A Project Report

On

**Vehicle Price Prediction System using ML**

For an Internship Program of 6 Months

**Submitted To:**



**Submitted By:**

Karmendra Bahadur Srivastava

Machine Learning Intern (6 Months)

UMID02042527293

**Made In:**

27th of September, 2025

**Intern Programme Duration:**

APRIL to OCTOBER 2025

**Candidate’s Declaration**

I hereby declare that the project titled "Vehicle Price Prediction System using ML" submitted is an original work carried out by me.

I further declare that the work presented in this project does not violate or infringe on the rights of any third party and is in complete compliance with academic and ethical standards. All sources of information and data have been properly cited and acknowledged.

Furthermore, I declare that this project, including all its contents, structure, code-base, and data processing logic, is my intellectual property and is protected under applicable copyright laws.

This project has been created strictly for academic, educational, and research purposes only. Any reproduction, reuse, distribution, or modification of this project — either partially or fully — is only permitted for non-commercial, fair use, with proper acknowledgment of the original creator.

Karmendra Bahadur Srivastva

[karmendra5902@gmail.com](mailto:karmendra5902@gmail.com)

UMID02042527293

27th of Sept, 2025

**Copyright © 2025**

This project report titled **“Vehicle Price Prediction System using ML”** and all its associated materials, including but not limited to:

* Source code
* Data processing logic
* Documentation
* Visualizations and user interface designs
* Generated content and intellectual insights

are the intellectual property of the Creator ‘Karmendra B Srivastava’.

No part of this project may be reproduced, stored in a retrieval system, transmitted, or distributed in any form or by any means—electronic, mechanical, photocopying, recording, or otherwise—without prior written permission of the author, except in the case of brief quotations embodied in critical reviews and certain other non-commercial uses permitted by copyright law.

This work has been created solely for **academic and educational purposes**. Any unauthorized use of this material for commercial purposes or plagiarism in any form is strictly prohibited and may lead to legal action under applicable copyright laws.

**All Rights Reserved.**

**Acknowledgement**

I would like to express my sincere gratitude to all those who supported and guided me throughout the development of this project titled "Vehicle Price Prediction System using ML".

I am especially thankful to my mentor/supervisor “Unified Mentor” for providing me with the database for training and the instructions throughout the duration of the project.

Furthermore, I am grateful to the open-source developer community and contributors for the tools, libraries, and resources that made this project possible.

Also the Various Sources for Compatibility, Usability and Portability with different systems that made the project as polished and possible to be worked on the future.

Lastly, I acknowledge the support of my institution in providing the education and experience to work on and present this project report.

Karmendra Bahadur Srivastava  
27th, Sept 2025

**Abstract**

In today’s fast-evolving automobile market, consumers often face challenges in evaluating the true value of a vehicle based solely on its specifications. The vast variety of models, diverse configurations, and frequent price fluctuations by manufacturers and dealers make it difficult for buyers to make informed decisions. For researchers and developers, creating robust datasets and tools that simulate such dynamic markets is both essential and complex. There is a growing need for a system that not only predicts vehicle prices with high accuracy but also provides meaningful insights to guide purchasing decisions.

This project addresses these challenges by offering a web-based Vehicle Price Prediction application powered by machine learning models trained on diverse feature sets such as make, model, year, engine specifications, fuel type, transmission, mileage, and drivetrain. Users can input known details of a vehicle to receive an estimated price, explore batch predictions through dataset uploads, and gain deeper insights into the vehicle market. The application also includes interactive analytics such as feature importance, price trends, and similar vehicle suggestions to enhance transparency and decision-making.

Beyond price prediction, the tool offers a user-friendly interface built using Streamlit, integrating visualizations and real-time interaction to improve engagement. It supports personalized inputs such as exterior and interior colors and delivers predictions in a visually appealing format. These features make the application not only a practical analytical tool for buyers, sellers, and researchers but also an informative platform that bridges the gap between data science and real-world vehicle valuation.

**List of Table**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Content** | **Page No** |
| 4.1 | Column's Descriptions | 4 |
| 4.2 | Example of the Dataset | 8 |
| 4.3 | Specified Selected Features helping in Prediction | 10 |
| 4.4 | Model Performance Comparison | 16 |
| 4.5 | Dataset and Columns Statisticss | 17 |
| 4.6 | Sample Input for a Sample Prediction | 42 |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Name of Figure** | **Page No** |
| 5.1 | Data Types in the Datasets | 9 |
| 5.2 | Distribution of the Prices | 18 |
| 5.3 | Correlation Matrix of different features | 19 |
| 5.4 | Top 10 Importance Features helping in Prediction | 20 |
| 5.5 | Price Distribution by Year | 21 |
| 5.6 | Price Distribution by Cylinders | 22 |
| 5.7 | Price Distribution By Doors | 22 |
| 5.8 | Price Distribution by Body Type | 23 |
| 5.9 | SHAP Values on model | 24 |
| 5.10 | Actual vs Predicted Price | 26 |
| 5.11 | Price vs Mileage | 27 |
| 5.12 | Sidebar of the Streamlit Application | 30 |
| 5.13 | Overview of the Application | 31 |
| 5.14 | About Section | 32 |
| 5.15 | Basic Mode | 33 |
| 5.16 | Full Prediction Mode | 34 |
| 5.17 | Batch Prediction Mode | 34 |
| 5.18 | Statistics of the Model | 35 |
| 5.19 | Dataset Explorer Overview | 36 |
| 5.20 | Featured Engineering | 37 |
| 5.21 | Featured Engineering Charts | 37 |
| 5.22 | Result as the Prediction with Price Range | 43 |

**List of Abbreviation**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Word** | **Abbreviation** |
| 6.1 | Artificial Intelligence | AI |
| 6.2 | Streamlit | ST |
| 6.3 | Extreme Gradient Boosting | XGBoost |
| 6.4 | Machine Learning | ML |
| 6.5 | United States Dollars | USD |
| 6.6 | Root Mean Squared Error | RMSE |
| 6.7 | United Mentor | UM |
| 6.8 | Jupyter Notebook | Jupyter |
| 6.9 | Coefficient of Determination | R2 |
| 6.10 | Application Programming Interface | API |
| 6.11 | User Interface/Experience | UI/UX |
| 6.12 | Comma-Separated Values | CSV |
| 6.13 | Application Programming Interface | API |
| 6.14 | Missing At Random | MAR |
| 6.15 | Pickle File | pkl |
| 6.16 | Exploratory Data Analysis | EDA |
| 6.17 | Web Application | Web app |
| 6.18 | SHapley Additive exPlanations | SHAP |

**Table of Content**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Topic** | **Page No** |
|  | Candidate’s Declaration | I |
| 2. | Acknowledgment | III |
| 3. | Abstract | IV |
| 4. | List of Tables | V |
| 5. | List of Figures | VI |
| 6. | List of Abbreviation | VII |
| 7. | Introduction   |  | | --- | | 7.1 Problem Statement | | 7.2 Analyzing the Project | | 7.3 Dataset Overview | | 7.4 Objectives of the Project | | **1**  1  2  3  5 |
| 8. | Data Collection and Preprocessing   |  | | --- | | 8.1 Data Source and Formatting | | 8.2 Features selected | | 8.3 Handling of Data Values | | **7**  7  10  11 |
| 9. | Project Concept and Model Training   |  | | --- | | 9.1 Technology and Model used | | 9.2 Model Training and Evaluation | | 9.3 Visual Insights and Interpretability |   9.4 Comparisons and Analysis | **14**  14  15  18  24 |
| 10. | System Architecture and Implementation   |  | | --- | | 10.1 System Architecture | | 10.2 Streamlit Working and Processing | | 10.3 Access the Program | |  | | **28**  28  37  40 |
| 11. | Project Concept and Model Training   |  | | --- | | 11.1 Sample Prediction and Outcomes | | 11.2 Visualization and Performance | | 11.3 Benefits and Limitations | | 11.4 Achievement and Conclusion | | **41**  41  44  45  46 |
| 12 | References | 48 |

**Introduction**

**7.1 Problem Statement**

In today’s dynamic automobile market, consumers, dealers, and fleet managers face significant challenges in estimating the true value of a vehicle based solely on its specifications. Thousands of models differ in brand, year of manufacture, engine type, mileage, fuel type, and features, while frequent price fluctuations add further complexity. This variability makes it difficult for stakeholders to determine a fair and competitive price without extensive research. For potential buyers and sellers, inaccurate price estimation can lead to financial loss, missed opportunities, or unfair market practices. Similarly, industry analysts and fleet managers require reliable insights to make informed decisions on investments, resale, and fleet optimization.

Manual vehicle valuation is not only time-consuming but also inherently subjective. Traditional methods often rely on personal judgment or outdated reference guides, which can result in inconsistent and unreliable valuations. Human biases, incomplete data, and the inability to process large datasets further reduce the accuracy of conventional approaches. Existing online tools and pricing platforms, while useful, often fall short in providing interactive, comparative, and data-driven insights. They rarely allow users to explore relationships between vehicle attributes and pricing trends in a flexible and intuitive manner.

The **Vehicle Price Prediction Project** addresses this gap by developing a robust, machine learning-powered system capable of predicting vehicle prices with high accuracy. Beyond simple estimation, the system offers comparative analysis of similar models, enabling users to understand the key factors influencing vehicle value. By leveraging historical pricing data, vehicle specifications, and market trends, the solution facilitates informed decision-making for buyers, sellers, dealers, and fleet managers alike. Ultimately, this project aims to transform vehicle pricing from a subjective guesswork process into a transparent, data-driven, and efficient system, reducing uncertainty and enhancing market confidence.

Manual estimation is inefficient, subjective, and prone to human bias. Furthermore, no single platform currently integrates real-time vehicle price prediction, batch analysis, and exploratory analytics in an interactive environment. This project addresses that gap by building an AI-powered Vehicle Price Prediction System powered by XGBoost — an advanced machine learning model — integrated into a modular, user-friendly Streamlit application.

