

Car Resale value Prediction

(RITHIESH C, SANKARAN B, MAGESH KUMAR N, JENISH T L)

(Team ID: PNT2022TMID54110)

1. INTRODUCTION

1.1 Project Overview

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 across cities in the United States. Our results show that Random Forest model and K-Means clustering with linear regression yield the best results, but are compute heavy. Conventional linear regression also yielded satisfactory results, with the advantage of a significantly lower training time in comparison to the aforementioned methods.

1.2 Purpose

Now a days deciding whether a used car is worth the posted price when you see listings online can be difficult. Several factors, including mileage, make, model, year, etc. can influence the actual worth of a car. From the perspective of a seller, it is also a dilemma to price a used car appropriately. Based on existing data, the aim is to use machine learning algorithms to develop models for predicting used car prices.

The main aim of this project is to predict the price of used cars using the various Machine Learning (ML) models. This can enable the customers to make decisions based on different inputs or factors namely

- Brand or Type of the car
- Model of the car
- Location
- Year of manufacturing
- Type of fuel
- Price range or Budget
- Type of transmission which the customer prefers like Automatic or Manual
- Mileage

to name a few characteristic features required by the customer. The project Car Price Prediction deals with providing the solution to these problems. Through this project, we will get to know which of the factors are significant and tell us how they affect the car's worth in the market

2 LITERATURE SURVEY

2.1 Existing Problem

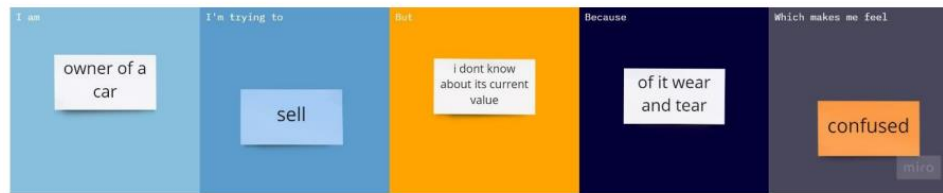
Deciding whether a used car is worth the posted price when you see listings online can be difficult. Several factors, including mileage, make, model, year, etc. can influence the actual worth of a car. From the perspective of a seller, it is also a dilemma to price a used car appropriately. Based on existing data, the aim is to use machine learning algorithms to develop models for predicting used car prices.

2.2 References

1. <https://www.kaggle.com/jpayne/852k-used-car-listings>
2. N. Monburinon, P. Chertchom, T. Kaewkiriya, S. Rungpheung, S. Buya and P. Boonpou, "Prediction of prices for used car by using regression models," 2018 5th International Conference on Business and Industrial Research (ICBIR), Bangkok, 2018, pp. 115-119.
3. Listiani M. 2009. Support Vector Regression Analysis for Price Prediction in a Car Leasing Application. Master Thesis. Hamburg University of Technology
4. Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. ACM, 2016.
5. Ke, Guolin, et al. "Lightgbm: A highly efficient gradient boosting decision tree." Advances in Neural Information Processing Systems. 2017.
6. Fisher, Walter D. "On grouping for maximum homogeneity." Journal of the American statistical Association 53.284 (1958): 789-798.
7. <https://scikit-learn.org/stable/modules/classes.html>: Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

2.3 Problem Statement Definition

Data science aims to make our predictions to be more accurate by having the many previous data collected as dataset as in our project with difficult economic conditions, it is likely that sales of second-hand imported cars and used cars will increase. In many developed countries, it is common to lease a car rather than buying it outright. After the lease period is over, the buyer has the possibility to buy the car at its residual value, i.e. its expected resale value. Thus, it is of commercial interest to sellers/financers to be able to predict the salvage value of cars with accuracy. In order to predict the resale value of the car, we proposed an intelligent, flexible, and effective system that is based on using regression algorithms. Considering the main factors which would affect the resale value of a vehicle a regression model is to be built that would give the nearest resale value of the vehicle. We will be using various regression algorithms and algorithm with the best accuracy will be taken as a solution, then it will be integrated to the web-based application where the user is notified with the status of his product



