Assignment 1 – Calvin Tran 20826392

Exercise 1: Equiprobability Contour

1.

Refer to Exercise_1_1.py for the code used to generate the function.

The samples for the class 1 are shown below:
[3.86317368 6.98028738]
[1.81099457 6.01214467]
[3.68588236 3.72158188]
[8.84837398 5.98633342]
[3.24418499 1.70899736]

The samples for class 2 are shown below:
[5.6974558 18.28342712]
[10.24735416 12.71358796]
[3.7893197 10.94008982]
[3.60590629 7.08063391]
[0.09702069 6.7808758]

We can calculate the means for each class:

$$\mu_1 = \begin{bmatrix} \frac{3.86317368 + 1.81099457 + 3.68588236 + 8.84837398 + 3.24418499}{5}, \\ \frac{6.98028738 + 6.01214467 + 3.72158188 + 5.98633342 + 1.70899736}{5} \end{bmatrix} = \begin{bmatrix} 4.29 \\ 4.88 \end{bmatrix}$$

$$\mu_2 = \begin{bmatrix} \frac{5.6974558 + 10.24735416 + 3.7893197 + 3.60590629 + 0.09702069}{5} \\ \frac{18.28342712 + 12.71358796 + 10.94008982 + 7.08063391 + 6.7808758}{5} \end{bmatrix} = \begin{bmatrix} 4.69 \\ 11.16 \end{bmatrix}$$

We can calculate the covariance matrices for each class. We will assume that the sample mean will be used to calculate the covariance matrices. Starting with class 1:

$$\Sigma_1 = \begin{bmatrix} Var(x_1) & Cov(x_1, x_2) \\ Cov(x_2, x_1) & Var(x_2) \end{bmatrix}$$

$$Var(x_1) = \frac{(3.86 - 4.29)^2 + (1.81 - 4.29)^2 + (3.69 - 4.29)^2 + (8.84 - 4.29)^2 + (3.24 - 4.29)^2}{5 - 1} = 7.13$$

$$Var(x_2) = \frac{(6.98 - 4.88)^2 + (6.01 - 4.88)^2 + (3.72 - 4.88)^2 + (5.99 - 4.88)^2 + (1.71 - 4.88)^2}{5 - 1} = 4.58$$

$$Cov(x_1, x_2) = Cov(x_2, x_1) = \frac{\begin{gathered}(3.86 - 4.29)(6.98 - 4.88) + (1.81 - 4.29)(6.01 - 4.88) + (3.69 - 4.29)(3.72 - 4.88) + \\ (8.84 - 4.29)(5.99 - 4.88) + (3.24 - 4.29)(1.71 - 4.88)\end{gathered}}{5 - 1} = 1.34$$

$$\Sigma_1 = \begin{bmatrix} 7.13 & 1.34 \\ 1.34 & 4.58 \end{bmatrix}$$

For class 2:

$$\mu_2 = \begin{bmatrix} \frac{5.6974558 + 10.24735416 + 3.7893197 + 3.60590629 + 0.09702069}{5} \\ \frac{18.28342712 + 12.71358796 + 10.94008982 + 7.08063391 + 6.7808758}{5} \end{bmatrix} = \begin{bmatrix} 4.69 \\ 11.16 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} Var(x_1) & Cov(x_1, x_2) \\ Cov(x_2, x_1) & Var(x_2) \end{bmatrix}$$

$$Var(x_1) = \frac{(5.70 - 4.69)^2 + (10.25 - 4.69)^2 + (3.79 - 4.69)^2 + (3.61 - 4.69)^2 + (0.10 - 4.69)^2}{5 - 1} = 13.74$$

$$Var(x_2) = \frac{(18.28 - 11.16)^2 + (12.71 - 11.16)^2 + (10.94 - 11.16)^2 + (7.08 - 11.16)^2 + (6.78 - 11.16)^2}{5 - 1} = 22.24$$

$$Cov(x_1, x_2) = Cov(x_2, x_1)$$

$$= \frac{\begin{array}{c}(5.70 - 4.69)(18.28 - 11.16) + (10.25 - 4.69)(12.71 - 11.16) + (3.79 - 4.69)(10.94 - 11.16) + \\ (3.61 - 4.69)(7.08 - 11.16) + (0.10 - 4.69)(6.78 - 11.16)\end{array}}{5 - 1}$$

$$= 10.13$$

$$\Sigma_2 = \begin{bmatrix} 13.74 & 10.13 \\ 10.13 & 22.24 \end{bmatrix}$$

2.

