Name: Sai Supriya Varugunda

Batch Code: LISUM25

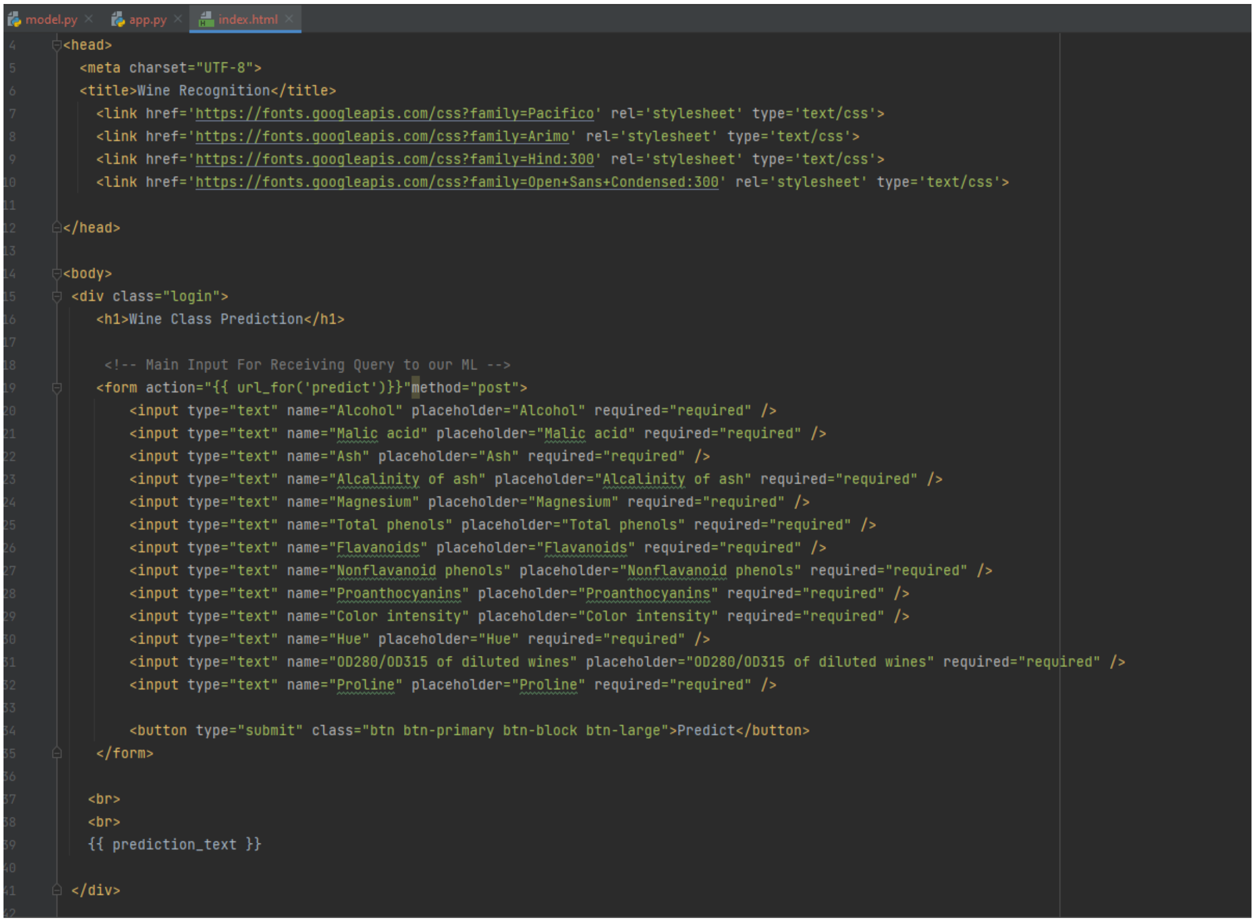
Submission date: 28/09/2023

Submitted to: Data Glacier Team

Deployment on Flask

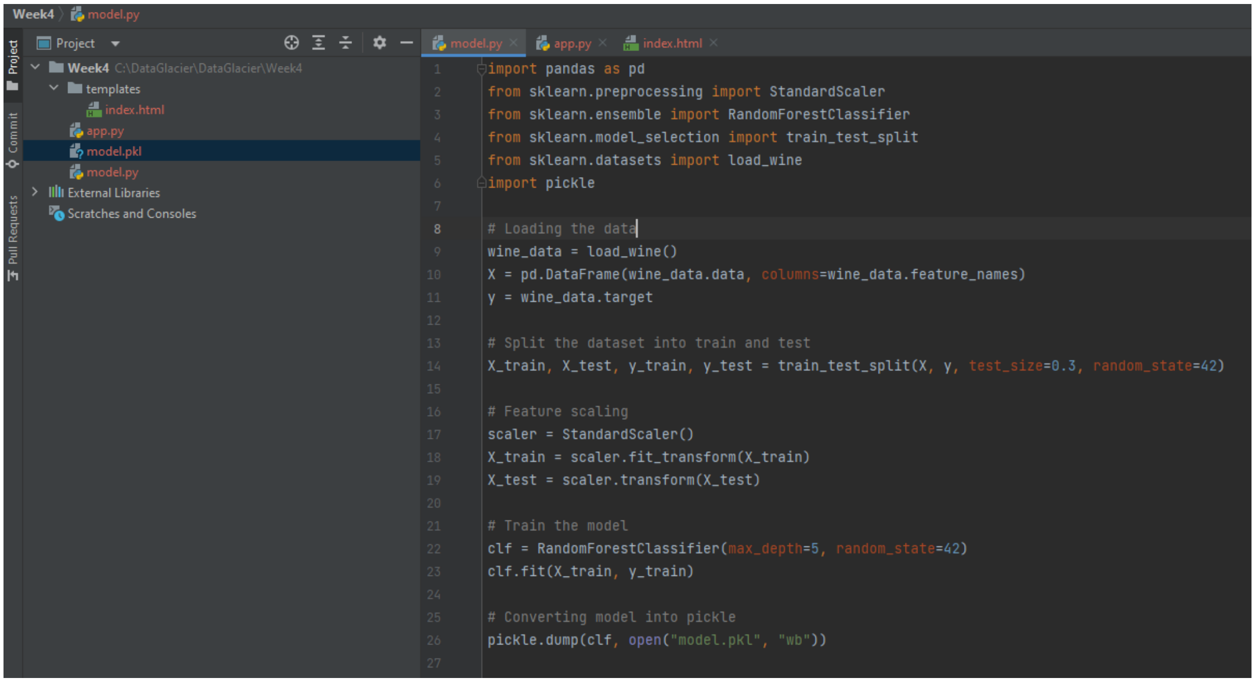
I have considered toy dataset of wine classification for model deployment on flask.

I have created an index.html and placed it in templates folder. It will take the input and give it to our model to predict the result.



