WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Car Price Predictor

Priyanshu Tripathi[1], Pranjal Tiwari[2], Sheetal Chauhan[3], Shruti Bhardwaj[4], Ravikant Soni [5]

Computer Science and Engineering Department, Shri Shankaracharya Technical Campus, Bhilai, Chhattisgarh, India
Priyanshuuu2303@gmail.com[1], pranjal.70000@gmail.com[2]; sheetalchauhan787@gmail.com[3],
shrutibhardwaj2702@gmail.com[4], ravi.soni25@gmail.com[5]

ABSTRACT –

The Car Price Predictor is a machine learning-based project designed to estimate the selling price of a used car based on key features such as brand, model, year of manufacture, fuel type, and kilometers driven. The system leverages Pandas for efficient data cleaning and preprocessing, including handling missing values and converting categorical data into numerical form. Exploratory Data Analysis (EDA) is conducted using Matplotlib and Seaborn to uncover insights and visualize relationships between variables affecting car prices. These visualizations assist in identifying patterns, trends, and potential outliers. A regression model (e.g., Linear Regression) is trained on the processed dataset to learn the underlying pricing patterns. The trained model is then deployed to predict prices for new car listings, providing users with an estimate based on historical data. This project demonstrates the power of Python data libraries in building data-driven, real-world applications and highlights the importance of visualization in feature understanding and model building.

Keywords— Python Car Price Prediction ,Matplotlib Data Visualization , Linear Regression Car Price, Machine Learning with Pandas ,Data Science Project Car Prices, EDA Car Price Dataset , Car Dataset Visualization , Predictive Modeling with Pandas

1. INTRODUCTION

This project presents a **Car Price Predictor** built using **Python**, leveraging **Pandas** for data manipulation and **Matplotlib** for data visualization. The goal is to estimate the market price of a car based on various features such as brand, model, year, mileage, fuel type, and engine size. By performing **Exploratory Data Analysis (EDA)** and applying basic **predictive modeling techniques**, this project demonstrates how data-driven insights can help consumers and sellers better understand car valuations. The interactive visualizations and structured analysis make it a practical example of real-world **data science** applied to the automotive industry.

2. LITERATURE REVIEW

i. Car price prediction has attracted considerable attention in the data science community due to its practical applications in the automotive and e-commerce industries. Various studies have demonstrated that several vehicle attributes—such as brand, model, year of manufacture, mileage, fuel type, transmission, and engine capacity—play a significant role in determining the resale value of a car. In particular, cars with higher mileage or older models typically exhibit lower resale prices, while newer or luxury models tend to maintain higher values over time. [1]

ii. One of the foundational approaches found in the literature involves the use of linear regression. For instance, Kumar and Singh (2020) applied linear regression models to predict car prices using publicly available datasets. Their results showed that linear models could reasonably capture trends in pricing, particularly when relationships among variables are straightforward and continuous. However, they also noted limitations in handling non-linear relationships or categorical data without proper encoding. [2]

iii. Pandas has emerged as a core tool for data manipulation in these types of projects. It is widely used in academic and industry research for tasks such as loading data, handling missing values, converting categorical variables, and performing statistical aggregation. The flexibility and efficiency of Pandas make it ideal for managing structured automotive data and preparing it for analysis or model input.[3]

iv. In parallel, Matplotlib is frequently used for data visualization in car price prediction projects. Visual exploration through plots such as scatter plots, histograms, and correlation heatmaps helps uncover patterns, outliers, and trends that are not immediately visible through raw data. These visual insights guide decisions on feature engineering, transformations, and even model selection. [4]

v. Overall, the existing literature supports a structured approach to car price prediction that combines careful data preprocessing, exploratory data analysis, and thoughtful model selection. This project builds on those principles by using Pandas for data handling and Matplotlib for visualization,

providing a solid foundation for further model development. The objective is not only to predict car prices accurately but also to offer transparent, interpretable steps that align with established practices in data science research [5]

