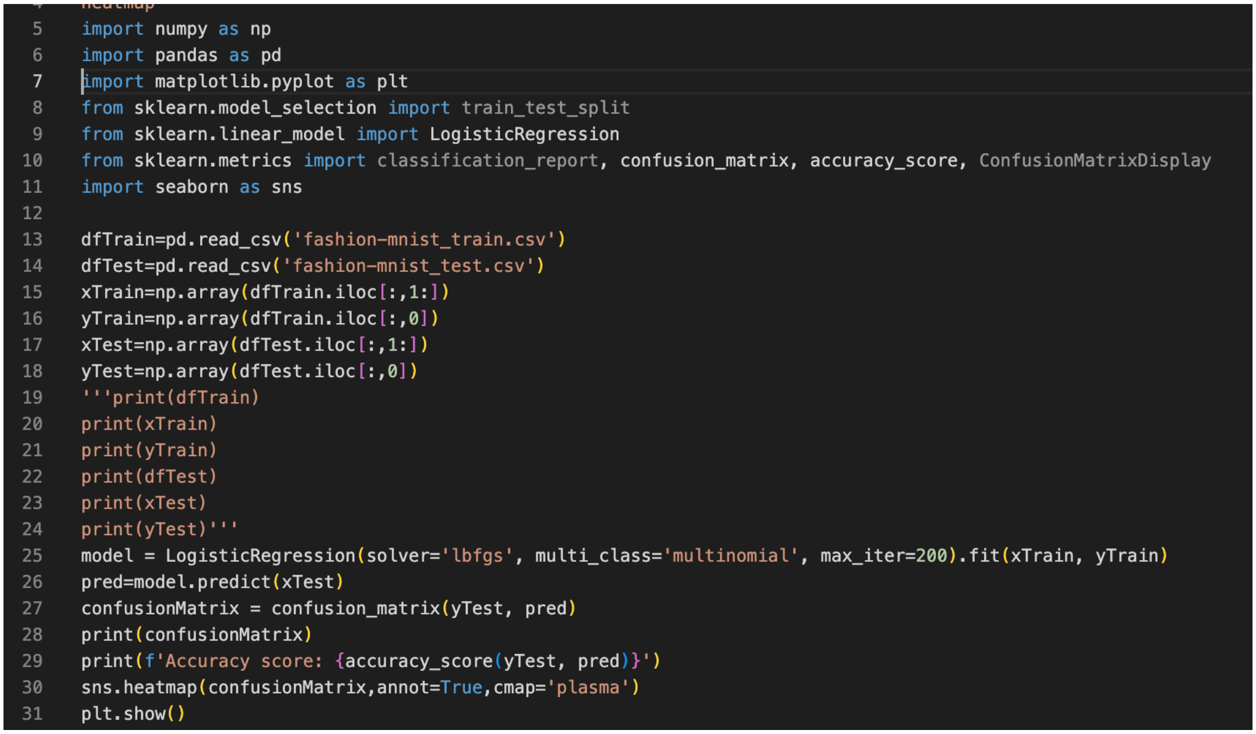


```python
heatmap
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score, ConfusionMatrixDisplay
import seaborn as sns

dfTrain=pd.read_csv('fashion-mnist_train.csv')
dfTest=pd.read_csv('fashion-mnist_test.csv')
xTrain=np.array(dfTrain.iloc[:,1:])
yTrain=np.array(dfTrain.iloc[:,0])
xTest=np.array(dfTest.iloc[:,1:])
yTest=np.array(dfTest.iloc[:,0])
'''print(dfTrain)
print(xTrain)
print(yTrain)
print(dfTest)
print(xTest)
print(yTest)'''
model = LogisticRegression(solver='lbfgs', multi_class='multinomial', max_iter=200).fit(xTrain, yTrain)
pred=model.predict(xTest)
confusionMatrix = confusion_matrix(yTest, pred)
print(confusionMatrix)
print(f'Accuracy score: {accuracy_score(yTest, pred)}')
sns.heatmap(confusionMatrix,annot=True,cmap='plasma')
