**Project report on**

**Price Prediction for Used Cars**

**Submitted By**

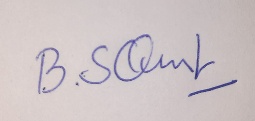
Group No. 4 - Batch: JAN 2025

**Group Members**

1. Mr. Anubhav V,
2. Ms. Agnes Louis,
3. Mr. Afrith H,
4. Mr. Sanjith B,
5. Ms. Pooja P,
6. Mr. Gajarajan V Y

**Research Supervisor**

Mr. P. V. Subramanian





Mr. Sanjith B Mr. P. V. Subramanian

Name and signature of the team leader Name and Signature of the mentor

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Sl NO** | **Topic** | **Page No** |
| 1 | Summary of problem statement, data and findings | 5 |
| 2 | OVERVIEW OF THE FINAL PROCESS | 5 |
| 3 | STEP BY STEP WALK-THROUGH OF THE SOLUTION | 6 |
| 4 | MODEL EVALUATION | 21 |
| 5 | COMPARISON TO BENCHMARK | 26 |
| 6 | VISUALIZATIONS | 26 |
| 7 | IMPLICATIONS | 27 |
| 8 | LIMITATIONS | 28 |
| 9 | CLOSING REFLECTIONS | 29 |
| 10 | APPENDIX | 31 |

**LIST OF FIGURES AND TABLES**

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **Figures** | **Page No.** |
| 3.1 | OpenRefine Interface for Creating a Project by Importing Data | 9 |
| 3.2 | OpenRefine Dialog for Splitting a Column by Separator | 9 |
| 3.3 | OpenRefine Data View After Splitting "engine" Column | 10 |
| 3.4 | Distribution for continuous numerical variables | 11 |
| 3.5 | Count plots for discrete numerical variables | 11-12 |
| 3.6 | Grouped Scatter plot for Categorical variables | 13 |
| 3.7 | Groupedcount plot for Categorical variables | 14 |
| 3.8 | Heatmap for numerical variables | 15 |
| 3.9 | Boxplot for numerical variables | 16 |
| 3.10 | ’Agostino’s K² Test results | 18 |
| 3.11 | Chi-square test validity | 18 |
| 4.1 | RMSE,Explained Variance,COX-R-SQUARE  across 10 Folds ( GLM ). | 23 |
| 4.2 | RMSE across 10 Folds ( Default Parameters). | 24 |
| 4.3 | Explained Variance across 10 Folds ( Default Parameters). | 24 |
| 4.4 | RMSE across 10 Folds ( Tuned Parameters). | 25 |
| 4.5 | Explained Variance across 10 Folds ( Tuned Parameters) | 26 |

**List of Tables**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **Tables** | **Page No.** |
| 3.1 | Count of Zero values | 8 |
| 3.2 | Chi-square independence test results | 20 |
| 4.1 | Models used for this project | 22 |
| 4.2 | Evaluation Metrics(Before hyperparameter Tunning) | 23 |
| 4.3 | Evaluation Metrics(After hyperparameter Tunning) | 25 |
| 7.1 | Table showing the variable importances | 28 |
| 10.1 | Reference to the dataset | 30-31 |
| 11.1 | Data Dictionary | 31 |
| 11.2 | OS Details | 32 |

# **SUMMARY OF PROBLEM STATEMENT, DATA AND FINDINGS**

In the automobile industry, the prediction of used cars has become a key focus area, with machine learning (ML) and artificial intelligence (AI) playing an increasingly vital role enabling buyers to make informed decisions and sellers to optimize the returns.

There is a need for a machine learning model that will accurately predict prices of the used cars for ensuring fair transactions, optimizing returns, and improving overall efficiency in the used car market.

**Data and findings**

The dataset contains the sales records of used cars for predicting the price of used cars. The data includes car brand, model, model\_year, milage, fuel\_type, engine, transmission, color of the interior and exterior of the car, accident history of the car, whether the car has a clean title or not. Detailed description is available in the Data dictionary placed in the subsection section 11.1 of Section 11 Appendix. There are 188533 rows and 13 columns in the dataset.

Target variable is **price**. We need to use the dataset to predict the **price** of the used cars.

We have employed the following machine learning models to predict the target variable:

1. Linear Regression
2. GLM
3. Decision Tree (CART)
4. Random Forest
5. KNN
6. XGBoost
7. CatBoost
8. LightGBM

The explained variance score represents the proportion of variance in the dependent variable that is explained by the independent variables in the model. Unlike R-squared, which quantifies explained variance, RMSE provides a direct measure of prediction error in the same units as the response variable. While the explained variance score is valuable, it's not the only metric to consider when evaluating non-linear models. Metrics like the mean squared error, RMSE (root mean squared error), and others should also be considered to get a comprehensive view of the model's performance.

# **OVERVIEW OF THE FINAL PROCESS**

**2.1 Overview of the Final Process**

Our methodology for predicting used car prices aimed to deliver a transparent, data-driven model to enhance pricing fairness and market efficiency. The approach integrated streamlined data preprocessing, feature engineering, diverse machine learning algorithms, and robust evaluation techniques. Below is a concise summary of the process, data features, preprocessing steps, algorithms, and combined techniques:

Salient Data Features: The dataset included 188,533 rows and 13 columns, with features like brand, model, model\_year, mileage, fuel\_type, engine, transmission, exterior/interior color, accident history, clean\_title, and price (target variable). It had missing values (e.g., 17.64% in HP, 11.36% in clean\_title), high-cardinality columns (model, engine), and skewed mileage and price distributions.

**2.2 Data Preprocessing :**

Data preprocessing involved MICE imputation, null correction, and feature drops (e.g., clean\_title). Engine fields were parsed via OpenRefine to extract horsepower, Litres, and fuel type. New features (fuel\_type\_new, color\_category) were added. Categorical variables were label-encoded; mileage and price normalized via MinMaxScaler. RFE selected 10 predictors. Outliers were capped using Winsorization.

Seven models captured diverse patterns: GLM (linear), CART (non-linear splits), KNN (similarity), Random Forest (ensemble), XGBoost (boosting), CatBoost (categorical), and LightGBM (efficient boosting). This ensured robust price prediction.

**2.3 Combination of Techniques:**

Model Training and Tuning: Models were trained with default parameters initially; GLM achieved Pseudo R² of 70% and RMSE of 0.130, CART had RMSE of 0.1517, KNN had RMSE of 0.1226. Hyperparameter tuning optimized Random Forest, KNN, CART, XGBoost, CatBoost, and LightGBM, while GLM used default settings for simplicity.

**Validation: 10-fold cross-validation ensured robustness and minimized overfitting.**

Evaluation Metrics: Used RMSE and Explained Variance Score; post-tuning, CatBoost and LightGBM led (RMSE ~0.1110, Explained Variance ~0.686), followed by Random Forest and XGBoost, with GLM, CART, and KNN as baselines.

EDA Integration: EDA identified key price drivers (mileage, model\_year, brand, cylinders, interior color, color\_category, cylinder\_layout), guiding feature engineering and model development.

Model Synergy: GLM provided linear insights, CART and KNN captured non-linear and similarity-based patterns, and ensemble models (Random Forest, XGBoost, CatBoost, LightGBM) excelled in complex relationships, ensuring balanced predictions.