We first determine the eigenvalues and eigenvectors of class 1:

$$\Sigma_1 = \begin{bmatrix} 7.13 & 1.34 \\ 1.34 & 4.58 \end{bmatrix}$$

$$\det(\lambda I - A) = 0$$

$$\det\left(\lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 7.13 & 1.34 \\ 1.34 & 4.58 \end{bmatrix}\right) = 0$$

$$\det \begin{bmatrix} \lambda - 7.13 & -1.34 \\ -1.34 & \lambda - 4.58 \end{bmatrix} = 0$$

$$\lambda^2 - 11.71\lambda + 30.86 = 0$$

$$\lambda_1 = 7.70, \lambda_2 = 4.01$$

For $\lambda_1 = 7.70$:

$$\left(7.70 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 7.13 & 1.34 \\ 1.34 & 4.58 \end{bmatrix}\right) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 0.57 & -1.34 \\ -1.34 & 3.12 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\left[\begin{array}{cc|c} 0.57 & -1.34 & 0 \\ -1.34 & 3.12 & 0 \end{array}\right]$$

$$\left[\begin{array}{cc|c} 1 & -2.35 & 0 \\ 0 & 0 & 0 \end{array}\right]$$

$$v_1 = \begin{bmatrix} 2.35 \\ 1 \end{bmatrix}$$

For $\lambda_2 = 4.01$:

$$\left(4.01 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 7.13 & 1.34 \\ 1.34 & 4.58 \end{bmatrix}\right) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} -3.12 & -1.34 \\ -1.34 & -0.57 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\left[\begin{array}{cc|c} -3.12 & -1.34 & 0 \\ -1.34 & -0.57 & 0 \end{array}\right]$$

$$\left[\begin{array}{cc|c} 1 & 0.43 & 0 \\ 0 & 0 & 0 \end{array}\right]$$

$$v_2 = \begin{bmatrix} -0.43 \\ 1 \end{bmatrix}$$

Thus, for class 1, the eigenvalues are 7.70 and 4.01. Their respective eigenvectors are $\begin{bmatrix} 2.35 \\ 1 \end{bmatrix}$ and $\begin{bmatrix} -0.43 \\ 1 \end{bmatrix}$.

We do the same for class 2:

$$\Sigma_2 = \begin{bmatrix} 13.74 & 10.13 \\ 10.13 & 22.24 \end{bmatrix}$$

$$\det(\lambda I - A) = 0$$
$$\det\left(\lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 13.74 & 10.13 \\ 10.13 & 22.24 \end{bmatrix}\right) = 0$$
$$\det \begin{bmatrix} \lambda - 13.74 & -10.13 \\ -10.13 & \lambda - 22.24 \end{bmatrix} = 0$$
$$\lambda^2 - 35.98\lambda + 202.96 = 0$$

$$\lambda_1 = 28.98, \lambda_2 = 7.00$$

For $\lambda_1 = 28.98$:

$$\left(28.98 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 13.74 & 10.13 \\ 10.13 & 22.24 \end{bmatrix}\right) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 15.24 & -10.13 \\ -10.13 & 6.74 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\left[\begin{array}{cc|c} 15.24 & -10.13 & 0 \\ -10.13 & 6.74 & 0 \end{array}\right]$$

$$\left[\begin{array}{cc|c} 1 & -0.66 & 0 \\ 0 & 0 & 0 \end{array}\right]$$

$$v_1 = \begin{bmatrix} 0.66 \\ 1 \end{bmatrix}$$

For $\lambda_2 = 7.00$:

$$\left(7.00\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}-\begin{bmatrix}13.74 & 10.13\\10.13 & 22.24\end{bmatrix}\right)\begin{bmatrix}x_1\\x_2\end{bmatrix}=\begin{bmatrix}0\\0\end{bmatrix}$$

$$\begin{bmatrix}-6.74 & -10.13\\-10.13 & -15.24\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix}=\begin{bmatrix}0\\0\end{bmatrix}$$

$$\left[\begin{array}{cc|c}-6.74 & -10.13 & 0\\-10.13 & -15.24 & 0\end{array}\right]$$

