CS613 Machine Learning - HW 1

McWelling Todman

Due: February 1^{st}, 2019

Theory Questions

1. Theory - Consider the following data:

$$\begin{bmatrix} -2 & 1 \\ -5 & -4 \\ -3 & 1 \\ 0 & 3 \\ -8 & 11 \\ -2 & 5 \\ 1 & 0 \\ 5 & -1 \\ -1 & -3 \\ 6 & 1 \end{bmatrix}$$

(a) Find the principle components of the data (you must show the math, including how you compute the eivenvectors and eigenvalues). Make sure you standardize the data first and that your principle components are normalized to be unit length. As for the amount of detail needed in your work imagine that you were working on paper with a basic calculator. Show me whatever you would be writing on that paper. (5pts).

The first step is to standardize our data so that the dimensions are centered around zero with a standard deviation of 1. In order to do this we calculate the mean and standard deviation of each measure. For convenience, we will refer to our matrix as A and call the left column of the matrix x (A_x) and the right column y (A_y).

$$\mu_x\text{: } (-2+(-5)+(-3)+0+(-8)+(-2)+1+5+(-1)+6)/10 = .9$$
$$\mu_y\text{: } (1+(-4)+1+3+11+5+0+(-1)+(-3)+1)/10 = 1.4$$
$$\sigma_x\text{: } \sqrt{\frac{\Sigma(X_i-\mu_x)}{10}} = 4.2282 \quad \sigma_y\text{: } \sqrt{\frac{\Sigma(y_i-\mu_y)}{10}} = 4.2740$$

Having computed our parameters, we are now able to standardize the data. For each element of each component feature of A (x,y) subtract the mean (μ) of the respective component feature, then divide by the standard deviation (σ) of the respective component feature.

$$\text{Standardization } A_x\text{: for all } x_i \in A_x \longrightarrow \frac{x_i-\mu_x}{\sigma_x}$$
$$\text{Standardization } A_y\text{: for all } y_i \in A_y \longrightarrow \frac{y_i-\mu_y}{\sigma_y}$$

Our resulting standardized matrix takes the form:

$$\begin{bmatrix} -0.2602 & -0.0936 \\ -0.9697 & -1.2635 \\ -0.4967 & -0.0936 \\ 0.2129 & 0.3744 \\ -1.6792 & 2.2462 \\ -0.2602 & 0.8423 \\ 0.4494 & -0.3276 \\ 1.3954 & -0.5615 \\ -0.0237 & -1.0295 \\ 1.6319 & -0.0936 \end{bmatrix}$$

With the standardization process complete, we are ready to compute our covariance matrix. Before computing our matrix, there are several things we should note:

i. We are working with 2-dimensional data, so our matrix will be 2x2.
ii. Our data is standardized, meaning variance should be equal to 1.
iii. Our covariances should be identical, as Cov(x,y) == Cov(y,x).

$$Variance(x) = \frac{\Sigma(x_i - \mu_x)^2}{n-1}$$
$$Variance(y) = \frac{\Sigma(y_i - \mu_y)^2}{n-1}$$
$$Covariance(x,y) = \frac{\Sigma(x_i - \mu_x)(y_i - \mu_y)}{n-1} = Covariance(y,x)$$

Out resulting matrix is of the form:

$$\begin{bmatrix} Var(x) & Cov(x,y) \\ Cov(y,x) & Var(y) \end{bmatrix}$$

Which corresponds to the following:

$$\begin{bmatrix} 1.0 & -0.4083 \\ -0.4083 & 1.0 \end{bmatrix}$$

Having constructed our covariance matrix C, our next objective is to find the eigenvalues through eigendecomposition. We must solve:

$$Determinant(C) = |A - \lambda I| = \begin{bmatrix} 1.0 - \lambda & -0.4083 \\ -0.4083 & 1.0 - \lambda \end{bmatrix}$$

$$= (1 - \lambda)(1 - \lambda) + (-0.4083)^2$$

$$0 = \lambda^2 - 2\lambda + 0.83329111$$

By the quadratic formula our eigenvalues are:

$$\frac{-b +/- \sqrt{b^2 - 4ac}}{2a} \text{ where } a = 1, b = -2, c = 0.83329111$$

$$\lambda_1 : \frac{2 + \sqrt{4 - 4(0.83329111)}}{2} = \frac{2 + \sqrt{0.66683556}}{2} = 1.4083$$

$$\lambda_2 : \frac{2 - \sqrt{4 - 4(0.83329111)}}{2} = \frac{2 - \sqrt{0.66683556}}{2} = 0.5917$$