This methodology combined efficient preprocessing, targeted feature engineering, and a diverse model set—GLM, Decision Tree (CART), KNN, Random Forest, XGBoost, CatBoost, and LightGBM—to deliver accurate price predictions, fostering transparency and informed decision-making in the used car market.

# **STEP -BY-STEP WALK THROUGH OF THE SOLUTION**

**1. Define the Goal**

The objective of this project was to predict the price of used cars using historical sales data. The aim was to create a transparent, data-driven model to help buyers and sellers make informed decisions, reduce pricing inconsistency, and enhance market efficiency.

**2. Get and Understand the Data**

* **Dataset size:** 188,533 rows and 13 columns.
* **Key features:** brand, model, model\_year, milage, fuel\_type, engine, transmission, exterior color, interior color, accident history, clean\_title, and price (target variable).
* **Numerical columns**

The dataset includes six numerical columns:

1. **id**: Unique identifier (not used for modeling).
2. **model\_year**: Year the car was manufactured (0 missing).
3. **milage**: Total mileage of the car (0 missing).
4. **price**: Sale price of the car, target variable (0 missing).
5. **HP**: Horsepower (33,251 missing).
6. **Litres**: Engine displacement (6,730 missing)

* **Categorical Columns**

The dataset includes seven categorical columns:

1. **brand:** 59 unique classes (e.g., Toyota, Ford; 59 missing).
2. **model:** 59 unique classes (high cardinality; 59 missing).
3. **fuel\_type:** 7 unique classes (e.g., Gasoline, Diesel; 5,864 missing).
4. **engine:** 1,117 unique classes (e.g., engine specifications; 925 missing).
5. **ext\_col:** 319 unique classes (exterior colors; 455 missing).
6. **int\_col:** 156 unique classes (interior colors; 4,537 missing).
7. **accident:** 2 unique classes (None, At least 1 accident; 2,452 missing).
8. **clean\_title:** 1 unique class (Yes; 21,419 missing).
9. **fuel\_type\_new:** Derived column (861 missing).
10. **color\_category:** Derived column (0 missing).
11. **transmission:** 52 unique classes (e.g., Automatic, Manual; 87 missing).
12. **cylinder\_layout**: 5 unique classes (Unknown, V-engine, Inline/Straight, Boxer, W-engine; 0 missing).

* **Observations:**
  + Missing values in multiple columns (e.g., HP – 17.64%, clean\_title – 11.36%, int\_col – 2.41%, fuel\_type – 3.11%).
  + Presence of high-cardinality categorical columns (model, engine) and skewed continuous variables like mileage and price.
  + No duplicate rows.

**3. Pre-process the Data (Cleaning & Transformation)**

* **Missing Value Treatment:**
  + The dataset contains missing values in several columns, as shown

| **#** | **Column** | **Count of Nulls** | **% of Null Values** |
| --- | --- | --- | --- |
| 1 | brand | 59 | 0.03% |
| 2 | model | 59 | 0.03% |
| 3 | fuel\_type | 5,864 | 3.11% |
| 4 | engine | 925 | 0.49% |
| 5 | ext\_col | 455 | 0.24% |
| 6 | int\_col | 4,537 | 2.41% |
| 7 | accident | 2,452 | 1.30% |
| 8 | clean\_title | 21,419 | 11.36% |
| 9 | HP | 33,251 | 17.64% |
| 10 | Litres | 6,730 | 3.57% |
| 11 | fuel\_type\_new | 861 | 0.46% |
| 12 | transmission | 87 | 0.05% |
| 13 | Cylinders | 9,809 | 5.20% |

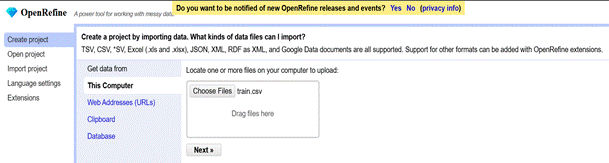
**Tab.3.1.** Count of Zero values

* + Multiple Imputation by Chained Equations (MICE) used for both numerical and categorical columns.
  + Invalid entries like “–” in fuel\_type converted to null and imputed.
* **Redundant Features:**
  + Removed id column as it had no predictive value.
  + Clean\_title dropped or treated as a separate category since it had only one non-null value.
* **Duplicated rows**

No duplicate rows are present in the dataset, ensuring data integrity for modelling.

* **Feature Engineering:**  Splitting columns is a common technique in feature engineering where a single, often composite, column is divided into multiple new columns. This is done when a column contains several distinct pieces of information concatenated together.
* For our project, we utilized **OpenRefine** for this purpose, specifically to split the 'engine' column into its constituent 'HP', 'Litres', and 'fuel\_type' components, thereby making these individual pieces of information more accessible and useful for our model.
* **Importing Data and Splitting 'engine' Column using OpenRefine:**

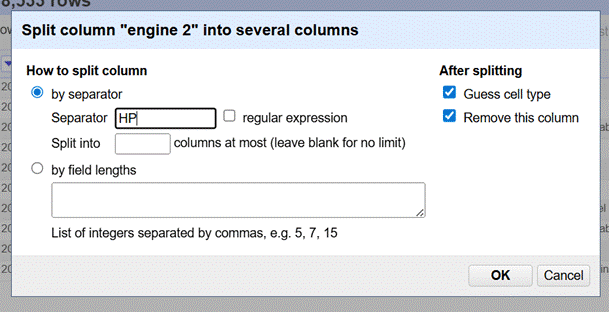
1. **Importing the Data:**
   * Open OpenRefine.
   * Click "Create Project" -> "Get data from Local Disk".
   * Browse and select your "train\_data" file.
   * Follow the prompts to configure parsing options (e.g., character encoding, column separators) and then click "Create Project".



**Fig. 3.1:** OpenRefine Interface for Creating a Project by Importing Data

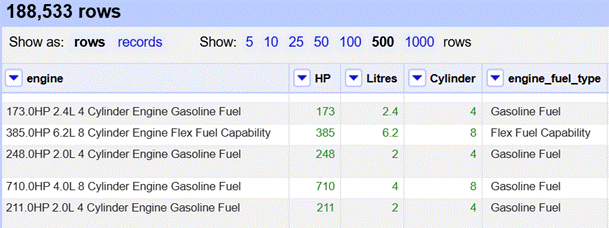
1. **Splitting 'engine' Column:**
   * **Select Column:** In the OpenRefine interface, locate and click the dropdown arrow next to the 'engine' column header.

* + **Choose Split Option:** From the menu that appears, navigate to "Edit column" -> "Split into several columns".



**Fig. 3.2:** OpenRefine Dialog for Splitting a Column by Separator

* + **Define Separator:** In the "Split column" dialog box, in the "Separator" field, type "HP". This action, based on the structure of your data, allowed for the extraction of 'HP', 'Litres', 'Cylinder', and 'engine\_fuel\_type' into new, distinct columns, effectively transforming unstructured engine details into usable features.



**Fig.3.3**: OpenRefine Data View After Splitting "engine" Column

* + Created derived variables such as fuel\_type\_new and color\_category.
* **Encoding and Scaling:**
* **Encoding:** To prepare the categorical features for machine learning models, ***Label Encoding***was applied. This technique assigns a unique integer to each category, transforming the non-numerical categorical data into a numerical format that is compatible with most machine learning algorithms.
* **Scaling:** In Data Processing, we try to change the data in such a way that the model can process it without any problems. And Feature Scaling is one such process in which we transform the data into a better version. Feature Scaling is done to normalize the features in the dataset into a finite range.

