Name: Nitish Jena

Batch code: LISUM39

Submission date: 28 November 2024

Submitted to: Data Glacier (**VI Team**)

# Snapshot of each step of deployment

1. Loading and Preparing the Dataset

```
import pandas as pd

# Load your toy dataset
dataset = pd.read_csv('toy.csv')

# Display the first few rows of the dataset
print(dataset.head())
```

```
   Age  Salary  Experience  Gender  Target
0   56   38392          30       1       1
1   46   60535          35       0       1
2   32   82256           7       1       1
3   25   65222          16       0       1
4   38   93335          26       1       0
```

2. Splitting Data and

```
# Features and target
X = dataset.drop('Target', axis=1)  # Independent variables
y = dataset['Target']  # Dependent variable (output)

print("Features:")
print(X.head())
print("\nTarget:")
print(y.head())
```

```
Features:
   Age  Salary  Experience  Gender
0   56   38392          30       1
1   46   60535          35       0
2   32   82256           7       1
3   25   65222          16       0
4   38   93335          26       1

Target:
0    1
1    1
2    1
3    1
4    0
Name: Target, dtype: int64
```

3. Training the Model

```
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
import joblib

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train the Random Forest model
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)
```

4. Saving the Trained Model

```
# Save the trained model
joblib.dump(model, 'toy_model.pkl')
print("Model saved as 'toy_model.pkl'")
```

```
Model saved as 'toy_model.pkl'
```

5. Creating the Flask App

```
from flask import Flask, request, jsonify
from pyngrok import ngrok
import joblib
import pandas as pd

app = Flask(__name__)

# Load the saved model
model = joblib.load('toy_model.pkl')

@app.route('/')
def home():
    return "Welcome to the Toy Dataset Prediction API!"

@app.route('/predict', methods=['POST'])
def predict():
    # Receive JSON data
    data = request.get_json()
    features = pd.DataFrame(data, index=[0])  # Convert JSON to DataFrame
    prediction = model.predict(features)  # Predict
    return jsonify({'prediction': int(prediction[0])})
```

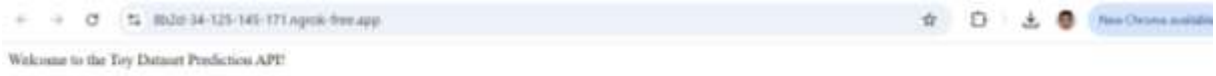
6. Running the Flask App

```
# Expose the app using ngrok
public_url = ngrok.connect(5000)
print(f"Public URL: {public_url}")
app.run(port=5000)
```

```
Public URL: NgrokTunnel: "https://8b2d-34-125-145-171.ngrok-free.app" -> "http://localhost:5000"
 * Serving Flask app '__main__'
 * Debug mode: off
```

7. Testing the Flask App

(Testing / Route)




8. Testing /predict Endpoint using POSTMAN