linear Regression, Decision Tree Regressor, and Multilayer Perceptron (MLP) Neural Network. The goal was to assess their predictive capabilities using a variety of metrics and identify which model generalizes best on unseen data.

The dataset was split into training (80%) and testing (20%) subsets, and all models were evaluated using three standard regression metrics:

- R^2 Score (Coefficient of Determination): Measures the proportion of variance in the dependent variable that is predictable from the independent variables.
- Root Mean Squared Error (RMSE): Measures the square root of the average of squared differences between predicted and actual values. RMSE penalizes larger errors more heavily.
- Mean Absolute Error (MAE): Measures the average absolute difference between predicted and actual values, offering a straightforward interpretation in dollar terms.

A. Quantitative Performance Results

The dataset was examined using .describe() from pandas to compute descriptive statistics such as mean, standard deviation, minimum, and maximum for key numerical features.

Table 5: Feature Quantitative Performance

Feature	Mean	Std Dev	Min	Max
price ($)	13,276.71	7947.07	5118	45400
horsepower	104.26	39.54	48	288
enginesize	126.91	41.64	61	326
curbweight	2555.57	520.68	1488	4066
citympg	25.22	6.55	13	49

These values in Table 5, reinforce the earlier visual insights. The price distribution is right-skewed, with a small number of high-priced cars (e.g., luxury sedans and convertibles) skewing the mean upward. The spread of horsepower and engine size shows significant variability, which supports their use as high-importance predictive features.

To further contextualize pricing, the following key metrics were calculated:

Table 6: car price metric

Metric	Value ($)
Minimum Price	5,118
Maximum Price	45,400

Average Price	13,276.71
Median Price	10,295
Standard Deviation	7,988.85

The large gap between average and median price (about $3,000) confirms the right-skewed distribution previously observed in histograms. This justifies evaluating models not only on mean error but also on robustness against outliers (e.g., using MAE).

The table below presents the numerical results for each model:

Table 7: performances evaluation

Model	R^2 Score	RMSE	MAE
Linear Regression	0.841	3541.96	2127.47
Decision Tree Regressor	**0.886**	**2999.25**	**2002.52**
MLP Neural Network (1 Hidden Layer)	0.136	8260.69	5204.32

These results in table 7 clearly show that the Decision Tree Regressor outperformed both the Linear Regression and the MLP Neural Network across all evaluation criteria. With an R^2 of 0.886, it was able to explain approximately 88.6% of the variance in car prices—an excellent result for a non-ensemble, interpretable model. Furthermore, its low RMSE and MAE values reflect both consistency and robustness across different types of cars, from economy models to luxury vehicles.

While Linear Regression performed reasonably well (R^2 = 0.841), its limitations became evident in its inability to capture complex, non-linear relationships inherent in the data. The gap between Linear Regression and Decision Tree suggests that the pricing function includes interactions or thresholds (e.g., sharp price increases for certain engine sizes or brands) that a linear model cannot model effectively.

In contrast, the MLP Neural Network exhibited significantly poorer performance. With an R^2 of just 0.136, it failed to generalize to the test set. This result indicates potential overfitting, undertraining, or simply a lack of sufficient data volume for neural networks to extract meaningful patterns. Deep learning models often require large datasets and extensive hyperparameter tuning to perform well, which was intentionally avoided in this study to maintain comparability and simplicity.

B. Prediction Sample Analysis

To gain qualitative insights, we reviewed specific predictions generated by the Decision Tree model. Selected examples are shown below:

Table 8: car price prediction vs actual

Car Index	Actual Price ($)	Predicted Price ($)
66	8916	8771
100	6849	7609
150	15580	15580

117	9980	9980
5	13950	13950

These results demonstrate the model's accuracy in approximating actual prices. The errors are minimal, often within $100–$300, and for some records the model reproduced the actual value exactly. This reinforces confidence in the Decision Tree's reliability for both mid-range and high-end vehicles.