The concept of standardization comes into picture when continuous independent variables are measured at different scales. Data scaling is applied to numeric columns. In our dataset we have two continuous numerical

**columns**: milage and price

* While standard and robust scalers are effective for transforming data, they might not be ideal when all feature values must remain non-negative, as is the case with 'price' and 'milage' in our dataset. For these specific variables, a ***MinMaxScaler*** is preferred. This scaler transforms features to a specified range, typically between 0 and 1, by subtracting the minimum value and dividing by the range (max - min). This ensures that all scaled values are positive, maintaining the inherent non-negative nature of the original data, which is crucial for interpretability and compatibility with certain models.
* **Feature Selection:** In regression problems, RFE helps identify the most important features for predicting a continuous target variable by iteratively removing less significant features. It works by building a model, assessing feature importance, and then removing the least important features, repeating this process until the desired number of features is reached.

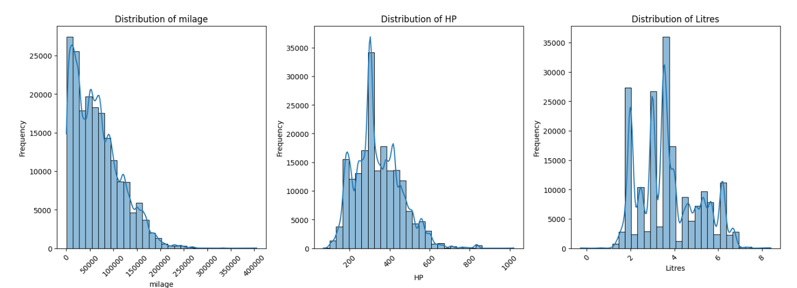
We have selected 10 features which are important in predicting the target variable.

The important features selected for the model are:

1. model\_EDA,
2. model\_year\_EDA,
3. transmission\_EDA,
4. Litres,
5. Cylinders,
6. cylinder\_layout,
7. fuel\_type,
8. color\_category,
9. accident,
10. Milage.

**4. Exploratory Data Analysis (EDA)**

* **Univariate Analysis:**
* **Continuous Numeric**



**Fig.3.4.** Distribution for continuous numerical variables

* Mileage: Distribution is right-skewed, with most cars having lower mileage.
* HP: Concentrated around mid-range horsepower (200–400 HP) with fewer high-performance vehicles.
* Litres: Shows multiple peaks, indicating popular engine sizes around 2.0, 3.0–3.5, and 5.0 litres.
* **Discrete numerical**

A screenshot of a graph

AI-generated content may be incorrect.A graph showing the number of brands

AI-generated content may be incorrect.

A graph showing a number of different colored bars

AI-generated content may be incorrect.A graph with numbers and a number of objects

AI-generated content may be incorrect.

A graph showing different colored squares

AI-generated content may be incorrect.A graph with a bar and a number of objects

AI-generated content may be incorrect.A graph with different colored bars

AI-generated content may be incorrect.A graph with multiple colored squares

AI-generated content may be incorrect.

**Fig.3.5.Count plots for discrete numerical variables**

* **Top 10 Frequencies of model\_year\_EDA**: Most model years are categorized as "Others," but among specific years, recent models (2021, 2018, 2020, 2022) are the most frequent.
* **Top 10 Frequencies of brand\_EDA:** A large proportion of vehicles are from unlisted brands ("Others"), with Ford, Mercedes-Benz, BMW, and Chevrolet being the most represented named brands.
* **Top 10 Frequencies of transmission\_EDA:** Automatic transmissions, particularly 1-speed and 8-speed, are dominant, while manual transmissions are considerably less common.
* **Top 10 Frequencies of fuel\_type:** Gasoline is overwhelmingly the primary fuel type, with hybrid and electric vehicles being far less common**.**
* **Top 10 Frequencies of accident:** The vast majority of vehicles in the dataset have no reported accidents or damage.
* **Top 10 Frequencies of color\_category:** "Luxury" is the most prevalent color category, followed by "Premium" and "Standard," suggesting a leaning towards higher-end vehicle segments.
* **Top 10 Frequencies of cylinder\_layout**: Many cylinder layouts are unknown, but among known types, V-engines are the most frequent, followed by inline/straight engines.
* **Top 10 Frequencies of model\_EDA:** The dataset features a very wide range of car models, with a vast majority falling into the "Others" category, indicating no single model has significant dominance
* **Bivariate Analysis:**
* **Numerical variable Vs Target variable**

A graph of a graph of a model

AI-generated content may be incorrect.A screenshot of a graph

AI-generated content may be incorrect.A graph of different sizes and sizes

AI-generated content may be incorrect.

**Fig.3.6.** **Grouped Scatter plot for Categorical variables**

* **Scatter Plot of HP vs price:** While most vehicles fall within a lower price range regardless of HP, higher-priced vehicles tend to have higher horsepower, and there are distinct price tiers at higher HP levels.
* **Scatter Plot of Litres vs price (first plot):** Similar to HP, most cars are lower priced regardless of engine size in litres, but top-tier prices are exclusive to vehicles with larger engines (2-6 litres).
* **Scatter Plot of Litres vs price (second plot - duplicate):** Again, while many vehicles are low-priced across engine sizes, the highest prices are observed for vehicles with engine sizes between approximately 2 and 6 litres.
* **Scatter Plot of Cylinders vs price:** Most vehicles are at a lower price point regardless of cylinder count, but the highest-priced vehicles exclusively have 8, 10, or 12 cylinders, and some at 16 cylinders.
* **Scatter Plot of model\_year vs price:** Newer model years (post-2000) generally show a wider range of prices including the highest price points, while older models are predominantly in the lower price ranges.
* **Scatter Plot of milage vs price:** As mileage increases, the price of vehicles generally decreases, with the highest-priced vehicles having very low mileage.
* **Categorical variable Vs Target variable**

A graph of a brand

AI-generated content may be incorrect.A graph of a graph of a graph

AI-generated content may be incorrect.A graph of a number of blue rectangular bars

AI-generated content may be incorrect.A graph of a graph showing the price of fuel

AI-generated content may be incorrect.A graph showing a diagram

AI-generated content may be incorrect.A screenshot of a computer

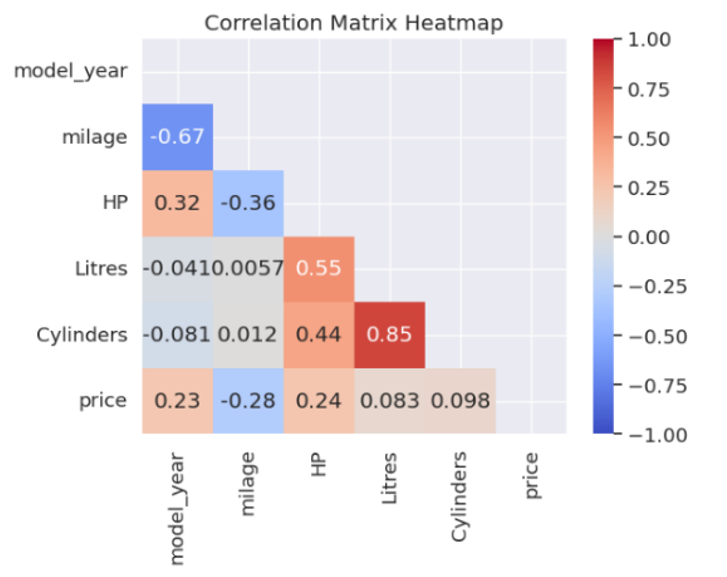
AI-generated content may be incorrect.A graph of different colors

AI-generated content may be incorrect.A graph of blue bars

AI-generated content may be incorrect.

