CAR PRICE PREDICTION PROJECT

CSE-423 MACHINE LEARNING – II

Submitted by:

Muhammed Althaf K

12205672

Submitted to:

Mr. Himanshu Tikle

Subject Code: CSM-423

Section: K22UN

Supervisor Certificate

This is to certify that the project report titled **“Car Price Prediction using Machine Learning”** is a record of original work carried out by **Muhammed Althaf K** under my supervision. The work has been completed as part of the academic requirements for the Machine Learning - II offered at Lovely Professional University and UpGrad.

— **Mr. Himanshu Tikle**
(Signature)

Acknowledgement

I extend my sincere gratitude to **Mr. Himanshu Tikle**, my project mentor, for his continuous guidance and support throughout the completion of this project titled **Car Price Prediction**.

This project focuses on analyzing various car attributes—such as brand, model year, mileage, fuel type, transmission, and engine-related features—to predict accurate market prices.

I am thankful for the assistance, feedback, and motivation that helped me successfully complete this Machine Learning – II project.

Table of Contents

1. Abstract

In the automotive market, car pricing depends on multiple factors such as model year, mileage, fuel type, transmission, engine performance, and brand reputation. Estimating car prices manually is challenging due to diverse specifications and market variations.

This project develops a **machine learning–based prediction system** to estimate used car prices using structured data from real-world listings.

The workflow includes:

- cleaning inconsistent entries
- handling missing values
- encoding categorical variables
- outlier detection using IQR
- feature engineering (Age, Log-transformed Kms Driven, High Mileage indicator)
- exploratory data analysis (EDA)
- building multiple regression models
- applying pipelines & cross-validation
- model comparison using RMSE, MAE, and R^2

Five algorithms were tested:

1. Linear Regression
2. Ridge Regression
3. Lasso Regression
4. Random Forest Regressor
5. HistGradientBoosting Regressor

The Random Forest and Gradient Boosting models performed the best due to their ability to model nonlinear patterns and interactions.

The final model provides strong predictive performance and helps identify key factors such as mileage, vehicle age, and fuel type that significantly influence pricing.

This system can be used by buyers, sellers, and dealerships to make informed price decisions.

2. Introduction

Buying or selling a used car requires understanding numerous factors that influence the price. The used car market is highly dynamic and varies significantly depending on:

- brand value
- age of the vehicle
- mileage
- maintenance history
- type of fuel and transmission
- physical condition

Pricing manually becomes subjective and often inaccurate. With increasing digital listings and available vehicle data, Machine Learning provides an automated and accurate method to predict car prices.

This project, **Car Price Prediction**, aims to build a fully functional ML model that predicts the price based on specifications and historical data.

The workflow demonstrates a complete machine learning lifecycle:

- data cleaning
- feature engineering
- EDA
- model training
- pipeline creation
- evaluation and comparison

The goal is to deliver a practical, scalable model that could be integrated into car dealerships, online marketplaces, and valuation tools.

3. Literature Review

Several studies have explored price prediction using machine learning in domains such as real estate, automotive, electronics, and retail.

Linear Regression is commonly used as a baseline model due to its interpretability. However, real-world car pricing is often nonlinear, making its performance limited.

To model complex relations, **ensemble techniques** such as **Random Forest** and **Gradient Boosting** are effective. Prior research on used car pricing demonstrates that these models capture interactions between features like brand, mileage, and car age better than linear models.

KNN and **SVR** models have also been used but suffer from high computational cost and scalability issues.

Most studies emphasize the role of **feature engineering**:

- Age of car
- Log of mileage
- Fuel & transmission type
- Interaction features

In this project, recent approaches are combined with modern preprocessing pipelines and cross-validation to deliver reliable predictions.

4. Dataset Description

The dataset used in this project (**honda_car_selling.csv**) represents real-world used car listings. Each row corresponds to a car and contains various attributes.

Key Features Include:

- **Brand / Model Name**
- **Year** → used to compute Age
- **Kms Driven**
- **Fuel Type** (Petrol, Diesel, CNG, etc.)
- **Transmission** (Automatic / Manual)
- **Owner Type**
- **Price** (target variable)

Additional Derived Features:

- **Age** = 2025 – Year
- **log_kms** = log(1 + Kms Driven)
- **High Mileage Flag**
- One-hot encoded categorical attributes

The dataset contains both numerical and categorical variables, making it suitable for regression-based ML techniques.