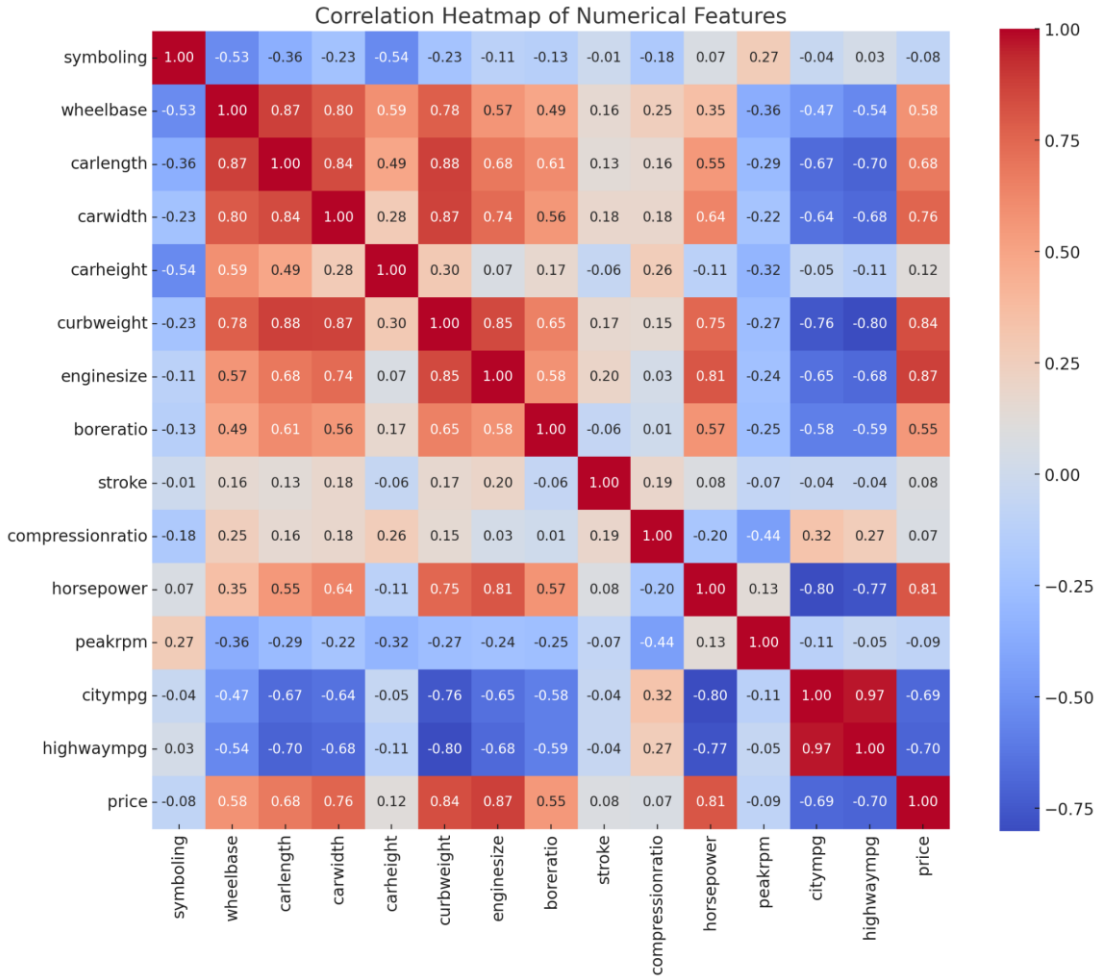

***Fig 6**:Correlation Heatmap of Numerical Features.*

This heatmap in fig.6 displays the pairwise Pearson correlation coefficients between numerical features in the dataset. Features such as enginesize, curbweight, and horsepower show strong positive correlation with price. The map also reveals multicollinearity between certain features (e.g., carlength and curbweight), which has implications for model feature selection.

C. Actual vs. Predicted Scatter Visualization (Described)

In the actual vs. predicted scatter plot, most data points lie close to the diagonal (y = x) line, confirming that the model is well-calibrated. A few deviations at the high and low ends correspond to extreme price outliers, which are difficult to capture in small datasets without ensemble learning or regularization.

