Data Science Internship at Data Glacier

Week 4: Flask Deployment

Name: Trusti Bhavik Patel

Batch code: LISUM10

Submission date: 28/06/2022

Submitted to: Data Glacier

Table of Content

1. Introduction

In this task we are building a machine learning model in python language. We’ll use Flask framework to deploy machine learning model on local server.

We’ll train machine learning model API to score the quality of red wine based on various given input. Moving forward we’ll run and deploy that model on local server and get the results from web.

2. Dataset Information

The dataset is about the quality of the red wine. We have several columns as a contents in the wine. We'll use those data as an input to predict the quality of wine.

Dataset includes 12 columns and 1599 rows.

index	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0	2.6	0.098	25	67	0.9968	3.2	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15	54	0.997	3.26	0.65	9.8	5

[First 3 rows from dataset]

Input variables (based on physicochemical tests):

1 - fixed acidity

2 - volatile acidity

3 - citric acid

4 - residual sugar

5 - chlorides

6 - free sulfur dioxide

7 - total sulfur dioxide

8 - density

9 - pH

10 - sulphates

11 - alcohol

Output variable (based on sensory data):

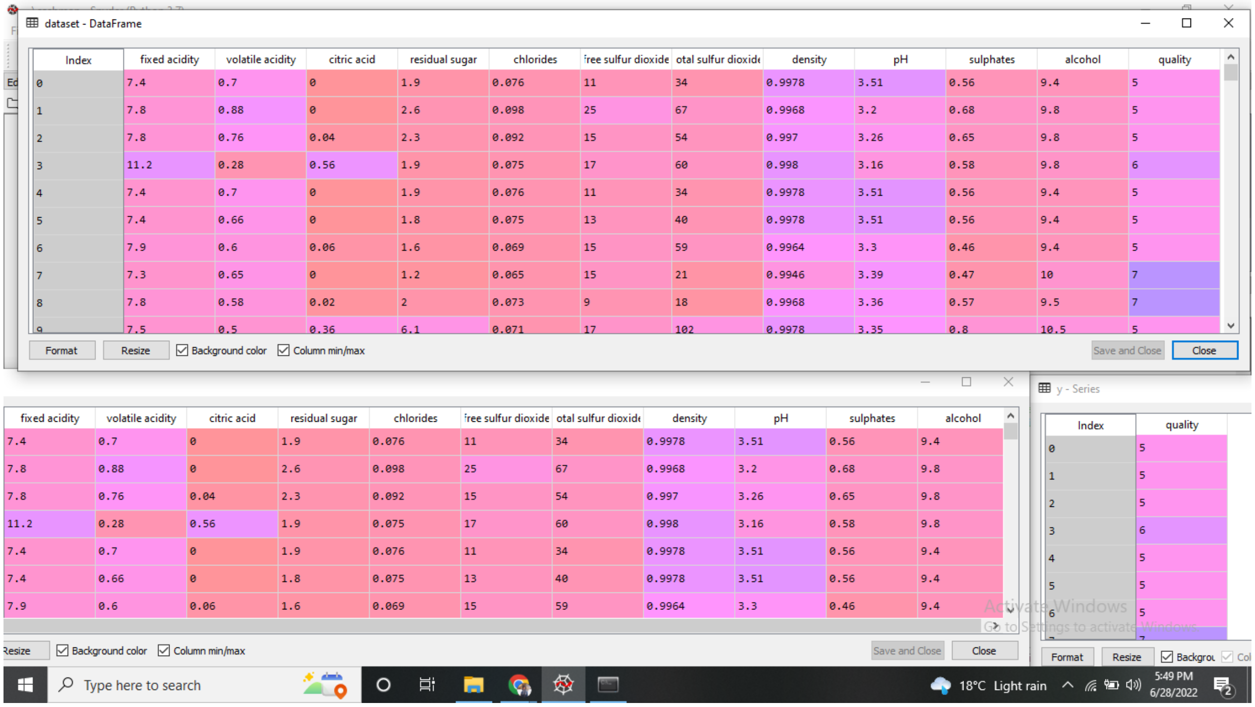
12 - quality (score between 0 and 10)

- *If the score is greater than 6.5 then the quality is good.*

3. Machine Learning Model

To create a machine learning model we have imported different libraries and load the dataset into python file.

