Question 5
PCA for Dimensionality Reduction

Atishay Jain
Gohil Megh Hiteshkumar

October 15, 2022

1 Idea

For given question we can say that,
for a given digit each pixel is a random variable and each instance of those images corresponding to that digit are the values drawn from the random variable in 784 dimensional space.
That is there are total of 784 random variables for a digit. So, to reduce the dimensions

- first step is to calculate mean and covariance matrix for each digit.
- from covariance matrix get eigenvectors and eigenvalues.
- arrange eigenvalues in descending order and filter first 84 eigenvalues and then filter eigenvectors corresponding to those eigenvectors.
- now get new coordinates from these 84 eigenvectors and use them to reconstruct the image.

2 Implementation

2.1 Initial steps

- Convert each entry(int data type) of image matrix to floating-point data type.
- Convert the image matrix(28×28) to image vector(784×1)
- Calculate mean(784×1) of each digit.

$$\mu_n = \frac{\sum_{i=1}^{N} I_i}{N} \tag{1}$$

  where N is total number of instances of that digit n.

- Calculate covariance matrix of each digit.

$$C_n(i,j) = \frac{(X_i - \mu'_{ni})(X_j - \mu'_{nj})^T}{N} \tag{2}$$

  where, X_i and X_j are the vectors of size $1 \times N$ and μ'_{ni} and μ'_{nj} are also vectors of same size with each entry of these vectors equal to i^{th} and j^{th} value of μ_n of digit n. We can say that X_i is a RV representing the $(i\%28, \lfloor i/28 \rfloor)$ pixel.

2.2 To compute 84 coordinates from 784 coordinates

- C_n is a real valued 784×784 symmetric matrix. According to spectral theorem C_n has 784 real eigenvalues and 784 real-valued eigenvectors(784×1).

- using eigs function in matlab, get all those eigenvectors and corresponding eigenvalues.

- let $v_1, v_2, v_3, ..., v_{784}$ be the eigenvectors of C_n and $\lambda_1, \lambda_2, \lambda_3, ..., \lambda_{784}$ be the corresponding eigenvalues arranged in descending order using eigs function of matlab. Eigenvalues are arranged in descending order to get maximum total dispersion of the original data (for the chosen digit) within the 84 dimensional hyperplane

- Suppose $I = x_1, x_2, x_3, ..., x_{784}$ be the image row vector then $c_new = c_1, c_2, c_3, ..., c_{84}$ be the new coordinates of image vector such that,
  $c_1 = I \times v_1$
  $c_2 = I \times v2$
  .
  .
  .
  $c_{84} = I \times v_{84}$

2.3 To regenerate image using those 84 coordinates

- To get the image we have to multiply each of the 84 coordinates to corresponding 84 eigenvectors.

- Let our new image be I_new, the value of I_{new} can be obtained by following way,

$$I_{new} = \sum_{i=1}^{84} c_i.v_i \tag{3}$$

  where, I is our original image and . here represent multiplication of scalar to a matrix.