**7.2 Analyzing the Project**

The Vehicle pricing is a multifaceted challenge, shaped by numerous interdependent factors such as brand, model, engine specifications, mileage, fuel type, year of manufacture, body type, trim level, drivetrain, condition, location, and market demand. These factors interact in complex ways, making accurate price estimation a difficult task. Traditional manual valuation methods are impractical for modern vehicle markets because of several critical limitations:

* **Large Dataset Size and Variability:** The automobile market encompasses thousands of models with varying specifications, features, and configurations. The volume and complexity of such data make manual analysis inefficient and prone to error. Additionally, frequent market fluctuations and regional differences add layers of complexity that cannot be handled effectively without automation.
* **Complex Relationships Between Features:** Vehicle price is rarely determined by a single attribute. Instead, it results from complex interactions between multiple features — for example, how mileage affects price differently depending on the age of the car, or how brand reputation interacts with trim level and condition. Manual methods cannot reliably capture these non-linear relationships.
* **Lack of Accessible Tools for Batch and Comparative Analysis:** Current valuation tools often focus on single-vehicle estimates without offering the ability to process large datasets or perform comparative analysis across multiple vehicles. This limits users’ ability to understand market trends and relative pricing.
* **Absence of an Intuitive Interface for Exploration and Prediction:** Existing solutions rarely provide a user-friendly interface that enables interactive exploration of vehicle attributes, instant predictions, and visualization of comparative pricing trends. This gap limits the practical usefulness of vehicle price prediction tools for diverse stakeholders.

Therefore, there is a strong need for a data-driven vehicle price prediction system that addresses these challenges. Such a system should integrate:

* Accurate machine learning models capable of understanding complex patterns and delivering precise price predictions.
* Batch processing capabilities to evaluate multiple vehicles at scale.
* Comparative analysis tools to allow side-by-side evaluation of different models or configurations.
* An interactive, intuitive interface for easy exploration of features, predictions, and market insights.

This approach would transform vehicle valuation from a slow, subjective process into a fast, transparent, and reliable system, empowering buyers, sellers, dealers, and analysts to make informed decisions with confidence.

**7.3 Dataset Overview**

The dataset used in this project is a comprehensive collection of 979 vehicle entries, encompassing diverse models from multiple manufacturers. Each entry represents a unique vehicle listing, containing 17 attributes that describe its specifications, features, and pricing details. This dataset serves as the foundation for training and evaluating the vehicle price prediction model, enabling the system to learn complex relationships between vehicle attributes and their market value.

The dataset includes a mix of categorical, numerical, and textual attributes, providing a rich basis for feature engineering and exploratory data analysis. The attributes are as follows:

**Table 1: Column's Descriptions**

|  |  |
| --- | --- |
| **Column Name** | **Description** |
| name | The full name of the vehicle, including make, model, and trim |
| car descriptions | A brief description of the vehicle, often including key features and selling points. |
| make | The manufacturer of the vehicle (e.g., Ford, Toyota, BMW). |
| model | The model name of the vehicle. |
| year | The year the vehicle was manufactured. |
| price | The price of the vehicle in USD. |
| engine | Details about the engine, including type and specifications |
| cylinders | The number of cylinders in the vehicle's engine. |
| fuel | The type of fuel used by the vehicle (e.g., Gasoline, Diesel, Electric). |
| mileage | The mileage of the vehicle, typically in miles. |
| transmission | The type of transmission (e.g., Automatic, Manual). |
| trim | The trim level of the vehicle, indicating different feature sets or packages |
| body | The body style of the vehicle (e.g., SUV, Sedan, Pickup Truck). |
| doors | The number of doors on the vehicle. |
| Exterior\_color | The exterior color of the vehicle. |
| Interior\_color | The interior color of the vehicle. |
| drivetrain | The drivetrain of the vehicle (e.g., All-wheel Drive, Front-wheel Drive). |

With nearly 1,000 records spanning multiple brands and configurations, the dataset is sufficiently diverse to capture different market segments but also limited enough to require careful handling for robust modeling, The combination of textual descriptions, categorical features, and numerical values requires tailored preprocessing steps, including encoding, normalization, and text feature extraction. Some entries contain missing values or inconsistent formatting, requiring cleaning and imputation before model training, Many features are interrelated, such as year and mileage or make and price, which must be addressed during feature engineering to prevent multicollinearity and improve model accuracy.

**7.4 Objectives of the Project**

**7.4.1 Overview of the System**

The primary goal of this project is to design, develop, and deploy a robust, accurate, and user-friendly vehicle price prediction system that addresses the challenges outlined earlier. The system will be designed with scalability, usability, and data-driven decision-making in mind.

The specific objectives of this project are as follows:

1. **Accurate Price Prediction**  
   Train and deploy an XGBoost Regressor model that predicts vehicle prices with high accuracy (RMSE: ~6,970; R²: 0.841).
2. **Interactive Prediction Interface**  
   Create a user-friendly Streamlit-based web application that allows:

* Basic predictions based on filtering brand, model, and price range.
* Detailed predictions via full specification inputs.

1. **Batch Prediction Module**  
   Allow users to upload vehicle datasets for batch price prediction, with visual analytics and downloadable results.
2. **Insights & Analytics Dashboard**  
   Include dataset visualization, feature importance analysis, price trend exploration, and engineered features for richer insights.
3. **Extendable Architecture**  
   Design a modular backend to enable future expansions, such as:

* Real-time API integration.
* Downloading Options.
* Visualizations of the inputs

**7.4.2 Relevance**

This tool serves a wide spectrum of users: from end consumers checking resale value, to small retailers looking to automate pricing, to data science learners exploring ML deployment workflows. By fusing usability with intelligence, the application demonstrates how real-world regression models can be embedded in responsive interfaces.

Additionally, the inclusion of many features such as single, batch and modular design elevates this project beyond a basic model demo — showcasing how data science products can be fun, user-friendly, and practical for diverse users.

**Data Collection and Preprocessing**

**8.1 Data Source and Formatting**

The dataset used for this project was sourced from the Unified Mentor Vehicle Dataset Repository, a curated dataset that contains comprehensive information on nearly 1,000 vehicles from various manufacturers. The dataset includes a wide range of vehicle specifications, features, and associated market prices, making it ideal for developing a robust vehicle price prediction system.

The dataset is provided in CSV (Comma-Separated Values) format, a standard structure for tabular data that facilitates easy manipulation and integration with data analysis and machine learning workflows. The CSV format offers compatibility with a variety of data science tools and libraries, ensuring efficient processing and accessibility.

.

For this project, Python was the primary tool for data manipulation and preprocessing. Specifically:

* Pandas: Used for data ingestion, exploration, transformation, and cleaning.
* NumPy: Utilized for numerical operations and array-based computations.
* Scikit-learn and other preprocessing libraries: Applied for feature encoding, scaling, and transformation prior to model training.

**Sample of Dataset Columns:**

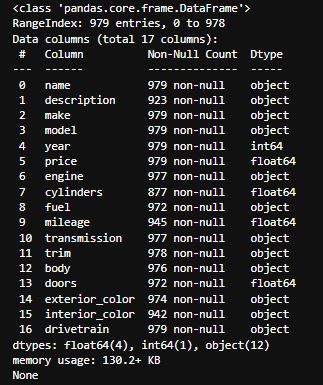
**Table 2: Example of the Dataset**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| name | make | model | engine | Cyli. | fuel | mileage | trans | trim | Body/door |
| Wagoneer II | Jeep | Wagoneer | 24V GDI | 6 | Gas | 10 | 8-Speed | S- II | SUV/4 |
| RAM 3500 | RAM | 3500 | 24V DDI | 6 | Diesel | 10 | 6-Speed | Larmaie | Pickup/4 |

The dataset contains a diverse set of attributes shown in fig[1], including:

* **Numerical Features**: year, price, mileage, cylinders, doors.
* **Categorical Features**: make, model, fuel type, transmission, trim, body type, drivetrain.
* **Textual Features**: name, description, engine specifications, which require specialized handling such as natural language processing or feature extraction.

This diversity requires tailored preprocessing strategies to ensure the dataset is clean, consistent, and suitable for machine learning modeling.



*Figure 1: Data Types in the Datasets*

This rigorous data collection and preprocessing pipeline ensures that the vehicle dataset is clean, structured, and enriched for modeling. By preparing the data in this manner, the project can deliver a high-performing vehicle price prediction system that leverages accurate feature representations and mitigates noise, inconsistencies, and bias inherent in raw datasets.

**8.2 Features Selected**

Although the original dataset contained 17 features, not all of them equally contribute to predicting vehicle prices. Feature selection was performed to retain only the most relevant attributes, ensuring model efficiency, interpretability, and ease of use for end-users. This process reduces noise, improves prediction accuracy, and enhances the overall usability of the system.

Feature importance was evaluated using model-based techniques, specifically leveraging the XGBoost Regressor’s built-in feature importance metric. The following features were selected based on their predictive power and practical relevance to users:

**Table 3: Specified Selected Features helping in Prediction**

|  |  |
| --- | --- |
| **Features** | **Importance in Predictions (/1)** |
| fuel\_Diesel | ~0.14 |
| fuel\_Electric | ~0.092 |
| fuel\_Gasoline | ~0.09 |
| Model\_EQ5 450 | ~0.051 |
| Engine\_24V GDI DDHC Turbo | ~0.05 |
| Engine\_24V GDI DDHC Twin Turbo | ~0.045 |
| make\_BMW | ~0.04 |
| Body\_Sedan | ~0.022 |
| Make\_Mercedes-Benz | ~0.021 |
| Model\_Compass | ~0.0205 |

The Features favoured us in the training help in the following way:

* Optimized Model Performance: Removing low-importance or redundant features reduces noise and improves computational efficiency.
* Simplified User Input: By limiting the number of required inputs, the system becomes more accessible without sacrificing accuracy.
* Industry Relevance: The selected features align with automotive industry valuation practices and customer expectations.
* Explainability: The selected features provide intuitive insights into the price prediction process, enhancing trust and user engagement.

**8.3 Handling of Data Values**

The dataset underwent a rigorous preprocessing pipeline to transform raw inputs into clean, consistent, and machine-learning-ready formats. This process included encoding categorical variables, cleaning inconsistencies, handling missing values, scaling numerical features, and splitting the data for model training and evaluation.

**8.3.1 Encoding**

Encoding is essential for converting categorical variables into numerical formats that machine learning algorithms can process effectively. For this project:

* One-Hot Encoding was applied to categorical features such as make, model, fuel, transmission, trim, body, doors, exterior\_color, interior\_color, and drivetrain.

