

1. Starting with the **train.py** file to train the model

```
PS P:\Great Learning\Qualities in Intelligent Systems\Week 14 Assignment\Part-2> python .\train.py
Windows-10-10.0.22631-SP0
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 2023, 13:26:23) [MSC v.1916 64 bit (AMD64)]
NumPy 1.24.3
SciPy 1.11.4
Shape of the training data
(1300, 160)
LDA model trained
Neural network model trained
Random forest model trained
Training complete
PS P:\Great Learning\Qualities in Intelligent Systems\Week 14 Assignment\Part-2> python .\inference.py
Windows-10-10.0.22631-SP0
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 2023, 13:26:23) [MSC v.1916 64 bit (AMD64)]
NumPy 1.24.3
SciPy 1.11.4
Shape of the test data
(1300, 160)
Linear Discriminant Analysis Model
LDA score and classification:
0.6915384615384615
LDA Prediction: [ 0  0  0 ... 25 25 25]
LDA Classification Report:
      precision    recall  f1-score   support

0         0.57        0.90        0.70         50
1         1.00        0.32        0.48         50
2         1.00        0.88        0.94         50
3         0.22        0.32        0.26         50
4         0.80        0.80        0.80         50
5         0.59        0.68        0.63         50
6         0.89        0.80        0.84         50
7         0.14        0.10        0.12         50
8         0.42        0.88        0.57         50
9         0.92        0.44        0.59         50
10        0.61        0.80        0.69         50
11        0.75        0.80        0.78         50
12        1.00        1.00        1.00         50
13        1.00        0.92        0.96         50
14        1.00        0.04        0.08         50
15        0.00        0.00        0.00         50
16        0.10        0.22        0.14         50
17        1.00        1.00        1.00         50
18        1.00        1.00        1.00         50
19        1.00        1.00        1.00         50
20        1.00        1.00        1.00         50
21        1.00        1.00        1.00         50
22        1.00        1.00        1.00         50
23        1.00        1.00        1.00         50
24        0.89        1.00        0.94         50
25        0.90        0.88        0.89         50

accuracy          0.69       1300
macro avg         0.73        0.69       0.68       1300
weighted avg      0.73        0.69       0.68       1300

Neural Networks Model
NN score and classification:
0.6615384615384615
NN Prediction: [ 0  0  0 ... 25 25 24]
NN Classification Report:
      precision    recall  f1-score   support

0         0.57        0.90        0.70         50
1         1.00        0.32        0.48         50
2         1.00        0.88        0.94         50
3         0.22        0.32        0.26         50
4         0.80        0.80        0.80         50
5         0.59        0.68        0.63         50
6         0.89        0.80        0.84         50
7         0.14        0.10        0.12         50
8         0.42        0.88        0.57         50
9         0.92        0.44        0.59         50