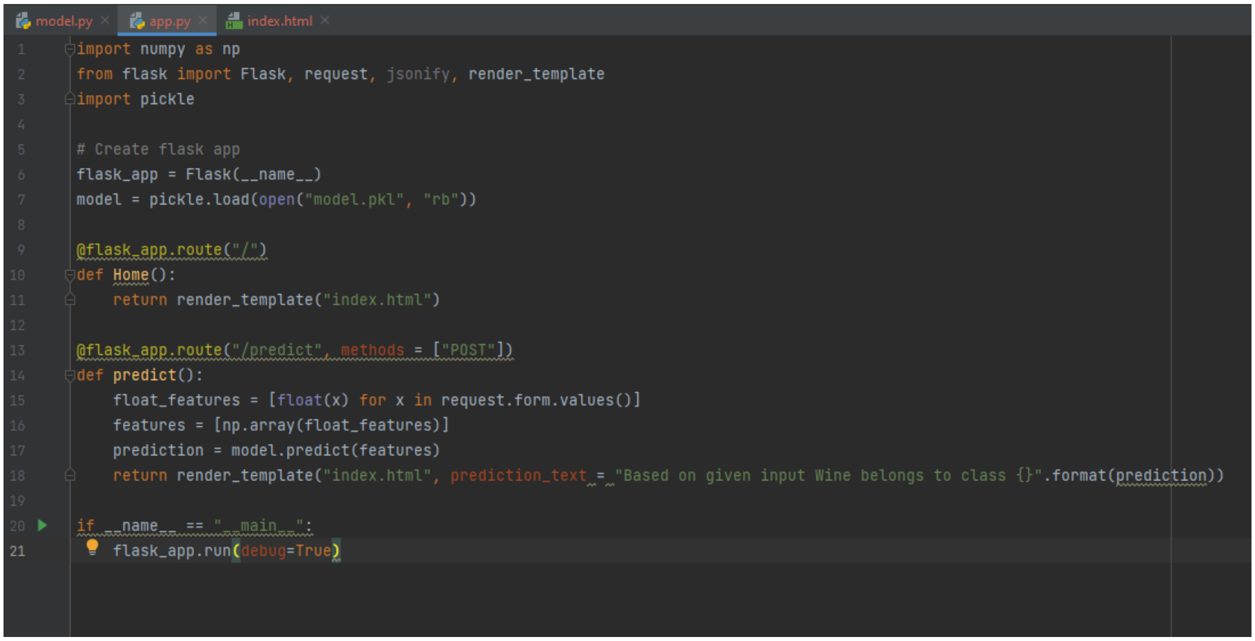
```html
<head>
  <meta charset="UTF-8">
  <title>Wine Recognition</title>
    <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
    <link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
    <link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
    <link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text/css'>

</head>

<body>
 <div class="login">
    <h1>Wine Class Prediction</h1>

     <!-- Main Input For Receiving Query to our ML -->
    <form action="{{ url_for('predict')}}"method="post">
        <input type="text" name="Alcohol" placeholder="Alcohol" required="required" />
        <input type="text" name="Malic acid" placeholder="Malic acid" required="required" />
        <input type="text" name="Ash" placeholder="Ash" required="required" />
        <input type="text" name="Alcalinity of ash" placeholder="Alcalinity of ash" required="required" />
        <input type="text" name="Magnesium" placeholder="Magnesium" required="required" />
        <input type="text" name="Total phenols" placeholder="Total phenols" required="required" />
        <input type="text" name="Flavanoids" placeholder="Flavanoids" required="required" />
        <input type="text" name="Nonflavanoid phenols" placeholder="Nonflavanoid phenols" required="required" />
        <input type="text" name="Proanthocyanins" placeholder="Proanthocyanins" required="required" />
        <input type="text" name="Color intensity" placeholder="Color intensity" required="required" />
        <input type="text" name="Hue" placeholder="Hue" required="required" />
        <input type="text" name="OD280/OD315 of diluted wines" placeholder="OD280/OD315 of diluted wines" required="required" />
        <input type="text" name="Proline" placeholder="Proline" required="required" />

        <button type="submit" class="btn btn-primary btn-block btn-large">Predict</button>
    </form>

   <br>
   <br>
   {{ prediction_text }}

 </div>
```

Then created model.py used Random Forest classifier and created a pickle file from it.



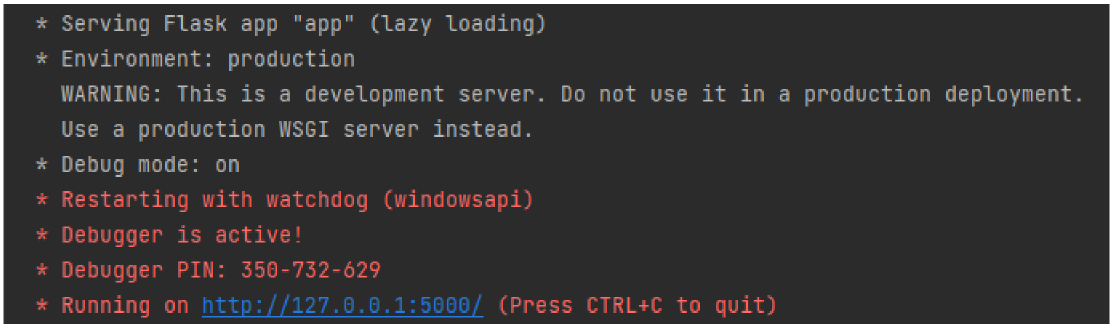
```python
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.datasets import load_wine
import pickle

# Loading the data
wine_data = load_wine()
X = pd.DataFrame(wine_data.data, columns=wine_data.feature_names)
y = wine_data.target

# Split the dataset into train and test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Feature scaling
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Train the model
clf = RandomForestClassifier(max_depth=5, random_state=42)
clf.fit(X_train, y_train)

# Converting model into pickle
pickle.dump(clf, open("model.pkl", "wb"))
```

Then finally created app.py that loads the model using pickle take the input and posts the prediction.