**Fig.3.7. Grouped count plot for Categorical variables**

* **Transmission\_EDA vs. Price:** Automatic transmissions, particularly those with more gears, tend to be slightly pricier, but the variation within automatic types is minimal.
* **Ext\_col\_EDA vs. Price:** Certain exterior colors like "Green" and "Orange" are associated with higher vehicle prices, while "Silver Metallic," "Gold," and "Brown" are linked to lower prices.
* **Cylinder\_layout vs. Price (first plot):** Vehicles with a "W-engine" layout are significantly more expensive than all other cylinder layouts.
* **Brand\_EDA vs. Price:** Porsche cars command the highest average prices, while Toyota and Jeep are among the least expensive brands.
* **Cylinders vs. Price:** A higher number of cylinders generally correlates with a higher average vehicle price.
* **Model\_EDA vs. Price:** Specific high-end models like the Porsche 911 Carrera S and 1500 Laramie drive up average prices significantly.
* **Model\_year\_EDA vs. Price:** Newer model years (e.g., 2023, 2022) tend to have higher average prices than older models.
* **Int\_col\_EDA vs. Price:** "Orange" and "Others" interior colors are associated with higher average prices, whereas "Gray" and "Beige" are linked to lower prices.
* **Cylinder\_layout vs. Price (second plot):** "W-engine" vehicles are substantially more expensive than cars with other cylinder layouts.
* **Fuel\_type vs. Price:** Electric vehicles have the highest average price, with flex-fuel vehicles being the least expensive.
* **Correlation:**



**Fig.3.8.** Heatmap for numerical variables

* Litres & Cylinders (0.85), HP & Litres (0.55), and Model\_year & Milage (-0.67).
* **Outlier Treatment:**
* **Checking for outliers:**

A graph showing a diagram of a mileage

AI-generated content may be incorrect.A graph showing a graph of a graph

AI-generated content may be incorrect.A graph of a graph with lines and dots

AI-generated content may be incorrect.A graph of a price

AI-generated content may be incorrect.

**Fig.3.9.** Boxplot for numerical variables

**Observations**:

* **Litres:** The box plot for 'Litres' shows a relatively compact distribution with a few outliers, including one distinct red outlier, indicating some vehicles might have significantly different engine capacities compared to the majority.
* **HP:** The 'HP' box plot reveals a skewed distribution with a considerable number of outliers extending to much higher horsepower values, suggesting a significant range in vehicle power.
* **milage:** The 'milage' box plot displays a wide spread and numerous outliers, particularly on the higher end, indicating a broad range of vehicle mileage and several vehicles with exceptionally high mileage.
* **price:** The 'price' box plot exhibits a highly skewed distribution with many outliers, especially at the higher price range, highlighting a significant variation in vehicle prices and some very expensive outliers.
* **Treatment of outliers:**
* Outlier Treatment Confirmed: Outliers in milage, HP, Litres, and price have been successfully treated.
* Method Used: The treatment employed was **capping (winsorization)**.
* Evidence of Capping:
  + For milage, HP, and price, their respective maximum values now equal their upper quartiles, indicating upper tail capping.
  + For Litres, its maximum value equals its upper quartile, and its minimum value equals its lower quartile (which is now positive), showing both upper and lower tail capping/correction.
* Current Outlier Status: Based on the provided IQR statistics, **n**one of these variables currently exhibit outliers.

**5. Checking for statistical significance of variable**

1. **Numerical columns:**

The **Shapiro-Wilk test** is used to determine if a sample of data comes from a normally distributed population. It is particularly useful for smaller sample sizes (typically less than 50). Alternatives to the Shapiro-Wilk test **D’Agostino’s K² Test** This test can be used for larger sample sizes (n >= 50).

The **D’Agostino’s K² Test** is a statistical test used to determine whether a dataset follows a normal distribution, and it’s particularly useful when you want to assess both skewness and kurtosis simultaneously.When to Use D’Agostino’s K² Test

* Testing for Normality in Continuous Data Ideal when you need to validate the assumption of normality before applying parametric tests like linear regression, t-tests, or ANOVA.
* When You Suspect Skewness or Kurtosis Unlike tests that focus on one aspect (e.g., Shapiro-Wilk for general shape), D’Agostino’s K² combines skewness and kurtosis into a single omnibus test statistic.
* Moderate to Large Sample Sizes It performs best with n ≥ 20, and its accuracy improves with larger datasets. For very small samples, Shapiro-Wilk might be more reliable.

**Numerical columns:** Model\_year, Milage, HP, Litres, Cylinders, and Price.

Formal statistical tests revealed that none of the numerical columns exhibit a Gaussian (normal) distribution, with all p-values being 0.000.

* **Consistent Non-Normality**: The null hypothesis of normality was rejected for model\_year, milage, HP, Litres, Cylinders, and price.
* **Implications for Analysis**: This finding is crucial as many statistical methods and machine learning models assume normally distributed data.
* **Next Steps**: For future analysis, it will be necessary to consider data transformations (e.g., logarithmic) or utilize non-parametric statistical tests and models that do not rely on the assumption of normality.

A screenshot of a computer code

AI-generated content may be incorrect.

**Fig. 3.10.** D’Agostino’s K² Test results

1. **Categorical** **variables:**

The χ2 - (Chi Sqaure) test of independence analysis utilizes a cross-tabulation table between the variables of interest r rows and c columns.

A chi-square test is not appropriate for analyzing the relationship between categorical and numerical variables.

* **Conversion of Numerical Target Variable to Categorical**

To use the Chi-square test, our original numerical target variable was transformed into a categorical (binned) format. This means we grouped its values into distinct categories (e.g., 'Low', 'Medium', 'High'). This conversion is essential because the Chi-square test works only with categorical data to assess relationships.

A graph of a distribution of binned target variable

AI-generated content may be incorrect.

**Fig.3.11.** Chi-square test validity

Based on the cell counts, it is possible to test if there is a relationship, dependence, between the variables and to estimate the strength of the relationship.

* **Chi-square Test Assumptions**

We've confirmed that the key assumptions for the Chi-square test are met for our analysis:

**Categorical Data:** Both the independent variables and our newly binned target variable are categorical, fulfilling this requirement.

**Independent Observations:** We assume our data collection ensured that each observation is independent, meaning one data point doesn't influence another. This is typically achieved through proper sampling methods.

**Expected Frequencies:** We have checked each categorical independent variable with our target categorical variable using Chi Square test of independence. We observed the assumption for chi-square test of independence (No more than 20% of the cells have and expected cell count < 5) is satisfied.

