PROJECT REPORT - CAR PURCHASE PREDICTION USING ML

Team ID	**Team-592746**
Project Name	**Project – Car Purchase Prediction using ML**

NAME	REGISTRATION NO.	EMAIL ID
SANMATHI PRIYA K R	20MIS0363	sanmathipriya.kr2020@vitstudent.ac.in
SRIJA BASAK	21BCE3506	srija.basak2021@vitstudent.ac.in
GIRISH ASHOK WAGH	21BKT0006	girishashok.wagh2021@vitstudent.ac.in

1. INTRODUCTION

1.1 Project Overview

This ML model predicts the likelihood of potential buyers of purchasing a car or not, based on factors such as gender,age and salary. The ML model is integrated into a user friendly interface, thus facilitating easy predictions for users.

1.2 Purpose

This ML model will be extremely beneficial for the automotive industry as it offers insights for tailored marketing strategies. Upon being trained with large data, and due to feature engineering, the model attains a high accuracy rate, thus making the predictions dependable upon.

2. LITERATURE SURVEY

2.1 Existing Problem

The automotive industry, while continuously evolving, faces inherent challenges in its marketing strategies that can hinder efficient customer engagement and resource utilization. Traditional marketing approaches often rely on demographic information, providing a limited understanding of individual preferences and behaviors. This results in generic marketing campaigns that may not effectively resonate with potential car buyers.

Furthermore, there is a prevailing under utilization of advanced technologies such as machine learning in predicting customer behavior for car purchases. Many businesses still rely on conventional strategies that may not harness the full potential of data-driven decision-making. In addition, the industry struggles with inefficient resource allocation, where marketing efforts are dispersed across a broad audience, including individuals who may not be actively considering a car purchase.

In light of these challenges, there is a clear need for innovative solutions that leverage data analytics and machine learning to revolutionize marketing strategies in the automotive sector.

2.2 → Problem Statement Definition

The automotive industry faces a critical challenge in its traditional marketing approaches, characterized by generic strategies that lack the precision needed to engage potential car buyers effectively. The current strategies often overlook the nuanced factors influencing individual car purchasing decisions, such as lifestyle, preferences, and financial considerations.

Furthermore, the industry has been slow to embrace advanced technologies, particularly machine learning, in predicting customer behavior for car purchases. The under utilization of data-driven decision-making tools hampers the industry's ability to adapt to evolving consumer trends, resulting in a lack of agility and responsiveness in marketing strategies.

Our project seeks to address this multifaceted problem by developing an innovative machine learning solution that leverages customer data to predict car purchases accurately. By integrating advanced algorithms and thorough data preprocessing techniques, the project aims to provide a personalized and data-driven approach to marketing in the automotive sector. The goal is to revolutionize traditional strategies, offering insights for tailored marketing approaches that enhance customer experiences, optimize resource allocation, and ultimately drive data-powered decisions in the industry.

4.REQUIREMENT ANALYSIS

4.1 → Functional requirement

The functional requirements of the predictive car purchase model encompass a comprehensive set of features designed to ensure accurate predictions, user-friendliness, and seamless integration. These functionalities are integral to the success of the project and align with the overarching goals of transforming marketing strategies in the automotive industry.

Data Input and Processing:

- *Objective:* Allow users to input their demographic information, including age, income, and relevant historical purchase patterns.

Machine Learning Model:

- *Objective:* Develop and implement advanced machine learning algorithms for predicting car purchase likelihood.

User Interface:

- *Objective:* Create a user-friendly interface for seamless interaction.

Integration with Existing Systems:

- *Objective:* Ensure the model seamlessly integrates with existing marketing and customer relationship management systems.

Scalability:

- *Objective:* Design the system to handle varying data volumes and user loads.

4.2 → Non-Functional requirements

The non-functional requirements of the predictive car purchase model focus on qualities that are critical for the success, usability, and performance of the system. These aspects ensure that the model not only meets its functional objectives but also operates in a manner that aligns with industry standards and user expectations.

Performance:

- *Objective:* The system should provide real-time predictions within an acceptable response time.

Scalability:

- *Objective:* The system should be scalable to accommodate an increasing volume of users and data.

Reliability:

Objective: Ensure the model's reliability and stability under varying conditions.

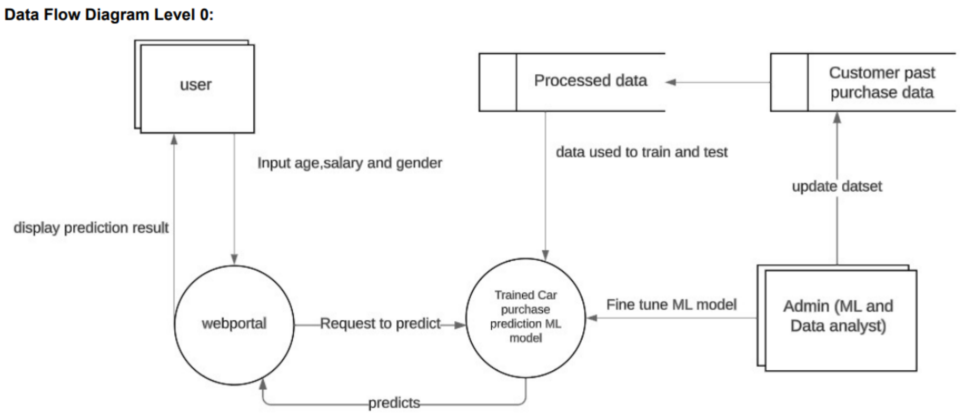
Usability:

- *Objective:* Design an interface that is user-friendly and accessible.

5. Project Design

5.1 → Data flow diagrams and user stories.

Data flow diagram


Data Flow Diagram Level 0:
user
Input age,salary and gender
display prediction result
webportal
Request to predict
predicts
Processed data
data used to train and test
Customer past purchase data
update datset
Trained Car purchase prediction ML model
Fine tune ML model
Admin (ML and Data analyst)

User stories describe the interactions between the user and the system, capturing the intended functionalities from the user's perspective.

1. **As a user, I want to input my demographic information easily so that the model can provide an accurate prediction of my likelihood to make a car purchase.**
   - *Acceptance Criteria:* The user interface should have clear input fields for age, income, and historical purchase patterns, and the input process should be intuitive.
2. **As a marketing manager, I want the machine learning model to provide accurate and reliable predictions, enabling targeted marketing strategies.**
   - *Acceptance Criteria:* The machine learning model should undergo rigorous training and validation, achieving a predefined level of predictive accuracy.