Plugging our eigenvalues back in, we are able to derive the eigenvectors:

$$\lambda_1: \quad \begin{bmatrix} 1.0 - 1.4083 & -0.4083 \\ -0.4083 & 1.0 - 1.4083 \end{bmatrix} \longrightarrow \begin{bmatrix} -0.4083 & -0.4083 \\ -0.4083 & -0.4083 \end{bmatrix}$$

$$\text{Solve:} \quad \begin{bmatrix} -0.4083 & -0.4083 \\ -0.4083 & -0.4083 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

$$\text{Normalized Vector} = \frac{V}{|V|}: \begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} -0.7071 \\ 0.7071 \end{bmatrix}$$

$$\lambda_2: \quad \begin{bmatrix} 1.0 - 0.5917 & -0.4083 \\ -0.4083 & 1.0 - 0.5917 \end{bmatrix} \longrightarrow \begin{bmatrix} 0.4083 & -0.4083 \\ -0.4083 & 0.4083 \end{bmatrix}$$

$$\text{Solve:} \quad \begin{bmatrix} 0.4083 & -0.4083 \\ -0.4083 & 0.4083 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\text{Normalized Vector} = \frac{V}{|V|}: \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 0.7071 \\ 0.7071 \end{bmatrix}$$

(b) Project the data onto the principal component corresponding to the largest eigenvalue found in the previous part (3pts).

$$\text{Solve:} \begin{bmatrix} -0.2602 & -0.0936 \\ -0.9697 & -1.2635 \\ -0.4967 & -0.0936 \\ 0.2129 & 0.3744 \\ -1.6792 & 2.2462 \\ -0.2602 & 0.8423 \\ 0.4494 & -0.3276 \\ 1.3954 & -0.5615 \\ -0.0237 & -1.0295 \\ 1.6319 & -0.0936 \end{bmatrix} \begin{bmatrix} -0.7071 \\ 0.7071 \end{bmatrix} = \text{Projection}$$

$$\text{Projection} = \begin{bmatrix} 0.1178 \\ -0.2077 \\ 0.2850 \\ 0.1142 \\ 2.7756 \\ 0.7796 \\ -0.5494 \\ -1.3838 \\ -0.7112 \\ -1.2201 \end{bmatrix}$$

2. Theory - Consider the following data:

$$\text{Class 1} = \begin{bmatrix} -2 & 1 \\ -5 & -4 \\ -3 & 1 \\ 0 & 3 \\ -8 & 11 \end{bmatrix}, \text{Class 2} = \begin{bmatrix} -2 & 5 \\ 1 & 0 \\ 5 & -1 \\ -1 & -3 \\ 6 & 1 \end{bmatrix}$$

(a) Compute the information gain for each feature. You could standardize the data overall, although it won't make a difference. (5pts).

Feature 1		
P (Class 1)	N (Class 2)	H (Entropy)
$P_{-8} = 1$	$N_{-8} = 0$	0
$P_{-5} = 1$	$N_{-5} = 0$	0
$P_{-3} = 1$	$N_{-3} = 0$	0
$P_{-2} = 1$	$N_{-2} = 1$	1
$P_{-1} = 0$	$N_{-1} = 1$	0
$P_0 = 1$	$N_0 = 0$	0
$P_1 = 0$	$N_1 = 1$	0
$P_5 = 0$	$N_5 = 1$	0
$P_6 = 0$	$N_6 = 1$	0