plt.show()
```

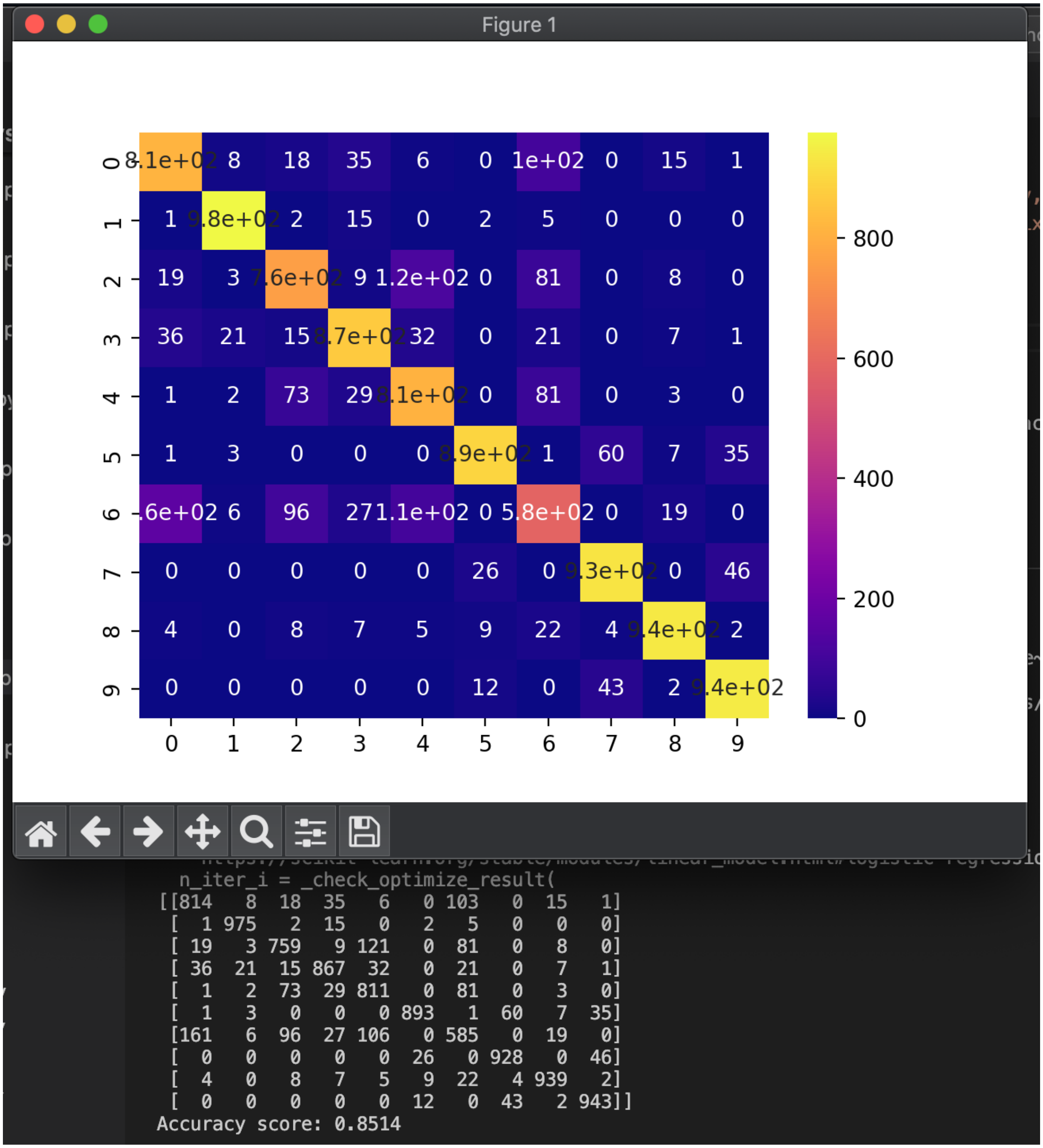
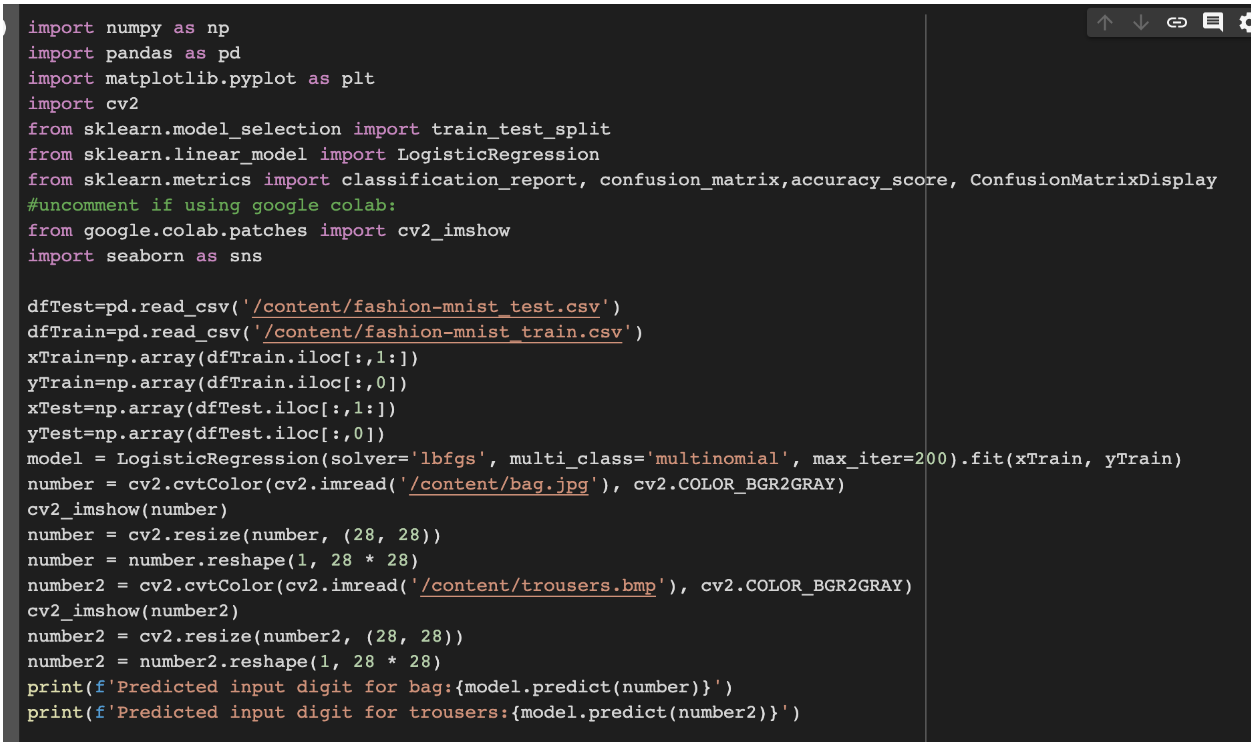


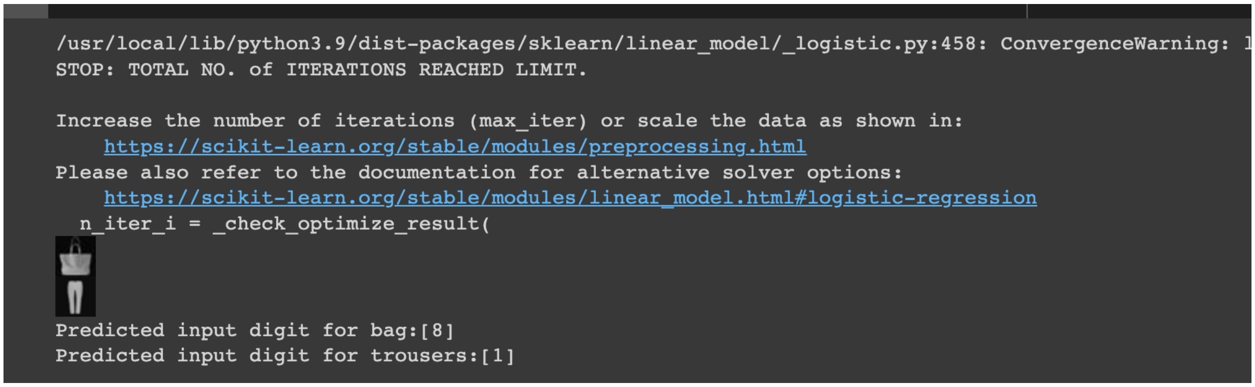
Figure 1
0 8.1e+02 8 18 35 6 0 1e+02 0 15 1
1 1 9.8e+02 2 15 0 2 5 0 0 0
2 19 3 7.6e+02 9 1.2e+02 0 81 0 8 0
3 36 21 15 8.7e+02 32 0 21 0 7 1
4 1 2 73 29 8.1e+02 0 81 0 3 0
5 1 3 0 0 0 8.9e+02 1 60 7 35
6 .6e+02 6 96 27 1.1e+02 0 5.8e+02 0 19 0
7 0 0 0 0 0 26 0 9.3e+02 0 46
8 4 0 8 7 5 9 22 4 9.4e+02 2
9 0 0 0 0 0 12 0 43 2 9.4e+02
0 1 2 3 4 5 6 7 8 9
800
600
400
200
0

```
  https://scikit-learn.org/stable/modules/linear_model.html#logistic-regressio
  n_iter_i = _check_optimize_result(
[[814   8  18  35   6   0 103   0  15   1]
 [  1 975   2  15   0   2   5   0   0   0]
 [ 19   3 759   9 121   0  81   0   8   0]