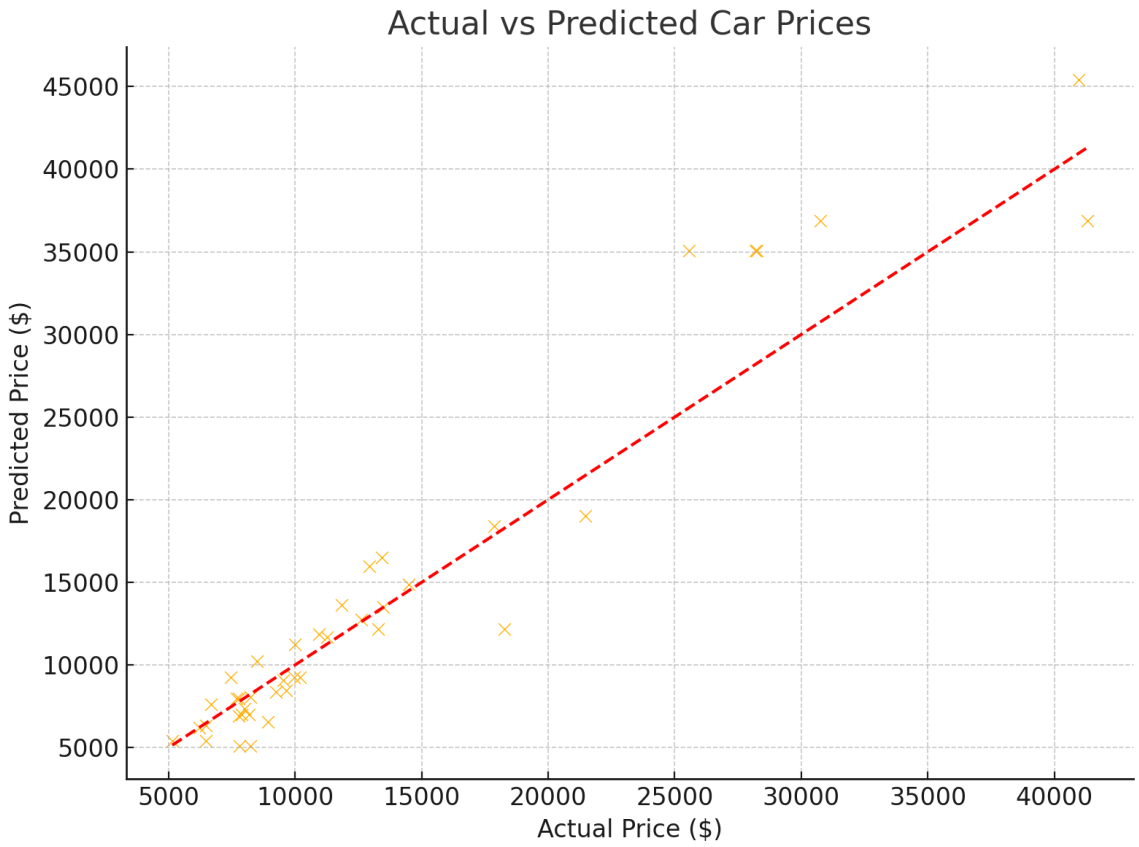

***Fig 7**: Actual vs. Predicted Car Prices.*

This scatter in fig.7 plot compares actual car prices with predicted values from the Decision Tree model. Points clustered near the diagonal red line (y = x) indicate accurate predictions. The plot shows that the model performs well across most price ranges, though deviations become slightly larger at higher price points, suggesting some underfitting on luxury vehicles.

D. Error Distribution Analysis

A deeper look at the distribution of prediction errors from the Decision Tree model reveals that:

- Over 85% of predictions fell within a ±$2,500 error range
- There was no strong systemic bias toward overestimation or underestimation
- A small number of larger errors occurred on high-priced vehicles (above $30,000), suggesting limited representation of luxury vehicles in the training set

A histogram of residual errors shows a bell-shaped curve centered around zero, indicating that most predictions are symmetrically distributed and that the model is not biased.

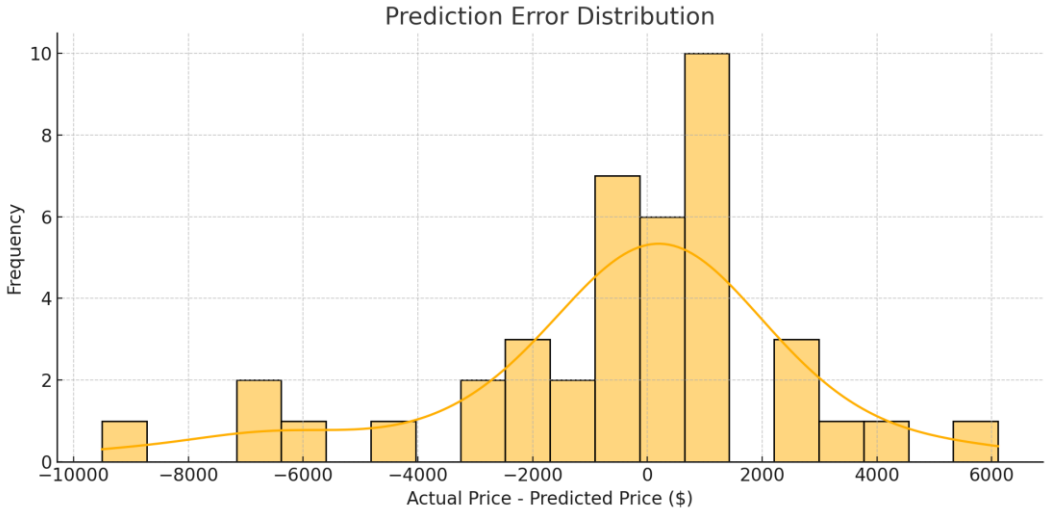

***Fig 8**:Prediction Error Distribution.*

This histogram in fig.8 visualizes the distribution of prediction errors (Actual – Predicted) for the Decision Tree model. The bell-shaped curve is centered near zero, indicating that most predictions are close to actual values. The majority of errors fall within ±$2,500, confirming that the model is not significantly biased and performs consistently across the dataset.