6. PROJECT PLANNING AND SCHEDULING

The steps followed to prepare this project include:

i) Downloading the dataset
ii) Data Preprocessing
iii) Selecting the appropriate ML model based on the performance metrics
iv) Creating a user interface using HTML to enter data of the required features for prediction of the results
v) Deploying the ML model using Flask

7. CODING AND SOLUTIONING

Since the ML model has to predict whether a person can purchase a car or not based on features such as age and salary, applying a classification algorithm would be appropriate for this project.

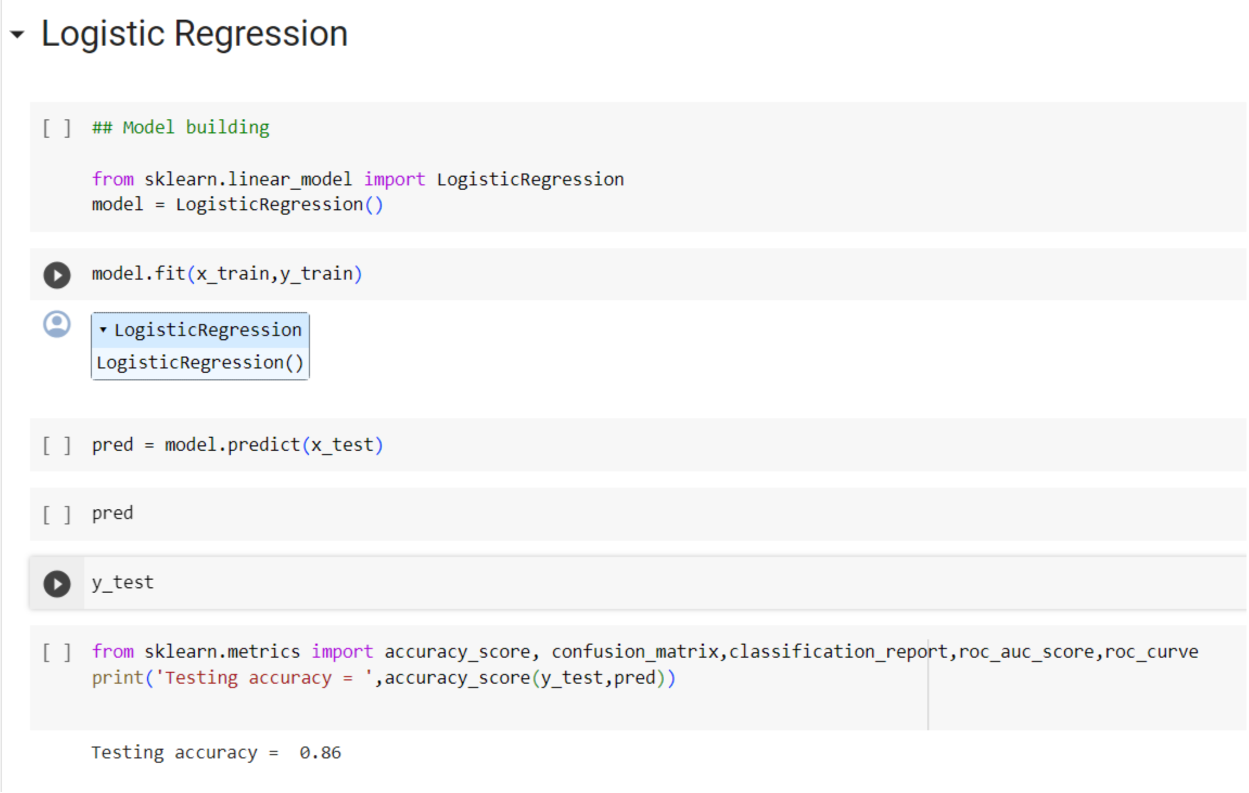
We have used a dataset and have trained classification models such as Logistic Regression, Decision Tree,Random Forest algorithms and Kernel SVM classifier algorithm.

8. PERFORMANCE TESTING

8.1 Performance Metrics

Out of the 4 ML models we have trained, the testing accuracy of each are as mentioned below:

Logistic Regression : 0.86
Random Forest Classifier: 0.91
Decision Tree Classifier: 0.925
Kernel SVM : 0.865



▾ Logistic Regression

```
## Model building

from sklearn.linear_model import LogisticRegression
model = LogisticRegression()
```

```
model.fit(x_train,y_train)
```

```
▾ LogisticRegression
LogisticRegression()
```

