AUTOMOBILE PRICE PREDICTION USING MACHINE LEARNING

A PROJECT REPORT

Submitted by

AKASHKUMAR.A.H [REGISTER NO:211422104032]

ALWIN.M [REGISTER NO: 211422104042]

ADHITHYAKUMAR.A.S [REGISTER NO: 211422104020]

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE,

CHENNAI- 600123.

(An Autonomous Institution Affiliated to Anna University, Chennai)

OCTOBER 2024

BONAFIDE CERTIFICATE

Certified that this project report **"AUTOMOBILE PRICE PREDICTION"** is the bonafide work of "**AKASHKUMAR.A.H (211422104032), ALWIN.M (211422104042), ADHITHYAKUMAR.A.S (211422104020) "** who carried the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr.L.JABASHEELA, M.E.,Ph.D.,	**DR.V.SATHIYA**
HEAD OF THE DEPARTMENT	**SUPERVISOR**
DEPARTMENT OF CSE, PANIMALAR ENGINEERING COLLEGE, NASARATHPETTAI, POONAMALLEE, CHENNAI-600 123.	DEPARTMENT OF CSE, PANIMALAR ENGINEERING NASARATHPETTAI, POONAMALLEE, CHENNAI-600 123.

Certified that the above candidates were examined in the End Semester Project Viva-Voce Examination held on...........................

INTERNAL EXAMINER	**EXTERNAL EXAMINER**

DECLARATION BY THE STUDENT

We **AKASHKUMAR.A.H (211422104032), ALWIN.M (211422104042), ADHITHYAKUMAR.A.S (211422104020)"** hereby declare that this project report titled "**AUTOMOBILE PRICE PREDICTION USING MACHINE LEARNING**", under the guidance of **Dr.V.SATHIYA.,** is the original work done by us and we have not plagiarized or submitted to any other degree in any university by us.

1. AKASHKUMAR.A.H

2. ALWIN.M

3. ADHITHYAKUMAR.A.S

ACKNOWLEDGEMENT

We would like to express our deep gratitude to our respected Secretary and Correspondent **Dr.P.CHINNADURAI, M.A., Ph.D.** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We express our sincere and hearty thanks to our Directors **Tmt.C.VIJAYARAJESWARI**, **Dr.C.SAKTHI KUMAR,M.E.,Ph.D** and **Dr.SARANYASREE SAKTHI KUMAR B.E.,M.B.A.,Ph.D.,** for providing us with the necessary facilities to undertake this project.

We also express our gratitude to our Principal **Dr.K.Mani, M.E., Ph.D.** who facilitated us in completing the project.

We thank the Head of the CSE Department, **Dr.L.JABASHEELA, M.E.,Ph.D.,** for the support extended throughout the project.

We would like to thank my Project Guide **Dr.V.SATHIYA.,** and all the faculty members of the Department of CSE for their advice and encouragement for the successful completion of the project.

AKASHKUMAR.A.H

ALWIN.M

ADHITHYAKUMAR.A.S

ABSTRACT

The prediction of automobile prices is a critical task in the automotive industry, user to find the price for their needed product and dealership pricing strategies. This study aims to develop a machine learning-based model to predict automobile prices based on a range of features, including technical specifications, brand, mileage, age, seller type, fuel type, engine, maximum power and market conditions. By leveraging a dataset of historical car prices and vehicle attributes, regression algorithm linear regression is employed and compared to identify the best performing model. The dataset was divided and modified to fit the regression, thus ensure the performance of the regression. Feature engineering and data preprocessing techniques, including handling missing data and normalization, are also applied to enhance model accuracy. It demonstrates the predictive power of the model, enabling more precise pricing in both new and used car markets. The findings of this research have potential applications in car dealerships, online automotive platforms, and for consumers seeking fair market values for their vehicles.

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

1.1 OVERVIEW

Automobile Price Prediction is the process of using machine learning or statistical models to predict a certain value for the automobile with regards to price in the market. Accurate price forecasting can greatly help out many stakeholders including the manufacturers, dealership, and consumers. This problem typically involves historical data containing different features (attributes) describing the automobile, such as: make, model, type, fuel, manufactured year, seller type, and many others in order to estimate the price.

This report provides the development of an ML-based approach for predicting the price of automobiles. As the use of machine learning techniques continues to advance, we can create predictive models to provide precise and data-driven predictions of prices. The crucial Steps in Automobile Price Forecasting: (i) Data Collection- Some public data sets, such as automobile data available at the UCI Machine Learning Repository, Manufacturer information and historical data (ii) Feature Engineering- which involves choosing a wide range of features (attributes) about the automobile, like make and model, type of engine, kind of fuel, year of manufacturer, type of seller, etc. to predict a price. (iii) Data Preprocessing- raw data must be cleaned and preprocessed, for example, Filling missing values or dropping incomplete records, Converting non-numeric data like car make, body type into numeric form using label encoding or one-hot encoding, Some features like maximum power or engine size might require to be scaled to have the same scale in models (iv) Model Building - Linear Regression is appropriate for understanding how the price of the automobile is related to other features to forecast price. (v) Prediction- After which the trained model can predict prices for new data in terms of input features. (vi) Deployment - Once the model is built and tested, it could be placed in front of the users through APIs or web applications, which will take in car features as input and give an estimated price.