10        0.61        0.80        0.69         50
11        0.75        0.80        0.78         50
12        1.00        1.00        1.00         50
13        1.00        0.92        0.96         50
14        1.00        0.04        0.08         50
15        0.00        0.00        0.00         50
16        0.10        0.22        0.14         50
17        1.00        1.00        1.00         50
18        1.00        1.00        1.00         50
19        1.00        1.00        1.00         50
20        1.00        1.00        1.00         50
21        1.00        1.00        1.00         50
22        1.00        1.00        1.00         50
23        1.00        1.00        1.00         50
24        0.89        1.00        0.94         50
25        0.90        0.88        0.89         50

accuracy          0.69       1300
macro avg         0.73        0.69       0.68       1300
weighted avg      0.73        0.69       0.68       1300
```

2. Then running the **inference.py** file to test the model

```
LDA Prediction: [ 0  0  0 ... 25 25 25]
LDA Classification Report:
      precision    recall  f1-score   support

0         0.57        0.90        0.70         50
1         1.00        0.32        0.48         50
2         1.00        0.88        0.94         50
3         0.22        0.32        0.26         50
4         0.80        0.80        0.80         50
5         0.59        0.68        0.63         50
6         0.89        0.80        0.84         50
7         0.14        0.10        0.12         50
8         0.42        0.88        0.57         50
9         0.92        0.44        0.59         50
10        0.61        0.80        0.69         50
11        0.75        0.80        0.78         50
12        1.00        1.00        1.00         50
13        1.00        0.92        0.96         50
14        1.00        0.04        0.08         50
15        0.00        0.00        0.00         50
16        0.10        0.22        0.14         50
17        1.00        1.00        1.00         50
18        1.00        1.00        1.00         50
19        1.00        1.00        1.00         50
20        1.00        1.00        1.00         50
21        1.00        1.00        1.00         50
22        1.00        1.00        1.00         50
23        1.00        1.00        1.00         50
24        0.89        1.00        0.94         50
25        0.90        0.88        0.89         50

accuracy          0.69       1300
macro avg         0.73        0.69       0.68       1300
weighted avg      0.73        0.69       0.68       1300

Neural Networks Model
NN score and classification:
0.6615384615384615
NN Prediction: [ 0  0  0 ... 25 25 24]
NN Classification Report:
      precision    recall  f1-score   support

0         0.57        0.90        0.70         50
1         1.00        0.32        0.48         50
2         1.00        0.88        0.94         50
3         0.22        0.32        0.26         50
4         0.80        0.80        0.80         50
5         0.59        0.68        0.63         50
