

Deployment on Flask(Week4)

Name: Thanuja Modiboina

Batch code: LISUM19

Submission date: 28/03/2023

Submitted to: Data Glacier

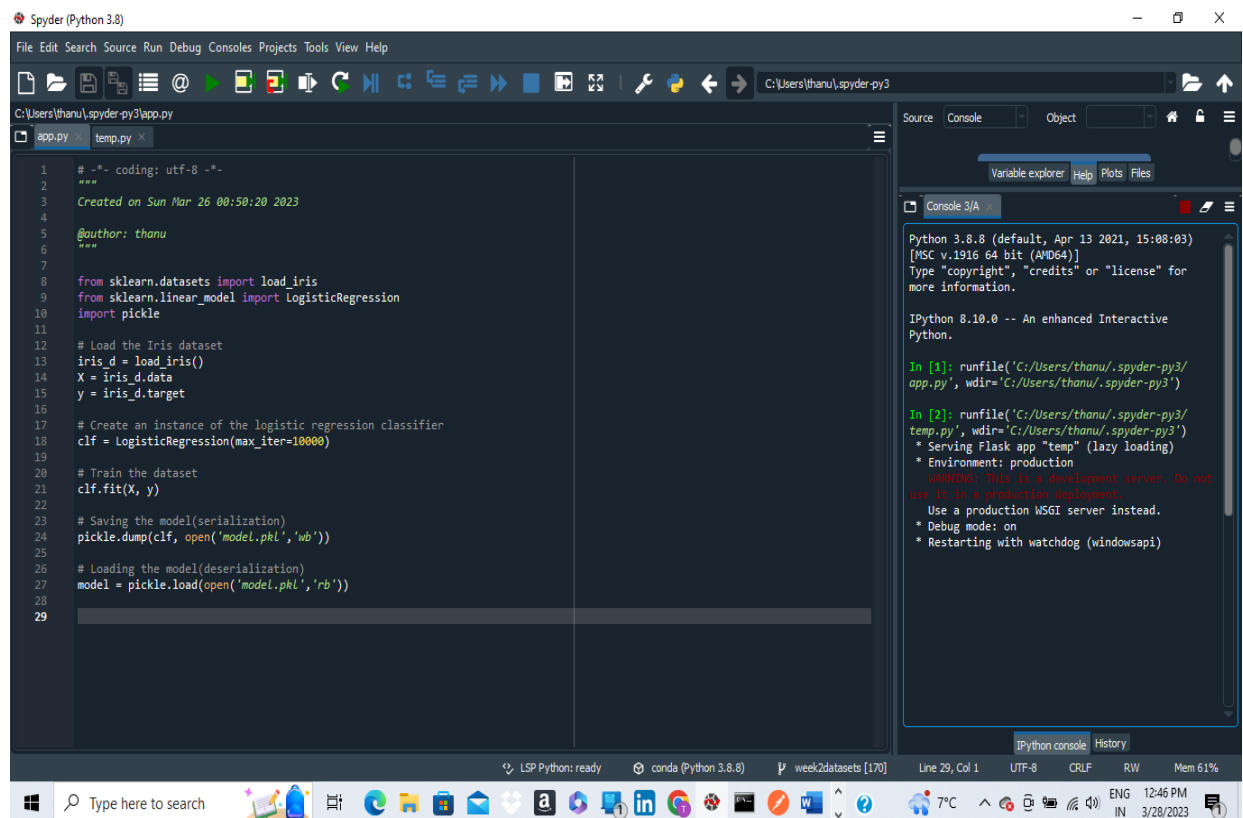
Steps of deployment:

1. Select toy data.

Here, we are selecting 'iris data'. This data consists of 150 observations and 4 features(sepal length, sepal width, petal length, and petal width). There are 3 species of iris flowers Setosa, Versicolor, and Virginica. These are the target variables.

2. Train and save the model.

Below is the code for training the model to predict species of the flower based on its four features and saving it.

