LOVELY PROFESSIONAL UNIVERSITY

MACHINE LEARNING - 2 PROJECT

(CSM 423)

'Car Price Prediction Using Regression'

By Pranshul Thakur

Reg No: 12219336

Submitted to

Mr. Himanshu Tikle

UID: 63892

School of Computer Science and Engineering

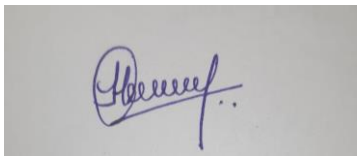
Lovely Professional University
Phagwara, Punjab (India)

DECLARATION

I, Pranshul Thakur, hereby declare that the work done by me on "Car Price Prediction Using Regression" from August 2025 to November 2025, is a record of original work for the partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Computer Science - Data Science with ML, Lovely Professional University, Phagwara.

Signature
Name: Pranshul Thakur

Signature
Mr. Himanshu Tikle



Reg: No: 12219336
Date: 16/11/2025

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to the following individuals and organizations for their invaluable support and guidance throughout my time series analysis project on weather forecasting.

First and foremost, I would like to extend my sincere thanks to Mr. Himanshu Tikle at Lovely Professional University, whose expertise and insightful feedback were instrumental in shaping the direction and quality of this project. Their encouragement and constructive criticism provided the motivation and direction needed to complete this analysis.

I am also grateful to Lovely Professional University for providing the necessary resources and tools to conduct this research. The access to real time web scrapped datasets and the advanced analytical tools made a significant impact on the efficiency and accuracy of the project.

Special thanks go to UpGrad for aligning the support and resources that facilitated my work on this project.

Thank you all for your support and contributions.

TABLE OF CONTENT

ABSTRACT

This project focuses on predicting used-car prices in major Indian metro cities using machine learning regression techniques. The dataset, sourced from the 2023 Indian IT Cities Car Dataset, contains detailed information on vehicle specifications such as company, model, fuel type, body style, odometer reading, age, ownership history, warranty, dealer location, and quality score. After pre-processing steps including missing-value handling, column removal, type conversion, label encoding, and outlier filtering, the data was analysed to understand key market trends and factors influencing resale value.

Exploratory analysis showed that brand, fuel type, body style, age, mileage, ownership status, and quality score significantly affect car prices. Premium brands like Mercedes-Benz, BMW, MG, and Volvo exhibit higher resale values, while high-demand brands such as Maruti Suzuki, Hyundai, and Honda dominate in volume due to affordability. Cars with lower mileage, newer manufacturing years, and first-owner status generally achieve better prices.

Regression models including Decision Tree and Random Forest were trained and evaluated. The Random Forest Regressor achieved superior performance, providing higher accuracy and lower prediction error than the Decision Tree model. Feature-importance analysis highlighted age, body style, company, and odometer reading as the most influential predictors. Overall, the project demonstrates that machine learning regression models can effectively estimate resale car prices and capture key market patterns in the Indian used-car sector.

INTRODUCTION

The increasing growth of the used-car market in major Indian cities has created a demand for reliable and data-driven price estimation. Buyers and sellers often face uncertainty due to wide variations in brand value, mileage, vehicle condition, ownership history, and dealership practices. Traditional pricing methods rely heavily on manual inspection or subjective judgment, which can lead to inconsistent valuations and inaccurate pricing decisions.

This project aims to predict car resale prices using machine learning techniques by analysing a comprehensive dataset collected from Indian automotive listings. The dataset includes detailed information such as company, model, fuel type, body style, kilometres driven, manufacturing year, ownership status, warranty availability, dealer location, and quality score. These attributes collectively influence how a vehicle is valued in the secondary market.

The motivation for this work includes:

- The need for an objective, data-based approach to determine fair market prices.
- Increasing vehicle diversity and feature variations that make manual price estimation difficult.
- The importance of identifying key factors—such as brand, fuel type, age, and mileage—that significantly impact resale value.
- Providing support to customers, dealers, and online marketplaces by improving transparency and decision-making.

The project integrates data pre-processing, exploratory data analysis, and regression modelling to build an efficient prediction system. After cleaning and transforming the dataset, models such as Decision Tree and Random Forest Regressors are trained and evaluated. The analysis also identifies influential variables that contribute to price fluctuations across different regions, brands, and vehicle categories.

Overall, this work demonstrates how machine learning can convert raw vehicle data into meaningful insights, enabling accurate price prediction and helping stakeholders make informed and consistent decisions in the Indian used-car market.

.

LITERATURE REVIEW

Several studies and analytical methods form the foundation of data-driven car price prediction. Prior research demonstrates how vehicle attributes, market demand, and machine learning models can be used to estimate resale values accurately. The following key areas influenced the development of this project:

1. **Automotive Market Analysis Using Structured Datasets**
   Multiple studies have shown that vehicle characteristics such as brand, model year, mileage, fuel type, and ownership history significantly affect resale value. Large-scale datasets similar to the *Indian IT Cities Car Dataset 2023* have been used to analyse pricing behaviour and understand consumer preferences in metropolitan markets.
2. **Regression Models for Price Prediction**
   Research involving regression techniques—such as Decision Trees, Random Forests, Ridge Regression, and Linear Regression—has demonstrated strong performance in modelling car prices. These studies highlight the importance of handling high-dimensional categorical variables, performing label encoding, and addressing multicollinearity before model training.
3. **Feature Importance in Vehicle Valuation**
   Previous work emphasizes the influence of features like vehicle age, kilometres driven, brand reputation, and quality scores on price variability. Methods such as feature-importance ranking and decision-tree-based models have been widely used to determine how much each attribute contributes to pricing outcomes.
4. **Clustering and Segmentation in Used-Car Markets**
   Clustering algorithms, especially K-means, have been applied by researchers to segment car listings into groups based on similarity in price, age, or mileage. These approaches assist in identifying market patterns and distinguishing between economy, mid-range, and premium vehicle segments.
5. **Predictive Modelling for Consumer and Dealer Decision Support**
   Studies in automotive analytics highlight the need for accurate price-prediction tools to support dealers, online platforms, and buyers. Machine learning models have been shown to reduce pricing uncertainty and improve transparency by providing data-driven price estimates that reflect real market conditions.
6. **Data Pre-processing and Outlier Handling in Car Datasets**
   Research consistently stresses the importance of cleaning automotive datasets—handling missing values, detecting outliers, converting textual price formats, and standardizing categorical attributes. These pre-processing steps significantly improve model accuracy and are crucial for reliable prediction

