A MAJOR PROJECT REPORT ON

CAR PRICE PREDICTION USING MACHINE LEARNING WITH PYTHON

Submitted to Kakatiya University

for the partial fulfillment of the requirements for the

award of the degree of

Bachelor of Science (Data Science)

Submitted by

<<Chiluka Abhinaya >>(086214182)

Under the guidance of

Mr. K. SRIDHAR
(Head of the Department of Computer Science)

VAAGDEVI DEGREE & P.G COLLEGE
(Affiliated to Kakaitya University)
Hanamkonda, Telangana, India 506001.

Viswambhara Educational Society Ph: 0870-2455188

VAAGDEVI DEGREE & P.G COLLEGE

Accredited by NAAC with 'A' Grade

(Approved by A.I.C.T.E., New Delhi & Affiliated to Kakatiya University)

2-2-457/A, Kishanpura, Hanamkonda – 506001, Warangal, T.S.

CERTIFICATE

This is to certify that **Chiluka Abhinaya** student of **Bachelor of Science (Data Science)**, Vaagdevi Degree & P.G College affiliated to Kakatiya University have successfully completed the project entitled **"CAR PRICE PREDICTION USING MACHINE LEARNING WITH PYTHON"** at Vaagdevi Degree & P.G College,Hanamkonda in partial fulfillment of requirement for the award of the**Bachelor of Science (Data Science)** from **Kakatiya University** is a record of bonafied work carried out by them under my supervision.

Head of the Department **Principal**

External Examiner

DECLARATION

I, the undersigned here by declare that the project **CAR PRICE PREDICTION USING MACHINE LEARNING WITH PYTHON** with special reference to *Rapid Technologies – Hyderabad*, developed and submitted by us to **KAKATIYA UNIVERSITY**, Hanamkonda in partial fulfillments for the award of degree of **Bachelor of Science (Data Science)** under the guidance of **Mr. K. SRIDHAR**, is our original work and implemented by us.

Place: Hanamkonda.

Date:

Chiluka Abhinaya

ACKNOWLEDGEMENT

I wish to take this opportunity to express my sincere gratitude and deep sense of respect to our beloved principal, **Dr. A. SHESHA CHALAM**, Vaagdevi Degree & P.G College, Hanamkonda for making me available all the required assistance and for his support and inspiration to carry out this work in the institute.

I express my heartfelt thanks to the Head of the Department of computer Science, **Mr. K. Sridhar** for providing me with necessary infrastructure and there by giving me freedom to carry out this project.

I am also thankful to **Mr. K. Sridhar, HoD of CS** for providing the excellent facilities, motivation and valuable guidance throughout the project work. With his co-operation and encouragement I completed the project work in time.

I owe an enormous debt of gratitude to **Mr. K. Sridhar,HoD of CS** for guiding me from the beginning through the end of the project with his intellectual advices and insightful suggestions. I truly value his consistent feedback on my progress, which was always constructive and encouraging and ultimately drove me to the right direction.

Finally, I express my thanks to all the faculty members for their co-operation in completing the project.

Last but not least I thank to my parents who inspired me always to do the best.

Chiluka Abhinaya

INDEX

ABSTRACT

The price of a car depends on a lot of factors like the goodwill of the brand of the car, features of the car, horsepower and the mileage it gives and many more. Car price prediction is one of the major research areas in machine learning.

One of the main areas of research in machine learning is the prediction of the price of cars. It is based on finance and the marketing domain. It is a major research topic in machine learning because the price of a car depends on many factors.

If one ignores the brand of the car, a car manufacturer primarily fixes the price of a car based on the features it can offer a customer. Later, the brand may raise the price depending on its goodwill, but the most important factors are what features a car gives you to add value to your life., We will walk through the task of training a car price prediction model with machine learning using the Python programming language.

This project is to find a multiple linear regression model by using Python Language from a given used car price data and predict a used car price on the basis of the test data.The data was from one of Kaggle's datasets

https://www.kaggle.com/nehalbirla/motorcycle-dataset

The selling price was the target variable and, other variables were used for features.

INTRODUCTION

Determining whether the listed price of a used car is a challenging task, due to the many factors that drive a used vehicle's price on the market.The focus of this project is developing machine learning models that can accurately predict the price of a used car based on its features, in order to make informed purchases. We implement and evaluate various learning methods on a dataset consisting of the sale prices of different makes and models . We will compare the performance of various machine learning algorithms like Linear Regression, Randomforest Regression, Gradient Boosting Regression, and choose the best out of it. Depending on various parameters we will determine the price of the car. Regression Algorithms are used because they provide us with continuous value as an output and not a categorized value because of which it will be possible to predict the actual price a car rather than the price range of a car. User Interface has also been developed which acquires input from any user and displays the Price of a car according to user's inputs.

PROBLEM STATEMENT

The used car market is a huge and important market for car manufacturers. The second-hand car market is also very likely linked to new car sales. Selling used cars at new car retail and handling lease returns and fleet returns from car rental companies require car manufacturers to be involved in the used car market.

Automakers face several problems in the used market. The deep mess in the world, the general problem of more people, increased competition from other manufacturers and the trend toward electronic cars are just some of the factors that make it difficult to sell used vehicles on the used car market, reducing sales margins. Automakers, therefore, require good decision support systems to maintain the profit of the used car business. A core component of such a system is a predictive model that estimates the selling price based on vehicle attributes and other factors. Although previous studies have explored statistical modelling of resale costs, few studies have attempted to predict resale costs with maximum accuracy to support decision making. As a result, the answers to the following questions are unclear:

i) how predictable are resale prices.

ii) the relative accuracy of various forecasting methods, and whether some methods are particularly effective.

iii) Given those market research agencies specialize in estimating residual values, does it makes sense for automakers to invest in their resale cost prediction models?

The purpose of this work is to provide more accurate answers to those questions. The present project comes under the Regression category. This project is all about predicting the used car's prices. In our day to life, everyone wants a car, but budget is the problem, so in this project build a model that will take certain parameters as arguments and result or predict the price of the car based on given parameters. This project's goals are to build a machine learning model which takes car features as input and predicts the cost of the reused car. Compare the most used machine learning regression models which give less error and predict the more accurate value of the price of the car.

SYSTEM ANALYSIS

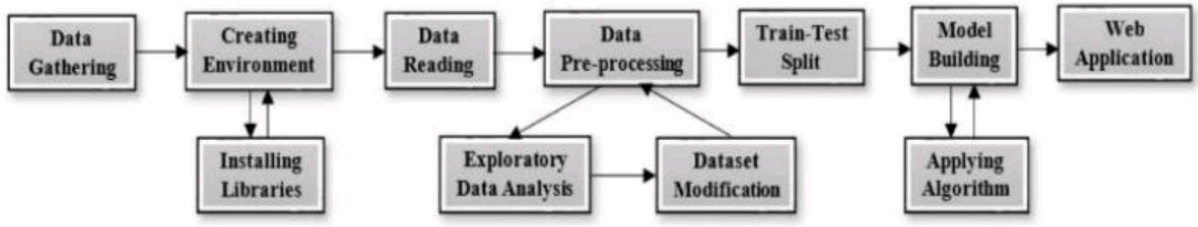
1.EXISTING SYSTEM

Existing system includes a process where a seller decides a price randomly and buyer has no idea about the car and it's value in the present day scenario.In fact,seller also has no idea about the car's existing value or the price he should be selling the car at.To overcome this problem we have developed a model which will be highly effective.

Machine learning Algorithms are used because they provide us with continuous value as an output and not a categorized value Because of which it will be possible to predict the actual price a car rather than the price range of a car.User Interface has also been developed which acquires input from any user and displays the Price of a car according to user's inputs.

2.PROPOSED SYSTEM

Fig:Work flow of study



TECHNOLOGY USED :

Python is mainly used in this project to implement machine learning algorithms since it contains a lot of built-in methods in the form of packaged libraries and modules. During project implementation, Python, Pandas, NumPy, Matplotlib, Seaborn, Scikit-Learn, Plotly, and Pickle libraries were used.