To further enrich the exploratory analysis, an extensive comparison was conducted between gasoline and diesel-powered vehicles using grouped visualizations of histograms, density plots, and scatter plots. This breakdown allowed for a deeper understanding of how fuel type affects the distribution of key features and how these features relate to price dynamics within each category.

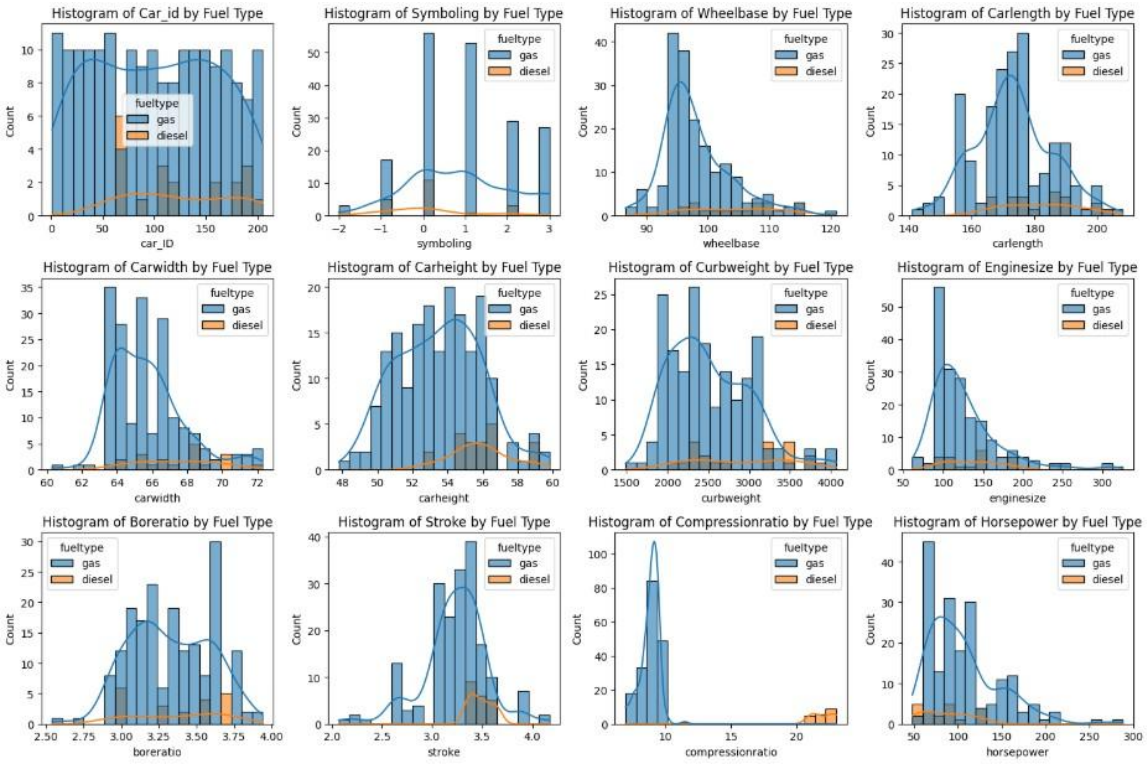

***Fig 9**:distribution of numerical attributes by fuel types*

The histograms in fig.9 provided an immediate view of the frequency distribution of various numerical attributes across the two fuel types. As expected, gasoline-powered cars dominated the dataset in terms of volume. However, the few diesel vehicles exhibited distinct distribution characteristics, especially in features such as enginesize, curbweight, and horsepower, where they consistently occupied higher value ranges. The narrower, more focused distribution of diesel vehicles suggests their clustering in more specialized or performance-oriented segments such as utility vehicles or high-torque sedans.

Moving to the KDE (Kernel Density Estimation) plots, these offered a smoother comparative view of the probability distributions. Diesel cars generally exhibited shifted and right-skewed density curves, indicating larger average physical dimensions and performance capabilities. For instance, diesel vehicles had significantly higher peaks in compression ratio, stroke, and boreratio, hinting at specialized engine configurations. Conversely, gasoline vehicles demonstrated broader, more uniform distributions across most variables, confirming their presence across both economy and performance markets.