1.2 PROBLEM DEFINITION

It would be a problem of building a machine learning model to predict the price of a car, considering features like make, model, year, mileage, and other attributes in engine specifications. This is a regression problem, where a continuous numerical value-the car's price- estimates the output based on historical data. The model would be trained on features such as car type, fuel type, horsepower, and luxury features to learn how to predict prices with a high degree of precision. Issues comprise missing data, feature scaling, multicollinearity between attributes, and outliers, which may affect predictions in biased ways. The idea would be to create a model that is generalizable-to take an unseen input from a user with a car's specification, it would give them an estimated price. This solution will prove helpful to car buyers and sellers when trying to approximate the cost of a car.

CHAPTER 2

LITERATURE SURVEY

No	Paper Ti	Authors	Year	Techniques Used	Dataset	Key Findings	Accuracy/Results
1	Car Price Prediction Using Machine Learning	Mittal, M., & Jain, S.	2020	Linear Regression, Decision Trees, Random Forest, XGBoost	Kaggle Car Dataset	XGBoost provided the most accurate price predictions due to its ability to handle non-linear data and feature interactions.	XGBoost RMSE = 2150
2	Used Car Price Prediction with Machine Learning	Ibrahim, N., & Victor, N.	2019	Random Forest, Gradient Boosting, Linear Regression	Used car data from Kaggle	Random Forest and Gradient Boosting performed better than Linear Regression, with Random Forest showing higher accuracy.	Random Forest RMSE = 1700
3	Predicting Used Car Prices Using Machine Learning	Zhang, Q., & Wang, L.	2019	Artificial Neural Networks (ANN), Linear Regression	Used car sales data from China	ANN captured complex relationships and interactions between features better than Linear Regression.	ANN $R^2 = 0.82$

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Car price prediction is quite widely used in both research and commercial applications. Models ranging from traditional machine learning models to sophisticated AI-driven platforms have been proposed for pricing car correctly. Such systems primarily depend upon historical information gathered from numerous sources, and most also employ listings of cars, sales records, as well as web-scraped information. Some examples of the existing systems currently widely used for automobile price predication: (i) Simple Machine Learning Systems- Linear Regression Model, Decision Trees & Random Forests, Support Vector Machines (SVM) (i) Sophisticated Machine Learning Systems- Gradient Boosting Models (GBM, XGBoost, LightGBM), Neural Networks and Deep Learning (iii) Online Websites and Mobile Apps- Kelley Blue Book (KBB), Edmunds, TrueCar (iv) AI-based Systems- CarGurus, AutoTrader.

3.2 PROPOSED SYSTEM

In this system approach, the car price prediction system will use advanced machine learning techniques to forecast more accurate, scalable, and user-friendly predictions of new and used cars. To do this, it's going to build upon existing methodologies by making use of real-time data ingestion, feature rich in analysis and explainability, all together with modern practices of deployment for it to reach out to a wider audience like buyers, sellers, and dealerships.

3.3 FEASIBILITY STUDY

The purpose of carrying out a feasibility study is not only to arrive at the solution but also to have an idea of the extent. In the process of the study, the problem definition becomes clear, and aspects of the problem to be included in the system are determined. Hence, benefits are estimated with more accuracy during this stage.

3.4 Technical Feasibility

Technical feasibility evaluates the hardware requirements, software technology, available personnel etc., as per the requirements it provides sufficient memory to hold and process.

1) Machine Learning algorithm - Linear Regression
2) Visual Studio Code
3) Google Drive
4) IDE: Google Colab

3.5 SOFTWARE REQUIREMENTS

- Programming language: Python
- Technology: Machine Learning
- Operating System: Windows 7
- Tools: Visual Studios Code, Google Collab

CHAPTER 4

SYSTEM DESIGN

4.1 FLOW DIAGRAM

This project requires a dataset which have both images and their caption. The dataset should be able to train the image captioning model.

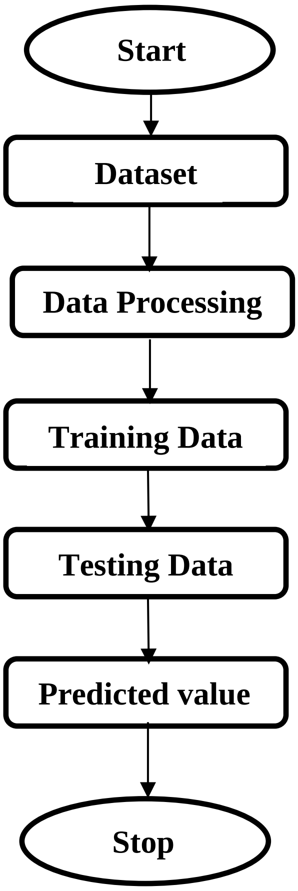


Fig. 4.1 Working flow of the model

4.2 DATASET

Data for automobile price prediction must encompass several features describing characteristics of cars that can influence their prices. The dataset includes 8100 car details encompassed within the following attributes: Price, Year, Milage, Engine, Maximum power, Number of seats, Brand, Fuel Type, Model, and type of Seller.