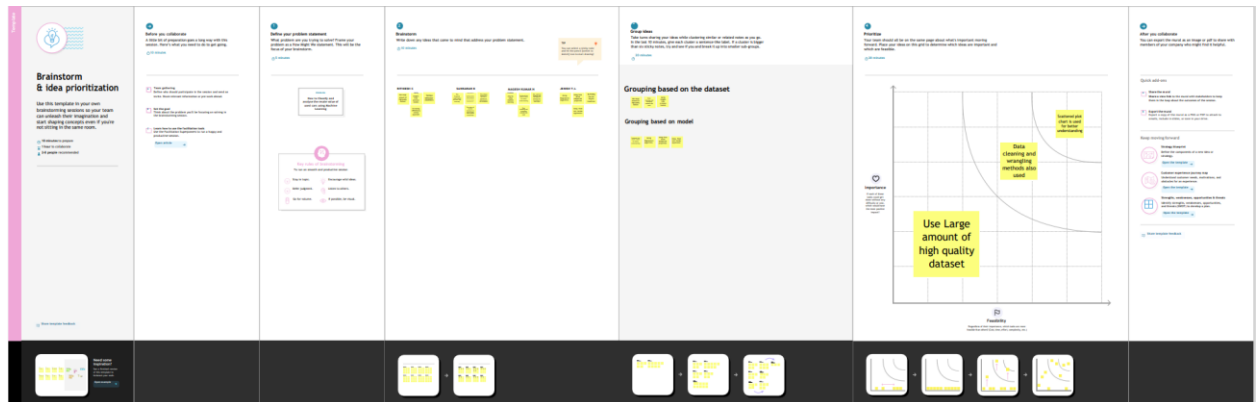
Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Owner of a car	Sell	I don't know about its current value	Of it wear and tear	Confused
PS-2	Owner of a car	Find its resale value	I am concerned about the minor dents	Of a small accident	A little worried

3 IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming



3.3 Proposed Solution

S.no	Parameter	Description
1	Problem Statement (Problem to be solved)	<ul style="list-style-type: none"> The main aim of this project is to predict the price of used cars using the various Machine Learning (ML) models. The project should take parameters related to used car as inputs and enable the customers to make decisions by their own.
2	Idea / Solution description	The model is to be built that would give the nearest resale value of the vehicle. By using these best accuracy value will be taken as a solution and it will be integrated to the web-based application where the user is notified with the status of his product.
3	Novelty / Uniqueness	Used car price prediction is effectively used to determine the worthiness of the car by their own within few minutes by using various features such as year, model, mileage(km), etc.
4	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> If the user wants to buy or sell a own car it helps users to predict the correct valuation by their own. A loss function is to be optimized and mainly a weak learner can make predictions for used cars easily

5	Business Model (Revenue Model)	It helps users to predict the correct valuation of the car remotely with perfect valuation and without human intervention like car dealers in the process to eliminate biased valuation predicted by the dealer.
6	Scalability of the Solution	Using Stored data and machine learning approaches, this project proposed a scalable framework for predicting values for different type of used cars present all over India.

3.4 Problem Solution fit

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS Who is your customer? i.e. working parents of 9-5 y.o. kids Both used car sellers and buyers	6. CUSTOMER CONSTRAINTS CC What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices. <ul style="list-style-type: none">To determine the worthiness of the car by their own within few minutesA loss function is to be optimized by spending money for dealers, brokers to buy or sell a car.	5. AVAILABLE SOLUTIONS AS Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking <ul style="list-style-type: none">In the past User cannot find the value of used car buy their own without prior knowledge about cars.A person who don't know much about the car can also make predictions for used cars easily.	Explore AS, differentiate
	2. JOBS-TO-BE-DONE / PROBLEMS J&P Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides. To build a supervised machine learning model using regression algorithms for forecasting the value of a vehicle based on multiple attributes such as <ul style="list-style-type: none">Condition of EngineAge of the used carKilometers drivenNumber of owners	9. PROBLEM ROOT CAUSE RC What is the real reason that this problem exists? What is the back story behind the need to do this job? i.e. customers have to do it because of the change in regulations. <ul style="list-style-type: none">The price predicted by the dealers or brokers for used car is not trustful.users can predict the correct valuation of the car remotely without human intervention like car dealers.User can eliminate biased valuation predicted by the dealer.	7. BEHAVIOUR BE What does your customer do to address the problem and get the job done? i.e. Directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace) <ul style="list-style-type: none">The History of Your Car's condition and documents produced by them will be suspicious.The model is to be built that would give the nearest resale value of the vehicle by eliminating anonymous value predicted by the humans.	
Focus on J&P, tap into BE, understand RC	3. TRIGGERS TR What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news. users can predict the correct valuation of the car by their own like olx, cars24 and other car resale value prediction websites by using model, year, owner, etc.	10. YOUR SOLUTION SL If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour. <ul style="list-style-type: none">The main aim of this project is to predict the price of used cars using the Machine Learning (ML) algorithms and collection data's about different cars. The project should take parameters related to used car as inputs and enable the customers to make decisions by their own.	8. CHANNELS OF BEHAVIOUR CH 8.1 ONLINE What kind of actions do customers take online? Extract online channels from #7 8.2 OFFLINE What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development. <ul style="list-style-type: none">customer should predict the worth of the car by using different parameters given by the owner.User Should confirm the details provided about the vehicle in RTO online.user can decide by seeing the exterior and interior condition of the car.User can test the performance of the car and to buy it up in a affordable price based on its condition.	Focus on J&P, tap into BE, understand RC
Identify strong TR & EM	4. EMOTIONS: BEFORE / AFTER EM How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design. Before: <ul style="list-style-type: none">User will be in fear about the biased values predicted by the humans based on the condition of the car. After: <ul style="list-style-type: none">user can determine the worthiness of the car by their own without human intervention.			

