ECE 283: Homework 3

Due: May 19, 2025 (Mon)

Your Name:

Collaborators:

Instructions: Since this homework doesn't have any math question, you will just submit your Jupyter notebook.

1 Principal Component Analysis

1) 2-D Synthetic Data Generation. We will generate the same two-dimensional synthetic Gaussian mixture data as in Homework 2, but just for Class 1:

- Component C: $\pi_C = \frac{3}{4}$, $\boldsymbol{\mu}_C = (0,0)^T$. $\mathbf{C}_C$ with eigenvalue, eigenvector pairs $\lambda_1 = 1$, $\mathbf{u}_1 = (\cos\theta, \sin\theta)^T$, $\lambda_2 = 2$, $\mathbf{u}_2 = (-\sin\theta, \cos\theta)^T$ with $\theta = \frac{\pi}{3}$.
- Component D: $\pi_D = \frac{1}{4}$, $\boldsymbol{\mu}_D = (-6,-4)^T$. $\mathbf{C}_D$ has eigenvalue, eigenvector pairs: $\lambda_1 = 2$, $\mathbf{u}_1 = (\cos\theta, \sin\theta)^T$, $\lambda_2 = 1$, $\mathbf{u}_2 = (-\sin\theta, \cos\theta)^T$ with $\theta = \frac{\pi}{4}$.

2) PCA. (10pts) Generate 200 samples from the above Gaussian mixture distribution. Write a function that finds all d principal components for a d-dimensional dataset. Using this function, find the first and second principal component of the synthetic data. Plot the first and second principal component vectors along with the scatter plot of the data points. (Note: You will have to center the data for PCA and then shift it back to the original mean for plotting.)

3) PCA on the Olivetti faces dataset. First, we will load the Olivetti faces dataset and follow the data preparation steps given here. Run your PCA function on the preprocessed face data.

(a) (2pts) In this case, we will first have to flatten the face images to vectors. If we reshape the principal components back to the original image shapes, we call those *eigenfaces*. Obtain the first and second principal components $\mathbf{v}_1$ and $\mathbf{v}_2$ and plot the eigenfaces.

(b) (2pts) Find the projection of the original images onto $\mathbf{v}_1$ and $\mathbf{v}_2$, i.e., $z_1 = \mathbf{x}^T\mathbf{v}_1$ and $z_2 = \mathbf{x}^T\mathbf{v}_2$. Scatter plot (z_1, z_2) onto the 2D space.

(c) (2pts) Reconstruct the faces with the first two principal components. Choose three random images and show the original and the reconstructed images side-by-side.

(d) (2pts) Now, reconstruct the faces with the first 10, 20, and 200 principal components and repeat the same side-by-side plots.

(e) (2pts) Gradually vary the number of components we use for reconstruction from 1 to 4096. Compute RMSE for reconstruction. Plot RMSE vs. the number of components. Discuss what you observe.

2 XORNet

1) XORNet with predefined weights. First, we will design a 2-layer neural network that takes two inputs x_1 and x_2 and outputs x_1 XOR x_2.

(a) (2pts) Draw your neural network architecture and specify weights. Explain your design.

(b) (2pts) Implement the neural network you designed with PyTorch. Initialize the weights with the weights you chose in (a). Print out the results for inputs $(0,0),(0,1),(1,0),(1,1)$.

2) Training XORNet. Now, instead of specifying weights, we want to learn weights of XORNet.

(a) (4pts) First, construct a training dataset we can use for this purpose. Choose a loss function, and train with backpropagation until you have small loss. Print the weights you learned and the neural network results for inputs $(0,0),(0,1),(1,0),(1,1)$.

(b) (2pts) Compare the weights and results from 1) and 2). Discuss your thoughts.