Then to train the model we have used first 11 columns mentioned above and in the results we get column 12 as output. In below screenshot we can see the full dataset and then the dataset used for training and for testing.



Index	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	otal sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0	2.6	0.098	25	67	0.9968	3.2	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15	54	0.997	3.26	0.65	9.8	5
3	11.2	0.28	0.56	1.9	0.075	17	60	0.998	3.16	0.58	9.8	6
4	7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
5	7.4	0.66	0	1.8	0.075	13	40	0.9978	3.51	0.56	9.4	5
6	7.9	0.6	0.06	1.6	0.069	15	59	0.9964	3.3	0.46	9.4	5
7	7.3	0.65	0	1.2	0.065	15	21	0.9946	3.39	0.47	10	7
8	7.8	0.58	0.02	2	0.073	9	18	0.9968	3.36	0.57	9.5	7
9	7.5	0.5	0.36	6.1	0.071	17	102	0.9978	3.35	0.8	10.5	5

fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	otal sulfur dioxide	density	pH	sulphates	alcohol
7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4
7.8	0.88	0	2.6	0.098	25	67	0.9968	3.2	0.68	9.8
7.8	0.76	0.04	2.3	0.092	15	54	0.997	3.26	0.65	9.8
11.2	0.28	0.56	1.9	0.075	17	60	0.998	3.16	0.58	9.8
7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4
7.4	0.66	0	1.8	0.075	13	40	0.9978	3.51	0.56	9.4
7.9	0.6	0.06	1.6	0.069	15	59	0.9964	3.3	0.46	9.4

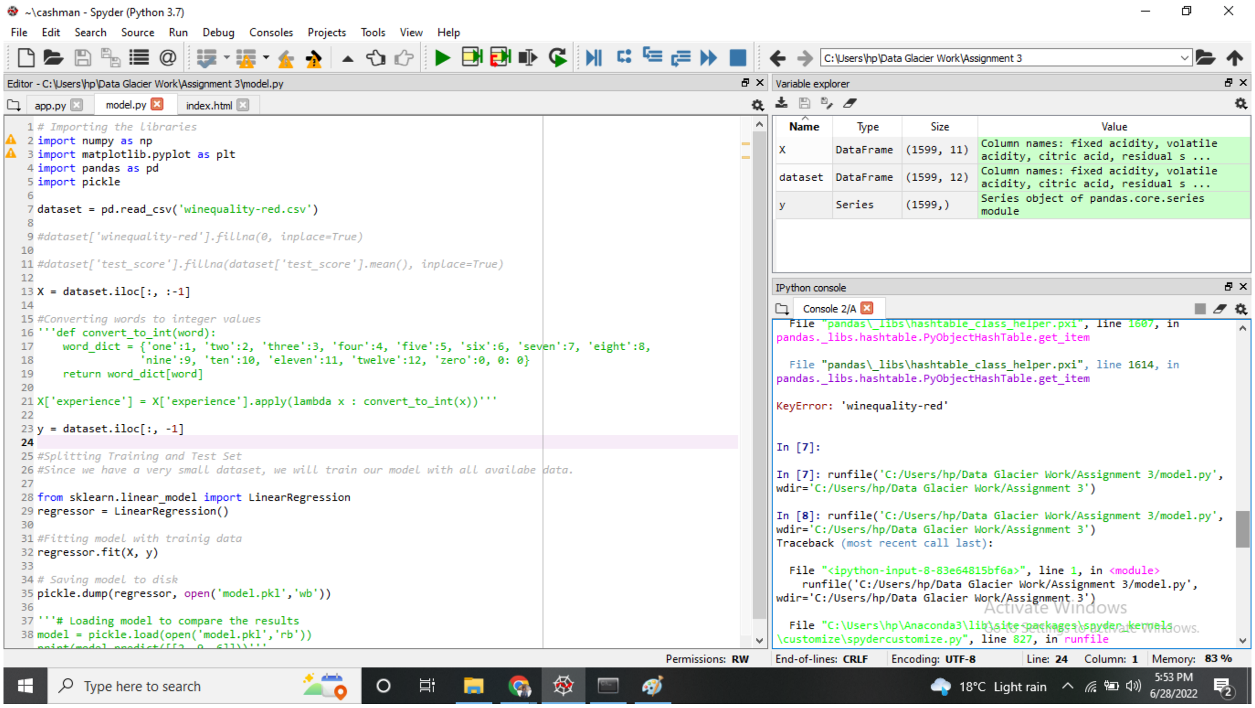
Index	quality
0	5
1	5
2	5
3	6
4	5
5	5
6	5

[Train and Test Dataset]

3.1 Model.py File

- We have used linear regression to train the model for this dataset.

