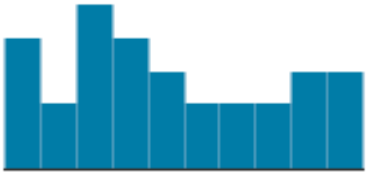
Name: Matthew Zhou
Batch code: LISUM20
Submission Date: 5/7/2023
Submitted to: Canvas/Github

Dataset: Salary_Data

# YearsExperience	# Age	# Salary
Integer value representing the experience of a person(in years - integer value)	This is the age(integer value) of the person.	This is the salary earned(integer value) by the person of the given age and years of experience.
 1.1 10.5	 21 38	 37.7k 122k

Model Creation and Serialization:

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
import pickle

df = pd.read_csv("Salary_Data.csv")

X = df.loc[:, df.columns != "Salary"]
y = df["Salary"]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=.2)
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
print(f"Accuracy: {mean_squared_error(y_test, y_pred)}")
pickle.dump(model, open("model.pickle", "wb"))
```

Accuracy: 28688602.326753754

HTML Template for Website:

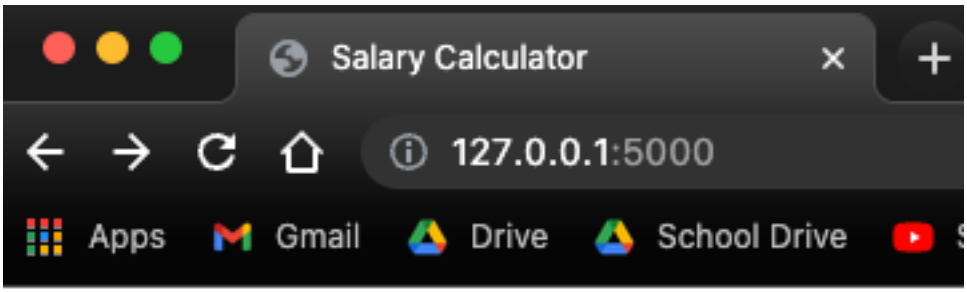
```html
<!DOCTYPE html>
<html>
<head>
    <title>Salary Calculator</title>
</head>
<body>
    <h1>Salary Calculator</h1>
    <form action="/predict" method="post">
        <label for="yearsOfExperience">Years of Experience:</label>
        <input type="number" id="yearsOfExperience" name="yearsOfExperience" required>
        <br>
        <label for="age">Age:</label>
        <input type="number" id="age" name="age" required>
        <br>
        <button type="submit">Calculate Salary</button>
    </form>
    <br>
    {{ prediction_text }}
</body>
</html>
```

Deserialization of Model, Prediction using POST:

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

app = Flask(__name__)
model = pickle.load(open("model.pickle", "rb"))

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

@app.route('/predict', methods=['POST'])
def predict():
    X = np.array([[int(x) for x in request.form.values()]])
    pred = model.predict(X)
    out = round(pred[0], 2)

    return render_template("index.html", prediction_text="Predicted Salary: {}".format(out))

app.run(port=5000)
```

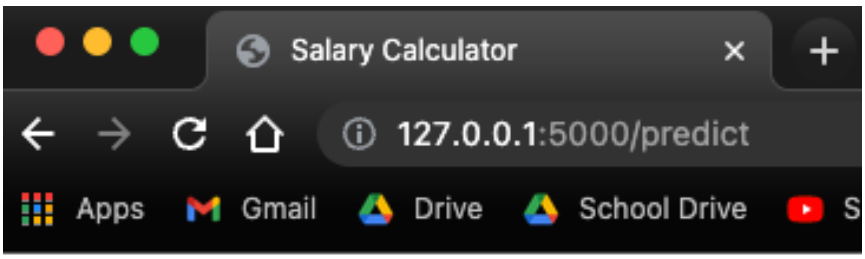
Example Run:

```
(dg_env) mattjzhou@lawn-143-215-109-225 FlaskDeployment % python app.py
 * Serving Flask app 'app'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
```



Salary Calculator

Years of Experience:
Age:
Calculate Salary



Salary Calculator

Years of Experience:
Age:
Calculate Salary

Predicted Salary: 87373.06

```
127.0.0.1 - - [30/Apr/2023 07:18:39] "GET / HTTP/1.1" 200 -
/Users/mattjzhou/opt/miniconda3/envs/dg_env/lib/python3.10/site-packa
 fitted with feature names
  warnings.warn(
127.0.0.1 - - [30/Apr/2023 07:19:01] "POST /predict HTTP/1.1" 200 -
```

Update to local network

```
app.run(host='0.0.0.0', port=50000)
```

Created AWS EC2 Instance in Ubuntu

Instance summary for i-0a6fc169883131b57 Info

Updated less than a minute ago

Connect | Instance state | Actions

Instance ID i-0a6fc169883131b57	Public IPv4 address 18.221.143.185 | open address	Private IPv4 addresses 172.31.34.249
IPv6 address –	Instance state Running	Public IPv4 DNS ec2-18-221-143-185.us-east-2.compute.amazonaws.com | open address
Hostname type IP name: ip-172-31-34-249.us-east-2.compute.internal	Private IP DNS name (IPv4 only) ip-172-31-34-249.us-east-2.compute.internal	
Answer private resource DNS name IPv4 (A)	Instance type t2.micro	Elastic IP addresses –
Auto-assigned IP address 18.221.143.185 [Public IP]	VPC ID vpc-0a44db18f65d9fb3d	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role –	Subnet ID subnet-053ab94c6c7286940	Auto Scaling Group name –
IMDSv2 Optional		

SSH into instance

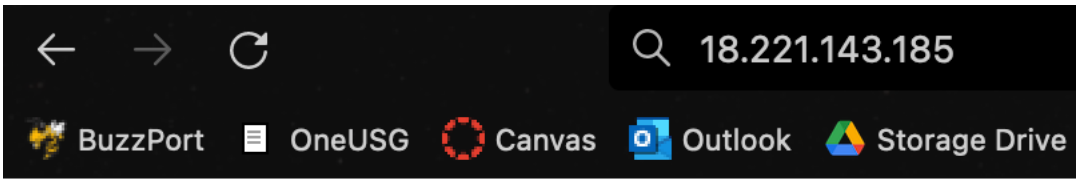
```
(dg_env) mattjzhou@lawn-143-215-109-225 Downloads % ssh -i "test.pem" ubuntu@ec2-18-221-143-185.us-east-2.compute.amazonaws.com
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-1031-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sun May  7 07:08:45 UTC 2023

  System load:  0.080078125       Processes:             96
  Usage of /:   32.7% of 7.57GB   Users logged in:       0
  Memory usage: 27%               IPv4 address for eth0: 172.31.34.249
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

44 updates can be applied immediately.
23 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


Last login: Sun May  7 06:56:22 2023 from 71.179.81.113
ubuntu@ip-172-31-34-249:~$
```

Clone repo and download packages

Run in instance

```
ubuntu@ip-172-31-34-249:~/dg_w4$ python3 app.py
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://0.0.0.0:50000/ (Press CTRL+C to quit)
```

Example run at web IP address



Salary Calculator

Years of Experience:

Age:

Calculate Salary