1. One-Hot Encoding avoids ordinal assumptions and allows the model to learn distinct impacts of each category.
2. Implemented using scikit-learn’s OneHotEncoder to ensure consistency during both training and runtime prediction.

* Label Encoding was applied selectively to maintain interpretability for features where a numeric representation was more meaningful in the user interface.
* Binary Features (e.g., presence or absence of a specific feature) were retained as integer values (0, 1) for simplicity and efficiency. These were converted to human-readable labels when displayed to users.
* Numerical Features such as year, mileage, and price were scaled using StandardScaler.
* Standardization brings features to a consistent range with mean = 0 and standard deviation = 1, improving convergence of the model and reducing bias due to scale differences.

**8.3.2 Data Cleaning**

Data cleaning ensured consistency, reduced noise, and prepared the dataset for reliable model training:

* **Standardization of Feature Names:** All feature names were converted to lowercase with underscores replacing spaces for consistent reference across scripts.
* **Missing Value Treatment:** Null values were minimal but addressed using appropriate imputation strategies:

1. Numerical columns: replaced with mean or median values.
2. Categorical columns: replaced with the mode or a special "unknown" category.

* **Outlier Detection and Treatment:**

1. Visualized using box plots and z-score filtering.
2. Outliers were retained unless they had a significant negative impact on model performance, ensuring the model could learn realistic market variances without overfitting to noise.

* **Consistency Checks:** Ensured all encoded values, feature names, and data formats were aligned across training and prediction environments.

**8.3.3 Trained Dataset Preparation**

To ensure robust model evaluation and prevent overfitting, the cleaned dataset was split into training and testing subsets:

* Training Set (80%): Used for fitting the machine learning model and tuning hyperparameters.
* Testing Set (20%): Used exclusively to evaluate the generalization performance of the model on unseen data.

The split was randomized to ensure an unbiased representation of the dataset across subsets. Stratified sampling was considered for categorical variables to preserve their proportional representation.

.

**8.3.4 Model Readiness**

Upon completion of preprocessing:

* The XGBoost regression model was trained on the cleaned dataset, ensuring high predictive accuracy.
* Preprocessing objects and the trained model were saved as serialized .pkl files.
* At runtime, the Streamlit application dynamically loads these files to:

1. Ensure consistent preprocessing between training and inference.
2. Deliver high-quality predictions in both single and batch prediction modes.
3. Support scalability and integration of future extensions such as API endpoints and additional feature inputs.

This structured approach to data handling guarantees model stability, accuracy, and scalability, ensuring the Vehicle Price Prediction System operates effectively in real-world conditions.

.

**Project Concept and Model Training**

**9.1 Technology and Model Used**

This project leverages modern machine learning and web development tools to build an interactive and reliable vehicle price prediction system. The system allows users to input vehicle specifications and instantly receive an estimated market price using a trained regression model.

The goal is to deliver an intuitive, accurate, and interpretable tool for vehicle buyers, analysts, and enthusiasts.

The system integrates several technologies:

1. **Python:** The primary programming language that comforts me for both data science and web development in this project due to its readability and strong ecosystem.
2. **Jupyter Notebook:** Used for exploratory data analysis (EDA), data preprocessing, model experimentation, and training. Its interactive environment provides a readable interface enabled for quick iteration and visual insights.
3. **XGBoost (Extreme Gradient Boosting):** Selected as the final model for its ability to handle numeric and categorical data efficiently while maintaining high prediction accuracy and robustness against outliers.
4. **Scikit-learn:** Provided utilities for preprocessing, encoding, scaling, model training, evaluation, and serialization.
5. **Joblib:** Utilized to serialize the trained model, allowing the web app to load predictions without retraining.
6. **Streamlit:** The framework Used to build the interactive web interface for real-time price predictions.
7. **Matplotlib & Seaborn:** Used for visualizing the dataset, correlation analysis, feature importance, and model performance.
8. **HTML/CSS:** Used to embed the styles of the UI and enhance the UX directly into the interface for an engaging user experience.

Together, these tools form a robust and modular pipeline—from raw dataset ingestion to polished prediction delivery—allowing seamless integration from raw data to user-facing price prediction.

**9.2 Model Training and Evaluation**

Given the context of the problem, To ensure reliability, accuracy and generalization ability of the model were assessed using various performance metrics, to see how well the predicted prices aligns with our desired result.

**9.2.1 Model Training Process**

The processed dataset from Chapter 8 was split into 80% training and 20% testing subsets to ensure robust model evaluation and prevent overfitting. The model training pipeline involved several preprocessing steps and experimentation with multiple algorithms to determine the most suitable model for vehicle price prediction.

The key steps were as follows:

* **Data Preprocessing:** Normalization of numeric features using StandardScaler to ensure all features have zero mean and unit variance, improving convergence during model training. Encoding of categorical features via OneHotEncoder to transform categorical variables into numerical binary vectors, enabling algorithms to process them effectively.
* **Model Training:** Four different machine learning algorithms were trained:

1. Random Forest Regressor — an ensemble method leveraging bagging and decision trees for improved prediction stability.
2. Decision Tree — a depth searching framework optimized for speed and efficiency, particularly effective for many variables datasets.
3. XGBoost — an advanced gradient boosting model known for high accuracy and robust handling of complex datasets.
4. Linear Regression: The Most basics of the Machine Learning plotting process used for simple and linear datasets.

* **Model Evaluation:**

1. Performance was compared using Root Mean Square Error (RMSE) and R² scores.
2. RMSE was chosen to evaluate prediction error magnitude, while R² measured the proportion of variance explained by the model.

* **Model Selection:** XGBoost outperformed the other models with the lowest RMSE and the highest R² score, making it the optimal choice for predicting vehicle prices.

**Table 4: Model Performance Comparison**

|  |  |  |
| --- | --- | --- |
| **Model** | **RMSE** | **R2 Score** |
| LinearRegression | 7445.5382 | 0.8184 |
| RandomForest | 8152.2691 | 0.7823 |
| XGBoost | 6969.4871 | 0.8409 |
| DecisionTree | 9025.0506 | 0.7332 |

Table 4 shows that XGBoost delivered the best performance, with a lowest RMSE of 6969.49 and a highest R² score of 0.841, confirming its suitability for vehicle price prediction.

**9.2.2 Statistical Overview**

A statistical description of the training dataset is presented, providing insights into feature distributions and central tendencies. This analysis helps verify data quality and understand feature importance in model training.

**Table 5: Dataset and Columns Statistics**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Columns** | **Count** | **Unique** | **Frequency of Top** | **Mean** | **Std** | **min** | **50%** | **max** |
| Name | 979 | 354 | 33 | - | - | - | - | - |
| Make | 979 | 28 | 192 | - | - | - | - | - |
| Model | 979 | 151 | 61 | - | - | - | - | - |
| Engine | 977 | 100 | 120 | - | - | - | - | - |
| Cylinder | 877 | - | - | 4.97 | 1.39 | 0.00 | 4.00 | 8.00 |
| Fuel | 972 | 7 | 647 | - | - | - | - | - |
| Mileage | 945 | - | - | 67.34 | 510.83 | 4.00 | 8.00 | 9711.00 |
| Transm. | 977 | 38 | 312 | - | - | - | - | - |
| Trim | 978 | 197 | 68 | - | - | - | - | - |
| Body | 976 | 8 | 690 | - | - | - | - | - |
| Doors | 972 | - | - | 3.94 | 0.27 | 2.00 | 4.00 | 5.00 |

**9.2.3 Model Residual Analysis**

To better understand model performance, a residual analysis was conducted for the XGBoost model. Residuals (differences between predicted and actual values) were plotted to assess prediction accuracy.

Key findings:

* Residual distribution was approximately centered around zero, indicating unbiased predictions.
* A slight increase in residual variance was observed for higher-priced vehicles, suggesting the model could be further refined for extreme values.
* No significant patterns were detected in residuals versus predicted values, confirming model stability.

To further ensure model generalization:

* Cross-validation with k-fold (k=5) was performed, confirming consistent performance across folds.
* Future improvements could include hyperparameter tuning, feature engineering, and integration of external datasets such as market trends or geographic location.

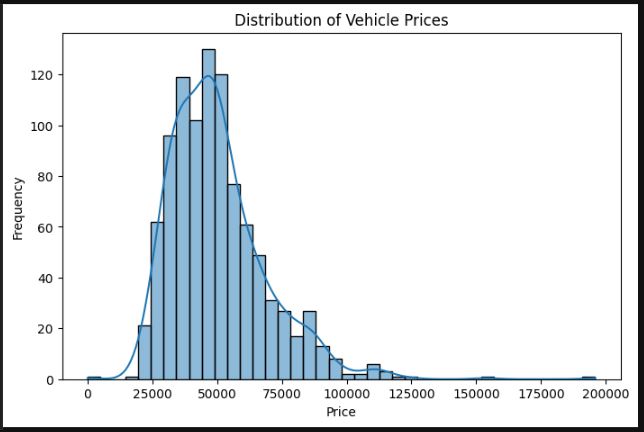
**9.3 Visual Insights and Interpretability**

Interpretability of machine learning models is essential to ensure predictions are not only accurate but also explainable. This section presents detailed visual analyses of vehicle price distribution, feature relationships, and model interpretability results, providing both qualitative and quantitative insights.

**9.3.1 Distruibution of Vehicle Price**

The price distribution exhibits a bell-shaped curve, suggesting that most vehicles in the dataset cluster within a certain price range, while fewer vehicles exist at extreme low and high prices.

* Observation: Majority of vehicles are priced between 25,000 and 100,000, with the upper limit and lower limit touching 200,000 and 5,000.
* Insight: The bell curve indicates a balanced dataset with moderate variance, though extreme price points may require additional attention during model training.

