

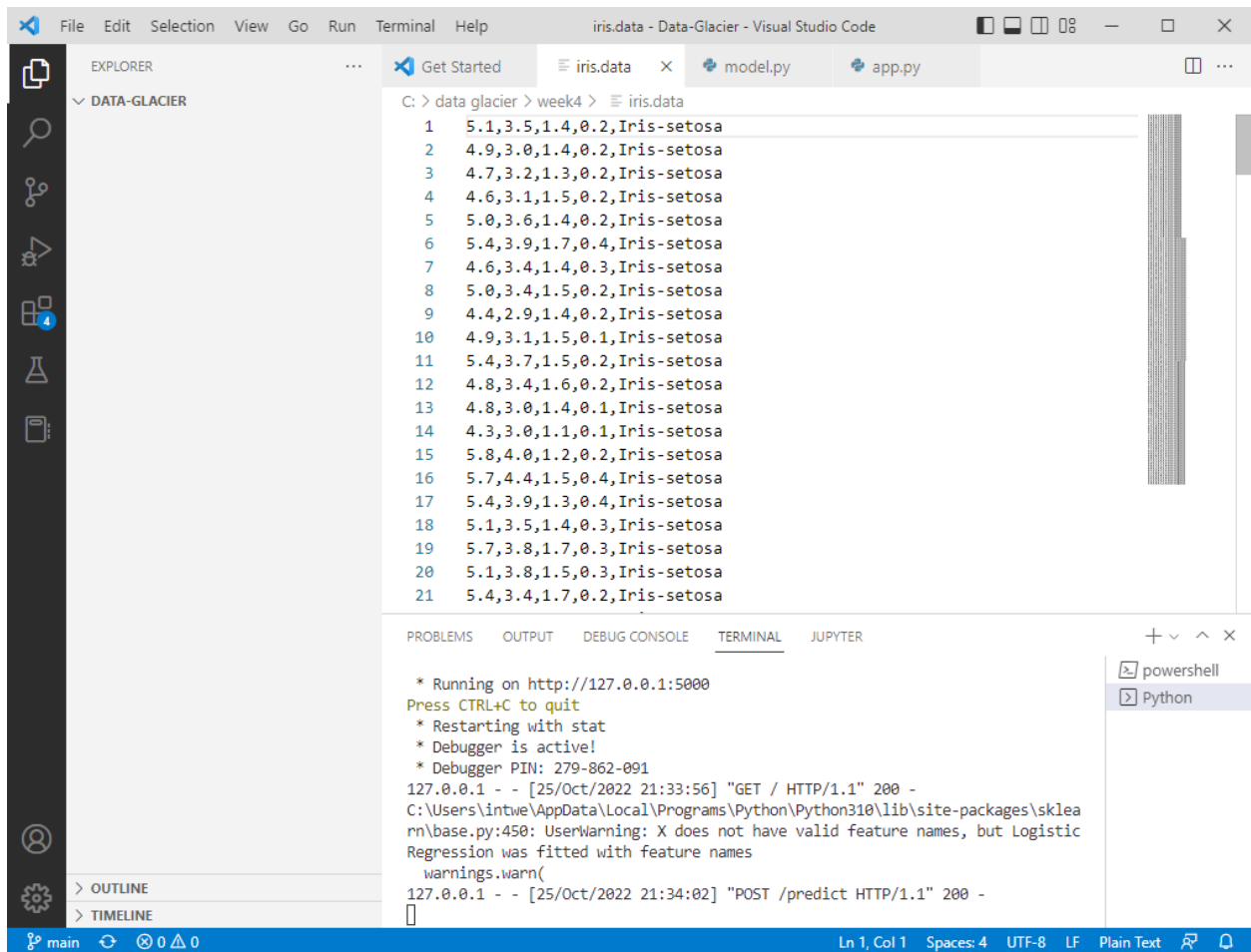
Name: Wen Li

Batch code: LISUM14

Submission date: 2022-10-25

1. Iris data set