- In below screenshot we can see the code for creating machine learning model and creating pickle file.


~\cashman - Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\hp\Data Glacier Work\Assignment 3
Editor - C:\Users\hp\Data Glacier Work\Assignment 3\model.py
app.py
model.py
index.html
Variable explorer
IPython console
Console 2/A
Permissions: RW
End-of-lines: CRLF
Encoding: UTF-8
Line: 24
Column: 1
Memory: 83 %
Type here to search
18°C Light rain
5:53 PM
6/28/2022

```python
# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import pickle

dataset = pd.read_csv('winequality-red.csv')

#dataset['winequality-red'].fillna(0, inplace=True)

#dataset['test_score'].fillna(dataset['test_score'].mean(), inplace=True)

X = dataset.iloc[:, :-1]

#Converting words to integer values
'''def convert_to_int(word):
    word_dict = {'one':1, 'two':2, 'three':3, 'four':4, 'five':5, 'six':6, 'seven':7, 'eight':8,
                'nine':9, 'ten':10, 'eleven':11, 'twelve':12, 'zero':0, 0: 0}
    return word_dict[word]

X['experience'] = X['experience'].apply(lambda x : convert_to_int(x))'''

y = dataset.iloc[:, -1]

#Splitting Training and Test Set
#Since we have a very small dataset, we will train our model with all availabe data.

from sklearn.linear_model import LinearRegression
regressor = LinearRegression()

#Fitting model with trainig data
regressor.fit(X, y)

# Saving model to disk
pickle.dump(regressor, open('model.pkl','wb'))

'''# Loading model to compare the results
model = pickle.load(open('model.pkl','rb'))
print(model.predict([[2, 9, 6]]))'''
```

Name	Type	Size	Value
X	DataFrame	(1599, 11)	Column names: fixed acidity, volatile acidity, citric acid, residual s ...
dataset	DataFrame	(1599, 12)	Column names: fixed acidity, volatile acidity, citric acid, residual s ...
y	Series	(1599,)	Series object of pandas.core.series module