Feature 2		
P (Class 1)	N (Class 2)	H (Entropy)
$P_{-4} = 1$	$N_{-4} = 0$	0
$P_{-3} = 0$	$N_{-3} = 1$	0
$P_{-1} = 0$	$N_{-1} = 1$	0
$P_0 = 0$	$N_0 = 1$	0
$P_1 = 2$	$N_1 = 1$	$\frac{-2}{3}\log_2(\frac{2}{3}) - \frac{1}{3}\log(\frac{1}{3})$
$P_3 = 1$	$N_3 = 0$	0
$P_5 = 0$	$N_5 = 1$	0
$P_{11} = 1$	$N_{11} = 0$	0

$$IG_1 = \tfrac{1+1}{10}(1) = .2$$

$$IG_2 = \tfrac{2+1}{10}(\tfrac{-2}{3}\log_2(\tfrac{2}{3}) - \tfrac{1}{3}\log(\tfrac{1}{3}) = 0.275$$

(b) Which feature is more discriminating based on results in part a (1pt)?

The information gain (IG) from feature 2 is greater, therefore feature 2 is more discriminating.

(c) Using LDA, find the direction of projection (you must show the math, however for this one you don't have to show the computation for finding the eigenvalues and eigenvectors). Normalize this vector to be unit length (5pts).

The first step is to combine and standardize the data:

$$\text{Class 1} = \begin{bmatrix} -2 & 1 \\ -5 & -4 \\ -3 & 1 \\ 0 & 3 \\ -8 & 11 \end{bmatrix}, \text{Class 2} = \begin{bmatrix} -2 & 5 \\ 1 & 0 \\ 5 & -1 \\ -1 & -3 \\ 6 & 1 \end{bmatrix} \longrightarrow \begin{bmatrix} -0.2602 & -0.0936 \\ -0.9697 & -1.2635 \\ -0.4967 & -0.0936 \\ 0.2129 & 0.3744 \\ -1.6792 & 2.2462 \\ -0.2602 & 0.8423 \\ 0.4494 & -0.3276 \\ 1.3954 & -0.5615 \\ -0.0237 & -1.0295 \\ 1.6319 & -0.0936 \end{bmatrix}$$

Next, we must compute the means for each class:

$$\text{Class 1: } \begin{bmatrix} -0.2602 & -0.0936 \\ -0.9697 & -1.2635 \\ -0.4967 & -0.0936 \\ 0.2129 & 0.3744 \\ -1.6792 & 2.2462 \end{bmatrix} \Longrightarrow \mu_1 = \begin{bmatrix} -0.6386 & 0.2340 \end{bmatrix}$$

$$\text{Class 2: } \begin{bmatrix} -0.2602 & 0.8423 \\ 0.4494 & -0.3276 \\ 1.3954 & -0.5615 \\ -0.0237 & -1.0295 \\ 1.6319 & -0.0936 \end{bmatrix} \longrightarrow \mu_2 = \begin{bmatrix} 0.6386 & -0.2340 \end{bmatrix}$$

Followed by the scatter matrices for each class:

$$\text{Class 1: } \begin{bmatrix} -0.2602 & -0.0936 \\ -0.9697 & -1.2635 \\ -0.4967 & -0.0936 \\ 0.2129 & 0.3744 \\ -1.6792 & 2.2462 \end{bmatrix} \Longrightarrow \sigma_1^2 = (|C_1| - 1) * Cov(Class_1) = \begin{bmatrix} 2.0808 & -1.6490 \\ -1.6490 & 6.5255 \end{bmatrix}$$

$$\text{Class 2: } \begin{bmatrix} -0.2602 & 0.8423 \\ 0.4494 & -0.3276 \\ 1.3954 & -0.5615 \\ -0.0237 & -1.0295 \\ 1.6319 & -0.0936 \end{bmatrix} \longrightarrow \sigma_2^2 = (|C_2| - 1) * Cov(Class_2) = \begin{bmatrix} 2.8415 & -0.5312 \\ -0.5312 & 1.9270 \end{bmatrix}$$