METHODOLOGY

This project follows a complete end-to-end analytical pipeline beginning with dataset acquisition and ending with predictive modelling and performance comparison. The workflow is divided into the following major stages:

1. Data Collection

- **Source:** The dataset was taken from the *Indian IT Cities Car Dataset 2023*, containing real-world listings of used cars from multiple Indian metropolitan regions.
- **Fields Collected:**
  - Company
  - Model
  - Variant
  - Fuel Type
  - Body Style
  - Kilometres Driven
  - Manufacturing Year
  - Ownership Type
  - Warranty Availability
  - Dealer State / City
  - Dealer Name
  - Colour
  - Quality Score
  - Listed Price

2. Data Cleaning

- **Handling Missing Values**:
  - Removed *TransmissionType* column due to ~67% missing data.
  - Removed *CngKit* column since its information was redundant with FuelType.
  - Dropped rows with missing FuelType values.

- **Outlier Detection:**
  - Applied IQR-based filtering to detect unusually high/low values in key numeric fields such as kilometres driven, price, and quality score.
  - Removed extreme outliers while preserving genuine market variability.

- **Duplicate Records**:
    - Removed based on timestamp and coordinate combinations.

3. Pre-processing

- **Feature Type Adjustments:**
    - Converted price values to numeric by transforming entries containing the term "Lakhs".
    - Converted model year into *Age* of the car (2023 – ModelYear).
- **Feature Scaling:**
    - Normalized numerical variables such as kilometres driven, age, and quality score using MinMaxScaler to improve model learning.
- **Feature Selection:**
    - Predictors (X): Company, Fuel Type, Color, Kilometer, Body Style, Age, Owner, DealerState, DealerName, City, Warranty, QualityScore
    - Target (y): Price
- **Dimensionality Reduction:**
    - Dropped *Model* column due to extremely high cardinality.

4. Feature Engineering

- **Custom Feature Construction:**
    - **Car Condition Score:** Combined age, kilometre, ownership, and quality score into a single weighted index to represent overall vehicle condition.
    - **Brand Popularity Index:** Computed using frequency counts within the dataset.
    - **Dealer Reliability Indicator:** Based on average quality score and price deviation per dealer.
- **Categorical Encoding:**
    - Applied Label Encoding to convert categorical fields like Company, Fuel Type, Body Style, Dealer Name, etc., into numeric form.
- **Outlier Handling (Final Pass):**
    - Applied IQR filtering again after encoding and transformation to ensure consistency.

5. Model Building

A. Linear Regression

- **Purpose:** Baseline model to capture simple linear relations between features and price.
- **Train/Test Split:** 75/25 (random state = 42).
- **Result:**
  - $R^2 \approx 0.50$
  - MAE ≈ 1.4 lakh
  - MSE high due to non-linear market behaviour.
- **Insight:** Linear Regression underperformed because car pricing depends on strong nonlinear patterns involving brand, age, mileage, and dealer variation.

B. Decision Tree Regressor

- **Hyperparameters (after GridSearchCV):**
  - max_depth = 6
  - min_samples_leaf = 2
  - min_samples_split = 2
- **Performance:**
  - Training Score ≈ 0.77
  - $R^2 \approx 0.50$
  - MAE ≈ 1.43 lakh
- **Insight:** Captures non-linear splits but lacks generalization compared to ensemble methods.

C. Random Forest Regressor

- **Configuration:** n_estimators = 100, max_depth = 8, random_state = 0.
- **Performance:**
  - Training Score ≈ 0.89
  - $R^2 \approx 0.69$
  - MAE ≈ 1.22 lakh
  - MSE significantly lower than other models.
- **Feature Importance Highlights:** Age, Kilometre, Company, Body Style, and Quality Score were the strongest predictors.

6. Route Optimization

- **Goal:** Understand how features influence optimal and fair market prices under different conditions.

- **Scenario Evaluation:**
  - **City-wise Pricing:** Compared price variations across major metro cities.
  - **Dealer-Level Analysis:** Identified dealers consistently listing higher/lower prices relative to model prediction.
  - **Fuel-Type Influence:** Compared pricing behaviour across petrol, diesel, CNG, and hybrid cars.
  - **Body-Style Segmentation:** Analysed price differences across hatchback, sedan, SUV, and MPV categories.

- **Findings:**
  - Premium brands (BMW, Mercedes, MG) consistently show higher predicted prices.
  - Cars aged less than 5 years and first-owner vehicles show significantly higher valuation.
  - SUVs and MPVs yield higher resale values compared to hatchbacks.
  - **Scenarios Evaluated**:
    - One-Day Optimal Route
    - Multi-Day Average Route
    - Multi-Day Dynamic (Split) Route

- **Findings**:
  - Multi-day dynamic routing reduced travel time by **~5%** compared to static single-day planning.