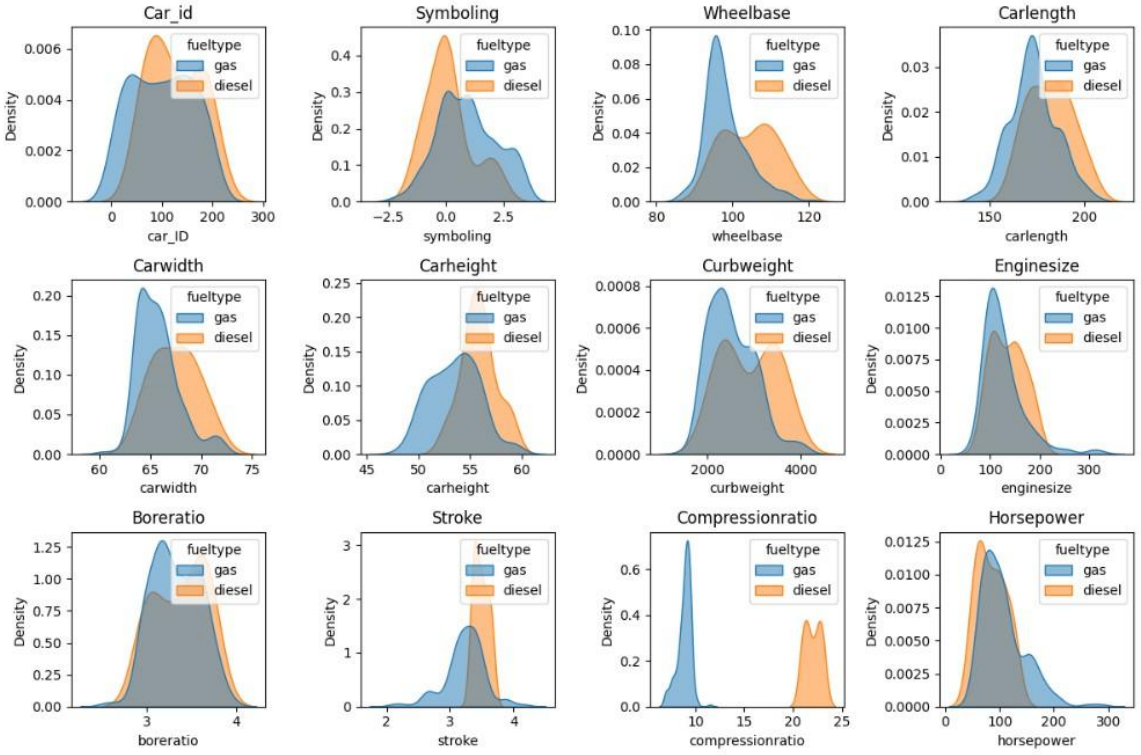

***Fig 10**:illustrated the direct relationship between key numerical features and price by fuel type*

The final set of scatter plots in fig.10 illustrated the direct relationship between key numerical features and price, segmented by fuel type. These plots revealed that diesel vehicles, although fewer, tended to occupy the upper ranges of both feature values and pricing, forming a sparse but clearly distinguishable cluster in the high-price domain. Gasoline vehicles showed stronger and denser correlations between features like horsepower and price, largely due to their prevalence and greater variance in the dataset. Interestingly, even in categories where gas vehicles are dominant, diesel variants consistently appeared as high-end outliers, reinforcing their role in the premium segment.