DATA GATHERING:

The source of the data is the web portal Kaggle.com, where vehicle data sets are provided by Cardekho for the sale and purchase of cars. The dataset contained the following features: car name, year, selling price, present or current price, kilometres driven, fuel type: diesel, petrol, or CNG (compressed natural gas), seller type: dealer or individual, transmission: automatic or manual, owner (number of previous owners).

CREATE ENVIRONMENT:

An environment is created using the Anaconda prompt. This environment would separate our project area from the other default environment (base) or other previously created environments. All the packages, libraries, and modules that we need can be manually installed in theenvironment created in this way, making it an advantageous step. In such an environment, we can make changes according to our needs

DATA READING:

The first step is to import and read the csv file for the research. The dataset is extensively examined in terms of null values, shape, columns, numerical and categorical features, dataset columns, unique values of each feature, data information, and so on

DATA PRE-PROCESSING :

Some of the data features were renamed for clarity (Present Price = Initial Price, Owner = Previous Owners), and some features that were not important for analysis were removed. In exploratory data analysis, we use statistical graphics and other visualisation techniques to describe the important aspects of data. Top Selling Vehicles, Year vs. Number of Available Vehicles, Selling Price vs. Initial Price, Vehicle Fuel Type, Transmission Type, Seller Type, Age, Selling Price v/s Age, Selling Price v/s Seller Type, Selling Price v/s Transmission, Selling Price v/s Fuel Type, Selling Price v/s Previous Owners, Initial Price vs Selling Price, Selling Price v/s Kilometers Driven, pairplot, heatmaps, and other visualisations are used to gain a better understanding of data. Following EDA, One Hot Encoding approach is used to deal with the dataset's categorical features.

After that, the dataset's correlation characteristics are generated and thoroughly analysed by visualising several plots. Then the features allocation of data is where the dependent feature (Selling Price) and independent features (Initial Price, Kilometers Driven, Previous Owners, Age, and so on) are then allocated for further processing.

TRAIN-TEST SPLIT :

Once the dependent and independent features have been assigned, we proceed with the splitting of the dataset into training and testing data. We use 80% of the data to train our model and 20% to test it.

MODEL BUILDING :

Following the Train-Test split, data modeling is complete, and the process of building the model begins. The model is defined, along with a few parameters, for future implementation. After the model is built, various algorithms are used to create the final results.

There are two phases in the build a model:

i. Training: The model is trained by using the data in the dataset and fits a model based on the model algorithm chosen accordingly.

ii. Testing: The model is provided with the inputs and is tested for its accuracy.

Afterwards, the data that is used to train the model or test it, has to be appropriate. The model is built to detect and predict the cost of a used car and good models must be selected.

SYSTEM REQUIREMENTS

Functional Requirements:

- The pre-processing image techniques are like image contrast, image enhancement,image cropping and removing noises if any is present.
- In this, the images Involves outsourcing background noise and removing reflections and masking portions of images.
- It describes the actual condition of the plant based on the pixel values

Non-Functional Requirements:

- **Scalability:** Our application is scalable to use any amount of possible dataset
- **Performance:** This application performs better with higher system configuration and even more better in cloud environment (Anaconda).
- **Reliability:** The data loaded and processed this environment is reliably processed, good in performance and no scope for data loss.
- **Availability:** The infrastructure we are using local is available until the system is available, for high availability it is better to choose cloud (Anaconda)
- **Usability:** Being showcased local the application is available to the system user and can be extended to multi-use in case we host this to cloud(Anaconda).

MODULES DESCRIPTION

In this project we have Two modules:

1) Data pre-processing Module.
2) Prediction Module.

1. Data Pre-processing Module:

In this module first gather the data(dataset) for prediction model. Data comes in all forms, most of it being very messy and unstructured. Perform effective data processing on the sample and extract the features. They rarely come ready to use. Datasets, large and small, come with a variety of issues- invalid fields, missing and additional values, and values that are informs different from the one can require. In order to bring it to workable or structured form, need to “clean” the data, and make it ready to use. Some common cleaning includes, removing unnecessary data, removing noise etc. In this case, the data has some days where some factors weren’t recorded.so that need to clean the data before applying it on our model.

2. Prediction Module:

Once the data is cleaned, In this module that cleaned data can be used as an input to our Linear regression model ,Random Forest Regression and Gradient Boosting Regression. This is done by plotting a line that fits our scatter plot the best, i.e., with the least errors. This gives value predictions, i.e., how much, by substituting the independent values in the line equation. It will use Scikit-learns linear regression model to train the dataset. Once the model is trained, that can give own inputs for the various columns to predict the used car price based on these attributes. Data Collection, Collect sufficient data samples and legitimate software samples. Train and Test Modelling, Split the data into train and test data Training will be used for training the model and Test data to check the performance. Feature Selection, further select the main features for classification.

SYSTEM ARCHITECTURE

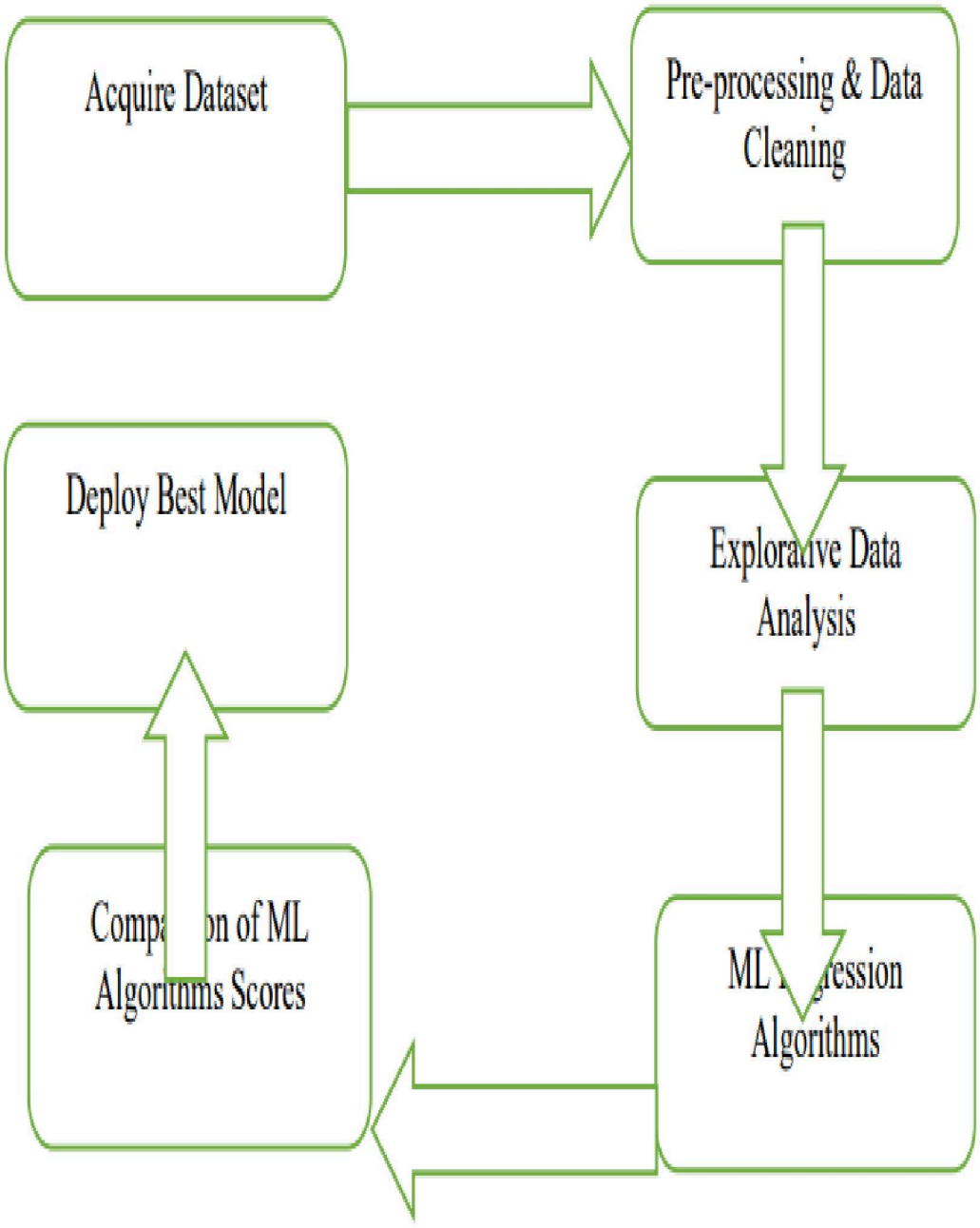


Fig: 1 Architecture of the Proposed System

DATASET

We collected cars dataset from https://www.kaggle.com/nehalbirla/motorcycle-dataset