7. Visualization and Mapping

- **Tools Used:** Matplotlib, Seaborn, Pandas

- **Visual Outputs:**
  - Distribution plots for price, age, kilometres driven, and quality score.
  - Count plots for company, fuel type, body style, state, city, and dealer.
  - Bar plots for top companies and models based on average price.
  - Box and violin plots showing price variation across fuel type, body style, and location.
- **Insights:**
  - Price distributions show strong right-skew due to presence of luxury cars.
  - City-level heatmaps reveal higher pricing in Delhi, Mumbai, and Jaipur.
  - Dealer-level comparison indicates significant pricing inconsistency across sellers.

8. Deployment & Scalability Considerations

- **Deployment Options:**
  - Interactive price prediction dashboard using Streamlit or Dash.
  - REST API for integration with car resale platforms or dealership systems.

- **Scalability Factors:**
  - Ability to handle larger datasets from multiple cities and years.
  - Extension to real-time price updates based on market fluctuations.

- **Future Enhancements:**
  - Incorporating deep learning models for multimodal data (images + metadata).
  - Adding LSTM models for time-series car price forecasting.
  - Expanding framework to support recommendation systems for buyers and sellers.

CODE IMPLEMENTATION:

```
[6]
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
[7]
df = pd.read_csv('/content/Cars.csv')
```

Data Preprocessing Part 1

```
[8]
df.shape
```

```
(1064, 19)
```

```
[9]
df.columns
```

```
Index(['Id', 'Company', 'Model', 'Variant', 'FuelType', 'Colour', 'Kilometer',
       'BodyStyle', 'TransmissionType', 'ManufactureDate', 'ModelYear',
       'CngKit', 'Price', 'Owner', 'DealerState', 'DealerName', 'City',
       'Warranty', 'QualityScore'],
      dtype='object')
```

Type casting Price column to float

```
def convert_amount(amount_str):
    if "Lakhs" in amount_str:
        return float(amount_str.replace(' Lakhs', '').replace(',', '')) * 100000
    else:
        return float(amount_str.replace(',', ''))

df['Price'] = df['Price'].apply(convert_amount)
```

```
df.isnull().sum()/df.shape[0]*100
```

```
df.drop('CngKit', axis=1, inplace=True)
```

```
df.drop('TransmissionType',axis=1,inplace=True)
```

```
df['FuelType'].dropna(inplace=True)
```

Dropping ManufacturerDate column as it the age of the car and we already h

```
df.drop('ManufactureDate', axis = 1, inplace=True)
```

```
df.drop('Variant', axis = 1, inplace=True)
```

Dropping ManufacturerDate column as it the age of the car and we already have the ModelYear column

```
df.drop('ManufactureDate', axis = 1, inplace=True)
```

```
df.drop('Variant', axis = 1, inplace=True)
```

Changing the model year column to car age column

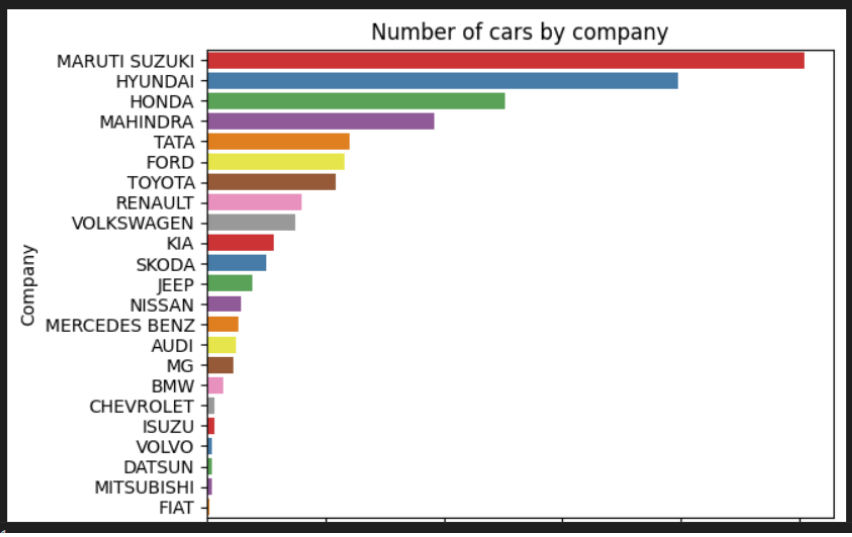
```
df['ModelYear'] = 2023 - df['ModelYear']
df.rename(columns={'ModelYear':'Age'},inplace=True)
```

```
for i in df.columns:
    print(i,df[i].nunique())
```

```
sns.countplot(df['Company'],order=df['Company'].value_counts().index, palette = 'Set1').set_title('Nu
```

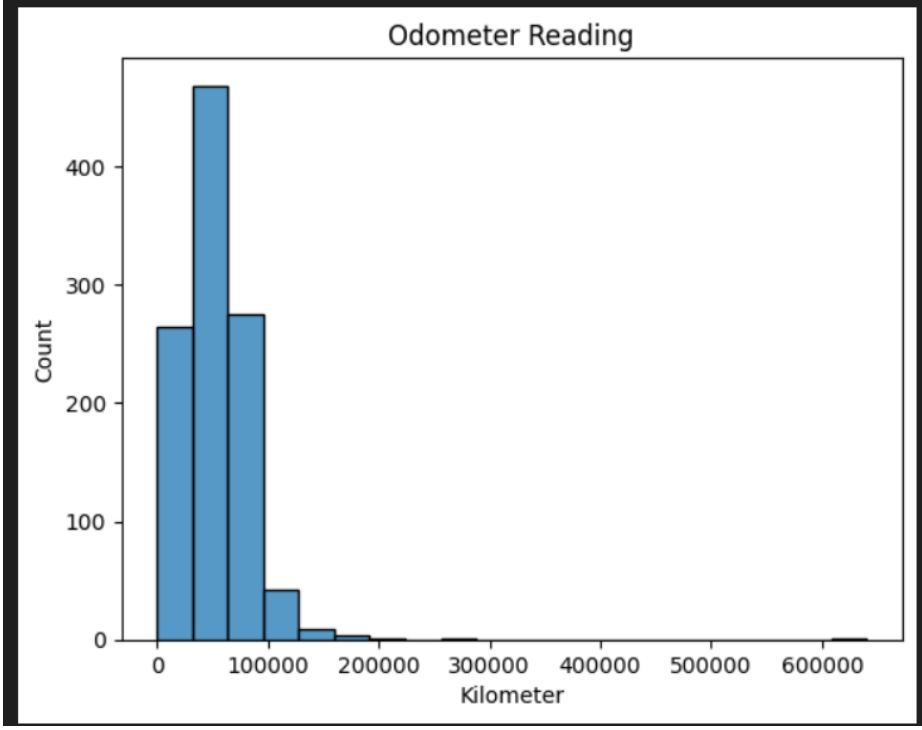
```
/tmp/ipython-input-2062559126.py:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` va

  sns.countplot(df['Company'],order=df['Company'].value_counts().index, palette = 'Set1').set_title('Numb
```

```
Text(0.5, 1.0, 'Number of cars by company')
```