Each of the independent variables, brand\_EDA, model\_EDA, transmission\_EDA, cylinder\_layout, fuel\_type, ext\_col\_EDA, int\_col\_EDA, color\_category, accident, model\_year\_EDA and the target variable, target\_binned are dependent.

**Assumptions**

* The two samples are independent
* No expected cell count is = 0
* No more than 20% of the cells have and expected cell count < 5

**Hypothesis**

Null hypothesis H0: Variables are independent

Alternative hypothesis H1: Variables are NOT independent

Since a chi-square test is not appropriate for analyzing the relationship between categorical and numerical variables, we can convert the numerical target variable into a categorical variable.

* Converting a continuous variable into discrete bins may lose information and introduce bias if not done carefully.
* After binning, check the distribution of observations across categories to ensure validity for the chi-square test.

**Categorical Columns :** brand, model, fuel\_type, engine, ext\_col, int\_col, accident, clean\_title, fuel\_type\_new, color\_category, transmission, and cylinder\_layout.

Chi-Square tests were conducted for several categorical variables against a **binned target variable.**

**Chi-square independence test**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Variable** | **P\_ value** | **% of expected frequency cells having cell count < 5** |
| 1 | brand\_EDA | 0% | 0% |
| 2 | model\_EDA | 0% | 0% |
| 3 | transmission\_EDA | 0% | 0% |
| 4 | cylinder\_layout | 0% | 0% |
| 5 | fuel\_type | 0% | 0% |
| 6 | ext\_col\_EDA | 0% | 0% |
| 7 | int\_col\_EDA | 0% | 0% |
| 8 | color\_category | 0% | 0% |
| 9 | accident | 0% | 0% |
| 10 | model\_year | 0% | 0% |

**Tab.3.2.** Chi-square independence test results

* **Assumption Met:** For all variables, 0.00% of expected cell counts were less than 5, satisfying a key assumption of the χ2 test.
* **Significant Dependence:** All tests yielded a p-value of 0.000, indicating that every independent categorical variable tested (brand\_EDA, model\_EDA, transmission\_EDA, cylinder\_layout, fuel\_type, ext\_col\_EDA, int\_col\_EDA, color\_category, accident, model\_year\_EDA) is statistically dependent on the binned target variable.
* **Implication:** These variables are all significantly related to the target and are therefore strong candidates for inclusion in predictive models.

**6. Build Models**

The following models were built:

* Linear Regression
* Generalized Linear Model (GLM – Gaussian, identity link)
* Decision Tree (CART)
* Random Forest
* K-Nearest Neighbors (KNN)
* XGBoost
* CatBoost
* LightGBM

**7. Model Evaluation**

* **Validation Strategy:** 10-fold cross-validation.
* **Base Model:** GLM produced Pseudo R² = 70%, with RMSE = 0.1334 (train) and 0.1348 (test), indicating no overfitting.
* **Hyperparameter Tuning:** Applied to models (Decision Tree, Random Forest, K-Nearest Neighbors (KNN), XGBoost, CatBoost, LightGBM) for better performance.
* **Metrics Used:** RMSE and Explained Variance Score for model comparison.

**8. Insights from Best Model and EDA**

* **Key Drivers of Price:** mileage, model\_year, brand, cylinders, interior color, color category, and cylinder layout.
* **Market Insight:**
  + Automatic transmission and electric vehicles fetch higher prices.
  + Low-mileage and newer model year vehicles command premium pricing.
* **Impact:** The best-performing ensemble models improved predictive accuracy and provided actionable insights for pricing strategies, supporting transparency and trust in the used car market.

# **MODEL EVALUATION**

Model evaluation was central to the study, focusing on understanding the RMSE metric due to its importance in identifying the best regression model to predict the price of used cars.

1) Linear Regression

2) GLM

3) Decision Tree (CART)

4) Random Forest

5) KNN

6) XGBoost

7) CatBoost

8) LightGBM

**Models Assessed:**

The study evaluated seven machine learning models:

| **#** | **Model** | **Details** |
| --- | --- | --- |
| 1 | Linear Regression | Linear Regression models the linear relationship between a dependent variable (the target or label) and one or more independent variables (features). |
| 2 | GLM | Generalized Linear Models (GLMs) in machine learning are a flexible class of statistical models that extend traditional linear regression to accommodate response variables with error distributions other than the normal distribution. This allows GLMs to model a wider variety of data types and relationships. |
| 3 | K-Nearest Neighbors (KNN) | KNN is a simple algorithm that predicts the output for a new data point based on the similarity (distance) to its nearest neighbors in the training dataset, used for both classification and regression tasks. |
| 4 | Decision Tree (CART) | A decision tree splits data into branches based on feature values, creating a tree-like structure. |
| 5 | Random Forest | Random forest is an ensemble method that combines multiple decision trees. |
| 6 | XGBoost | Gradient Boosting algorithms such as XGBoost, LightGBM, CatBoost build models sequentially, meaning each new model corrects errors made by previous ones. Combines weak learners (like decision trees) to create a strong predictive model. |
| 7 | CatBoost | CatBoost is an open-source gradient boosting library developed by Yandex, designed to handle categorical features effectively and efficiently in machine learning tasks. Its name is a portmanteau of "Categorical" and "Boosting." |
| 8 | LightGBM | Light Gradient Boosting Machine (LightGBM) is an open-source, distributed, high-performance gradient boosting framework developed by Microsoft. It utilizes a tree-based learning algorithm and is widely used in machine learning for tasks such as classification, regression, and ranking. |