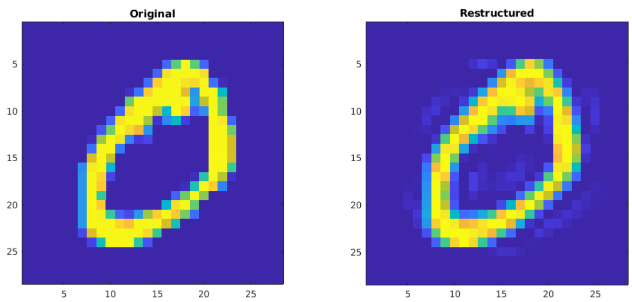

Figure 1: Images for digit 0

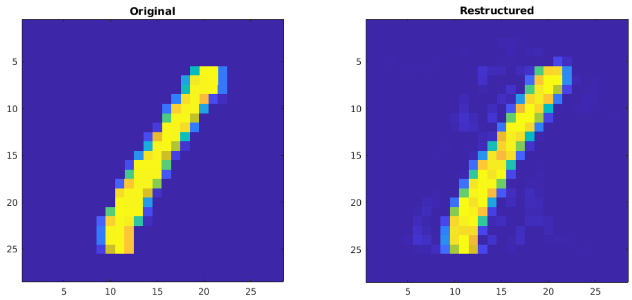

Figure 2: Images for digit 1

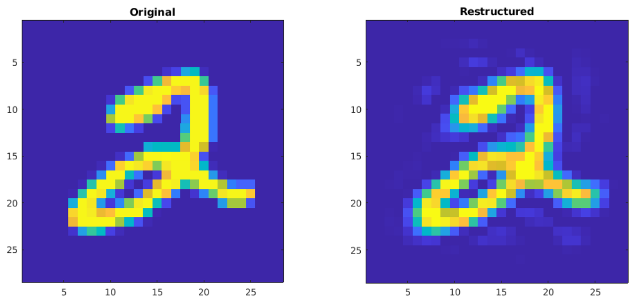

Figure 3: Images for digit 2

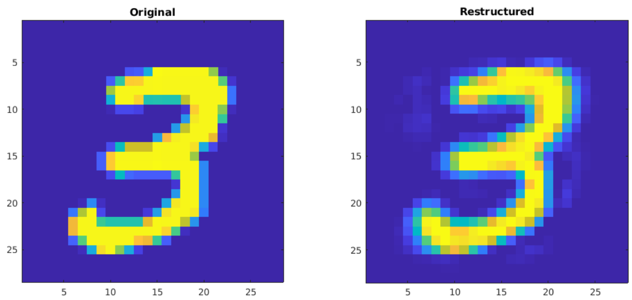

Figure 4: Images for digit 3

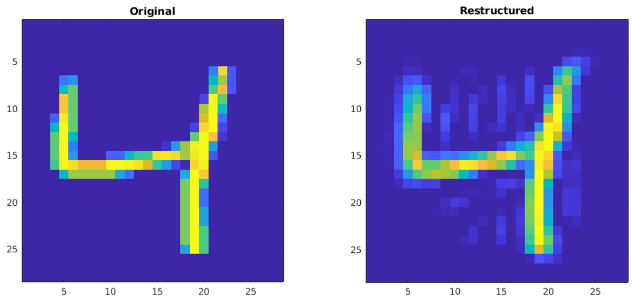

Figure 5: Images for digit 4

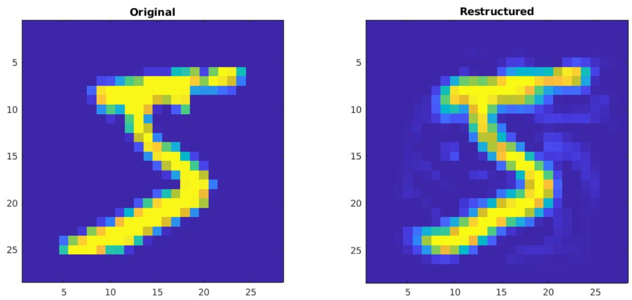

Figure 6: Images for digit 5

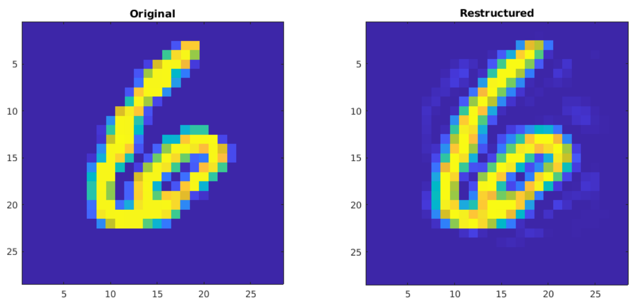

Figure 7: Images for digit 6

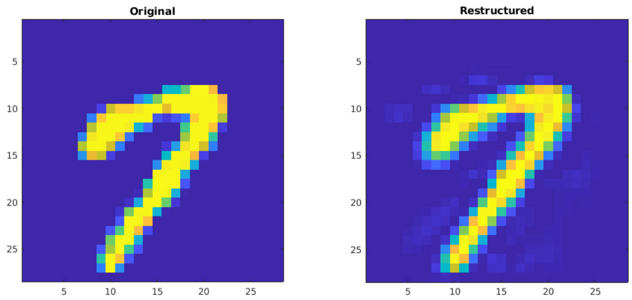

Figure 8: Images for digit 7

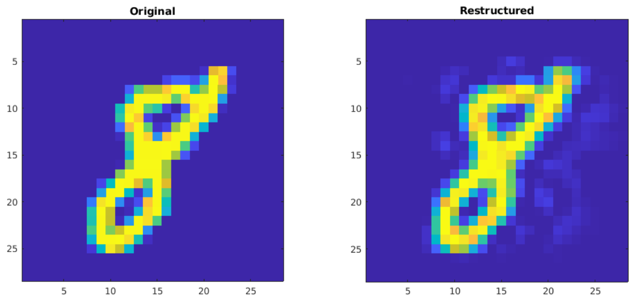

Figure 9: Images for digit 8

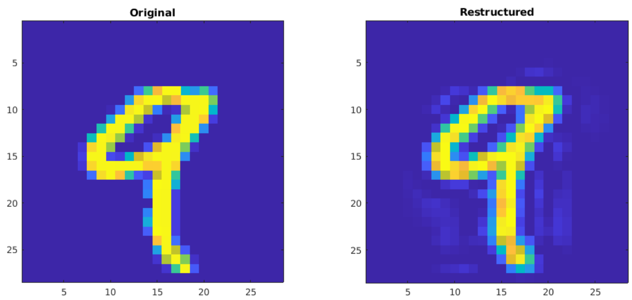

Figure 10: Images for digit 9