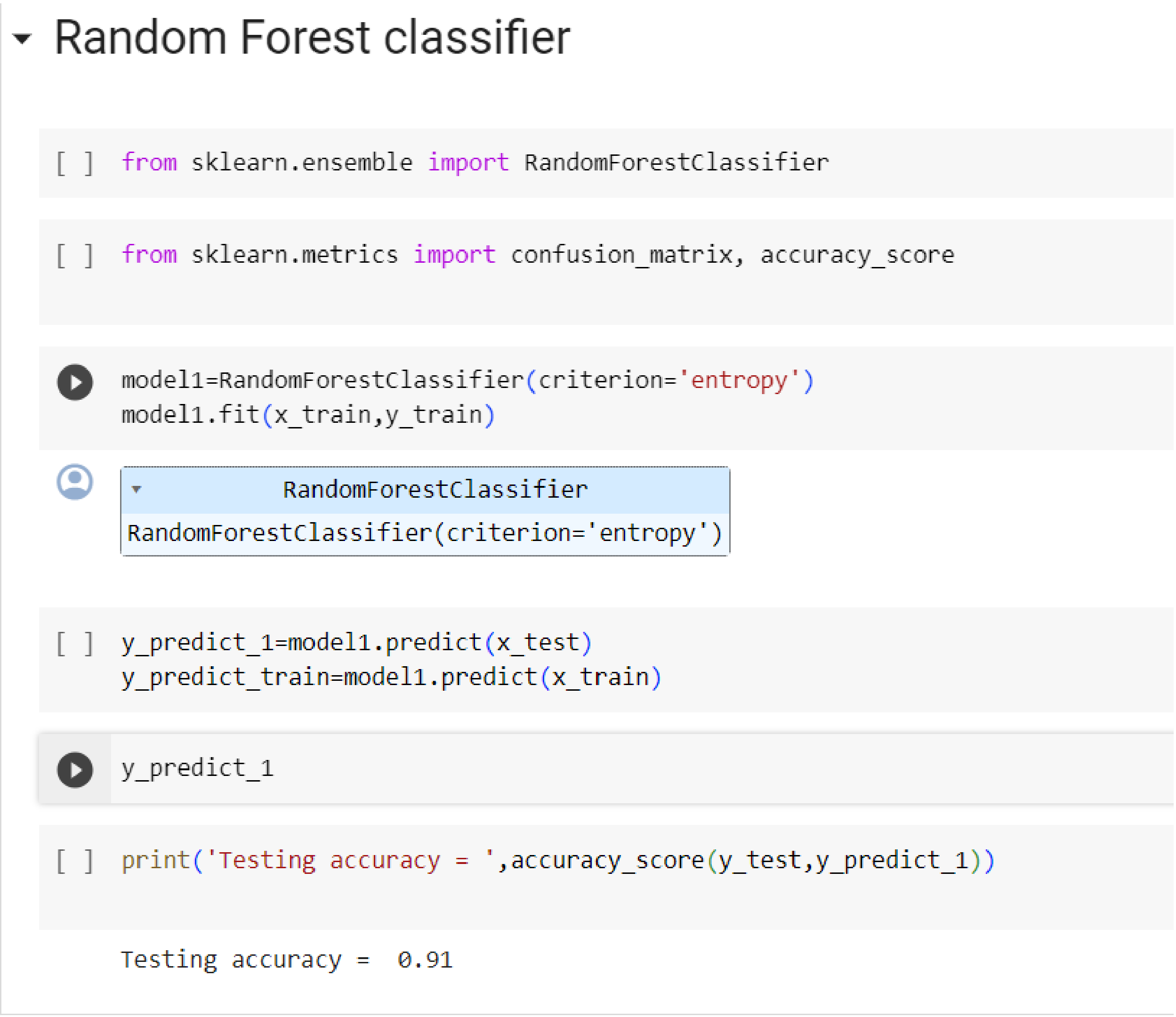
```
pred = model.predict(x_test)
```

```
pred
```

```
y_test
```

```
from sklearn.metrics import accuracy_score, confusion_matrix,classification_report,roc_auc_score,roc_curve
print('Testing accuracy = ',accuracy_score(y_test,pred))
```

```
Testing accuracy =  0.86
```



Random Forest classifier

```
[ ] from sklearn.ensemble import RandomForestClassifier
```

```
[ ] from sklearn.metrics import confusion_matrix, accuracy_score
```

```
model1=RandomForestClassifier(criterion='entropy')
model1.fit(x_train,y_train)
```

```
▾          RandomForestClassifier
RandomForestClassifier(criterion='entropy')
```

```
[ ] y_predict_1=model1.predict(x_test)
    y_predict_train=model1.predict(x_train)
```

```
y_predict_1
```

```
[ ] print('Testing accuracy = ',accuracy_score(y_test,y_predict_1))
```

```
Testing accuracy =  0.91
```

Decision Tree

```
[ ]  from sklearn.tree import DecisionTreeClassifier
```

```
[ ]  model2=DecisionTreeClassifier(max_depth=4,splitter='best',criterion='entropy')
```

```
[ ]  model2.fit(x_train,y_train)
```

```
                  DecisionTreeClassifier
DecisionTreeClassifier(criterion='entropy', max_depth=4)
```

```
[ ]  y_predict_2=model2.predict(x_test)
```

```
[ ]  y_test
```

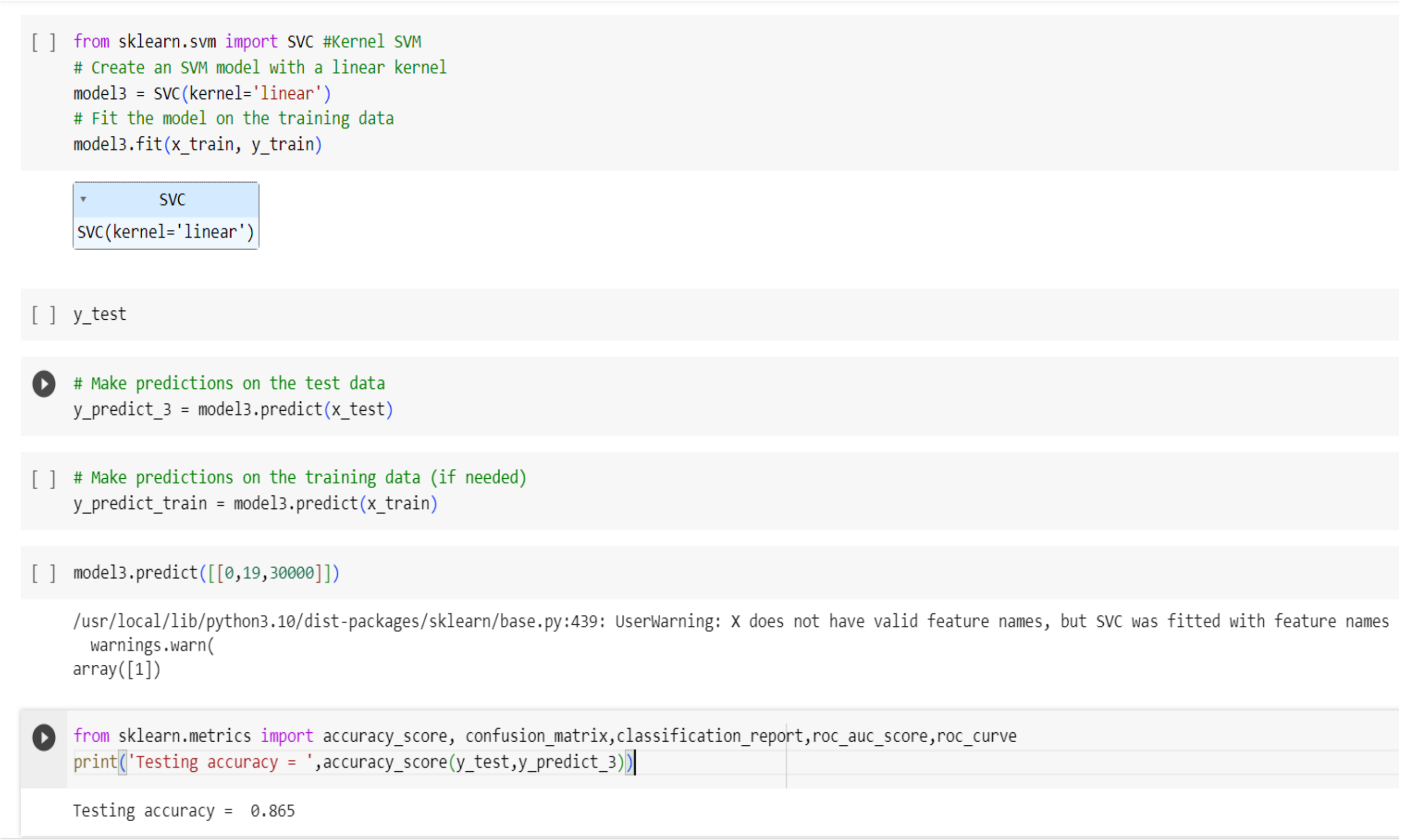
```
[ ]  y_predict_2
```

```
[ ]  y_predict_train=model2.predict(x_train)
```

```
[ ]  from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
     print('Testing Accuracy = ', accuracy_score(y_test,y_predict_2))
```

```
Testing Accuracy =  0.925
```