**Tab.4.1.** Models used for this project

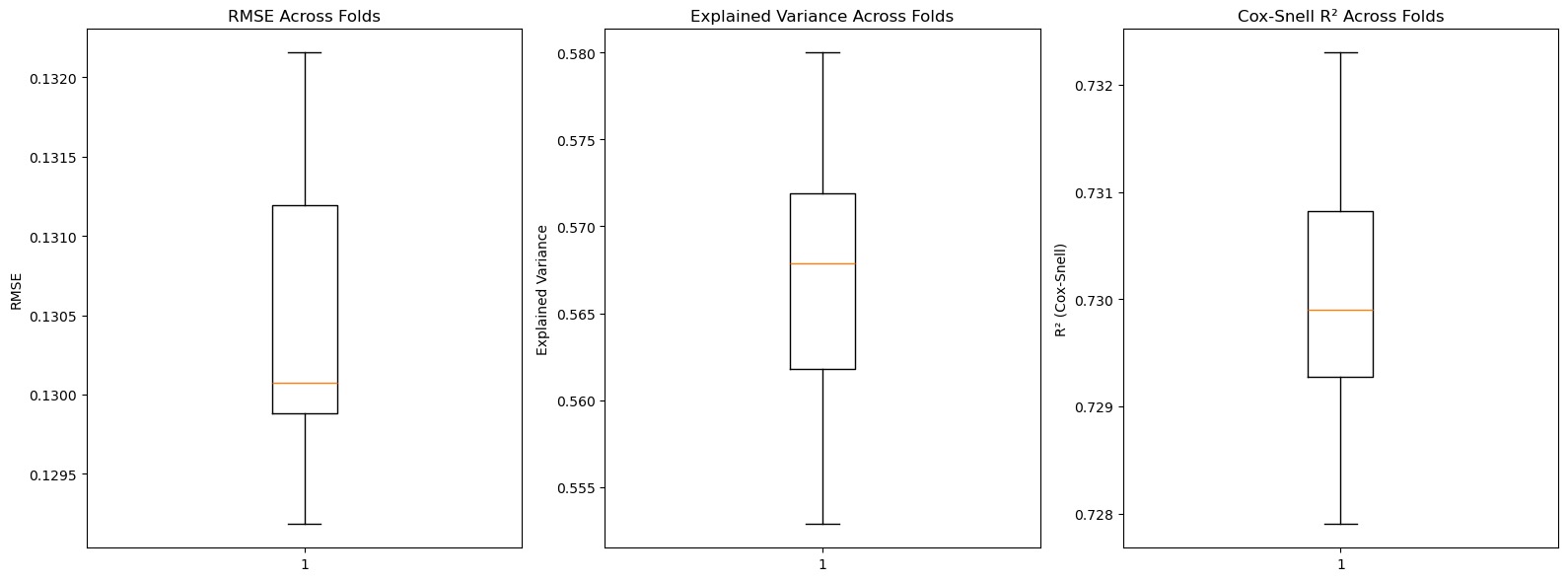
**Evaluation Methodology:**

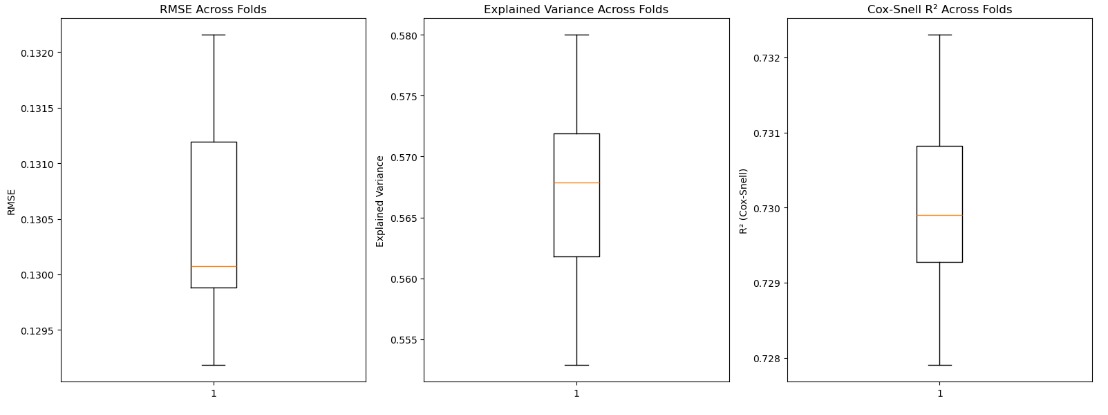
* **K-Fold Cross-Validation**: A 10-fold cross-validation method was used where appropriate to ensure model robustness and prevent overfitting.
* **Default vs. Tuned Parameters**: Models were first assessed with default parameters and later with hyperparameter tuning to optimize performance. Tuning was conducted systematically to balance RMSE improvement with computational efficiency.

1. **Before hyper-parameter tuning the models:**

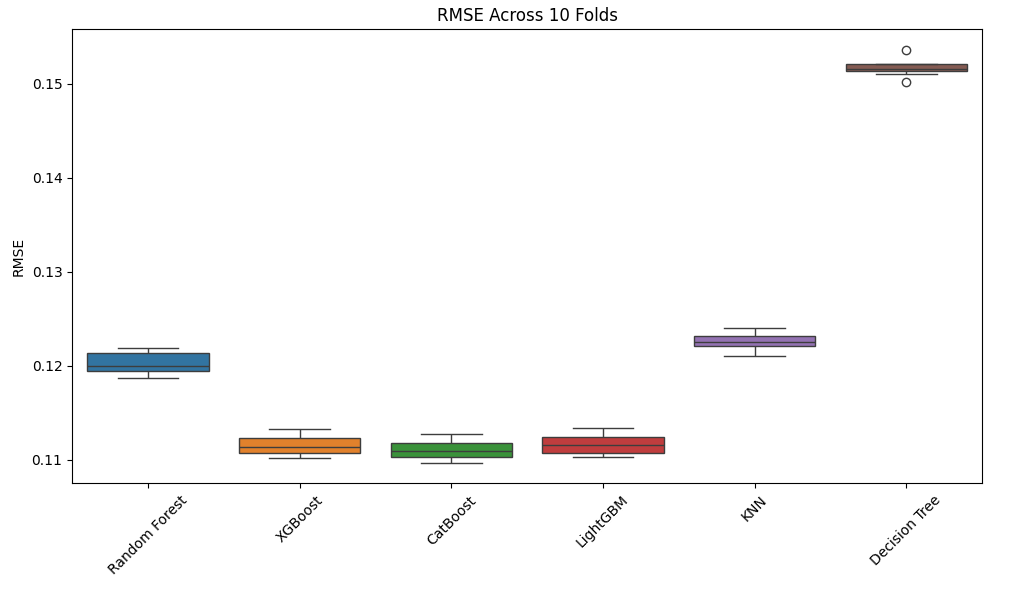
| **#** | **Model** | **RMSE Average ± Std. dev.** | **Explained variance** | **Time Taken (in sec)** |
| --- | --- | --- | --- | --- |
| 1 | GLM | 0.1300 ± 0.0010 | 0.567 | 7.884 |
| 2 | K-Nearest Neighbors (KNN) | 0.1226 ± 0.0008 | 0.6177 | 41.1824 |
| 3 | DecisionTree (CART) | 0.1517 ± 0.0008 | 0.4146 | 11.1196 |
| 4 | Random Forest | 0.1203 ± 0.0011 | 0.6320 | 732.7619 |
| 5 | XGBoost | 0.1115 ± 0.0010 | 0.6839 | 13.0142 |
| 6 | CatBoost | 0.1110 ± 0.0010 | 0.6863 | 270.5055 |
| 7 | LightGBM | 0.1116 ± 0.0010 | 0.6831 | 23.6379 |