 [ 36  21  15 867  32   0  21   0   7   1]
 [  1   2  73  29 811   0  81   0   3   0]
 [  1   3   0   0   0 893   1  60   7  35]
 [161   6  96  27 106   0 585   0  19   0]
 [  0   0   0   0   0  26   0 928   0  46]
 [  4   0   8   7   5   9  22   4 939   2]
 [  0   0   0   0   0  12   0  43   2 943]]
Accuracy score: 0.8514
```



```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import cv2
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
import seaborn as sns

dfTest=pd.read_csv('/content/fashion-mnist_test.csv')
dfTrain=pd.read_csv('/content/fashion-mnist_train.csv')
xTrain=np.array(dfTrain.iloc[:,1:])
yTrain=np.array(dfTrain.iloc[:,0])
xTest=np.array(dfTest.iloc[:,1:])
yTest=np.array(dfTest.iloc[:,0])
model = LogisticRegression(solver='lbfgs', multi_class='multinomial', max_iter=200).fit(xTrain, yTrain)
number = cv2.cvtColor(cv2.imread('/content/bag.jpg'), cv2.COLOR_BGR2GRAY)
cv2_imshow(number)
number = cv2.resize(number, (28, 28))
number = number.reshape(1, 28 * 28)
number2 = cv2.cvtColor(cv2.imread('/content/trousers.bmp'), cv2.COLOR_BGR2GRAY)
cv2_imshow(number2)
number2 = cv2.resize(number2, (28, 28))
number2 = number2.reshape(1, 28 * 28)
print(f'Predicted input digit for bag:{model.predict(number)}')
print(f'Predicted input digit for trousers:{model.predict(number2)}')
```



```
/usr/local/lib/python3.9/dist-packages/sklearn/linear_model/_logistic.py:458: ConvergenceWarning: l
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  n_iter_i = _check_optimize_result(

Predicted input digit for bag:[8]
Predicted input digit for trousers:[1]
```