4.3 DATA PROCESSING

In the process of preparing raw data to train machine learning models, data preparation is a crucial task which should be clean and formatted well and must be appropriate for prediction. In the Automobile price prediction context, the phase of data preparation contains a plethora of tasks such as those dealing with missing value, encoding categorical variables, normalization of numerical data, and feature engineering. A properly processed data is fundamental to ensure that the machine learning model achieves high accuracy as well as generalization to predict car prices.

4.4 TRAINING THE DATA

Training the data is one of the important steps that come while building a model for automobile price prediction. Once we have cleaned and processed our data and then split it into training and test sets, it gets passed to machine learning algorithms that are going to learn patterns between its features - like mileage, year, size of engine - and target variables, which is price. Linear Regression is probably the simplest and most interpretable machine learning algorithm used to predict a continuous target variable like automobile price. In this case, of course, while predicting car prices, a linear regression models the relationship between the input features (like mileage, age, engine size, etc.) and the target variable - 'price' - as a linear combination of these features.

4.5 TESTING AND PREDICTING

Then, the performance of the model on the test set has to be analyzed, which it was not used during training. This is to understand how well it generalizes to unseen data. This also aims at avoiding overfitting - that phenomenon whereby a model performs well in training data but very poorly on new data. Once the model's performance on the test set is satisfactory, it can be used to predict prices for new, unseen data.

It is a critical phase of the life cycle of an automobile price prediction model. The testing stage ensures the accuracy and generalization of the model to a yet unseen dataset; while the predicting step allows it to be applied to real-world problems and predict car prices for any new dataset. It calls for continuous monitoring of the model followed by its retraining so that it stays relevant in this changing market.

CHAPTER 5

SYSTEM IMPLEMENTATION

5.1 Project File Structure

1. **Carsdetails.csv –** It is the Dataset SpreadSheet format which consist of total 8100 cars details which includes Price, Year, Milage, Engine, Maximum power, Number of seats, Brand, Fuel Type, Model, Seller type.
2. **Model.pkl –** It consist of our trained model of the Carsdetails dataset.
3. **App.py –** This code is part of a Streamlit-based web application that predicts car prices using a pre-trained machine learning model
4. **Training.ipynb –** It is the interactive Python notebook where you train and run the code

5.2 Building the Project

Initially it is started by initializing the Colab notebook server by typing Colab in the console of your project folder. It will open up the interactive Python notebook where you can train your dataset. Create a Python3 notebook and name it as Training.ipynb.

5.3 Getting Data

The **Carsdetails.csv** contains of all datas of cars includes Price, Year, Milage, Engine, Maximum power, Number of seats, Brand, Fuel Type, Model, Seller type.

