Report On Price Prediction of Car Data Using Various Supervised Machine Learning Algorithms

ML can be broadly categorized into supervised and unsupervised approaches. Supervised ML is centered around predicting an output from a set of inputs, whereas unsupervised ML centers around creating data-driven patterns and groupings within the input data without a labeled output. In this study, we are doing a comparison by training our model on different supervised ML algorithms

- Multiple Linear Regressor
- Decision Tree Regressor
- KNN Regressor
- Random Forest Regressor
- Gradient Boosting Regressor

Data Overview

- The data set includes features such as car specifications and price.
- Tasks: Data cleaning, exploratory data analysis (EDA), data preprocessing, splitting the data, and model training.

1. **Data Cleaning**

It involves removing duplicated rows. In our data set, there are 4456 duplicates, which we dropped. Then, we Converted numerical columns to float while handling non-numeric values using pd.to_numeric(errors= 'coerce').

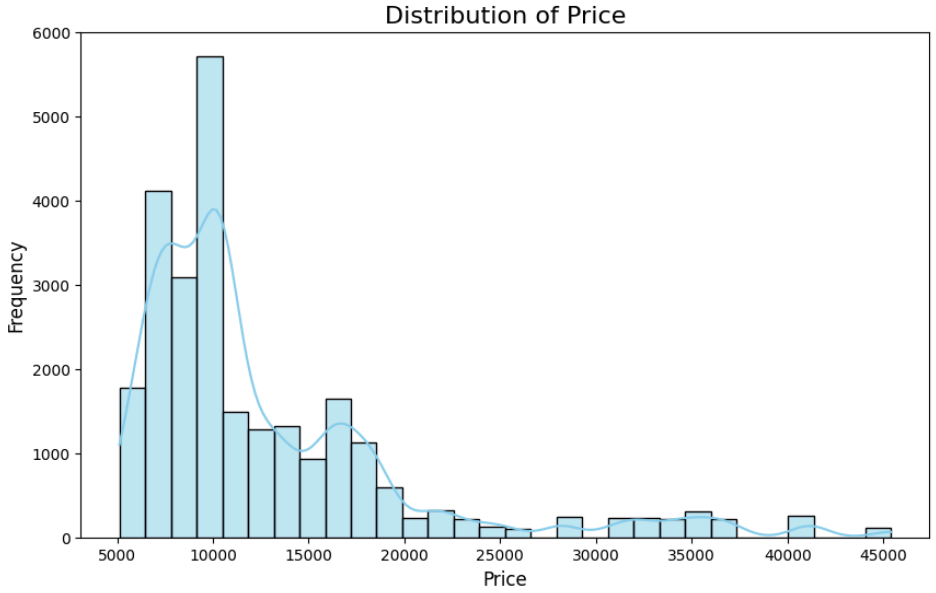
Data cleaning also involves imputation where we impute missing numerical values and categorical columns.

When choosing an imputation method for missing values, the goal is to select a method that best represents the underlying distribution and relationships in the data without introducing bias or distorting the results. Here's an explanation of the methods and why they are suitable for specific variables:

- Numerical data is **normally distributed** (symmetrical and without significant outliers) and consistent because the mean, 50% (median), and the 25%-75% range are close. So we impute these column with Mean.
- In categorical column data is imputed with most repeated values(mode)
- As there are 22 unique values in the make imputation technique is slightly different, Group by the columns 'fuel-type', 'aspiration', and ‘num-of-doors' and impute the 'make' column with the mode (most frequent value) within each group.

2. **EDA: Exploratory data analysis** involves analyzing the car price data set to understand its structure, visualize key features like price distribution and correlations, and explore patterns such as car sales by manufacturer and price ranges across brands. This process helps uncover insights and prepares the data for the machine-learning model.