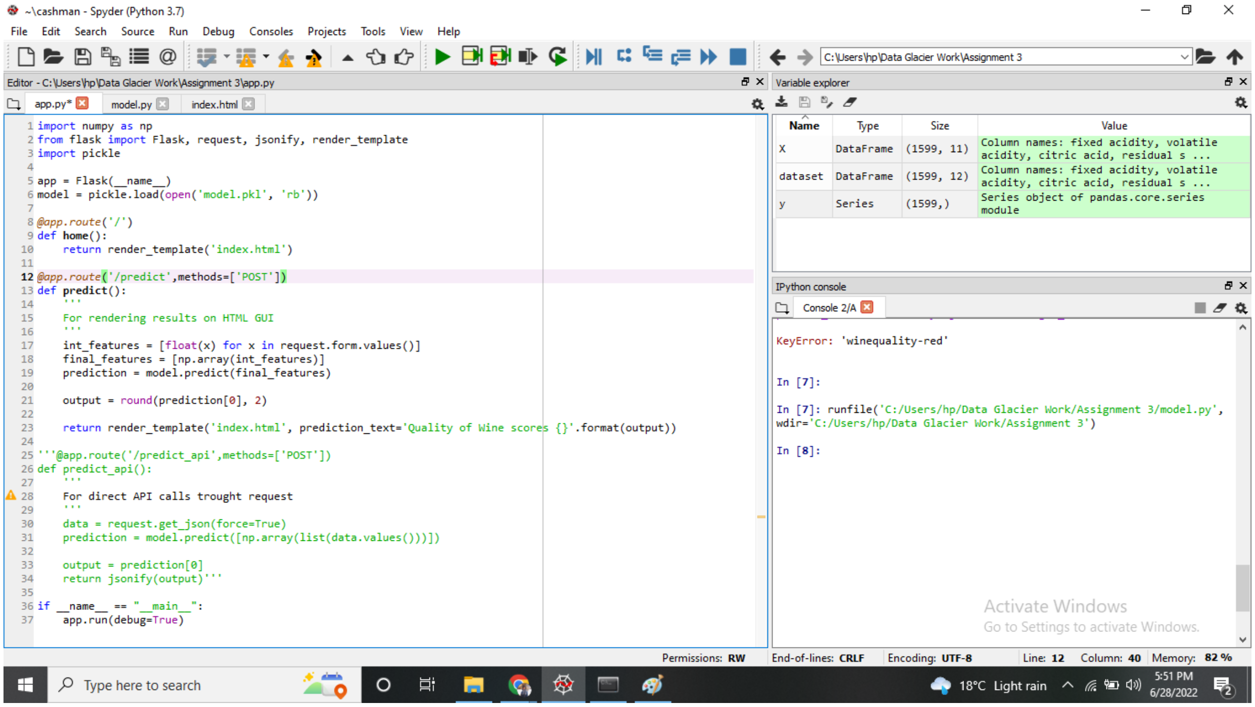
```
  File "pandas\_libs\hashtable_class_helper.pxi", line 1607, in
pandas._libs.hashtable.PyObjectHashTable.get_item

  File "pandas\_libs\hashtable_class_helper.pxi", line 1614, in
pandas._libs.hashtable.PyObjectHashTable.get_item

KeyError: 'winequality-red'


In [7]:

In [7]: runfile('C:/Users/hp/Data Glacier Work/Assignment 3/model.py',
wdir='C:/Users/hp/Data Glacier Work/Assignment 3')

In [8]: runfile('C:/Users/hp/Data Glacier Work/Assignment 3/model.py',
wdir='C:/Users/hp/Data Glacier Work/Assignment 3')
Traceback (most recent call last):

  File "<ipython-input-8-83e64815bf6a>", line 1, in <module>
    runfile('C:/Users/hp/Data Glacier Work/Assignment 3/model.py',
wdir='C:/Users/hp/Data Glacier Work/Assignment 3')

  File "C:\Users\hp\Anaconda3\lib\site-packages\spyder_kernels
\customize\spydercustomize.py", line 827, in runfile
```

[Model.py file]

4. Flask Model generation and Deployment

4.1 App.py file

- To deploy the model on the flask we first need to generate the app.py file which contains the route of the python file and it will help to deploy the flask code.


~\cashman - Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\hp\Data Glacier Work\Assignment 3
Editor - C:\Users\hp\Data Glacier Work\Assignment 3\app.py
app.py* model.py index.html
Variable explorer
IPython console
Console 2/A
Activate Windows
Go to Settings to activate Windows.
Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 12 Column: 40 Memory: 82 %
Type here to search
18°C Light rain
5:51 PM
6/28/2022

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

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

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

@app.route('/predict',methods=['POST'])
def predict():
    '''
    For rendering results on HTML GUI
    '''
    int_features = [float(x) for x in request.form.values()]
    final_features = [np.array(int_features)]
    prediction = model.predict(final_features)

    output = round(prediction[0], 2)

    return render_template('index.html', prediction_text='Quality of Wine scores {}'.format(output))

'''@app.route('/predict_api',methods=['POST'])
def predict_api():
    '''
    For direct API calls trought request
    '''
    data = request.get_json(force=True)
    prediction = model.predict([np.array(list(data.values()))])

    output = prediction[0]
    return jsonify(output)'''

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

Name	Type	Size	Value
X	DataFrame	(1599, 11)	Column names: fixed acidity, volatile acidity, citric acid, residual s ...
dataset	DataFrame	(1599, 12)	Column names: fixed acidity, volatile acidity, citric acid, residual s ...
y	Series	(1599,)	Series object of pandas.core.series module

```
KeyError: 'winequality-red'


In [7]:

In [7]: runfile('C:/Users/hp/Data Glacier Work/Assignment 3/model.py',
wdir='C:/Users/hp/Data Glacier Work/Assignment 3')

In [8]:
```

[App.py file]

- We use (__name__) to initialize the flask instance. Let Flask know where to find the HTML template folder (templates) in the same directory.