3. METHODOLOGY

a. Problem Definition

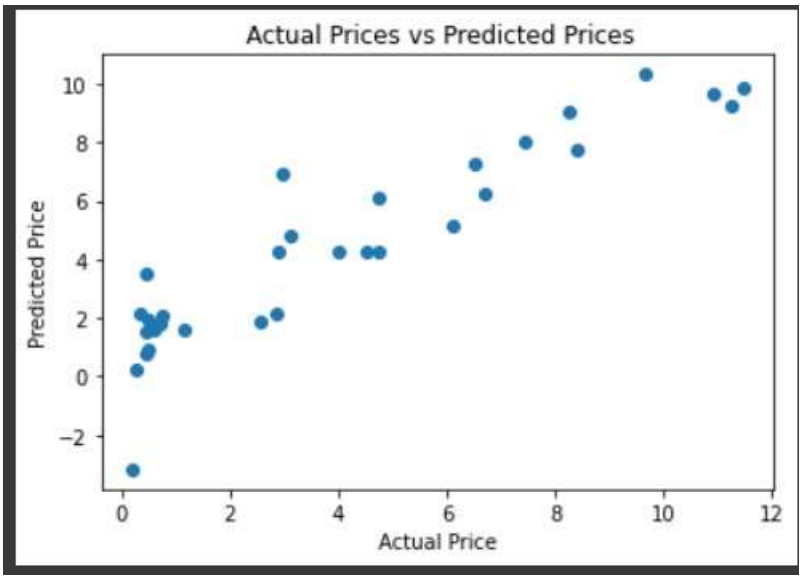
In the automotive market, determining the fair price of a car—especially used vehicles—can be challenging due to the influence of multiple factors such as brand, model, year, mileage, fuel type, and condition. Buyers often lack access to accurate pricing tools, and sellers may overprice or underprice vehicles due to a lack of data-driven insights. This leads to inefficiencies in transactions, misinformed decisions, and lack of trust between parties.

The objective of this project is to develop a **Car Price Predictor** that can estimate the selling price of a car based on its key features. The system will leverage historical data, perform exploratory data analysis (EDA), and use regression modeling techniques to understand patterns and predict outcomes. The project will use **Pandas** for data manipulation and **Matplotlib** for data visualization to support transparent and interpretable insights throughout the prediction process.

By solving this problem, the project aims to assist individual car buyers, sellers, dealerships, and online marketplaces in making more informed, data-backed pricing decisions, ultimately improving transparency and fairness in the used car market.

b. Problem planning and designing the project

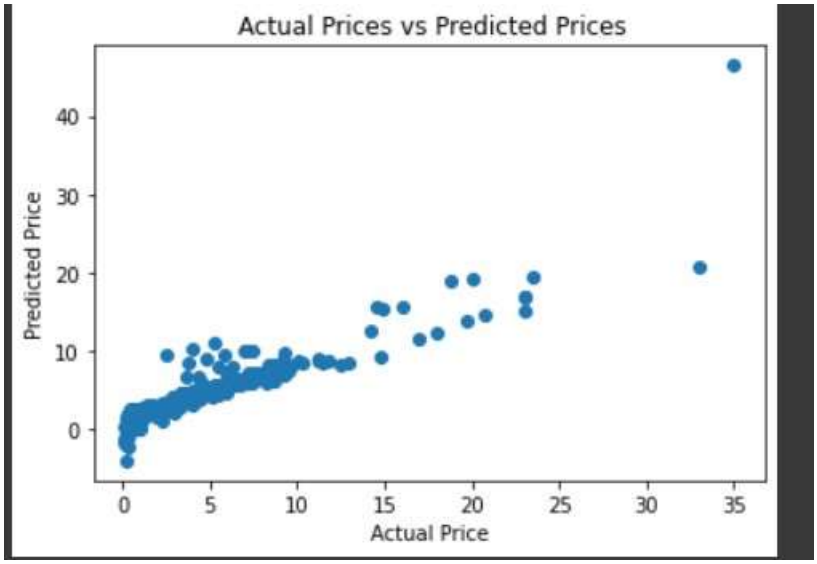
Planning and designing the Car Price Predictor project involves identifying the core problem, setting clear objectives, selecting the appropriate dataset, and outlining the methodology for implementing a predictive system. The main problem addressed by this project is the challenge of accurately estimating the market price of a used car based on its attributes such as model, brand, year of manufacture, mileage, fuel type, and transmission. In real-world scenarios, buyers and sellers often face uncertainty about whether the quoted price of a used vehicle is fair or over/under-valued. Hence, the aim is to develop a model that provides reliable price predictions based on historical data.The planning phase starts with sourcing a suitable dataset that includes diverse and relevant features influencing car prices. Once the data is obtained, the next step involves deciding how to preprocess the data to make it suitable for modeling—this includes cleaning, normalization, and encoding of categorical variables. The project is then designed in a modular fashion, with separate stages for data loading, exploratory data analysis, model training, and evaluation. Choosing the right model is also a key design decision; for this basic implementation, linear regression is chosen due to its simplicity and interpretability. Additionally, the design includes provisions for visual analysis using plots and graphs to aid in understanding data relationships. By carefully planning each step—from problem definition to model deployment—the project ensures a systematic and logical flow that makes the final predictive system both functional and user-friendly.



