

# Deployment on Flask

Name: Yihsuan Sun

Email: [yihluan.sun88@gmail.com](mailto:yihluan.sun88@gmail.com)

Internship Batch: LISUM09

Submission date: 05/28/2022

Submitted to: Data Glacier Internship

## Data:

toy\_dataset.csv

## Goal:

- Deploy the model on flask
- To predict a person is ill or not

## Step 1: Data preprocessing

- Load the dataset
- Replace city name with number
- Replace gender with number

```
Desktop > FlaskPractice > app.py > ...
1  from flask import Flask, render_template, request, jsonify
2  import pandas as pd
3  from sklearn.ensemble import RandomForestClassifier
4  from sklearn.model_selection import train_test_split
5  from sklearn.preprocessing import StandardScaler
6  import pickle
7  import numpy as np
8
9
10 # Load dataset
11 df = pd.read_csv("toy_dataset.csv")
12 # read first few rows
13 print(df.head())
14
15 # replace
16 df = df.replace(
17     to_replace=['Austin', "Boston", "Dallas", "Los Angeles", "Mountain View", "New York City", "San Diego", "Washington D.C."],
18     value=[1,2,3,4,5,6,7,8,])
19 #print(df.head())
20 #print(df.tail())
21
22 df = df.replace(
23     to_replace=['Male', "Female"],
24     value=[0,1])
25
```

## Step 2:

- Select independent and dependent variables, and then split the data into training and testing
- Feature scaling
- Fit model
- Pickle

```

26 # Select independent and dependent variables\
27 X = df[["City", "Gender", "Age", "Income"]]
28 y = df["Illness"]
29
30 # Splitting the data into training and testing
31 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
32
33 # Feature scaling
34 sc = StandardScaler()
35 X_train = sc.fit_transform(X_train)
36 X_test = sc.transform(X_test)
37
38 # model
39 classifier = RandomForestClassifier()
40
41 # fit model
42 classifier.fit(X_train, y_train)
43
44 # pickle
45 pickle.dump(classifier, open("model.pkl", "wb"))
46
47

```

### Step 3:

- Deploy the model on flask

```

49 app = Flask(__name__)
50
51 model = pickle.load(open("model.pkl", "rb"))
52
53 @app.route("/")
54 def home():
55     return render_template("index.html")
56
57 @app.route("/predict", methods = ['POST'])
58 def predict():
59     float_features = [float(x) for x in request.form.values()]
60     features = [np.array(float_features)]
61     prediction = model.predict(features)
62
63     return render_template("index.html", prediction_text = "Is the person ill? (Yes or No) {}".format(prediction))
64
65
66 if __name__ == '__main__':
67     app.run(port = 3000, debug=True)

```

### Step 4:

- Create documents designed to be displayed in a web browser (html)

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>
5        Tutorial
6      </title>
7    </head>
8
9    <body>
10     <div class = "login">
11       <h1 class="text-center">
12         Illness prediction
13       </h1>
14
15       <form action="{{ url_for('predict')}}"method="post">
16         <input type="text" name="City" placeholder="Austin(1)/Boston(2)/Dallas(3)/Los Angeles(4)/Mountain View(5)/New York City(6)/San Diego(7)/Washington
17         <input type="text" name="Gender" placeholder="Male(0)/Female(1)" required="required" />
18         <input type="text" name="Age" placeholder="age" required="required" />
19         <input type="text" name="Income" placeholder="income" required="required" />
20
21         <button type="submit" class="btn btn-primary btn-block btn-large">Predict</button>
22       </form>
23
24       <br>
25       <br>
26       {{prediction_text}}
27     </div>
28   </body>
29 </html>
30

```

## Results:

Enter the city, gender, age, and income to get the prediction result, the result will show whether a person is ill or not. (Yes or No)

This is an example.

Before adding:

### Illness prediction

Austin(1)/Boston(2)/Dallas(3)	Male(0)/Female(1)	age	income	Predict
-------------------------------	-------------------	-----	--------	---------

Enter the city, gender, age, and income

### Illness prediction

8	0	66	3000	Predict
---	---	----	------	---------

Get the prediction

Tutorial

localhost:3000/predict

HP ConnectedGmailYouTubeMapsTranslateGoogle TranslateCanvasGrammarly

## Illness prediction

Austin(1)/Boston(2)/Dallas(3)

Male(0)/Female(1)

age

income

Predict

Is the preson ill? (Yes or No) ['No']

## Conclusion:

This web page uses the model to display the predicted results