6         0.89        0.80        0.84         50
7         0.14        0.10        0.12         50
8         0.42        0.88        0.57         50
9         0.92        0.44        0.59         50
10        0.61        0.80        0.69         50
11        0.75        0.80        0.78         50
12        1.00        1.00        1.00         50
13        1.00        0.92        0.96         50
14        1.00        0.04        0.08         50
15        0.00        0.00        0.00         50
16        0.10        0.22        0.14         50
17        1.00        1.00        1.00         50
18        1.00        1.00        1.00         50
19        1.00        1.00        1.00         50
20        1.00        1.00        1.00         50
21        1.00        1.00        1.00         50
22        1.00        1.00        1.00         50
23        1.00        1.00        1.00         50
24        0.89        1.00        0.94         50
25        0.90        0.88        0.89         50

accuracy          0.69       1300
macro avg         0.73        0.69       0.68       1300
weighted avg      0.73        0.69       0.68       1300
```

```
File Edit Selection View Go Run ... Week 14 Assignment
PROBLEMS PORTS SQL CONSOLE OUTPUT DEBUG CONSOLE TERMINAL
NN Prediction: [ 0 0 0 ... 25 25 24]
NN Classification Report:
precision recall f1-score support
0 0.75 1.00 0.85 50
1 1.00 0.64 0.78 50
2 0.96 0.90 0.93 50
3 0.96 0.96 0.96 50
4 0.40 0.16 0.23 50
5 0.41 0.42 0.42 50
6 0.85 0.98 0.91 50
7 0.15 0.20 0.17 50
8 0.40 0.52 0.45 50
9 0.58 0.72 0.64 50
10 0.62 0.32 0.42 50
11 0.62 0.74 0.67 50
12 0.86 0.86 0.86 50
13 0.82 0.80 0.81 50
14 0.38 0.20 0.26 50
15 0.84 0.84 0.84 50
16 0.80 0.80 0.80 50
17 0.96 1.00 0.98 50
18 1.00 1.00 1.00 50
19 1.00 1.00 1.00 50
20 0.98 1.00 0.99 50
21 1.00 1.00 1.00 50
22 0.93 0.84 0.88 50
23 0.93 1.00 0.96 50
24 0.89 1.00 0.94 50
25 1.00 0.88 0.94 50

accuracy 0.66 1300
macro avg 0.68 0.66 0.66 1300
weighted avg 0.68 0.66 0.66 1300

Random Forest Classifier Model
Random Forest Classifier score and classification:
0.6330769230769231
Prediction: [ 0 0 25 ... 25 25 25]
Classification Report:
precision recall f1-score support
0 0.55 0.44 0.49 50
1 1.00 0.54 0.70 50
2 1.00 0.88 0.94 50
3 0.16 0.10 0.12 50
4 0.23 0.16 0.19 50
5 0.62 0.20 0.30 50
6 0.75 0.82 0.78 50
7 0.15 0.16 0.15 50
8 0.30 0.70 0.42 50
9 0.74 0.40 0.52 50
10 0.49 0.76 0.60 50
11 0.65 0.70 0.67 50
12 0.95 0.78 0.86 50
13 0.83 0.86 0.84 50
14 0.77 0.68 0.72 50
15 0.00 0.00 0.00 50
16 0.00 0.00 0.00 50
17 0.57 0.70 0.63 50
18 0.83 0.90 0.87 50
19 1.00 1.00 1.00 50
20 1.00 1.00 1.00 50
21 1.00 1.00 1.00 50
22 1.00 0.88 0.94 50
23 0.93 0.86 0.90 50
24 0.89 0.94 0.91 50
25 0.59 1.00 0.74 50

accuracy 0.63 1300
macro avg 0.65 0.63 0.63 1300
weighted avg 0.65 0.63 0.63 1300

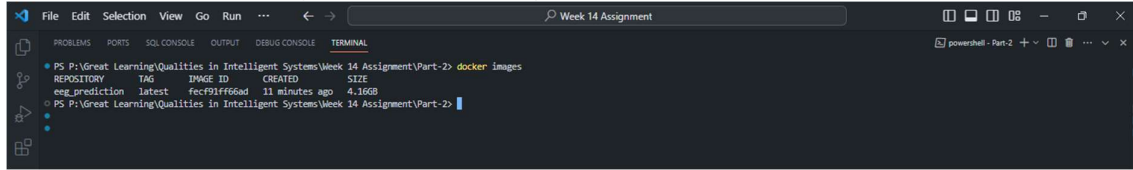
Testing complete
PS P:\Great Learning\Qualities in Intelligent Systems\Week 14 Assignment\Part-2>
```

3. Building the Docker image with name eeg_prediction

```
File Edit Selection View Go Run ... Week 14 Assignment