WE & SL grows together!

4 REQUIREMENT ANALYSIS

4.1 Functional requirement

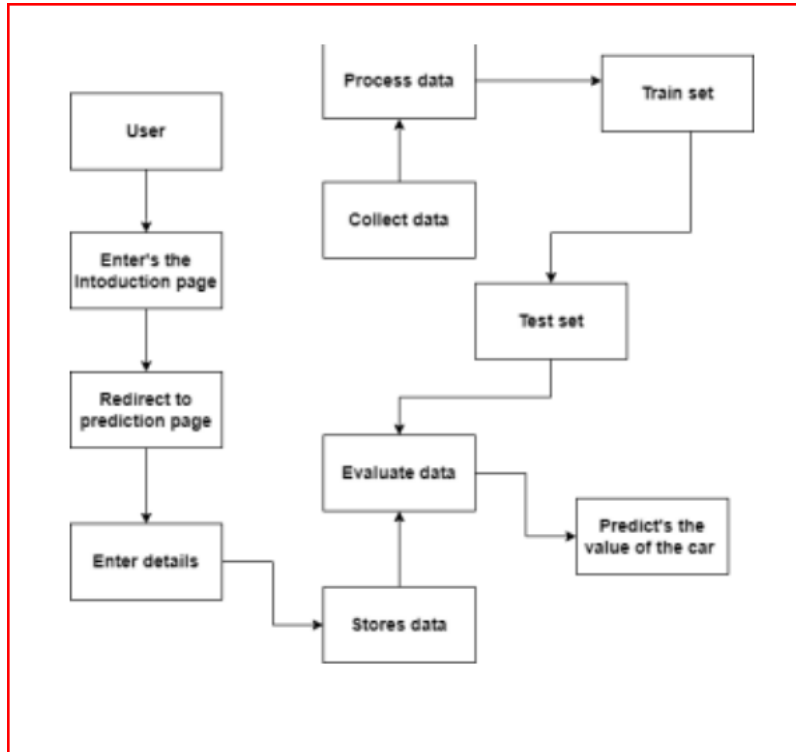
FR No.	Functional Requirements	Sub Registration
FR-1	Registration	Registration can be done using mobile number or gmail and needed some user information
FR-2	Login	User only log in by user id and password, Which is given during registration
FR-3	Delivery confirmation	Confirmation via email and phone number
FR-4	Assistance	Bot is integrated with the application to make the usability simple

4.2 Non-Functional requirement

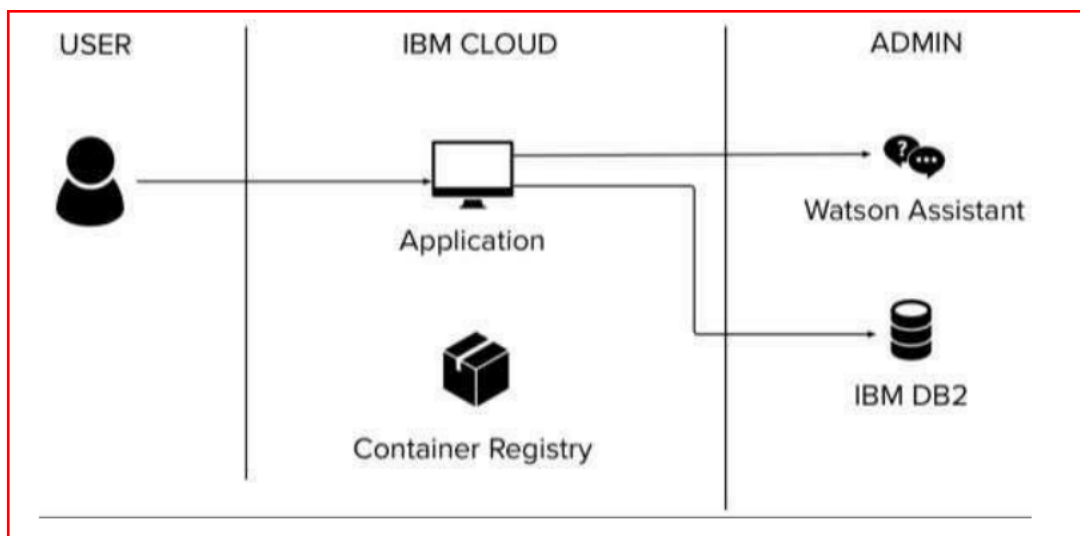
NFR No.	Non-Functional Requirements	Description
NFR-1	Usability	A user-friendly interface with chat bot to make usability efficient
NFR-2	Security	Secured connection HTTPS should be established for transmitting requests and responses
NFR-3	Reliability	The system should handle excepted as well as unexpected errors and exceptions to avoid termination of the program
NFR-4	Performance	The system shall be able to handle multiple requests at any given point in time and generate an appropriate response.
NFR-5	Availability	It is a cloud based web application so user can access without any platform limitations ,just using a browsers with a internet connection is enough for use the application