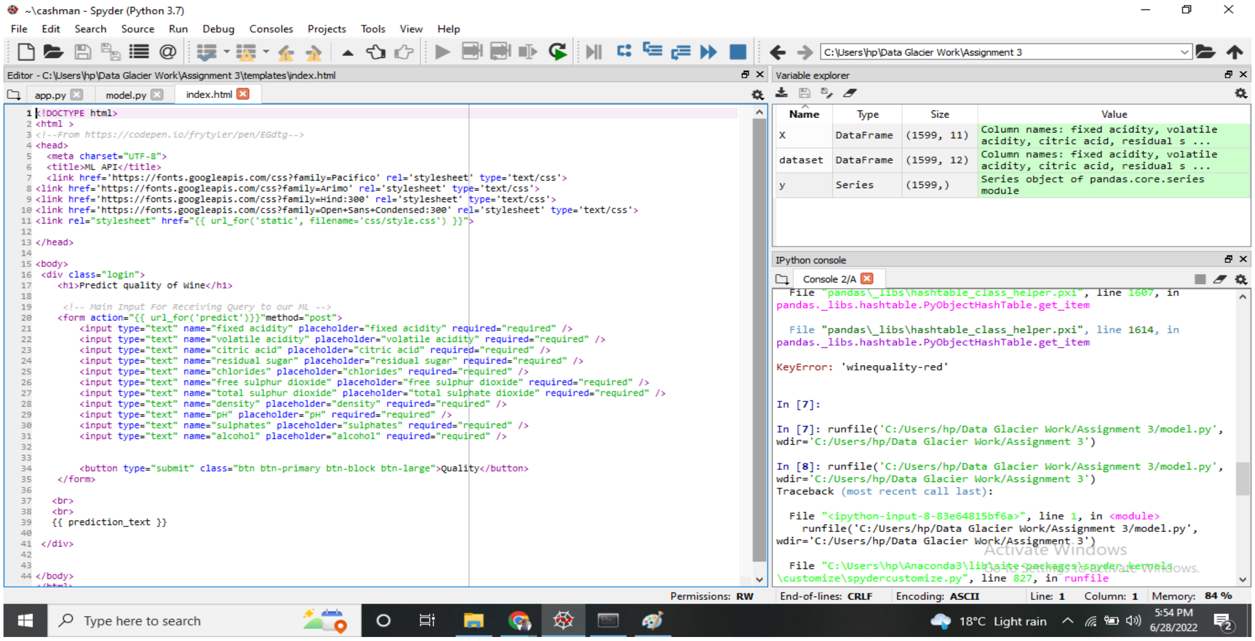
- Route decorator (@app.route('/')) to specify the URL that should trigger the execution of the home function. The home function will be rendered to index.html file.

- We used the POST method to transport the form data to the server in the message body.

- Lastly, we used the run function to only run the application on the server when the Python interpreter directly executes this script, which we ensured using the if statement with __name__ == '__main__'.

4.2 Index.html file

- Here is the HTML file to execute the python file on the web. It contains all the column name and the data to be predicted.



~\cashman - Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
Editor - C:\Users\hp\Data Glacier Work\Assignment 3\templates\index.html
app.py model.py index.html
C:\Users\hp\Data Glacier Work\Assignment 3
Variable explorer
IPython console
Console 2/A
Permissions: RW End-of-lines: CRLF Encoding: ASCII Line: 1 Column: 1 Memory: 84 %
Type here to search
18°C Light rain 5:54 PM 6/28/2022

```html
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>ML API</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'>
<link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">

</head>

<body>
 <div class="login">
    <h1>Predict quality of Wine</h1>

     <!-- Main Input For Receiving Query to our ML -->
    <form action="{{ url_for('predict')}}"method="post">
        <input type="text" name="fixed acidity" placeholder="fixed acidity" required="required" />
        <input type="text" name="volatile acidity" placeholder="volatile acidity" required="required" />
        <input type="text" name="citric acid" placeholder="citric acid" required="required" />
        <input type="text" name="residual sugar" placeholder="residual sugar" required="required" />
        <input type="text" name="chlorides" placeholder="chlorides" required="required" />
        <input type="text" name="free sulphur dioxide" placeholder="free sulphur dioxide" required="required" />
        <input type="text" name="total sulphur dioxide" placeholder="total sulphate dioxide" required="required" />
        <input type="text" name="density" placeholder="density" required="required" />
        <input type="text" name="pH" placeholder="pH" required="required" />
        <input type="text" name="sulphates" placeholder="sulphates" required="required" />
        <input type="text" name="alcohol" placeholder="alcohol" required="required" />


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

   <br>
   <br>
   {{ prediction_text }}

 </div>


</body>
</html>
```