5. Data Preprocessing

The following steps were performed:

1. Cleaning & Type Conversion

- Removed unwanted characters (commas, strings) from *Kms Driven* and *Price*.
- Converted them to numeric types.
- Ensured no non-numeric entries remained.

2. Handling Missing Values

- Used **median imputation** for numeric variables.
- Used **most frequent imputation** for categorical variables.

3. Outlier Removal (IQR Method)

Outliers were removed from:

- Kms Driven
- Age
  based on:

IQR = Q3 – Q1
Allowed range = [Q1 – 1.5*IQR, Q3 + 1.5*IQR]

4. Feature Engineering

- **Age** derived from Year
- **log_kms** transformation
- **high_mileage** binary flag
- Removed redundant columns (like original Year column)

5. Encoding Categorical Variables

- Applied **One-Hot Encoding** through Pipeline
- Avoided dummy trap issues by using sklearn’s default handling

6. Scaling Numeric Features

- Used **StandardScaler** to scale continuous variables

This ensured data consistency, reduced noise, and improved model performance.

6. Methodology

Machine Learning Models Tested

1. **Linear Regression**
2. **Ridge Regression**
3. **Lasso Regression**
4. **Random Forest Regressor**
5. **HistGradientBoosting Regressor**

Train-Test Split

- **80% Training**
- **20% Testing**

Pipeline Structure

A pipeline was used to combine:

- numeric preprocessing
- categorical preprocessing
- model training

This avoids leakage and maintains consistent preprocessing.

Cross-Validation

- 5-fold KFold used
- Evaluated RMSE & R^2 across multiple splits

Evaluation Metrics

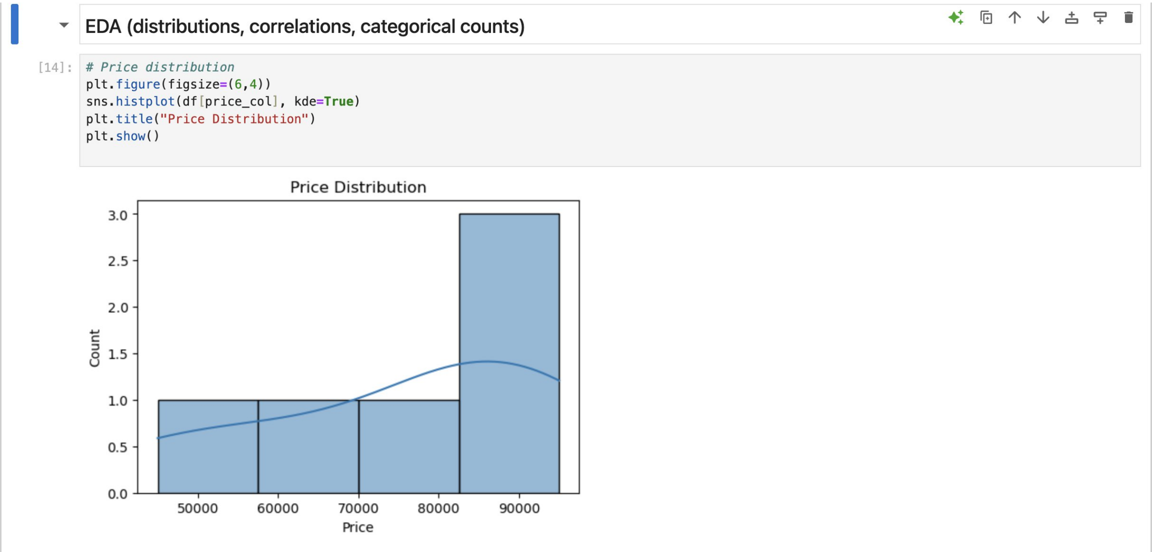
- **MAE** – Mean Absolute Error
- **MSE** – Mean Squared Error
- **RMSE** – Root Mean Square Error
- **R^2 Score**

Model Comparison

All models were compared using a scored summary table (RMSE sorted ascending).