5 PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture



5.3 User Stories

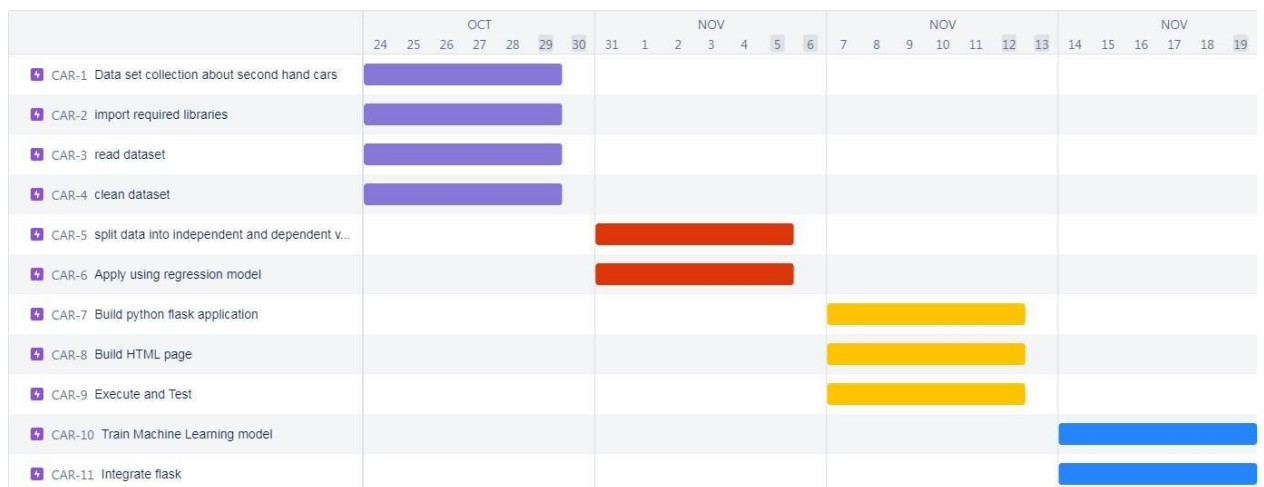
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Sprint-1	Pre-process data	USN-1	Collect Dataset	1	Low
Sprint-1		USN-2	Import required libraries	1	Low
Sprint-1		USN-3	Read and clean data sets	2	Low
Sprint-2	Model building	USN-1	Split data into independent and dependent variables	3	Medium
Sprint-2		USN-2	Apply using regression model	3	Medium
Sprint-3	Application building	USN-1	Build python flask application and HTML page	5	High
Sprint-3		USN-2	Execute and test	5	High
Sprint-4	Training the model	USN-1	Train machine learning model	5	High
Sprint-4		USN-2	Integrate flask	5	High

6 PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6.2 Sprint Delivery Schedule



7 CODING & SOLUTIONING

7.1 Feature 1

We have created a webpage as simple and user friendly that a user should understand how to interface at the first time they the webpage for that we have used html for developing a webpage

This shows the document type is html

```
<!DOCTYPE html>
```

Webpage is created in English language

```
<html lang="en">
```

header part of the webpage

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
```

body part of the webpage

```
<body>

  <div style="color:red">
    <form action="{{ url_for('predict')}}" method="post">
```

It gives the top most heading as Predictive analysis

```
<h2>Predictive analysis</h2>
```

It gives the heading as Year

```
<h3>Year</h3>
```

We needs to get a input to the year so we were using input tag to get a input the user needs to enter the year of the car purchased