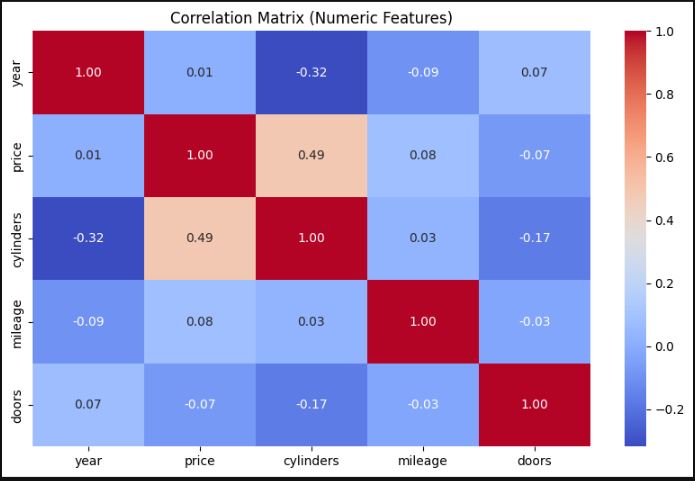
· 

*Figure 2: Distribution of the Prices*

The Graph Shows the Distribution favours the mid ranged vehicles of more than 25,000 and at the amount of 75,000 where afterwards the dip is quite big, This shows that the Dataset will be more accurate in these ranges and outside them it may have a bit of issue in the accuracy factor of prediction of the vehicle price.

**9.3.2 Correlation Matrix**

The correlation heatmap offers insight into feature relationships and highlights key drivers for price prediction.



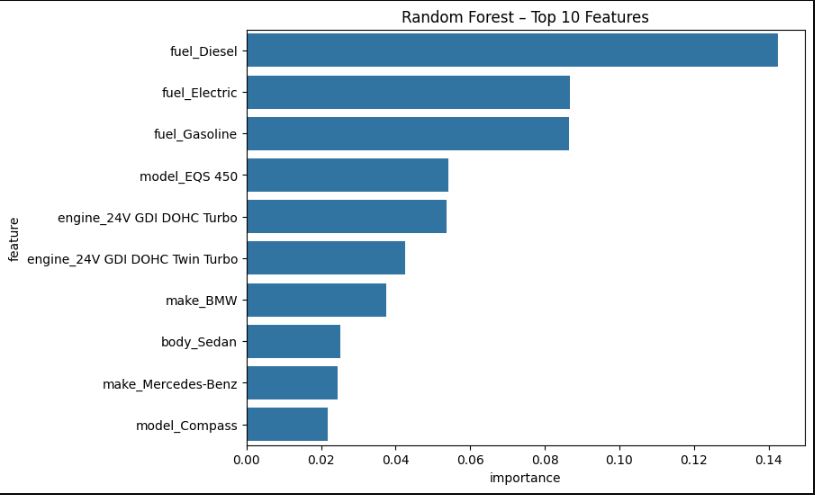
*Figure 3: Correlation Matrix of different features*

The Correlation Matrix points out a lot of the things for us to notice, The Selection of the Features were determined by how important they are in the final model prediction and during the training process.

1. The Most Noticable Correlation is the Cylinders and Price, this is to say that most of the changes in Cylinders will impact the price in a positive correlation, if the no, of cylinders increases then the predicted price will also be increased for that vehicle,
2. Cylinder and Year: They are negatively correlated to each other at ~-0.32 of the times the change in one makes the other variable negativily impacted on them.
3. Cylinder and Doors: though not as significantly but they are impacting each other ~-0.17 of the times one changes the other will get the impact in the opposite way.

**9.3.3 Importance among the Important Features**

This visualization reveals the importance, The most important factor were not just the given columns, it was the combination of various factors and such that were present in the dataset itself, it helped us to figure out the pattern a vehicle may tend to if the certain parameter is selected on a Featured Column.

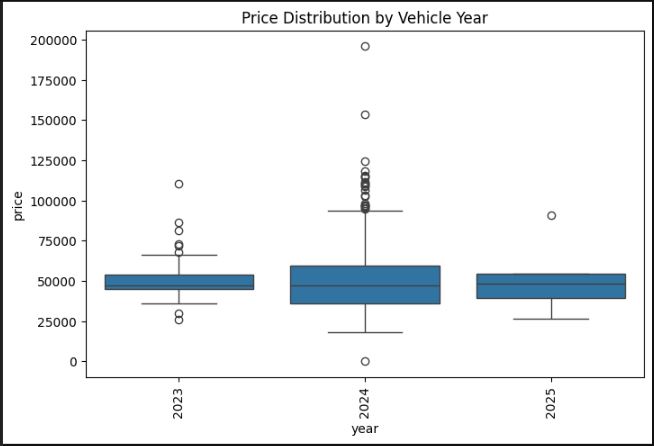
******

*Figure 4: Top 10 Importance Features helping in Prediction*

**9.3.4 Price Distribution by Various Factors**

This visualization highlights how vehicle prices vary across different model with different parameters, offering insights into depreciation and consumer valuation of different kinds of models.

1. **Price Distribution by Year:**



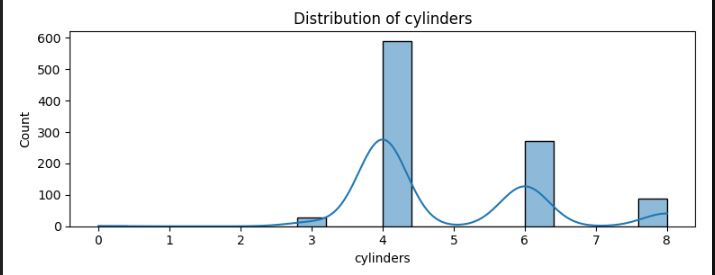
*Figure 5: Price Distribution by Year*

Observation:

* Prices demonstrate a steady upward trend with newer model years.
* Vehicles manufactured between 2023–2025 show a distinct upward shift, indicating that the newest models command a premium relative to older ones.
* Earlier years display wider price variability, reflecting differences in condition, mileage, and trim levels, while more recent years show tighter clustering, suggesting more uniform pricing in newer inventory.
* The Dataset provides a lot more cars for the Year 2024 but the average cost of the same has been increased in the next years or so.

1. **Price Distribution by Cylinders:**

This visualization examines how vehicle prices vary depending on engine cylinder count.

******

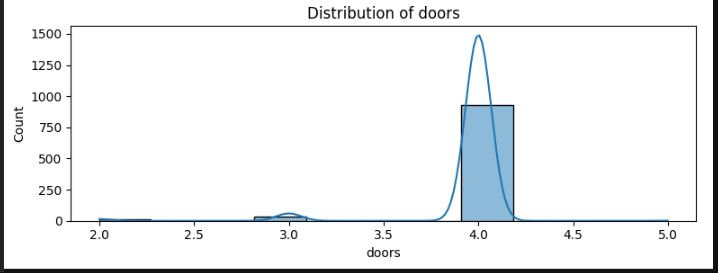
*Figure 6: Price Distribution by Cylinders*

Observation:

* Vehicles with 4 cylinders dominate the dataset, showing moderate price ranges.
* 6-cylinder and 8-cylinder vehicles are priced higher on average, reflecting their larger engine capacity and performance orientation.
* Outliers are present in all groups, particularly for high-performance 8-cylinder models, where luxury and sports vehicles command premium prices.
* The clear tiered price progression from 4 → 6 → 8 cylinders highlights consumer willingness to pay more for higher performance.

1. **Price Distribution by Doors:**

This visualization highlights pricing patterns across different door configurations.



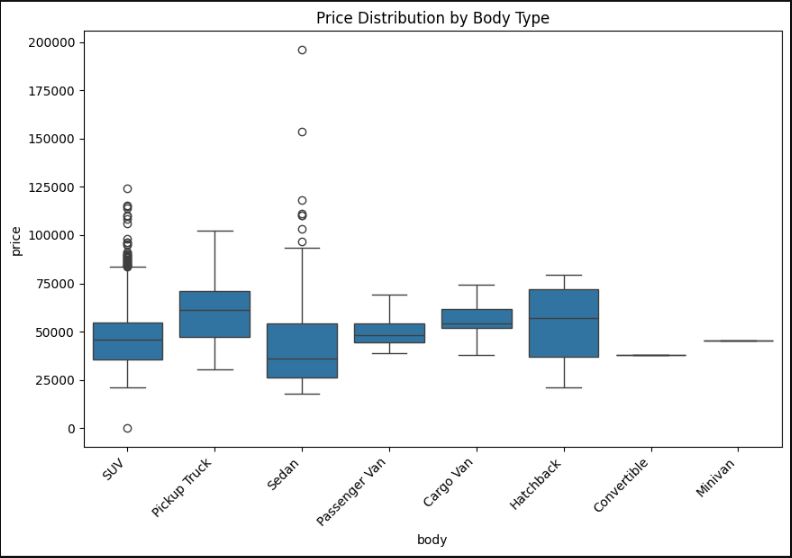
*Figure 7: Price Distribution By Doors*

Observation:

* The vast majority of vehicles in the dataset have 4 doors, covering sedans, SUVs, and pickups.
* 2-door vehicles are less common, but their prices vary widely, likely reflecting coupes and specialty models.
* 3-door and 5-door variants appear infrequently, suggesting niche categories (e.g., hatchbacks, compact cars).
* Door count alone has limited predictive power for price, as it mostly reflects body style rather than inherent value.

1. **Price Distribution by Body Types:**

This visualization reveals how vehicle body style affects pricing.

******

*Figure 8: Price Distribution by Body Type*

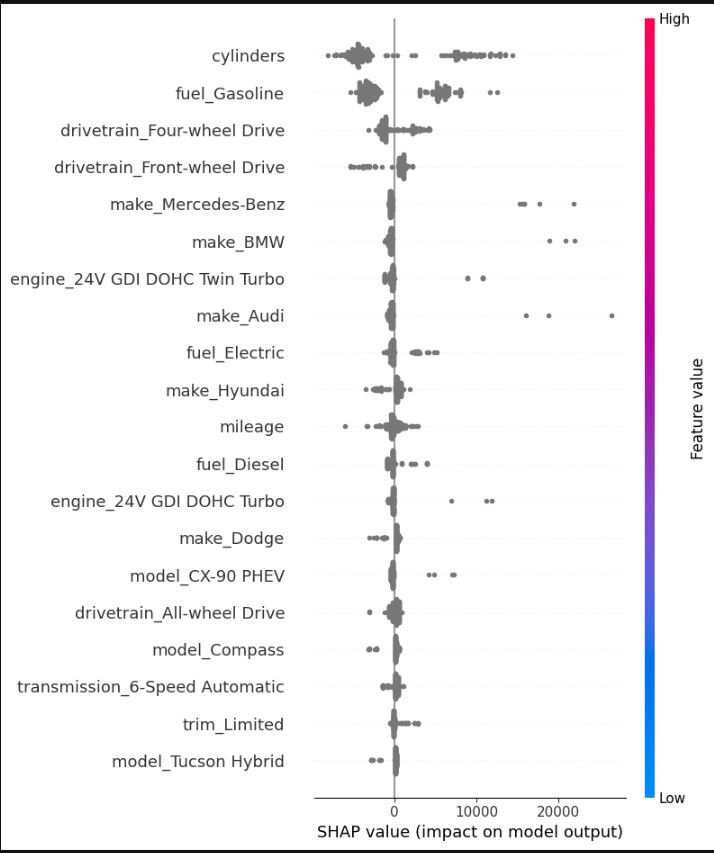
Observation:

* SUVs and Pickup Trucks show the widest spread of prices, with many high-value outliers, reflecting their popularity across both mainstream and luxury segments.
* Sedans exhibit broad price variability, from entry-level compact cars to high-end luxury sedans.
* Passenger Vans, Cargo Vans, and Hatchbacks fall into mid-price ranges with tighter distributions.
* Convertibles and Minivans are less represented, with relatively consistent but narrower price bands.
* Body type is a critical determinant of vehicle price, strongly tied to consumer preferences and use cases.

