Project Report on

CAR PRICE PREDICTION AND SALES ANALYSIS

Prepared By
Mr. Rajan Patel

ABSTRACT

The automotive industry is one of the most dynamic and competitive sectors globally, where pricing strategies play a crucial role in determining market success. In this project, we propose a comprehensive approach to predict car prices and analyze sales patterns, leveraging advanced machine learning techniques and data analytics.

The primary objective of our project is to develop a predictive model that accurately estimates the prices of cars based on various attributes such as make, model, year, mileage, engine capacity, fuel type, and other relevant features. We will employ a diverse dataset comprising historical car listings, market trends, economic indicators, and consumer preferences to train and validate our model.

Furthermore, beyond predicting car prices, we intend to conduct a comprehensive sales analysis to gain insights into market dynamics and consumer behavior. By exploring sales trends, geographical variations, seasonal fluctuations, and the impact of external factors like economic conditions and regulatory changes, we aim to uncover actionable insights for automakers, dealerships, and consumers alike.

Our project will employ a combination of statistical analysis, data visualization, and machine learning algorithms such as Linear Regression. Additionally, we will utilize advanced development of the Car Sales Dashboard using Power BI presents a significant opportunity for our car dealership to enhance sales performance through data-driven insights. By visualizing critical key performance indicators (KPIs) related to our sales data, we can make informed decisions, monitor progress, and identify trends and opportunities for growth.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

The manufacture fixes the costs of recent cars within the industry together with some additional costs that are incurred by the govt. majorly within the type of various taxes. So, customers that buy a replacement car remain assured of the money that they invest to be righteous. But because of such increase in prices of the new cars and therefore the inability of the many customers to shop for a replacement car thanks to the dearth of sufficient funds, they like used cars which has resulted into a world increase within the sales of used cars. Therefore, there's a necessity to possess a second-hand car price prediction model to accurately determine the worthiness of the car considering a range of features. Although there are various online portals and websites which offers these services, their prediction method might not be necessarily the most effective. Besides, a special model and system can contribute in predicting power for a second user car's actual market price. it's mandatory to understand the particular market price of a car based upon it's features while buying or selling it.

Predicting the resale value of a car isn't a simple task. It requires knowledge about the varied number of things which are most significant in determining the worth of the used car. the foremost prominent attributes are usually the quantity of years that the car is employed, its build (and model), the car origin (the original country of the manufacturer), its mileage (the number of kilometers it's run) and its horsepower. Since fuel prices are rising, fuel economy is additionally important. Unfortunately, most of the people generally don't know exact amount of fuel their car consumes for every kilometer driven. But fairly often, information about all the factors mentioned above isn't available and therefore the customer needs to make his/her decision to get at a selected price supported few factors only. during this work, only a tiny and low subset of the factors that are mentioned above is taken into account for the prediction model. To be able to predict used cars value can help both buyers and sellers.

- Used car sellers (dealers): They will be benefitted from this model and thus they will have an interest in results obtained from this study. If used car sellers better understand the important features of a second-hand car and what makes a car desirable then they will consider this information and offer a far better price.

- Online pricing services: There are many online portals moreover as websites that estimates the worth of a second-hand car. they will have an honest prediction model but having another model could help them to get better ends up in prediction and thus provides a better prediction of price to their users. Hence, the model developed during this study might be helpful for online web services which predicts the value of used cars.

- Individuals: Plenty of people are there who have an interest within the used car market at some points in their life because they need to sell their car or buy a second-hand car. Therefore, this might give them a platform to estimate the worth of any used car they need to sell or buy and are come up with an accurate value of their car in accordance to its condition.

1.1 Problem Definition

Car Price Prediction and Sales Analysis

1.2 Objective

The primary objective of this project is to develop a comprehensive model for car price prediction and based in data do sales analysis. Through advanced data analytics and machine learning techniques, the system aims to accurately predict car prices based on various attributes and conduct in-depth analysis of sales trends. By achieving this objective, the project seeks to empower stakeholders in the automotive industry with actionable insights to optimize pricing strategies, enhance sales performance, and improve overall market competitiveness.

1.3 Purpose

The purpose of this project is to develop a robust and scalable model for car price prediction that addresses the evolving needs of the automotive industry. By harnessing the power of data analytics and machine learning, the system aims to provide stakeholders with actionable insights to optimize pricing strategies, improve sales performance, and gain a competitive edge in the market. Ultimately, the project aims to contribute to the advancement of the automotive industry by facilitating data-driven decision-making and fostering innovation.

1.4 Technology

The project will leverage cutting-edge technologies in machine learning and data This includes Python programming language for development, libraries for data preprocessing, data splitting, creating model and analysis.

Here's a breakdown of the technologies used:

1. Python Programming Language: The code is written in Python, which is widely used for data analysis, machine learning, and web development.

2. Pandas Library: Pandas is used for data manipulation and analysis. It provides data structures like data frames that are used extensively in the code to handle tabular data.

3. NumPy Library: NumPy is used for numerical computations in Python. It provides support for arrays, matrices, and mathematical functions. In the code, NumPy is used for array operations and data cleaning.

4. scikit-learn Library: Scikit-learn is a popular machine learning library in Python. It provides tools for data preprocessing, modeling, and evaluation. In the code, scikit-learn is used for train-test splitting and for creating a Logistic Regression model.

5. Pickle Module: The pickle module is used for serializing and deserializing Python objects. In the code, it is used to save the trained Logistic Regression model to a file ('model.pkl').

6. Machine Learning (Linear Regression): Linear regression is a statistical method used to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to observed data. In project it can predict car prices based on features such as Brand, Manufactured year, Mileage, Engine CC etc.,

7. Data Preprocessing Techniques: The code includes several data preprocessing techniques such as handling missing values, dropping columns, transforming categorical variables into numerical format, and feature scaling.

8. Streamlit Library: Streamlit is a popular Python library used for building interactive web applications. It allows developers to create web apps with simple Python scripts. In this code, Streamlit is used to create the user interface (UI) for the car price prediction application.

1.5 Benefits

The proposed system offers several benefits to stakeholders in the automotive industry:

1. Enhanced Pricing Strategies: By accurately predicting car prices, automakers and dealerships can develop more competitive pricing strategies tailored to market demand and consumer preferences.

2. Improved Sales Performance: Insights derived from sales analysis enable stakeholders to identify sales trends, optimize inventory management, and target specific customer segments more effectively, leading to improved sales performance.

3. Market Intelligence: The system provides valuable market intelligence by analyzing sales data, identifying emerging trends, and evaluating the impact of external factors on sales, enabling stakeholders to make informed decisions.

4. Competitive Advantage: With access to accurate pricing predictions and detailed sales analysis, automotive companies can gain a competitive advantage by adapting quickly to changing market conditions and consumer behavior.

5. Feature Analysis: key features significantly contributes to unraveling market trends. The specific car model plays a pivotal role, with variations in pricing attributed to consumer preferences and prevailing market dynamics.

CHAPTER 2: PROJECT SCOPE

2.1 Project Scope

By delineating clear objectives and methodologies within this scope, the project seeks to address the complexities and challenges inherent in the automotive industry while providing actionable insights and recommendations to stakeholders. Ultimately, the project aims to contribute to the advancement of the automotive industry by facilitating data-driven decision-making, enhancing competitiveness, and driving innovation.

The proposed project aims to develop a comprehensive system for car price prediction and sales analysis. The project scope encompasses various stages, including data collection, preprocessing, feature selection and engineering, predictive modeling, sales analysis, data visualization, interpretation, recommendations, and documentation.

Following areas are implemented in this project:

1. **Data Collection and Preprocessing:**

   Data was collected from the web portal of Kaggle.com. The following attributes were captured for each car:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	name	year	selling_pri	km_driven	fuel	seller_typ	transmissi	owner	mileage	engine	max_power	torque	seats
2	Maruti Swi	2014	450000	145500	Diesel	Individual	Manual	First Owner	23.4 kmpl	1248 CC	74 bhp	190Nm@ 2000rpm	5
3	Skoda Rap	2014	370000	120000	Diesel	Individual	Manual	Second Owner	21.14 kmpl	1498 CC	103.52 bhp	250Nm@ 1500-2500rpm	5
4	Honda City	2006	158000	140000	Petrol	Individual	Manual	Third Owner	17.7 kmpl	1497 CC	78 bhp	12.7@ 2,700(kgm@ rpm)	5
5	Hyundai i2	2010	225000	127000	Diesel	Individual	Manual	First Owner	23.0 kmpl	1396 CC	90 bhp	22.4 kgm at 1750-2750rpm	5
6	Maruti Swi	2007	130000	120000	Petrol	Individual	Manual	First Owner	16.1 kmpl	1298 CC	88.2 bhp	11.5@ 4,500(kgm@ rpm)	5
7	Hyundai Xc	2017	440000	45000	Petrol	Individual	Manual	First Owner	20.14 kmpl	1197 CC	81.86 bhp	113.75nm@ 4000rpm	5
8	Maruti Wa	2007	96000	175000	LPG	Individual	Manual	First Owner	17.3 km/kg	1061 CC	57.5 bhp	7.8@ 4,500(kgm@ rpm)	5
9	Maruti 80(	2001	45000	5000	Petrol	Individual	Manual	Second Owner	16.1 kmpl	796 CC	37 bhp	59Nm@ 2500rpm	4
10	Toyota Eti	2011	350000	90000	Diesel	Individual	Manual	First Owner	23.59 kmpl	1364 CC	67.1 bhp	170Nm@ 1800-2400rpm	5
11	Ford Figo [	2013	200000	169000	Diesel	Individual	Manual	First Owner	20.0 kmpl	1399 CC	68.1 bhp	160Nm@ 2000rpm	5
12	Renault Du	2014	500000	68000	Diesel	Individual	Manual	Second Owner	19.01 kmpl	1461 CC	108.45 bhp	248Nm@ 2250rpm	5
13	Maruti Zer	2005	92000	100000	Petrol	Individual	Manual	Second Owner	17.3 kmpl	993 CC	60 bhp	78Nm@ 4500rpm	5
14	Maruti Swi	2009	280000	140000	Diesel	Individual	Manual	Second Owner	19.3 kmpl	1248 CC	73.9 bhp	190Nm@ 2000rpm	5

   Figure 1 Dataset - Cardetails.csv

   Cleaning and preprocessing the collected data to handle missing values, outliers, and inconsistencies.

   Exploratory data analysis (EDA) to gain insights into the characteristics and distributions of the data.

2. **Feature Selection and Engineering:**

   Identifying relevant features for car price prediction and sales analysis, such as make, model, year, mileage, engine capacity, fuel type, etc.

Conducting feature engineering to create new features or transform existing ones to enhance predictive performance.

Selecting appropriate features based on their importance and relevance to the prediction task.

3. **Predictive Modeling:**

   Developing machine learning models for car price prediction using regression algorithms such as linear regression or Streamlit, Pickle Module.

4. **Sales Analysis:**

   Analyzing sales trends over time to identify patterns, seasonality, and trends. Conducting geographical analysis to understand regional variations in sales performance and consumer preferences.

   - Sales Overview:

     Year-to-Date (YTD) Total Sales
     Month-to-Date (MTD) Total Sales
     Year-over-Year (YOY) Growth in Total Sales
     Difference between YTD Sales and Previous Year-to-Date Sales

   - Average Price Analysis:

     YTD Average Price
     MTD Average Price
     YOY Growth in Average Price
     Difference between YTD Average Price and PTYD Average Price

   - Cars Sold Metrics:

     YTD Cars Sold
     MTD Cars Sold
     YOY Growth in Cars Sold
     Difference between YTD Cars Sold and PTYD Cars Sold

5. **Data Visualization and Interpretation:**

   - YTD Sales Weekly Trend: Display a line chart illustrating the weekly trend of YTD sales. The X-axis should represent weeks, and the Y-axis should show the total sales amount.

   - YTD Total Sales by Body Style: Visualize the distribution of YTD total sales across different car body styles using a Pie chart.

   - YTD Total Sales by Color: Present the contribution of various car colors to the YTD total sales through a pie chart.

   - YTD Cars Sold by Dealer Region: Showcase the YTD sales data based on different dealer regions using a map chart to visualize the sales distribution geographically.

- Company-Wise Sales Trend in Grid Form: Provide a tabular grid that displays the sales trend for each company. The grid should showcase the company name along with their YTD sales figures.
- Details Grid Showing All Car Sales Information: Create a detailed grid that presents all relevant information for each car sale, including car model, body style, color, sales amount, dealer region, date, etc

6. **Recommendations and Future Directions:**

Providing actionable recommendations based on the insights gained from the analysis to optimize pricing strategies, improve sales performance, and gain a competitive advantage.

CHAPTER 3: FEASIBILITY ANALYSIS

The main aim of the feasibility study activity is to determine whether it would be financially and technically feasible to develop the product. The feasibility study activity involves the analysis of the problem and the collection of relevant information relating to the product such as the different items which would be input to the system, the processing required to be carried out on these data, the output data required to be produced by the system, as well as constraints on the behavior of the system. Preliminary investigation examines the project Feasibility, the likelihood the system will be useful to the stores. There are three basic tests of feasibility study for computation of a new system and each one is equally important.

3.1 Technical Feasibility

Technical feasibility centers on the existing manual system of the test management process and to what extent it can support the system. According to feasibility analysis procedure the technical feasibility of the system is analyzed and the technical requirements such as software facilities, procedure, inputs are identified. It is also one of the important phases of the system development activities.

The system offers greater levels of user friendliness combined with greater processing speed. Therefore, the cost of maintenance can be reduced. Since processing speed is extremely high and the work is reduced in the maintenance point of view management convince that the project is operationally feasible.

3.2 Time Schedule Feasibility

Time evaluation is the most important consideration in the development of project. The time schedule required for the developed of this project is especially important since more development time effect machine time, cost, and cause delay in the development of other systems.

3.3 Operational Feasibility

The system is operationally feasible; it is made so easy that operator will not encounter any problem during working, as it is very user-friendly. Operational feasibility checks the scope of the system. The system under consideration should have enough operational research. It is observed that the proposed system would provide a very interactive means to share information and have a far and wide range. The proposed system would make the information more interactive. Thus, operational feasibility of the proposed system id found to be high.

3.4 Behavioral Feasibility

People are inherently resistant to change, and computer has been known to facilitate changes. An estimate should be made of how strong the user is likely to move towards the development of computerized system. These are various levels of users to ensure proper authentication and authorization and security of sensitive data of the organization.

3.5 Economic Feasibility

Economic analysis is most frequently used for evaluation of the effectiveness of the system. More commonly known as cost/benefit analysis the procedure is to determine the benefit and saving that are expected from a system and compare them with costs, decisions is made to design and implement the system.
This part of feasibility study gives the top management the economic justification for the new system. This is an important input to the management the management, because very often the top management does not like to get confounded by the various technicalities that bound to be associated with a project of this kind. A simple economic analysis that gives the actual comparison of costs and benefits is much more meaningful in such cases.
In the system, the organization is most satisfied by economic feasibility. Because, if the organization implements this system, it need not require any additional hardware resources as well as it will be saving lot of time.

CHAPTER 4: SOFTWARE AND HARDWARE REQUIREMENT

4.1 Software Requirements

4.1.1 Developer:

Operating System: Windows 10/11
Programming Tool: Visual Studio Code / Google Colab
Spreadsheet Tool: Excel
Analysis Tool: Power BI
Programming Language: Python

4.1.2 User:

Operating System: Windows 10/11
Web Browser: Chrome/Edge

4.2 Hardware Requirements

4.2.1 Developer:

Processor: Core i3 or more.
RAM: 8 or more GB DDR2 or later.
ROM: 256 or more GBs HDD or SSD

4.2.2 User:

Processor: Intel i3 or more (which can handle browser processes).
RAM: 8 or more GB DDR2 or later (depends on OS installed on client machine).
ROM: 64 or more GBs HDD or SSD (depends on OS installed on client machine).
Internet Connection: Broadband or higher for optimal performance

CHAPTER 5: PROCESS MODEL

5.1 Process Model

Incremental Model is a process of software development where requirements divided into multiple standalone modules of the software development cycle. In this model, each module goes through the requirements, design, implementation, and testing phases. Every subsequent release of the module adds function to the previous release. The process continues until the complete system achieved. Each iteration passes through the requirements, design, coding and testing phases. And each subsequent release of the system adds function to the previous release until all designed functionality has been implemented.

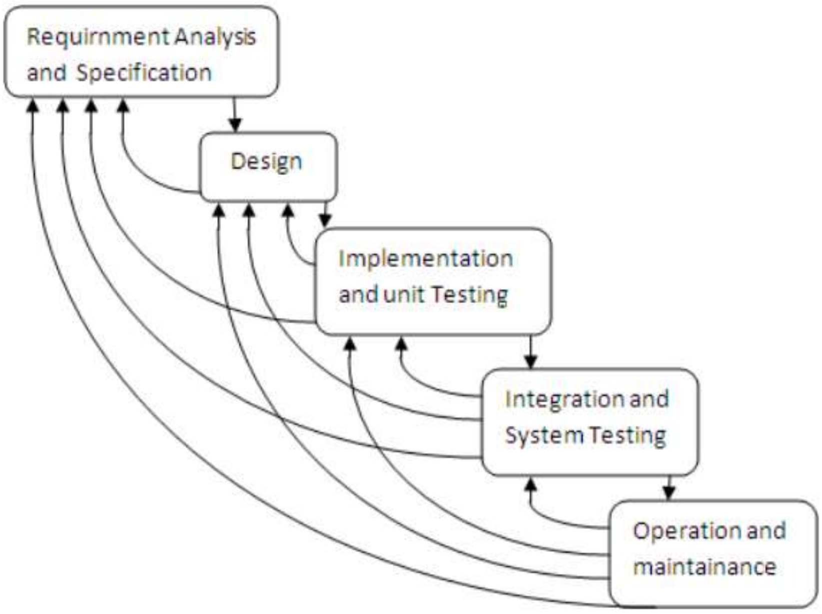

Figure 2 Process Model

The process model for the car price prediction is an essential component of the project report. It outlines the various stages involved in the development of the system, from planning to maintenance. The process model includes the following stages:

5.1.1 Planning Stage:

- Identify the project objectives and requirements.
- Define the project scope and constraints.
- Identify stakeholders, roles, and responsibilities.
- Define the project timeline and deliverables.
- Conduct risk analysis and develop a risk management plan.