```
<input id="first" name="Year" type="number ">
```

It gives the heading as Showroom Price and using the input tag to get the showroom price from the user

<h3>What is the Showroom Price?(In lakhs)</h3>
<input id="second" name="Present_Price" required="required">

It gives the heading as How Many Kilometers Drived ? and using the input tag to get the Kilometers driven from the user

<h3>How Many Kilometers Drived?</h3><input id="third" name="Kms_Driven" required="required">

It gives the heading as How much owners previously had the car (0 or 1 or 3) ?and using the input tag to get the No of Owner used the car from the user

<h3>How much owners previously had the car(0 or 1 or 3) ?</h3>
<input id="fourth" name="Owner" required="required">

It gives the heading as What Is the Fuel type ? and using the input tag to get the Fuel Type of the car from the user

<h3>What Is the Fuel type?</h3>
<select name="Fuel_Type_Petrol" id="fuel" required="required">

Having the options as Petrol or Diesel or CNG

<option value="Petrol">Petrol</option>
<option value="Diesel">Diesel</option>
<option value="Diesel">CNG</option>
</select>

It gives the heading as Are you A Dealer or Individual and using the input tag to get the Seller Type and giving the options as Dealer or Individual of the car from the user

<h3>Are you A Dealer or Individual</h3>
<select name="Seller_Type_Individual" id="resea" required="required">
<option value="Dealer">Dealer</option>
<option value="Individual">Individual</option>
</select>

It gives the heading Transmission type and using the input tag to get the Type of Transmission and giving the options as is the car Manual or Automatic from the user

<h3>Transmission type</h3>
<select name="Transmission_Mannual" id="research" required="required">
<option value="Mannual">Manual Car</option>
<option value="Automatic">Automatic Car</option>
</select>

<button id="sub" type="submit ">Calculate the Selling Price</button>


```
</form>
```

```
<br><br><h3>{{ prediction_text }}<h3>  
</div>
```

Using CSS to style our webpage

```
<style>  
  body {  
    background-color: lightslategray;  
    text-align: center;  
    padding: 0px;  
  }  
  
  #research {  
    font-size: 18px;  
    width: 100px;  
    height: 23px;  
    top: 23px;  
  }  
  
  #box {  
    border-radius: 60px;  
    border-color: 45px;  
    border-style: solid;  
    font-family: cursive;  
    text-align: center;  
    background-color: rgb(168, 131, 61);  
    font-size: medium;  
    position: absolute;  
    width: 700px;  
    bottom: 9%;  
    height: 850px;  
    right: 30%;  
    padding: 0px;  
    margin: 0px;  
    font-size: 14px;  
  }  
  
  #fuel {
```

```
width: 83px;
height: 43px;
text-align: center;
border-radius: 14px;
font-size: 20px;
}

#fuel:hover {
  background-color: coral;
}

#research {
  width: 99px;
  height: 43px;
  text-align: center;
  border-radius: 14px;
  font-size: 18px;
}

#research:hover {
  background-color: coral;
}

#resea {
  width: 99px;
  height: 43px;
  text-align: center;
  border-radius: 14px;
  font-size: 18px;
}

#resea:hover {
  background-color: coral;
}

#sub {
  width: 120px;
  height: 43px;
  text-align: center;
  border-radius: 14px;
  font-size: 18px;
}

#sub:hover {
```

```
        background-color: darkcyan;
    }

    #first {
        border-radius: 14px;
        height: 25px;
        font-size: 20px;
        text-align: center;
    }

    #second {
        border-radius: 14px;
        height: 25px;
        font-size: 20px;
        text-align: center;
    }

    #third {
        border-radius: 14px;
        height: 25px;
        font-size: 20px;
        text-align: center;
    }

    #fourth {
        border-radius: 14px;
        height: 25px;
        font-size: 20px;
        text-align: center;
    }
</style>
</body>

</html>
```

7.2 Feature 2

For the backend we have used python flask and using the collected dataset of the used cars details we made our backend that works simple and more accurate to give user more accurate value

Importing the libraries we needed using the keyword import

```
from flask import Flask, render_template, request
import jsonify
import requests
import pickle
import numpy as np
import sklearn
from sklearn.preprocessing import StandardScaler
```

Flask constructor takes the name of the current module name and load the model pkl file and get the data from it and predict the values accordingly

```
app = Flask(__name__)
model = pickle.load(open('random_forest_regression_model.pkl', 'rb'))
```

@app.route keyword is used to link the pkl file and python code to the HTML file

```
@app.route('/', methods=['GET'])
```

Render template renders that if the webpage doesn't work properly it will move to that page

```
def Home():
    return render_template('index.html')
standard_to = StandardScaler()
@app.route("/predict", methods=['POST'])
```

Creating a function getting the inputs from the html input forms and predict the output accordingly

```
def predict():
    Fuel_Type_Diesel=0
    if request.method == 'POST':
        Year = int(request.form['Year'])
        Present_Price=float(request.form['Present_Price'])
        Kms_Driven=int(request.form['Kms_Driven'])
        Kms_Driven2=np.log(Kms_Driven)
        Owner=int(request.form['Owner'])
```