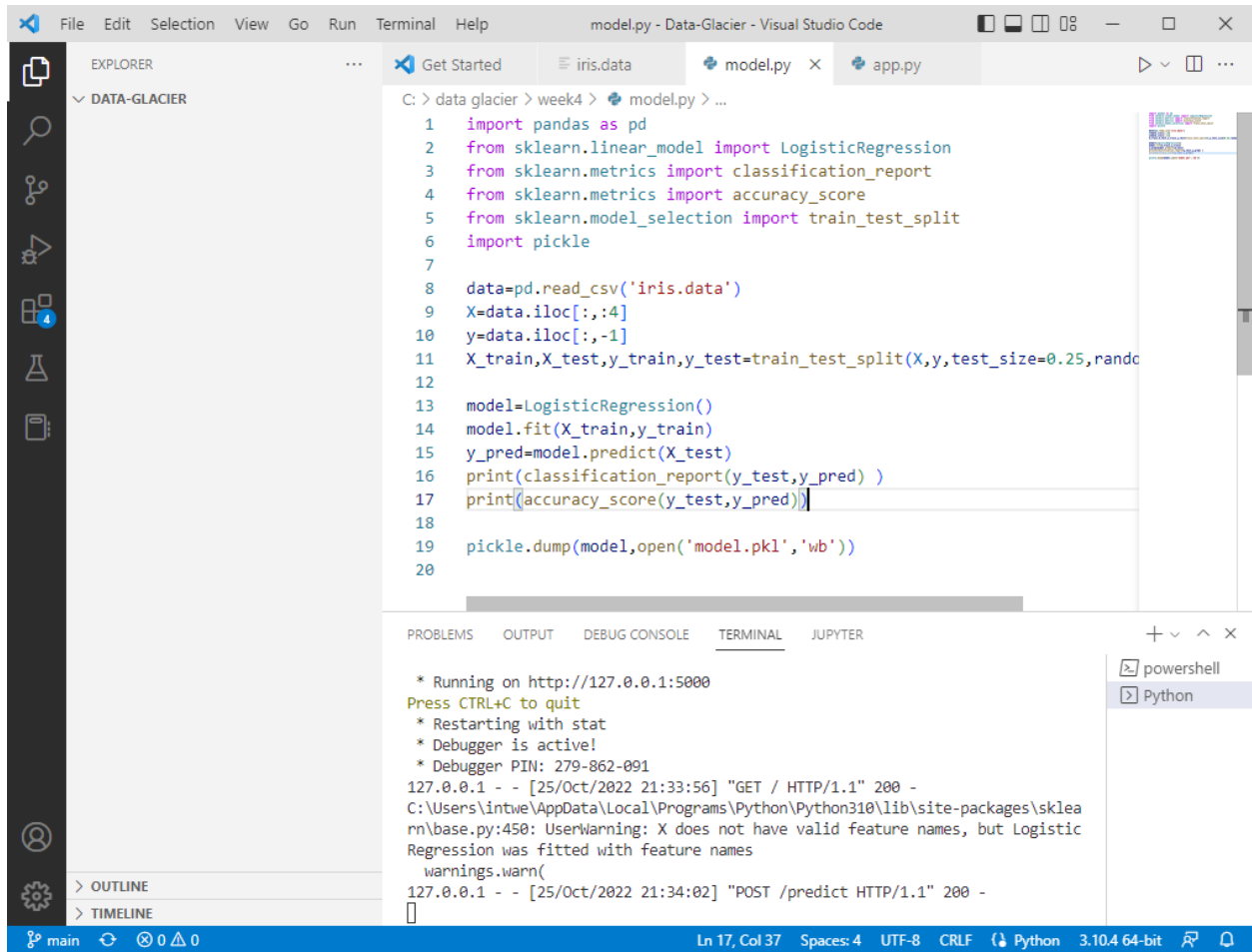
The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are NOT linearly separable from each other.