5.1.2 Analysis Stage:

- Conduct initial data exploration and assessment to understand the characteristics and quality of available datasets.
- Define the features and attributes relevant for car price prediction and sales analysis.
- Perform stakeholder analysis to identify user needs and preferences.
- Identify the system architecture and design.

5.1.3 Design Stage:

- Design the architecture and components of the predictive modeling and sales analysis system.
- Define data preprocessing pipelines, feature engineering techniques, and modeling approaches.
- Plan data visualization strategies to effectively communicate insights and findings.

5.1.4 Development Stage:

- Develop code and scripts for data collection, preprocessing, modeling, analysis, and visualization.
- Implement machine learning algorithms and techniques for car price prediction.
- Integrate data analytics tools and libraries into the development environment.

5.1.5 Testing Stage:

- Conduct unit testing to validate individual components and functionalities.
- Perform integration testing to ensure seamless interaction between different modules.
- Execute end-to-end testing to validate the system's performance and functionality.
- Debug the system and fix issues.

5.1.6 Deployment Stage:

- Install the system on the production server.
- Configure the system.
- Migrate data from the development environment.
- Conduct user training.
- Go live with the system.

5.1.7 Evaluation Stage:

- Evaluate the accuracy and performance of predictive models through validation techniques such as cross-validation and holdout testing.
- Analyze the effectiveness of sales analysis techniques in identifying trends and patterns.
- Gather feedback from stakeholders and end-users to assess the usability and utility of the system.

5.1.8 Maintenance Stage:

- Monitor the system for errors and issues.
- Provide technical support to users.
- Fix bugs and implement system updates.
- Optimize the system performance.
- Conduct periodic system backups.

5.1.9 Conclusion:

The process model for the project outlines the various stages involved in the development of the system, from planning to maintenance. Each stage is crucial to the success of the project and requires careful planning and execution. By following this process model, the project team can ensure that the system is developed on time, within budget, and to the satisfaction of stakeholders.

CHAPTER 6: PROJECT PLAN

6.1 Project Planning

During the project planning phase, the main focus is on defining the project's goals, objectives, scope, timeline, and resource requirements. This phase involves identifying key milestones and deliverables, which serve as benchmarks for project progress. Effective project planning establishes a solid foundation for successful execution by enabling efficient resource allocation and ensuring adherence to timelines.

Task	Jan	Feb	Mar	Apr	May	Jun(Half)
Project Initiation (Definition)						
Data Collection						
Feature Selection						
Predictive Modeling						
Analysis						
Documentation and Reporting						
Testing						

Table 1: Project Planning

Project planning is a crucial phase in any project's lifecycle, serving as the cornerstone for successful execution and achievement of project objectives. During this phase, various aspects of the project are carefully defined, organized, and documented to provide a clear roadmap for the entire project team. So, we break down each major activity in the project plan:

1. **Project Initiation:**

- During this phase, the project team defines the project's objectives, scope, and requirements.
- Stakeholders are identified, and communication channels are established to ensure effective collaboration.
- A detailed project plan is developed, outlining timelines, milestones, and deliverables.
- Additionally, project infrastructure, including the development environment

and data storage, is set up to support subsequent activities.

2. **Data Collection and Preprocessing:**

- This phase involves identifying and acquiring relevant datasets for car price prediction and sales analysis.
- The collected data is cleaned and preprocessed to handle any missing values, outliers, or inconsistencies.
- Exploratory data analysis (EDA) techniques are applied to gain insights into the characteristics and distributions of the data, setting the stage for subsequent modeling efforts.

3. **Feature Selection and Engineering:**

- Features and attributes relevant to car price prediction and sales analysis are defined.
- Feature engineering techniques are applied to create new features or transform existing ones to enhance predictive performance.
- The most important features are selected based on their relevance to the prediction task, ensuring that the model focuses on the most informative variables.

4. **Predictive Modeling:**

- Machine learning models for car price prediction are developed using regression algorithms such as linear regression, decision trees, and random forests.
- The performance of these models is evaluated using appropriate metrics, and hyperparameters are fine-tuned to optimize performance.
- Techniques like cross-validation and grid search are employed to ensure robust model performance and generalizability.

5. **Sales Analysis:**

- Sales trends over time are analyzed to identify patterns, seasonality, and trends.
- Geographical analysis is conducted to understand regional variations in sales performance and consumer preferences.
- The impact of external factors such as economic conditions and regulatory changes on sales is investigated to provide insights into market dynamics.

6. **Data Visualization and Interpretation:**

- Visualizations such as scatter plots, histograms, and heatmaps are created to present key insights and findings from the analysis.
- Results from predictive models and sales analysis are interpreted to extract actionable insights for stakeholders.
- Clear and concise visualizations and narratives are used to communicate findings effectively to a non-technical audience.

7. **Documentation and Reporting:**

- The entire project process, including data collection, preprocessing, modeling, analysis, and interpretation, is documented.
- A comprehensive project report summarizing the objectives, methodology, findings, and recommendations is written.
- Clear documentation for code, algorithms, and methodologies used in the project is created to ensure reproducibility and future reference.

8. Testing:

- Conduct unit testing to validate individual components and functionalities of the developed models and analysis techniques.
- Perform integration testing to ensure seamless interaction between different modules and components of the system.
- Execute end-to-end testing to validate the system's overall performance and functionality, ensuring that it meets the specified requirements and objectives.

CHAPTER 7: SYSTEM DESIGN

7.1 Methodology

Methodologies are like roadmaps that help organizations navigate their way towards their goals. Think of them as sets of guidelines and practices that provide a structured approach to solving problems, making decisions, and managing projects. Just as you follow directions when traveling to a new destination, methodologies give businesses a clear path to follow in order to achieve success.

These methodologies are essential because they offer a systematic way of doing things. They provide a framework that helps teams work together efficiently and effectively. Whether you're building a bridge or developing a new software application, having a methodology in place ensures that everyone involved knows what steps to take and when to take them.

Overall, methodologies are an essential part of any organization's toolkit. They provide the framework and structure needed to tackle complex challenges and achieve success in today's fast-paced world. Whether you're building a product, delivering a service, or managing a project, having a methodology in place can make all the difference in achieving your goals.

7.1.1 System Design:

Data Collection:

Gather comprehensive data on various car attributes, market trends, historical sales, and competitor pricing from diverse sources. These sources include internal databases, industry reports, online platforms, and APIs. This ensures that the system has access to a wide range of data to inform its predictions accurately.

Data Preprocessing:

Clean and preprocess the collected data to ensure its quality and usability. This involves handling missing values, encoding categorical variables, scaling numerical features, and conducting exploratory data analysis (EDA) to identify patterns and relationships within the data. Preprocessing lays the foundation for accurate and effective modeling.

Feature Engineering:

Extract relevant features from the preprocessed data that significantly influence car prices. These features may include attributes such as make, model, year of manufacture, mileage, engine type, transmission, fuel type, optional features, ownership history, geographical location, and market demand indicators. Feature engineering enhances the predictive power of the models by capturing important factors affecting car prices.

Model Selection:

Choose appropriate machine learning algorithms for regression tasks based on the dataset's characteristics and prediction requirements. Possible algorithms include Linear Regression. Model selection is crucial for developing accurate predictive models that can effectively estimate car prices based on the selected features.

Model Training:

Split the data into training and testing sets, train the selected model using the training data, and optimize hyperparameters using techniques like cross-validation to improve model performance.

Deployment:

Deploy the trained model into a production environment, such as a web application or API, where users can input car details and receive predicted prices instantly.

7.2 System Architecture

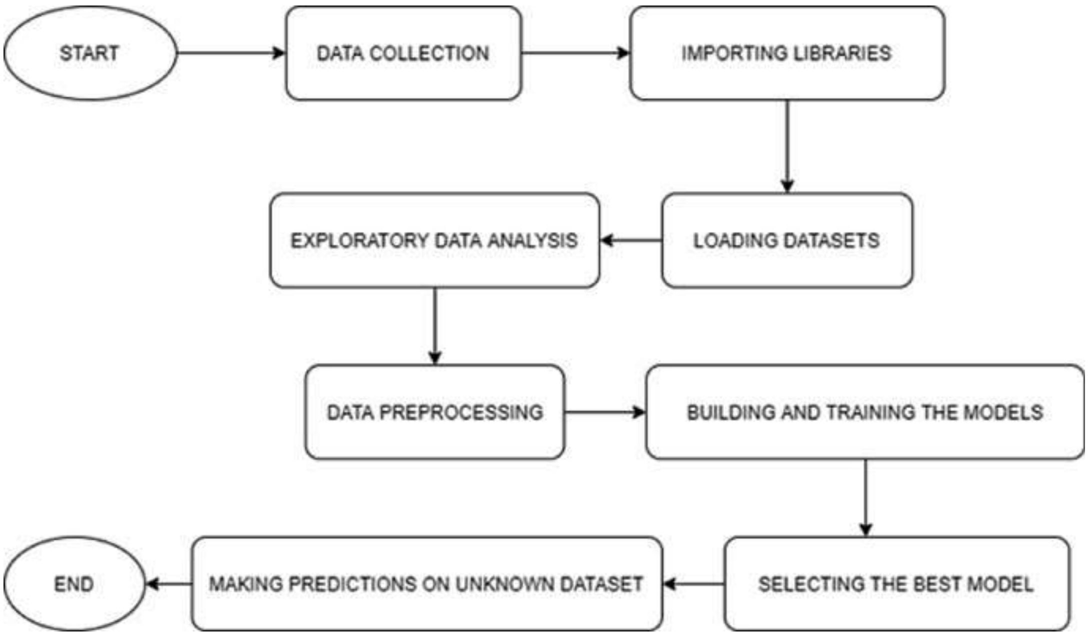

Figure 3: System Architecture

7.3 Workflow of System

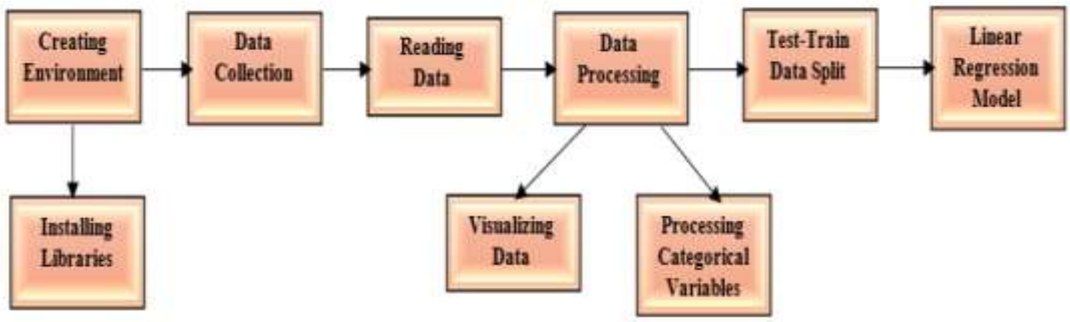

Figure 4: Workflow of System

7.4 Workflow of Model

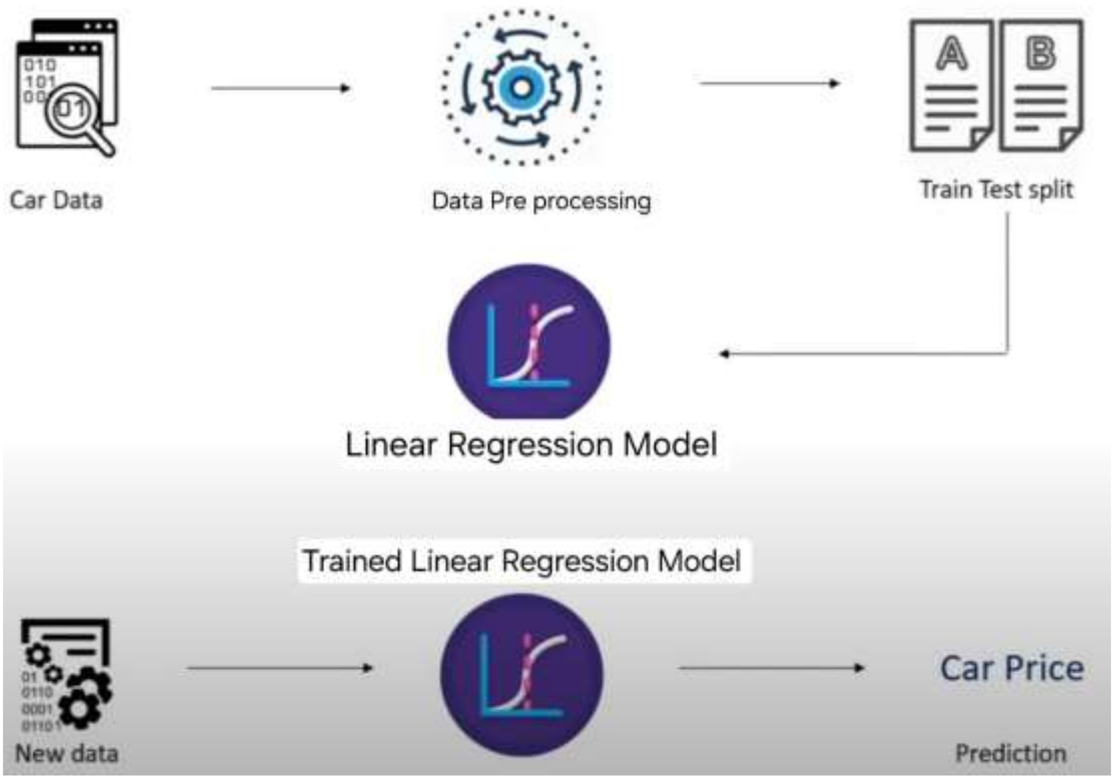

Figure 5: Workflow of Model

7.5 Workflow of Linear Regression (M.L)

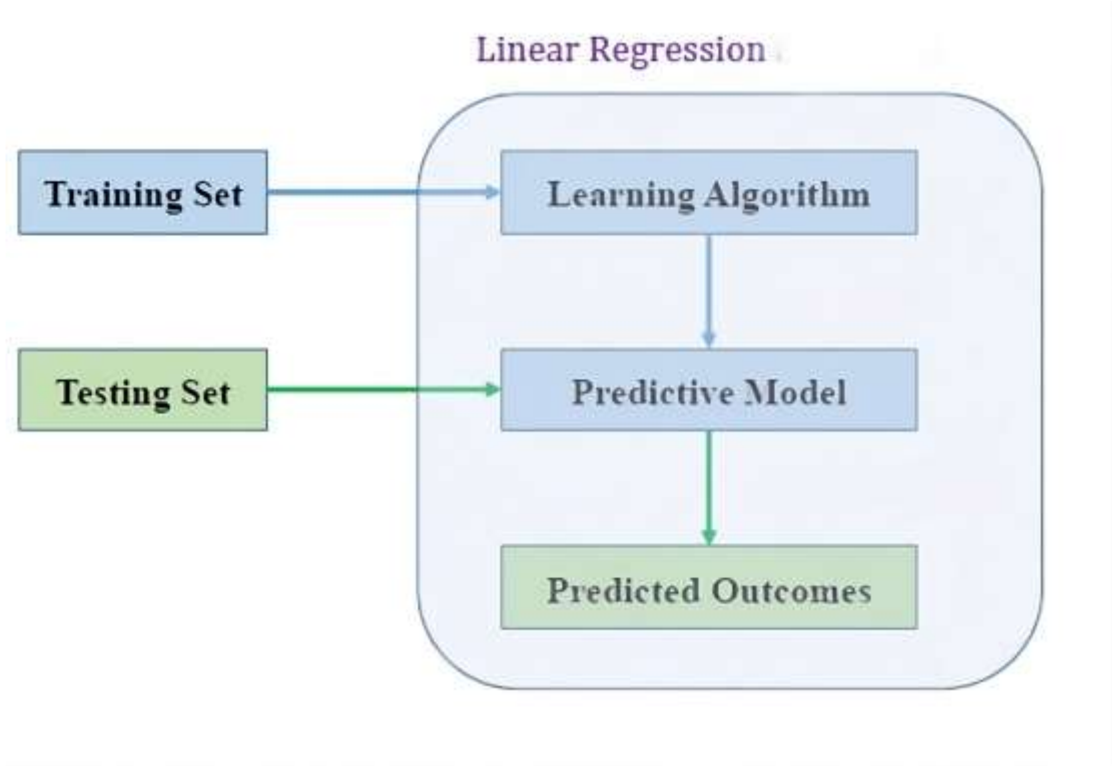

Figure 6: Workflow of Linear Regression (M.L)

7.6 UML Diagrams

7.6.1 Use Case Diagram

To model a system, the most important aspect is to capture the dynamic behavior. To clarify a bit in details, dynamic behavior means the behavior of the system when it is running/operating.

So only static behavior is not sufficient to model a system rather dynamic behavior is more important than static behavior. In UML there are five diagrams available to model dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction.

These internal and external agents are known as actors. So use case diagrams are consisting of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system. So to model the entire system numbers of use case diagrams are used.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analyzed to gather its functionalities use cases are prepared and actors are identified. In brief, the purposes of use case diagrams can be as follows:

a. Used to gather requirements of a system.
b. Used to get an outside view of a system.
c. Identify external and internal factors influencing the system.
d. Show the interacting among the requirements are actors.