1	Car_Name	Year	Selling_Pri	Present_P	Kms_Drive	Fuel_Type	Seller_Typ	Transmiss	Owner
2	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0
3	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0
4	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0
5	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0
6	swift	2014	4.6	6.87	42450	Diesel	Dealer	Manual	0
7	vitara brez	2018	9.25	9.83	2071	Diesel	Dealer	Manual	0
8	ciaz	2015	6.75	8.12	18796	Petrol	Dealer	Manual	0
9	s cross	2015	6.5	8.61	33429	Diesel	Dealer	Manual	0
10	ciaz	2016	8.75	8.89	20273	Diesel	Dealer	Manual	0
11	ciaz	2015	7.45	8.92	42367	Diesel	Dealer	Manual	0
12	alto 800	2017	2.85	3.6	2135	Petrol	Dealer	Manual	0
13	ciaz	2015	6.85	10.38	51000	Diesel	Dealer	Manual	0
14	ciaz	2015	7.5	9.94	15000	Petrol	Dealer	Automatic	0
15	ertiga	2015	6.1	7.71	26000	Petrol	Dealer	Manual	0

ALGORITHMS

1.LINEAR REGRESSION

Linear Regression is used to predict the value of a variable based on the value of another feature. The feature you want to predict is called the dependent variable. The label that is used to predict the value of another feature is called the independent variable. The LR equation is of the form $A = m + nB$, where B is the independent variable, A is the dependent variable, a is the intercept y, and n is the slope of the line.

2.RANDOM FOREST REGRESSION

Random Forest is a Supervised Learning Algorithm that employs the ensemble learning approach for classification and regression. Random forests are made up of trees that run parallel to each other and have no interaction while they develop. Random Forest is a meta-estimator that aggregates the outcomes of several predictions. It also aggregates numerous decision trees with certain modifications.

3.GRADIENT BOOSTING REGRESSION

This is a machine learning approach used to construct a prediction model for regression and classification problems. The prediction model generates an ensemble of weak prediction models, which are often decision trees. This method outperforms the random forest method in most cases.

EXPLORATORY DATA ANALYSIS

In this stage, we summarize the major characteristics of data using statistical graphics and other visualization tools. Various graphs and charts are plotted to gain a better understanding of the dataset and the relationships between its features.

The following bar graph depicts the number of vehicles of a certain age. cc

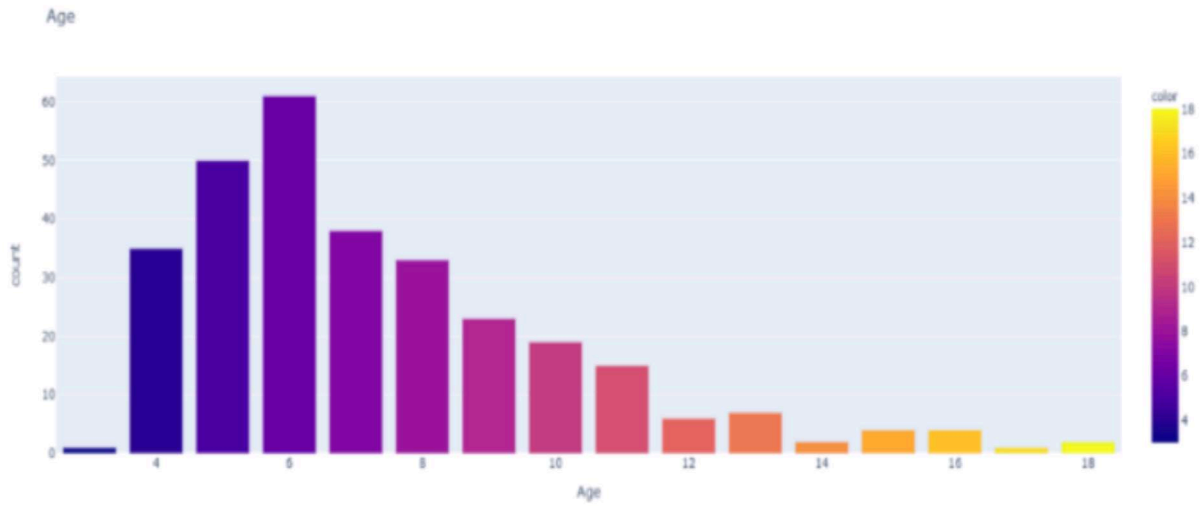


Fig: Count w.r.t Age

Vehicle count in relation to vehicle age

Comparison of each vehicle's selling price vs. age

The chart below depicts the selling price and age of a certain car. And it is easy to conclude that the selling price is high for a car of a young age.