**9.4 Comparisons and Analysis**

**9.4.1 SHAP Analysis**

This visualization utilizes SHAP (SHapley Additive exPlanations) values to illustrate the impact of various features on the model’s predicted vehicle prices.



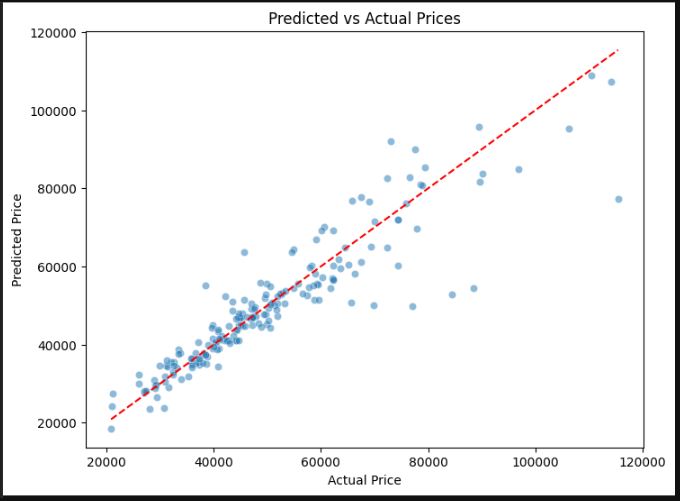
*Figure 9: SHAP Values on model*

* The number of cylinders stands out as one of the most influential features, displaying considerable SHAP value spread, which indicates its strong effect on price predictions.
* Fuel type, especially gasoline, and drivetrain configuration (four-wheel drive vs. front-wheel drive) contribute significantly, reflecting consumer preferences and mechanical influences on value.
* Brand attributes (e.g., Mercedes-Benz, BMW, Audi) show distinct positive SHAP impacts, confirming premium manufacturers’ strong price uplifts.
* Some features, such as engine configurations and limited trim or model types, show noteworthy influence for particular entries, underlining the importance of advanced specifications for higher-value vehicles.
* Electric fuel and hybrid models, although less common, register meaningful positive effects, likely due to technological advancements and ongoing market shifts toward alternative energy vehicles.
* Mileage still presents an expected but more nuanced relationship, providing moderate downward pressure on price depending on the context of other features.

**9.4.2 Predicted vs Actual Price**

This plot demonstrates the relationship between the predicted vehicle prices from the model and their corresponding actual sale prices.

* The data points closely follow the red dashed line (ideal fit), indicating strong overall agreement between the model's predictions and real market values.
* While there is a generally tight clustering along the line for the majority of price ranges, some scatter emerges at the high end, where prediction error increases, possibly due to outliers or lower sample sizes for expensive vehicles.
* The alignment signifies that the model effectively captures key determinants of price, providing high predictive accuracy in most scenarios.
* Marginal deviations at extreme price points highlight the potential for further refinement in modeling or additional feature engineering to address these complex cases.

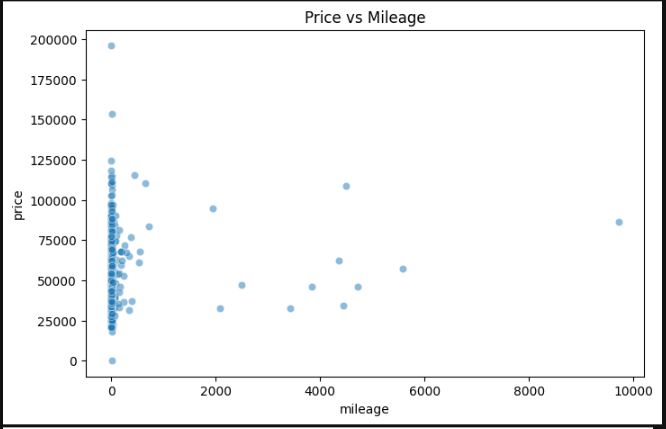


*Figure 10: Actual vs Predicted Price*

**9.4.3 Price vs Mileage**

The scatter plot provides insight into how vehicle price relates to mileage, one of the most classic predictors of used car value.

* There is a discernible cluster of vehicles with low mileage and a concentration of moderate prices, reflecting the typical inventory distribution.
* Vehicles with higher mileage tend to show greater price dispersion and, overall, a gentle downward trend in price, as expected based on depreciation patterns.
* Nevertheless, notable high-priced outliers exist even at substantial mileages, likely attributable to luxury or specialty vehicles that retain significant value despite usage.
* The spread also suggests that while mileage is important, other attributes (such as make, model, and engine type) can substantially modulate price outcomes, consistent with the findings from the SHAP analysis.



*Figure 11: Price vs Mileage*

All the figures shows that the model is well made for the front end integration and the prediction value will be as close as it can get, The Main features, along with the pattern recognitions of certain values for the certain selections really helps a lot in finalizing the predicted price of the sets of the inputs.

**System Architecture and Implementation**

**10.1 System Architecture**

The Vehicle Price Prediction System is designed to integrate a machine learning regression model with a modern, interactive Streamlit web application. The architecture emphasizes modularity, scalability, and maintainability, ensuring that each component—data preprocessing, model training, prediction logic, visualization, and user interaction—remains independent but interoperable.

The system is divided into two major phases:

* Model Training & Evaluation (executed in Jupyter Notebook or Python scripts)
* Interactive UI Deployment & Visualization (via Streamlit web application)

This layered approach allows for seamless updates: models can be retrained and optimized independently, while the web interface dynamically incorporates the latest version without requiring a full system overhaul.

**10.1.1 Model Training Pipeline**

The training pipeline defines how raw data is transformed into a production-ready machine learning model. The workflow is iterative and adaptable, enabling continuous improvements as new data becomes available.

1. **Data Loading and Exploration:** The dataset includes attributes such as make, model, year, engine, cylinders, fuel, mileage, transmission, trim, body, doors, exterior\_color, interior\_color, drivetrain, and price. Initial data inspection includes detecting missing values, identifying categorical and numerical attributes, and visualizing price trends across features.
2. **Feature Engineering**:

* **Data Cleaning:** Handling null values, inconsistent naming conventions, and outliers.
* **Categorical Encoding:** Label encoding for ordinal attributes (e.g., number of doors), one-hot encoding for nominal attributes (e.g., fuel type, transmission).
* **Numerical Normalization:** year, mileage, and engine displacement to improve model stability.
* **Exploratory Visualizations:** Feature importance plots, correlation heatmaps, and distribution graphs to guide preprocessing decisions.

1. **Model Selection**: Multiple regression-based algorithms were tested, including Linear Regression, Gradient Boosting, XGBoost, and Random Forest. The XGBoost was selected for its ability to capture nonlinear relationships, robustness with mixed data types, and resistance to overfitting.
2. **Model Evaluation**: Evaluation metrics include various factors such as R2 Score, MAR, RMSE, The R2 helped in picking the model for more stable results, cross-validation was used to validate model stability across different subsets of the dataset.
3. **Persistence**: The trained model and preprocessing pipeline were serialized ensuring for seamless integration and deployment, where the web application loads the model directy into the Streamlit app.

**10.1.2 Streamlit Web Application**

The second phase focuses on user interaction through a clean and professional Streamlit interface. The app is modular, with separate pages for different functionalities.

The application is logically separated into components:

1. main.py: Main application file
2. app\_page/home.py: Introduction Page
3. app\_page/single.py: Consists of Single Types of Prediction and Searching
4. app\_page/batch.py: The User can do multiple price prediction at ones
5. app\_page/extended.py: Contains Featured Engineering and Insights of the Data set and Model.

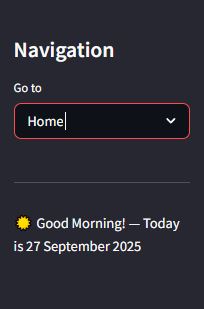
And more…

**10.1.3 UI Logic And Flow**

The user interface (UI) of the Vehicle Price Prediction System is designed to be intuitive, responsive, and user-centric, ensuring smooth navigation for both technical and non-technical users. The design philosophy emphasizes clarity, accessibility, and interactivity, enabling users to not only generate predictions but also explore insights and validate results with confidence.