```

Fuel_Type_Petrol=request.form['Fuel_Type_Petrol']
if(Fuel_Type_Petrol=='Petrol'):
    Fuel_Type_Petrol=1
    Fuel_Type_Diesel=0
else:
    Fuel_Type_Petrol=0
    Fuel_Type_Diesel=1
Year=2020-Year
Seller_Type_Individual=request.form['Seller_Type_Individual']
if(Seller_Type_Individual=='Individual'):
    Seller_Type_Individual=1
else:
    Seller_Type_Individual=0
Transmission_Mannual=request.form['Transmission_Mannual']
if(Transmission_Mannual=='Mannual'):
    Transmission_Mannual=1
else:
    Transmission_Mannual=0

prediction=model.predict([[Present_Price,Kms_Driven2,Owner,Year,Fuel_Type_Diesel,Fuel_Type_Petrol,Seller_Type_Individual,Transmission_Mannual]])
output=round(prediction[0],2)

If the output is not predicted then it shows as Sorry you cannot sell this car if it's predicted then it show as you can sell the car at this price

    if output<0:
        return render_template('index.html',prediction_texts="Sorry you cannot sell this car")
    else:
        return render_template('index.html',prediction_text="You Can Sell The Car at
{}".format(output))
    else:
        return render_template('index.html')