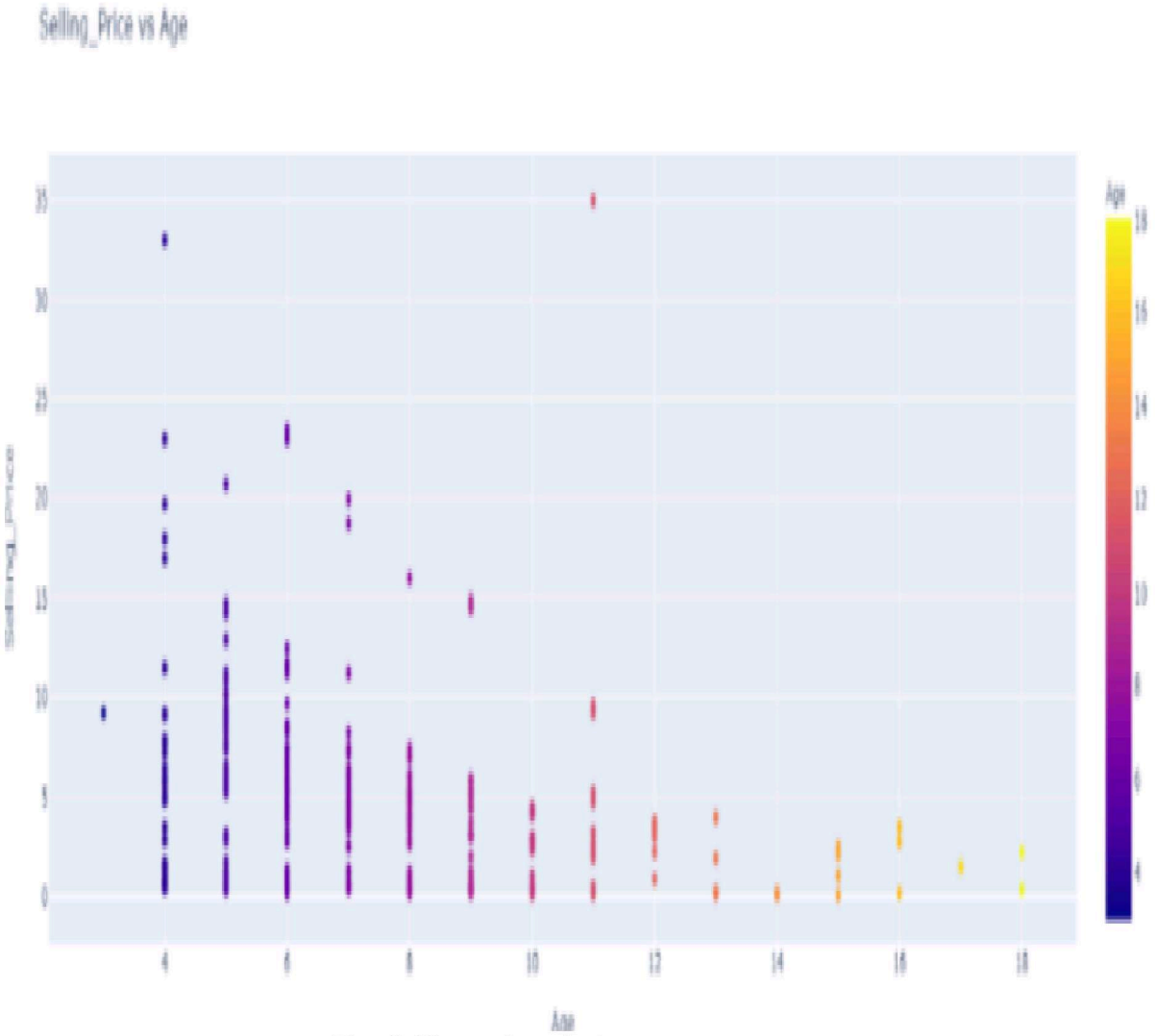


Fig: Selling price v/s Age

Comparison of Initial Price and Selling Price:

The graph below demonstrates the direct proportionality between Initial Price and Selling Price, which suggests that a higher initial price will result in a higher selling price.

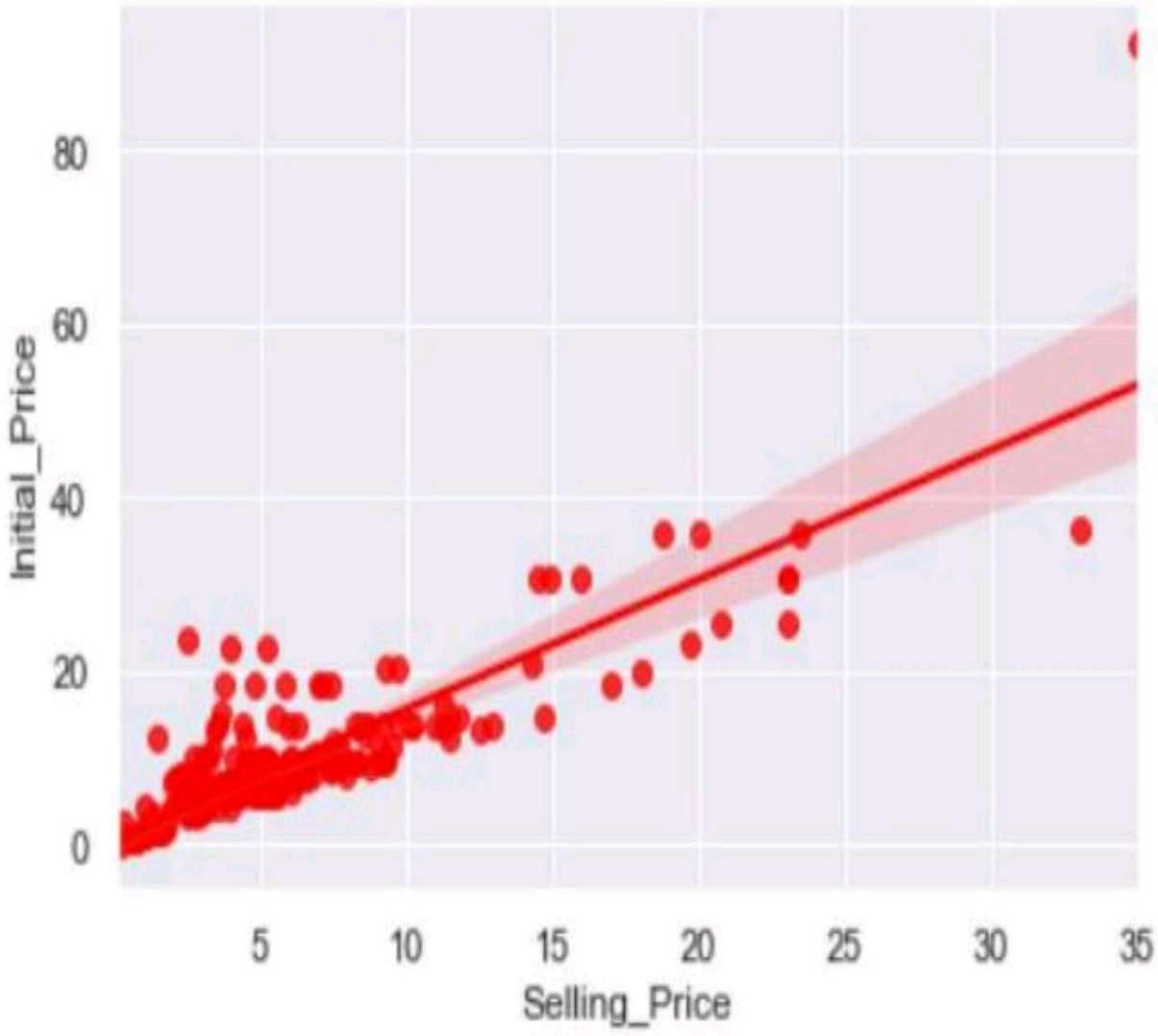


Fig: Initial price v/s Selling price

Comparision of Kilometers Driven vs. Selling Price:

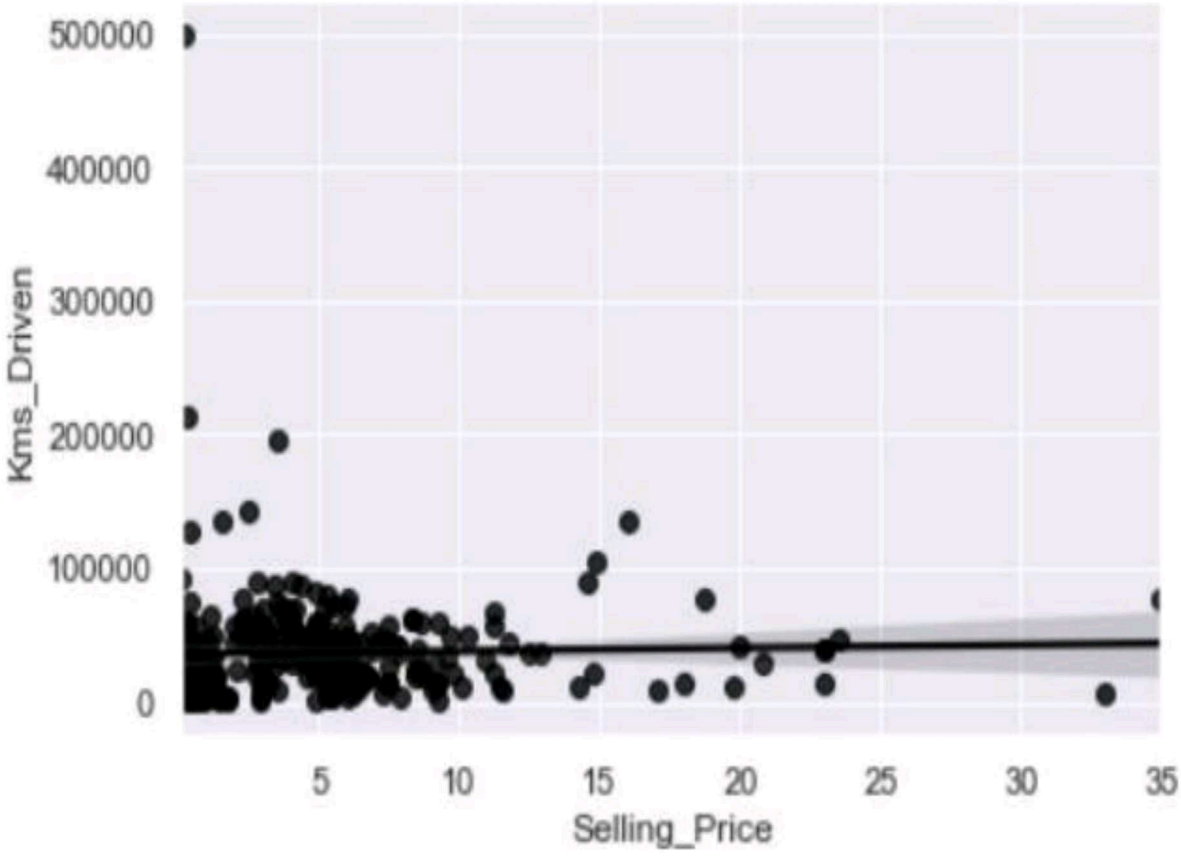


Figure : Kilometers Driven v/s Selling Price

The graph above shows that a vehicle with a high number of kilometers driven has a lower selling price than one with a low number of kilometers driven.