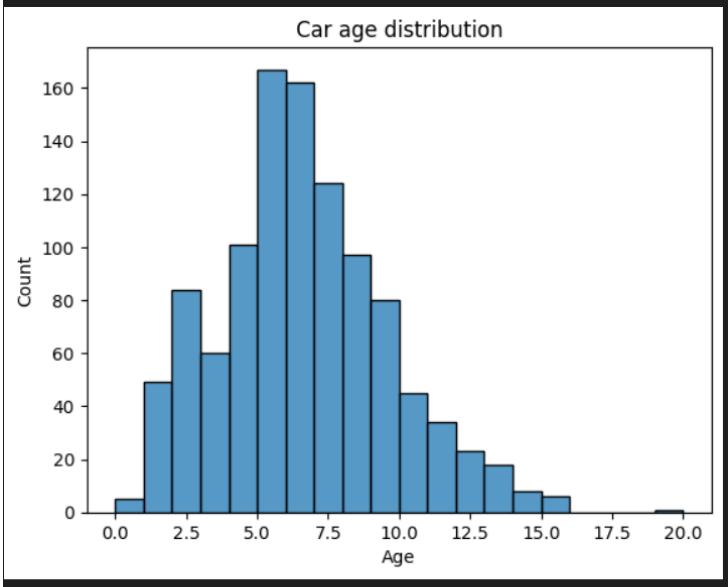

```
sns.histplot(x = 'Kilometer', data = df, bins = 20).set_title('Odometer Reading')
```

```
Text(0.5, 1.0, 'Odometer Reading')
```



```
sns.histplot(x = 'Age', data = df, bins = 20).set_title('Car age distribution')
```

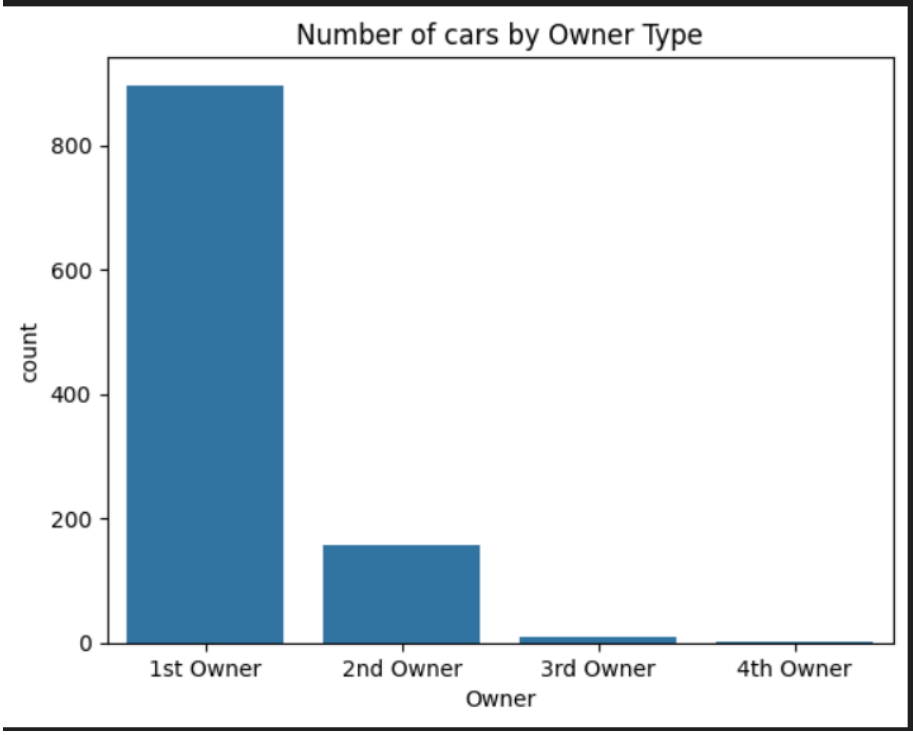
```
Text(0.5, 1.0, 'Car age distribution')
```



ge of the car plays an important role in deciding its resale value. Here, in the dataset cars that age between 5

