Homework 1

(Total 200 pts)

Due 5:00 pm on September 9, 2022 (Friday)

Canvas submission of your answers (with required plots and source codes attached) in a single PDF file ('hw1.pdf'), and then submit the following source code files:
Q2.m, Q3.py, Q4.m, Q5.m

1. (30 pts) We want to design a classifier to separate two pattern classes. Each pattern is a two-dimensional vector $\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$. Assume that the patterns belonging to each of the two classes are equally likely and the patterns of each class are samples from a Gaussian distribution. The mean vectors and covariance matrices of each of the two classes are $\boldsymbol{m_1} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$, $\boldsymbol{C_1} = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix}$, and $\boldsymbol{m_2} = \begin{bmatrix} 8 \\ 8 \end{bmatrix}$, $\boldsymbol{C_2} = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix}$, respectively. Determine analytically the equation for the decision boundary for a Bayes Classifier. Show your detailed derivations. Simply the equation as much as possible. Fill in the blank with your answer:
   The equation is $x_1 = (\quad\quad)x_2 + (\quad\quad)$.

2. (50 pts) A simulated dataset was generated according to the distributions specified in Q1 above, where each of two classes has 500 samples. The dataset file can be downloaded from the link below. This csv file has three columns. The first two column are values for x_1 and x_2, respectively. The third column contains the corresponding class label values.
   http://www.ece.uah.edu/~dwpan/course/ee610/hw/dataset_hw.csv
   (a) Read in the csv file to a matrix, using the *readmatrix* function in Matlab. From the matrix extract the values for x_1 and x_2, and the values for the class labels.
   Plot the dataset using
   >> gscatter (x1, x2, label, 'rb', 'ox')
   Add legends to the scatter plot above to indicate the class index for the samples. Specify the legend location to be 'northwest'.
   (b) Plot the decision boundary you obtained in Q1, in green color, on the same scatter plot for the datasets.
   (c) Calculate the accuracy of a classifier based on the decision boundary, where the accuracy is defined as the ratio of the number of correctly classified samples / total number of data samples. Tie-breaking rule: assign samples on the decision boundary (if any) to the class with label being 1.
   (d) Now train a Naïve Bayes classifier using the *fitcnb* function, and calculate its accuracy.
   (e) Compare the accuracies in (c) and (d), and comments on the results.
   (f) Attach the plot and the Matlab script you used.

3. (40 pts) Naïve Bayes Classifier in sklearn.
   (a) Use the numpy.loadtxt function to load in the file 'dataset_hw.csv' used in Q2. Note: the string for the full path for the infile might need a 'r' prefix to indicate the string is raw and those backslashes (/) in the string should not be treated as escapes. See https://docs.python.org/3/tutorial/introduction.html#strings.
   (b) Use *GaussianNB* in sklearn to train a Naïve Bayes classifier.

(c) What is the score of the classifier? Compare it with the accuracy in Q2 (d).
(d) Attach and upload the Python code you used 'Q3.py'.

4. (40 pts) Decision boundary based on estimated mean vector and covariance matrix.
   (a) Again, we use the dataset in Q2, by reading into Matlab the 'dataset_hw.csv' file.
   (b) Use the *mean* and *cov* functions to estimate the mean vector and covariance matrix of the samples belonging to each of two classes. Fill in the table below with your answers:

Class	Mean Vector	covariance matrix
1	$\boldsymbol{m_1} = [\quad\quad , \quad\quad]$	$\boldsymbol{C_1} = \begin{bmatrix} & \\ & \end{bmatrix}$
2	$\boldsymbol{m_2} = [\quad\quad , \quad\quad]$	$\boldsymbol{C_2} = \begin{bmatrix} & \\ & \end{bmatrix}$

   (c) Similar to Q1, determine analytically the equation for the decision boundary; however, this time we use the estimated mean vectors and covariance matrices found in (a). Show your derivations. Simply the equation as much as possible. Fill in the blank with your answer:
   The equation is $x_1 = (\quad\quad)x_2 + (\quad\quad)$.
   (d) Plot the decision boundary (in the dark color) on top of the scatter plot of the dataset, similar to Q2 (a) and (b). Use
   >> axis equal
   >> axis square
   (e) Attach the plot and the Matlab script you used.

5. (40 pts) Mahalanobis Distance.
   (a) Use the mean vectors and covariance matrices found in Q4 (b) to calculate the following distances and fill in the blanks with your answer.

Euclidean distance between $\boldsymbol{m_1}$ and $\boldsymbol{m_2}$	$D_E(\boldsymbol{m_1}, \boldsymbol{m_2}) =$
Mahalanobis distance from $\boldsymbol{m_1}$ to $\boldsymbol{m_2}$, calculated by using the formula $D_M = \sqrt{(\boldsymbol{m_1} - \boldsymbol{m_2})\boldsymbol{C_2^{-1}}(\boldsymbol{m_1} - \boldsymbol{m_2})^{\mathrm{T}}}$	$D_M =$
Mahalanobis distance from $\boldsymbol{m_1}$ to $\boldsymbol{m_2}$, by using the *mahal* function in Matlab.	$D_M =$

   Did you get the same results for the Mahalanobis distance? Comment on your results.
   (b) In the same graph, generate contour plots of isolines of Mahalanobis distances (calculated by using the *mahal* function) from $\boldsymbol{m_1}$ and $\boldsymbol{m_2}$, respectively. Superimpose the contour plots on the scatter plot (with the decision boundary) generated in Q4 (d). Use the following meshgrid for the contour plots:
   x1 = linspace(-2,14); x2 = linspace(-2,14); [X1,X2] = meshgrid(x1,x2);
   Turn on 'ShowText' for the contour plots.
   Comment on the relation between the isolines and the decision boundary.
   (c) Attach the plot and the Matlab script you used.