Kernel SVM



```
[ ] from sklearn.svm import SVC #Kernel SVM
    # Create an SVM model with a linear kernel
    model3 = SVC(kernel='linear')
    # Fit the model on the training data
    model3.fit(x_train, y_train)
```

```
    ▾        SVC
    SVC(kernel='linear')
```

```
[ ] y_test
```

```
    # Make predictions on the test data
    y_predict_3 = model3.predict(x_test)
```

```
[ ] # Make predictions on the training data (if needed)
    y_predict_train = model3.predict(x_train)
```

```
[ ] model3.predict([[0,19,30000]])

    /usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
      warnings.warn(
    array([1])
```

```
    from sklearn.metrics import accuracy_score, confusion_matrix,classification_report,roc_auc_score,roc_curve
    print('Testing accuracy = ',accuracy_score(y_test,y_predict_3))

    Testing accuracy =  0.865
```

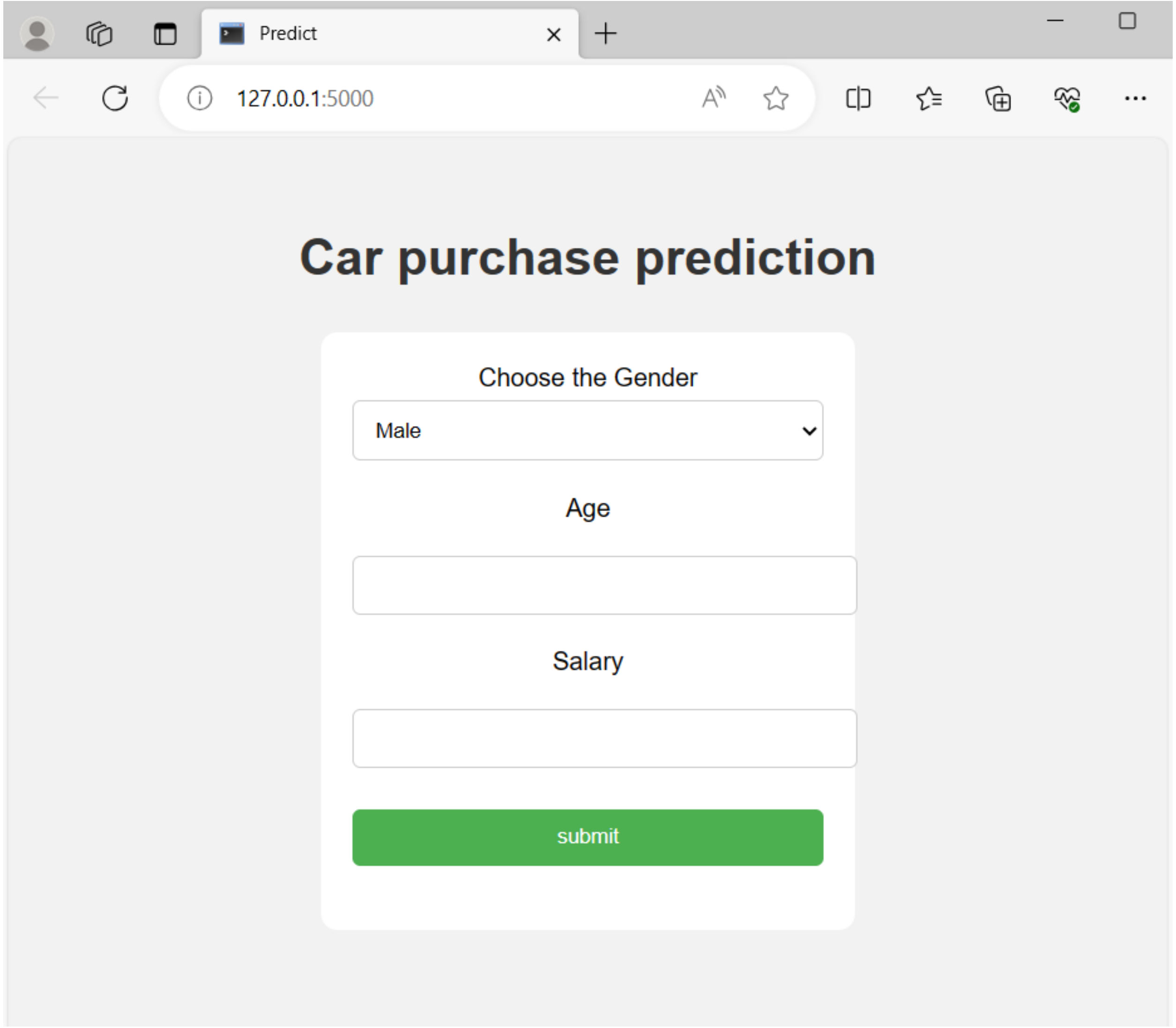
Hence, based on the testing accuracy of all the models tried on this dataset, Decision Tree algorithm has been chosen for this project.

9. RESULTS

The ML model has been deployed using Flask. Different input combinations have been provided to the web application and the outputs have been recorded.