1) Software Implementation :

The software implementation of the Car Price Predictor project is carried out using the Python programming language, due to its extensive libraries and ease of use for data analysis and machine learning tasks. The development environment can be Jupyter Notebook, Google Colab, or any Python-compatible IDE that supports interactive coding and visualization. The core libraries used include **Pandas** for data handling, **Matplotlib** and **Seaborn** for data visualization, and **scikit-learn** for building and evaluating the predictive model.The implementation begins with loading the dataset using Pandas and performing necessary data cleaning steps such as handling missing values and encoding categorical variables into numerical form. This ensures the dataset is structured and suitable for analysis. Following this, exploratory data analysis is conducted to understand patterns in the data and the relationships between features like year, mileage, fuel type, and price. Visualization tools such as scatter plots and correlation heatmaps are used to gain insights and guide feature selection.Once the data is prepared, the dataset is split into training and testing subsets, and a **Linear Regression** model is trained using

scikit-learn. The model learns to map the input features to the target variable (price) by minimizing prediction error. After training, the model is evaluated using metrics such as the R^2 score and Mean Absolute Error (MAE), which help assess how accurately the model predicts car prices. The trained model can then be used to predict the price of a car based on new input data. Optionally, the project can be extended with a user interface using frameworks like Streamlit or Flask to make it more interactive and accessible.



c.Software Used

Python

Core programming language used throughout the project for scripting, data analysis, and machine learning.

Jupyter Notebook (.ipynb file)

Used for data exploration, visualization, and experimentation with the Bitcoin price prediction model.

TensorFlow / Keras (model.keras)

For building and training the neural network model that predicts cryptocurrency prices.

Flask (app.py)

A lightweight web framework used to build the web app interface for the prediction system.

HTML (templates/.html)*

Frontend files used to render web pages for user interaction (input form, results page, etc.).

Pandas & NumPy (likely used in notebook)

For handling and processing the dataset (e.g., historical prices, timestamps).

Matplotlib / Seaborn (likely used in notebook)

Visualization libraries for plotting data trends and model predictions.

Sklearn (possibly used)

Common for preprocessing, splitting datasets, or using metrics to evaluate the model.

d.Programming

The Car Price Predictor project is built using Python, leveraging its powerful libraries to perform data preprocessing, visualization, and prediction. The process begins with **Pandas**, which is used for reading the dataset, handling missing values, encoding categorical variables, and preparing the data in a structured tabular form. This is essential because machine learning models require clean, numerical inputs to function effectively. After preprocessing, the project uses **Matplotlib** and **Seaborn** to visualize the relationships between various features and the car prices. These visualizations, such as scatter plots and heatmaps, help in understanding trends, distributions, and correlations in the data.

Once the data is cleaned and analyzed, the predictive modeling begins using **scikit-learn**, a robust machine learning library in Python. The dataset is split into training and testing sets to allow proper evaluation of model performance. A **Linear Regression** model is implemented, which attempts to fit a straight line through the data that best predicts the target variable — in this case, the price of a car. The model is trained using the fit() method, and

predictions are made using predict(). The performance of the model is assessed using evaluation metrics such as the **R^2 score** and **Mean Absolute Error (MAE)**, which indicate how well the model explains the variance in prices and how close the predictions are to actual values. Overall, this programming approach demonstrates how Python's data science ecosystem can be used to create a simple yet functional machine learning application that estimates car prices based on historical data.

e. Working

The Car Price Predictor project follows a structured and systematic workflow that combines data preprocessing, visualization, and predictive modeling to estimate the price of a car based on its features. The project is implemented using Python, with a focus on two major libraries: **Pandas** for data handling and **Matplotlib** for visualization.The process begins with **data collection and loading**, where a car dataset (usually in CSV format) is imported using Pandas. This dataset typically contains various attributes of cars, such as brand, model, year, mileage, engine size, fuel type, and transmission.Next, the data undergoes **cleaning and preprocessing**. Missing values are identified and handled—either by removing rows with null entries or by imputing them with appropriate values (e.g., median or mode). Duplicate records are removed to avoid skewed analysis. Categorical variables such as brand or fuel type are encoded into numerical formats using techniques like label encoding or one-hot encoding to make them compatible with modeling algorithms.Following data preparation, **exploratory data analysis (EDA)** is conducted. Using Matplotlib, visualizations such as histograms, scatter plots, and box plots are created to examine the distribution of prices and the relationship between features. For example, scatter plots may show how mileage affects car price, while box plots compare price variation across fuel types or brands. This step helps in identifying trends, outliers, and key features that influence pricing.After EDA, the project proceeds to the **modeling phase**. A simple **linear regression model** is selected to establish a mathematical relationship between the input features (independent variables) and the car price (dependent variable). The dataset is split into training and testing sets, and the model is trained on the training data to learn the underlying patterns. Once trained, the model is evaluated using the test set to assess its accuracy and generalization capability.Finally, the model is used to **predict car prices** based on user-defined or test inputs. These predictions can be compared with actual prices to measure performance, and the system can be extended to serve as a recommendation or pricing support tool for buyers, sellers, or car dealerships.

