ENGR 421 / DASC 521: Introduction to Machine Learning
Homework 07: Linear Discriminant Analysis

Deadline: May 13, 2022, 11:59 PM

In this homework, you will implement the linear discriminant analysis algorithm in Python. Here are the steps you need to follow:

1. You are given a multiclass classification data set, which contains 4000 clothing images of size 28 pixels × 28 pixels (i.e., 784 pixels). These images are from ten distinct classes, namely, “t-shirt/top”, “trouser”, “pullover”, “dress”, “coat”, “sandal”, “shirt”, “sneaker”, “bag”, and “ankle boot”. The figure below shows five sample figures from each class. You are given two data files:
    a. `hw07_data_set_images.csv`: clothing images,
    b. `hw07_data_set_labels.csv`: corresponding class labels.

2. Divide the data set into two parts by assigning the first 2000 images to the training set and the remaining 2000 images to the test set.

3. Calculate $\boldsymbol{S}_W$ and $\boldsymbol{S}_B$ matrices. Your matrices should be like the following figures. (20 points)

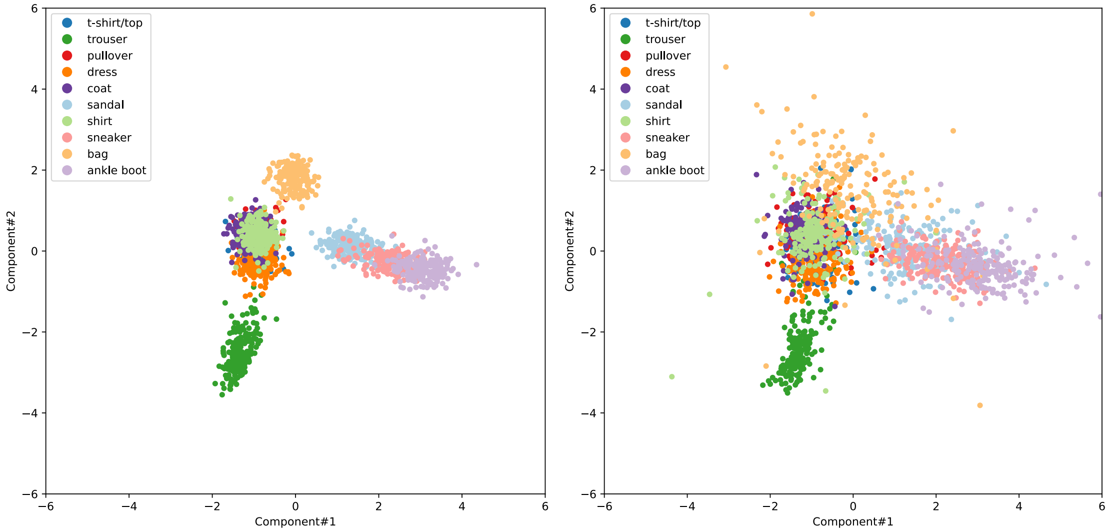
$$\boldsymbol{S}_W = \sum_{i=1}^{N}\sum_{c=1}^{K}(\boldsymbol{x}_i - \boldsymbol{\mu}_c)(\boldsymbol{x}_i - \boldsymbol{\mu}_c)^T y_{ic} \qquad \boldsymbol{S}_B = \sum_{i=1}^{N}\sum_{c=1}^{K}(\boldsymbol{\mu}_c - \boldsymbol{\mu})(\boldsymbol{\mu}_c - \boldsymbol{\mu})^T y_{ic}$$

```
print(SW[0:5, 0:5])
print(SB[0:5, 0:5])
[[195.0531401  138.98550725 138.24154589   273.64251208]
 [138.98550725 146.68407152 137.97108222   212.80522823]
 [138.24154589 137.97108222 237.79940915   222.04558827]
 [273.64251208 212.80522823 222.04558827 10381.73114607]]
[[0.8488599  0.86049275  1.26145411  4.64248792]
 [0.86049275 1.07392848  2.24791778  8.49977177]
 [1.26145411 2.24791778  6.68009085 25.25691173]
 [4.64248792 8.49977177 25.25691173 97.25635393]]
```

4. Calculate the largest nine eigenvalues and their corresponding eigenvectors of $\boldsymbol{S}_W^{-1}\boldsymbol{S}_B$ matrix. Your eigenvalues should be like the following figures. (10 points)

```
print(values[0:9])
[26.03085646 11.57909822 7.86259994
  5.12082313 3.60002676  3.22503144
  2.71288241 1.3688596   1.13340931]
```

5. Using the two eigenvectors that correspond to the largest two eigenvalues, project the training and test data points in a two-dimensional subspace using $z_i = \boldsymbol{W}^T(\boldsymbol{x}_i - \boldsymbol{\mu})$ formula. Draw these two-dimensional projections using a separate color for each class. Your figures should be like the following figures. (30 points)



6. Using the nine eigenvectors that correspond to the largest nine eigenvalues, project the training and test data points in a nine-dimensional subspace using $z_i = \boldsymbol{W}^T(\boldsymbol{x}_i - \boldsymbol{\mu})$ formula. Learn a k-nearest neighbor classifier by setting $k = 11$. Calculate confusion matrices on the training and test data points. Your confusion matrices should be like the following figures. (40 points)

```
print(confusion_matrix)
y_train   1    2    3    4    5    6    7    8    9    10
y_hat
1       203    0    1    1    0    0   15    0    0    0
2         0  192    0    1    0    0    0    0    0    0
3         0    0  198    0    3    0   11    0    0    0
4         2    2    1  202    3    0    4    0    0    0
5         0    0    8    1  201    0    9    0    0    0
6         0    0    0    0    0  175    0    3    0    0
7         9    0    2    6   11    0  168    0    1    0
8         0    0    0    0    0    3    0  196    0    1
9         0    0    0    0    0    0    0    0  183    0
10        0    0    0    0    0    0    0    0    0  184
print(confusion_matrix)
y_test   1    2    3    4    5    6   7    8    9    10
y_hat
1      142    1   21    7    1    2  42    0    6    2
2        0  183    1    4    2    0   1    0    0    0
3        3    1  115    1   25    1  26    0    5    0
4       17    3    4  167    9    7  14    0   12    4
5        0    0   44    8  143    5  39    0    3    1
6        1    1    2    0    1  118   3   18   17   12
7       28    0   16   11   17    5  83    0    3    0
8        0    0    0    1    0   32   0  167    2   14
9        4    0    2    1    1    7   4    0  135    0
10       0    0    0    0    0   25   1   19    5  172
```

What to submit: You need to submit your source code in a single file (.py file) named as ***STUDENTID.py***, where ***STUDENTID*** should be replaced with your 7-digit student number.

How to submit: Submit the file you created to Blackboard. Please follow the exact style mentioned and do not send a file named as ***STUDENTID.py***. Submissions that do not follow these guidelines will not be graded.

Late submission policy: Late submissions will not be graded.

Cheating policy: Very similar submissions will not be graded.