File Home Insert Page Layout Formulas Data Review View Help

A1 name

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	name	year	selling_pri	km_driven	fuel	seller_type	transmissi	owner	mileage	engine	max_powe	torque	seats
2	Maruti Swi	2014	450000	145500	Diesel	Individual	Manual	First Owne	23.4 kmpl	1248 CC	74 bhp	190Nm@	5
3	Skoda Rap	2014	370000	120000	Diesel	Individual	Manual	Second Ov	21.14 kmp	1498 CC	103.52 bh	250Nm@	5
4	Honda City	2006	158000	140000	Petrol	Individual	Manual	Third Own	17.7 kmpl	1497 CC	78 bhp	12.7@ 2,7	5
5	Hyundai i2	2010	225000	127000	Diesel	Individual	Manual	First Owne	23.0 kmpl	1396 CC	90 bhp	22.4 kgm a	5
6	Maruti Swi	2007	130000	120000	Petrol	Individual	Manual	First Owne	16.1 kmpl	1298 CC	88.2 bhp	11.5@ 4,5	5
7	Hyundai Xc	2017	440000	45000	Petrol	Individual	Manual	First Owne	20.14 kmp	1197 CC	81.86 bhp	113.75nm	5
8	Maruti Wa	2007	96000	175000	LPG	Individual	Manual	First Owne	17.3 km/k	1061 CC	57.5 bhp	7.8@ 4,50	5
9	Maruti 80	2001	45000	5000	Petrol	Individual	Manual	Second Ov	16.1 kmpl	796 CC	37 bhp	59Nm@ 2	4
10	Toyota Eti	2011	350000	90000	Diesel	Individual	Manual	First Owne	23.59 kmp	1364 CC	67.1 bhp	170Nm@	5
11	Ford Figo	2013	200000	169000	Diesel	Individual	Manual	First Owne	20.0 kmpl	1399 CC	68.1 bhp	160Nm@	5
12	Renault Du	2014	500000	68000	Diesel	Individual	Manual	Second Ov	19.01 kmp	1461 CC	108.45 bh	248Nm@	5
13	Maruti Zer	2005	92000	100000	Petrol	Individual	Manual	Second Ov	17.3 kmpl	993 CC	60 bhp	78Nm@ 4	5
14	Maruti Swi	2009	280000	140000	Diesel	Individual	Manual	Second Ov	19.3 kmpl	1248 CC	73.9 bhp	190Nm@	5
15	Maruti Swi	2007	200000	80000	Petrol	Individual	Manual	Second Owner					
16	Maruti Wa	2009	180000	90000	Petrol	Individual	Manual	Second Ov	18.9 kmpl	1061 CC	67 bhp	84Nm@ 3	5
17	Mahindra	2016	400000	40000	Petrol	Individual	Manual	First Owne	18.15 kmp	1198 CC	82 bhp	115Nm@	5
18	Maruti Erti	2016	778000	70000	Diesel	Individual	Manual	Second Ov	24.52 kmp	1248 CC	88.5 bhp	200Nm@	7
19	Hyundai i2	2012	500000	53000	Diesel	Individual	Manual	Second Ov	23.0 kmpl	1396 CC	90 bhp	22.4 kgm a	5
20	Maruti Alt	2002	150000	80000	Petrol	Individual	Manual	Second Ov	19.7 kmpl	796 CC	46.3 bhp	62Nm@ 3	5
21	Hyundai i2	2016	680000	100000	Diesel	Individual	Manual	First Owne	22.54 kmp	1396 CC	88.73 bhp	219.7Nm@	5
22	Mahindra	2011	174000	100000	Diesel	Individual	Manual	Second Ov	21.0 kmpl	1461 CC	64.1 bhp	160Nm@	5
23	Honda WR	2017	950000	50000	Diesel	Individual	Manual	First Owne	25.5 kmpl	1498 CC	98.6 bhp	200Nm@	5
24	Maruti Swi	2015	525000	40000	Diesel	Individual	Manual	First Owne	26.59 kmp	1248 CC	74 bhp	190Nm@	5
25	Maruti SX4	2012	600000	72000	Diesel	Individual	Manual	First Owne	21.5 kmpl	1248 CC	88.8 bhp	200Nm@	5
26	Tata Tigor	2018	500000	35000	Petrol	Individual	Manual	First Owne	20.3 kmpl	1199 CC	83.81 bhp	114Nm@	5
27	Maruti Bal	2016	575000	45000	Petrol	Individual	Manual	First Owne	21.4 kmpl	1197 CC	83.1 bhp	115Nm@	5
28	Maruti Alt	2017	275000	28000	Petrol	Individual	Manual	First Owne	24.7 kmpl	796 CC	47.3 bhp	69Nm@ 3	5
29	Chevrolet	2013	300000	70000	Diesel	Individual	Manual	First Owne	18.2 kmpl	1248 CC	73.8 bhp	172.5Nm@	7
30	Maruti Wa	2009	220000	120000	Petrol	Individual	Manual	First Owne	18.9 kmpl	1061 CC	67 bhp	84Nm@ 3	5
31	Maruti Om	2018	254999	25000	Petrol	Individual	Manual	First Owne	16.8 kmpl	796 CC	34.2 bhp	59Nm@ 2	8
32	Maruti Vit	2017	670000	70000	Diesel	Individual	Manual	First Owne	24.3 kmpl	1248 CC	88.5 bhp	200Nm@	5
33	Fiat Palio 1	2003	70000	50000	Petrol	Individual	Manual	Second Owner					
34	Maruti Om	2012	150000	35000	Petrol	Individual	Manual	Second Ov	14.0 kmpl	796 CC	35 bhp	6.1kgm@	5
35	Hyundai i2	2018	730000	2388	Petrol	Individual	Manual	First Owne	18.6 kmpl	1197 CC	81.83 bhp	114.7Nm@	5

Fig. 5.3.1 Dataset

5.4 ALGORITHM

Data Preprocessing

Import necessary libraries

- pandas, numpy, scikit-learn, pickle, streamlit.
- Load the car dataset from Cardetails.csv.
- Check for missing values and drop rows with missing data.
    - Drop any duplicates to clean the dataset.
- Extract features from columns:
    - Extract car brand names from the name column.
    - Convert the extracted brand names to numeric values.
- Clean numerical columns (mileage, engine, max_power):
    - Strip and convert values to float.
    - Replace empty values with 0.
- Encode categorical data:
    - Replace categorical values in columns such as fuel, seller_type, transmission, owner with numerical values.
- Drop irrelevant columns like torque.

Model Building

- Split the data into input (X) and output (y), where X contains features and y contains selling_price.
- Train-test split:
    - Divide the data into training and testing sets using train_test_split.
- Train a linear regression model on the training data.
- Evaluate the model by making predictions on the test set.
- Save the trained model using pickle.
- Web Application Deployment
- Create a Streamlit application:
    - Define the application header for "Car Price Prediction ML Model".

- Load car dataset and clean/preprocess similar to the training.
- Build UI components using streamlit:
  - Dropdowns and sliders for users to select car attributes (brand, year, km driven, fuel, etc.).
- When user clicks "Predict":
  - Collect input data.
  - Replace categorical user inputs with numerical values similar to preprocessing.
  - Use the trained model to predict car price.
- Display the predicted car price on the web page.