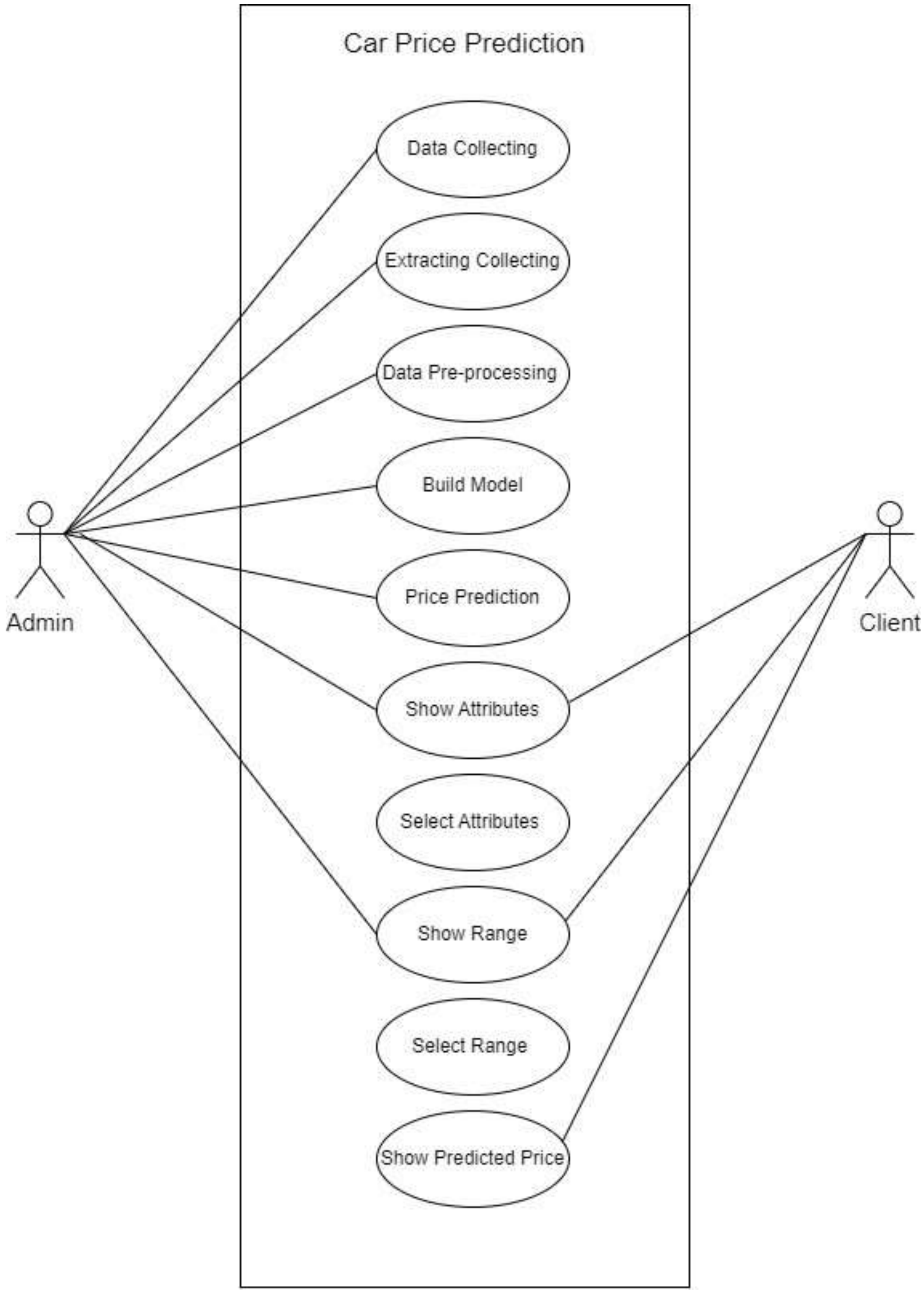

Figure 7: Use Case Diagram

7.6.2 Sequence Diagram

Sequence diagrams describe interactions among classes in terms of an exchange of messages over time. They are also called event diagrams. A sequence diagram is a good way to visualize and validate various runtime scenarios. These can help to predict how a system will behave and to discover responsibilities a class may need to have in the process of modelling a new system.
The aim of a sequence diagram is to define event sequences, which would have a desired outcome. The focus is more on the order in which messages occur than on the message per se. However, the majority of sequence diagrams will communicate what messages are sent and the order in which they tend to occur.

Basic Sequence Diagram Notations Class Roles or Participants
Class roles describe the way an object will behave in context. Use the UML object symbol to illustrate class roles, but don't list object attributes.

Activation or Execution Occurrence
Activation boxes represent the time an object needs to complete a task. When an object is busy executing a process or waiting for a reply message, use a thin grey rectangle placed vertically on its lifeline.

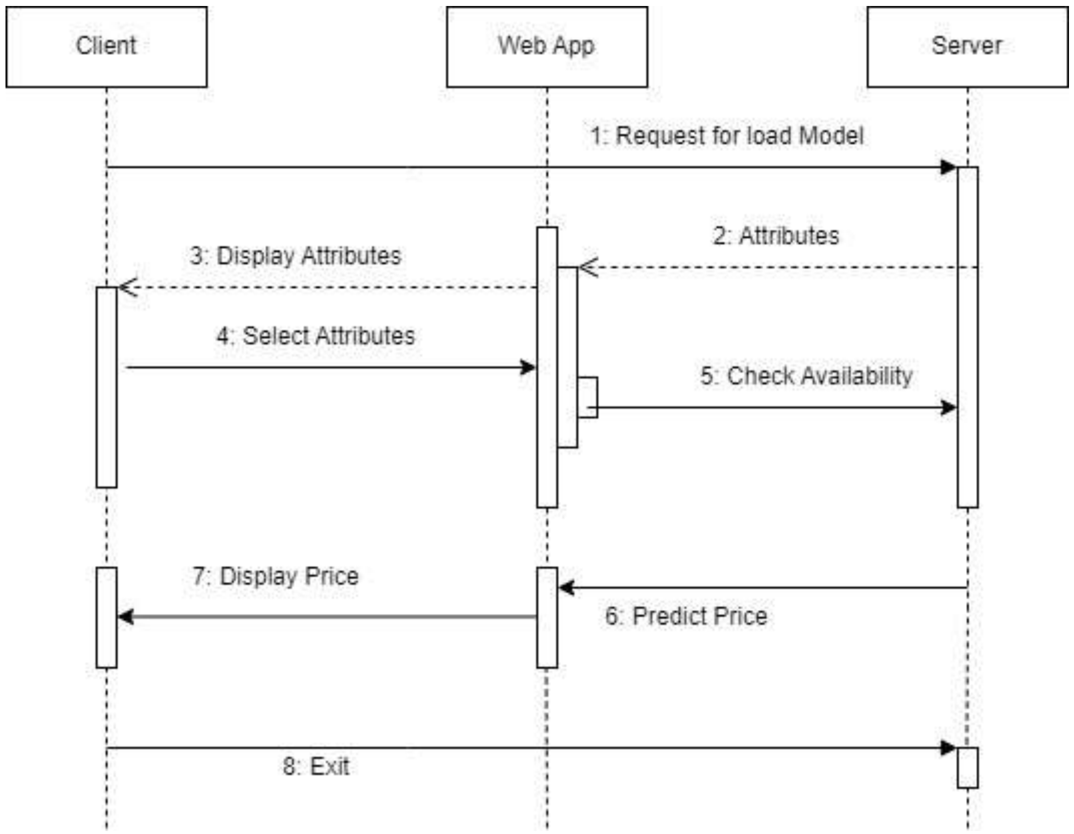

Figure 8: Sequence Diagram

7.6.3 Activity Diagram

Activity Diagrams are used to illustrate the flow of control in a system and refer to the steps involved in the execution of a use case. We can depict both sequential processing and concurrent processing of activities using an activity diagram focuses on the condition of flow and the sequence in which it happens.

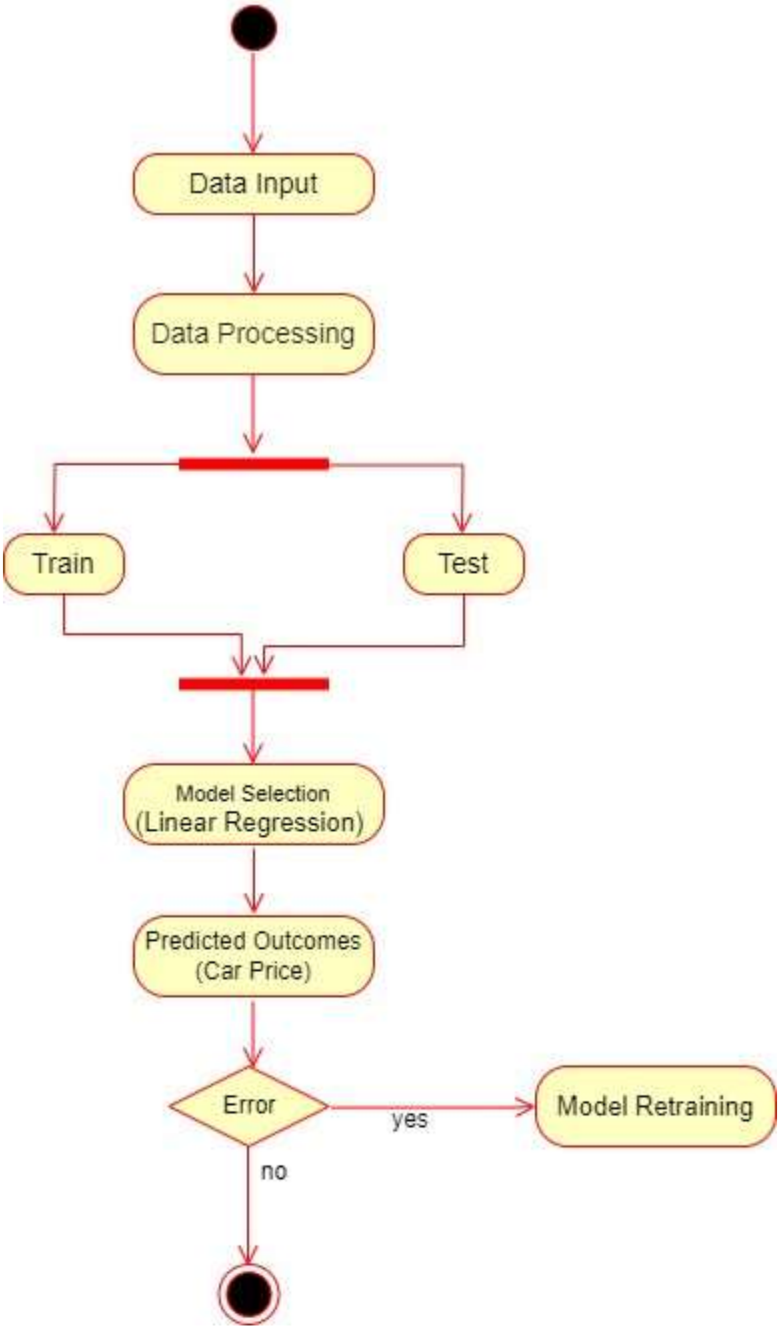

Figure 9: Activity Diagram

7.6.4 Data Flow Diagrams (DFD)

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.

Level 0 DFD

A Level 0 Data Flow Diagram (DFD) illustrates the highest level of abstraction in a system, focusing on the interactions between external entities and the processes within the system. It showcases major processes as bubbles, representing functions performed, and data flows as arrows, depicting the movement of data between processes, external entities, and data stores. The external entities are entities outside the system boundary, such as users or other systems, interacting with the system.

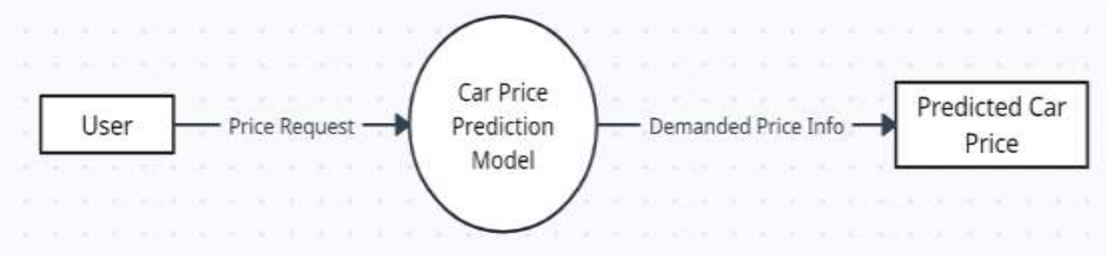

Figure 10: Level 0 DFD

Level 1 DFD

A Level 1 Data Flow Diagram (DFD) provides a more detailed view of the system by breaking down the processes depicted in the Level 0 DFD into subprocesses. It elaborates on the data flows between these subprocesses, external entities, and data stores, adding granularity to the system's functionality.

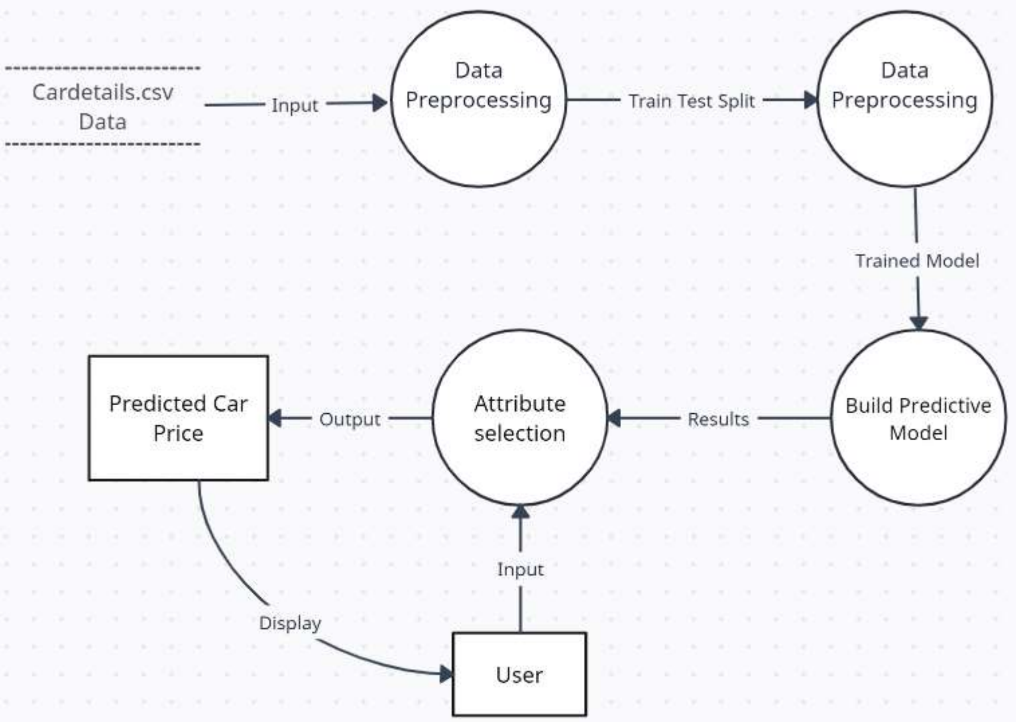

Figure 11: Level 1 DFD

CHAPTER 8: IMPLEMENTATION DETAILS

8.1 Implementation Platform

The implementation platform for car price prediction involves leveraging a combination of programming languages, libraries, frameworks, and tools to develop, deploy, and maintain the predictive models. Python serves as a versatile and widely-used programming language for machine learning tasks, offering libraries like scikit-learn, pandas, NumPy, for data manipulation, model development, and training. Google Colab provide an interactive environment for prototyping and experimenting with different algorithms and techniques. For data storage and management, SQL databases such as MySQL or PostgreSQL are suitable choices, enabling efficient data querying and storage of car attributes and pricing data. Integration with cloud platforms like AWS or Google Cloud facilitates scalability, resource management, and deployment of machine learning models as APIs or web applications. Version control systems like Git ensure collaboration, code management, and tracking changes throughout the development lifecycle. Continuous integration and deployment (CI/CD) pipelines automate testing, validation, and deployment processes, ensuring reliability, scalability, and rapid iteration of the car price prediction system.

8.2 Pseudo Code

Step 1: Import the required python libraries.

Step 2: Download the dataset and link it to the Google Colab.

Step 3: Read the dataset and perform operations on data.

Step 4: Data cleaning.

Step 5: Data Preprocessing.

Step 6: Saving the cleaned car data set after performing operations on data.

Step 7: Start training the Machine learning Model.

Step 8: Split features and target as x and y respectively.

Step 9: Split the new data into 80% of Training data and 20% of Testing data.

Step 10: Train the model with Training data and Testing data.

Step 11: Implementing one hot encoder and column transformer to model.

Step 12: Applying Linear Regression to the model.

Step 13: Fit the Linear Regression Model.

Step 14: If accuracy is good use the model for prediction else fit the model again, using other random states.

Step 15: Dump the Linear Regression model into our files using pickle.

Step 16: Open Visual Studio Code and extract the cleaned car.csv and LinearRegression model.pkl files into our project.

Step 17: Reading the model and dataset, make the prediction using python and streamlit from webpage.