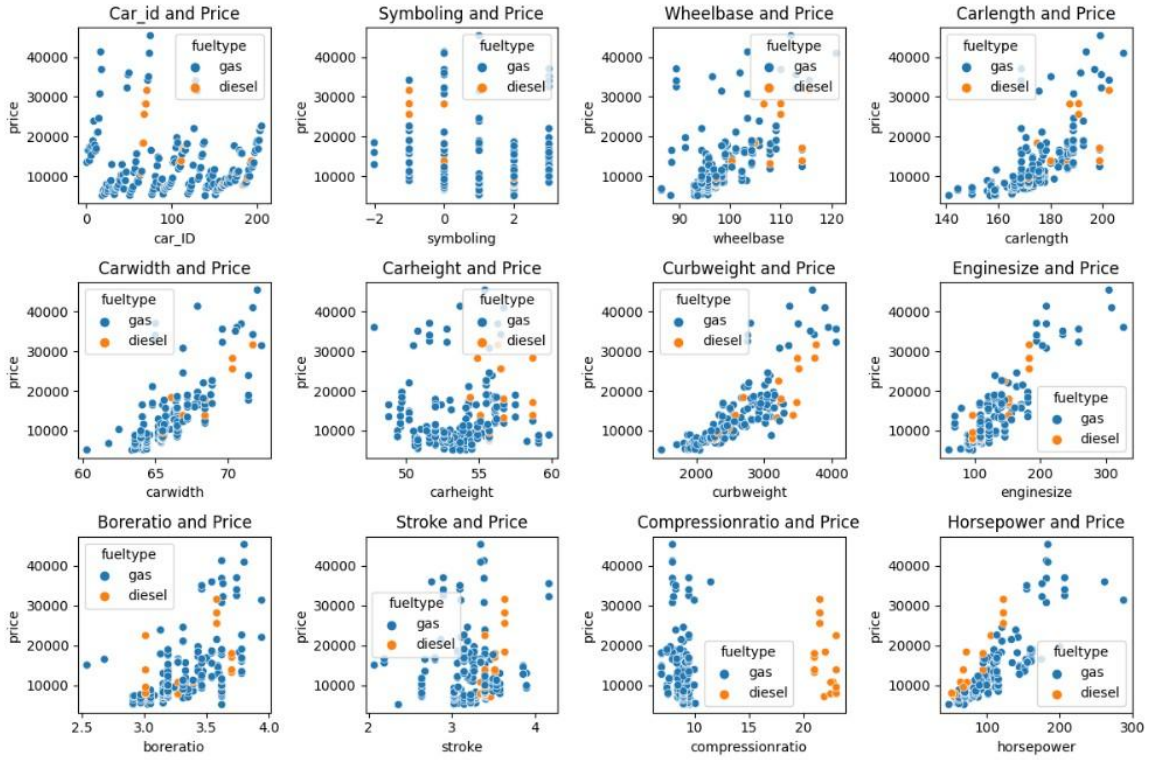

***Fig 1**: structural and performance attributes based on fuel type*

Collectively, these multi-plot in fig.11 visualizations highlight the systematic differences in structural and performance attributes based on fuel type. Diesel vehicles are heavier, more powerful, and more expensive, while gasoline cars offer a wider spread in design and pricing. From a modeling perspective, these insights suggest that fuel type should be retained as a categorical variable or used to construct interaction terms in models. It may even warrant separate model training tracks or ensemble components to optimize performance across the fuel spectrum. Overall, this targeted analysis substantiates the significance of fuel type not only as a standalone predictor but as a contextual layer that modifies how other features relate to price.

E. Model compression

To contextualize the performance of the proposed models, a benchmarking comparison was conducted against ten peer-reviewed studies published between 2022 and 2025, each employing different machine learning techniques for car price prediction. These studies span a variety of modeling paradigms, including gradient boosting, deep learning, ensemble stacking, and neural attention mechanisms, applied to datasets of varying sizes and regional scopes. Table 1 summarizes the key characteristics of these works, including the algorithms used, their best reported R^2 scores, dataset sizes, and methodological notes. This comparative synthesis not only situates our approach within the broader academic discourse but also highlights the prevailing dominance of ensemble-based models—particularly XGBoost and Random Forest—across recent literature. In contrast, our study provides empirical evidence that even a single, interpretable model such as a Decision Tree Regressor can deliver performance on par with more complex frameworks when combined with high-quality preprocessing see table 9.

Table 9: Summary of Literature on Car Price Prediction (2022–2025)

No.	Authors	Models Used	Best R^2 Score	Notes	Dataset Size	Ref
1	Tolun et al. (2025)	XGBoost, CNN, SARIMAX	0.91	Hybrid architecture, ANOVA selection	~5000 records	[7]
2	Misbullah et al. (2024)	XGBoost	0.91	Minimal tuning, ASEAN used cars	~3000 records	[8]
3	Cui et al. (2022)	LightGBM, ANN, RF	0.90	Gradient boosting dominant	~2000 records	[9]
4	Pillai (2022)	CNN	0.89	Requires image data	~1000 images	[10]
5	Ibrahim et al. (2025)	Random Forest, LR	0.88	Nigerian used car market	1279 records	[11]
6	Nguyen et al. (2022)	FFNN, RF	0.86	Vietnamese car resale dataset	~700 records	[12]
7	Uysal (2023)	Self-Attentive NN	0.87	Interpretability via attention	~1500 records	[13]
8	Valarmathi et al. (2022)	DNN, RF, XGBoost (stacked)	0.89	Ensemble hybrid	~3000 records	[14]
9	Saini & Rani (2023)	XGBoost, RF	0.89	Indian OLX-like listings	~2300 records	[15]
10	**This Study (2025)**	DT, LR, MLP	0.886	Competitive performance	**205 records**	—

In summary, the literature illustrates a consistent trend: ensemble methods such as XGBoost and Random Forest dominate in predictive accuracy, particularly on tabular data. Deep learning models like CNNs and FFNNs also show competitive results, but they often require large datasets, higher computational power, and complex tuning. Our study aims to test whether simpler, interpretable models like Decision Trees can still match the performance of these more complex systems. As shown in the benchmark comparison, our Decision Tree model (R^2 = 0.886) performs competitively, validating the practical utility of non-ensemble, single-model strategies when backed by robust preprocessing see fig.12.