ONE HOT ENCODING

The one hot coding approach is used to deal with the categorical variables in the dataset. It generates a sparse matrix or a dense array based on the parameters while creating a binary column for each category or parameter. Fuel Type, Seller Type, and Transmission were the three categorical variables in our dataset. Following one hot encoding, these variables are given a binary representation, so that for a car with a Fuel Type of Diesel, the value of Fuel_Type_Diesel is a binary 1 and the value of Fuel_Type_Petrol is a binary 0. The same procedure is applied for the remaining category variables.

Figure : Correlation Heatmap

Heatmap of Correlation Features for the Final Dataset: A dataset's correlation features define how close two variables are to having a linear relationship with each other. Features with a high correlation are more linearly dependent and have the same effect on the dependent variable. If two variables have a high correlation, we can always eliminate one of them. The heatmap of correlation is shown below, with darker colors representing high correlation and lighter colors representing low correlation.

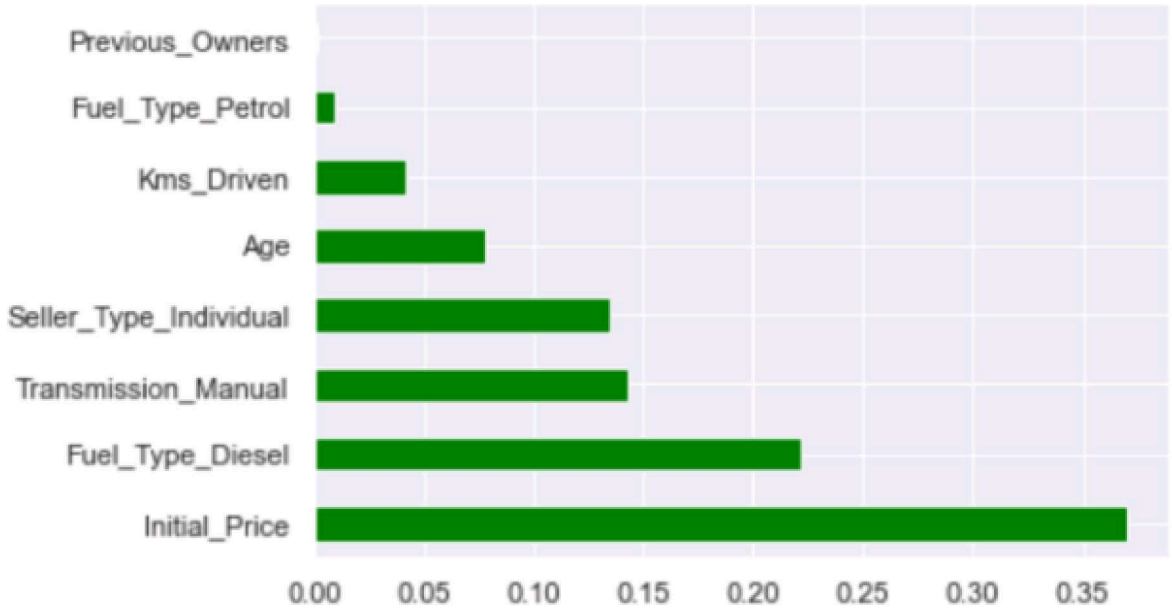
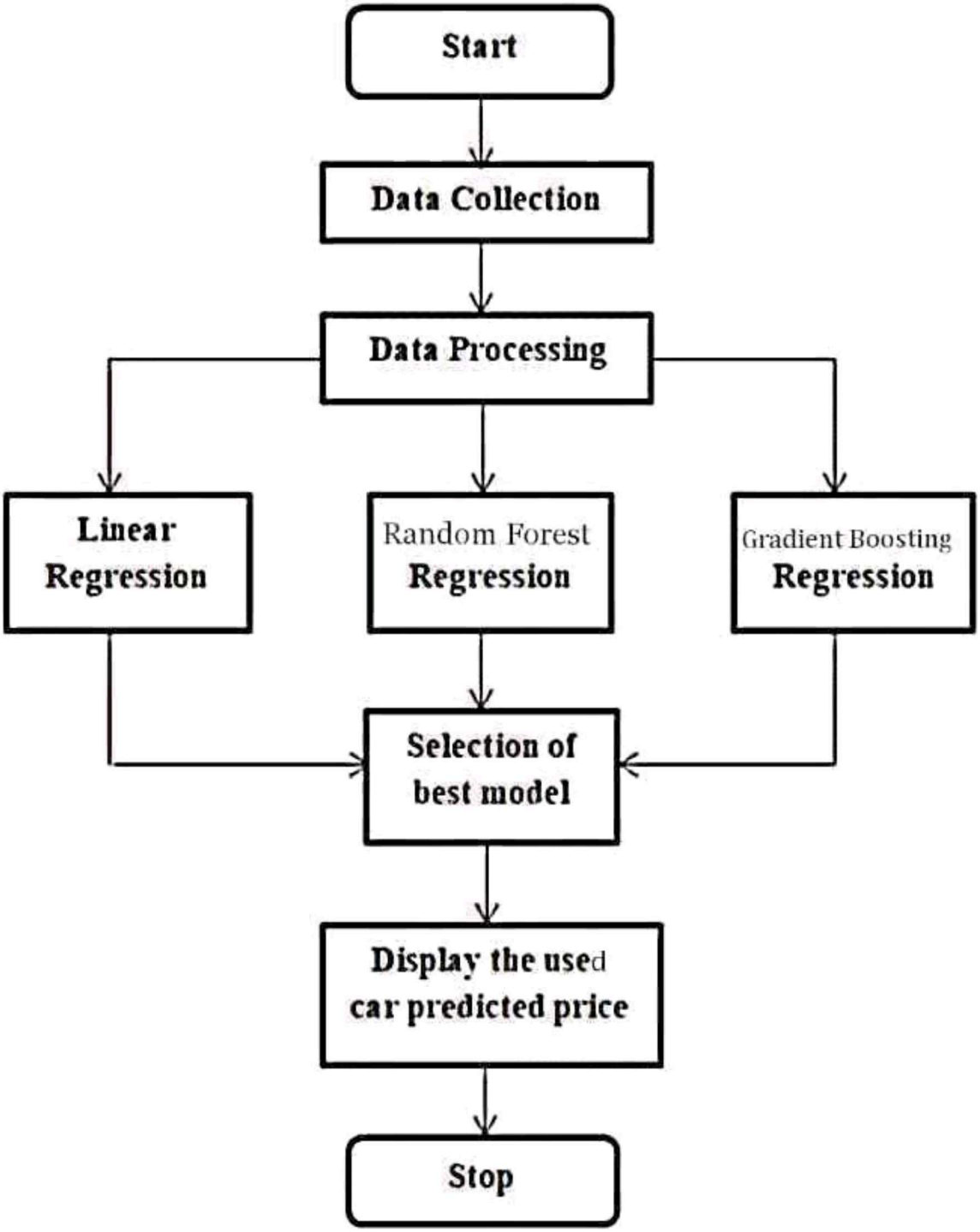