8.3 Google Collab Data set Implementation

1. Import the required python libraries:

```
import pandas as pd
import numpy as np

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

2. Read the Data:

```
cars_data = pd.read_csv('/content/Cardetails.csv')

cars_data.head()
```

	name	year	selling_price	km_driven	fuel	seller_type	transmission	owner	mileage	engine	max_power	torque	seats
0	Maruti Swift Dzire VDI	2014	450000	145500	Diesel	Individual	Manual	First Owner	23.4 kmpl	1248 CC	74 bhp	190Nm@ 2000rpm	5.0
1	Skoda Rapid 1.5 TDI Ambition	2014	370000	120000	Diesel	Individual	Manual	Second Owner	21.14 kmpl	1498 CC	103.52 bhp	250Nm@ 1500-2500rpm	5.0
2	Honda City 2017-2020 EXi	2006	158000	140000	Petrol	Individual	Manual	Third Owner	17.7 kmpl	1497 CC	78 bhp	12.7@ 2,700(kgm@ rpm)	5.0
3	Hyundai i20 Sportz Diesel	2010	225000	127000	Diesel	Individual	Manual	First Owner	23.0 kmpl	1396 CC	90 bhp	22.4 kgm at 1750-2750rpm	5.0
4	Maruti Swift VXI BSIII	2007	130000	120000	Petrol	Individual	Manual	First Owner	16.1 kmpl	1298 CC	88.2 bhp	11.5@ 4,500(kgm@ rpm)	5.0

Figure 12: Read Dataset

3. Drop unnecessary column from data:

```
cars_data.drop(columns=['torque'], inplace=True)
```

4. Updated Data

```
cars_data.head()
```

	name	year	selling_price	km_driven	fuel	seller_type	transmission	owner	mileage	engine	max_power	seats
0	Maruti Swift Dzire VDI	2014	450000	145500	Diesel	Individual	Manual	First Owner	23.4 kmpl	1248 CC	74 bhp	5.0
1	Skoda Rapid 1.5 TDI Ambition	2014	370000	120000	Diesel	Individual	Manual	Second Owner	21.14 kmpl	1498 CC	103.52 bhp	5.0
2	Honda City 2017-2020 EXi	2006	158000	140000	Petrol	Individual	Manual	Third Owner	17.7 kmpl	1497 CC	78 bhp	5.0
3	Hyundai i20 Sportz Diesel	2010	225000	127000	Diesel	Individual	Manual	First Owner	23.0 kmpl	1396 CC	90 bhp	5.0
4	Maruti Swift VXI BSIII	2007	130000	120000	Petrol	Individual	Manual	First Owner	16.1 kmpl	1298 CC	88.2 bhp	5.0

Figure 13: Updated Data

5. Show data info

cars_data.shape #(row,column)

(8128, 12)

cars_data.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8128 entries, 0 to 8127
Data columns (total 13 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   name           8128 non-null   object
 1   year           8128 non-null   int64
 2   selling_price  8128 non-null   int64
 3   km_driven      8128 non-null   int64
 4   fuel           8128 non-null   object
 5   seller_type    8128 non-null   object
 6   transmission   8128 non-null   object
 7   owner          8128 non-null   object
 8   mileage        7907 non-null   object
 9   engine         7907 non-null   object
 10  max_power      7913 non-null   object
 11  torque         7906 non-null   object
 12  seats          7907 non-null   float64
dtypes: float64(1), int64(3), object(9)
memory usage: 825.6+ KB
```

Figure 14: Data Info

6. Preprocessing technique

#Check NULL value
cars_data.isnull().sum()

```
name              0
year              0
selling_price     0
km_driven         0
fuel              0
seller_type       0
transmission      0
owner             0
mileage         221
engine          221
max_power       215
seats           221
dtype: int64
```

Figure 15: Check Null Value

#Drop NULL value
cars_data.dropna(inplace=True)

cars_data.shape

(7907, 12)

#Duplicate value Check
cars_data.duplicated().sum()

1189

#Drop Duplicate value
cars_data.drop_duplicates(inplace=True)

cars_data.shape

(6717, 13)

7. Data Analysis

7.1 Average Selling Price of Cars by Year

The "Average Selling Price of Cars by Year" chart visualizes the trend of average selling prices of cars over different years. Here's a description of the chart:

Description: The chart presents a graphical representation of the average selling prices of cars across various years. Each data point on the chart corresponds to the average selling price of cars for a specific year.

X-axis: Year- The horizontal axis represents the years in which the cars were sold. Each tick mark on the x-axis indicates a specific year.

Y-axis: Average Selling Price: The vertical axis represents the average selling prices of cars. The values on the y-axis indicate the average selling price of cars for each corresponding year.

Data Points: Each data point on the chart represents the average selling price of cars for a particular year. The data points are connected by lines to visualize the trend over time.

Overall, the "Average Selling Price of Cars by Year" chart serves as a valuable tool for visualizing and understanding the historical trends in the average selling prices of cars, thereby aiding in decision-making processes.

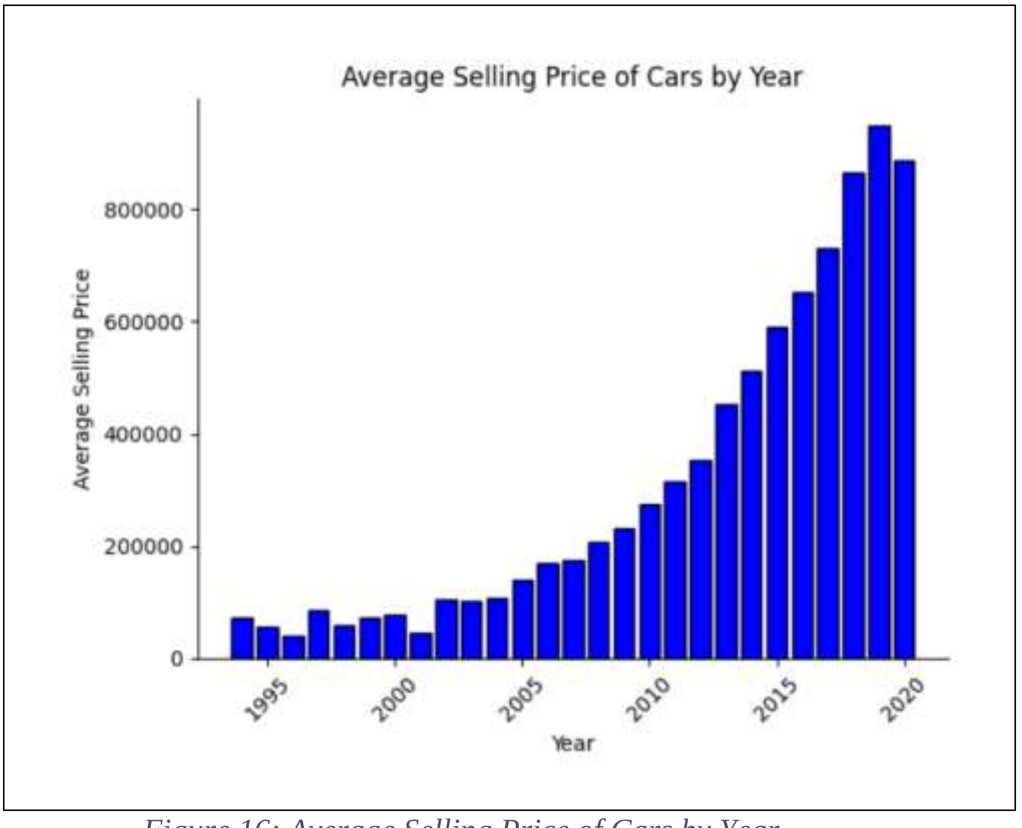

Figure 16: Average Selling Price of Cars by Year

7.2 Average Selling Price by Fuel Type

The "Average Selling Price by Fuel Type" chart displays the average selling price of cars categorized by different fuel types. Here's a description of the chart:

X-axis (Fuel Type): The x-axis represents the different fuel types of cars. Each category on the x-axis corresponds to a specific type of fuel used by the cars.

Y-axis (Average Selling Price): The y-axis represents the average selling prices of cars. It indicates the monetary value at which cars of each fuel type are sold, on average.

Chart Content: The chart consists of several bars, each representing a different fuel type. The height of each bar represents the average selling price of cars belonging to that particular fuel type. The color of each bar may vary to differentiate between different fuel types.

Overall, the "Average Selling Price by Fuel Type" chart serves as a visual representation of the relationship between fuel type and selling price in the car market, allowing viewers to quickly grasp key insights and trends.

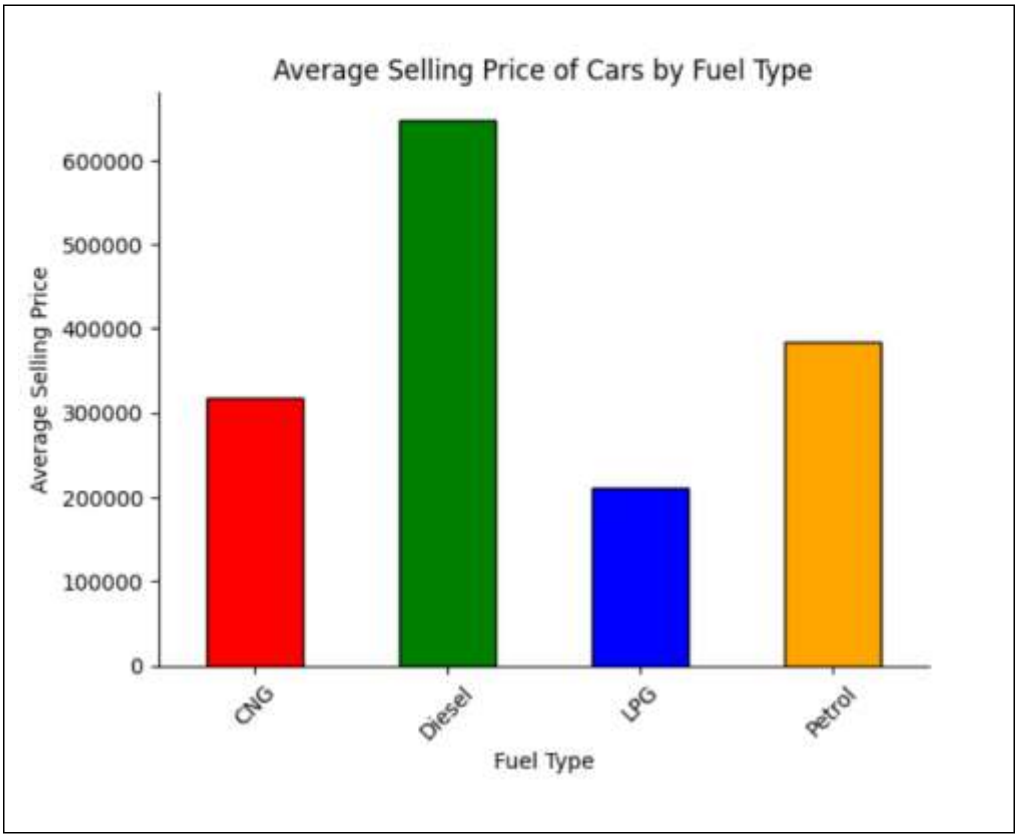

Figure 17: Average Selling Price by Fuel Type

7.3 Average Selling Price of Cars by no of Seats

The "Average Selling Price of Cars by Number of Seats" chart provides insight into how the average selling price of cars varies based on the number of seats they have. Here's a description of what the chart reveals:

X-axis (Number of Seats): The horizontal axis of the chart represents the number of seats in the cars. Each point on the x-axis corresponds to a specific number of seats, ranging from the minimum to the maximum number found in the dataset.

Y-axis (Average Selling Price): The vertical axis of the chart represents the average selling price of cars. The average selling price is calculated for each category of the number of seats. It indicates the typical price at which cars with a particular number of seats are sold.

Overall, the chart offers valuable information for stakeholders in the automotive industry, including manufacturers, dealerships, and consumers, helping them understand pricing dynamics and make informed decisions related to car purchases and sales.

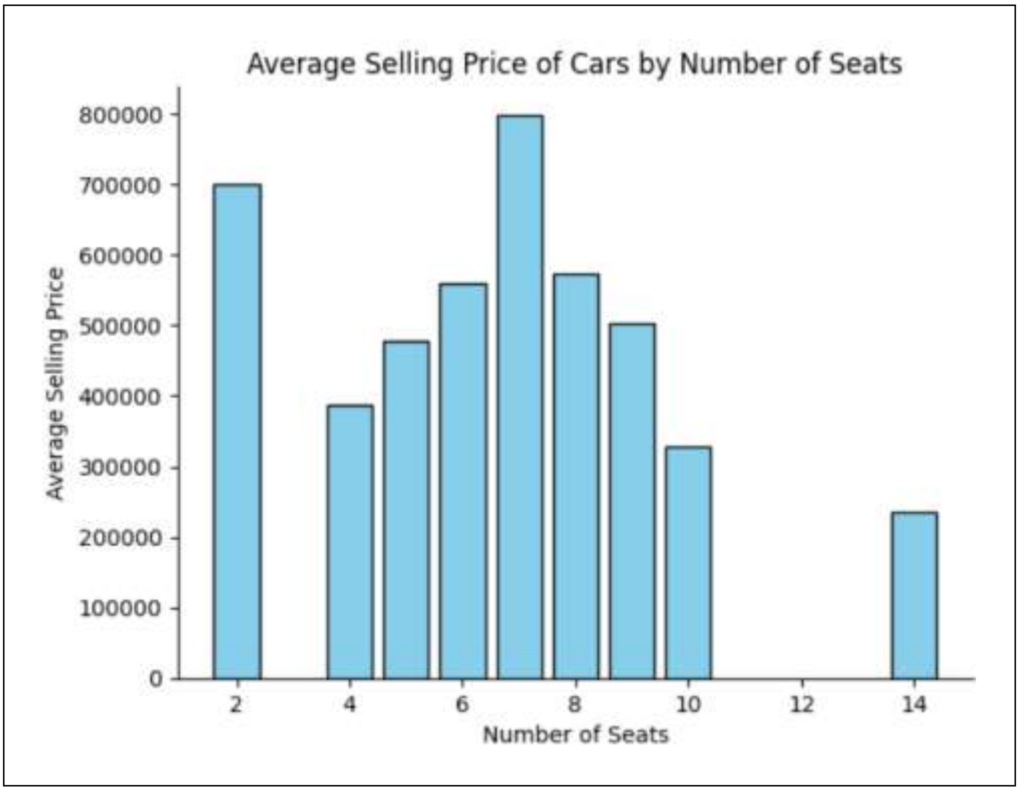

Figure 18: Average Selling Price of Cars by no of Seats

7.4 Distribution of Car Owners

The "Distribution of Car Owners" pie chart visualizes the proportion of cars owned by different categories of owners. The chart provides insights into the distribution of ownership among various owner types, such as "First Owner" and "Second Owner" so on.

Each slice of the pie represents a specific owner category, and the size of each slice corresponds to the proportion of cars owned by that category relative to the total number of cars. The chart is titled "Distribution of Car Owners" to clearly convey its purpose.

The percentage displayed within each slice indicates the proportion of cars owned by each category relative to the total number of cars, providing a quick reference for understanding the relative distribution of ownership.

Overall, the pie chart offers a visual summary of the distribution of car ownership by different owner types, enabling viewers to quickly grasp the relative prevalence of each ownership category within the dataset.

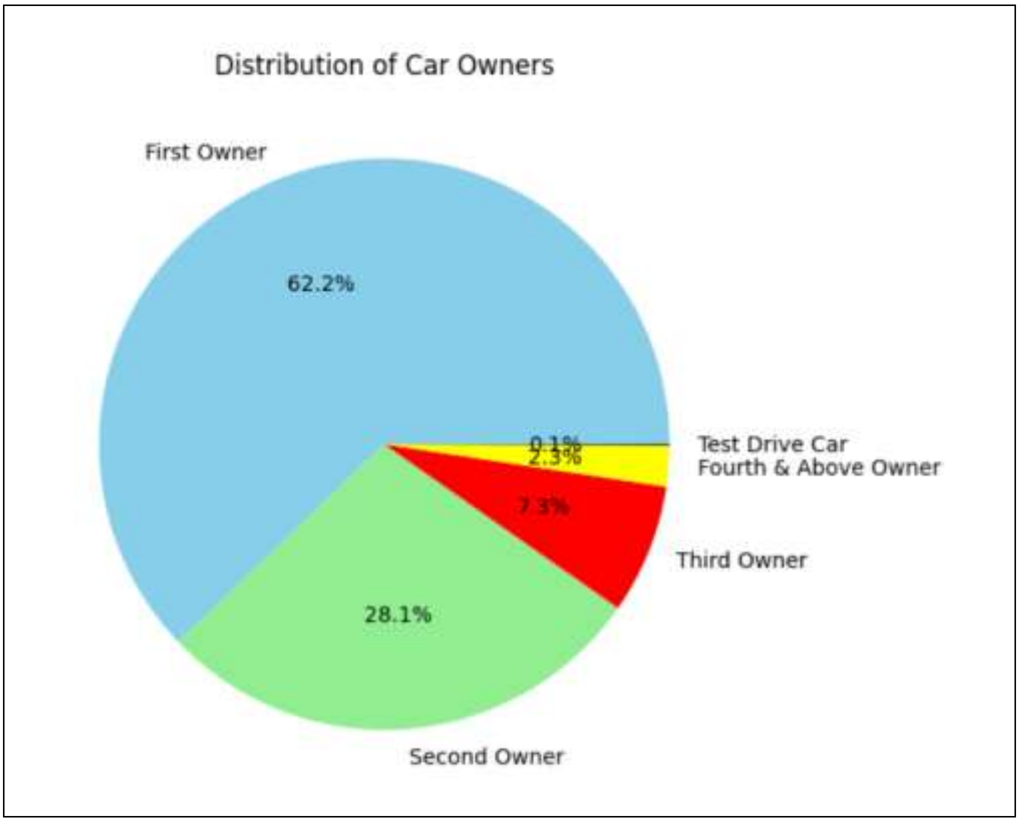

Figure 19: Distribution of Car Owners

7.5 Number of Cars by Transmission and Year

The line chart "Number of Cars by Transmission and Year" visualizes the distribution of cars over time based on their transmission types. Here's a detailed description of the chart:

X-axis (Year): The horizontal axis represents the years in which the cars were manufactured. Each tick on the x-axis corresponds to a specific year, indicating the time period covered by the data.

