Name: Ruizhe Zhang

Batch code: LISUM10: 30 May - 30 Aug 2022

Submission date: 06/30/2022

Submitted to: Canvas (GitHub URL)

Data Choosing

In order to understand how to use Flask better, I select the dataset showing NBA players' points, rebounds, assists and salary (shown in the files as "NBA.csv"), trying to predict NBA player's salary based on his performance:

	A	B	C	D	E	F	G	H
1	pts	rbd	ast	salary				
2	17.6	5	6.7	41.4				
3	27.2	7.9	7	41.4				
4	34.3	6.6	7.5	41.3				
5	25.3	7.8	10.2	39.2				
6	21.5	5.7	3.9	35.5				
7	14.4	3.2	4.4	34.5				
8	19.6	6.9	3.2	34.4				
9	19.9	6.7	6	34.4				
10	20.4	3.9	4.8	34.4				
11	27.1	7.1	4.9	34.4				
242	6	2.3	0.9	2.3				
243	6.9	2	2.5	2.3				
244	8.3	2.2	1.4	2.3				
245	10.1	2.2	0.8	2.3				
246	11.2	2	1.7	2.3				
247	3.6	2.3	0.5	2.2				
248	8.7	4.9	1.2	2.2				
249	8.8	2.1	2.4	2.1				
250	9.3	1.9	1.9	2.1				
251	3.7	2.6	0.4	2				
252	6.7	1.9	1.2	2				
253	7.6	4.5	1	2				
254								

Part 1: API

Model:

temp.py | app.py | model.py

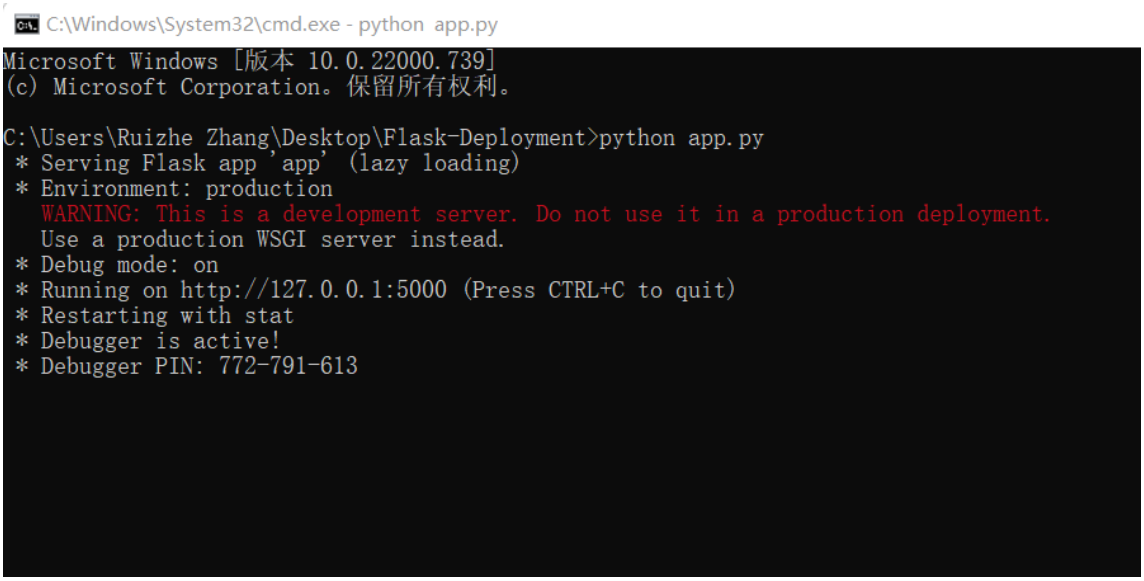
```python
# Importing the libraries
import numpy as np
import pandas as pd
import pickle

dataset = pd.read_csv('NBA.csv')

X = dataset.iloc[:, :3]

y = dataset.iloc[:, -1]

from sklearn.linear_model import LinearRegression
regressor = LinearRegression()

#Fitting model with trainig data
regressor.fit(X, y)

# Saving model to disk
pickle.dump(regressor, open('model.pkl','wb'))
```