TERMINAL
PS P:\Great Learning\Qualities in Intelligent Systems\Week 14 Assignment\Part-2> docker build -t eeg_prediction .
[*] Building 1246.1s (13/13) FINISHED
-> [internal] load build definition from Dockerfile
-> >> transferring dockerfile: 224B
-> [internal] load metadata for docker.io/jupyter/scipy-notebook:latest
-> [auth] jupyter/scipy-notebook:pull token for registry-1.docker.io
-> [internal] load .dockerignore
-> >> transferring context: 2B
-> [1/7] FROM docker.io/jupyter/scipy-notebook:latest@sha256:fc4bccc3cb49d9a15ee4d6c666adf1776c958da9fa94a4fba0455c4ad33
-> resolve docker.io/jupyter/scipy-notebook:latest@sha256:fc4bccc3cb49d9a15ee4d6c666adf1776c958da9fa94a4fba0455c4ad33
-> sha256:fc4bccc3cb49d9a15ee4d6c666adf1776c958da9fa94a4fba0455c4ad33 772B / 772B
-> sha256:1f44e4f08f4822ec325e955726384bd300485e4e26a50721a29213bc 6.79kB / 6.79kB
-> sha256:ad5f5cfebd3d8a59b0cca3f790f1e38a4e56bd3654c0ba05eb4f1e4ebd9e 18.57kB / 18.57kB
-> sha256:aec8493d972ef43bf04ee3cda059c0f787f8f59c32fb3e48c87cb22a12e 29.54kB / 29.54kB
-> sha256:f02c713666cac5862b5a1e834eeb38c7191d5289f468cb04e08c8428a4 8.82kB / 8.82kB
-> sha256:088f11c11e4e07b0c4cfefb996c1321c084f8924cdeb111c27f6f7c0d7 679B / 679B
-> sha256:644fb70bf54461f9a0257aeb0ba8c18cd855748a6d75e84a39a6ac1 32B / 32B
-> sha256:ef837346080ac4a916da58072cb116efc568083a264d1d8a0e87e76256d 1.92kB / 1.92kB
-> sha256:77e45ee94dc8a1ce570e013482758833c72193814e42518a3db2a8594baee 4.92kB / 4.92kB
-> sha256:a30f89a86cf7c4872f93f4a452925f126892670a91102c83672f443a 151B / 151B
-> sha256:dc426c7a2b736a5380a496743f808a6745803c271c783026ce48046 276B / 276B
-> sha256:abaa8376a592612692c1750cc7aa50e045c234d1807743a39206c224f2e 104.82kB / 104.82kB
-> sha256:aa99b9b9e49a6a4a86d1015fa7be3a1dc1d37153a0c0767dac5c3d9f9abb 4.40kB / 4.40kB
-> sha256:822c4c4cf6a6b7668f90516cd166125d5de917a2f18c086ee2a6e52c1b7e9a 181B / 181B
-> sha256:025166dc7b4776705c07b7b729803b721983164240f414746f13d0b0904aae 30.50kB / 30.50kB
-> extracting sha256:aec8493d972ef43bf04ee3cda059c0f787f8f59c32fb3e48c87cb22a12e 42.75
-> sha256:2c4c9587ee486ad78644a71621ddcb4d1633c5f7f5840eb5a7dea786 1.15kB / 1.15kB
-> sha256:de2cd875fa8d3f6e191d8b1c0d8bfb677c1a16a0f1a67d660ef00da6dac 1.42kB / 1.42kB
-> sha256:75d3599f52514ff31055684d51b4bf6781fe41672dea7a5fa31557e 1.43kB / 1.43kB
-> extracting sha256:1f44e4f08f4822ec325e955726384bd300485e4e26a50721a29213bc 4.75
-> extracting sha256:088f11c11e4e07b0c4cfefb996c1321c084f8924cdeb111c27f6f7c0d7 0.05
-> extracting sha256:444fb70bf54461f9a0257aeb0ba8c18cd855748a6d75e84a39a6ac1 0.05
-> extracting sha256:ef837346080ac4a916da58072cb116efc568083a264d1d8a0e87e76256d 0.05
-> extracting sha256:77e45ee94dc8a1ce570e013482758833c72193814e42518a3db2a8594baee 0.05
-> extracting sha256:a30f89a86cf7c4872f93f4a452925f126892670a91102c83672f443a 0.05
-> extracting sha256:dc426c7a2b736a5380a496743f808a6745803c271c783026ce48046 0.05
-> sha256:964fc34ff93b08c77f28ca318e4e121dc657731150ef42f322d7f521c3 137.74kB / 137.74kB
-> sha256:31973ea24708181baedfba471cb29e8cad519f6e0cd488b6692b04e1ada5d 178.43kB / 178.43kB