5.5 Training The Dataset And Predicting The Data

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
cars_data = pd.read_csv('Cardetails.csv')
cars_data.head()
cars_data.shape
```

```
(8128, 13)
```

```
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
```

```
torque          222
seats           221
    dtype: int 64
```

```
cars_data.dropna(inplace=True)
cars_data.shape
cars_data.duplicated().sum()
cars_data.drop_duplicates(inplace=True)
cars_data.shape
cars_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 6717 entries, 0 to 8125
Data columns (total 13 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   name           6717 non-null   object
 1   year           6717 non-null   int64
 2   selling_price  6717 non-null   int64
 3   km_driven      6717 non-null   int64
 4   fuel           6717 non-null   object
 5   seller_type    6717 non-null   object
 6   transmission   6717 non-null   object
 7   owner          6717 non-null   object
 8   mileage        6717 non-null   object
 9   engine         6717 non-null   object
 10  max_power      6717 non-null   object
 11  torque         6717 non-null   object
 12  seats          6717 non-null   float64
dtypes: float64(1), int64(3), object(9)
memory usage: 734.7+ KB
```

```
for col in cars_data.columns:
    print('Unique values of ' + col)
    print(cars_data[col].unique())
    print("=====================")
```

```
Unique values of name
['Maruti Swift Dzire VDI' 'Skoda Rapid 1.5 TDI Ambition'
 'Honda City 2017-2020 EXi' ... 'Tata Nexon 1.5 Revotorq XT'
 'Ford Freestyle Titanium Plus Diesel BSIV'
```

```
 'Toyota Innova 2.5 GX (Diesel) 8 Seater BS IV']
====================
Unique values of year
[2014 2006 2010 2007 2017 2001 2011 2013 2005 2009 2016 2012 2002 2015
 2018 2019 2008 2020 1999 2000 2003 2004 1994 1998 1997 1995 1996]
====================
Unique values of selling_price
[  450000   370000   158000   225000   130000   440000    96000    45000
   350000   200000   500000    92000   280000   180000   400000   778000
   150000   680000   174000   950000   525000   600000   575000   275000
   300000   220000   254999   670000   730000   650000   330000   366000
  1149000   425000  2100000   925000   675000   819999   390000  1500000
   700000  1450000  1090000   850000  1650000  1750000  1590000  1689999
  1425000   265000   190000   630000   540000   448000   745000  1025000
   235000  1700000  1200000   610000  2500000   484999   315000   290000
   455000   351000   535000   175000   565000   120000   725000   185000
   615000   270000   625000   866000   375000   522000   451999   475000
   780000   595000  1140000   360000   105000   135000   690000  3975000
  5150000  3200000  4100000  4500000  6000000  3790000  5800000  1864999
  2700000   795000  3400000  2650000  5850000   975000   805000  2625000
   811000   550000   645000  2550000   599000   875000   894999   340000
...
====================
Unique values of seats
[ 5.  4.  7.  8.  6.  9. 10. 14.  2.]
====================
```

```
def get_brand_name(car_name):
  car_name = car_name.split(' ')[0]
  return car_name.strip()
def clean_data(value):
  value = value.split(' ')[0]
  value = value.strip()
  if value == '':
    value = 0
  return float(value)
get_brand_name('Maruti  Swift Dzire VDI')
cars_data['name'] = cars_data['name'].apply(get_brand_name)
cars_data['name'].unique()
```

```
array(['Maruti', 'Skoda', 'Honda', 'Hyundai', 'Toyota', 'Ford', 'Mahindra',      'Tata',
```

```
'Chevrolet', 'Datsun', 'Jeep', 'Mercedes-Benz',
 'Mitsubishi', 'Audi', 'Volkswagen', 'BMW', 'Nissan', 'Lexus',
 'Jaguar', 'Land', 'MG', 'Volvo', 'Daewoo', 'Kia', 'Fiat', 'Force',
 'Ambassador', 'Ashok', 'Isuzu', 'Opel'], dtype=object)
```

```
cars_data['mileage'] = cars_data['mileage'].apply(clean_data)
cars_data['max_power'] = cars_data['max_power'].apply(clean_data)
cars_data['engine'] = cars_data['engine'].apply(clean_data)


for col in cars_data.columns:
  print('Unique values of ' + col)
  print(cars_data[col].unique())
  print("=====================")
```

```
cars_data['name'].replace(['Maruti', 'Skoda', 'Honda', 'Hyundai', 'Toyota', 'Ford', 'Renault',
 'Mahindra', 'Tata', 'Chevrolet', 'Datsun', 'Jeep', 'Mercedes-Benz',
 'Mitsubishi', 'Audi', 'Volkswagen', 'BMW', 'Nissan', 'Lexus',
 'Jaguar', 'Land', 'MG', 'Volvo', 'Daewoo', 'Kia', 'Fiat', 'Force',
 'Ambassador', 'Ashok', 'Isuzu', 'Opel'],[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31],inplace = True)
```