Y-axis (Number of Cars): The vertical axis represents the number of cars. It shows the count of cars for each combination of transmission type and year.

Lines: Each line on the chart represents a different transmission type (e.g., manual, automatic). The lines connect data points corresponding to the count of cars with that transmission type for each year.

Markers: Data points on the lines are marked with circular markers (o) for visibility and to indicate the count of cars for each transmission type in a particular year.

Overall, the line chart provides a clear visual representation of the relationship between the number of cars, transmission types, and years, facilitating insights into historical trends and patterns in the automotive industry.

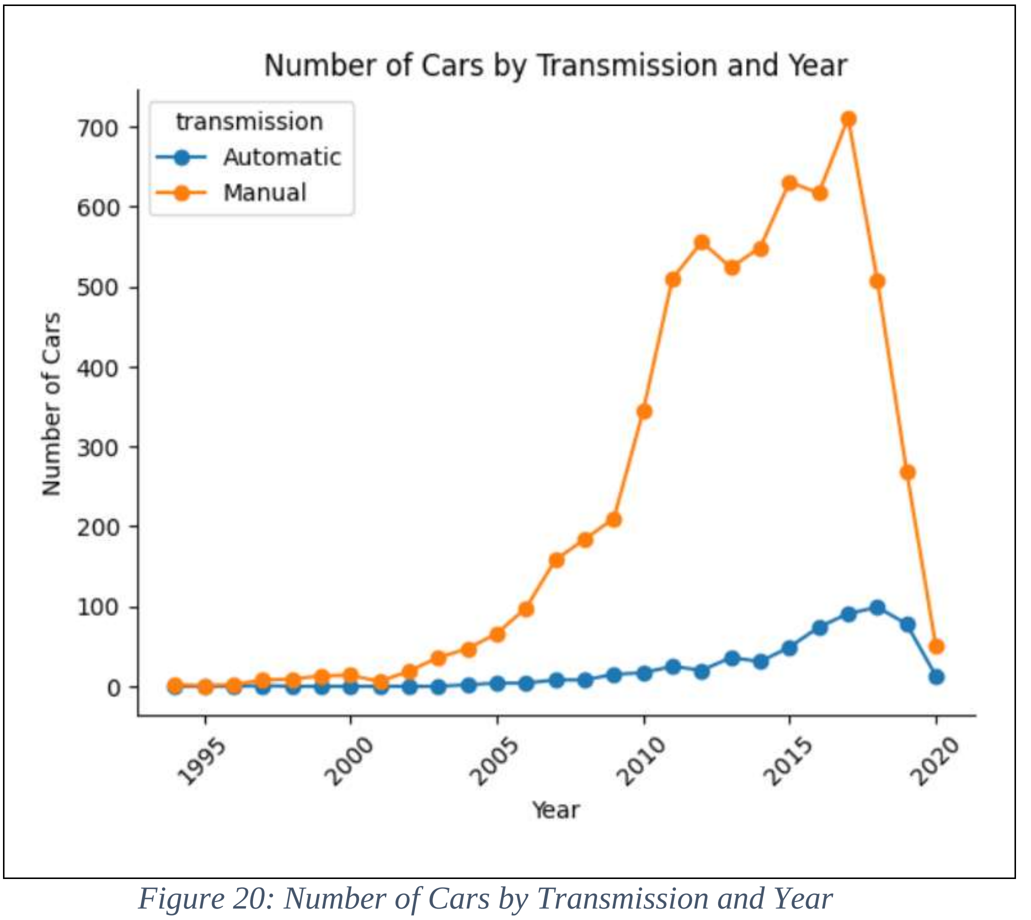

Figure 20: Number of Cars by Transmission and Year

8 Train & Test Data split

x_train, x_test, y_train, y_test = train_test_split(input_data, output_data, test_size=0.2)

- 80% of the dataset goes to train dataset.
- 20% of the dataset goes to test dataset.

9 Model Creation & Training

model = LinearRegression() #model

model.fit(x_train, y_train) #Train Model

```
▾ LinearRegression
LinearRegression()
```

Figure 21: Model Creation & Training

10 Predicting value from model

predict = model.predict(x_test) #predict value for test dataset
predict

```
array([ 655799.51277582, 1194891.21238762,  469426.34860355, ...,
        1712213.2606869 ,  750228.65117644,  324611.3214426 ])
```

Figure 22: Predicting value from model

input_data_model = pd.DataFrame(
[[5,2022,12000,1,1,1,1,12.99,2494.0,100.6,5.0]],

columns=['name','year','km_driven','fuel','seller_type','transmission','owner',' mileage','engine','max_power','seats'])

input_data_model

	name	year	km_driven	fuel	seller_type	transmission	owner	mileage	engine	max_power	seats
0	5	2022	12000	1	1	1	1	12.99	2494.0	100.6	5.0

Figure 23: Input data model

model.predict(input_data_model) #predicted price

```
array([977472.78668103])
```

11 Save Model

```
import pickle as pk
pk.dump(model,open('model.pkl','wb'))
```

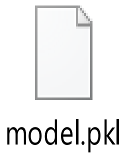

Figure 24: Model

CHAPTER 9: TESTING

9.1 Introduction to Testing

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. Testing is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

According to ANSI/IEEE 1059 standard, Testing can be defined as - A process of analyzing a software item to detect the differences between existing and required conditions (that is defects/errors/bugs) and to evaluate the features of the software item.

Who does Testing?

It depends on the process and the associated stakeholders of the project(s). In the IT industry, large companies have a team with responsibilities to evaluate the developed software in context of the given requirements. Moreover, developers also conduct testing which is called Unit Testing. In most cases, the following professionals are involved in testing a system within their respective capacities:

- Software Tester
- Software Developer
- Project Lead/Manager
- End User

Levels of testing include different methodologies that can be used while conducting software testing. The main levels of software testing are:

- Functional Testing
- Non-functional Testing

Functional Testing

This is a type of black-box testing that is based on the specifications of the software that is to be tested. The application is tested by providing input and then the results are examined that need to conform to the functionality it was intended for. Functional testing of a software is conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.

9.1.1 Software Testing Life Cycle

The process of testing a software in a well-planned and systematic way is known as software testing lifecycle (STLC).

Different organizations have different phases in STLC however generic Software Test Life Cycle (STLC) for waterfall development model consists of the following phases.

1. Requirements Analysis
2. Test Planning
3. Test Analysis
4. Test Design

● **Requirements Analysis**
In this phase testers analyze the customer requirements and work with developers during the design phase to see which requirements are testable and how they are going to test those requirements. It is very important to start testing activities from the requirements phase itself because the cost of fixing defect is very less if it is found in requirements phase rather than in future phases.

● **Test Planning**
In this phase all the planning about testing is done like what needs to be tested, how the testing will be done, test strategy to be followed, what will be the test environment, what test methodologies will be followed, hardware and software availability, resources, risks etc. A high level test plan document is created which includes all the planning inputs mentioned above and circulated to the stakeholders.

● **Test Analysis**
After test planning phase is over test analysis phase starts, in this phase we need to dig deeper into project and figure out what testing needs to be carried out in each SDLC phase. Automation activities are also decided in this phase, information needs to be done for software product, how will the automation be done, how much time will it take to automate and which features need to be automated. Non functional testing areas(Stress and performance testing) are also analyzed and defined in this phase.

● **Test Design**
In this phase various black-box and white-box test design techniques are used to design the test cases for testing, testers start writing test cases by following those design techniques, if automation testing needs to be done then automation scripts also needs to written in this phase.

9.2 Test Cases

1. Launching Home Page.
2. Displaying attributes.
3. Selecting attributes.
4. Price Prediction Test case with selecting correct attributes.
5. Price Prediction Test case without selecting one or more attributes.
6. Predict button Test case.

9.2.1 Launching Home Page

Test Scenario ID	Launching Web application		**Test Case ID**	HomePage-1A	
Test Case Description	Launching Home Page		**Test Priority**	High	
Pre-Requisite	NA		**Post Requisite**	NA	
Test Execution Steps:					
Action	**Inputs**	**Expected Output**	**Actual Output**	**Test Browser**	**Test Result**
Localhost //8501	Click on url: http://localhost:8501/	Home page	Home page launched	Chrome	Pass

Table 2: Launching Home Page

9.2.2 Displaying attributes

Test Scenario ID	Launching Home Page		**Test Case ID**	HomePage-1A	
Test Case Description	Displaying attributes		**Test Priority**	High	
Pre-Requisite	Valid URL		**Post Requisite**	NA	
Test Execution Steps:					
Action	**Inputs**	**Expected Output**	**Actual Output**	**Test Browser**	**Test Result**
Localhost //8501	Scroll and view	Displaying attributes	Displaying attributes	Chrome	Pass

Table 3: Displaying attributes

9.2.3 Selecting Attributes

Test Scenario ID	Attribute Selection	**Test Case ID**	HomePage-1A
Test Case Description	Selecting	**Test Priority**	High
Pre-Requisite	Valid URL	**Post Requisite**	NA

Test Execution Steps:

Action	**Inputs**	**Expected Output**	**Actual Output**	**Test Browser**	**Test Result**
Scroll and select	Selecting car brand	Car brand selected by user	Car brand selected	Chrome	Pass
Entering thevalue	Enter Car Manufactured Year	Car Manufactured Year is enteredby user	Car Manufactured Year range is entered	Chrome	Pass
Entering thevalue	Enter kilometers driven	Kilometers driven is enteredby user	Kilometers driven is entered	Chrome	Pass
Scroll and select	Selecting fuel type	Fuel type is selected by user	Fuel type selected	Chrome	Pass
Scroll and select	Selecting Seller type	Seller type is selected by user	Seller type selected	Chrome	Pass
Scroll and select	Selecting Transmission type	Transmission type is selected by user	Transmission type selected	Chrome	Pass
Scroll and select	Selecting Owner type	Owner type is selected by user	Owner type selected	Chrome	Pass
Entering thevalue	Enter Car Mileage	Car Mileage is enteredby user	Car Mileage is entered	Chrome	Pass

Entering the value	Enter Engine CC	Engine CC is entered by user	Engine CC is entered	Chrome	Pass
Entering thevalue	Enter Max Power	Max Power is entered by user	Max Power is entered	Chrome	Pass
Entering the value	Enter No of Seats	No of Seats is entered by user	No of Seats is entered	Chrome	Pass

Table 4: Selecting Attributes

9.2.4 Price Prediction Test case with selecting correct attributes

Test Scenario ID	Price Prediction		**Test Case ID**	HomePage-1A	
Test Case Description	Predicting the price of the car		**Test Priority**	High	
Pre-Requisite	Valid URL		**Post Requisite**	NA	
Test Execution Steps:					
Action	**Inputs**	**Expected Output**	**Actual Output**	**Test Browser**	**Test Result**
Click on predict price button	Clicking on predict pricebutton	Price is predicted at bottom	Price predicted	Chrome	Pass

Table 5: Price Prediction Test case with selecting correct attributes

9.2.5 Price Prediction Test case without selecting one or more attributes

Test Scenario ID	Price Prediction	**Test Case ID**	HomePage-1A
Test Case Description	Predicting the price of the car	**Test Priority**	High
Pre-Requisite	Valid URL	**Post Requisite**	NA

Test Execution Steps:

Action	**Inputs**	**Expected Output**	**Actual Output**	**Test Browser**	**Test Result**
Click on predictprice button without filling all attributes	Clicking on predict pricebutton	Fill all attributes	Fill all the attributes and price is not predicted.	Chrome	Pass
Click on predict price button with filling incorrect attributes	Clicking on predict pricebutton	Incorrect attributes	Price is not predicted.	Chrome	Pass

Table 6: Price Prediction Test case with selecting correct attributes

9.2.6 Predict button Test case

Test Scenario ID	Predict button	**Test Case ID**	HomePage-1A
Test Case Description	Testing Predict button	**Test Priority**	High
Pre-Requisite	Valid URL	**Post Requisite**	NA

Test Execution Steps:

Action	**Inputs**	**Expected Output**	**Actual Output**	**Test Browser**	**Test Result**
Click on Predict button	Click	Predict Price	Display Predicted Price	Chrome	Pass

Table 7: Predict button Test case

CHAPTER 10: USER MANUAL

10.1 Home Page of web app

This page showcases a user interface for a “Car Price Prediction ML Model.” Within this interface, users can input specific details about a car to predict its selling price.

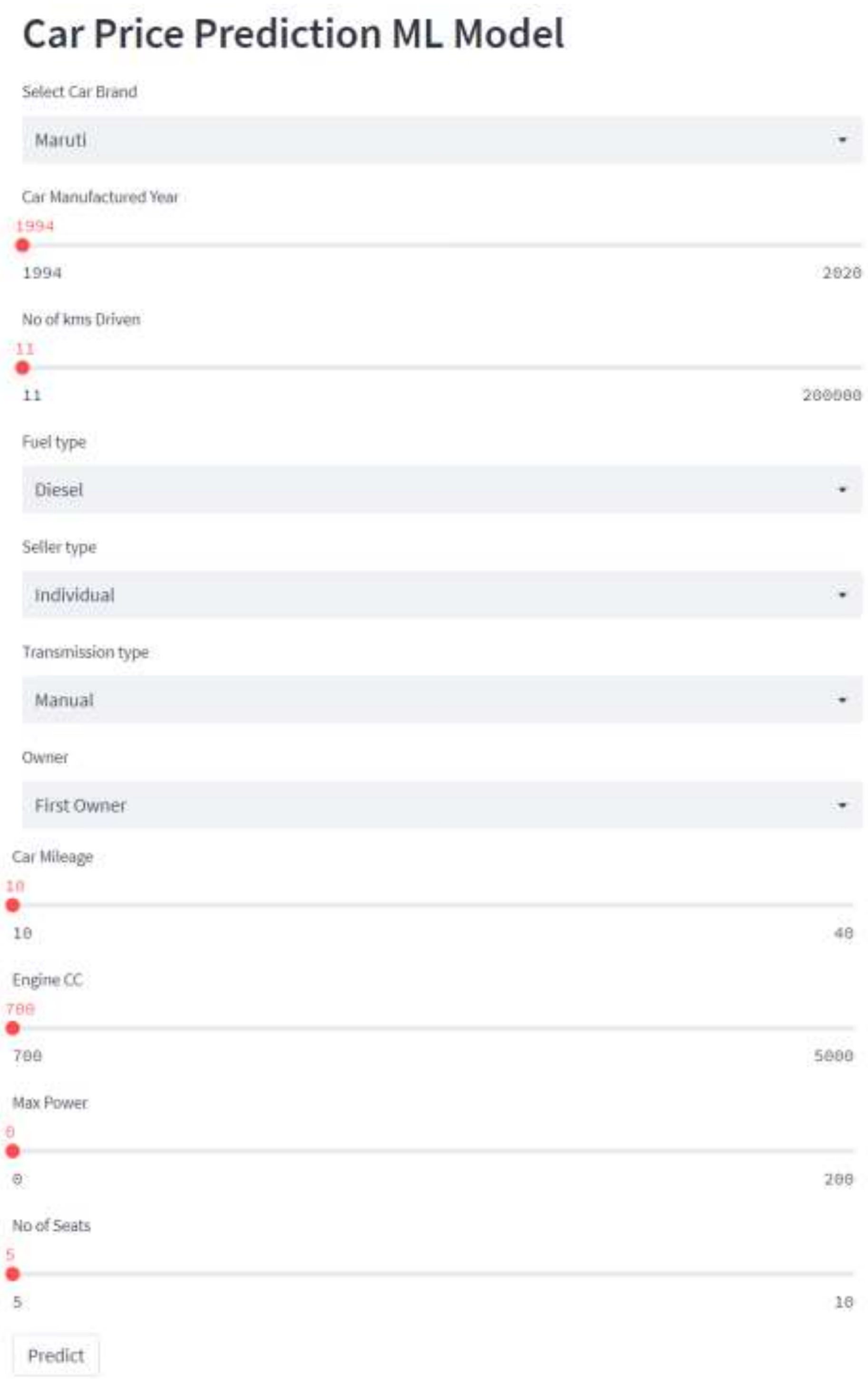

Figure 25: Home Page of web app

10.2 Displaying available car brands

This interface shows features dropdown menus for selecting car brands attributes. The "Select Car Brand" dropdown includes options like Maruti, Skoda, Honda, Hyundai, Toyota, Ford, Renault, Mahindra, and other car brand categories.