App:

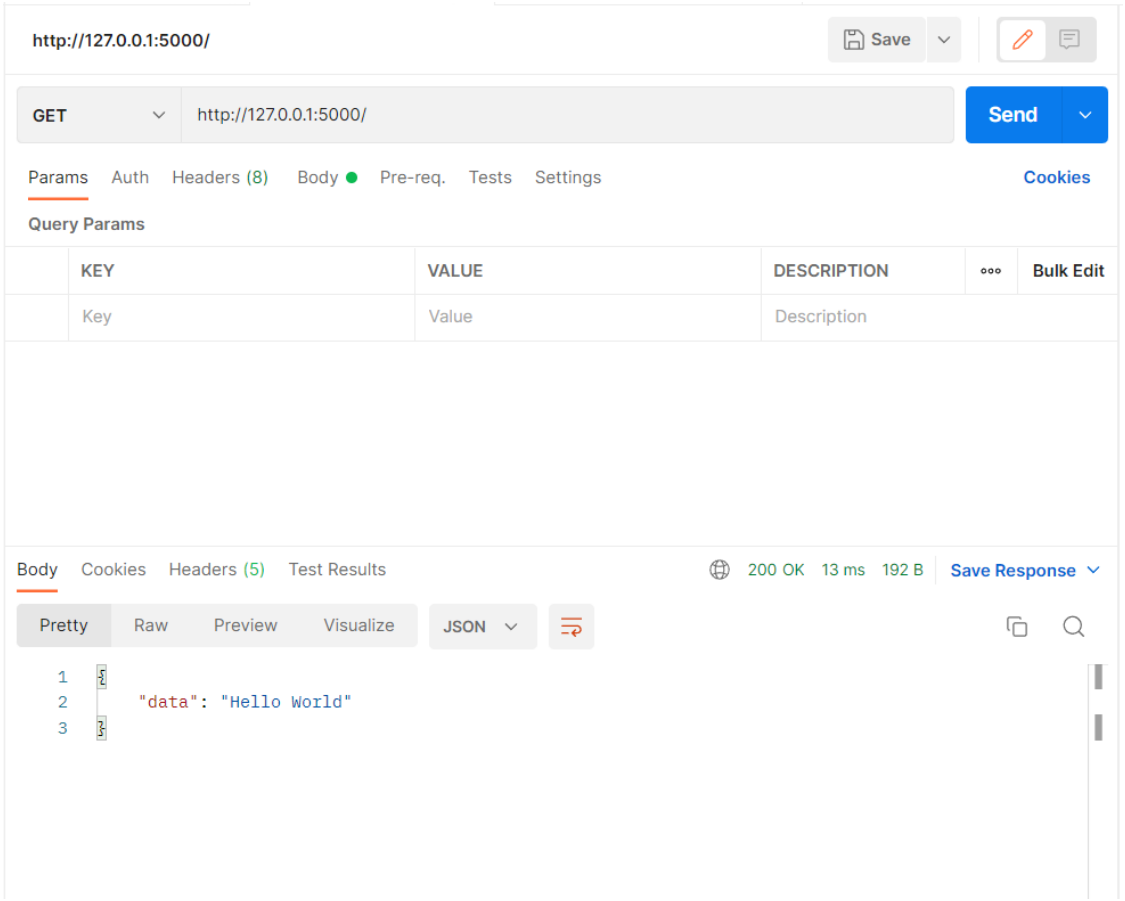
temp.py | app.py | model.py

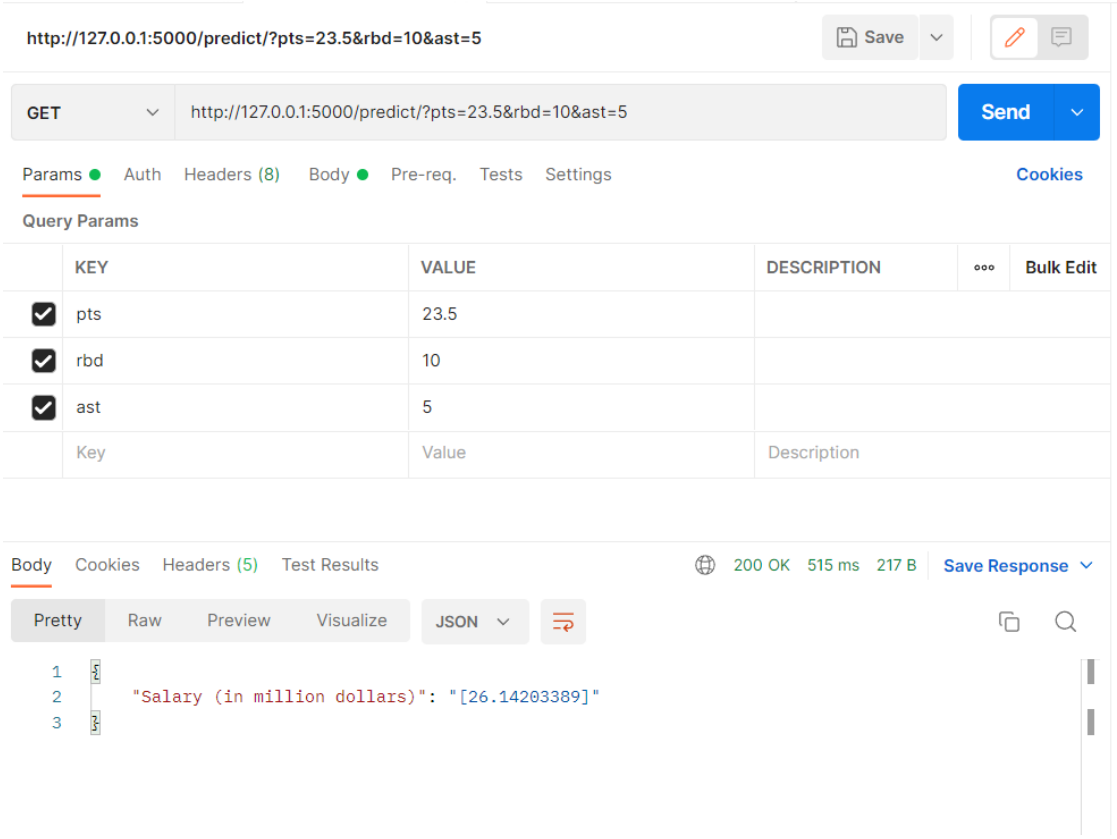
```python
from flask import Flask, request, jsonify
import pickle
import pandas as pd

app = Flask(__name__)

@app.route('/', methods=['GET','POST'])
def home():
    if(request.method=='GET'):

        data = "Hello World"