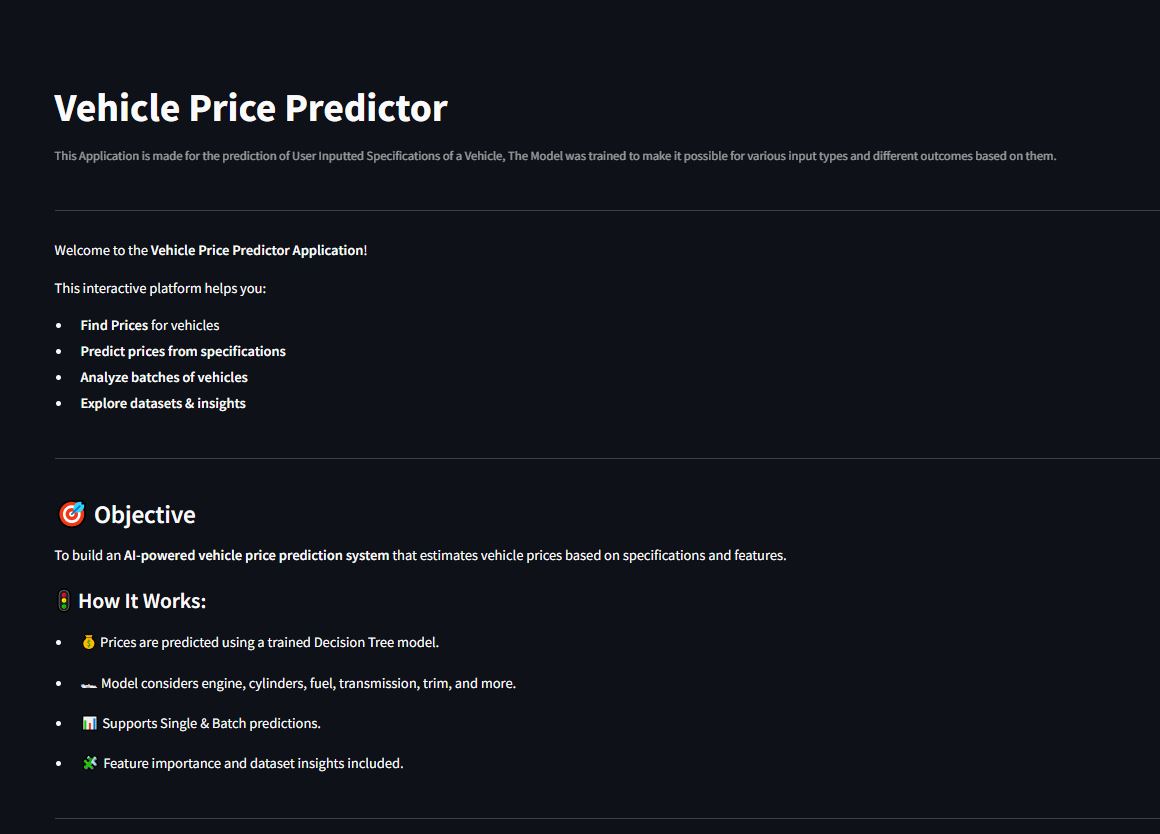
The UI is organized into multiple functional pages, all accessible from a global sidebar navigation panel. Each page is structured to support a specific user task, while maintaining a consistent design language across the application.

1. **Sidebar Navigation:** It Serves as the Primary Navigation hub for the application, The Sidebar Provides global access to Home, Prediction, Batch Prediction, Insights and Analysis sections, with a time and date of the day, It is integrated using the Streamlit components allowing users to switch seamlessly, It is designed to be as fixed and persistent, ensuring users always have navigation options available regardless of the current page.



*Figure 12: Sidebar of the Streamlit Application*

1. **Home page:** The Home Page acts as the entry point and introduction to the application. It provides a comprehensive overview so new users can quickly understand its purpose and functionality.



*Figure 13: Overview of the Application*

**How the App Works:** Explains step-by-step usage, from inputting vehicle details to generating predictions.

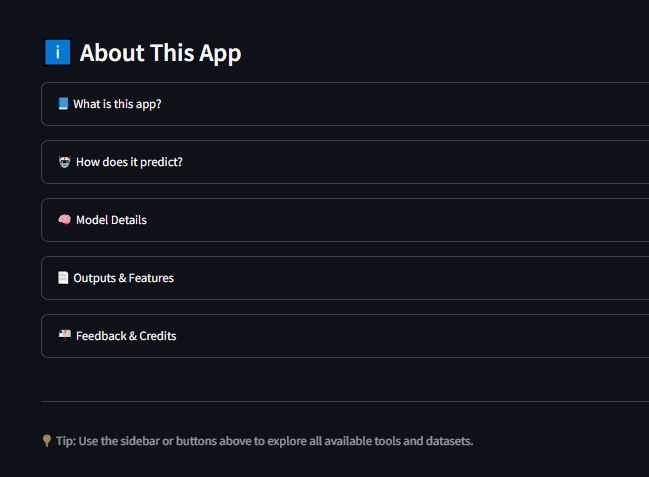
**How Predictions are Generated:** Outlines the role of the regression model, preprocessing, and the importance of data features.

**Model Details:** Provides metadata on the chosen algorithm, dataset size, last training date, and performance metrics.

**Output and Features:** Explains how results are displayed (single predictions, batch outputs, comparisons).

**Feedback Section:** Allows users to provide feedback or suggestions, enhancing system improvement.

Credits Section: Lists dataset sources, tools, and contributors.



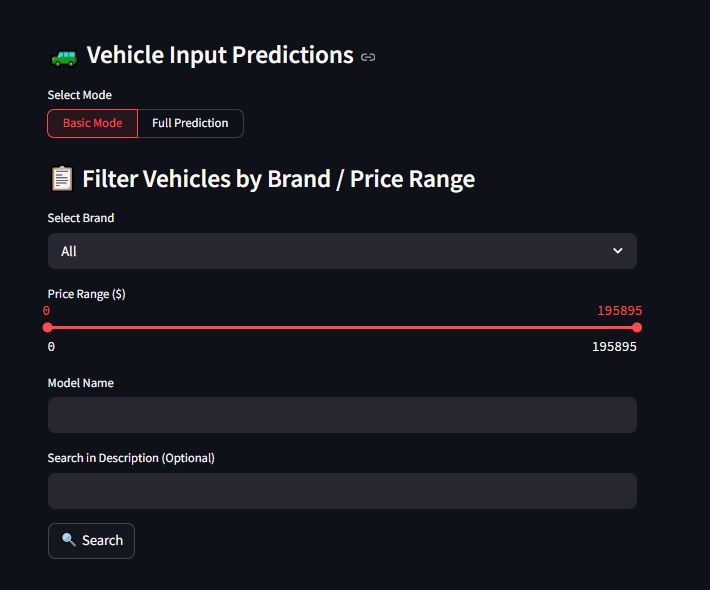
*Figure 14: About Section*

This page is designed with text blocks, collapsible sections, and callout boxes for readability.

1. **Prediction Page:** The Prediction page is central to the application, enabling interactive price estimation for vehicles. It is divided into two distinct modes:

**Basic Mode:**

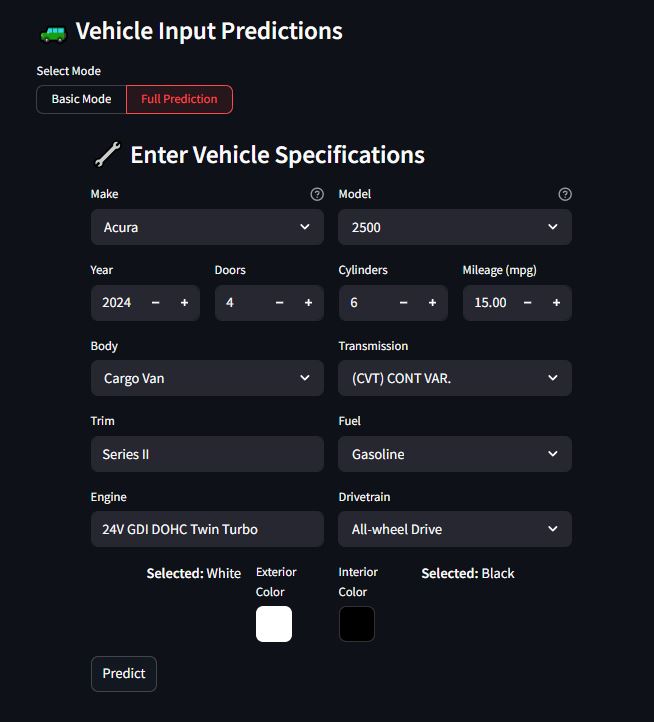
* Allows users to quickly estimate a car’s price by selecting only the make and model.
* Provides a simplified interface, suitable for casual users.
* Results are shown instantly, with predicted price displayed in a highlighted result box.



*Figure 15: Basic Mode*

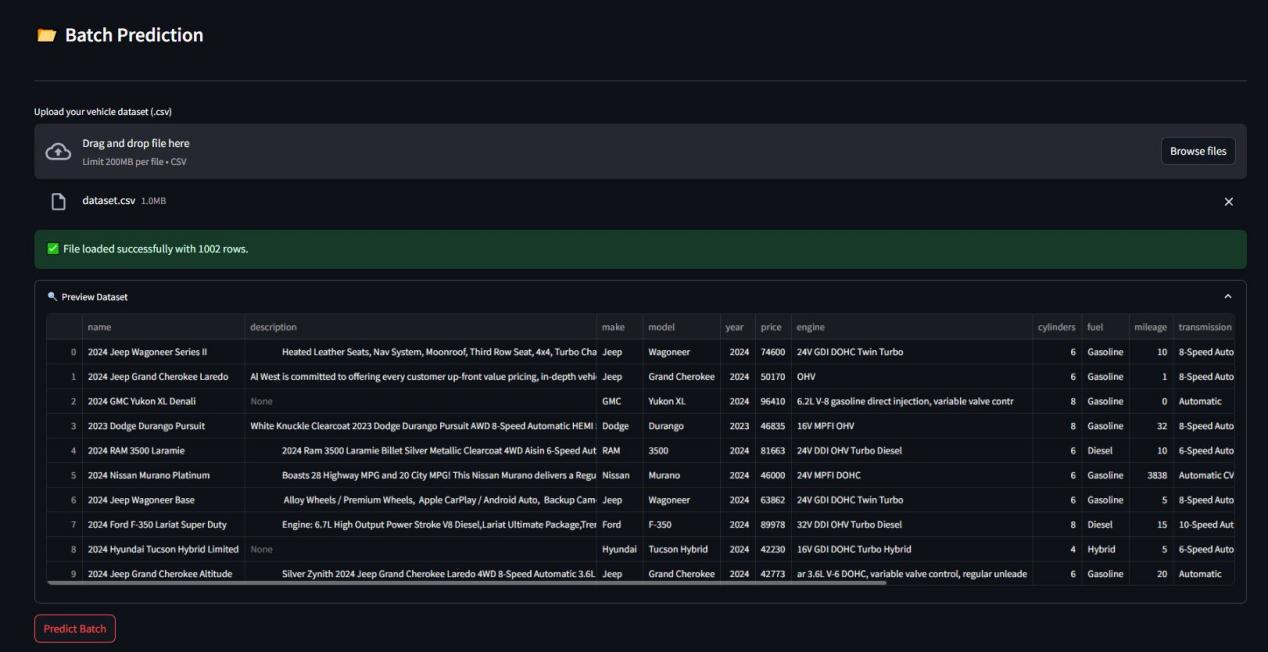
**Advanced Mode:**

* Designed for detailed, attribute-driven predictions.
* Users input multiple fields: year, mileage, fuel type, transmission, engine, trim, drivetrain, body type, etc.
* The prediction logic applies preprocessing (e.g., encoding, scaling) in the background before model inference.
* Results are presented with confidence ranges, possible error margins, and optional visualization (price positioning compared to dataset distribution).