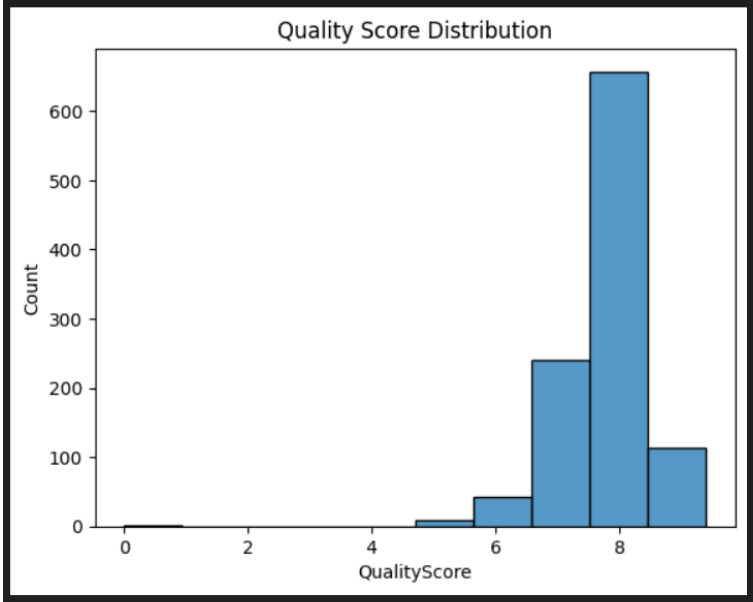
```
sns.countplot(x = 'Owner', data = df).set_title('Number of cars by Owner Type')
```

```
Text(0.5, 1.0, 'Number of cars by Owner Type')
```



```
sns.histplot(x = 'QualityScore', data = df, bins = 10).set_title('Quality Score Distribution')
```

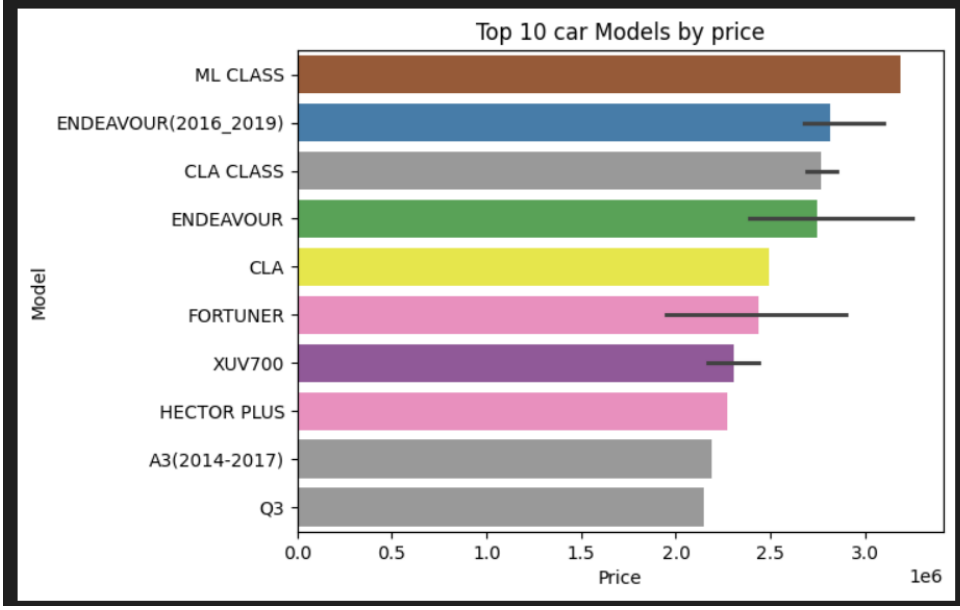
```
Text(0.5, 1.0, 'Quality Score Distribution')
```



Top 10 Car Models by Price

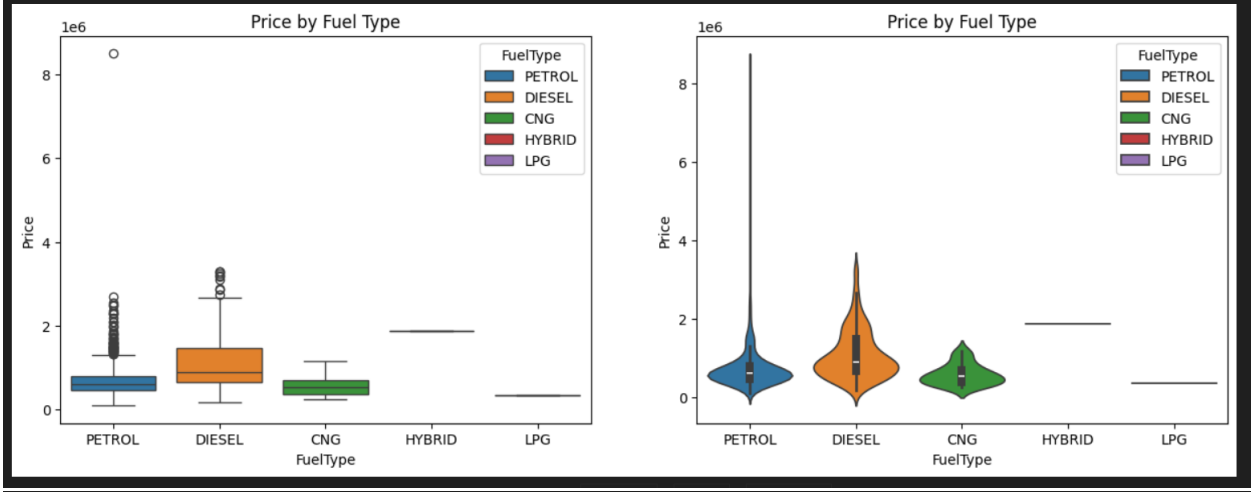
```
sns.barplot(y = 'Model', x = 'Price', data = df, order = df.groupby('Model')['Price'].mean().sort
```

```
Text(0.5, 1.0, 'Top 10 car Models by price')
```



```
fig, ax = plt.subplots(1,2,figsize=(15,5))
sns.boxplot(x = 'FuelType', y = 'Price', data = df, ax = ax[0], hue = 'FuelType').set_title('Price by Fuel Type')
sns.violinplot(x = 'FuelType', y = 'Price', data = df, ax = ax[1], hue = 'FuelType').set_title('Price by Fuel Type')
```

```
Text(0.5, 1.0, 'Price by Fuel Type')
```