9.1 Output screenshots

Front page



Car purchase prediction

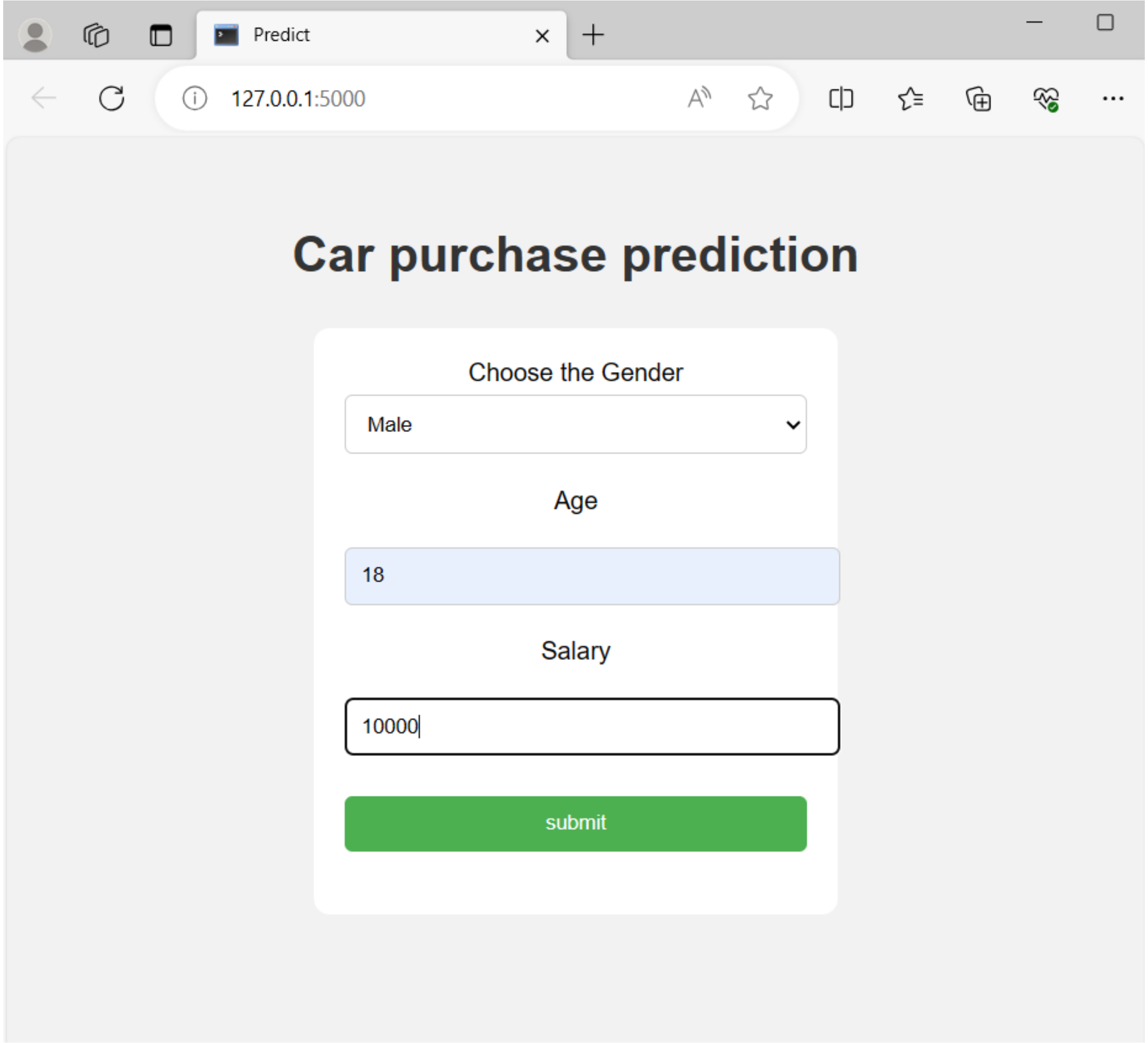
Choose the Gender

Male

Age

Salary

INPUT:
Gender: Male
Age: 18
Salary: 10000



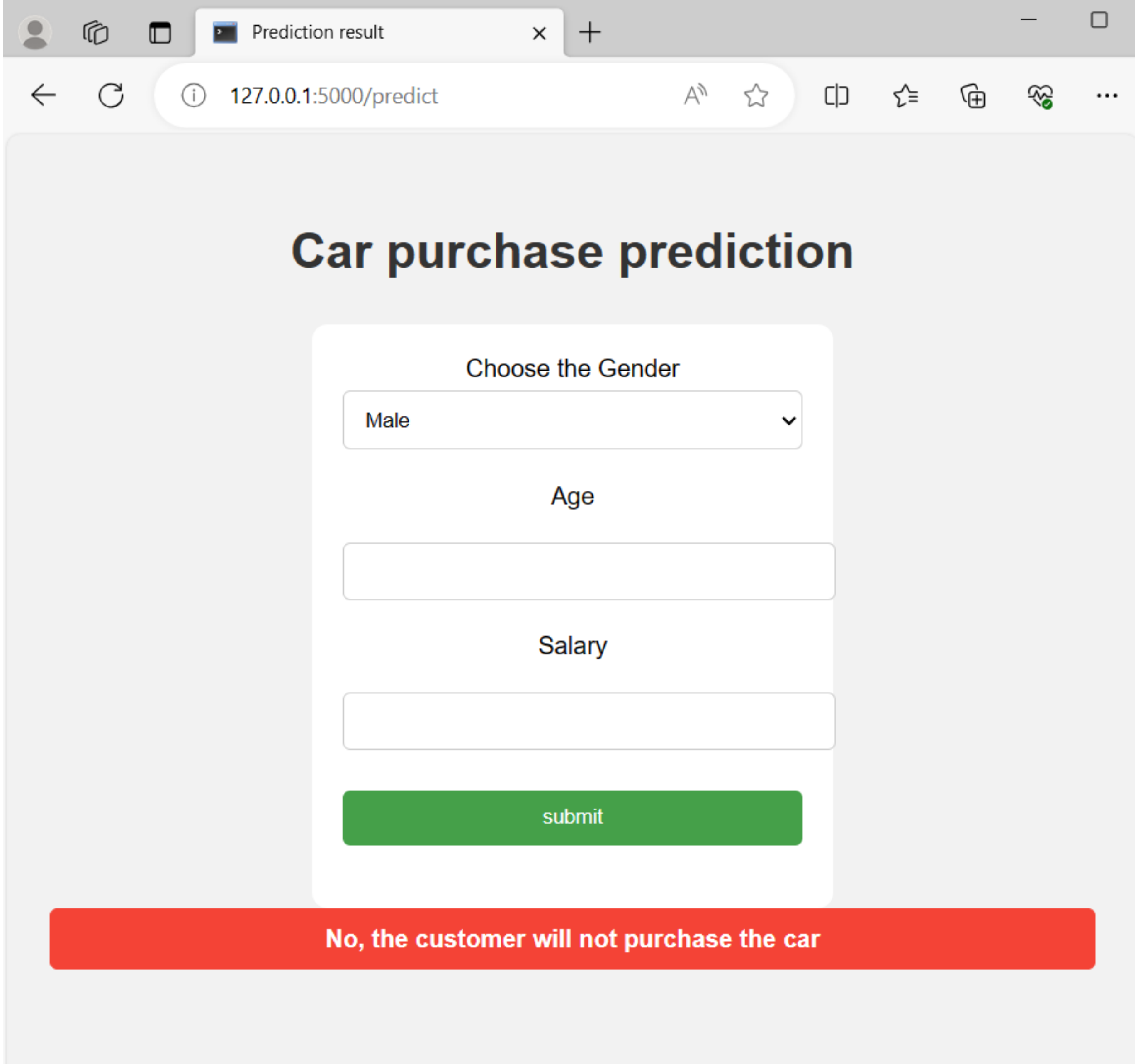
Car purchase prediction

Choose the Gender
Male

Age
18

Salary
10000

OUTPUT



Car purchase prediction

Choose the Gender

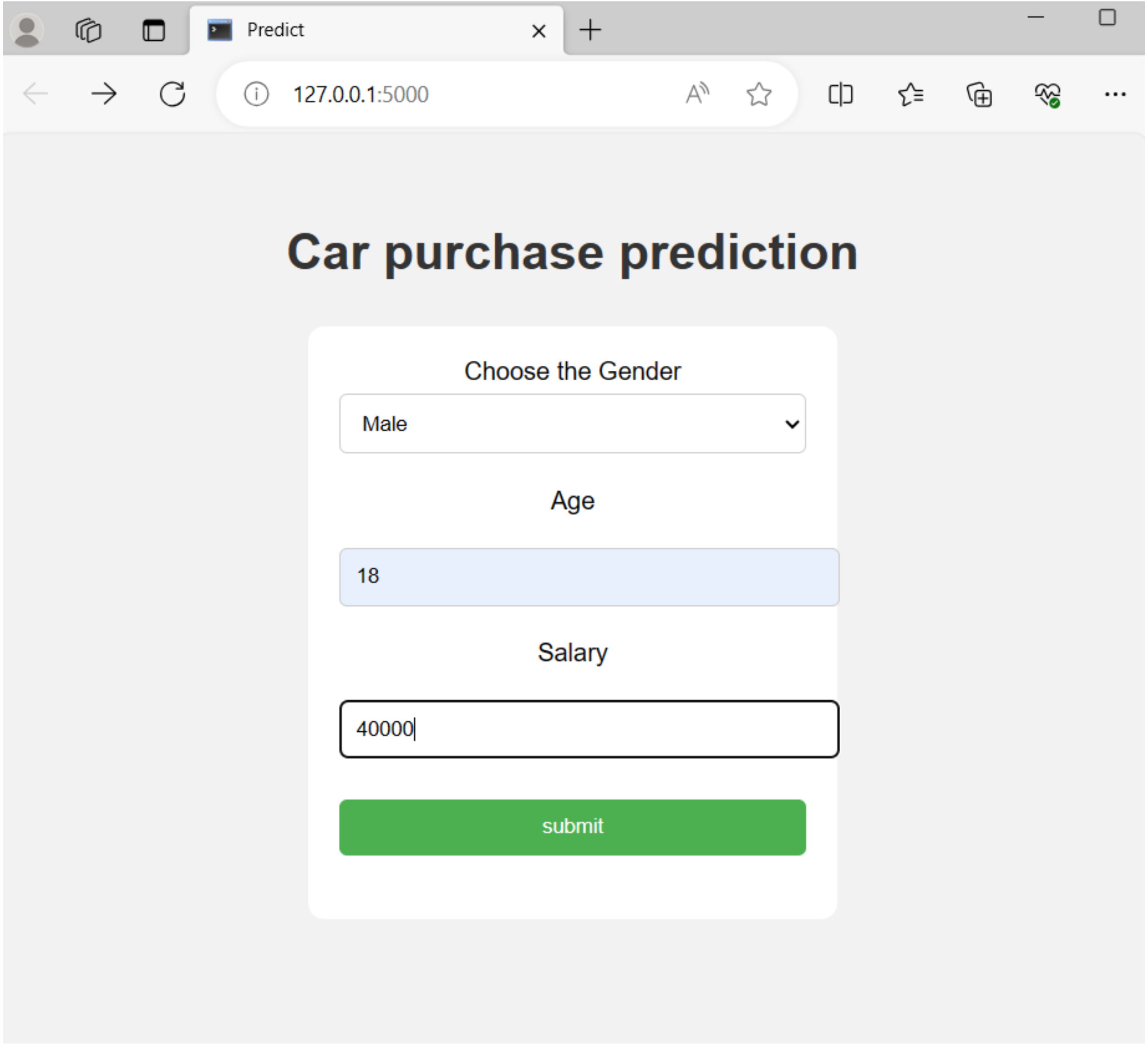
Male

Age

Salary

No, the customer will not purchase the car

INPUT
Gender: Male
Age: 18
Salary: 40000



Car purchase prediction

Choose the Gender

Male

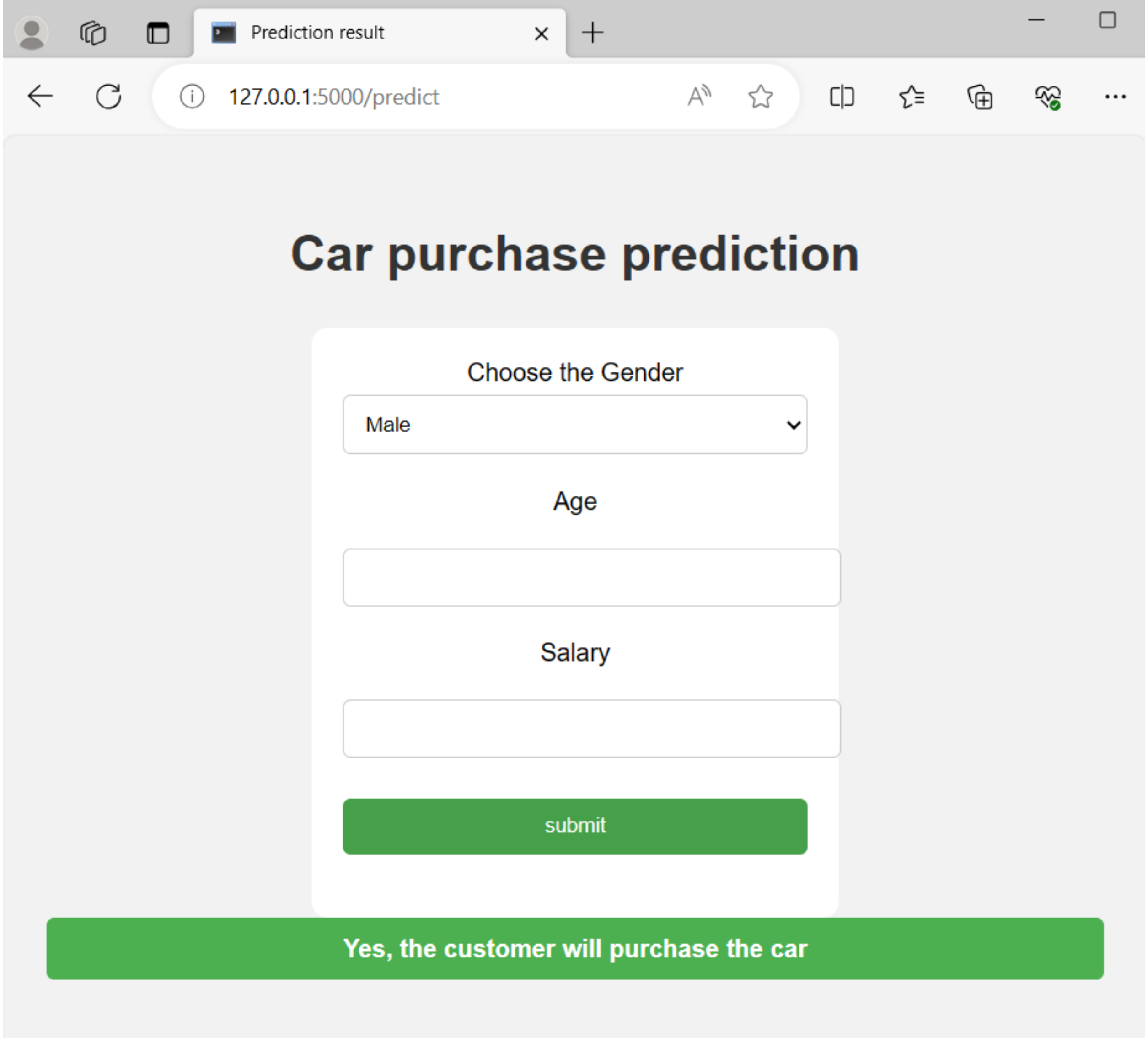
Age

18

Salary

40000

OUTPUT:



Car purchase prediction

Choose the Gender

Male

Age

Salary

Yes, the customer will purchase the car

10. Advantages and Disadvantages

Advantages.

1.Accurate Predictions:

The predictive car purchase model leverages advanced machine learning algorithms, leading to highly accurate predictions based on user demographic data.

2.Personalized Marketing:

By considering individual factors such as age, income, and historical purchase patterns, the model enables personalized marketing approaches.

3.Data-Driven Decision Making:

The project promotes data-driven decision-making in the automotive industry. Businesses can make informed choices regarding marketing strategies, resource allocation, and customer engagement based on the insights provided by the model.

4. Revolutionizing Marketing Strategies:

The project introduces a groundbreaking application of machine learning to drive data-powered decisions in the automotive sector.

Disadvantages.

1. **Dependence on Data Quality:**

The accuracy of predictions is highly dependent on the quality and relevance of the input data. Inaccuracies or biases in the data can affect the reliability of the model's predictions.

2. **Privacy Concerns:**