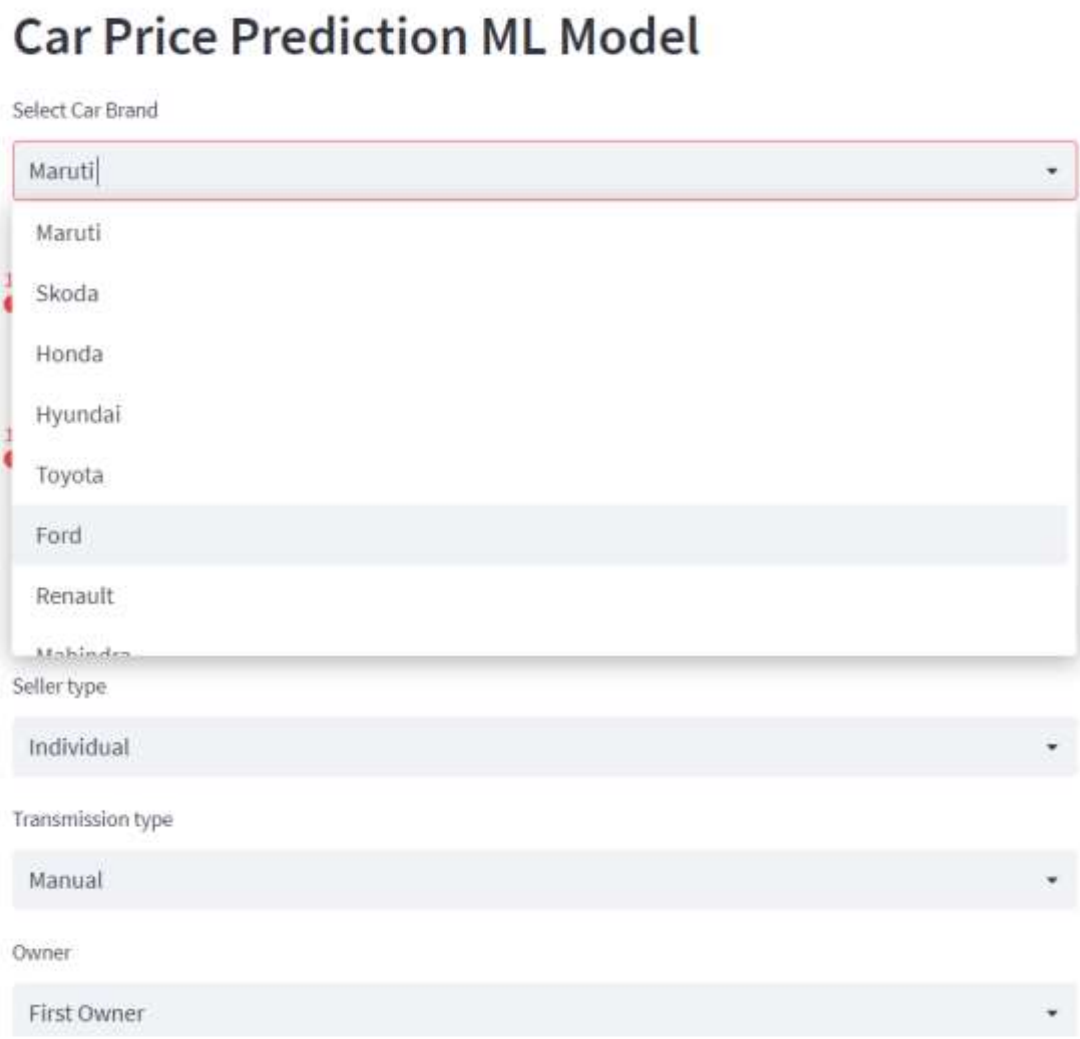

Figure 26: Displaying available car brands

10.3 Selecting Car Manufactured Year

This interface shows slider menus for selecting Car Manufactured Year range. The "Car Manufactured Year" slider includes range string from 1994 to 2020.

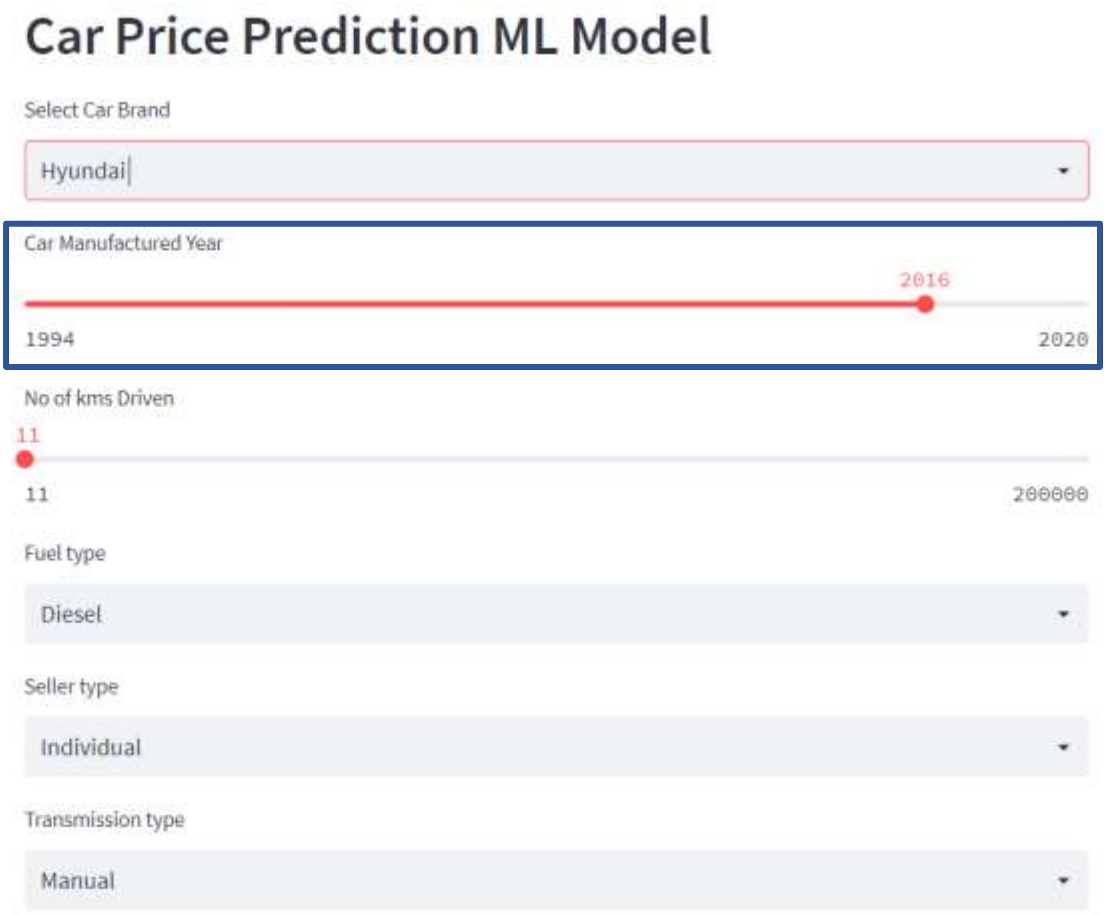

Figure 27: Selecting Car Manufactured Year

10.4 Selecting No of kms Car Driven

This interface shows slider menus for selecting No of kms Car Driven range. The “No of kms Driven” slider includes range string from 11 to 200000.

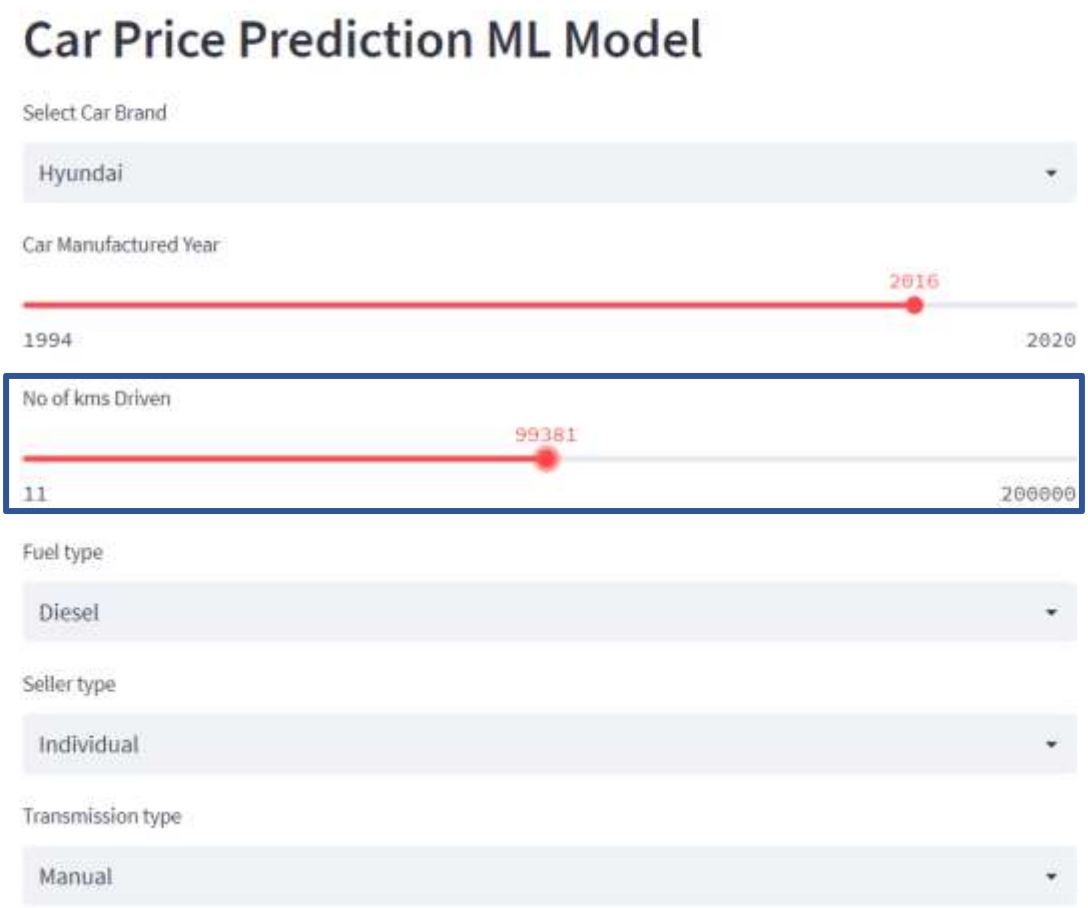

Figure 28: Selecting No of kms Car Driven

10.5 Selecting Car Fuel type

This interface shows features dropdown menus for selecting car fuel type attributes. The “fuel type” dropdown includes options like 'Diesel', 'Petrol', 'LPG', 'CNG'.

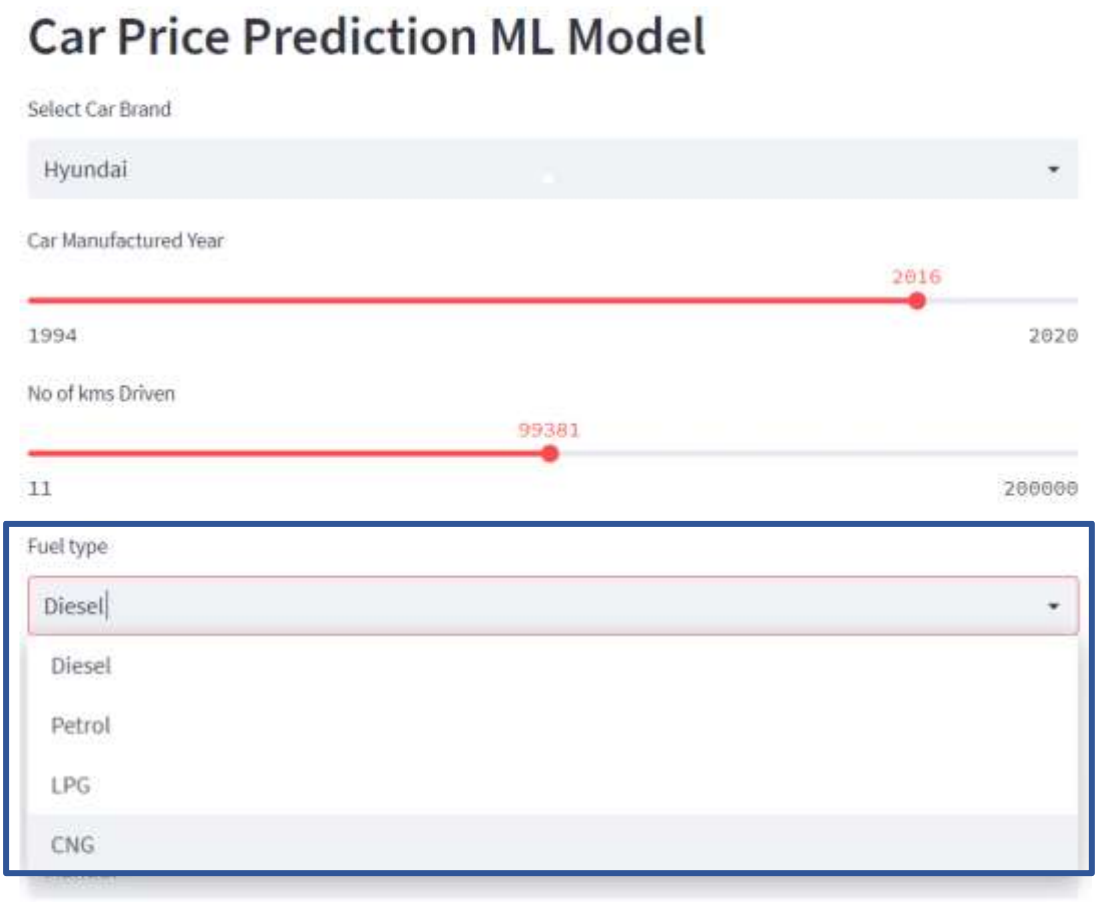

Figure 29: Selecting Car Fuel type

10.6 Selecting Car Seller type

This interface shows features dropdown menus for selecting car seller type attributes. The “Seller type” dropdown includes options like 'Individual', 'Dealer', 'Trustmark Dealer'.

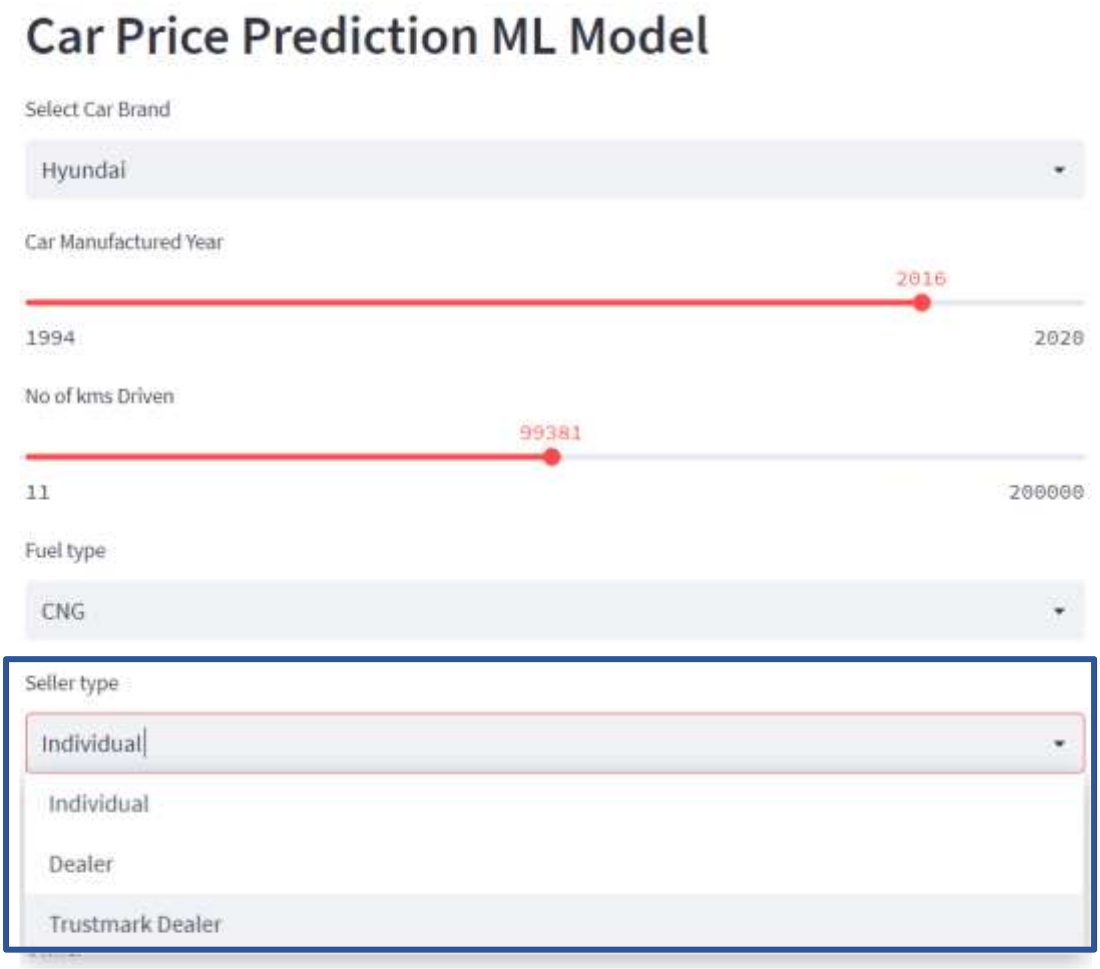

Figure 30: Selecting Car Seller type

10.7 Selecting Car transmission type

This interface shows features dropdown menus for selecting car transmission type attributes. The “Transmission Type” dropdown includes options like 'Diesel', 'Petrol', 'LPG', 'CNG'.