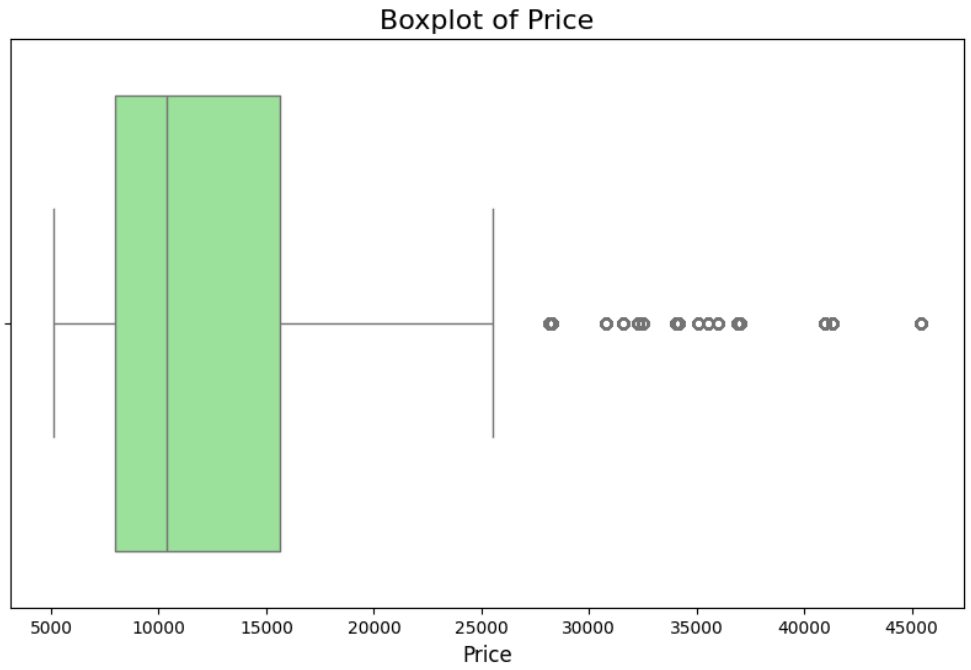
- Visualized target feature(price) distribution and skewness (histogram, box-plot).



This is a **histogram** shows the distribution of prices:

- **X-axis (Price):** Represents the range of prices (e.g., from 5,000 to 45,000).
- **Y-axis (Frequency):** Indicates the number of items (frequency) within each price range.
- **Bars (Rectangles):** Each bar represents the count of items in a specific price range.
- **Density Curve (Smooth Line):** A smoothed line showing the overall trend of the data distribution.

Observation: The graph shows a higher concentration of prices in the lower range (5,000–15,000), with fewer items in the higher price ranges (above 20,000). This indicates the data is **right-skewed**.