From the scatter matrices for each class, we are able to derive the within class scatter matrix S_W^{-1}:

$$S_W = \sigma_1^2 + \sigma_2^2 = \begin{bmatrix} 2.0808 & -1.6490 \\ -1.6490 & 6.5255 \end{bmatrix} + \begin{bmatrix} 2.8415 & -0.5312 \\ -0.5312 & 1.9270 \end{bmatrix} = \begin{bmatrix} 4.9223 & -2.1803 \\ -2.1803 & 8.4526 \end{bmatrix}$$

$$S_W^{-1} = \begin{bmatrix} 0.2294 & 0.0592 \\ 0.0592 & 0.1336 \end{bmatrix}$$

Finally, returning to the vectors containing the means of each class, we take the outer product to obtain our between class scatter matrix S_B:

$$S_B = (\mu_1 - \mu_2) \otimes (\mu_1 - \mu_2)^T = \begin{bmatrix} 1.6311 & -0.5976 \\ -0.5976 & 0.2190 \end{bmatrix}$$

Now that we have S_W^{-1} and S_B, we perform perform eigen decomposition on the product of these two matrices to find the eigenvector associated with the only non-zero eigenvalue:

$$S_W^{-1} S_B = \begin{bmatrix} 4.9223 & -2.1803 \\ -2.1803 & 8.4526 \end{bmatrix} \begin{bmatrix} 1.6311 & -0.5976 \\ -0.5976 & 0.2190 \end{bmatrix} = \begin{bmatrix} 0.3387 & -0.1241 \\ 0.0167 & -0.0061 \end{bmatrix}$$

$$Eig(S_W^{-1} S_B) : \lambda = 0.3326 \longrightarrow \begin{bmatrix} 0.9988 \\ 0.0492 \end{bmatrix}$$

(d) Project the data onto the principal component found in the previous part (3pts)

$$\begin{bmatrix} -0.2602 & -0.0936 \\ -0.9697 & -1.2635 \\ -0.4967 & -0.0936 \\ 0.2129 & 0.3744 \\ -1.6792 & 2.2462 \\ -0.2602 & 0.8423 \\ 0.4494 & -0.3276 \\ 1.3954 & -0.5615 \\ -0.0237 & -1.0295 \\ 1.6319 & -0.0936 \end{bmatrix} \begin{bmatrix} 0.9988 \\ 0.0492 \end{bmatrix} = \begin{bmatrix} -0.2644 \\ -1.0306 \\ -0.5007 \\ 0.2310 \\ -1.5667 \\ -0.2184 \\ 0.4327 \\ 1.3661 \\ -0.0742 \\ 1.6253 \end{bmatrix}$$

(e) Does the projection you performed in the previous part seem to provide good class separation? Why or why not (1pt)?

It's not great. The classes aren't particularly distinct. Perhaps the phenomena we are observing isn't quite as cut and dry as we initially imagined, consider we are unable to cleanly discriminate after and performing Fisher LDA. Perhaps if we had more features and/or observations it would be easier to distinguish through LDA.