-> extracting sha256:abaa8376a592612692c1750cc7aa50e045c234d1807743a39206c224f2e 234.55
-> sha256:96ee74439c77a5703d37c61ee62749216ebd514e813e96123a92076ccc967 1.47kB / 1.47kB
-> sha256:1f9ad23c07ac7228c821f2095f22735c64308e16873c52722cd5d113f7f880 437B / 437B
-> sha256:96ee74439c77a5703d37c61ee62749216ebd514e813e96123a92076ccc967 1.47kB / 1.47kB
-> sha256:1f9ad23c07ac7228c821f2095f22735c64308e16873c52722cd5d113f7f880 437B / 437B
-> sha256:d19266eb17753722a08d59ce152f5c2cd7642469c4cf6c20a824cc84c 1.26kB / 1.26kB
-> extracting sha256:31973ea24708181baedfba471cb29e8cad519f6e0cd488b6692b04e1ada5d 178.43kB / 178.43kB
-> extracting sha256:abaa8376a592612692c1750cc7aa50e045c234d1807743a39206c224f2e 234.55
-> sha256:96ee74439c77a5703d37c61ee62749216ebd514e813e96123a92076ccc967 1.47kB / 1.47kB
-> sha256:1f9ad23c07ac7228c821f2095f22735c64308e16873c52722cd5d113f7f880 437B / 437B
-> sha256:d19266eb17753722a08d59ce152f5c2cd7642469c4cf6c20a824cc84c 1.26kB / 1.26kB
-> extracting sha256:aa99b9b9e49a6a4a86d1015fa7be3a1dc1d37153a0c0767dac5c3d9f9abb 0.05
-> extracting sha256:822c4c4cf6a6b7668f90516cd166125d5de917a2f18c086ee2a6e52c1b7e9a 0.05
-> sha256:50894264e1e2168a701987221f5d6e6f5dccc207e5783a0aac482804 238B / 238B
-> extracting sha256:d2166dc7b4776705c07b7b729803b721983164240f414746f13d0b0904aae 9.46
-> sha256:96a202762a599a1ebad1a11e5d953f35736256a11599a8f369883cd04e3 546.28kB / 546.28kB
-> sha256:9a1656e6dc7764931ed13711e8d3d2866635d35740cfc1a1d303b0c1968d 279.53kB / 279.53kB
-> sha256:734ea8c3d94e1c3af924749a549398109f80abdad78e919717767c04ed76 597.22kB / 597.22kB
-> sha256:21816771274a261825f9d187c0669f4ac3b7398454970eb7042463 7.12kB / 7.12kB
-> extracting sha256:964fc34ff93b08c77f28ca318e4e121dc657731150ef42f322d7f521c3 238.75
-> extracting sha256:2c4c9587ee486ad78644a71621ddcb4d1633c5f7f5840eb5a7dea786 0.05
-> extracting sha256:de2cd875fa8d3f6e191d8b1c0d8bfb677c1a16a0f1a67d660ef00da6dac 0.05
-> extracting sha256:75d3599f52514ff31055684d51b4bf6781fe41672dea7a5fa31557e 0.05
-> extracting sha256:31973ea24708181baedfba471cb29e8cad519f6e0cd488b6692b04e1ada5d 100.15
-> extracting sha256:96ee74439c77a5703d37c61ee62749216ebd514e813e96123a92076ccc967 0.05
-> extracting sha256:1f9ad23c07ac7228c821f2095f22735c64308e16873c52722cd5d113f7f880 0.05
-> extracting sha256:d19266eb17753722a08d59ce152f5c2cd7642469c4cf6c20a824cc84c 0.05
-> sha256:9a1656e6dc7764931ed13711e8d3d2866635d35740cfc1a1d303b0c1968d 104.75
-> extracting sha256:734ea8c3d94e1c3af924749a549398109f80abdad78e919717767c04ed76 224.55
-> extracting sha256:a21a16771274a261825f9d187c0669f4ac3b7398454970eb7042463 3.75
-> [internal] load build context
-> >> transferring context: 2.69kB
-> [2/7] RUN pip install joblib
-> [3/7] COPY train.csv ./train.csv
-> [4/7] COPY test.csv ./test.csv
-> [5/7] COPY train.py ./train.py
-> [6/7] COPY inference.py ./inference.py
-> [7/7] RUN python3 train.py
-> exporting to image
-> exporting layers
-> writing image sha256:fec91ff66ad7c37ec26fae40d1d1939442b7a011b834e596f4d3e9569030
-> naming to docker.io/library/eeg_prediction

What's Next?