The collection and processing of user demographic data raise privacy concerns. Users may be hesitant to share personal information, and businesses must ensure compliance with privacy regulations to address these concerns adequately.

3. **Initial Implementation Cost:**

Developing and implementing the machine learning model, as well as integrating it into existing systems, may involve initial costs for businesses.

4. **Need for Continuous Training:**

To maintain high predictive accuracy, the machine learning model requires continuous training and updates. This demands ongoing resources and efforts to keep the model relevant and effective over time.

11. Conclusion

In conclusion, the development and implementation of the predictive car purchase model represent a significant stride towards revolutionizing the automotive industry's marketing strategies. Through the integration of advanced machine learning algorithms and thorough data preprocessing, our project has successfully addressed the existing challenges of traditional approaches.

The project's commitment to data-driven decision-making empowers automotive businesses to make informed choices, optimizing resource allocation and enhancing overall efficiency. The user-friendly interface facilitates seamless interactions, allowing users to input their information effortlessly and receive precise purchase likelihoods, thereby improving the overall customer experience.

However, it is crucial to acknowledge and address the potential challenges associated with data quality, privacy concerns, and the need for continuous model training. These considerations underscore the importance of ongoing vigilance and proactive measures to ensure the model's reliability and ethical use.In essence, our project signifies a groundbreaking application of machine learning in the automotive sector, propelling the industry towards a future where data-powered decisions drive marketing strategies

