24-787 Artificial Intelligence and Machine Learning for Engineering Design
Homework 4
Principal Component Analysis

Visualizing Principal Components

1. *(25 points)*

Suppose you are given the following two-dimensional dataset:

$$X = \begin{bmatrix} 2.5 & 2.4 \\ 0.5 & 0.7 \\ 2.2 & 2.9 \\ 1.9 & 2.2 \\ 3.1 & 3.0 \\ 2.3 & 2.7 \\ 2.0 & 1.6 \\ 1.0 & 1.1 \\ 1.5 & 1.6 \\ 1.1 & 0.9 \end{bmatrix}$$

(a) Compute the first and second principal components (e_1 and e_2) of this dataset. Show all your work. Plot the data in the original dimension space and show the principal components.

(b) Transform the data using both principal components (*i.e.* compute a_1 and a_2 for each data point) and plot this new representation.

(c) What is the PCA-optimal one-dimensional representation of the data? In this reduced dimension, what is the range of the data (the distance between the minimum and maximum points)?

When the Sample Size is less than the Dimensionality

2. *(35 points)*

Now, suppose your data is as follows:

$$X = \begin{bmatrix} -2 & 1 & 2 & -3 & 4 & 1 & 0 & 3 & 0 & 2 & 1 & 1 & 2 & 3 & -2 & -3 & 2 & 1 & 0 \\ 1 & 2 & -4 & 2 & -4 & 2 & 5 & 2 & 2 & 1 & -3 & 0 & 0 & 1 & -2 & 1 & 1 & -3 & -2 \\ 1 & -3 & 2 & 1 & 0 & -3 & -5 & -1 & 3 & 3 & -2 & -3 & -2 & -1 & 1 & 0 & 5 & 4 & 2 \\ 3 & -1 & 0 & 2 & 2 & -5 & -4 & -1 & 2 & -1 & 3 & 4 & 4 & 2 & 1 & 2 & -2 & 1 & -1 \end{bmatrix}$$

In this case, the number of samples (4) is much less than the number of dimensions (19), so you will need to use an approach similar to the face recognition problem discussed in lecture.

(a) Compute the 4x4 inner product matrix and determine the minimum squared error representations of these samples in three-dimensional space. Show all steps in your derivation.

(b) Calculate the weights needed to regenerate the second data sample using the three eigenvectors from part (a)?

(c) Determine the mean-squared error between each sample and its reconstructed version?

(d) Repeat part (c), but this time assume the dimensionality of the data is reduced to two instead of three.

(e) What is the Euclidean distance between the new data vector given below and each of the four samples in the reduced three-dimensional space? Which of the four samples is most similar to this new vector?

$$Y = \begin{bmatrix} 1 & 3 & 0 & 3 & -2 & 2 & 4 & 1 & 3 & 0 & -2 & 0 & 1 & 1 & -3 & 0 & 1 & -2 & -3 \end{bmatrix}$$

(f) Perform the same analysis as part (e) (*i.e.* nearest-neighbor classifier), but this time in the original dimension space. Do the results match? Does this make intuitive sense? Why or why not?

Hint: Make sure your eigenvectors are always unit vectors.

Face Recognition

3. *(40 points)*

In this programming exercise, you will be developing a simple classifier to recognize human faces. The image dataset (Yale face database) is provided with the assignment. It consists of 165 grayscale images (15 subjects, 11 images each), each of size 231x195. The provided MATLAB variables include:

`X`	The data, encoded as a 45045x165 matrix, in which the rows represent dimensions (pixels) and the columns indicate examples. To view an image, simply reshape the appropriate column vector into a 231x195 array. For example, the following line of code displays the 142nd image in the dataset: `imshow(reshape(data(:,142),231,195))`
`Y`	The labels, given as a 165x1 vector. Each label takes an integer value from 1-15, indicating which subject is shown in the corresponding image.
`testimages`	A vector of indices indicating which images to withhold for testing. For example, if the number 10 is included in this vector, do not use the 10th image when computing principal components.
`trainimages`	The complement of `testimages`, provided for completeness.

(a) Reduce the dimensionality of the data using PCA such that 90% of the variance is preserved. What is the new number of dimensions? In your report, provide visualizations of the first five eigenfaces (those with the highest eigenvalues) and include a bar graph showing the percentage of variance explained by each principal component in descending order.

(b) Try reconstructing the first training example with varying number of eigenfaces. In your report, juxtapose the original image with its reconstructed versions using [10, 20, 30, 40, 50] eigenfaces.

(c) Implement the nearest neighbor algorithm in the reduced dimension space to classify each of the test images. Report the overall test accuracy.

(d) Repeat part (c), but in the original dimension space. Does the accuracy change? Is this the result you would expect? Why or why not?

Extra credit: *(10 points max)* Use Linear Discriminant Analysis (LDA) to classify the test images. Compare your accuracy to the result in part (c).