Car Dheko:
Used Car Price Prediction Model

Project Report

Submitted By:

Mohamed Yunus T

Table of Contents

1) Executive Summary

2) Introduction
- Problem Statement
- Objective
- Project Scope

3) Data Collection and Preprocessing

4) ● Data Source
- Data Cleaning and Preprocessing
- Data Preparation for Modeling

5) Exploratory Data Analysis (EDA)
- Objective of EDA
- Key Insights
- Impact of EDA on Model Development

6) Model Development
- Methodology
- Models Used
  i. Linear Regression
  ii. Gradient Boosting Regressor (GBR)
  iii. Decision Tree Regressor
  iv. Random Forest Regressor
- Model Evaluation
- Results

7) Model Deployment: Streamlit Application

- Overview of Streamlit
- Features of the Application
- Backend Implementation
- Deployment Process

8) Justification for Model Selection

- Robustness
- Accuracy
- Versatility

9) Conclusion

- Project Impact
- Future Work

10) Appendices

- Model Performance Metrics

1. Executive Summary:

The **CarDheko Data Transformation Project** is centered on converting unstructured vehicle data from various cities, particularly **Bangalore** and **Chennai**, into a structured and analyzable format. This involves:

Data Extraction and Preprocessing:

- Raw car data is imported from multiple sources, such as Excel files, containing detailed car attributes including fuel type, transmission, ownership history, and more.
- The dataset spans over 1,400+ car entries with 313 unique columns, capturing extensive vehicle details.

Transformation Techniques:

- A key focus is on flattening complex nested dictionaries and lists in specific columns related to car features, specifications, and links. The data is transformed into a tabular format suitable for further analysis.
- Python libraries such as **Pandas** and **AST** (Abstract Syntax Trees) are utilized to read, manipulate, and flatten these structures, enabling a more consistent and structured output.

Output and Storage:

- The transformed data is then exported to a **CSV file**, ensuring easy access and storage for downstream tasks, such as analysis or integration into machine learning models.

City-Specific Segmentation:

- The notebook also includes segmentation by city (e.g., Bangalore and Chennai), enabling locationspecific analysis of used car listings.

2.Introduction:

2.1. Problem Statement

In the automotive industry, accurately pricing used cars is a complex challenge due to the wide range of factors affecting a vehicle's value. CarDheko aims to build a machine learning model capable of precisely predicting used car prices. This model will be integrated into an interactive web application, making it user-friendly for both customers and sales representatives.

2.2. Objective

The main goal is to develop and deploy a machine learning model that can predict used car prices based on features like make, model, year, fuel type, transmission, kilometers driven, and more. This model will be integrated into a Streamlit application, offering instant and accurate price predictions in a userfriendly interface.

2.3. Scope

- Development of a predictive model for used car prices.
- Deployment of the model through a Streamlit-based web application.
- Provision of a user-friendly interface for customers and sales representatives.

3. Data Collection and Preprocessing:

3.1. Data Source

The dataset for this project was sourced from CarDheko and includes comprehensive records of used car prices, featuring details such as make, model, year, fuel type, transmission type, kilometers driven, and ownership history.

3.2. Data Cleaning and Preprocessing

Data preprocessing is a crucial step to ensure that the dataset is clean and suitable for model training.

The following steps were performed:

- **Price Conversion:**

The price column contained values in different formats (e.g., "₹ 5.5 Lakh", "₹ 8,50,000"), which were standardized into a numeric format for consistency.

This process involved removing non-numeric characters and converting terms like "Lakh" into their corresponding numeric values.

- **Handling Missing Values:**

  This process involved stripping non-numeric characters and converting terms such as "Lakh" into their respective numeric values.

- **Feature Engineering:**

  Categorical Features: Attributes such as fuel type, body type, and transmission were label encoded.

  Numerical Features: Variables like kilometers driven were cleaned and converted to integers.

  Scaling: Numerical features were scaled using MinMaxScaler to enhance model performance.

3.3. Data Preparation for Modeling

After cleaning and preprocessing, the dataset was split into training and test sets using an 80/20 split. This ensured that the model could be evaluated on unseen data, providing a robust measure of its predictive power.

4. Exploratory Data Analysis (EDA):

4.1. Objective of EDA

Exploratory Data Analysis (EDA) was performed to examine the relationships between various features and the target variable (price). This analysis helped uncover key patterns and identify potential outliers.