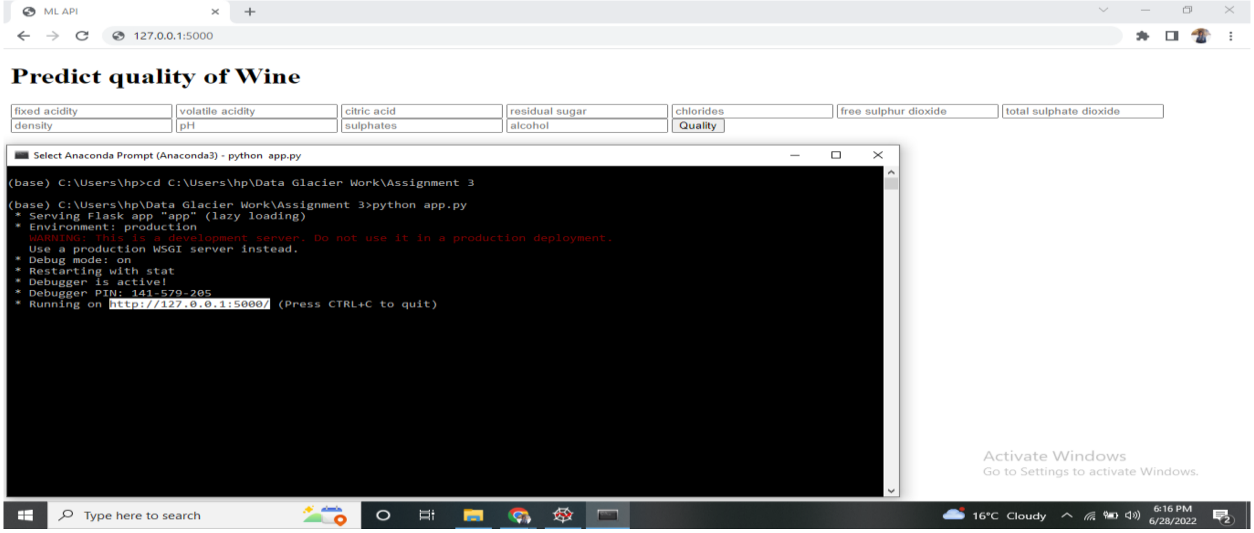
Name	Type	Size	Value
X	DataFrame	(1599, 11)	Column names: fixed acidity, volatile acidity, citric acid, residual s ...
dataset	DataFrame	(1599, 12)	Column names: fixed acidity, volatile acidity, citric acid, residual s ...
y	Series	(1599,)	Series object of pandas.core.series module

```
  File "pandas\_libs\hashtable_class_helper.pxi", line 1607, in
pandas._libs.hashtable.PyObjectHashTable.get_item

  File "pandas\_libs\hashtable_class_helper.pxi", line 1614, in
pandas._libs.hashtable.PyObjectHashTable.get_item

KeyError: 'winequality-red'


In [7]:

In [7]: runfile('C:/Users/hp/Data Glacier Work/Assignment 3/model.py',
wdir='C:/Users/hp/Data Glacier Work/Assignment 3')

In [8]: runfile('C:/Users/hp/Data Glacier Work/Assignment 3/model.py',
wdir='C:/Users/hp/Data Glacier Work/Assignment 3')
Traceback (most recent call last):

  File "<ipython-input-8-83e64815bf6a>", line 1, in <module>
    runfile('C:/Users/hp/Data Glacier Work/Assignment 3/model.py',
wdir='C:/Users/hp/Data Glacier Work/Assignment 3')

  File "C:\Users\hp\Anaconda3\lib\site-packages\spyder_kernels
\customize\spydercustomize.py", line 827, in runfile
```

[index.html file]

4.3 Executing Flask model in cmd

- Then we can simply go on cmd and run the app.py file. In below screenshot we can see that file ran and we got the link for local server.