*Figure 16: Full Prediction Mode*

1. **Batch Prediction Page:** The Batch Prediction module supports bulk price estimation for large datasets, making it suitable for dealerships, researchers, and analysts.

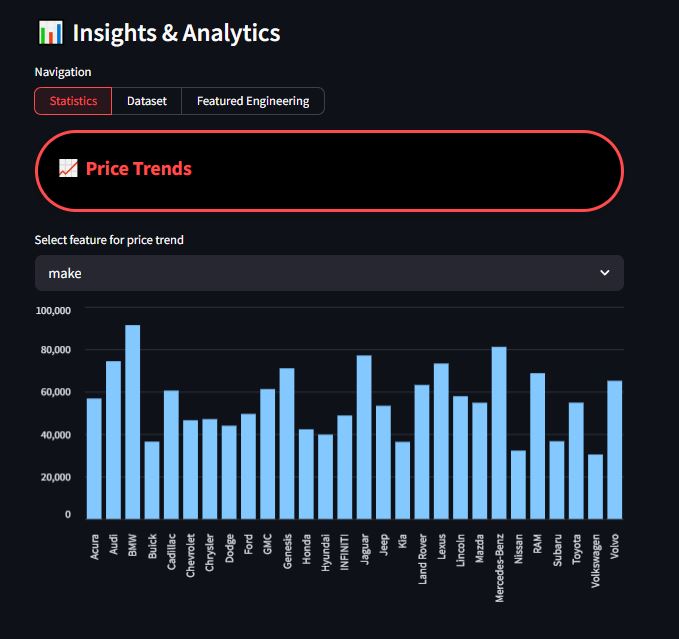


*Figure 17: Batch Prediction Mode*

The Batch Works when the User uploads a structured CSV files containing vehicle features, then after uploading it displays the user the preview of the uploaded .csv file, ones Predict button is pressed it adds the additional prediction\_price column to the table, along with various Graphs and Charts to visualized the prediction process and how the model acted to it, the user can also download the updated .csv file with prices attached to them.

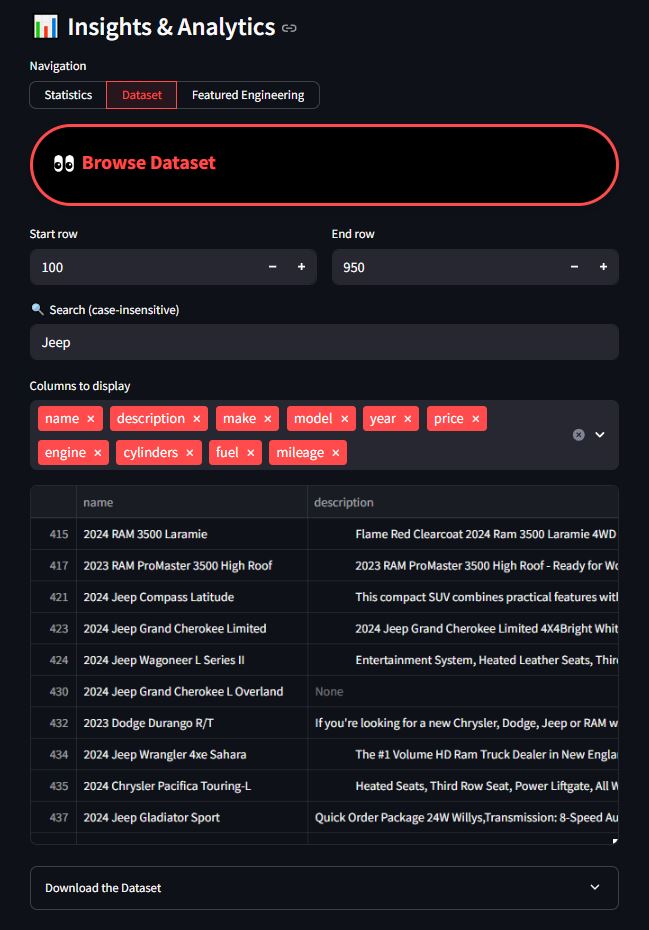
1. **Insights and Analysis Page:** The Insights and Analysis Page empowers users to explore both the dataset and model performance. It is organized into three interactive tabs:

* **Statistics of the Model:** Users can filter by make, year, body type, fuel, drivetrain, etc, Along with Generating interactive plots with feature selection on the x-axis and price on the y-axis, helping users analyze price variations and Provides a summary statistics table showing mean, median, standard deviation, and range for selected filters.



*Figure 18: Statistics of the Model*

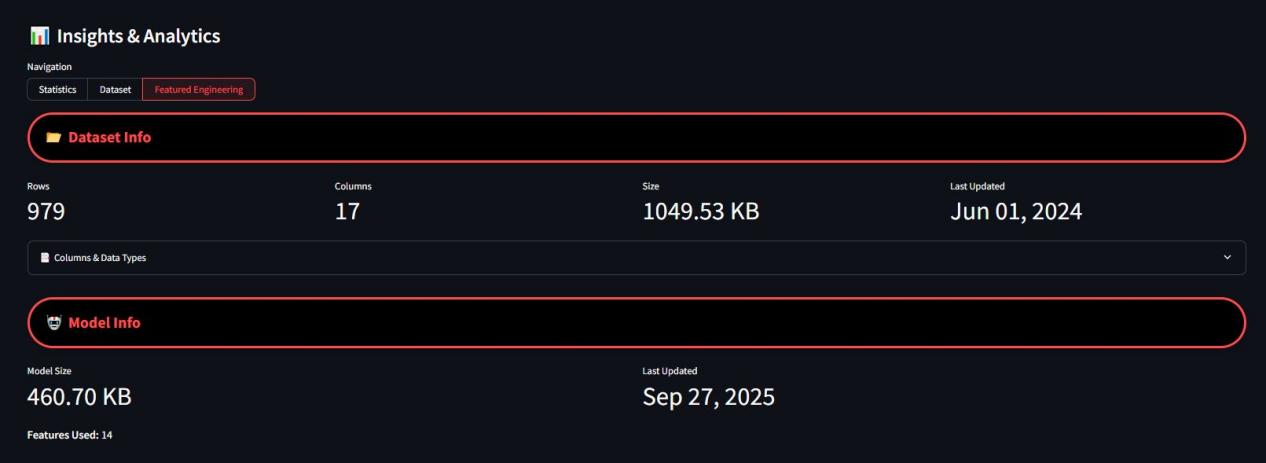
* **Dataset Explorer:** Users can browse the full dataset interactively. The Certain features that the user can interactive with are Row selection, Pagination, The Searching Option by Any type of column is also available, it focuses on the Column-based filters for exploration for the insights.



*Figure 19: Dataset Explorer Overview*

Also at the bottom, the user can download the data-set encapsulated in the st.expander function for further analysis.

* **Feature Engineering Dashboard:** It offers transparency into the data transformation and enrichment steps, The Dashboard shows information regarding the Dataset Info, Model Info, Engineered Featured which are derived attributes and more

****

*Figure 20: Featured Engineering*

It also has visualizations to help the user understands the dataset, training pipeline process and how integrated the data provided works with the frontend.



*Figure 21: Featured Engineering Charts*

**10.2 Streamlit Working and Processing**

The Streamlit framework powers the Vehicle Price Prediction System’s web interface, enabling rapid deployment of machine learning models into a fully interactive application. Streamlit was chosen due to its simplicity, seamless integration with Python, and support for real-time visualizations and interactivity without the need for complex frontend development.

The working of the app can be broken into two parts:

* Tab Structure and Navigation – how the user interacts with the system.
* Key Functional Features – the unique utilities that make the system interactive, professional, and user-friendly.

**10.2.1 How it Works**

The app follows a tab-based architecture, each tab serving a dedicated function. This modular design ensures users can navigate between prediction, batch processing, insights, and system documentation without confusion.

* **Home Tab**

1. Provides an overview of the application and a step-by-step guide on how to use the system.
2. Includes sections on system objectives, methodology, and workflow.

* **Vehicle Price Prediction Tab**

1. The core prediction engine of the application.
2. Offers two modes of operation:
3. Basic Mode: Simplified input where users select only the make and model to get a quick estimate.
4. Advanced Mode: Users input detailed attributes (year, mileage, fuel, drivetrain, transmission, etc.) for a more precise prediction.
5. Results are displayed clearly, with predicted prices highlighted for readability.

* **Batch Prediction Tab**

1. Designed for bulk processing of multiple vehicles simultaneously.
2. Workflow:
3. Upload a structured CSV file.
4. Preview data to verify correctness.
5. Process predictions (system appends a Predicted Price column).
6. View interactive charts summarizing results.
7. Download the processed dataset in CSV format for offline use.

* **Feature Engineering Tab**

1. Offers deeper insights into the dataset and model statistics.
2. Features include:
   * 1. Interactive dataset browsing (rows, search, filters).
     2. Graphical analysis (e.g., price vs year, mileage vs price).
     3. Summarized model statistics (dataset size, last update timestamp, feature distribution).
3. Helps users understand how the dataset influences predictions and validates the model’s reliability.

**10.2.2 Key Features**

The application incorporates a range of enhancements and utility functions that make the user experience smooth, professional, and practical:

* **Consistent Price Formatting**

1. All predictions are displayed in :.2f format, ensuring prices always appear with two decimal places.
2. This formatting enhances professionalism and readability, especially in comparison reports or batch results.

* **Preview Before Prediction**

1. For batch uploads, users can preview the dataset in a clean table format before running predictions.
2. Prevents incorrect file processing and ensures data quality checks are made early.

* **Visualization Support**

1. Built-in interactive graphs provide clarity and analysis.
2. Examples include:
   * 1. Price distributions by make, year, and fuel type.
     2. Batch prediction summaries with histograms and bar charts.
     3. Comparison visuals for multiple vehicles.
3. These visualizations enhance decision-making by transforming raw predictions into interpretable insights.