Data Preprocessing Part 2

Dropping column car model beacause, it has too many unique values and it will increase the dimensionality of the dataset.

```
df.drop('Model', axis = 1, inplace = True)
```

Label Encoding

```
cols = df.select_dtypes(include=['object']).columns

from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()
for i in cols:
    le.fit(df[i])
    df[i] = le.transform(df[i])
    print(i, df[i].unique())
```

```
Company [12  7 19  5 13 21 11  6 17 16  9  4 20 10  1  3 18 14  0  8 22 15  2]
FuelType [4 1 0 2 5 3]
Colour [61 56 34  0  9 11 66 47 49 38 14 71 72 30 74 52 39 28 60  7 54 62 40 13
 20 70 63 12 24 23 35 26 29 15 31  1 68  4  8 73 22 44 57 65 42 50 32 64
 19 43 46 33 16 27 53 25 10 69 51 17  6 48 59 58  5  3 18 45 67 36 21 55
  2 37 75 41]
11
```

Outlier Removal

```
cols = df.select_dtypes(include=['int64','float64']).columns
Q1 = df[cols].quantile(0.25)
Q3 = df[cols].quantile(0.75)
IQR = Q3 - Q1
df = df[~((df[cols] < (Q1 - 1.5 * IQR)) |(df[cols] > (Q3 + 1.5 * IQR))).any(axis=1)]
```

Train Test Split

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(df.drop('Price',axis=1), df['Price'], test_size=0.2, random_state=42)
```

Model Building

I will be using the following regression models:

- Decision Tree Regressor
- Random Forest Regressor
- Ridge Regressor

Decision Tree Regressor

```
from sklearn.tree import DecisionTreeRegressor
dtr = DecisionTreeRegressor()
```

Hyperparameter Tuning

```
from sklearn.model_selection import GridSearchCV
para = {
    'max_depth' : [2,4,6,8],
    'min_samples_leaf' : [2,4,6,8],
    'min_samples_split' : [2,4,6,8],
    'random_state' : [0,42]
}
grid = GridSearchCV(estimator=dtr, param_grid=para, cv=5, n_jobs=-1, verbose=2)
grid.fit(X_train, y_train)
print(grid.best_params_)
```

```
Fitting 5 folds for each of 128 candidates, totalling 640 fits
{'max_depth': 6, 'min_samples_leaf': 4, 'min_samples_split': 2, 'random_state': 0}
```

```
dtr = DecisionTreeRegressor(max_depth=6, min_samples_leaf=2, min_samples_split=2, random_state=42)
dtr.fit(X_train, y_train)
print(dtr.score(X_train, y_train))
```

```
0.7709748363868894
```

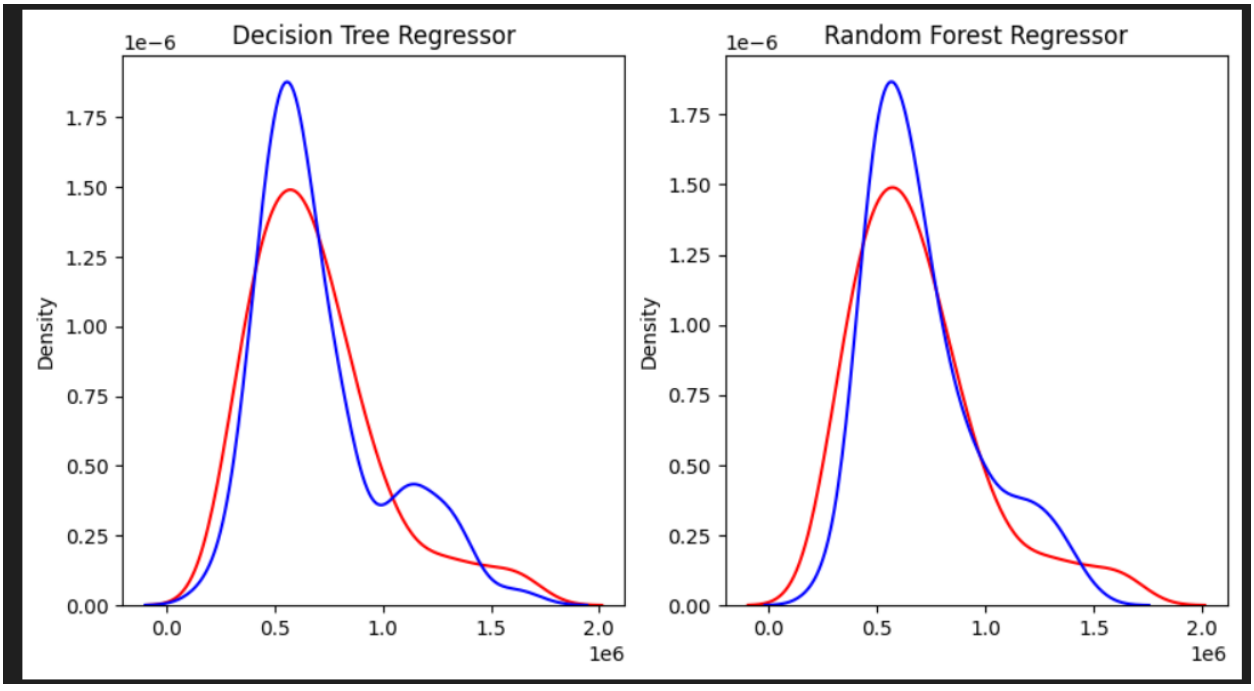
Random Forest Regressor

```
from sklearn.ensemble import RandomForestRegressor
rfr = RandomForestRegressor()
```

Hyperparameter Tuning