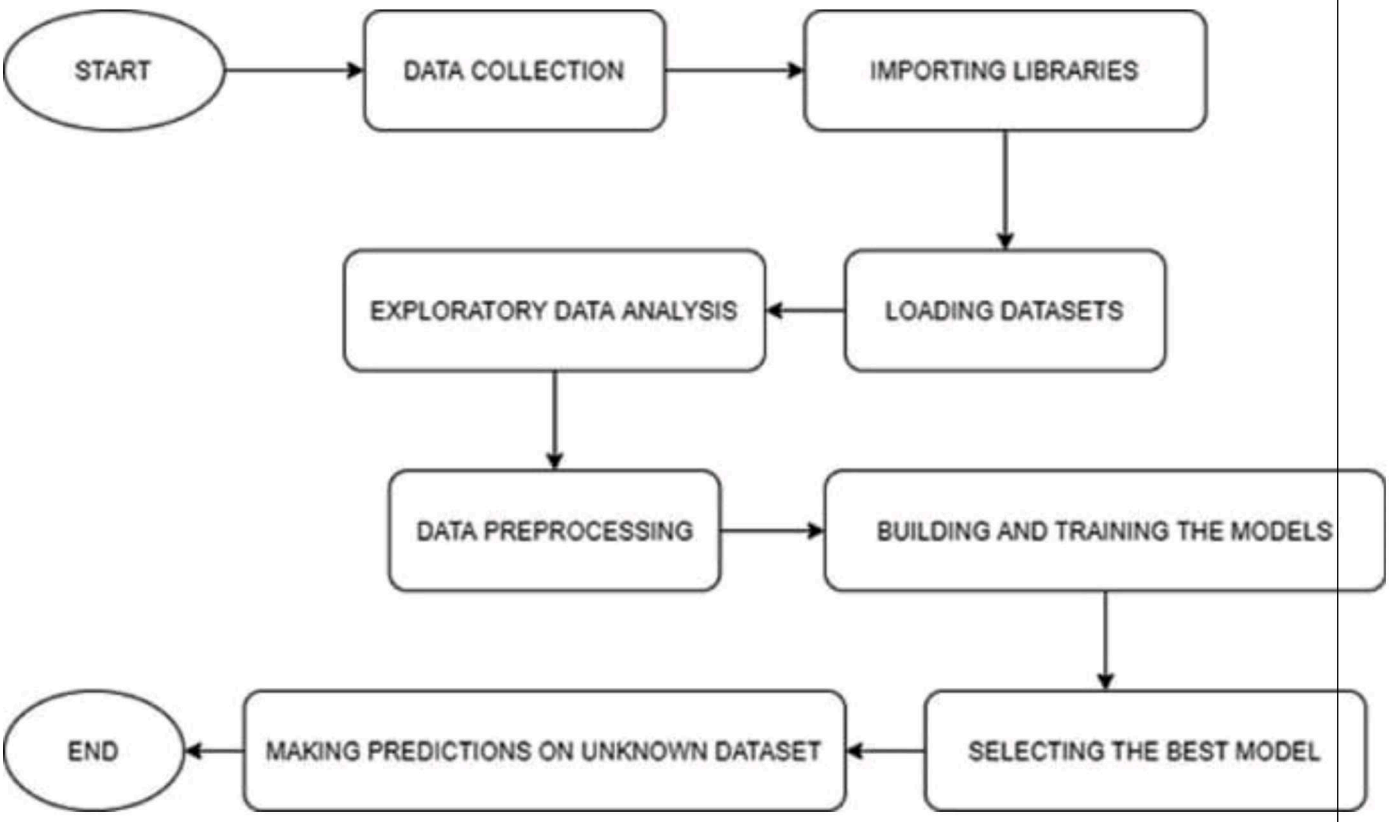
Fig: Feature Importance

Feature Importance of dataset: The feature importance technique provides a score to features in a feature set based on their usefulness in predicting the target variable. Initial Price is the most relevant feature in the provided dataset, while Previous Owners is the least important.

UML DIAGRAMS

WORK FLOW DIAGRAMS




START
DATA COLLECTION
IMPORTING LIBRARIES
LOADING DATASETS
EXPLORATORY DATA ANALYSIS
DATA PREPROCESSING
BUILDING AND TRAINING THE MODELS
SELECTING THE BEST MODEL
MAKING PREDICTIONS ON UNKNOWN DATASET
END

SOFTWARE ENVIRONMENT

Minimum Hardware Requirements:

- Processor - intel-i5
- Speed - 2.50 GHz
- Ram - 8 GB
- Hard Disk - 1TB

Software Requirements:

- Operating System - windows 10
- IDE - Anaconda
- Coding Language - python
- Technology - Machine Learning

SOURCE CODE

IMPORTING PACKAGES

```
import warnings
warnings.filterwarnings('ignore')
import pandas as pd
data = pd.read_csv("C:\Program Files\car data.csv")
```

Display Top 5 Rows of The Dataset

```
In [5]: data.head()
```

Out[5]:

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual

CHECK LAST 5 ROWS OF DATASET

```
In [6]: data.tail()
```

Out[6]:

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmissi
296	city	2016	9.50	11.6	33988	Diesel	Dealer	Manı
297	brio	2015	4.00	5.9	60000	Petrol	Dealer	Manı
298	city	2009	3.35	11.0	87934	Petrol	Dealer	Manı
299	city	2017	11.50	12.5	9000	Diesel	Dealer	Manı
300	brio	2016	5.30	5.9	5464	Petrol	Dealer	Manı

Find Shape of Our Dataset (Number of Rows And Number of Columns)

```
In [7]: data.shape
```

```
Out[7]: (301, 9)
```

```
In [8]: print("Number of Rows",data.shape[0])
        print("Number of Columns",data.shape[1])
```

```
Number of Rows 301
Number of Columns 9
```

Get Information About Our Dataset Like the Total Number of Rows, Total Number of columns

```
In [9]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 301 entries, 0 to 300
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Car_Name       301 non-null    object
 1   Year           301 non-null    int64
 2   Selling_Price  301 non-null    float64
 3   Present_Price  301 non-null    float64
 4   Kms_Driven     301 non-null    int64
 5   Fuel_Type      301 non-null    object
 6   Seller_Type    301 non-null    object
 7   Transmission   301 non-null    object
 8   Owner          301 non-null    int64
dtypes: float64(2), int64(3), object(4)
memory usage: 21.3+ KB
```

Check Null Values In The Dataset

```
In [10]: data.isnull().sum()

Out[10]: Car_Name          0
         Year              0
         Selling_Price     0
         Present_Price     0
         Kms_Driven        0
         Fuel_Type         0
         Seller_Type       0
         Transmission      0
         Owner             0
         dtype: int64
```

Get Overall Statistics About The Dataset

```
In [11]: data.describe()
```

Out[11]:

	Year	Selling_Price	Present_Price	Kms_Driven	Owner
count	301.000000	301.000000	301.000000	301.000000	301.000000
mean	2013.627907	4.661296	7.628472	36947.205980	0.043189
std	2.891554	5.082812	8.644115	38886.883882	0.247915
min	2003.000000	0.100000	0.320000	500.000000	0.000000
25%	2012.000000	0.900000	1.200000	15000.000000	0.000000
50%	2014.000000	3.600000	6.400000	32000.000000	0.000000
75%	2016.000000	6.000000	9.900000	48767.000000	0.000000
max	2018.000000	35.000000	92.600000	500000.000000	3.000000

Data Preprocessing

In [12]:
```
data.head(1)
```

Out[12]:

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual

In [14]:
```
import datetime
date_time = datetime.datetime.now()
data['Age']=date_time.year - data['Year']
data.head()
```

Out[14]:

