Flask deployment

Name	KABORE Rimbesougri
Batch code	LISUM03
Submitted date	13/09/2021
Submitted to	

Steps of deployment

1. **Selection of a simple model:**

   We have chosen a very notorious model : Boston Housing dataset.
   This dataset is a derived from information collected by the U.S. Census Service concerning housing in the area of Boston MA.
   It contains 506 rows and 14 columns.

2. **Building a machine learning model :**

   We have decided to build a **linear regression model** to predict MEDV based on the other 13 features.

```python
import pandas as pd
import numpy as np
import pickle

from sklearn.datasets import load_boston
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression

boston_dataset = load_boston()

boston = pd.DataFrame(boston_dataset.data, columns=boston_dataset.feature_names)
boston['MEDV'] = boston_dataset.target


X = boston.drop(['MEDV'], axis=1).values
Y = boston['MEDV'].values

X_train, X_test, Y_train, Y_test = train_test_split(X,Y,test_size=0.2,random_state=1)


reg = LinearRegression().fit(X_train, Y_train)
```

3. Saving the model :

The model was saved with the pickle module as "model.pkl"

```
21   reg = LinearRegression().fit(X_train, Y_train)
22
23   with open('model.pkl','wb') as f:
24       pickle.dump(reg,f)
```

4. Creating a html page as a template :

Based on the video from Dataglacier course and other researches, we have created an html page. The major objective was to insert values of the 13 features to predict the MEDV value.

This page is saved as "index.html" on the template folder.

```html
1   <!DOCTYPE html>
2   <html>
3
4   <head>
5       <meta charset="UTF-8">
6       <title>ML API</title>
7   </head>
8
9   <body>
10  <div class="login">
11    <h1 style="color: #5e9ca0;">Predict of Boston House Price</h1>
12    <form action="{{ url_for('predict') }}" method="post">
13      <input type="text" name="CRIM" placeholder="CRIM" required="required" />
14      <input type="text" name="ZN" placeholder="ZN" required="required" />
15      <input type="text" name="INDUS" placeholder="INDUS" required="required"/>
16      <input type="text" name="CHAS" placeholder="CHAS" required="required"/>
17      <input type="text" name="NOX" placeholder="NOX" required="required"/>
18      <input type="text" name="RM" placeholder="RM" required="required"/>
19      <input type="text" name="AGE" placeholder="AGE" required="required"/>
20      <input type="text" name="DIS" placeholder="DIS" required="required"/>
21      <input type="text" name="RAD" placeholder="RAD" required="required"/>
22      <input type="text" name="TAX" placeholder="TAX" required="required"/>
23      <input type="text" name="PTRATIO" placeholder="PTRATIO" required="required"/>
24      <input type="text" name="B" placeholder="B" required="required"/>
25      <input type="text" name="LSTAT" placeholder="LSTAT" required="required"/>
26
27      <button type="submit" class="btn btn-primary btn-block btn-large">Predict</button>
28      </form>
29
30    <br>
31    <br>
32    {{ prediction_text }}
33    <br>
34    <br>
35
36    <h2>Features description</h2>
37    <ul>
38        <li><b>CRIM :</b> per capita crime rate by town.</li>
39        <li><b>ZN :</b> proportion of residential land zoned for lots over 25,000 sq.ft.</li>
40        <li><b>INDUS :</b> proportion of non-retail business acres per town.</li>
41        <li><b>CHAS :</b> Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).</li>
42        <li><b>NOS :</b> nitrogen oxides concentration (parts per 10 million).</li>
43        <li><b>RM :</b> average number of rooms per dwelling.</li>
44        <li><b>AGE :</b> proportion of owner-occupied units built prior to 1940.</li>
45        <li><b>DIS :</b> weighted mean of distances to five Boston employment centres.</li>
46        <li><b>RAD :</b> index of accessibility to radial highways.</li>
```

5. Main code :

In this code we loaded the prediction model and template page.

Then a prediction was made to then be displayed on the page.

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

#app = Flask(__name__)
app = Flask(__name__,template_folder='template')
model = pickle.load(open('model.pkl', 'rb'))

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

@app.route('/predict',methods=['POST'])
def predict():
    '''
    For rendering results on HTML GUI
    '''
    features= list(request.form.to_dict(flat=False).values() )
    int_features = [int(x[0]) for x in features]
    final_features = [np.array(int_features)]
    prediction = model.predict(final_features)

    output = round(prediction[0], 2)

    return render_template('index.html', prediction_text='MEDV should be $ {}'.format(output))

@app.route('/predict_api',methods=['POST'])
def predict_api():
    '''
    For direct API calls trought request
    '''
    data = request.get_json(force=True)
    prediction = model.predict([np.array(list(data.values()))])

    output = prediction[0]
    return jsonify(output)

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

6. You only need to test it now !

Predict of Boston House Price

4	5	9	0
5	6	5	4
66	4	33	32
3	Predict		

Features description

- **CRIM :** per capita crime rate by town.
- **ZN :** proportion of residential land zoned for lots over 25,000 sq.ft.
- **INDUS :** proportion of non-retail business acres per town.
- **CHAS :** Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).
- **NOS :** nitrogen oxides concentration (parts per 10 million).
- **RM :** average number of rooms per dwelling.
- **AGE :** proportion of owner-occupied units built prior to 1940.
- **DIS :** weighted mean of distances to five Boston employment centres.
- **RAD :** index of accessibility to radial highways.
- **TAX :** full-value property-tax rate per \$10,000.
- **PTRATIO :** pupil-teacher ratio by town.
- **BLACK :** 1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town.
- **DIS :** lower status of the population (percent).
- **MEDV :** median value of owner-occupied homes in $1000s

MEDV should be $ -56.3