Name: Dhanashri Jagadale

Batch Number: LISUM16

Submission Date: 11/03/2023

Submitted URL:

Deployment on Flask in python

Step 1: Import the necessary python libraries to run the model.

```
import pandas as pd
import numpy as np
import pickle
#Importing the python libraries
```

Step 2: Reading the dataset

```
dhan= pd.read_csv('iris.data') #reading data
```

Step 3: Creating X and Y array

```
X = np.array(dhan.iloc[:, 0:4])
y = np.array(dhan.iloc[:, 4:])
```

Step 4: Training the machine learning model

```
#Training the machine learning model
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
y = le.fit_transform(y.reshape(-1))

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

from sklearn.svm import SVC
sv = SVC(kernel='linear').fit(X_train,y_train)
```

Step 5: Saving the model using pickle library

```
pickle.dump(sv, open('iri.pkl', 'wb')) #Saving the model using Pickle Library
```

Step 6: Deployment of the model on flask

The app1.py file

```python
import numpy as np

model = pickle.load(open('iri.pkl', 'rb'))

app = Flask(__name__)



@app.route('/')
def man():
    return render_template('home.html')


@app.route('/predict', methods=['POST'])
def home():
    data1 = request.form['a']
    data2 = request.form['b']
    data3 = request.form['c']
    data4 = request.form['d']
    arr = np.array([[data1, data2, data3, data4]])
    pred = model.predict(arr)
    return render_template('after.html', data=pred)


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

```python
from flask import Flask, render_template

app = Flask(__name__)


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

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

The basic.py file

```html
<html>
    <body bgcolor=#d4a3ae>

        <center>

        <h1> IRIS FLOWER DETECTION </h1><br>

        <form method="POST", action="{{url_for('home')}}">
           <b> First value :  <input type="text", name='a', placeholder="enter 1"> <br><br>
            Second value :  <input type="text", name='b', placeholder="enter 2"> <br><br>
            Third value :  <input type="text", name='c', placeholder="enter 3"> <br><br>
            Fourth value : <input type="text", name='d', placeholder="enter 4"> <br><br><br></b>
            <input type="submit" , value='predict!' >
        </form>

        <img src='static\flower1.jpg' alt="flower">

    </center>

    </body>
</html>
```