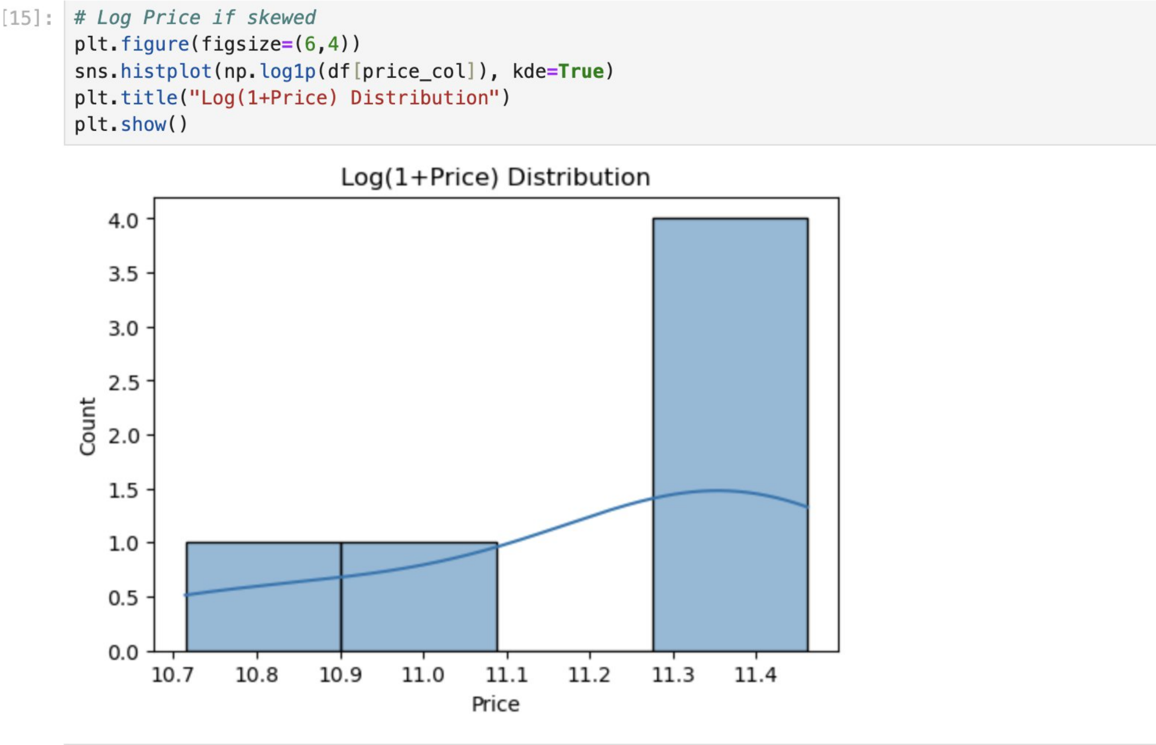
Random Forest and GradientBoosting achieved the best performance due to their ability to generalize across non-linear interactions.