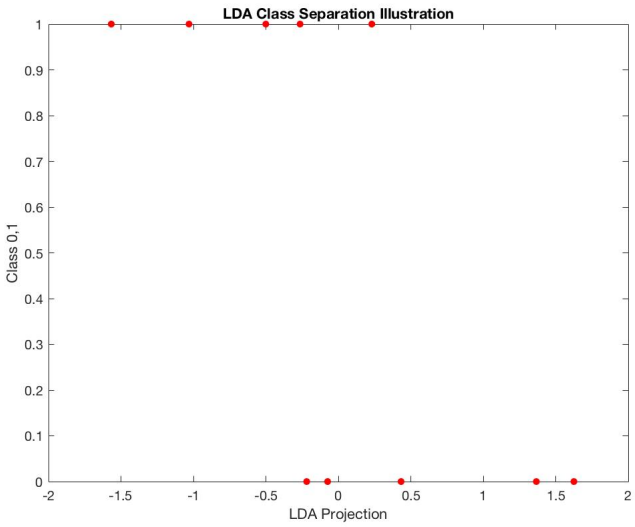

Figure 1: LDA Projection

1. Part 2: The visualization of the PCA result

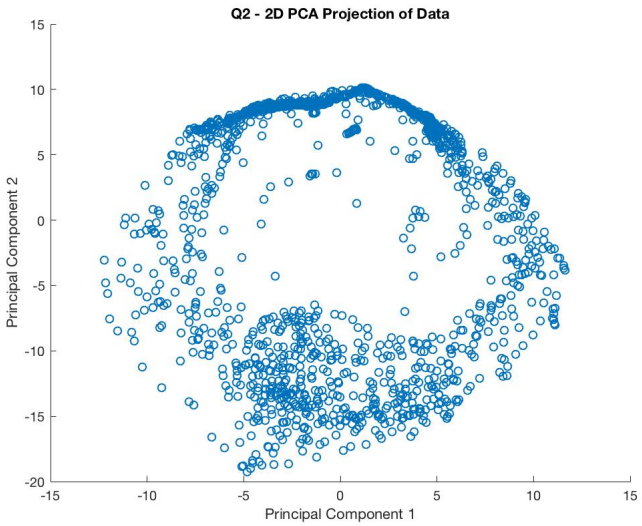

Figure 2: Initial Clustering

2. Part 3:

   (a) Number of principle components needed to represent 95% of information, k.

      i. 19 principal components needed.