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual

In [15]:
```
data.drop('Year',axis=1,inplace=True)
data.head()
```

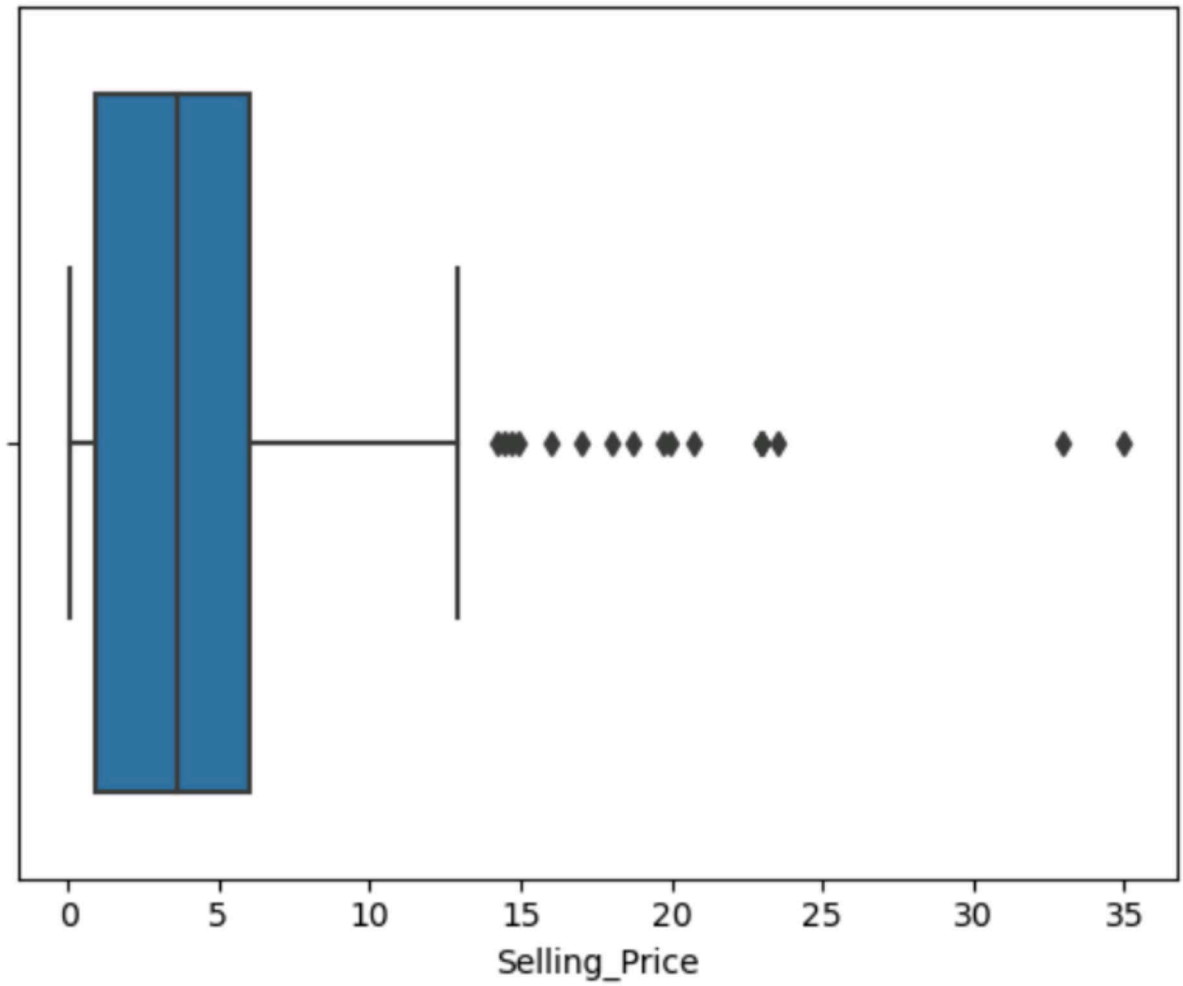
Out[15]:

	Car_Name	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Own
0	ritz	3.35	5.59	27000	Petrol	Dealer	Manual	
1	sx4	4.75	9.54	43000	Diesel	Dealer	Manual	
2	ciaz	7.25	9.85	6900	Petrol	Dealer	Manual	
3	wagon r	2.85	4.15	5200	Petrol	Dealer	Manual	
4	swift	4.60	6.87	42450	Diesel	Dealer	Manual	

Outlier Removal

```
In [16]: import seaborn as sns
         sns.boxplot(data['Selling_Price'])

Out[16]: <AxesSubplot:xlabel='Selling_Price'>
```



In [18]:
```
data = data[~(data['Selling_Price']>=33.0) & (data['Selling_Price']<=35.0)]
data.shape
```

Out[18]:
```
(299, 9)
```

Encoding the Categorical Columns

In [19]:
```
data.head(1)
```

Out[19]:

	Car_Name	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Own
0	ritz	3.35	5.59	27000	Petrol	Dealer	Manual	

In [20]:
```
data['Fuel_Type'].unique()
```

Out[20]:
```
array(['Petrol', 'Diesel', 'CNG'], dtype=object)
```

In [21]:
```
data['Fuel_Type'] = data['Fuel_Type'].map({'Petrol':0,'Diesel':1,'CNG':2})
data['Fuel_Type'].unique()
```

Out[21]:
```
array([0, 1, 2], dtype=int64)
```

In [22]:
```
data['Seller_Type'].unique()
```

Out[22]:
```
array(['Dealer', 'Individual'], dtype=object)
```

In [23]:
```
data['Seller_Type'] = data['Seller_Type'].map({'Dealer':0,'Individual':1})
data['Seller_Type'].unique()
```

Out[23]:
```
array([0, 1], dtype=int64)
```

In [24]:
```
data['Transmission'].unique()
```

Out[24]:
```
array(['Manual', 'Automatic'], dtype=object)
```

In [25]:
```
data['Transmission'] =data['Transmission'].map({'Manual':0,'Automatic':1})
data['Transmission'].unique()
```

Out[25]:
```
array([0, 1], dtype=int64)
```

In [26]:
```
data.head()
```

Out[26]:

	Car_Name	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Own
0	ritz	3.35	5.59	27000	0	0	0	
1	sx4	4.75	9.54	43000	1	0	0	
2	ciaz	7.25	9.85	6900	0	0	0	
3	wagon r	2.85	4.15	5200	0	0	0	
4	swift	4.60	6.87	42450	1	0	0	

Store Feature Matrix In X and Response(Target) In Vector y

```
In [27]: X = data.drop(['Car_Name','Selling_Price'],axis=1)
         y = data['Selling_Price']
         y
```

```
Out[27]: 0       3.35
         1       4.75
         2       7.25
         3       2.85
         4       4.60
                 ...
         296     9.50
         297     4.00
         298     3.35
         299    11.50
         300     5.30
         Name: Selling_Price, Length: 299, dtype: float64
```

Splitting The Dataset Into The Training Set And Test Set

```
from sklearn.model_selection import train_test_split
```

```
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.20,random_state=42)
```

Import The models

```
In [30]: from sklearn.linear_model import LinearRegression
         from sklearn.ensemble import RandomForestRegressor
         from sklearn.ensemble import GradientBoostingRegressor
```

Model Training

```
In [31]: lr = LinearRegression()
         lr.fit(X_train,y_train)

         rf = RandomForestRegressor()
         rf.fit(X_train,y_train)

         xgb = GradientBoostingRegressor()
         xgb.fit(X_train,y_train)

Out[31]: GradientBoostingRegressor()
```

Prediction on Test Data

In [32]:
```
y_pred1 = lr.predict(X_test)
y_pred2 = rf.predict(X_test)
y_pred3 = xgb.predict(X_test)
```

Evaluating the Algorithm

In [33]:
```
from sklearn import metrics
```

In [34]:
```
score1 = metrics.r2_score(y_test,y_pred1)
score2 = metrics.r2_score(y_test,y_pred2)
score3 = metrics.r2_score(y_test,y_pred3)
```