As the model evolves and adapts to emerging trends, it is poised to be a catalyst for positive change, offering actionable insights for tailored marketing strategies and ultimately reshaping the way the automotive industry engages with its customers.

12. Future Scope

The success of the predictive car purchase model opens up exciting avenues for future development and expansion, presenting opportunities to further enhance its impact on the automotive industry. The following areas represent key aspects of the future scope:

1.Integration of Additional Features:

Expand the model by incorporating additional relevant features beyond age, income, and historical purchase patterns. This may include socio-economic factors, regional trends, and emerging consumer behaviors, providing a more comprehensive understanding of individual preferences.

2. Advanced Machine Learning Techniques:

Explore and implement more advanced machine learning techniques and algorithms. Continuous research and development in the field can lead to innovations that improve predictive accuracy, model interpret ability

3. Real-time Predictions:

Work towards achieving real-time predictions to enhance the model's responsiveness. The ability to provide instant purchase likelihood estimations can further streamline decision-making processes for both users and businesses.

4. Global Market Expansion:

Extend the application of the model to a global scale, considering regional variations and market dynamics.

5.Expandability and Transparency:

Enhance the model's interpret ability and transparency. Develop methodologies to explain the rationale behind predictions, making it easier for businesses and users to understand the factors influencing the model's decisions.