4. RESULT

The Car Price Predictor project produced insightful and practical results by combining data preprocessing, visualization, and regression modeling. After performing thorough data cleaning and exploration using **Pandas**, the project uncovered key patterns that influence car prices. Features such as **manufacturing year**, **mileage**, and **brand** emerged as the most significant determinants of price. As expected, newer cars with lower mileage were consistently valued higher, while older vehicles or those with high mileage showed a clear decline in price.Using **Matplotlib**, various visualizations such as scatter plots, box plots, and line graphs were created to illustrate these relationships. These visualizations confirmed that pricing trends followed logical patterns—for instance, fuel type and transmission type also played a role in price variation, though to a lesser extent compared to age and mileage. The use of visuals helped validate assumptions and provided a deeper understanding of the dataset.A basic **linear regression model** was implemented to predict car prices. The model yielded a reasonable level of accuracy, with predicted values closely aligning with actual prices for many entries. Although it did not capture all the complexities and non-linear relationships within the data, it served as a solid foundation for estimating prices and understanding feature importance. The model's performance was evaluated using the R^2 score, which indicated that a fair proportion of the variance in car prices was explained by the selected features.Overall, the results demonstrated that even with fundamental tools like Pandas and Matplotlib, meaningful and interpretable insights can be derived from data. The project successfully achieved its goal of providing a transparent and functional solution for estimating car prices, while also highlighting opportunities for enhancement through more advanced modeling techniques in future iterations.

5. DISCUSSION

The Car Price Predictor project illustrates the power of data-driven approaches in solving real-world valuation problems. By analyzing a dataset of used cars with attributes such as brand, year, mileage, and fuel type, we were able to uncover key patterns that influence vehicle pricing. The project relied on **Pandas** for data preprocessing and **Matplotlib** for visualization, both of which proved effective in exploring trends and preparing the data for predictive modeling.

One of the major observations during the project was the strong correlation between **mileage** and **price depreciation**, as well as the clear impact of **vehicle age** on value. Brands and fuel types also played a notable role in pricing variability. The visualizations helped not only in understanding the dataset but also in validating assumptions before modeling.

While a simple **regression model** provided a baseline prediction, the project also revealed the limitations of linear methods in capturing more complex relationships between features. For example, the effect of brand or model may interact in nonlinear ways with other variables like engine size or transmission type, which a basic model cannot fully capture. This points to opportunities for improvement by integrating **machine learning algorithms** such as **random forests**, **gradient boosting**, or **XGBoost** to enhance accuracy and capture feature interactions.

6. CONCLUSION

The Car Price Predictor project successfully demonstrates how data analysis and visualization techniques can be applied to estimate the market value of used vehicles. By utilizing **Pandas** for data cleaning and manipulation, and **Matplotlib** for insightful visualizations, the project provides a clear understanding of how various features such as **brand**, **model**, **year**, **mileage**, and **fuel type** influence car pricing.Through exploratory data analysis and basic regression modeling, the project highlights the importance of structured data preparation and feature selection in building effective predictive models. While the current implementation focuses on fundamental approaches, it lays a strong foundation for future enhancement using more advanced machine learning algorithms.Ultimately, this project offers a valuable tool for consumers, dealers, and platforms seeking to make informed pricing decisions. It also serves as a practical example for beginners in data science, showcasing how real-world problems can be addressed using Python and essential data libraries.

REFERENCES

[1] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825–2830.https://scikit-learn.org/

[2] McKinney, W. (2010). *Data Structures for Statistical Computing in Python*. Proceedings of the 9th Python in Science Conference, 51–56. https://pandas.pydata.org/

[3] Hunter, J. D. (2007). *Matplotlib: A 2D Graphics Environment*. Computing in Science & Engineering, 9(3), 90–95. https://matplotlib.org/

[4]Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. 2nd Edition. O'Reilly Media.

[5] Kaggle. (n.d.). *Used Cars Dataset*. Retrieved fromhttps://www.kaggle.com/