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows the project structure with 'DATA-GLACIER' expanded. The main editor area displays the 'iris.data' file, which contains a list of 21 rows of data. Each row consists of five numerical values followed by the class name 'Iris-setosa'. The bottom panel shows the TERMINAL window with the following output:

```
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 279-862-091
127.0.0.1 - - [25/Oct/2022 21:33:56] "GET / HTTP/1.1" 200 -
C:\Users\intwe\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but Logistic Regression was fitted with feature names
warnings.warn(
127.0.0.1 - - [25/Oct/2022 21:34:02] "POST /predict HTTP/1.1" 200 -
```

2. Model development



The screenshot displays the Visual Studio Code interface with a Python script named `model.py` open. The script performs the following steps:

- Imports `pandas` as `pd`, `LogisticRegression` from `sklearn.linear_model`, `classification_report` and `accuracy_score` from `sklearn.metrics`, `train_test_split` from `sklearn.model_selection`, and `pickle`.
- Reads the `iris.data` CSV file into a `data` DataFrame.
- Extracts the first four columns as `X` and the last column as `y`.
- Splits the data into training and testing sets using `train_test_split` with a test size of 0.25.
- Creates a `LogisticRegression` model, fits it to the training data, and predicts on the test data.
- Prints the `classification_report` and the `accuracy_score`.
- Saves the trained model to a file named `model.pkl` using `pickle.dump`.

The terminal output shows the script running on `http://127.0.0.1:5000`. It includes a warning from `sklearn` about feature names and a successful POST request to the `/predict` endpoint.

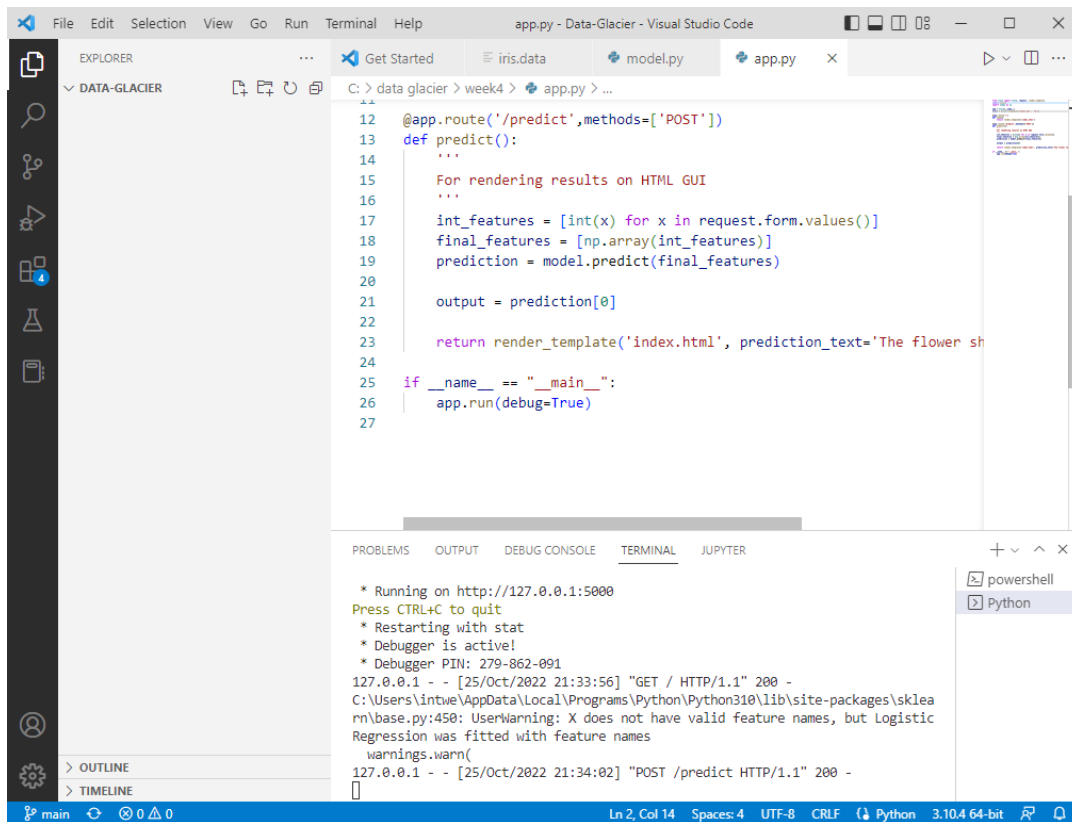
```
C: > data glacier > week4 > model.py > ...
1 import pandas as pd
2 from sklearn.linear_model import LogisticRegression
3 from sklearn.metrics import classification_report
4 from sklearn.metrics import accuracy_score
5 from sklearn.model_selection import train_test_split
6 import pickle
7
8 data=pd.read_csv('iris.data')
9 X=data.iloc[:, :4]
10 y=data.iloc[:, -1]
11 X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.25,rand
12
13 model=LogisticRegression()
14 model.fit(X_train,y_train)
15 y_pred=model.predict(X_test)
16 print(classification_report(y_test,y_pred) )
17 print(accuracy_score(y_test,y_pred))
18
19 pickle.dump(model,open('model.pkl','wb'))
20
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