```
cars_data['transmission'].unique()
cars_data['transmission'].replace(['Manual', 'Automatic'],[1,2],inplace = True)
cars_data['seller_type'].unique()
cars_data['seller_type'].replace(['Individual', 'Dealer', 'Trustmark 
Dealer'],[1,2,3],inplace = True)
cars_data['fuel'].unique()
cars_data['fuel'].replace(['Diesel', 'Petrol', 'LPG', 'CNG'],[1,2,3,4],inplace = True)
cars_data.reset_index(inplace=True)
cars_data['owner'].unique()
cars_data['owner'].replace(['First Owner', 'Second Owner', 'Third Owner',
       'Fourth & Above Owner', 'Test Drive Car'],[1,2,3,4,5],inplace = True)
del cars_data['torque']
cars_data.head()
cars_data.drop(columns=['index'],inplace=True)
input_data = cars_data.drop(columns=['selling_price'])
output_data = cars_data['selling_price']
```

```
x_train, x_test, y_train, y_test = train_test_split(input_data, output_data, test_size=0.2)
model = LinearRegression()
model.fit(x_train, y_train)
LinearRegression
LinearRegression()
predict = model.predict(x_test)
predict
```

```
array([828119.64756832, 935339.70077798, 688027.90744548, ...,
298962.40939222, 584343.83636625, 307302.57692838])
```

```
x_train.head(1)
input_data_model =
pd.DataFrame([[4,2023,5900,1,2,2,1,17.01,1582.0,126.2,5.0]],columns=['name', 'year',
'km_driven', 'fuel', 'seller_type', 'transmission', 'owner', 'mileage', 'engine', 'max_power',
'seats'])
input_data_model
model.predict(input_data_model)
```

```
array([1633722.07865916])
```

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

DEPLOYMENT-app.py

```
import pandas as pd
import numpy as np
import pickle as pk
import streamlit as st
model = pk.load(open('model.pkl','rb'))
st.header('Car Price Prediction ML Model')
cars_data = pd.read_csv('Cardetails.csv')
def get_brand_name(car_name):
    car_name = car_name.split(' ')[0]
    return car_name.strip()
cars_data['name'] = cars_data['name'].apply(get_brand_name)
name = st.selectbox('Select Car Brand', cars_data['name'].unique())
year = st.slider('Car Manufactured Year', 1994,2024)
```