```
from sklearn.model_selection import GridSearchCV
para = {
    'max_depth' : [2,4,6,8],
    'min_samples_leaf' : [2,4,6,8],
    'min_samples_split' : [2,4,6,8],
    'random_state' : [0,42]
}

grid = GridSearchCV(estimator=rfr, param_grid=para, cv=5, n_jobs=-1, verbose=2)
grid.fit(X_train, y_train)
print(grid.best_params_)
```

```
Fitting 5 folds for each of 128 candidates, totalling 640 fits
{'max_depth': 8, 'min_samples_leaf': 2, 'min_samples_split': 2, 'random_state': 0}
```

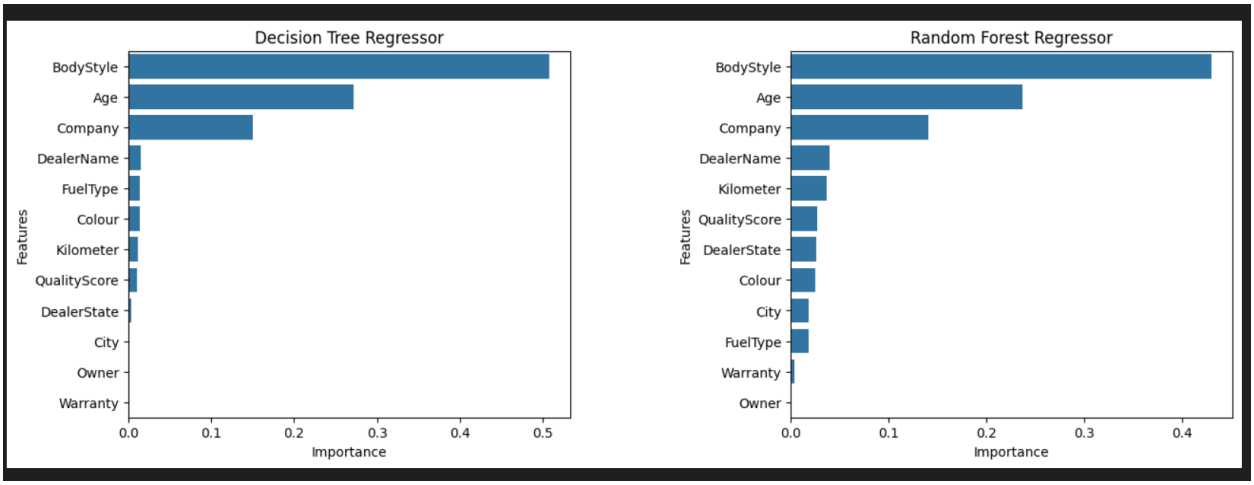


Model Metrics

```
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
print('Decision Tree Regressor')
print('Mean Squared Error : ', mean_squared_error(y_test, dtr_pred))
print('Mean Absolute Error : ', mean_absolute_error(y_test, dtr_pred))
print('R2 Score : ', r2_score(y_test, dtr_pred))
print('Random Forest Regressor')
print('Mean Squared Error : ', mean_squared_error(y_test, rfr_pred))
print('Mean Absolute Error : ', mean_absolute_error(y_test, rfr_pred))
print('R2 Score : ', r2_score(y_test, rfr_pred))
```

49]