In [36]:
```
print(score1,score2,score3)
```

```
0.6790884983129406 0.7200166222018176 0.8707555170106966
```

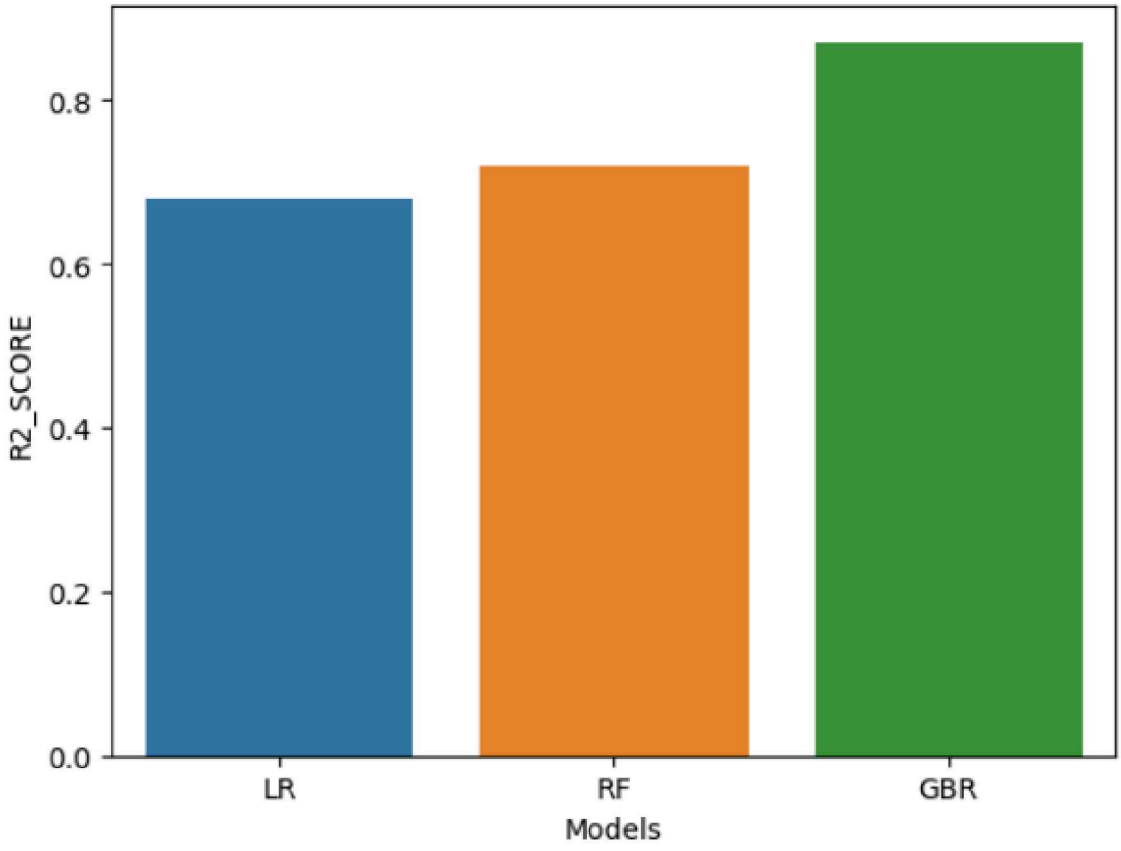
In [37]:
```
final_data = pd.DataFrame({'Models':['LR','RF','GBR'],
                           "R2_SCORE":[score1,score2,score3]})
final_data
```

Out[37]:

	Models	R2_SCORE
0	LR	0.679088
1	RF	0.720017
2	GBR	0.870756

```
In [38]: sns.barplot(final_data['Models'],final_data['R2_SCORE'])

Out[38]: <AxesSubplot:xlabel='Models', ylabel='R2_SCORE'>
```



Save The Model

In [43]:
```
xgb = GradientBoostingRegressor()
xgb_final = xgb.fit(X,y)
```

In [45]:
```
import joblib
joblib.dump(xgb_final,'car_price_predictor')
```

Out[45]:
```
['car_price_predictor']
```

In [46]:
```
model = joblib.load('car_price_predictor')
```

Prediction on New Data

In [51]:
```
import pandas as pd
data_new = pd.DataFrame({
    'Present_Price':6.9,
    'Kms_Driven':27000,
    'Fuel_Type':0,
    'Seller_Type':0,
    'Transmission':0,
    'Owner':1,
    'Age':9
},index=[0])
```

In [52]:
```
model.predict(data_new)
```

Out[52]:
```
array([4.58469195])
```

RESULTS

After applying regression algorithms to the model, the r_2 scores are

	Models	R2_SCORE
0	LR	0.679088
1	RF	0.7200172
2	GBR	0.870756

The Gradient Boosting Regression Algorithm has the best r_2 score of 0.870756 when all regression method's r_2 scores are compared, which simply implies that the Gradient Boosting Regression Algorithm has delivered the most accurate predictions when compared to the other algorithms.

FUTURE SCOPE

The developed machine learning model can be exported as a "Python class" and deployed as an open source, ready-to-use price predictor model, which can then be easily integrated with third-party websites. The model can be greatly optimised by using neural networks by designing deep learning network topologies, employing adaptive learning rates, and training on data clusters rather than the entire dataset.

A car price prediction has been a high-interest research area, as it requires noticeable effort and knowledge of the field expert. A considerable number of distinct attributes are examined for reliable and accurate predictions. The major step in the prediction process is the collection and pre-processing of the data. In this project, data was normalized and cleaned to avoid unnecessary noise for machine learning algorithms. Applying machine algorithms to the data set accuracy was less than 70%. Therefore, the ensemble of multiple machine learning algorithms has been proposed and this combination of ML methods gains an accuracy of 93%. This is a significant improvement compared to the single machine learning method approach. However, the drawback of the proposed system is that it consumes much more computational resources than a single machine learning algorithm. Although this system has achieved astonishing performance in the car price prediction problem, it can also be implemented using an advanced machine learning model and with Deep learning techniques to improve its efficiency and accuracy. Moreover, as innovation has been increased in automobiles and we can observe Electric vehicles have gained public attention and are preferred by most than a normal car

CONCLUSION

Predicting used car prices is a difficult task due to the large number of features and parameters that must be examined in order to get reliable findings. The first and most important phase is data collection and preprocessing. The model was then defined and built in order to implement algorithms and generate results.

REFERENCES

- Data Set https://www.kaggle.com/nehalbirla/motorcycle-dataset
- Regression Analysis Using :- Multiple Regression tutorialspoint.com

[1] V.Sravan kiran,Rajath kala,V. Nagesh,S., Navdeep USED CAR PRICE PREDICTION ,jespublication Vol 13, Issue 06, June/2022

[2]Abishek R*1,Chandrashekar,Sameerchand Pudaruth inCAR PRICE PREDICTION USING MACHINE LEARNING TECHNIQUES,irjmets,Volume:04/Issue:02/February-2022

[3]Sai Prasad Potharaju Sameerchand Pudaruth, Predicting the Price of Used Cars using Machine Learning Techniques,

[4]International Journal of Information & Computation Technology,ISSN 0974-2239 Volume 4,

[5]Number 7 (2014), pp. 753-764

[6]Yadav, A., Kumar, E., & Yadav, P. K. (2021). Object detection and used car price

[7]predicting analysis system (UCPAS) using machine learning technique. Linguistics and Culture Review,1131-1147. https://doi.org/10.21744/lingcure.v5nS2.1660

[8]Enis Gegic,Du, J., Xie, L. Schroeder,Gelman, A., Hill,CAR PRICE PREDICTION IN THE USA BY

[9]USING LINEAR REGRESSION International Journal of Economic Behavior, vol. 11 n. 1, 2021, 99-108

[10]Nitis Monburinon , Prajak Chertchom ,Thongchai Kaewkiriya in Prediction of Prices for Used Car by Using Regression Models, IEEE vol. 4, no. 7, pp. 753–764,2018