```
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 279-862-091
127.0.0.1 - - [25/Oct/2022 21:33:56] "GET / HTTP/1.1" 200 -
C:\Users\intwe\AppData\Local\Programs\Python\Python310\lib\site-packages\sklea
rn\base.py:450: UserWarning: X does not have valid feature names, but Logistic
Regression was fitted with feature names
warnings.warn(
127.0.0.1 - - [25/Oct/2022 21:34:02] "POST /predict HTTP/1.1" 200 -
[]
```

main 0 0 Ln 17, Col 37 Spaces: 4 UTF-8 CRLF Python 3.10.4 64-bit

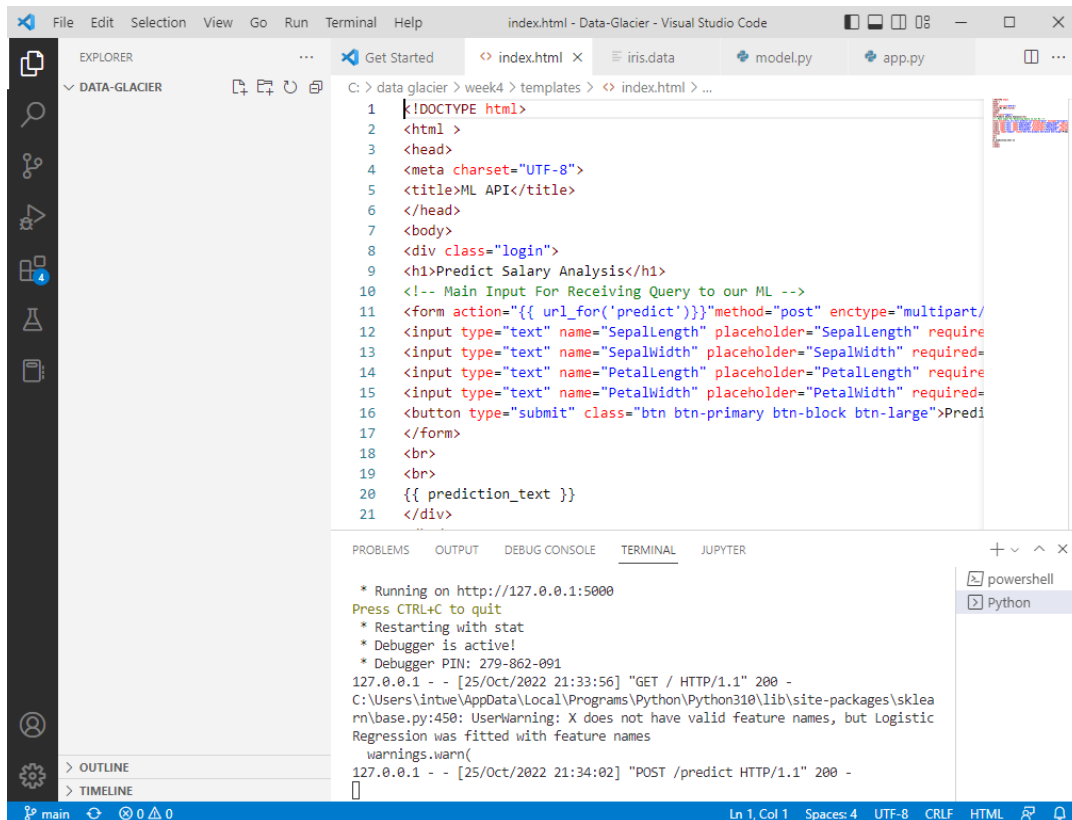
3. Web app



The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying the project structure: DATA-GLACIER, iris.data, model.py, and app.py. The main editor window displays the app.py file, which contains a Flask application. The code defines a route for '/predict' using the POST method. Inside the predict function, it processes the request form values into features, uses a pre-trained model to predict the species, and returns the result using a template. The terminal at the bottom shows the application running on http://127.0.0.1:5000, with a POST request to /predict.