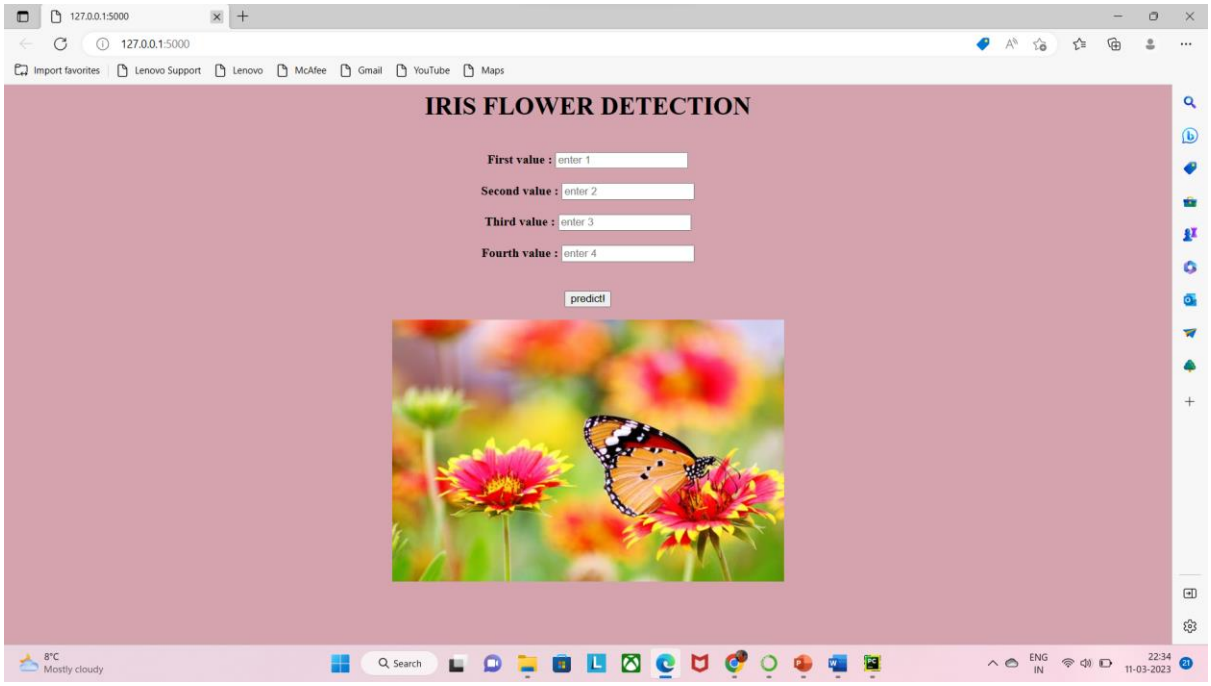
The html code for home page

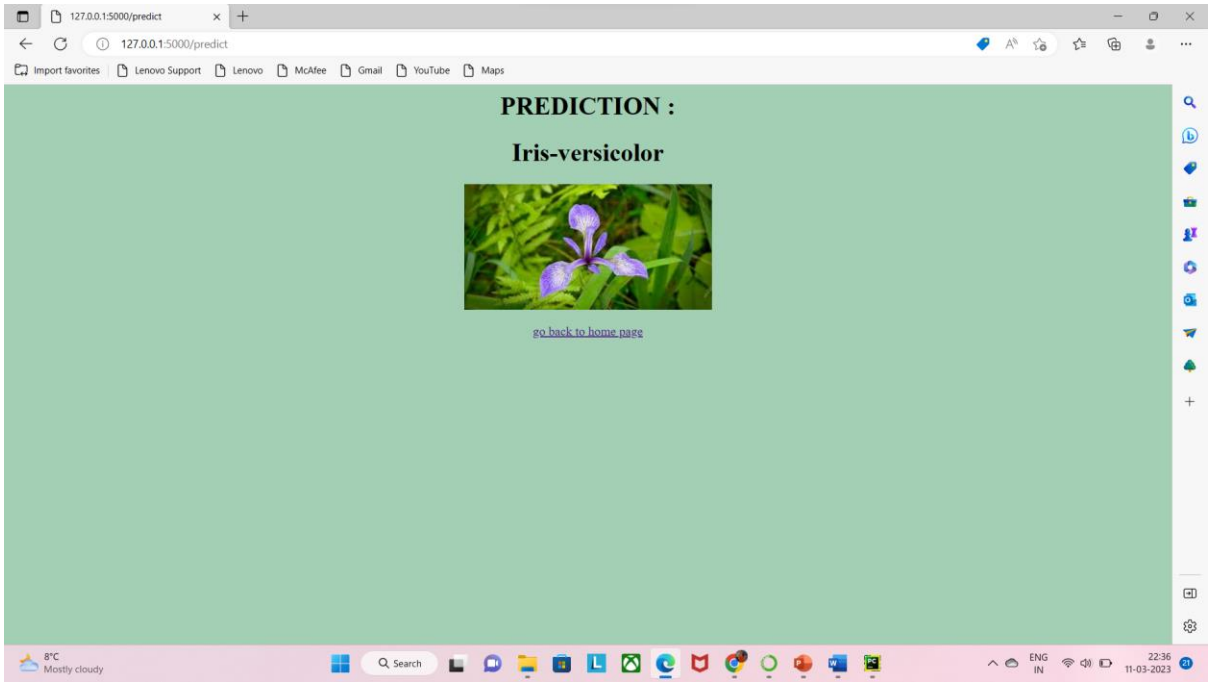
```
<html>

<body bgcolor=#a3cfb4>

    <center>

        <h1>  PREDICTION :  </h1>

    {%if data == 0%}
    <h1>Iris-setosa</h1>
    <img src='static\setosa.jpg'>

    {%else%}
    <h1>Iris-versicolor</h1>
    <img src='static\verci.jpg'>

    {%endif%}

        <br><br>
    <a href='/'>go back to home page</a>

    </center>

</body>

</html>
```

The Html code for Predict page

Step 6: Result after running

```
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
 * Restarting with watchdog (windowsapi)
C:\Users\dhana\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\base.py:318:
https://scikit-learn.org/stable/model_persistence.html#security-maintainability-limitations
```


127.0.0.1:5000
127.0.0.1:5000
Import favorites
Lenovo Support
Lenovo
McAfee
Gmail
YouTube
Maps
IRIS FLOWER DETECTION
First value :
enter 1
Second value :
enter 2
Third value :
enter 3
Fourth value :
enter 4
predictl
8°C
Mostly cloudy
Search
ENG
IN
22:34
11-03-2023


127.0.0.1:5000/predict
127.0.0.1:5000/predict
Import favorites
Lenovo Support
Lenovo
McAfee
Gmail
YouTube
Maps
PREDICTION :
Iris-versicolor
go back to home page
8°C
Mostly cloudy
Search
ENG
IN
22:36
11-03-2023