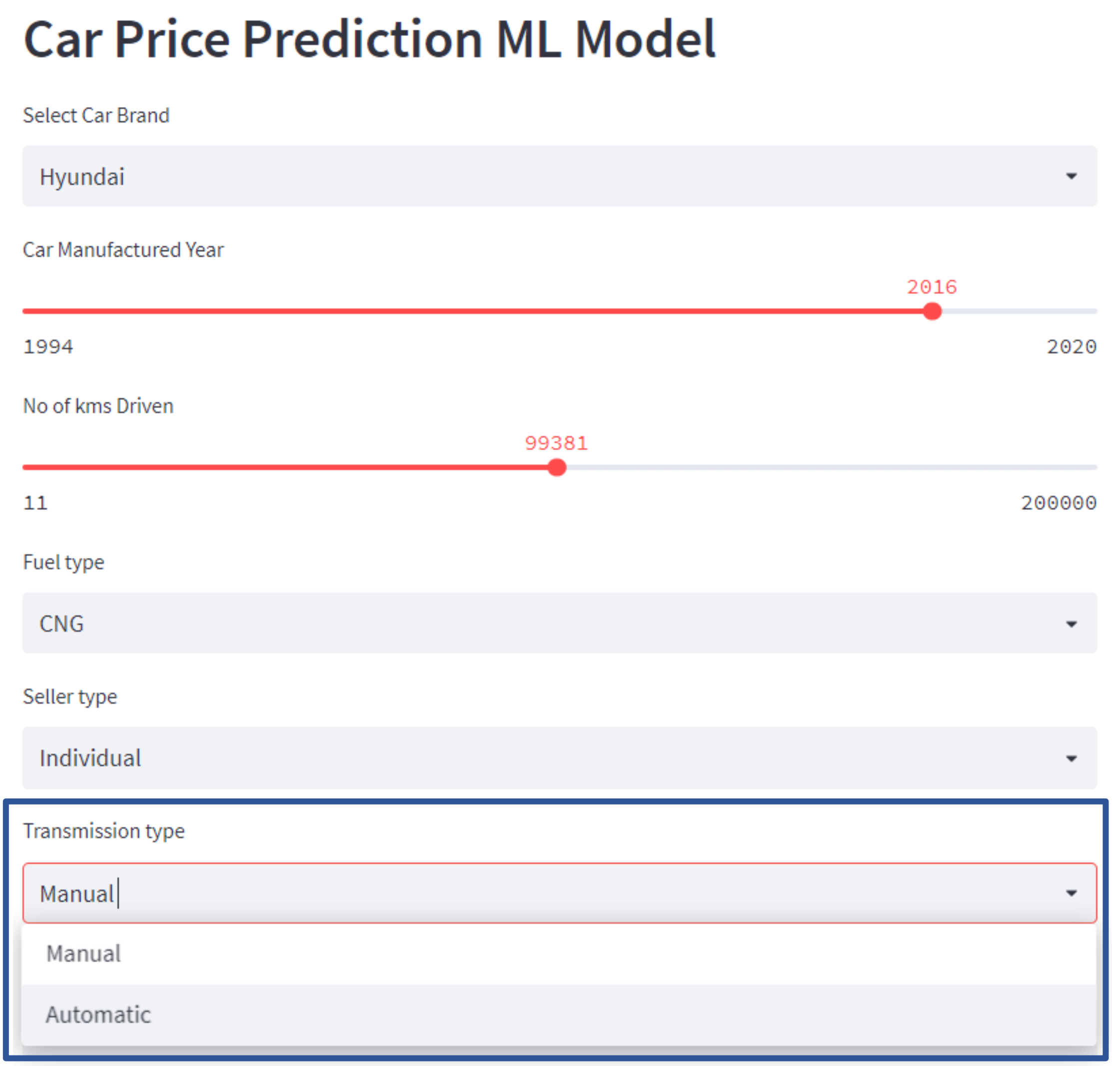

Figure 31: Selecting Car transmission type

10.8 Selecting Car Owner

This interface shows features dropdown menus for selecting car owner type attributes. The “Owner” dropdown includes options like 'First Owner', 'Second Owner', 'Third Owner', 'Fourth & Above Owner', 'Test Drive Car'.

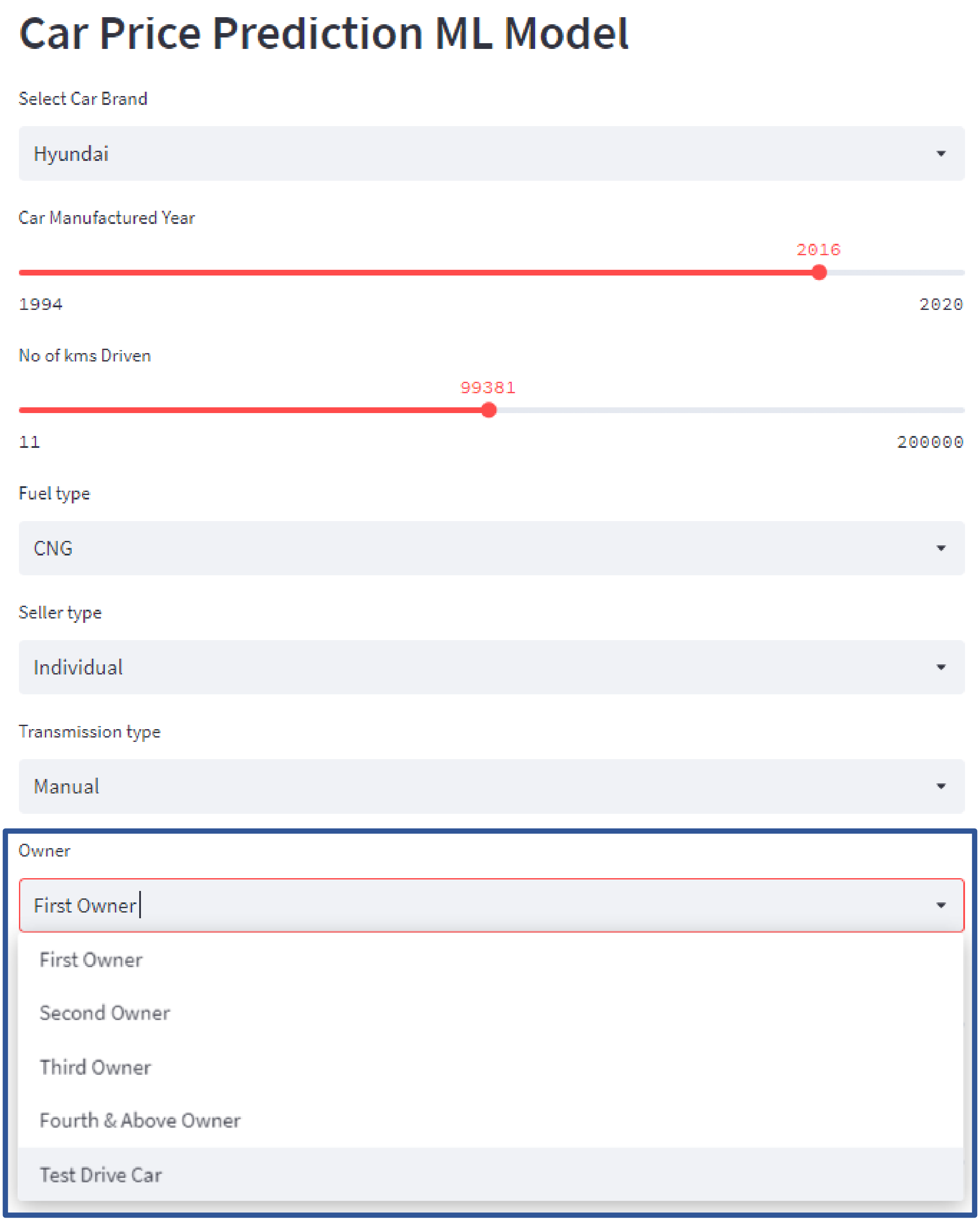

Figure 32: Selecting Car Owner

10.9 Selecting Car Mileage

This interface shows slider menus for selecting Car Mileage. The “Car Mileage” slider includes range string from 10 to 40.

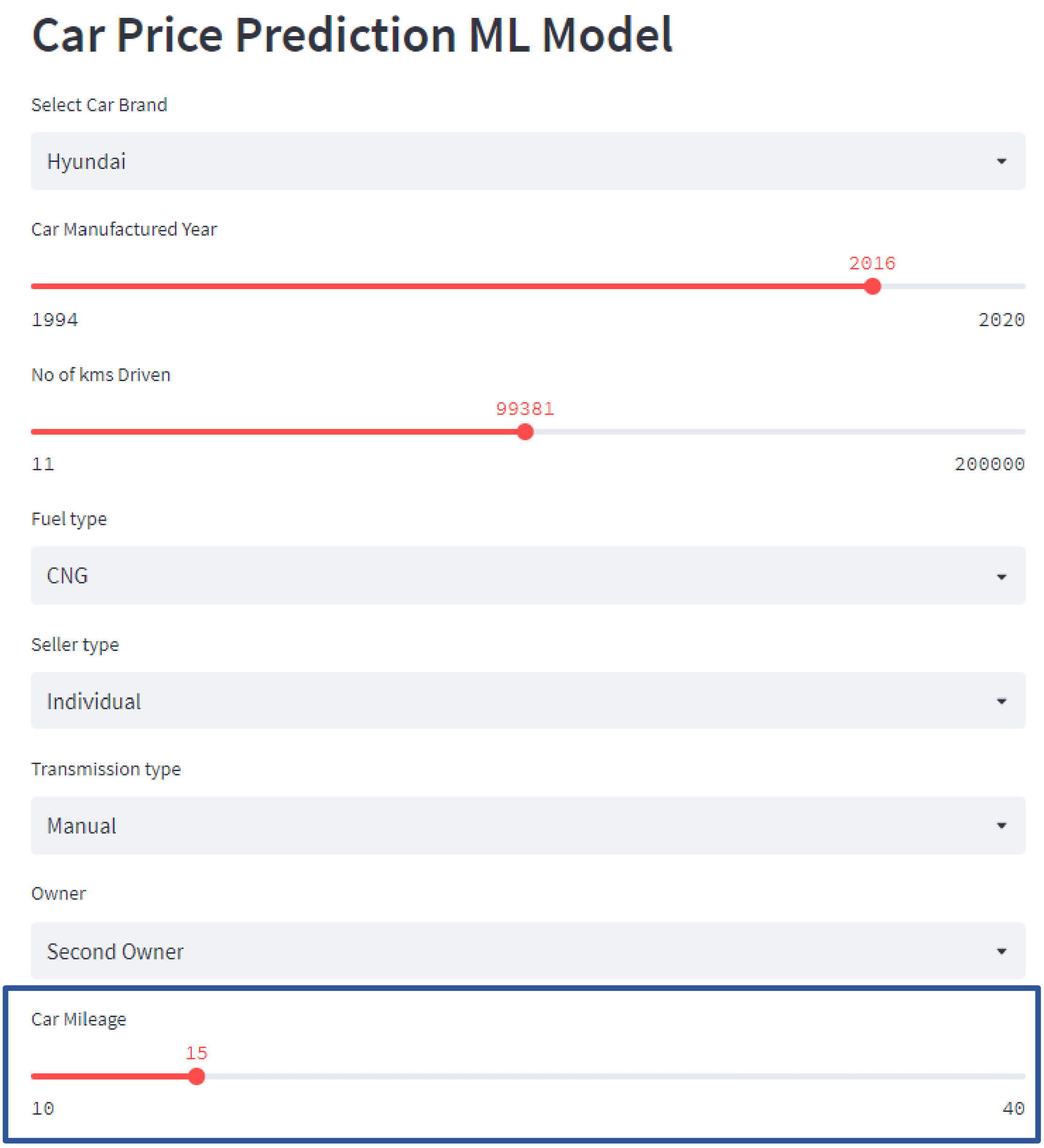

Figure 33: Selecting Car Mileage

10.10 Selecting Car Engine CC

This interface shows slider menus for selecting Car Engine CC. The “Engine CC” slider includes range string from 700 to 5000.

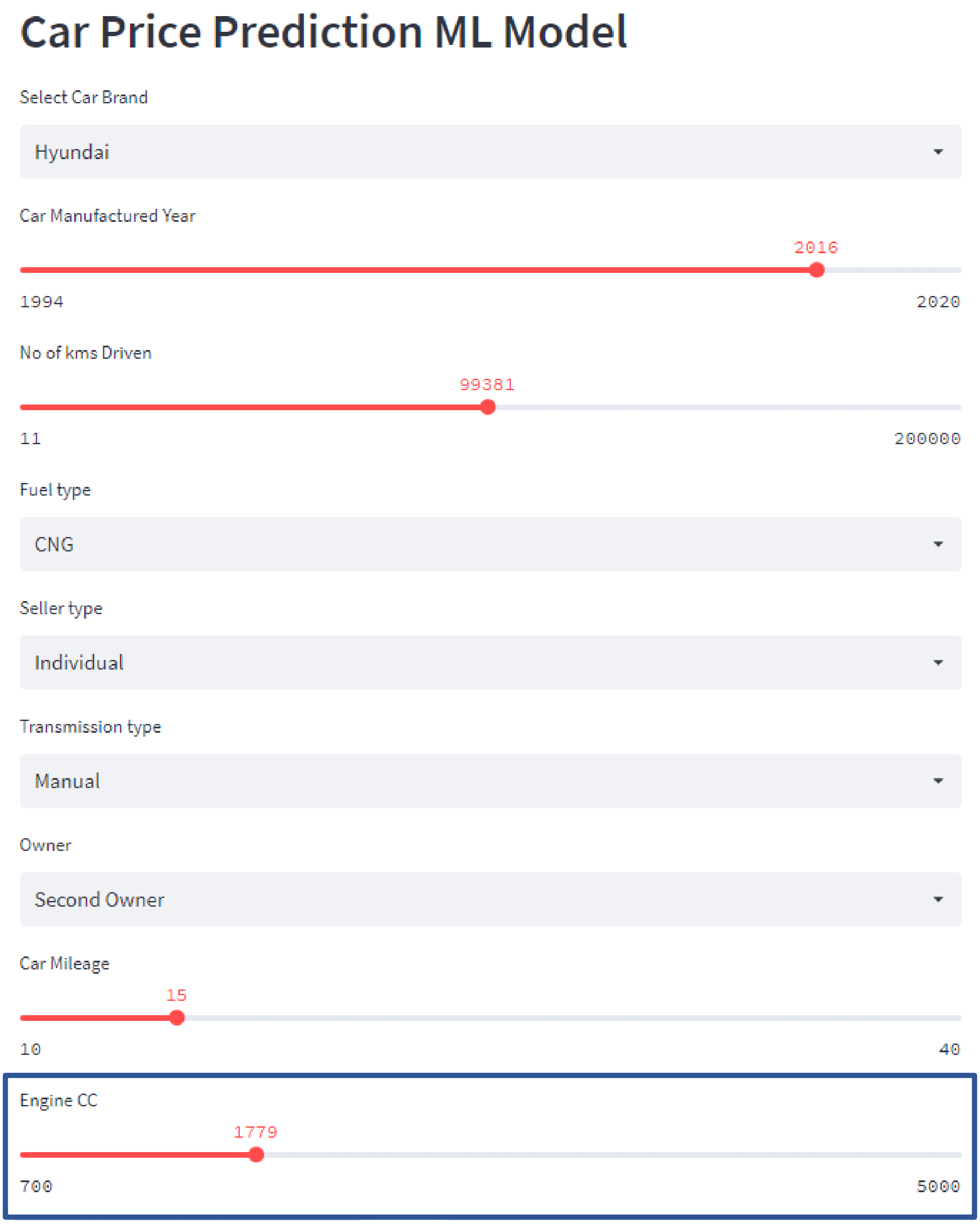

Figure 34: Selecting Car Engine CC

10.11 Selecting Car Max Power

This interface shows slider menus for selecting Car Max Power. The “Max Power” slider includes range string from 0 to 200.

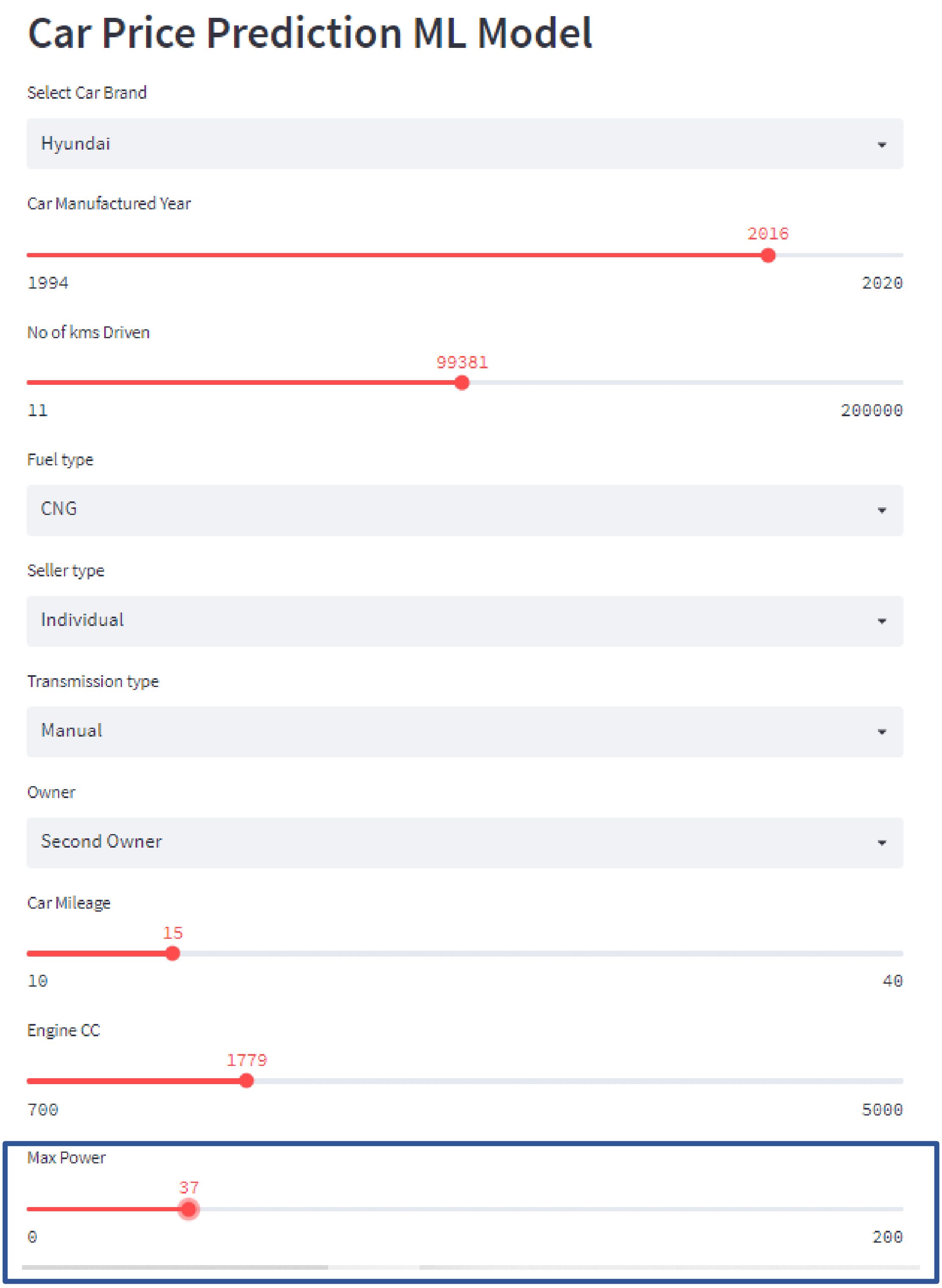

Figure 35: Selecting Car Max Power

10.12 Selecting No of Car Seats

This interface shows slider menus for selecting No of Car Seats. The "No of Seats" slider includes range string from 5 to 10.