**Tab.4.2.** Evaluation Metrics(Before hyperparameter Tunning)

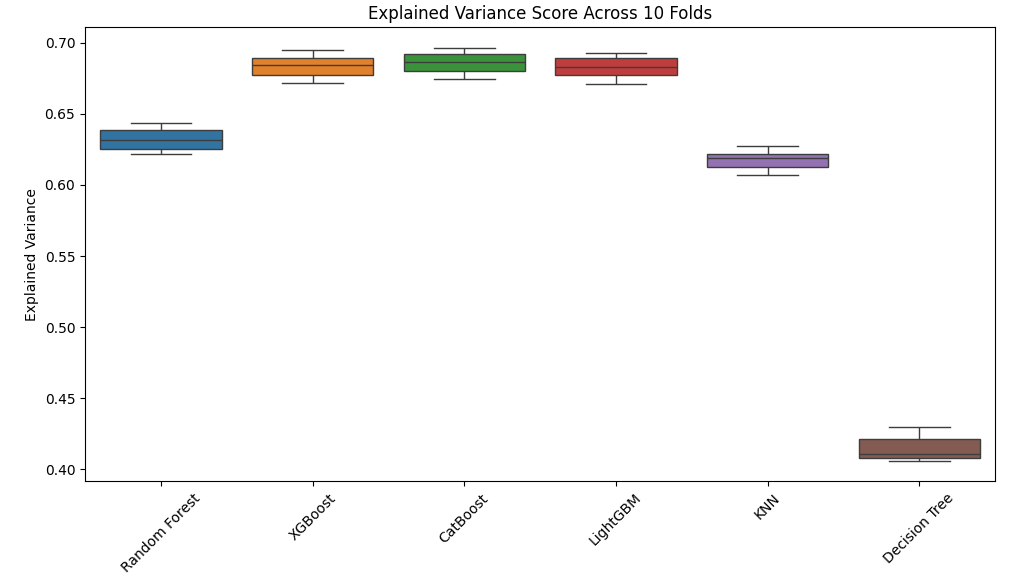




**Fig.4.1.** RMSE,Explained Variance,COX-R-SQUARE across 10 Folds ( GLM ).



**Fig.4.2.** RMSE across 10 Folds ( Default Parameters).

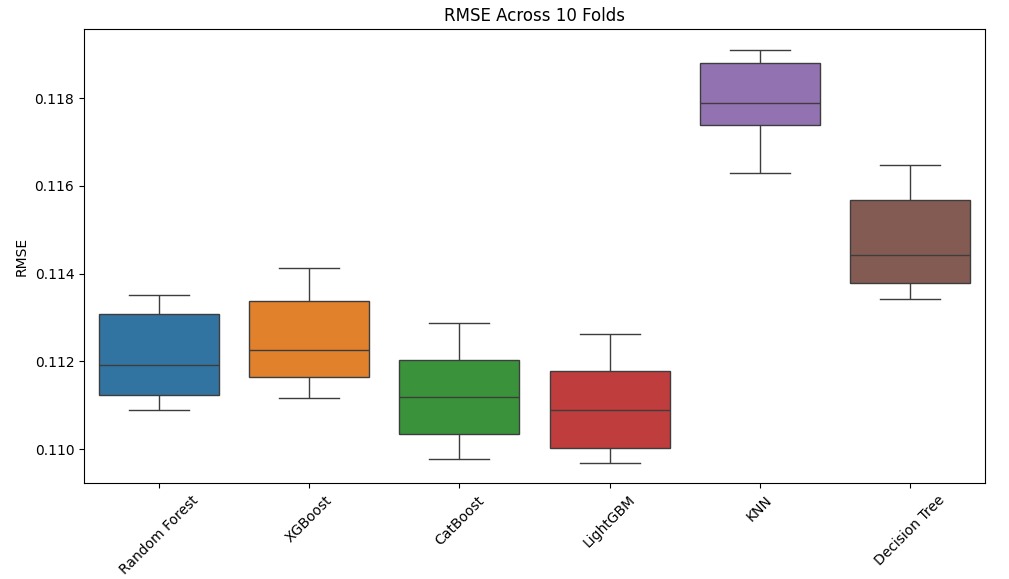


**Fig.4.3.** Explained Variance across 10 Folds ( Default Parameters).

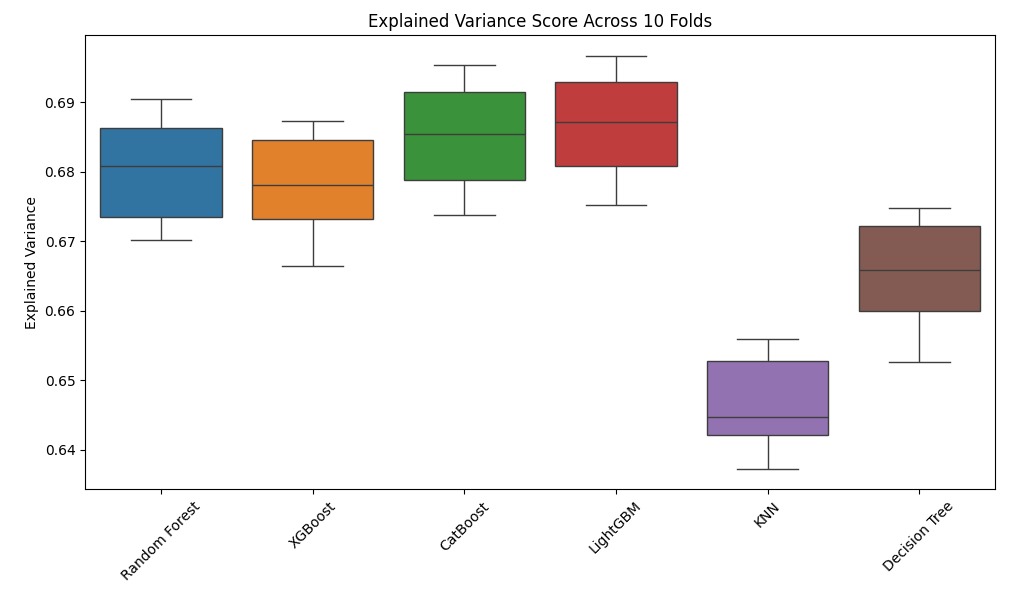
1. **After hyper-parameter tuning the models:**

| **#** | **Model** | **RMSE Average ± Std. dev.** | **Explained variance** | **Time Taken (in sec)** |
| --- | --- | --- | --- | --- |
| 1 | K-Nearest Neighbors (KNN) | 0.1179 **±** 0.0009 | 0.6463 | 545.7488 |
| 2 | Decision Tree (CART) | 0.1147 **±** 0.0010 | 0.6654 | 4.9811 |
| 3 | Random Forest | 0.1121 **±** 0.0010 | 0.6802 | 394.2421 |
| 4 | XGBoost | 0.1125 **±** 0.0010 | 0.6782 | 67.2635 |
| 5 | CatBoost | 0.1112 **±** 0.0010 | 0.6854 | 137.1177 |
| 6 | LightGBM | 0.1110 **±** 00.0010 | 0.6868 | 51.4159 |

**Tab.4.3.** Evaluation Metrics(After hyperparameter Tunning)



**4.4.** RMSE across 10 Folds ( Tuned Parameters).



**4.5.** Explained Variance across 10 Folds ( Tuned Parameters).

# **COMPARISON TO BENCHMARK**

The performance of the models was benchmarked using Base model, GLM on the same dataset.

Most of the important assumptions of linear regression have failed. We use generalized linear models when our residuals are not normally distributed but we might also use them when our data are non-linear. We observe that the mean explained variance is 0.567 and Mean and Std. deviation is 0.130 ± 0.001.

After performing hyper-parameter tuning using Bayesian optimization over hyper parameters and K Fold cross validation, we observe that the best model is **LightGBM** giving the highest explained variance of 0.6867 and the lowest RMSE, average RMSE and std. deviation as 0.1110 ± 0.0010. **LightGBM**, a gradient boosting framework, is frequently used for predicting used car prices due to its efficiency and accuracy. It can handle large datasets and a variety of features, making it suitable for the complexities of the used car market.

# **VISUALIZATION(S)**

**Model Comparison Chart:** **LightGBM** with tuned parameters showed the most significant improvements, validating their suitability for this problem.

A chart with different colored boxes

AI-generated content may be incorrect.

**Fig.6.1.** Boxplot comparison of RMSE measure across various models

# **IMPLICATIONS**

We observed that the features, **milage**, **brand\_EDA, model\_year\_EDA, Cylinders** are the top four most influencing factors for predicting the price of used cars.