* **Download Support**
  1. Users can export results in CSV format across different modules (batch predictions, dataset insights, etc.).
  2. Ensures compatibility with external tools like Excel, Google Sheets, and BI dashboards.
* **Professional UI and Design**

1. The application adopts a clean and modern interface, enhanced with CSS-based cards, stylized tabs, and color-coded outputs.
2. Interactive cards highlight key statistics (e.g., dataset size, missing values).
3. The design ensures clarity, readability, and a polished look, suitable for academic, research, and business use cases.

**10.3 Access the Program**

To Run the Program it can be done using a single line bash command:

>> streamlit run main.py

or

Open the file:

Run.bat in the root folder

**Project Concept and Model Training**

**11.1 Sample Prediction and Outcome**

To evaluate the effectiveness of the System in practical scenarios, multiple test cases were executed using random demo inputs. We validate the practical functionality and accuracy of the Application and Comparison System, various test cases were executed using the given vehicles specifications and various inputs. These covered a wide range of feature combinations—varying Brand, model, year, style, trim, and more.

The purpose of these tests was threefold:

* Accuracy: Verify whether predicted prices align with market trends.
* Robustness: Test system performance under incomplete, noisy, or rare input conditions.
* Usability: Assess how easily end-users can navigate the system, interpret results, and make comparisons.

Once these parameters are entered, the system applies preprocessing transformations (encoding, normalization, feature mapping) before forwarding the input data into the trained XGBoost model.

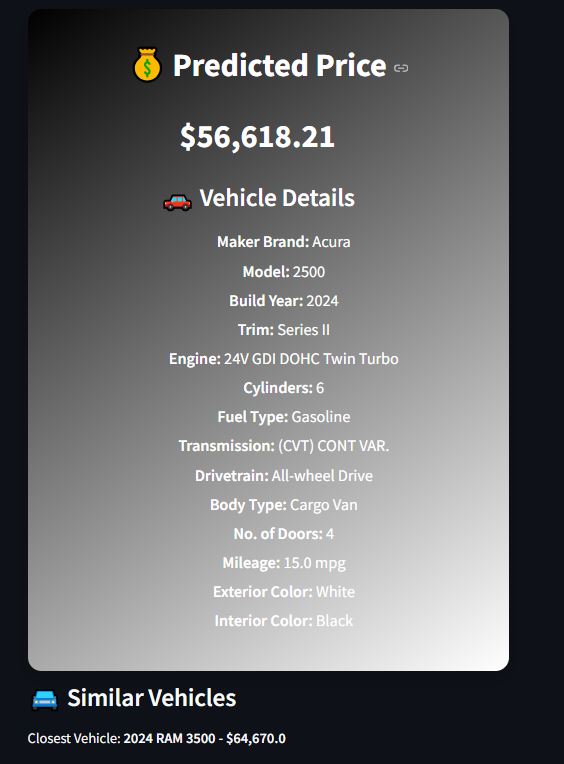
The system’s prediction and comparison capabilities enable users to:

* Explore Market Trends: Observe how vehicle prices change across mileage, model years, or fuel types.
* Make Informed Decisions: Buyers can benchmark multiple options before purchasing; sellers can verify fair listing prices.
* Experiment with Configurations: By altering inputs such as mileage or transmission, users can visualize how changes affect valuation.
* Save and Export Predictions: Session results can be stored and downloaded in CSV format for offline analysis.

Here is a sample output performed using the application in Single Prediction:

*Table 6: Sample Input for a Sample Prediction*

|  |  |
| --- | --- |
| **Columns** | **Input** |
| **Maker Brand:**  **Model Name:**  **Made In Year:**  **No. Of Doors:**  **No. Of Cylinders:**  **Mileage (mpg):**  **Body Type:**  **Transmission:**  **Trim Type:**  **Fuel Category:**  **Engine Type:**  **DriveTrain:**  **Exterior Color:**  **Interior Color:** | Acura  2500  2024  4  6  15.00  Cargo Van  (CVT) CONT VAR.  Series II  Gasoline  24V GDI DOHC Twin Turbo  All-wheel Drive  White  Black |



*Figure 32: Result as the Prediction with Price Range*

**11.2 Visualization and Performance**

Visualization plays a central role in the Vehicle Price Prediction and Comparison System, as it not only improves interpretability of results but also enhances user trust in the model’s predictions. By supplementing raw numeric outputs with interactive charts, plots, and tables, the system ensures that users—whether technical or non-technical—can understand the underlying patterns and validate outcomes effectively.

The system incorporates a suite of visual and analytical tools:

1. **Prediction Confidence Bar Chart:** Displays the predicted price along with confidence intervals.
2. **Comparison Graphs:** Side-by-side bar charts are used for comparing custom vehicle entries with vehicles from the dataset.
3. **Multi Session Comparison Summary:** The system is capable of storing multiple vehicle predictions within a session.
4. **Feature Importance Plot:** A feature importance ranking chart identifies the most influential attributes in price determination.
5. **Distribution Graphs:** The system includes distribution plots that display how vehicle prices are spread across the dataset for similar configurations.
6. **Correlation Heatmap:** Provides a matrix-style heatmap illustrating correlations between vehicle attributes and final price.

The combination of rich visualizations and robust performance metrics makes the system both reliable and user-friendly. Users gain confidence not only from the numeric outputs but also from the graphical insights that contextualize predictions within broader market dynamics.

**11.3 Benefits and Limitations**

The Vehicle Price Prediction System brings several impactful advantages:

1. **User-Friendly Interface**: Streamlit-based design with tabs for prediction, comparison, and exploration.
2. **Custom vs Dataset Comparison**: Users can compare predicted values with actual dataset entries.
3. **Interactive Visualizations**: Graphical summaries enhance transparency and engagement.
4. **Export Option**: Predictions can be saved as .csv for record keeping of the predicted price.
5. **Randomization Tool:** Speeds up testing with one-click with random filling.
6. **Expandable Design:** Modular architecture for easy integration of new models and datasets.

However, the system also has some limitations:

1. **Dataset Constraints:** The system is trained on the Unified Provided Dataset, which may not generalize perfectly to the modern day phone prices with features is not accurate and outdated.
2. **Static Dataset:** Current implementation requires manual dataset updates, as no live scraping is integrated.
3. **Confidence Intervals:** While provided, they may widen for rare or uncommon vehicle configurations.

These limitations provide direction for future improvements and scalability considerations. Nevertheless, it serves as a strong proof-of-concept and fulfil the objective for an intelligent Prediction systems.

**11.4 Achievements and Conclusion**

**11.4.1 Achievements**

This project demonstrates a complete machine learning lifecycle, from data preprocessing and model training to deployment and interactive visualization. Key accomplishments include:

1. Successfully trained and integrated a high-accuracy XG Boosting Regression Model.
2. Developed a St interface with functional tabs covering: Single Prediction, Batch Prediction, and Dataset Exploration.
3. Incorporated downloads options along with an option to delete it as well and change it as well and downloading the dataset for exploration.
4. Incorporated custom vs dataset comparison tools with visual support.
5. Built a Modular, Maintainable and Extendable Architecture for future scaling,

**11.4.2 Use Case**

The Vehicle Price Prediction and Comparison System demonstrates broad applicability across multiple domains, bridging the gap between machine learning research, industry needs, and user convenience. Its versatility allows it to function as both a practical valuation tool and a learning resource:

a) Car Buyers & Sellers

1. For Buyers: Provides instant price estimates to ensure they are not overpaying for a used car and allows comparisons between different models, trims, and years to find the best value-for-money option.
2. For Sellers: Offers a benchmark for fair pricing before listing a vehicle for sale along with Prevents undervaluing, ensuring the seller receives a price aligned with market standards also helps in facilitates quicker sales by building trust with buyers who see transparent, data-driven valuations.

b) Data Science Projects: The System serves as a regression-based template project for data science practitioners along with Data preprocessing and feature engineering and Deployment of the model.

c) Educational Demonstrations: The application is a complete end-to-end ML pipeline, making it highly suitable for teaching and demonstration purposes.

**11.4.3 Conclusion**

The Vehicle Price Prediction and Comparison System represents the successful integration of data-driven machine learning models with a user-friendly and interactive interface, resulting in a tool that is both powerful and practical. By enabling real-time price estimation, interactive data exploration, and meaningful side-by-side comparisons, the system provides end-users with actionable insights that can directly influence decision-making in the highly dynamic automobile market.

Advanced regression techniques, feature engineering, and model evaluation processes operate seamlessly in the background, while the front-end interface remains intuitive enough for individuals with little or no technical expertise. This balance ensures that the tool can serve a diverse audience ranging from everyday car buyers and sellers to dealerships, analysts, and students learning machine learning concepts.

Beyond its immediate functionality, the project demonstrates the potential of machine learning systems as scalable frameworks for real-world deployment. What currently exists as a prototype can be further extended to incorporate live data integration from car listing platforms, mobile application deployment for on-the-go usage, and multilingual support to broaden accessibility across global markets. Additionally, incorporating feedback loops, adaptive learning, and cloud-based services could transform the system into a fully autonomous solution that continuously improves with new data.

In conclusion, the Vehicle Price Prediction and Comparison System is more than a demonstration of technical implementation; it is a proof-of-concept for the fusion of predictive modeling, visualization, and interactive design. It establishes a foundation that can inspire both academic exploration and industrial adoption, highlighting how intelligent systems can provide clarity, transparency, and confidence in markets traditionally characterized by uncertainty and negotiation.

**Reference**

1. **Scikit-learn Documentation**,  
   <https://scikit-learn.org/stable/>
2. **Streamlit Documentation**,  
   [https://docs.streamlit.io/](https://docs.streamlit.io/" \t "_new)
3. **Pandas Library Documentation**,  
   <https://pandas.pydata.org/docs/>
4. **Matplotlib and Seaborn – Data Visualization Tools**,  
   [https://matplotlib.org/](https://matplotlib.org/" \t "_new)  
   <https://seaborn.pydata.org/>
5. **Jupyter Notebook Documentation**,  
   <https://jupyter.org/documentation>
6. **Python Official Documentation**,  
   [https://docs.python.org/3/](https://docs.python.org/3/" \t "_new)