View a summary of image vulnerabilities and recommendations + docker scout quickview
PS P:\Great Learning\Qualities in Intelligent Systems\Week 14 Assignment\Part-2>
```

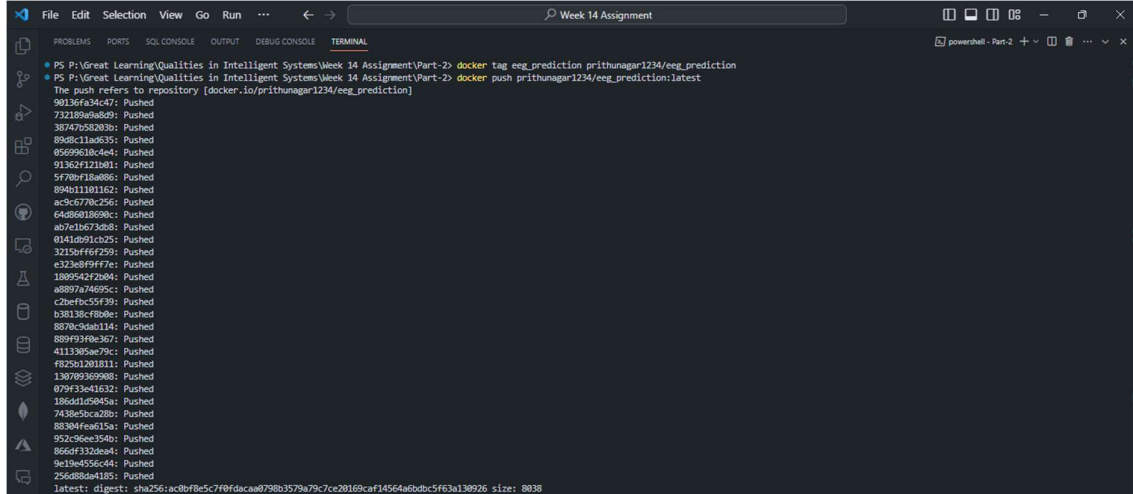
4. To view the Docker Image run **docker images**



```
File Edit Selection View Go Run ... Week 14 Assignment
PROBLEMS PORTS SQL CONSOLE OUTPUT DEBUG CONSOLE TERMINAL
PS P:\Great Learning\Qualities in Intelligent Systems\Week 14 Assignment\Part-2> docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
eeg_prediction latest fecf91f466ad 11 minutes ago 4.16GB
PS P:\Great Learning\Qualities in Intelligent Systems\Week 14 Assignment\Part-2>
```

5. Tagging the docker image and then pushing my docker image to my dockerhub account using the command

docker tag eeg_prediction prithunagar1234/eeg_prediction
docker push prithunagar1234/eeg_prediction:latest



```
File Edit Selection View Go Run ... Week 14 Assignment
PROBLEMS PORTS SQL CONSOLE OUTPUT DEBUG CONSOLE TERMINAL
PS P:\Great Learning\Qualities in Intelligent Systems\Week 14 Assignment\Part-2> docker tag eeg_prediction prithunagar1234/eeg_prediction
PS P:\Great Learning\Qualities in Intelligent Systems\Week 14 Assignment\Part-2> docker push prithunagar1234/eeg_prediction:latest
The push refers to repository [docker.io/prithunagar1234/eeg_prediction]
98136fa34c47: Pushed
732289e98d9: Pushed
3874765203b: Pushed
89d8c11ad635: Pushed
05699618c4e4: Pushed
91362f121001: Pushed
5f70bf13a886: Pushed
894b11101162: Pushed
ac9c6770c256: Pushed
64d86018690c: Pushed
ab7e1b673d8d: Pushed
0141d991cb29: Pushed
3215bfff6f259: Pushed
e323d8f9ff7e: Pushed
1809542f20a5: Pushed
a8897a74699c: Pushed
c2bfc55f39: Pushed
b38138cf8b0e: Pushed
8870c3dab114: Pushed
889f93f0e367: Pushed
4113305a79c: Pushed
f82501201811: Pushed
130709369908: Pushed
079f13e411632: Pushed
186d4105d84a: Pushed
7438e50ca28b: Pushed
88304fe615a: Pushed
952c96ee354b: Pushed
866df332de4d: Pushed
9c19e4556c44: Pushed
256d88de4185: Pushed
latest: digest: sha256:ac8bf8e5c7f0fdcaab0798b3579a79c7ce20169caf14564a60dbc5fe3a130926 size: 8038
```