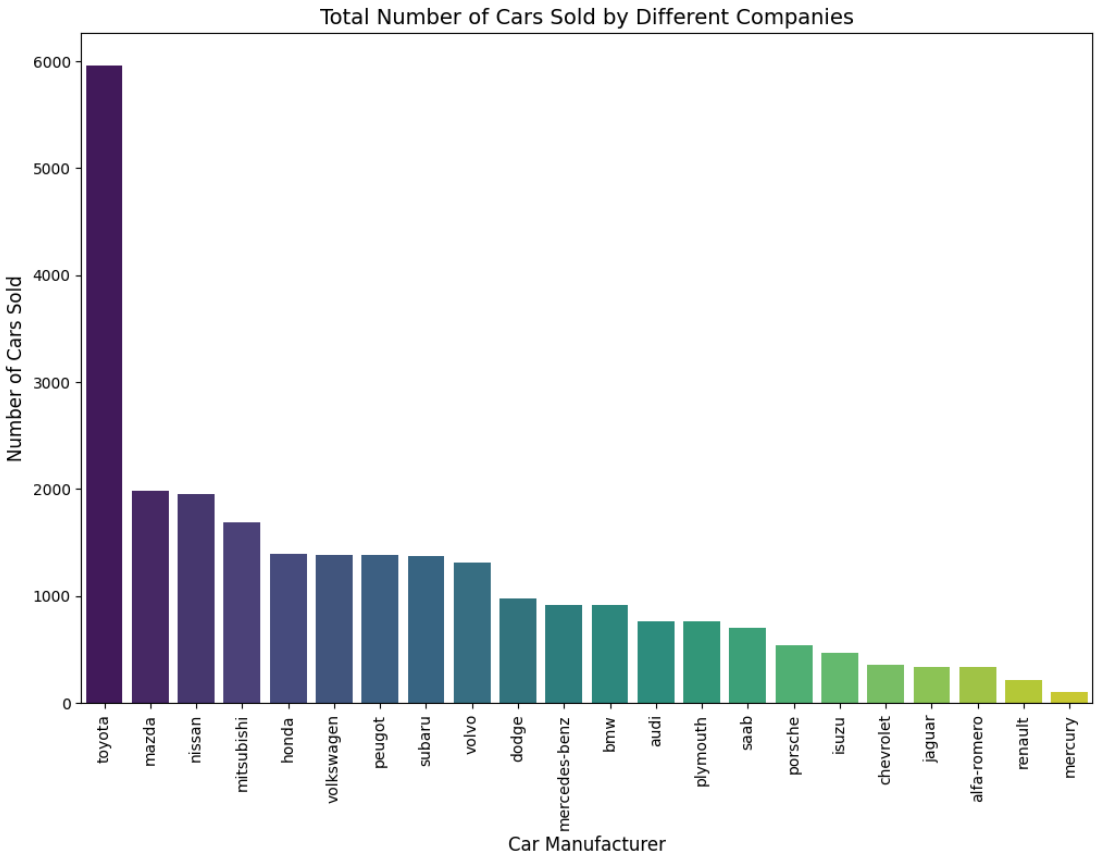


This is a box-plot representing the distribution of prices. Here's a breakdown:

- **Box:** The green box represents the **interquartile range (IQR)**, which contains the middle 50% of the data (from the 25th percentile to the 75th percentile). The line inside the box shows the **median** (the middle value).
- **Whiskers:** The horizontal lines extending from the box show the range of prices within 1.5 times the IQR from the box.
- **Outliers:** The dots beyond the whiskers represent **outliers**, which are prices that are unusually high compared to the rest of the data.

Observation: Most of the prices are concentrated between 5,000 and 15,000. There are several outliers, indicating some prices are significantly higher, going up to 45,000. This confirms the right-skewed nature of the data.

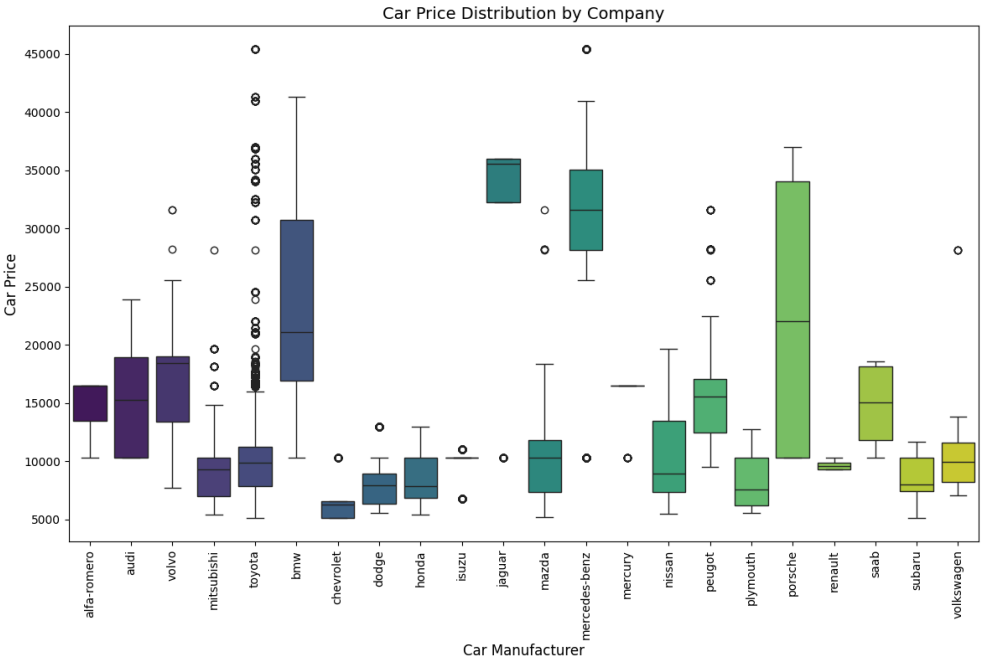
- Explored total cars sold by the manufacturer and their price distribution using bar-plot and box-plot.