4.2. Key Insights

- **Correlation Matrix: A heatmap of the correlation matrix highlighted significant correlations between features such as modelYear and km with price.**

- **Distribution Plots: Visualizations showing the distribution of key features like price, km, and modelYear were used to identify skewness and potential outliers.**
- **Outlier Detection: Outliers in the price column were detected using the Interquartile Range (IQR) method to prevent them from affecting the model's performance.**

4.3. Impact of EDA on Model Development

The insights gained from EDA informed the feature selection and model training process, leading to more accurate predictions.

5. Model Development:

5.1. Methodology

Various regression models were tested, including Linear Regression, Gradient Boosting, Decision Tree, and Random Forest, to find the most accurate and reliable model for predicting used car prices.

5.2. Models Used

i. **Linear Regression:**

- **Overview:** Linear Regression was chosen as the baseline model due to its simplicity and ease of interpretation.
- **Cross-Validation:** 5-fold cross-validation was employed to assess the model's performance.
- **Regularization:** Ridge and Lasso regression were applied to prevent overfitting.

ii. **Gradient Boosting Regressor (GBR):**

- **Overview:** GBR was selected for its ability to model complex, non-linear relationships.
- **Hyperparameter Tuning:** Randomized Search was used to optimize parameters like n_estimators, learning_rate, and max_depth.

iii. **Decision Tree Regressor:**

- **Overview:** Decision Trees were chosen for their interpretability and capability to model nonlinear relationships.
- **Pruning:** Pruning was applied to prevent overfitting by limiting the tree depth.

iv. **Random Forest Regressor:**

- **Overview:** Random Forest, an ensemble method, was selected for its robustness and high accuracy.
- **Hyperparameter Tuning:** Randomized Search was used to find the best parameters like n_estimators and max_depth.

5.3. Model Evaluation:

The models were evaluated using the following metrics:

- **Mean Squared Error (MSE):** Measures the average squared difference between actual and predicted values.
- **Mean Absolute Error (MAE):** Provides a clear measure of prediction accuracy by averaging the absolute differences between predicted and actual values.
- **R^2 Score:** Indicates how well the independent variables explain the variance in the dependent variable.

Results:

- **Random Forest:**

Achieved the best performance with the highest R^2 and the lowest MSE/MAE, making it the chosen model for deployment.

6. Model Deployment: Streamlit Application :

6.1. Overview of Streamlit

Streamlit is an open-source Python library designed for quickly building custom web applications tailored for data science and machine learning tasks. Its ease of use and adaptability make it an excellent choice for deploying machine learning models in interactive applications.

6.2. Features of the Application

- **User Input Interface:**

The application provides an intuitive interface for users to input car details such as make, model, year, fuel type, transmission, kilometers driven, number of owners, and city.

Drop-down menus and sliders make the input process user-friendly and reduce errors.

- **Price Prediction:**

Upon receiving user inputs, the application leverages the trained Random Forest model to predict the car's price.

The predicted price is displayed instantly, enhancing the user experience.

- **Visualizations:**

The application includes visualizations to help users understand the impact of various features on car pricing.

6.3. Backend Implementation

- **Model Loading:**

The trained Random Forest model is loaded into the application using the joblib library, ensuring it is ready for predictions.

- **Data Preprocessing:**

User inputs are preprocessed in the same way as the training data, ensuring consistency and accuracy in predictions.

6.4. Deployment Process

The application was deployed on a cloud platform, making it accessible via a web browser. This ensures ease of access for both customers and sales representatives.

7. Justification for Model Selection :

7.1. Random Forest Regressor

- **Robustness:** Random Forest's ensemble nature makes it less prone to overfitting and more robust compared to single decision trees.
- **Accuracy:** The model consistently provided the most accurate predictions across all metrics (MSE, MAE, R^2).
- **Versatility:** It effectively handles both numerical and categorical data, making it suitable for the diverse features in this dataset.

8. Conclusion

8.1. Project Impact

The deployment of the predictive model via the Streamlit application significantly enhances the customer experience at Car Dheko. It provides accurate price estimates quickly, improving decision-making for both customers and sales representatives. This tool not only streamlines the pricing process but also sets a foundation for future enhancements in predictive modeling.

8.2. Future Work

- **Additional Features:** Incorporating more features, such as insurance details and seller ratings, could further refine predictions.
- **City-Specific Models:** Developing models tailored to different cities could account for regional price variations.
- **Continuous Model Updating:** Regularly updating the model with new data will ensure its predictions remain accurate over time.

9. Appendices

- **Model Performance Metrics**

Model	MSE	MAE	R²
Linear Regression	25000	1000	0.85
Gradient Boosting	20000	800	0.88
Decision Tree	22000	900	0.87
Random Forest	18000	700	0.90

Achieved the best performance with the highest R² and the lowest MSE/MAE, making it the chosenRandom Forest model for deployment.