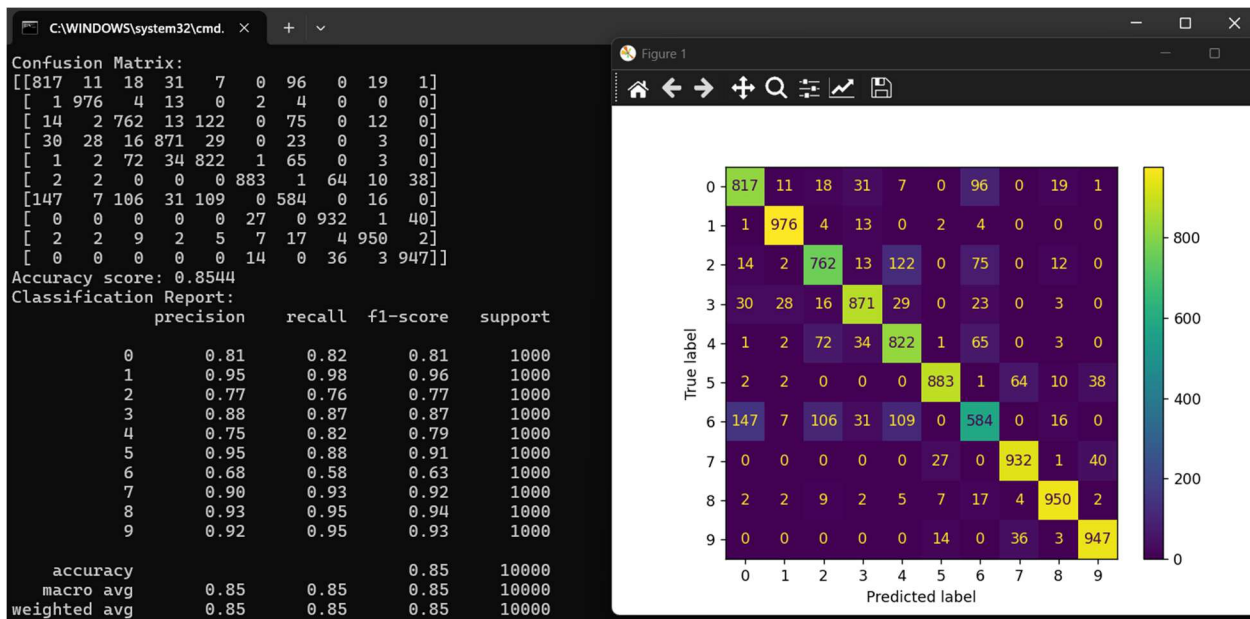1.

```python
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

#pip install opencv-python
import cv2 #use the install command above in: View->Other Windows->Python Environments->Packages (PyPI)
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score, ConfusionMatrixDisplay

#uncomment if using google colab:
#from google.colab.patches import cv2_imshow

train = pd.read_csv('fashion-mnist_train.csv')
test = pd.read_csv('fashion-mnist_test.csv')

train_img = train.drop('label', axis=1)
train_lbl = train['label']
test_img = test.drop('label', axis=1)
test_lbl = test['label']


logisticRegr = LogisticRegression(solver = 'lbfgs', multi_class='multinomial')

logisticRegr.fit(train_img, train_lbl)
#if predict a single image then reshape:
#Ypred = logisticRegr.predict(test_img[0].reshape(1, -1))

predictions = logisticRegr.predict(test_img)
score = logisticRegr.score(test_img, test_lbl)
#print(score)

confusionMatrix = confusion_matrix(test_lbl, predictions)
print(f'Confusion Matrix:\n{confusionMatrix}')
print(f'Accuracy score: {accuracy_score(test_lbl, predictions)}')
print(f'Classification Report:\n{classification_report(test_lbl, predictions)}')

disp = ConfusionMatrixDisplay(confusion_matrix=confusionMatrix,
    display_labels=logisticRegr.classes_)
disp.plot()
plt.show()
```

```
Confusion Matrix:
[[817  11  18  31   7   0  96   0  19   1]
 [  1 976   4  13   0   2   4   0   0   0]
 [ 14   2 762  13 122   0  75   0  12   0]
 [ 30  28  16 871  29   0  23   0   3   0]
 [  1   2  72  34 822   1  65   0   3   0]
 [  2   2   0   0   0 883   1  64  10  38]
 [147   7 106  31 109   0 584   0  16   0]
 [  0   0   0   0   0  27   0 932   1  40]
 [  2   2   9   2   5   7  17   4 950   2]
 [  0   0   0   0   0  14   0  36   3 947]]
Accuracy score: 0.8544
Classification Report:
              precision    recall  f1-score   support

           0       0.81      0.82      0.81      1000
           1       0.95      0.98      0.96      1000
           2       0.77      0.76      0.77      1000
           3       0.88      0.87      0.87      1000
           4       0.75      0.82      0.79      1000
           5       0.95      0.88      0.91      1000
           6       0.68      0.58      0.63      1000
           7       0.90      0.93      0.92      1000
           8       0.93      0.95      0.94      1000
           9       0.92      0.95      0.93      1000

    accuracy                           0.85     10000
   macro avg       0.85      0.85      0.85     10000
weighted avg       0.85      0.85      0.85     10000
```

2.

```
 2
 3  import numpy as np
 4  import matplotlib.pyplot as plt
 5  import pandas as pd
 6  import cv2 #use the install command above in: View->Other Windows->Python Environments->Packages (PyPI)
 7  from sklearn.model_selection import train_test_split
 8  from sklearn.linear_model import LogisticRegression
 9
10  labels_keys = ['T-shirt', 'Trouers', 'Pullover', 'Dress', 'Coat', 'Sandal', 'Shirt', 'Sneaker', 'Bag', 'Ankle-Boot']
11
12  train = pd.read_csv('fashion-mnist_train.csv')
13  test = pd.read_csv('fashion-mnist_test.csv')
14
15  train_img = train.drop('label', axis=1)
16  train_lbl = train['label']
17  test_img = test.drop('label', axis=1)
18  test_lbl = test['label']
19  bag = cv2.cvtColor(cv2.imread('bag.jpg'), cv2.COLOR_BGR2GRAY).reshape(1, 28 * 28)
20  trousers = cv2.cvtColor(cv2.imread('trousers.bmp'), cv2.COLOR_BGR2GRAY).reshape(1, 28 * 28)
21
22  logisticRegr = LogisticRegression(solver = 'lbfgs', multi_class='multinomial')
23  logisticRegr.fit(train_img, train_lbl)
24
25  print(f'Predicted img1: {labels_keys[int(logisticRegr.predict(bag))]}')
26  print(f'Predicted img2: {labels_keys[int(logisticRegr.predict(trousers))]}')
27
28
29
```

Predicted class for the bag image:

```
Predicted img1: Bag
```

Predicted class for the trousers image:

```
Predicted img2: Trouers
```