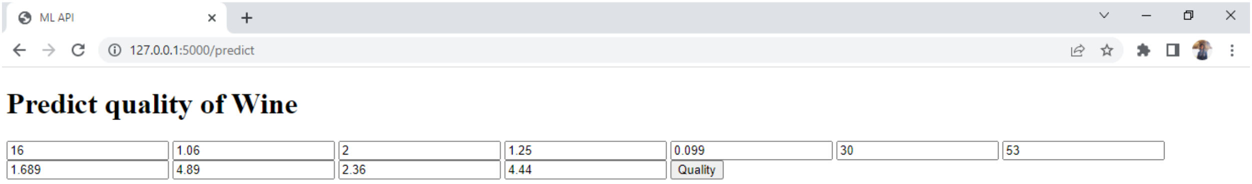


```
(base) C:\Users\hp>cd C:\Users\hp\Data Glacier Work\Assignment 3

(base) C:\Users\hp\Data Glacier Work\Assignment 3>python 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: on
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 141-579-205
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

[cmd running screen]

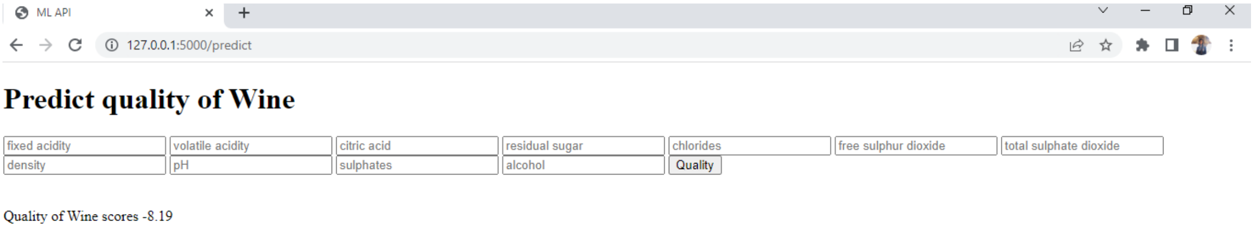
4.4 Results

Result 1 :



Predict quality of Wine

16 | 1.06 | 2 | 1.25 | 0.099 | 30 | 53
1.689 | 4.89 | 2.36 | 4.44



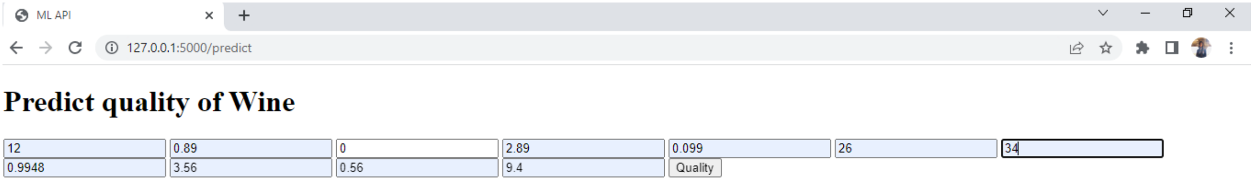
Predict quality of Wine

fixed acidity | volatile acidity | citric acid | residual sugar | chlorides | free sulphur dioxide | total sulphate dioxide
density | pH | sulphates | alcohol

Quality of Wine scores -8.19

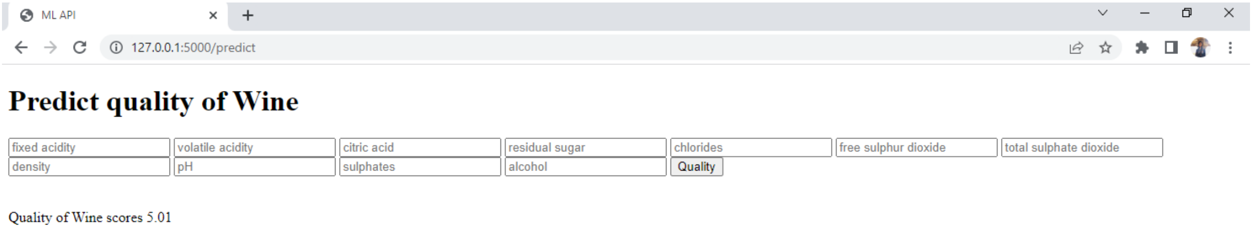
[Result of predicted data 1]

Result 2:



Predict quality of Wine

12 | 0.89 | 0 | 2.89 | 0.099 | 26 | 34
0.9948 | 3.56 | 0.56 | 9.4



Predict quality of Wine

fixed acidity | volatile acidity | citric acid | residual sugar | chlorides | free sulphur dioxide | total sulphate dioxide
density | pH | sulphates | alcohol

Quality of Wine scores 5.01

[Result of predicted data 2]