**Problem 1. Principal Component Analysis**

Download three.txt and eight.txt, which can be found in our Piazza page. Each has 200 handwritten digits. Each line is for a digit, vectorized from a 16x16 gray scale image.

(a) Each line has 256 numbers: they are pixel values (0=black, 255=white) vectorized from the image as the first column (top down), the second column, and so on. Visualize using python the two gray scale images corresponding to the first line in three.txt and the first line in eight.txt.

(b) Put the two data files together (threes first, eights next) to form a $n \times d$ matrix $X$ where $n = 400$ digits and $d = 256$ pixels. The $i$-th row of $X$ is $x_i^\top$, where $x_i \in \mathbb{R}^d$ is the $i$-th image in the combined data set. Compute the sample mean $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$. Visualize $\bar{x}$ as a 16x16 gray scale image.

Proved it's correct with np.cov()

(c) Center $X$ using $\bar{x}$ above. Then form the sample covariance matrix $S = \frac{X^\top X}{n-1}$. Show the 5x5 submatrix $S(1 \dots 5, 1 \dots 5)$.

(d) Use appropriate software/library to compute the two largest eigenvalues $\lambda_1 \geq \lambda_2$ and the corresponding eigenvectors $v_1, v_2$ of $S$. For example, in python one can use `scipy.sparse.linalg.eigs`. Show the value of $\lambda_1, \lambda_2$. Visualize $v_1, v_2$ as two 16x16 gray scale images. Hint: you may need to scale the values to be in the valid range of grayscale ([0, 255] or [0,1] depending on which function you use). You can shift and scale them in order to show a better picture. It is best if you can show an accompany 'colorbar' that maps gray scale to values.

(e) Now we project (the centered) $X$ down to the two PCA directions. Let $V = [v_1, v_2]$ be the $d \times 2$ matrix. The projection is simply $XV$. (To be precise, these are the coefficients along the principal directions, not the projection itself.) Show the resulting two coordinates for the first line in three.txt and the first line in eight.txt, respectively.

(f) Report the average reconstruction error $\frac{1}{n}\sum_{i=1}^{n} \|x_i V V^\top - x_i\|^2$, where $x_i \in \mathbb{R}^{1 \times d}$ is the $i$-th row of the centered data matrix $X$.

(g) Now plot the 2D point cloud of the 400 digits after projection. For visual interest, color points in three.txt red and points in eight.txt blue. But keep in mind that PCA is an unsupervised learning method and it does not know such class labels.