6. User Feedback Mechanism:

Introduce a user feedback mechanism to gather insights and improve the user experience continuously. Incorporating user input and preferences can contribute to the refinement and evolution of the model.

7. Ethical Considerations and Bias Mitigation:

Place a strong emphasis on ethical considerations and bias mitigation. Implement measures to detect and address any biases in the model, ensuring fair and equitable predictions for users from diverse backgrounds.

As the predictive car purchase model evolves, its future scope extends beyond the confines of its initial implementation, presenting an exciting journey towards innovation, adaptability, and continued positive impact on the automotive industry.

13. APPENDIX

Source code

ML model using Decision Tree algorithm (Here, file name is Bestmodel.py)

```
import numpy as np

import pandas as pd



df = pd.read_csv('car_data.csv')

df.head()



df.shape
df.Purchased.value_counts()

df.isnull().sum()

df = df.drop(columns = ['User ID'],axis = 1)

df.head()

from sklearn.preprocessing import LabelEncoder
le =LabelEncoder()

df.Gender = le.fit_transform(df.Gender)

df.head()

import matplotlib.pyplot as plt
import seaborn as sns

sns.displot(df.Age)
```

```
df.describe()

sns.boxplot(df.AnnualSalary)

sns.boxplot(df.Age)

df.corr()

sns.heatmap(df.corr(),annot =True)

df.corr().Purchased.sort_values(ascending=False)

df.head()

## X and y split

X =df.drop(columns =['Purchased'],axis =1)

X.head()
```

```
y =df.Purchased

y.head()


from sklearn.preprocessing import MinMaxScaler

scale =MinMaxScaler()


scaled_x = pd.DataFrame(scale.fit_transform(X),columns =X.columns)

scaled_x.head()


# Train test split


from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test = train_test_split(scaled_x,y,test_size = 0.2,random_state = 0)


x_train.shape


x_test.shape


## Model building

from sklearn.tree import DecisionTreeClassifier
```

```
model1=DecisionTreeClassifier(max_depth=4,splitter='best',criterion='entropy')

model1.fit(x_train,y_train)

y_predict_1=model1.predict(x_test)

y_predict_1

y_predict_train=model1.predict(x_train)


import pickle


pickle.dump(model1,open('car_.pkl','wb'))
```

Code for Flask app.py

```
from flask import Flask, render_template, request
import pickle
import numpy as np

# Load the trained model
model1 = pickle.load(open('car_.pkl', 'rb'))

app = Flask(__name__)

@app.route('/')
def start():
   return render_template('index.html')

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

```
def predict():
    rd = int(request.form['rd'])
    as_val = int(request.form['as'])
    s = request.form['s']

    # Add age and salary limits
    age_limit = 18  # Minimum age limit
    salary_limit = 14000  # Minimum salary limit

    # Convert state to binary (Male=1, Female=0)
    if s == 'cal':
        s = 1
    elif s == 'flo':
        s = 0

    # Check if input values meet the specified limits
    if rd >= age_limit and as_val >= salary_limit:
        input_data = [[s, rd, as_val]]
        prediction = model1.predict(input_data)
        prediction_bool = bool(prediction[0])
    else:
        prediction_bool = False  # If input values do not meet the limits

    return render_template('result.html', y=prediction_bool)



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

Code for index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <link rel="stylesheet" type="text/css" href="/static/styles.css">
   <title>Predict</title>
</head>
<body>
   <h1>Car purchase prediction</h1>
   <form action ="/predict" method = "post">

   <label for ="Gender">Choose the Gender</label>
   <select name = "s">
   <option value ="cal">Male</option>
   <option value ="flo">Female</option>
   </select>
   <p>Age</p>
   <p><input type ="text" name ="rd"/></p>
   <p>Salary</p>
   <p><input type ="text" name ="as"/></p>
   <p><input type ="submit" value ="submit"/></p>

</form>
</body>
```

```
</html>
```

Code for result.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" type="text/css" href="/static/styles.css">
    <title>Prediction result</title>
</head>
<body>
    <h1>Car purchase prediction</h1>
    <form action ="/predict" method = "post">
    <label for ="Gender">Choose the Gender</label>
    <select name = "s">
    <option value ="cal">Male</option>
    <option value ="flo">Female</option>
    </select>
    <p>Age</p>
    <p><input type ="text" name ="rd"/></p>
    <p>Salary</p>
    <p><input type ="text" name ="as"/></p>
    <p><input type ="submit" value ="submit"/></p>
</form>
<div class="message-box {% if y %}success{% else %}error{% endif %}">
        <b>{% if y %}Yes, the customer will purchase the car{% else %}No, the customer will not purchase the car{% endif %}</b>
```

```
</div>
</body>
</html>
```

Code for styles.css

```
/* styles.css */

body {
  font-family: Arial, sans-serif;
  background-color: #f2f2f2;
  text-align: center;
  padding: 20px; /* Add padding to the body to center the content */
 }

 h1 {
  color: #333;
  margin: 30px 0; /* Add margin at the top and bottom */
 }

 form {
  background-color: #fff;
  border-radius: 10px;
  padding: 20px;
  width: 300px;
  margin: 0 auto;
 }
 /* Styles for the success message box */
.message-box.success {
 background-color: #4CAF50; /* Green background color for success message */
```

```
  color: white; /* White text color for success message */
  padding: 10px; /* Padding around the message text */
  border: 1px solid #4CAF50; /* Green border for success message */
  border-radius: 5px; /* Rounded corners for the message box */
}

/* Styles for the error message box */
.message-box.error {
  background-color: #f44336; /* Red background color for error message */
  color: white; /* White text color for error message */
  padding: 10px; /* Padding around the message text */
  border: 1px solid #f44336; /* Red border for error message */
  border-radius: 5px; /* Rounded corners for the message box */
}

  select, input {
    width: 100%;
    padding: 10px;
    margin: 5px 0;
    border: 1px solid #ccc;
    border-radius: 5px;
  }

  input[type="submit"] {
    background-color: #4caf50;
    color: #fff;
    border: none;
    padding: 10px 20px;
    border-radius: 5px;
  cursor: pointer;
 }

 input[type="submit"]:hover {
  background-color: #45a049;
 }

 h2 {
  margin: 20px 0; /* Add margin at the top and bottom of h2 elements */
  color: #333;
 }
```

Demo link

https://drive.google.com/file/d/1_VxiF2XCDZl7n-ke_ZWr380iZi0h_DaB/view?usp=sharing