```
  km_driven = st.slider('No of kms Driven', 11,200000)
  fuel = st.selectbox('Fuel type', cars_data['fuel'].unique())
  seller_type = st.selectbox('Seller  type', cars_data['seller_type'].unique())
  transmission = st.selectbox('Transmission type', cars_data['transmission'].unique())
  owner = st.selectbox('Seller  type', cars_data['owner'].unique())
  mileage = st.slider('Car Mileage', 10,40)
  engine = st.slider('Engine CC', 700,5000)
  max_power = st.slider('Max Power', 0,200)
  seats = st.slider('No of Seats', 5,10)
  if st.button("Predict"):
      input_data_model = pd.DataFrame(
      [[name,year,km_driven,fuel,seller_type,transmission,owner,mileage,engine,max_
      power,seats]],
      columns=['name','year','km_driven','fuel','seller_type','transmission','owner','mileage',
      'engine','max_power','seats'])
      input_data_model['owner'].replace(['First Owner', 'Second Owner', 'Third Owner',
        'Fourth & Above Owner', 'Test Drive Car'],  [1,2,3,4,5], inplace=True)
      input_data_model['fuel'].replace(['Diesel', 'Petrol', 'LPG', 'CNG'],[1,2,3,4], inplace=True)
      input_data_model['seller_type'].replace(['Individual', 'Dealer', 'Trustmark Dealer'],[1,2,3],
  inplace=True)
      input_data_model['transmission'].replace(['Manual', 'Automatic'],[1,2], inplace=True)
      input_data_model['name'].replace(['Maruti', 'Skoda', 'Honda', 'Hyundai', 'Toyota', 'Ford', 'Renault',
        'Mahindra', 'Tata', 'Chevrolet', 'Datsun', 'Jeep', 'Mercedes-Benz',
        'Mitsubishi', 'Audi', 'Volkswagen', 'BMW', 'Nissan', 'Lexus',
        'Jaguar', 'Land', 'MG', 'Volvo', 'Daewoo', 'Kia', 'Fiat', 'Force',
        'Ambassador', 'Ashok', 'Isuzu', 'Opel'],
                        [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,3
  1], inplace=True)
car_price = model.predict(input_data_model)
st.markdown('Car Price is going to be '+ str(car_price[0])
```

CHAPTER 6

PERFORMANCE ANALYSIS

6.1 Linear Regression

The simplest and easiest to interpret machine learning algorithms used for predicting a continuous target variable, like the automobile price, is Linear Regression. In the context of predicting car prices, linear regression models the relationship between the input features, such as mileage, age, engine size, etc., and the target variable of interest, price. It assumes that the relationship between the dependent variable-the price-and the independent variables is linear.

$$y=\beta 0+\beta 1x1+\beta 2x2+\cdots+\beta nxn$$

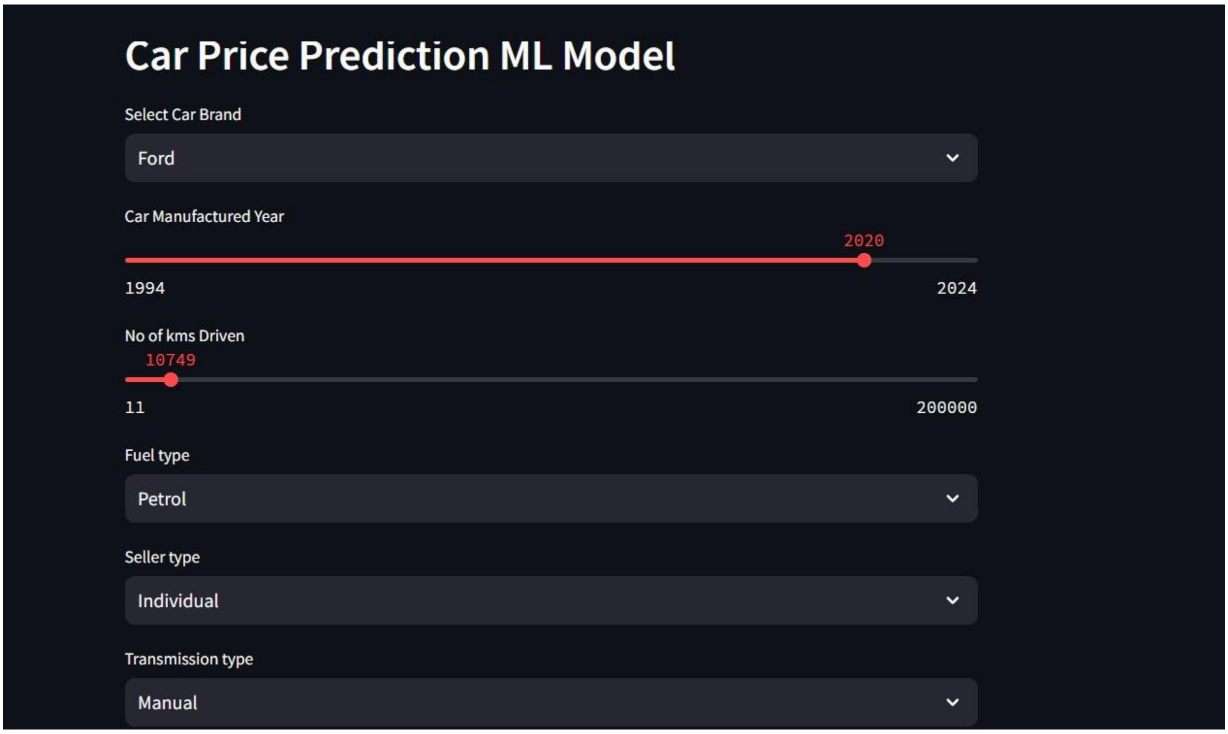
- y: Predicted car price (the dependent variable).
- x1, x2, ..., xn: The features (independent variables), such as mileage, age, engine size, etc.
- β0: The **intercept** (the base price of the car when all features are zero).
- β1, β2, ..., βn: The **coefficients** (slopes), representing the change in car price for a unit change in each feature.

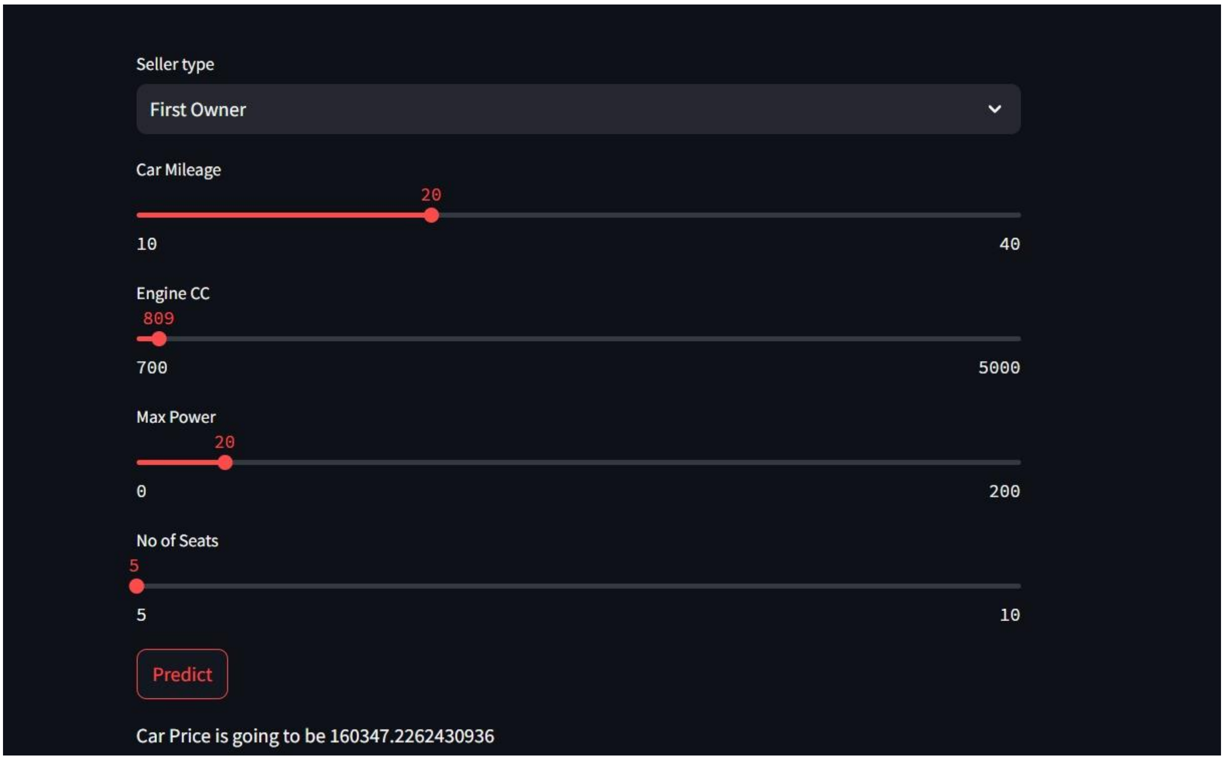
6.2 Regularization Techniques

The problem with these is that they may over-fit with the presence of too many features or multicollinearity, where values are highly correlated with independent variables. To avoid this, regularization techniques like Ridge and Lasso regression are used:

- **Ridge Regression (L2 Regularization):** Adds a penalty for large coefficients in the form of the sum of the squares of the coefficients.
- **Lasso Regression (L1 Regularization):** It introduces a penalty for large coefficients in the form of the sum of the absolute values of the coefficients, also leading to feature selection.