```
12 @app.route('/predict',methods=['POST'])
13 def predict():
14     ...
15     For rendering results on HTML GUI
16     ...
17     int_features = [int(x) for x in request.form.values()]
18     final_features = [np.array(int_features)]
19     prediction = model.predict(final_features)
20
21     output = prediction[0]
22
23     return render_template('index.html', prediction_text='The flower sh
24
25 if __name__ == "__main__":
26     app.run(debug=True)
27
```

Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 279-862-091
127.0.0.1 - - [25/Oct/2022 21:33:56] "GET / HTTP/1.1" 200 -
C:\Users\intwe\AppData\Local\Programs\Python\Python310\lib\site-packages\sklea
rn\base.py:450: UserWarning: X does not have valid feature names, but Logistic
Regression was fitted with feature names
warnings.warn(
127.0.0.1 - - [25/Oct/2022 21:34:02] "POST /predict HTTP/1.1" 200 -



The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying the project structure: DATA-GLACIER, iris.data, model.py, and app.py. The main editor window displays the index.html file, which contains the HTML template for the web application. The template includes a login form with input fields for SepalLength, SepalWidth, PetalLength, and PetalWidth, and a submit button. The form action is set to the /predict endpoint. The terminal at the bottom shows the application running on http://127.0.0.1:5000, with a POST request to /predict.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>ML API</title>
6 </head>
7 <body>
8 <div class="login">
9 <h1>Predict Salary Analysis</h1>
10 <!-- Main Input For Receiving Query to our ML -->
11 <form action="{{ url_for('predict')}}" method="post" enctype="multipart/
12 <input type="text" name="SepalLength" placeholder="SepalLength" require
13 <input type="text" name="SepalWidth" placeholder="SepalWidth" require
14 <input type="text" name="PetalLength" placeholder="PetalLength" require
15 <input type="text" name="PetalWidth" placeholder="PetalWidth" require
16 <button type="submit" class="btn btn-primary btn-block btn-large">Predi
17 </form>
18 <br>
19 <br>
20 {{ prediction_text }}
21 </div>
```

Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 279-862-091
127.0.0.1 - - [25/Oct/2022 21:33:56] "GET / HTTP/1.1" 200 -
C:\Users\intwe\AppData\Local\Programs\Python\Python310\lib\site-packages\sklea
rn\base.py:450: UserWarning: X does not have valid feature names, but Logistic
Regression was fitted with feature names
warnings.warn(
127.0.0.1 - - [25/Oct/2022 21:34:02] "POST /predict HTTP/1.1" 200 -

4. HTML

← → 127.0.0.1:5000

Predict Salary Analysis

5. Prediction

← → 127.0.0.1:5000/predict

Predict Salary Analysis

The flower should be *Iris-virginica*