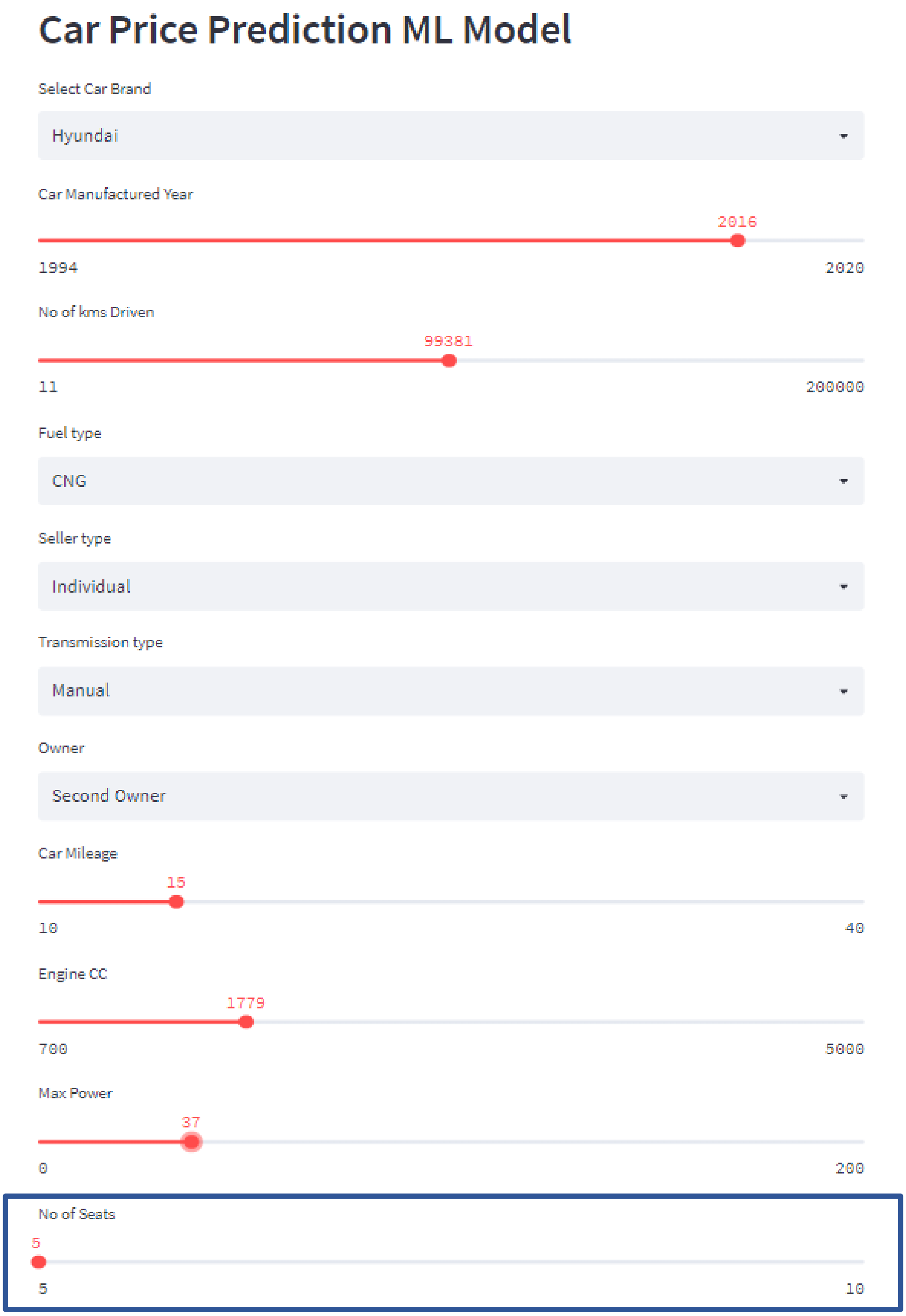

Figure 36: Selecti Selecting No of Car Seats

10.13 Displaying Predicted Price

This interface includes a “Predict” button for generating the prediction. The predicted price for the specified car details is displayed as “Car Price is going to be 16922.13.”

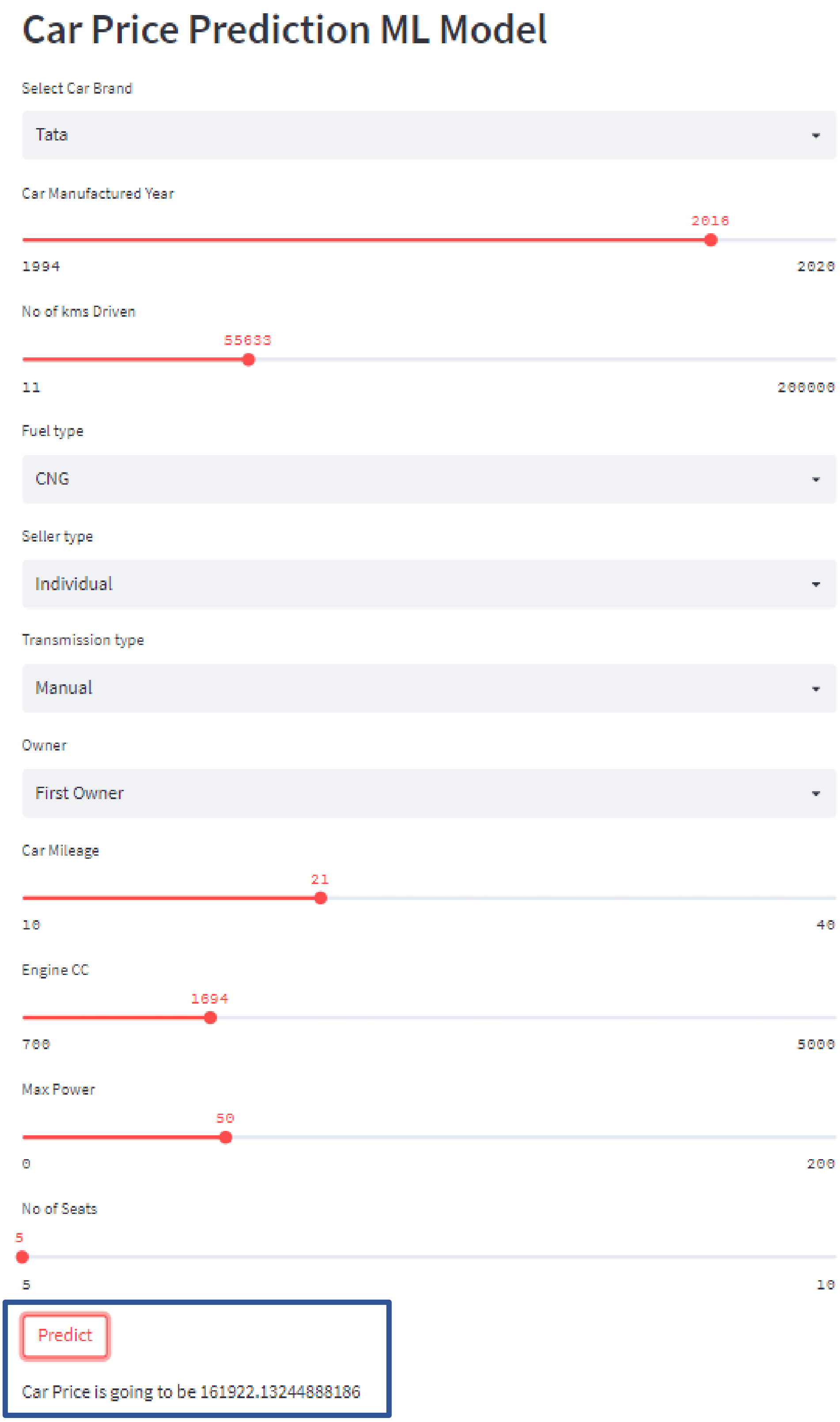

Figure 37: Displaying Predicted Price

10.14 Car Sales Dashboard Overview

Figure 38: Car Sales Dashboard Overview

The "Car Sales Dashboard Overview" provides a high-level view of car sales performance. It displays several key performance indicators (KPIs), including:

1) Sales Insight: This section shows year-to-date (YTD) total sales, YTD average price, and YTD cars sold. It also displays the percent change for each metric compared to the previous time period. For example, in the image, YTD total sales is $70.8 million, which is a 23.59% increase from the previous period.

2) YTD Sales Weekly Trend: This chart shows the trend of YTD sales over time.

3) YTD Total Sales by Body Style: This section visualizes the distribution of YTD total sales across different body styles (e.g., sedan, SUV, hatchback).

4) YTD Total Sales by Color: This section shows the distribution of YTD total sales across different car colors (e.g., pale white, black, red).

5) YTD Cars Sold by Dealer Region: This section shows sales performance across different dealer regions.

6) Company Wise Sales Trend: This table displays sales figures for each car company, including YTD average price, YTD cars sold, and YTD total sales. It also shows the percentage contribution of each company to YTD total sales.

By using this dashboard, car salespeople and managers can gain insights into sales performance and identify areas for improvement. For example, they can see which car models are selling well, which regions are performing well, and how sales are trending over time. This information can be used to develop targeted sales strategies and make data-driven decisions.

10.15 Car Sales Dashboard Details

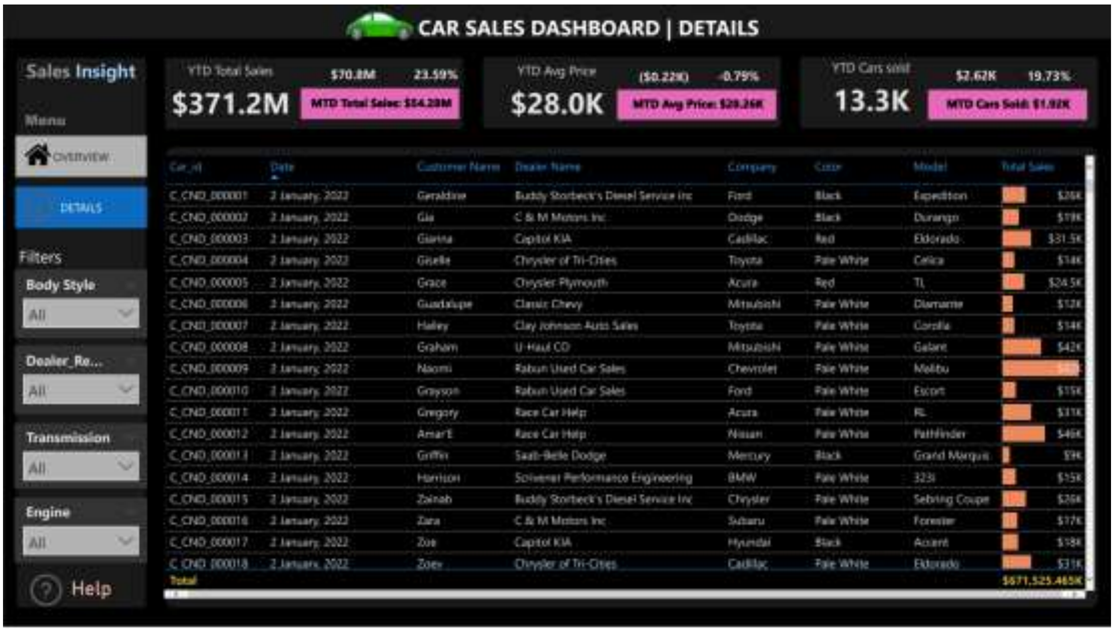

Figure 39: Car Sales Dashboard Details

The "Car Sales Dashboard Details" which is a subset of a larger "Car Sales Dashboard". This dashboard focuses on the details of individual car sales.

1) Sales Insight: This section provides a quick overview of sales performance for the chosen time period. In the image, it shows that YTD total sales is $70.8 million, which is a 23.59% increase from the previous period.

2) MTD Total Sales: This section shows month-to-date (MTD) total sales figures, including MTD total sales, MTD average price and MTD cars sold.

3) Details: This table displays detailed information about each car sold, including Car ID, Date, Customer Name, Dealer Name, Company, Color, Model, and Total Sales.

4) Filters: Users can filter the data in the Details table by various criteria, such as Body Style, Dealer Region, Engine, and Transmission.

This dashboard allows car dealerships to track individual car sales performance and identify trends. For example, they can see which car models are selling well, which dealerships are performing well, and which customer segments are buying which cars. This information can be used to identify sales opportunities and improve marketing campaigns.

CHAPTER 11: CONCLUSION AND FUTURE WORK

11.1 Conclusion

In conclusion, the project on car price prediction and sales analysis has offered valuable insights into the complex dynamics of the automotive market. Through meticulous data collection, preprocessing, and analysis, we have successfully developed predictive models capable of estimating car prices based on a myriad of attributes. Our examination of sales trends has illuminated patterns in consumer behaviour, enabling us to discern fluctuations in demand and the popularity of certain car models over time. Key findings include the identification of significant features influencing car prices, such as make, model, mileage, and transmission type, which have been instrumental in developing accurate pricing models. Looking ahead, there are several avenues for future exploration and enhancement. These include refining feature engineering techniques, improving model accuracy through advanced machine learning algorithms, implementing dynamic pricing strategies based on real-time market data, and conducting customer segmentation analysis to tailor marketing approaches effectively. Additionally, expanding the analysis to encompass broader geographic regions and integrating data from external sources could provide deeper insights into regional variations and consumer sentiment. By addressing these areas, the project can continue to evolve and offer invaluable tools for stakeholders in the automotive industry to make informed decisions and remain competitive in an ever-evolving market landscape.

11.2 Future Work

Looking forward, there are several way for future exploration and enhancement in the domain of car price prediction and sales analysis. Firstly, exploring advanced machine learning algorithms and techniques. Additionally, implementing dynamic pricing strategies based on real-time market data and consumer demand signals could optimize pricing decisions and maximize revenue for stakeholders. Moreover, conducting segmentation analysis to identify distinct customer segments based on preferences, demographics, and purchasing behavior would enable targeted marketing and sales strategies tailored to specific customer groups. Furthermore, extending the analysis to encompass additional geographic regions or market segments would provide insights into regional variations in car pricing and sales dynamics, facilitating more localized decision-making.

There are several areas for future exploration and enhancement:

1. Dynamic Pricing Strategies: Implement dynamic pricing strategies based on real-time market data and consumer demand signals to optimize pricing decisions and maximize revenue.
2. Customer Segmentation: Conduct segmentation analysis to identify distinct customer segments based on preferences, demographics, and purchasing behavior. Tailor marketing and sales strategies to target these segments effectively

3. Market Expansion: Extend the analysis to encompass additional geographic regions or market segments to gain insights into regional variations in car pricing and sales dynamics.

4. Integration with External Data Sources: Integrate data from external sources such as social media, review platforms, and industry reports to enrich the analysis and gain deeper insights into consumer sentiment and market trends.

ANNEXURE

References

[1]. no. 22, pp. 12 693–12 700, 2018. [12] E. Gegic, B. Isakovic, D. Keco, Z. Masetic, and J. Kevric, ―Car price prediction using machine learning techniques,‖ 2019.

[2]. N. Pal, P. Arora, P. Kohli, D. Sundararaman, and S. S. Palakurthy, ―How much is my car worth? a methodology for predicting used cars prices using random forest,‖ in Future of Information and Communication Conference. Springer, 2018, pp. 413–422.

[3]. R. Ragupathy and L. Phaneendra Maguluri, ―Comparative analysis of machine learning algorithms on social media test,‖ International Journal of Engineering and Technology(UAE), vol. 7, pp. 284–290, 03 2018.

[4]. Kaggle Datasets: https://www.kaggle.com/datasets

[5]. Python Programming Language: https://www.python.org/

[6]. Brownlee, J. (2019). Machine Learning Mastery: A Gentle Introduction to Scikit-Learn. [Online]: https://machinelearningmastery.com/a-gentle-introduction-to-scikit-learn-a-python-machine-learning-library/

[7]. McKinney, W. (2018). Python for Data Analysis. O'Reilly Media.

[8]. Microsoft Power BI Documentation Official documentation provided by Microsoft covering various aspects of Power BI, including tutorials, guides, and reference materials. https://docs.microsoft.com/en-us/power-bi/

[9]. Power BI Tutorial: A Comprehensive Guide to Get Started. Towards Data Science. https://towardsdatascience.com/tagged/power-bi-tutorials