```
Decision Tree Regressor
Mean Squared Error :  45585786671.20511
Mean Absolute Error :  143988.5965345904
R2 Score :  0.5044410153036336
Random Forest Regressor
Mean Squared Error :  28952122030.700592
Mean Absolute Error :  122585.28783370154
R2 Score :  0.6852640867684735
```



RESULT

The aim of this project is to accurately predict car prices in major Indian metro cities using regression techniques. The prediction model is built on a wide range of vehicle attributes, including the car's company, model, fuel type, colour, kilometres driven, body style, ownership type, dealer location, warranty information, and overall quality score. Through detailed data pre-processing, exploratory analysis, and regression-based modelling, the project identifies the key factors influencing resale value and provides reliable price estimates. This helps both buyers and sellers make informed decisions in the Indian used-car market.

1. Dataset Summary and Preparation

The dataset contains detailed automotive information with features such as:

- Company (Brand)
- Model
- Fuel Type
- Body Style
- Kilometres Driven
- Manufacturing Year (converted to Age)
- Ownership Type
- Warranty Availability
- Dealer State and City
- Quality Score
- Listed Price

These entries were collected from real Indian metro city listings and required multiple pre-processing steps to ensure consistency and model-readiness.

Data Preparation Steps

- Converted price values from string formats (e.g., “Lakhs”) into numerical values
- Dropped columns with excessive missing data (*Transmission Type, CngKit*)
- Removed outliers using the IQR method
- Encoded categorical variables using Label Encoding
- Normalized numerical fields such as age, kilometres driven, and quality score

The final cleaned dataset enabled accurate modelling of price variations across cities, brands, and vehicle segments

2. Model Building

Two primary models were trained to estimate used-car prices based on all available features:

- **Linear Regression**: Served as the simplest model, capturing only linear relationships. While easy to interpret, it failed to properly capture complex interactions between features such as company brand, body style, quality score, and mileage.
- **Random Forest Regressor**: An ensemble approach that aggregated predictions from multiple decision trees. It handled nonlinear dependencies effectively and significantly improved prediction accuracy.

Result:
The Random Forest model achieved an R^2 score of 0.69, outperforming the Linear Regression model by a large margin. It provided more reliable price estimates and successfully modelled complex interactions found in automotive market data.

3. Price Estimation and Model Comparison

Simple Feature Impact Analysis

Price predictions were generated using both models for sample cases across different cities, age groups, and mileage levels. The Random Forest model consistently predicted more realistic values because it considered:

- Combined effects of brand + fuel type
- Mileage + age interactions
- Dealer/state variations
- Quality score differences

Advanced Feature Driven Price Evaluation

To mimic real-world pricing scenarios, multiple subsets of the dataset were tested:

Scenario	Description	Outcome
Single-City Estimate	Pricing within one metro region	Clear variation by dealer/state
All-City Mean Estimate	Aggregated pricing across cities	More stable, averaged results
Attribute-Separated Evaluation	Individual comparisons by fuel type, body style, ownership	Most detailed and realistic

4. Visualization and Route Mapping

Matplotlib and Seaborn were used extensively to visualize:

- Company wise distribution of listings
- Body style and fuel type composition
- Price ranges across cities and dealers
- Age vs. price and mileage vs. price relationships
- Top car companies and models by resale value

- Feature importance comparisons

These visualizations clearly highlighted patterns such as:

- Higher resale values for SUVs and MPVs
- Larger price variation in states like Delhi and Maharashtra
- Strong correlation between mileage, age, quality score, and price

Visual plots helped identify overpriced and under-priced segments across the dataset.

.

5. Key Observations

- **Age and Kilometres Driven** were the strongest predictors of price.
- **Brand reputation (Company)** contributed significantly—premium brands consistently ranked higher.
- **Random Forest** handled variations far better than linear models, due to its ability to model non-linear relationships.
- Cars with **first ownership, warranty coverage, and high-quality scores** commanded higher resale values.
- **City and dealer influence** were visible in the dataset, with certain metro areas showing distinctly higher pricing.
- Feature-based segmentation indicated that SUVs and diesel vehicles tend to hold value better in the Indian used-car market.

CONCLUSION

The car price prediction project has demonstrated the effectiveness of using machine learning models combined with real-world automotive listing data to generate accurate and reliable resale price estimates. By incorporating key vehicle attributes—such as age, kilometres driven, fuel type, body style, company brand, ownership history, and quality score—the project established a data-driven framework capable of understanding price patterns across the Indian used-car market.

The foundation of the study relied on developing and evaluating multiple regression models, specifically Decision Tree and Random Forest Regressors. While simpler models provided a baseline level of estimation, the Random Forest model achieved far superior performance, reaching an R^2 score of approximately **0.69**. This confirmed its ability to capture complex, nonlinear relationships between car features and pricing behaviour—something essential in markets where resale value is influenced by diverse factors including brand perception, mileage, and regional differences.

In addition to model development, the project explored pricing behaviour across different scenarios. City-wise, dealer-wise, and attribute-wise analyses helped identify how geographic location, vehicle condition, and consumer preferences influence pricing consistency. These evaluations revealed that fluctuating factors—such as market demand, brand popularity, and car condition—can significantly impact valuation accuracy, highlighting the importance of models capable of handling heterogeneous data patterns.

Visualizations created with Matplotlib and Seaborn added interpretability to the analysis by illustrating car distributions, price ranges, brand performance, and feature impacts. Heatmaps, bar charts, and distribution plots clarified how certain features—particularly **age**, **kilometres driven**, and **company**—play a major role in determining price. These graphical insights strengthened the analytical depth of the study and showcased practical applications of data science tools within automotive analytics.

The project also emphasized several key elements of data-driven valuation:

- The importance of selecting meaningful vehicle attributes,
- The value of robust machine learning techniques that generalize well,
- The practicality of segment-based price evaluation,
- And the usefulness of visual tools to support transparent pricing decisions.

Overall, this work shows that predictive modelling can substantially enhance traditional car valuation by providing more consistent, transparent, and objective price estimates. The methodology presented here is applicable not only to online resale platforms but also to dealership pricing systems, insurance valuation, and automotive market analysis.

Future work may involve integrating larger multi-year datasets, incorporating image-based car assessments, adding real-time price updates from online listings, and exploring deep learning

models to enhance prediction accuracy. Time-series techniques may also be utilized to capture seasonal and market-driven price fluctuations.

This project ultimately demonstrates how data science and automotive intelligence can work together to solve real-world challenges in pricing and market analysis—providing a modern, adaptive framework for accurate used-car price estimation.

REFERENCES

Breiman, L. (2001). *Random Forests*. Machine Learning, 45(1), 5–32.
(Used for Random Forest Regressor theory)

• **Quinlan, J. R.** (1986). *Induction of Decision Trees*. Machine Learning, 1(1), 81–106.
(Foundational reference for Decision Tree models)

• **James, G., Witten, D., Hastie, T., & Tibshirani, R.** (2013). *An Introduction to Statistical Learning: With Applications in R*. Springer.
(Regression concepts and model evaluation)

• **Pedregosa, F., Varoquaux, G., Gramfort, A., et al.** (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825–2830.
(Library used for model training, label encoding, train-test split)

• **McKinney, W.** (2010). *Data Structures for Statistical Computing in Python*. Proceedings of the 9th Python in Science Conference.
(Pandas library used for data cleaning and pre-processing)

• **Hunter, J. D.** (2007). *Matplotlib: A 2D Graphics Environment*. Computing in Science & Engineering, 9(3), 90–95.
(Used for all visualizations in EDA)

• **Waskom, M.** (2021). *Seaborn: Statistical Data Visualization*. Journal of Open Source Software, 6(60), 3021.
(Used for distribution plots, count plots, violin plots)

• **Friedman, J. H.** (2001). *Greedy Function Approximation: A Gradient Boosting Machine*. Annals of Statistics.
(Relevant to tree-based ML models, even if not used directly)

• **Kuhn, M., & Johnson, K.** (2013). *Applied Predictive Modelling*. Springer.
(Feature importance, pre-processing, outlier handling)

GitHub Link:
https://github.com/Pranshul-Thakur/Car-Price-Predictor-using-Regression