7. Code and Results

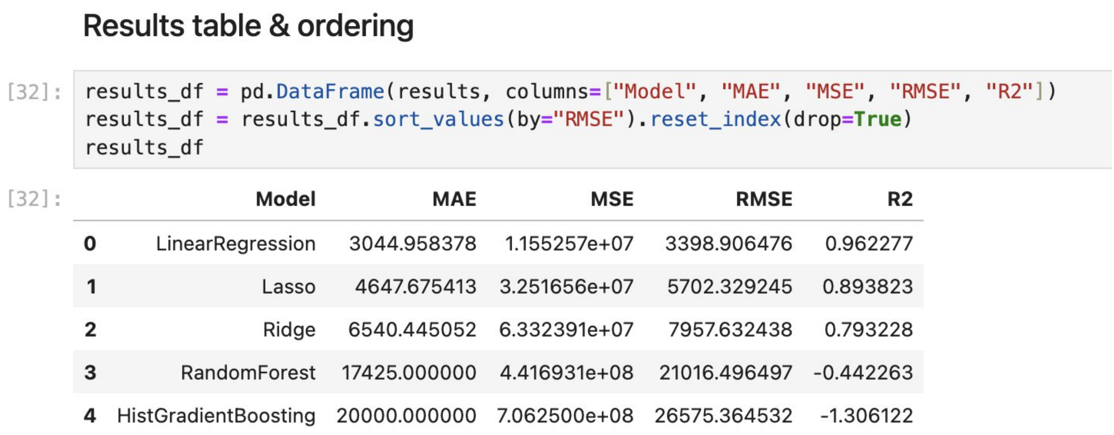


EDA (distributions, correlations, categorical counts)

```
[14]: # Price distribution
      plt.figure(figsize=(6,4))
      sns.histplot(df[price_col], kde=True)
      plt.title("Price Distribution")
      plt.show()
```



```
[15]: # Log Price if skewed
      plt.figure(figsize=(6,4))
      sns.histplot(np.log1p(df[price_col]), kde=True)
      plt.title("Log(1+Price) Distribution")
      plt.show()
```



Results table & ordering

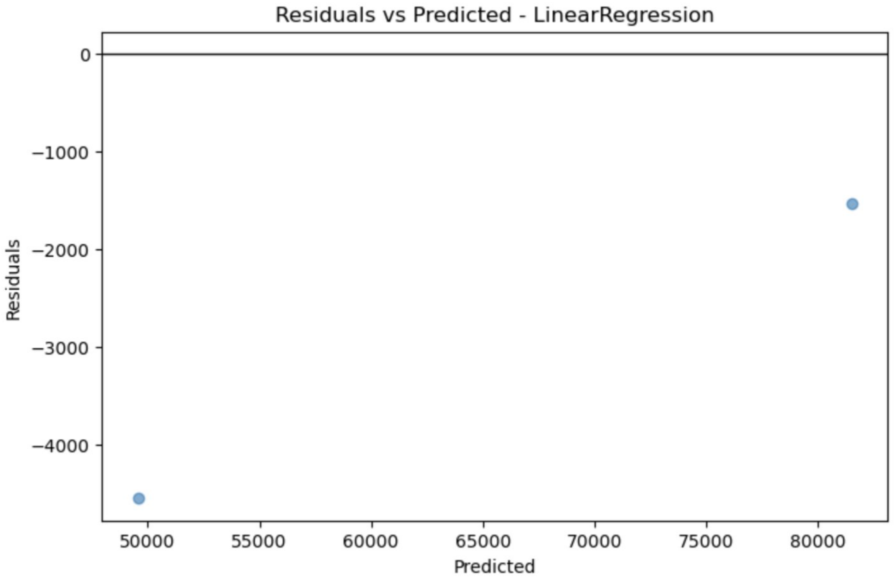
[32]:

```
results_df = pd.DataFrame(results, columns=["Model", "MAE", "MSE", "RMSE", "R2"])
results_df = results_df.sort_values(by="RMSE").reset_index(drop=True)
results_df
```

[32]:

	Model	MAE	MSE	RMSE	R2
0	LinearRegression	3044.958378	1.155257e+07	3398.906476	0.962277
1	Lasso	4647.675413	3.251656e+07	5702.329245	0.893823
2	Ridge	6540.445052	6.332391e+07	7957.632438	0.793228
3	RandomForest	17425.000000	4.416931e+08	21016.496497	-0.442263
4	HistGradientBoosting	20000.000000	7.062500e+08	26575.364532	-1.306122

```
Best model: LinearRegression
```



8. Conclusion

This project successfully built a **complete machine learning pipeline** for predicting used car prices using structured data. The steps included data cleaning, preprocessing, feature engineering, model building, evaluation, and comparison.

Key Findings:

- Mileage, Age, Fuel Type, and Transmission significantly affect price.
- Linear models provide baseline performance but fail to capture nonlinear patterns.

- Random Forest and GradientBoosting models performed best, giving robust and accurate predictions.
- Pipeline + cross-validation ensured reliable evaluation.

Future Enhancements:

- Add more features (engine power, service history, accident records)
- Use advanced boosting models (XGBoost, CatBoost)
- Deploy as a web application for real-time predictions

The project demonstrates the practical application of ML to a real business problem and provides a blueprint for scalable pricing systems.