This **histogram** shows the number of cars sold by different manufacturers

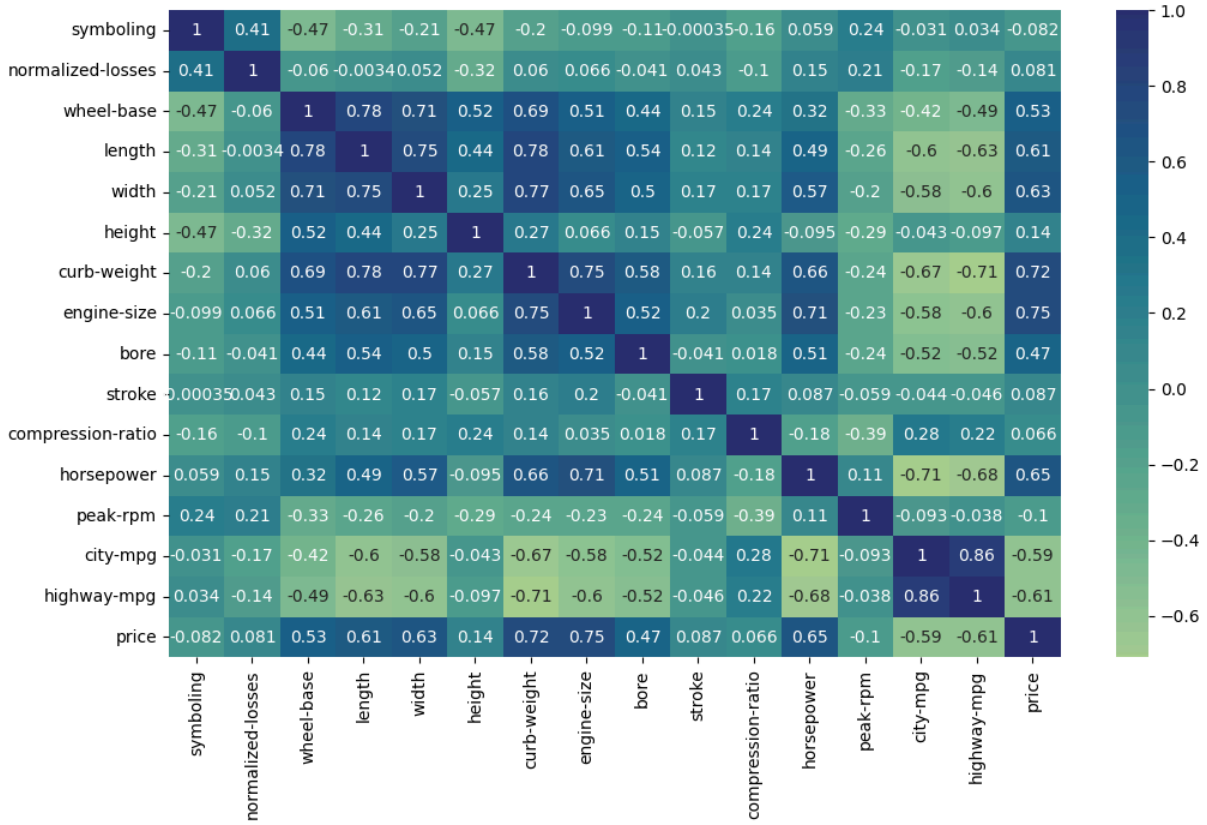
- **axis (Price):** Represents the different car manufacturers.
- **Y-axis (Number of sold cars):** Indicates the number of vehicles sold

Observation: Toyota sells the highest number of cars whereas Mercury sells the lowest the distribution shows from the highest to lowest number of cars sold by different manufacturers



This box plot visualizes the distribution of price by car company (make)
Observation: The box plot reveals significant variation in car prices across manufacturers. Luxury brands like **Jaguar**, **Mercedes-Benz**, and **Porsche** dominate the high-price segment with compact ranges, while budget-friendly brands like **Suzuki** and **Volkswagen** have lower prices and narrower ranges.
Brands like **Toyota** and **Chevrolet** show diverse lineups with wide price ranges and numerous outliers, indicating offerings that cater to both budget and premium markets. Outliers in brands such as **Jaguar** and **Toyota** suggest specialized or high-end models. Overall, the plot highlights clear distinctions between economy and luxury brands, along with variability in pricing strategies.