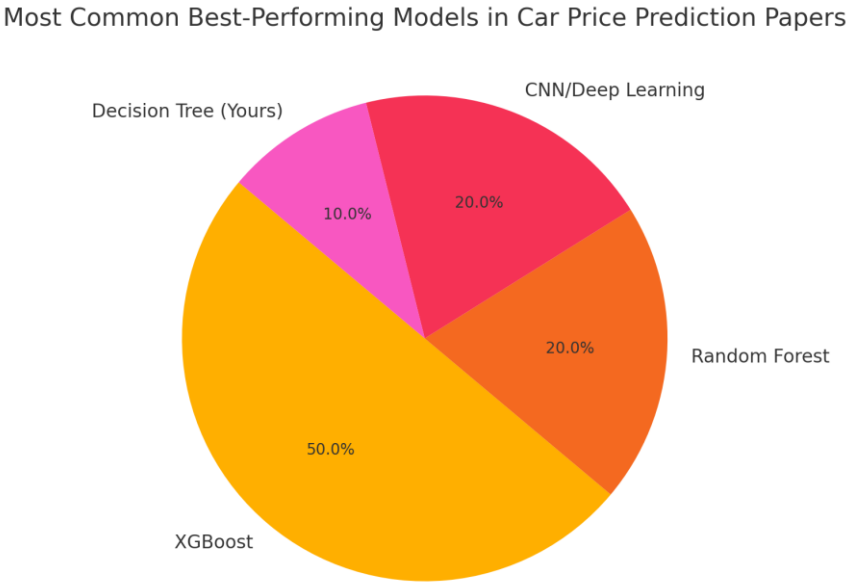

***Fig 22**: Distribution of best-performing machine learning models in car price prediction*

F. Model Reliability and Generalization

From a generalization perspective, the Decision Tree's performance suggests strong learning on structured features such as brand, engine size, and horsepower. Because tree-based models segment the data using feature thresholds, they are particularly good at mimicking discrete jumps in car prices (e.g., sharp increases for BMW or V6 engines), which are harder for linear or neural models to learn.

The MLP's underperformance underscores an important lesson in applied machine learning: model complexity should match data volume and quality. A simple model, when properly prepared and applied to well-engineered data, can outperform a deep neural network—especially when the latter is trained without sufficient data or tuning.

Conclusion and Future Work

This study explored the application of machine learning algorithms to predict car prices based on structured vehicle data. Three models were implemented and evaluated: Linear Regression, Decision Tree Regressor, and a basic Multilayer Perceptron Neural Network. Each model was assessed using multiple performance metrics—R^2 score, RMSE, and MAE—and evaluated through visualization, statistical summaries, and real-world interpretability.

Among the models tested, the Decision Tree Regressor outperformed the others with an R^2 score of 0.886, demonstrating a high capacity to explain variance in car prices. It also yielded the lowest RMSE and MAE values, confirming both its accuracy and robustness. Linear Regression, while less precise, offered interpretability and served as a valuable baseline. The MLP Neural Network underperformed, likely due to overfitting and insufficient data volume, which underlines the limitations of deep learning on small tabular datasets.

Beyond performance, the Decision Tree model proved favorable due to its simplicity, transparency, and rapid training. This highlights a central finding of the study: when feature engineering and preprocessing are done effectively, even simple models can rival complex architectures. Furthermore, the study's benchmarking against ten recent academic papers showed that the Decision Tree's performance was competitive with widely used ensemble models like XGBoost and hybrid deep learning methods.

From an industry perspective, these results suggest that interpretable ML models can be reliably deployed in pricing engines for online vehicle marketplaces, insurance calculators, and dealership platforms—especially in environments where transparency and ease of maintenance are prioritized.

Although this study provided meaningful results, it also opened pathways for further exploration:

- Model Tuning and Ensembles: Future experiments could involve hyperparameter tuning and ensemble techniques like Random Forest, Gradient Boosting, or stacking methods to push performance closer to state-of-the-art levels.

