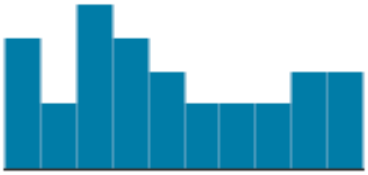
Name: Matthew Zhou
Batch code: LISUM20
Submission Date: 4/30/2023
Submitted to: Canvas/Github

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:

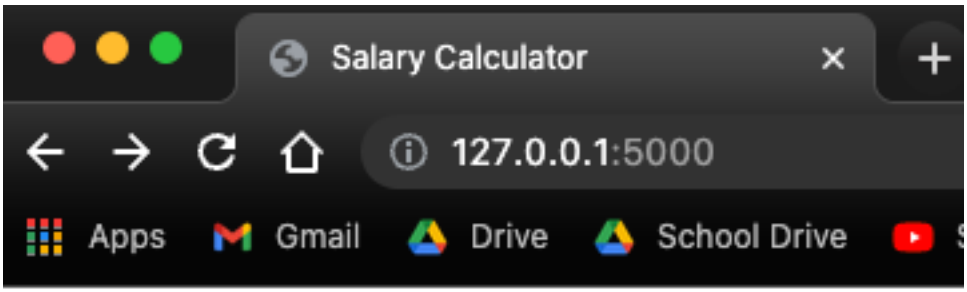
```html
<!DOCTYPE html>
<html>
<head>
    <title>Salary Calculator</title>
</head>
<body>
    <h1>Salary Calculator</h1>
    <form action="/predict" method="post">
        <label for="yearsOfExperience">Years of Experience:</label>
        <input type="number" id="yearsOfExperience" name="yearsOfExperience" 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)
```

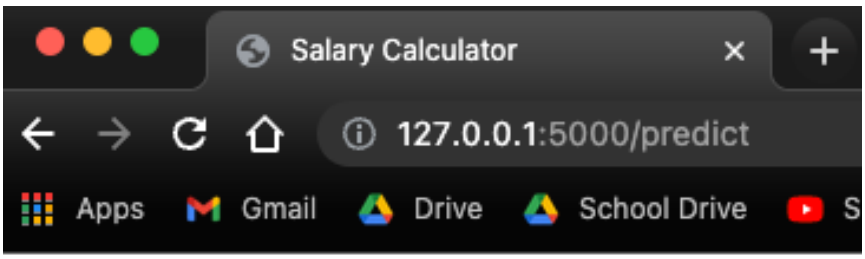
Example Run:

```
(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 -
```