6.3 RESULT


Car Price Prediction ML Model
Select Car Brand
Ford
Car Manufactured Year
2020
1994
2024
No of kms Driven
10749
11
200000
Fuel type
Petrol
Seller type
Individual
Transmission type
Manual


Seller type
First Owner
Car Mileage
20
10
40
Engine CC
809
700
5000
Max Power
20
0
200
No of Seats
5
5
10
Predict
Car Price is going to be 160347.2262430936

CHAPTER 7

CONCLUSION

A practical, efficient and user-friendly solution for estimating the price of cars based on several key features is identified by a machine learning model of this system. This means the system would predict car prices by taking into account factors like car brand, year of manufacture, kilometers driven, fuel type, transmission, and its ownership status, among other relevant characteristics, all of which can be leveraged from historical data.

The new method was able to efficiently automate the complicated process of the estimation of car prices, which is one of the benefits both car sellers and buyers get in terms of decision-making, as well as dealers, through the immediate prediction of prices. As far as the integration of the model with Streamlit seems to open up an interface where the user will be required to input the details of the car that they are interested in acquiring. That is to say, the accuracy of predictions is highly dependent on the quality and breadth of the training data in addition to the capacity of this model to generalize to unseen data.

Further, retraining of the system with continuous updated input would bring improvements, in addition to learning new things introduced by changes in market trends, newer car models, and changing preferences by users. In a nutshell, the application will depict how machine learning can be very positively applied toward solving real-world problems within the automotive industry.

REFERENCE

[1] Pandey Abhishek, Rastogi Vanshika and Singh Sanika, Car's Selling Price Prediction using Random Forest Machine Learning Algorithm, March 2020.

[2] C. V. Narayana, C. L. Likhitha, S. Bademiya and K. Kusumanjali, "Machine Learning Techniques To Predict The Price Of Used Cars: Predictive Analytics in Retail Business", 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 1680-1687, 2021.

[3] Vehicle Price Prediction using SVM Techniques S.E.Viswapriya Durbaka Sai Sandeep Sharma Gandavarapu Sathya kiran International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 9, no. 8, June 2020, ISSN 2278–3075.

[4] C. Longani, S. Prasad Potharaju and S. Deore, "Price Prediction for Pre-Owned Cars Using Ensemble Machine Learning Techniques", Recent Trends in Intensive Computing, 2021.

[5] K. Samruddhi and R. Ashok Kumar, "Used Car Price Prediction using KNearest Neighbor Based Model", International Journal of Innovative Research in Applied Sciences and Engineering, vol. 4, no. 3, pp. 686-689, 2020.

[6] A. W. Laveena D'Costa, "Predicting true value of used car using multiple linear regression model", International Journal of Recent Technology and Engineering, vol. 8, pp. 42-45, 2020.

[7] A. W. Laveena D'Costa, "Predicting true value of used car using multiple linear regression model", International Journal of Recent Technology and Engineering, vol. 8, pp. 42-45, 2020.

[8] C. K. Puteria and L. N. Safitri, "Analysis of linear regression on used car sales in indonesia", Institute of Physics Publishing, vol. 1469, 2020.