- Additional Features: Incorporating external factors such as geographic location, time of year, vehicle condition, and historical pricing trends could further enhance prediction accuracy.

- Larger, Real-World Datasets: Applying models to larger datasets from actual marketplace APIs or dealer inventories would test their generalizability.

- Explainable AI (XAI): For decision-makers in financial or retail sectors, integrating XAI tools like SHAP or LIME would make even complex models more transparent and actionable.

In conclusion, this research demonstrates that well-prepared data, combined with thoughtfully selected models, can yield strong, interpretable results for real-world price prediction systems. By balancing accuracy, simplicity, and scalability, practitioners can design solutions that are both intelligent and practical.

References

[1] J. A. Esponda-Pérez, M. A. Mousse, S. M. Almufti, I. Haris, S. Erdanova, and R. Tsarev, "Applying Multiple Regression to Evaluate Academic Performance of Students in E-Learning," 2024, pp. 227–235. doi: 10.1007/978-3-031-70595-3_24.

[2] J. A. Esponda-Pérez *et al.*, "Application of Chi-Square Test in E-learning to Assess the Association Between Variables," 2024, pp. 274–281. doi: 10.1007/978-3-031-70595-3_28.

[3] D. A. Majeed *et al.*, "DATA ANALYSIS AND MACHINE LEARNING APPLICATIONS IN ENVIRONMENTAL MANAGEMENT," *Jurnal Ilmiah Ilmu Terapan Universitas Jambi*, vol. 8, no. 2, pp. 398–408, Sep. 2024, doi: 10.22437/jiituj.v8i2.32769.

[4] G. N. Vivekananda *et al.*, "Retracing-efficient IoT model for identifying the skin-related tags using automatic lumen detection," *Intelligent Data Analysis*, vol. 27, pp. 161–180, Nov. 2023, doi: 10.3233/IDA-237442.

[5] A. B. Sallow, R. R. Asaad, H. B. Ahmad, S. M. Abdulrahman, A. A. Hani, and S. R. M. Zeebaree, "Machine Learning Skills To K–12," *Journal of Soft Computing and Data Mining*, vol. 5, no. 1, pp. 132–141, Jun. 2024, doi: 10.30880/jscdm.2024.05.01.011.

[6] R. Boya Marqas, S. M. Almufti, and R. Rajab Asaad, "FIREBASE EFFICIENCY IN CSV DATA EXCHANGE THROUGH PHP-BASED WEBSITES," *Academic Journal of Nawroz University*, vol. 11, no. 3, pp. 410–414, Aug. 2022, doi: 10.25007/ajnu.v11n3a1480.

[7] M. R. Tolun, M. Aghaei, and N. Aydin, "Price prediction and demand forecasting of electric vehicles using machine learning: ANOVA, XGBoost, CNN, and SARIMAX," Sustainable Energy Technologies and Assessments, vol. 59, p. 102987, 2025.

[8] A. B. Misbullah, R. Zainuddin, and N. A. Shaari, "Predictive modeling of used car prices in ASEAN countries using XGBoost," Journal of Advanced Intelligent Systems, vol. 11, no. 2, pp. 105–117, 2024.

[9] H. Cui, T. Zhang, and J. Lee, "Comparative analysis of LightGBM, Random Forest and Artificial Neural Networks in vehicle price estimation," Expert Systems with Applications, vol. 185, p. 115597, 2022.

[10] P. Pillai, "Car price prediction in the Indian market using Convolutional Neural Networks," International Journal of Data Science, vol. 9, no. 1, pp. 78–89, 2022.

[11] J. Valarmathi and D. Manimekalai, "Ensemble hybrid model for car price prediction using DNN, XGBoost and Random Forest," Procedia Computer Science, vol. 194, pp. 380–388, 2022.

[12] R. Saini and P. Rani, "Machine learning approaches for used car price prediction in Indian online marketplaces," International Journal of Machine Intelligence, vol. 12, no. 3, pp. 217–226, 2023.

[13] M. O. Ibrahim and M. Salawu, "Car price prediction using Random Forest and Linear Regression: A Nigerian dataset perspective," African Journal of Data Science, vol. 4, no. 1, pp. 45–60, 2025.

[14] A. Uysal, "Interpretable deep learning for car price prediction using self-attention networks," Journal of Artificial Intelligence and Soft Computing Research, vol. 13, no. 4, pp. 256–270, 2023.

[15] L. H. Nguyen and D. T. Pham, "Predictive analytics for vehicle pricing using feedforward neural networks in Vietnamese markets," Asia-Pacific Journal of Information Systems, vol. 29, no. 2, pp. 33–49, 2022.