        return jsonify({'data':data})

@app.route('/predict/')
def predict():
    model = pickle.load(open('model.pkl', 'rb'))
    pts = request.args.get('pts')
    rbd = request.args.get('rbd')
    ast = request.args.get('ast')

    test_df = pd.DataFrame({'pts':[pts],'rbd':[rbd],'ast':[ast]})

    pred_price = model.predict(test_df)
    return jsonify({'The salary of this player should be':str(pred_price)})

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

Deployment on Flask:



Predicting using endpoints:





Part 2: Heroku

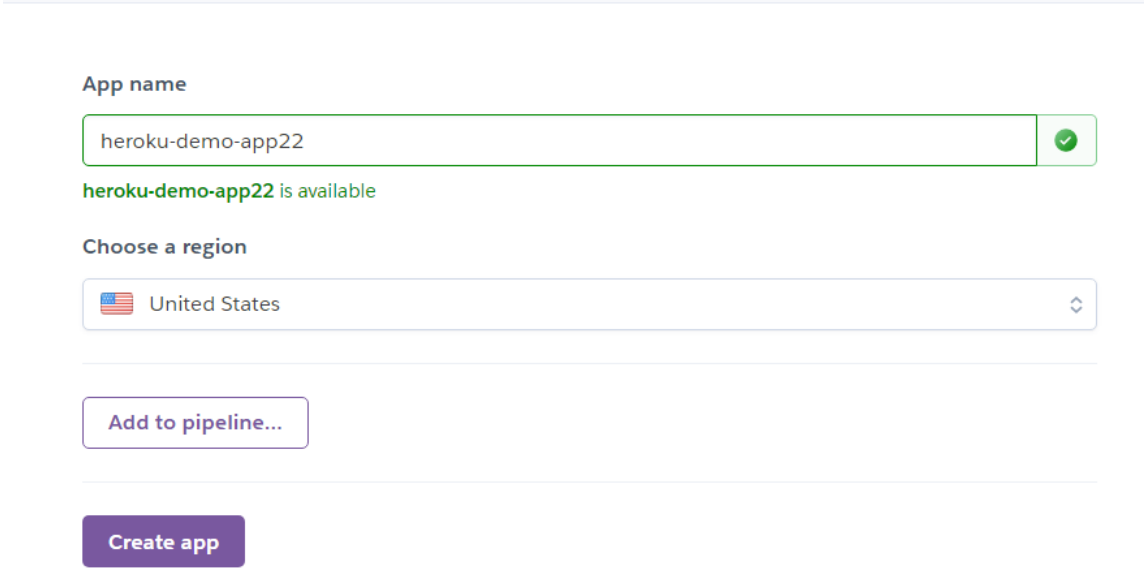
Model:

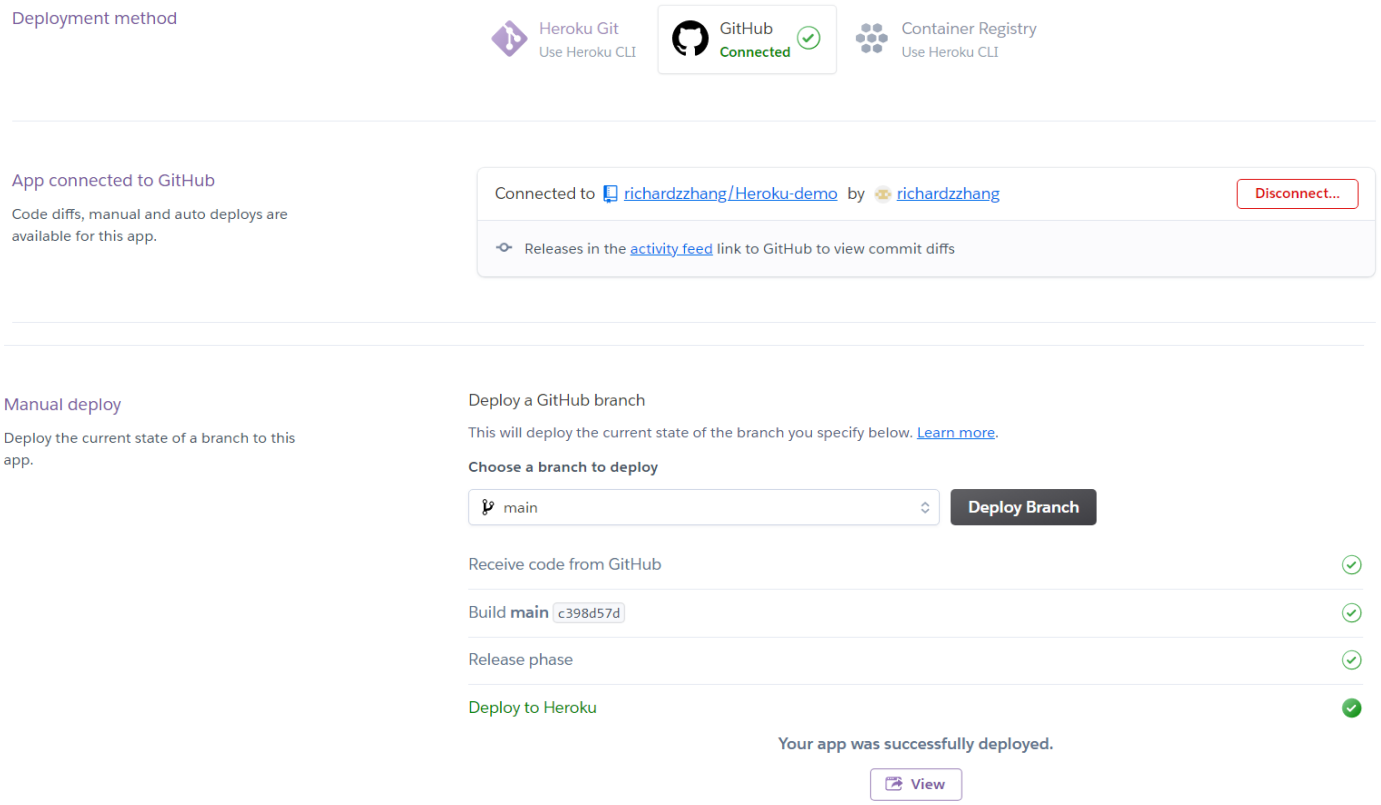
```python
# Importing the libraries
import numpy as np
import pandas as pd
import pickle

