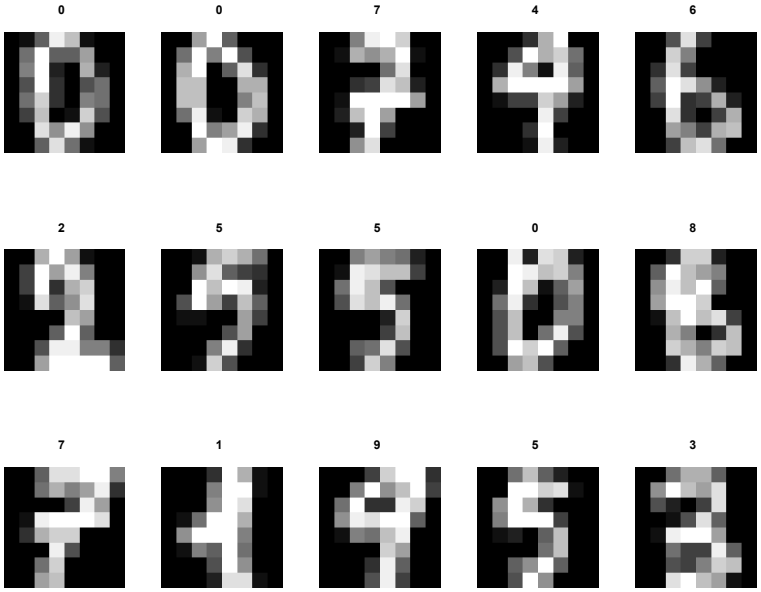
HW #6: Dimensionality Reduction & Clustering
Due: 10/20/21 (Wednesday) 11:59PM

Using Optdigits from the UCI repository, implement PCA in Python. Using eigenvectors, reconstruct the digit images and calculate the reconstruction error. Also, train a nearest mean classifier and calculate the prediction accuracies.

1. Load the data files (optdigits.tra & optdigits.tes) in Python. Note that the first file contains training data and the second file contains test data. Analyze the data file structure by reading the UCI repository website where you downloaded the files. After you read the data in, plot the first 15 images in "optdigits.tra" file as shown below. This practice is for understanding the data structure as well as meaning of the data.



2. Now, we will apply PCA to reduce the original dimension (64) of the data to two dimensions. To do that, try to form the covariance matrix and solve an eigenvalue problem using "eig" function in ***numpy*** module. Plot eigenvalues vs. eigenvector curve (Scree graph) as shown in Figure 6.4. in your textbook.

3. Next, select two eigenvectors with the largest and second largest eigenvalues. Then, project the original data to the eigenvectors and plot the resulting 2-dimensional data points. Try to label the first 100 data points as shown in Figure 6.5 in the textbook.

4. Now, reverse-project the data of the reduced dimension back to the original-dimensional space and calculate the reconstruction error using Eq. 6.12 in the textbook. Plot the reconstructed digit images like those in Problem 1 and compare them.

5. In the original-dimensional space, train a nearest mean classifier using the training data set and calculate the prediction accuracies by applying the trained classifier to the test data set. Report the prediction accuracy for each digit. Repeat the process in the reduced dimensional space formed by the two eigenvectors obtained in Problem 3. Report the prediction accuracy for each digit. Try to show all results in a single table.