   (b) Visualization of primary principle component

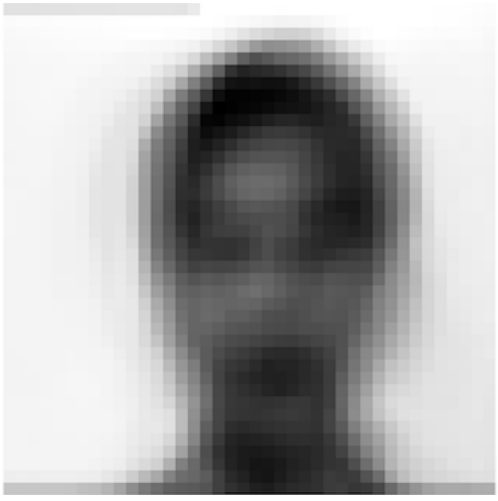

Figure 3: Original image

   (c) Visualization of the reconstruction of the first person using

      i. Original image

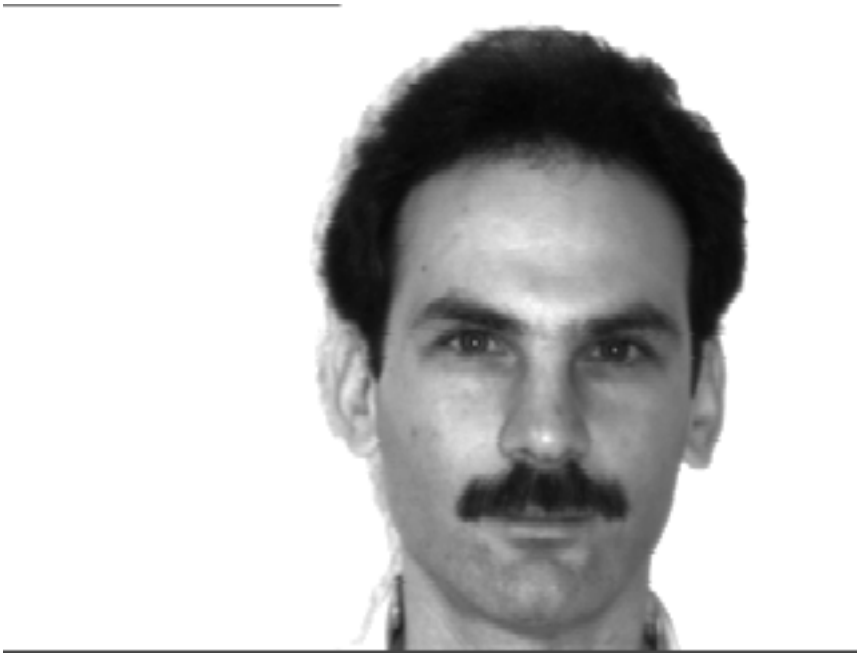

Figure 4: Original image

ii. Single principle component

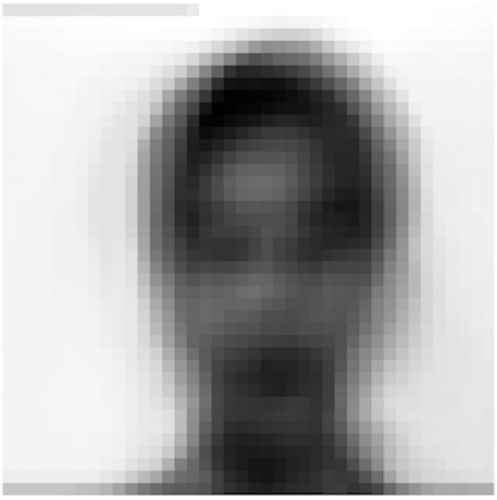

Figure 5: Reconstruction via Primary Principal Component

iii. k principle components.

Figure 6: Reconstruction via k (k=19) Principal Components - 95% retention

3. Part 4: The visualization of at least one k-means clustering process including:

   (a) The initial setup visualization

   No idea what I am supposed to do here. I would have asked if there were more time. Sorry about that.

   (b) The initial cluster assignment visualization

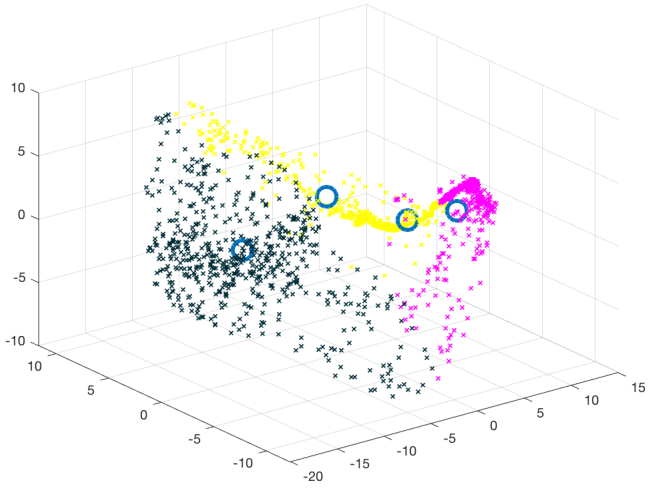

Figure 7: kMeans visualization - initial clustering - k = 4

(c) The final cluster assignment visualization

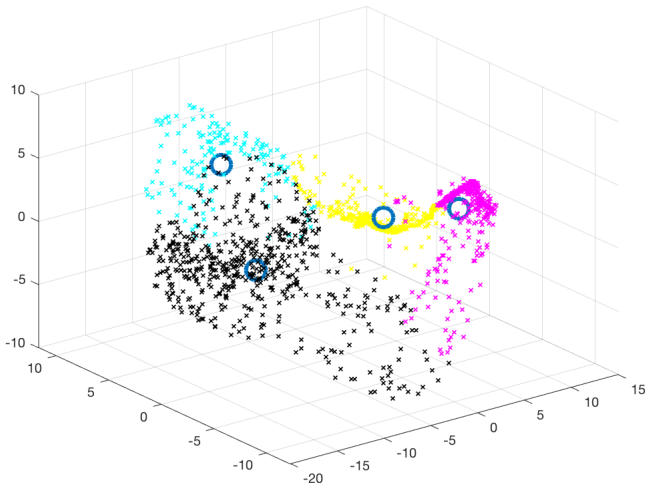

Figure 8: kMeans visualization - final clustering - k = 4

Source Code - Including any necessary makefiles, etc..

readme.txt file - The readme.txt file should contain information on how to run your code to reproduce your results.

Sample videos of at least three different runs where you vary k. Their filenames should be in the format

$$K\_[k]\_....[ext]$$

where:

(a) [k] is the value of k passed to your function.

(b) [ext] is the file extension.

For example, if we created an AVI video with $k = 2$ the file name would be

$$K\_2.avi$$

Do not include spaces or special characters (other than the underscore character) in your file and directory names. Doing so may break our grading scripts.