if __name__=="__main__":
    app.run(debug=True)

```


8 TESTING

8.1 Test Cases

The test case has the Present_Price , Kms driven , Owner , Years_Old , Fuel_Type , Seller_Type , Transmission_Type to predict the value of the used car

Service Details - IBM Cloud x IBM x IBM Watson Studio x +

dataplatform.cloud.ibm.com/ml-runtime/deployments/71547275-5076-442b-8559-57f19ed6680e/test?space_id=1236baca-5570-434f-92be-60a8fe1...

IBM Watson Studio Search in your workspaces Buy ? ? Ritihesh C's Account Dallas RC

Deployments / CAR PRICE / CAR /

CAR PRICE Deployed Online

API reference **Test**

Enter input data

Text input **JSON input**

Enter data manually or use a CSV file to populate the spreadsheet. Max file size is 50 MB.

[Download CSV template](#) [Browse local files](#) [Search in space](#) [Clear all](#)

	Present_Price (float64)	Kms_Driven (int64)	Owner (int64)	Years_Old (int64)	Fuel_Type_Diesel (uint8)	Fuel_Type_Petrol (uint8)	Seller_Type_In
1							

0 rows, 8 columns

Predict

8.2 User Acceptance Testing

The inputs that are given by the user is acceptable and able to give the predicted output

Service Details - IBM Cloud x IBM x IBM Watson Studio x +

dataplatfom.cloud.ibm.com/ml-runtime/deployments/71547275-5076-442b-8559-57f19ed6680e/test?space_id=1236baca-5570-434f-92be-60a8fe1...

IBM Watson Studio Search in your workspaces Buy ? Rithiesh C's Account Dallas RC

Deployments / CAR PRICE / CAR /

CAR PRICE Deployed Online

API reference **Test**

Enter input data

Text input JSON input

Enter data manually or use a CSV file to populate the spreadsheet. Max file size is 50 MB.

[Download CSV template](#) [Browse local files](#) [Search in space](#) [Clear all](#)

at64)	Kms_Driven (int64)	Owner (int64)	Years_Old (int64)	Fuel_Type_Diesel (uint8)	Fuel_Type_Petrol (uint8)	Seller_Type_Individual (uint8)	Transmisstion
1	10000	1	5	0	1	1	0

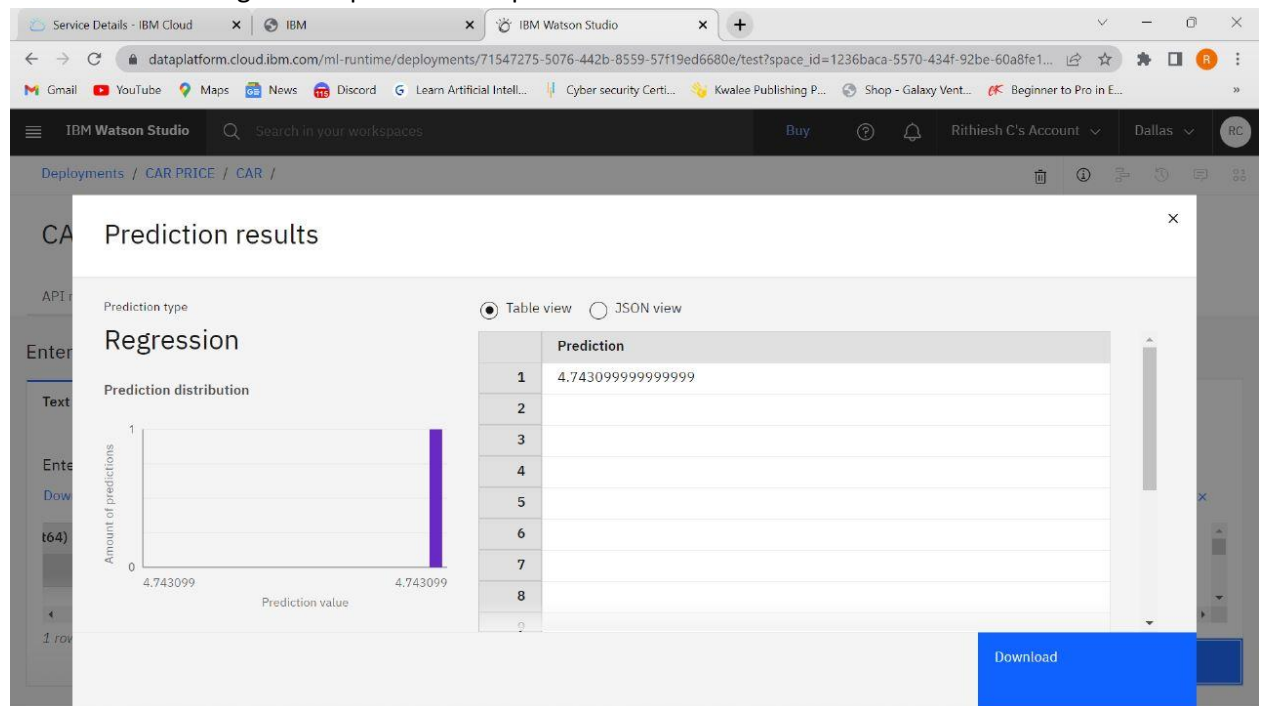
1 row, 8 columns

Predict

9 RESULTS

9.1 Performance Metrics

The trained model gives the predicted output



10 ADVANTAGES & DISADVANTAGES

PROS

- .1 Good at learning complex and non-linear relationships
- .2 Highly explainable and easy to interpret
- .3 Robust to outliers
- .4 No feature scaling is required

CONS

- .1 Consumes more time
- .2 Requires high computational power

11 CONCLUSION

We started with understanding the use case of machine learning in the Automotive industry and how machine learning has transformed the driving experience. Moving on, we looked at the various factors that affect the resale value of a used car and performed exploratory data analysis (EDA). Further, we build a Random Forest Regression model to predict the resale value of a used car. Finally, we evaluated the performance of the model using the R squared score and Residual Plot.

We could have also used simpler regression algorithms like Linear Regression and Lasso Regression. Still, we need to make sure there are no outliers in the dataset before implementing them. Pair plots and scatter plots help visualize the outliers

12 FUTURE SCOPE

For better performance, we plan to judiciously design deep learning network structures, use adaptive learning rates and train on clusters of data rather than the whole dataset. To correct for overfitting in Random Forest, different selections of features and number of trees will be tested to check for change in performance.

13 APPENDIX

Source Code

WEBSITE

FRONTEND -HTML

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>

<body>

  <div style="color:red">
    <form action="{{ url_for('predict')}}" method="post">
      <h2>Predictive analysis</h2>
```

```

    <h3>Year</h3>
    <input id="first" name="Year" type="number ">
    <h3>What is the Showroom Price?(In lakhs)</h3><br><input id="second"
name="Present_Price" required="required">
    <h3>How Many Kilometers Drived?</h3><input id="third" name="Kms_Driven"
required="required">
    <h3>How much owners previously had the car(0 or 1 or 3) ?</h3><br><input id="fourth"
name="Owner" required="required">
    <h3>What Is the Fuel type?</h3><br><select name="Fuel_Type_Petrol" id="fuel"
required="required">
        <option value="Petrol">Petrol</option>
        <option value="Diesel">Diesel</option>
        <option value="Diesel">CNG</option>
    </select>
    <h3>Are you A Dealer or Individual</h3><br><select name="Seller_Type_Individual"
id="resea" required="required">
        <option value="Dealer">Dealer</option>
        <option value="Individual">Individual</option>
    </select>
    <h3>Transmission type</h3><br><select name="Transmission_Mannual" id="research"
required="required">
        <option value="Mannual">Manual Car</option>
        <option value="Automatic">Automatic Car</option>
    </select>
    <br><br><button id="sub" type="submit ">Calculate the Selling Price</button>
    <br>

