

Name : Mohamed Sayed Hassan

Batch Code: LISUM02

Submission Date: 8/23/2021

Submitted to: Data glacier

Flask Deployment

1-choose the dataset:

I worked on **Iris** dataset. It includes three iris species with 50 samples each as well as some properties about each flower. One flower species is linearly separable from the other two, but the other two are not linearly separable from each other.

2- Train the model:

I used the gaussian model to learn from the data.

```
1 # Load libraries
2 from sklearn import datasets
3 from sklearn.naive_bayes import GaussianNB
4 import random
5 import pickle
6
7 def binary(a, low, high, x):...
18
19 def split(x, y, percent):...
39
40 def accuracy(test, pred):...
46
47 dataset=datasets.load_iris()
48
49 x=dataset.data
50 y=dataset.target
51
52 x_train,x_test,y_train,y_test=split(x,y,0.7)
53 gnb=GaussianNB()
54 gnb.fit(x_train,y_train)
```

3-Serializing the Model:

Using pickle I serialized the model and saved it in a file called model.

```
55 |
56 | pickle.dump(gnb.open('C:/Users/sika/PycharmProjects/flask/model'.wb'))
57 |
```

4-html page:

Designing the page and the form that should take the features of the iris flower and output the prediction of it is kind.

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <link rel="stylesheet" href="{{ url_for('static', filename='node_modules/bootstrap/dist/css/bootstrap.min.css')}}">
6      <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css')}}">
7      <title>Title</title>
8  </head>
9  <body>
10     <h1>predict the Iris</h1>
11
12     <div>
13         <form action="{{url_for('predict')}}" method="post" >
14
15             <div class="content">
16                 <div class="row">
17                     <div class="offset-5">
18                         <div class="form-group row">
19                             <input type="text" name="sepal width" placeholder="sepal width" />
20                         </div>
21                         <div class="form-group row">
22                             <input type="text" name="sepal width" placeholder="sepal length" />
23                         </div>
24                         <div class="form-group row">
25                             <input type="text" name="sepal width" placeholder="petal width" />
26                         </div>
27                         <div class="form-group row">
28                             <input type="text" name="sepal width" placeholder="petal length" />
29                         </div>
30                         <button type="submit" class="btn btn-primary btn-large btn-block">submit</button>
31                     </div>
32                 </div>
33             </div>
34         </div>
35     </form>
36 </div>
37 <h2>
38     {{prediction}}
39 </h2>
40 </body>
41 </html>
```

5-Deserialization the model:

Now I used pickle again to deserialize the trained model and opened it inside my app to use it in prediction.

6-Flask app:

I made two routes the first 1 is the home page which renders the index.html which contains the form.

The second route is ' /predict ' which shows the prediction of the model

```
1 from flask import Flask,render_template,request
2 import numpy as np
3 import pickle
4
5 app=Flask(__name__)
6 model=pickle.load(open('model','rb'))
7 @app.route('/')
8 def home():
9     return render_template('index.html')
10 @app.route('/predict',methods=['POST'])
11 def predict():
12     features=[float(x) for x in request.form.values()]
13     features=np.array(features)
14     prediction=model.predict(features)
15     if prediction==0:
16         prediction="iris-Setosa"
17     elif prediction == 1:
18         prediction= "Iris-versicolor"
19     elif prediction == 2:
20         prediction = "Iris-virginica"
21     return render_template('index.html',prediction=(prediction))
22
23 if __name__ == "__main__":
24     app.run(port=5000,debug=True)
25
```

7- The web app:

Four features as input to predict the type of the iris flower then after submitting the form the answer will be outputted under the form in a url ending with ' /predict '

predict the Iris

1

2

5

1

submit

predict the Iris

sepal width

sepal length

petal width

petal length

submit

Iris-versicolor