- Correlation heat map was plotted to analyze relationships between numerical features and price.



This is a heat map that displays the correlation matrix for various features of a data set, including the target variable price. Here's a summary of the key insights from the heat map:

General Observations:

Correlation Strengths:

- Values closer to **1** (dark blue) indicate a strong positive correlation.
- Values closer to **-1** (dark green) indicate a strong negative correlation.
- Values around **0** indicate little to no correlation.

Diagonal Line: The diagonal of the heat map shows a perfect correlation (**1**) since every feature is perfectly correlated with itself.

Key Insights About price:

Strong Positive Correlations:

- engine-size (~0.87): Larger engine sizes are strongly associated with higher prices.
- curb-weight (~0.83): Heavier vehicles are positively correlated with price.
- horsepower (~0.81): More powerful engines correlate with higher prices.

Negative Correlations:

- highway-mpg (~-0.71): Higher fuel efficiency on highways is negatively correlated with price, meaning fuel-efficient cars tend to be less expensive.
- city-mpg (~-0.69): Similar trend as highway-mpg.

Moderate Correlations:

- width (~0.76) and length (~0.75): Larger dimensions (wider and longer cars) are moderately correlated with higher prices.
- normalized-losses (~0.08): Weak correlation with price, indicating it's not a strong predictor.

Other Features:
- **compression-ratio** has a weak or no clear correlation with most features.
- **length and width** (~0.75), which suggests that larger cars tend to have proportional dimensions.
- **curb-weight and engine-size** (~0.77), indicating heavier cars tend to have larger engines.

Takeaway:
The features with the strongest correlations to price are engine-size, curb-weight, and horsepower (positive correlations) and highway-mpg and city-mpg (negative correlations). These are likely the most significant predictors of price in this data-set.

3. **Preprocessing Data**:

Before doing data processing data is split into Separate features and target variables. Preprocessing involves cleaning and transforming raw data to prepare it for modeling. This includes handling missing values (e.g., using mean, median, or mode for imputation) and scaling numerical features (e.g., using StandardScaler). Categorical features are encoded (e.g., OneHotEncoder) to convert them into numerical format. The processed data is then split into training and test sets for model evaluation.

4. **Data Modeling**:

Modeling involves training machine learning algorithms on the preprocessed training data to make predictions. Different regression models like Linear Regression, Decision Tree, KNN, Random Forest, and Gradient Boosting are used. Each model is trained and tested, and performance metrics like Mean Squared Error (MSE) and R^2 score are calculated to evaluate and compare their effectiveness on the test data.
Comparison between different models:

	Model	MSE	R2 Score
0	Linear Regression	1.202282e+07	0.776765
1	KNearestNeighbour	1.081870e+07	0.799123
2	Decision Tree	1.786026e+07	0.668378
3	Random Forest	1.118239e+07	0.792370
4	Gradient Boosting	9.322795e+06	0.826898

Analysis

1. **Gradient Boosting** performs the best with the lowest MSE (9,322,795) and the highest R^2 score (0.826898), indicating it makes the most accurate predictions and explains the variance in the data well.
2. **K-Nearest Neighbour (KNN)** also performs well, with the second-lowest MSE (10,818,700) and a good R^2 score (0.799123). It's slightly less effective than Gradient Boosting.
3. **Random Forest** has a slightly higher MSE (11,182,390) compared to KNN but maintains a strong R^2 score (0.792370), showing consistent performance.

4. **Linear Regression** performs reasonably well with an MSE of 12,022,820 and an R^2 score of 0.776765. However, it is outperformed by the non-linear models.
5. **Decision Tree** has the highest MSE (17,860,260) and the lowest R^2 score (0.668378), indicating it is the least effective model for this task.

Conclusion

Gradient Boosting is the most suitable model for this dataset as it achieves the best balance of accuracy (low MSE) and explanatory power (high R^2). KNN and Random Forest are also strong contenders, but Decision Tree and Linear Regression are less effective in this case. Depending on the context, Gradient Boosting should be the preferred choice for making predictions.