A graph with blue bars

AI-generated content may be incorrect.

**Fig.7.1.** Barplot showing the variable importance plot

|  |  |
| --- | --- |
| **Feature** | **Importances\*** |
| Milage | 0.3 |
| brand\_EDA | 0.14 |
| Model\_year\_EDA | 0.12 |
| Cylinders | 0.11 |
| int\_col\_EDA | 0.09 |
| color-Category | 0.06 |
| Cylinder\_layout | 0.05 |
| Model\_year\_EDA | 0.05 |
| Model\_EDA | 0.05 |
| accident | 0.03 |

**Tab.7.1.** Table showing the variable importances

Note:\* It is the normalized score of the total impurity or variances reduction attributed to each feature over all trees.

**We observe that**

* Lower **mileage** typically correlates with higher resale value due to reduced wear and tear.
* **Brand** reputation plays a major role in perceived reliability and desirability.
* **Newer models** tend to command higher prices due to updated features and lower depreciation.
* The **number of cylinders** affects engine performance and fuel efficiency. While, more cylinders may imply power, they can also signal higher fuel and maintenance costs, which may reduce appeal in certain markets.

Ref: Driven by Data: Analyzing Price & Trends in The Used Car Market <https://www.ijfmr.com/papers/2025/3/46706.pdf>

# **LIMITATIONS**

* **Missing Data Imputation Bias:**

MICE imputation, while robust, may introduce bias in predictions if the missing data patterns differ significantly from reality.

* **Model Generalizability:** The models’ performance may vary across different datasets, particularly those with cars from different continents & countries.
* **Limited Feature Scope:** While the selected features were impactful, the exclusion of other potentially important variables including but not restricted to:
  + Seller & Market factors such as location (Urban vs. Rural markets), Listing Platform
  + Aesthetic & Comfort Features such as Infotainment system, Sunroof, Leather Seats
  + Safety Features such as ABS, airbags, parking sensors, and crash ratings

# **CLOSING REFLECTIONS**

This study highlights the transformative potential of machine learning in predicting the price of the used cars by combining robust preprocessing, feature selection, and advanced modeling techniques, the study achieved superior RMSE scores, demonstrating practical applicability.

**Future Directions:**

1. Incorporate modern datasets with recent data to validate and refine the models.
2. Explore ensemble and deep learning methods to enhance performance further.
3. Employ Real-World Collaboration Strategy such as

* **Partner Selection**
  + Target regional dealerships, online platforms like CarDekho or Cars24, and fleet management firms.
  + Prioritize those with historical transaction data and a digital footprint for smoother integration.
* **Pilot Framework**
  + Deploy your models for price suggestions, valuation assistance, or fraud detection in their workflow.
  + Run A/B tests comparing model-generated prices with expert estimates or market closing prices.
* **Feedback Loop**
  + Collect structured input from sales teams, customers, and backend analysts.
  + Track deviations between predicted and actual sale prices to fine-tune features or model weights.
* **Metrics to Monitor**
  + Pricing accuracy (RMSE, Explained Variance)
  + Customer trust and engagement
  + Time to close a sale
  + Feedback sentiment from users
* **Iteration & Expansion**
* Use insights to refine models — maybe explore ensemble stacking or retrain with new regional data.
* After success, consider licensing or SaaS deployment for broader adoption.

1. **References and Bibliography**

1) R. Swarnkar, R. Sawant, H. R and S. P. (2023) Multiple Linear Regression Algorithm-based Car Price Prediction

https://www.researchgate.net/publication/386117316\_Predicting\_Vehicle\_Prices\_Using\_Machine\_Learning\_A\_Case\_Study\_with\_Linear\_Regression

2) Predicting Used Car Prices with Regression Techniques, Saurabh Kumar, Avinash Sinha

Volume 72 | Issue 6 | Year 2024 | Article Id. IJCTT-V72I6P118 | DOI: https://doi.org/10.14445/22312803/IJCTT-V72I6P118

https://ijcttjournal.org/archives/ijctt-v72i6p118

3) Price Prediction of Used Cars Using Machine Learning, Published in: 2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT)

Date of Conference: 22-24 November 2021, Date Added to IEEE Xplore: 08 February 2022, ISBN Information: DOI: 10.1109/ICESIT53460.2021.9696839

Publisher: IEEE

4) Price Prediction of Used Cars Using Machine Learning, Published in: 2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT)

Date of Conference: 22-24 November 2021, Date Added to IEEE Xplore: 08 February 2022, ISBN Information: DOI: 10.1109/ICESIT53460.2021.9696839, Publisher: IEEE

https://ieeexplore.ieee.org/document/9696839

|  |  |
| --- | --- |
| **Original owner of data** | Srinivasa Rao Bittla |
| **Data set information** | The dataset containing sales records of used cars for predicting the price of used cars. The data includes car brand, model, model\_year, milage, fuel\_type, engine, transmission, color of the interior and exterior of the car, accident history of the car, whether the car has a clean title or not. There are 188533 rows and 13 columns in the dataset. |
| **Any past relevant articles using the dataset** | https://www.kaggle.com/code/sasakitetsuya/feature-engineering-ideas-for-car-price-prediction/notebook |
| **Reference** | <https://www.kaggle.com/code/sasakitetsuya/feature-engineering-ideas-for-car-price-prediction/input>  **Citation:**  @misc{used-car-price-prediction-during-inflation,  author = {Srinivasa Rao Bittla},  title = {Used car price prediction during inflation},  year = {2025},  howpublished = {\url{https://kaggle.com/competitions/used-car-price-prediction-during-inflation}},  note = {Kaggle}  }  Linear :  {https://scikit-learn.org/stable/modules/linear\_model.html}  GLM :  {https://www.geeksforgeeks.org/machine-learning/generalized-linear-models/}  {https://www.numberanalytics.com/blog/deep-dive-generalized-linear-models} |
| **Link to web page** | <https://www.kaggle.com/code/sasakitetsuya/feature-engineering-ideas-for-car-price-prediction/input> |

**Tab 10.1** Reference to the dataset

# **ANNEXURE**

## **Data dictionary**

| **#** | **Data column** | **Description** |
| --- | --- | --- |
| 1 | id | Unique identifier for each car entry |
| 2 | brand | Car Brand |
| 3 | model | Car Model |
| 4 | model\_year | Year of the car model |
| 5 | milage | Mileage of the car |
| 6 | fuel\_type | The type of fuel the car uses |
| 7 | engine | Engine specifications, including horsepower and engine size |
| 8 | transmission | Transmission type |
| 9 | ext\_col | Exterior color of the car |
| 10 | int\_col | Interior color of the car |
| 11 | accident | Accident history of the car |
| 12 | clean\_title | Indicates if the car has a clean title |
| 13 | price | The target variable, representing the car's price |

**Tab:11.1 –** Data Dictionary

| **#** | **Parameter** | **Values** |
| --- | --- | --- |
| 1 | sklearn version | 1.6.1 |
| 2 | Python version | 3.11.13 |
| 3 | System Platform | linux |
| 4 | Generic name of the operating system | posix |
| 5 | Platform Architecture | ('64bit', 'ELF') |
| 6 | Total RAM | 13.0:GB |
| 7 | Total Execution time | 0.0 Min: 49.0 Seconds: 55.012992 |

**Tab:11.2 –** OS Details