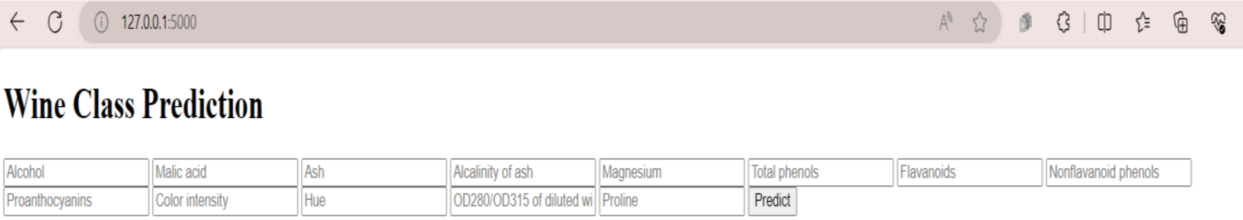
```python
import numpy as np
from flask import Flask, request, jsonify, render_template
import pickle

# Create flask app
flask_app = Flask(__name__)
model = pickle.load(open("model.pkl", "rb"))

@flask_app.route("/")
def Home():
    return render_template("index.html")

@flask_app.route("/predict", methods = ["POST"])
def predict():
    float_features = [float(x) for x in request.form.values()]
    features = [np.array(float_features)]
    prediction = model.predict(features)
    return render_template("index.html", prediction_text = "Based on given input Wine belongs to class {}".format(prediction))

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

Now executed this app.py which gave below link:



```
 * 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: on
 * Restarting with watchdog (windowsapi)
 * Debugger is active!
 * Debugger PIN: 350-732-629
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

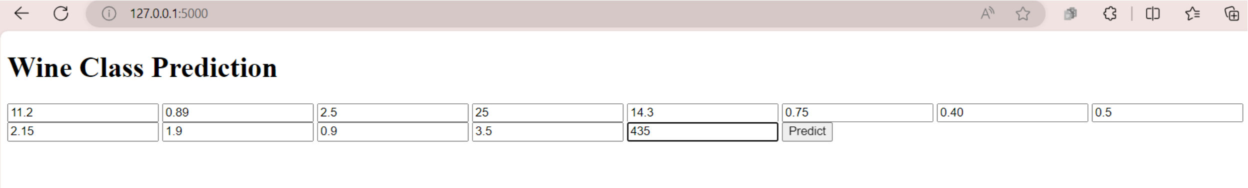
My model is now deployed to this URL and is ready to predict wine class.



Wine Class Prediction

Alcohol | Malic acid | Ash | Alcalinity of ash | Magnesium | Total phenols | Flavanoids | Nonflavanoid phenols
Proanthocyanins | Color intensity | Hue | OD280/OD315 of diluted wi | Proline | Predict

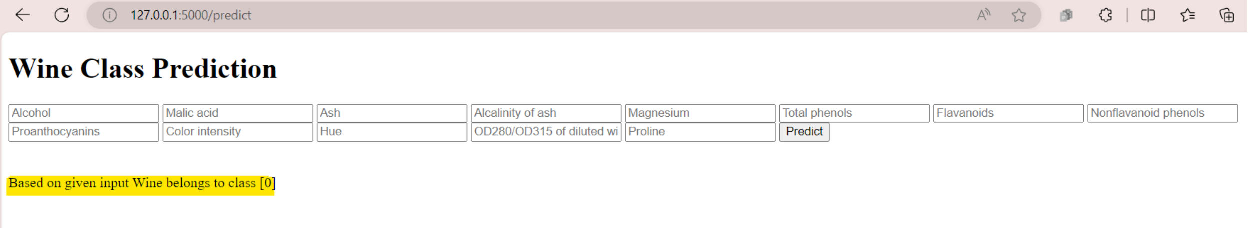
Giving input values into all these fields:



Wine Class Prediction

11.2 | 0.89 | 2.5 | 25 | 14.3 | 0.75 | 0.40 | 0.5
2.15 | 1.9 | 0.9 | 3.5 | 435 | Predict

Now by clicking on predict button I got wine class predictions as below:



Wine Class Prediction

Alcohol | Malic acid | Ash | Alcalinity of ash | Magnesium | Total phenols | Flavanoids | Nonflavanoid phenols
Proanthocyanins | Color intensity | Hue | OD280/OD315 of diluted wi | Proline | Predict

Based on given input Wine belongs to class [0]