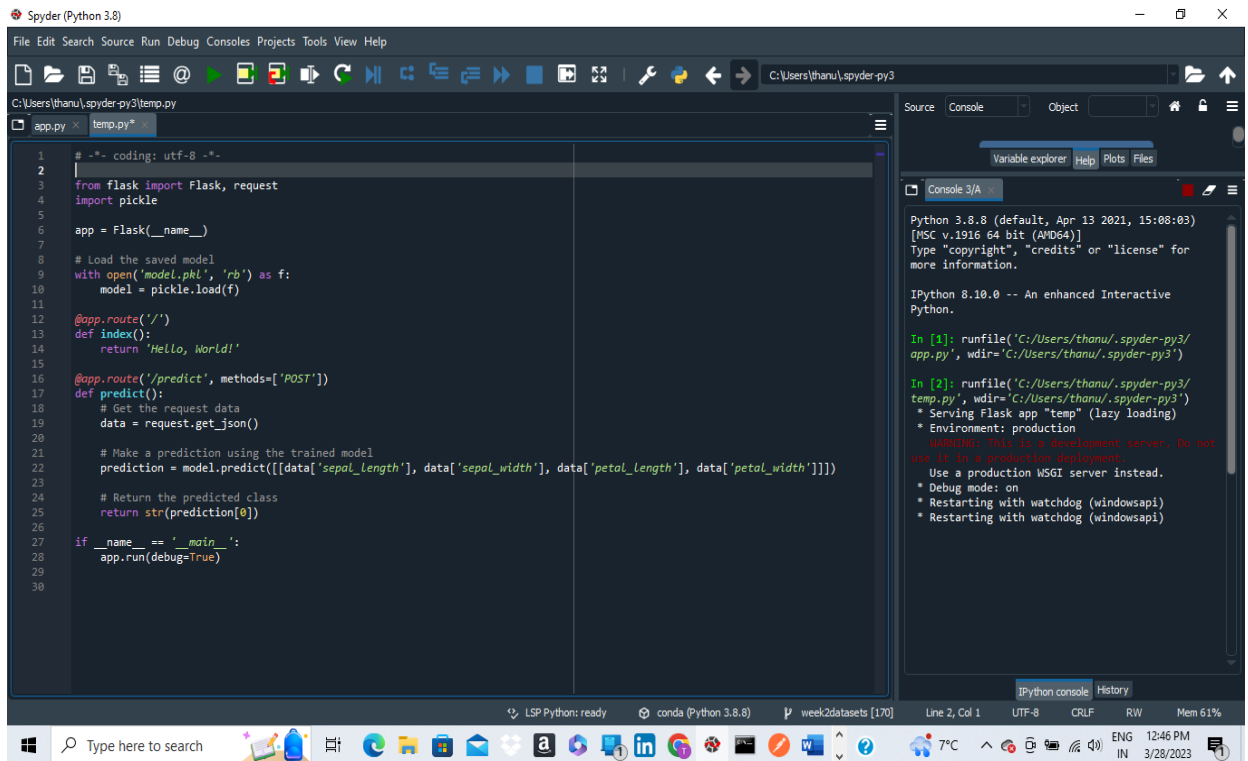


```
1  #- coding: utf-8 -*-
2  """
3  Created on Sun Mar 26 00:50:20 2023
4
5  @author: thanu
6  """
7
8  from sklearn.datasets import load_iris
9  from sklearn.linear_model import LogisticRegression
10 import pickle
11
12 # Load the Iris dataset
13 iris_d = load_iris()
14 X = iris_d.data
15 y = iris_d.target
16
17 # Create an instance of the logistic regression classifier
18 clf = LogisticRegression(max_iter=10000)
19
20 # Train the dataset
21 clf.fit(X, y)
22
23 # Saving the model(serialization)
24 pickle.dump(clf, open('model.pkl', 'wb'))
25
26 # Loading the model(deserialization)
27 model = pickle.load(open('model.pkl', 'rb'))
28
29
```

The screenshot shows the Spyder Python IDE interface. The main editor displays the Python code for training a logistic regression model on the Iris dataset. The code includes imports for sklearn, loading the dataset, training the model, and saving it as a pickle file. The console on the right shows the execution output, including the version of Python and the environment details.

3. Deploy the model on a flask.

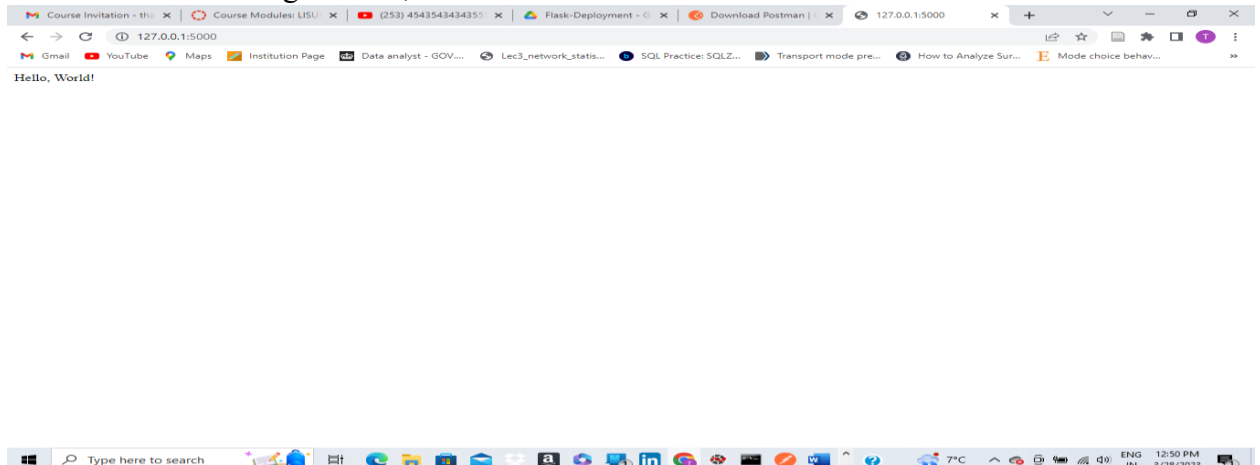
We are creating the flask web app to deploy the model here.



- The code for deployment is run in the command prompt. Then copy the URL of the website and paste it into the web browser.



We can see the message 'Hello, World'.



5. To make a prediction, we used Postman to send a POST request to <http://127.0.0.1:5000/predict> with the JSON data containing the features. We got the result as 1 for the given data in JSON.