```

```

</form>

```

```

    <br><br><h3>{{ prediction_text }}</h3>
</div>

```

```

<style>
    body {
        background-color: lightslategray;
        text-align: center;
    }

```

```
padding: 0px;  
}
```

```
#research {  
  font-size: 18px;  
  width: 100px;  
  height: 23px;  
  top: 23px;  
}
```

```
#box {  
  border-radius: 60px;  
  border-color: 45px;  
  border-style: solid;  
  font-family: cursive;  
  text-align: center;  
  background-color: rgb(168, 131, 61);  
  font-size: medium;  
  position: absolute;  
  width: 700px;  
  bottom: 9%;  
  height: 850px;  
  right: 30%;  
  padding: 0px;  
  margin: 0px;  
  font-size: 14px;  
}
```

```
#fuel {  
  width: 83px;  
  height: 43px;  
  text-align: center;  
  border-radius: 14px;  
  font-size: 20px;  
}
```

```
#fuel:hover {  
  background-color: coral;  
}
```

```
#research {  
  width: 99px;  
  height: 43px;  
  text-align: center;
```

```
border-radius: 14px;  
font-size: 18px;  
}
```

```
#research:hover {  
  background-color: coral;  
}
```

```
#resea {  
  width: 99px;  
  height: 43px;  
  text-align: center;  
  border-radius: 14px;  
  font-size: 18px;  
}
```

```
#resea:hover {  
  background-color: coral;  
}
```

```
#sub {  
  width: 120px;  
  height: 43px;  
  text-align: center;  
  border-radius: 14px;  
  font-size: 18px;  
}
```

```
#sub:hover {  
  background-color: darkcyan;  
}
```

```
#first {  
  border-radius: 14px;  
  height: 25px;  
  font-size: 20px;  
  text-align: center;  
}
```

```
#second {  
  border-radius: 14px;  
  height: 25px;  
  font-size: 20px;  
  text-align: center;
```

```
    }

    #third {
        border-radius: 14px;
        height: 25px;
        font-size: 20px;
        text-align: center;
    }

    #fourth {
        border-radius: 14px;
        height: 25px;
        font-size: 20px;
        text-align: center;
    }
</style>
</body>

</html>
```

BACKEND -PYTHON

```
from flask import Flask, render_template, request
import jsonify
import requests
import pickle
import numpy as np
import sklearn
from sklearn.preprocessing import StandardScaler
app = Flask(__name__)
model = pickle.load(open('random_forest_regression_model.pkl', 'rb'))
@app.route('/', methods=['GET'])
def Home():
```



```
return render_template('index.html')
```

```
standard_to = StandardScaler()
```

```
@app.route("/predict", methods=['POST'])
```

```
def predict():
```

```
    Fuel_Type_Diesel=0
```

```
    if request.method == 'POST':
```

```
        Year = int(request.form['Year'])
```

```
        Present_Price=float(request.form['Present_Price'])
```

```
        Kms_Driven=int(request.form['Kms_Driven'])
```

```
        Kms_Driven2=np.log(Kms_Driven)
```

```
        Owner=int(request.form['Owner'])
```

```
        Fuel_Type_Petrol=request.form['Fuel_Type_Petrol']
```

```
        if(Fuel_Type_Petrol=='Petrol'):
```

```
            Fuel_Type_Petrol=1
```

```
            Fuel_Type_Diesel=0
```

```
        else:
```

```
            Fuel_Type_Petrol=0
```

```
            Fuel_Type_Diesel=1
```

```
        Year=2020-Year
```

```
        Seller_Type_Individual=request.form['Seller_Type_Individual']
```

```
        if(Seller_Type_Individual=='Individual'):
```

```
            Seller_Type_Individual=1
```

```
        else:
```

```
            Seller_Type_Individual=0
```

```
        Transmission_Mannual=request.form['Transmission_Mannual']
```

```
        if(Transmission_Mannual=='Mannual'):
```

```
            Transmission_Mannual=1
```

```
else:

    Transmission_Mannual=0

prediction=model.predict([[Present_Price,Kms_Driven2,Owner,Year,Fuel_Type_Diesel,Fuel_Type_Petrol
,Seller_Type_Individual,Transmission_Mannual]])

output=round(prediction[0],2)

if output<0:

    return render_template('index.html',prediction_texts="Sorry you cannot sell this car")

else:

    return render_template('index.html',prediction_text="You Can Sell The Car at {}".format(output))

else:

    return render_template('index.html')

if __name__=="__main__":

    app.run(debug=True)
```

GitHub & Project Demo Link

Github link :

<https://github.com/IBM-EPBL/IBM-Project-34709-1660273047>

Project demo link :

<https://youtu.be/GUJEOywAf0g>