dataset = pd.read_csv('NBA.csv')

X = dataset.iloc[:, :3]

y = dataset.iloc[:, -1]

from sklearn.linear_model import LinearRegression
regressor = LinearRegression()

#Fitting model with trainig data
regressor.fit(X, y)

# Saving model to disk
pickle.dump(regressor, open('model.pkl','wb'))

# Loading model to compare the results
model = pickle.load(open('model.pkl','rb'))
print(model.predict([[36, 9, 12]]))
```

App:

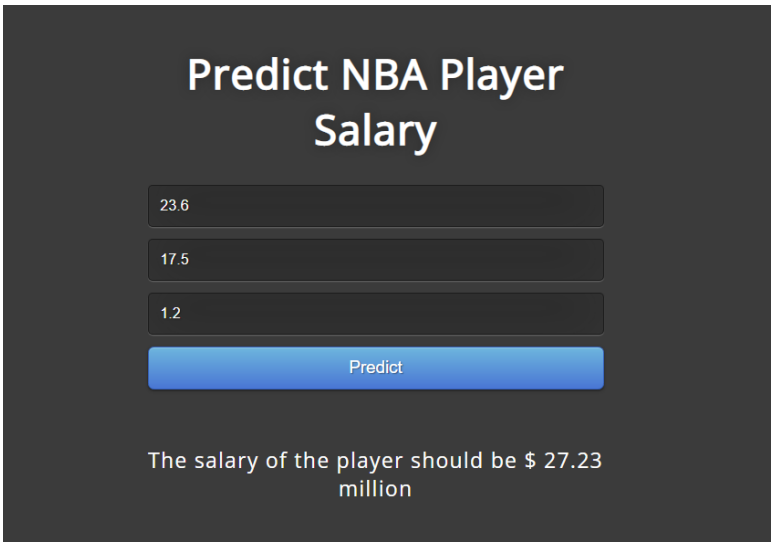
```python
import numpy as np
from flask import Flask, request,render_template
import pickle

app = Flask(__name__)
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
    '''
    int_features = [float(x) for x in request.form.values()]
    final_features = [np.array(int_features)]
    prediction = model.predict(final_features)

    output = round(prediction[0], 2)

    return render_template('index.html', prediction_text='The salary of the player should be $ {} million'.format(output))

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

Deployment on Heroku:




Deployment method
Heroku Git
Use Heroku CLI
GitHub
Connected
Container Registry
Use Heroku CLI
App connected to GitHub
Code diffs, manual and auto deploys are available for this app.
Connected to richardzzhang/Heroku-demo by richardzzhang
Disconnect...
Releases in the activity feed link to GitHub to view commit diffs
Manual deploy
Deploy the current state of a branch to this app.
Deploy a GitHub branch
This will deploy the current state of the branch you specify below. Learn more.
Choose a branch to deploy
main
Deploy Branch
Receive code from GitHub
Build main c398d57d
Release phase
Deploy to Heroku
Your app was successfully deployed.
View

Test:


Predict NBA Player Salary
23.6
17.5
1.2
Predict
The salary of the player should be $ 27.23 million