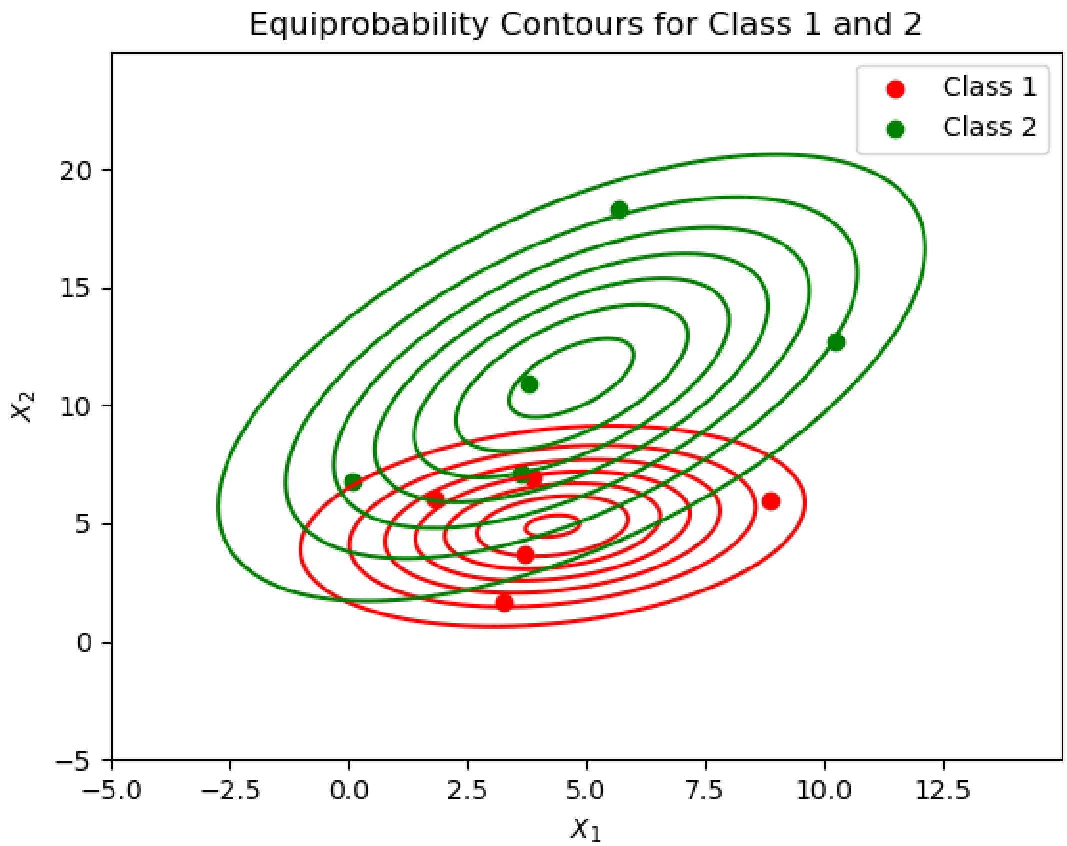
$$\left[\begin{array}{cc|c}1 & 1.50 & 0\\0 & 0 & 0\end{array}\right]$$

$$v_2=\begin{bmatrix}-1.50\\1\end{bmatrix}$$

The eigenvalues for class 2 are 28.98 and 7.00. Their respective eigenvectors are $\begin{bmatrix}0.66\\1\end{bmatrix}$ and $\begin{bmatrix}-1.50\\1\end{bmatrix}$.

3.

The code to plot the equiprobability contours for the two classes can be found in Exercise_1_3.py.



4.

The contours represent the points in a class that have an equal probability of occurring, according to the sample data. The centre of each contour represents the mean of the respective class. The axes of each contour represent the eigenvectors of the covariance matrix of the sample data for each class while the length of these axes represent the square roots of the eigenvalues of the sample data for each class.

We must also note that these equiprobability contours do not represent the actual two classes that were defined in the question, as the mean and covariance matrix for the sample was used, rather than the actual mean and covariance matrix that each class was defined with.

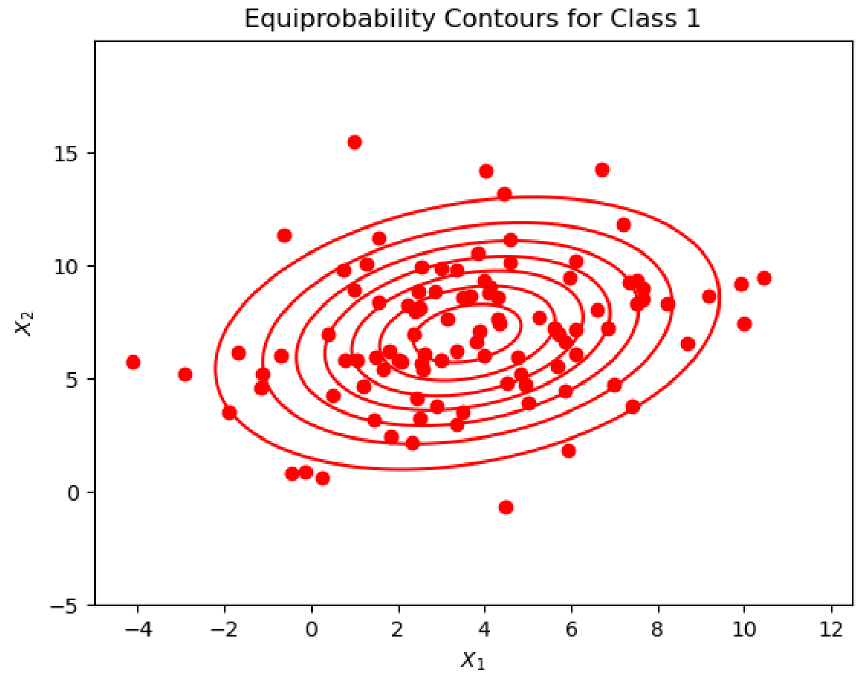
5.

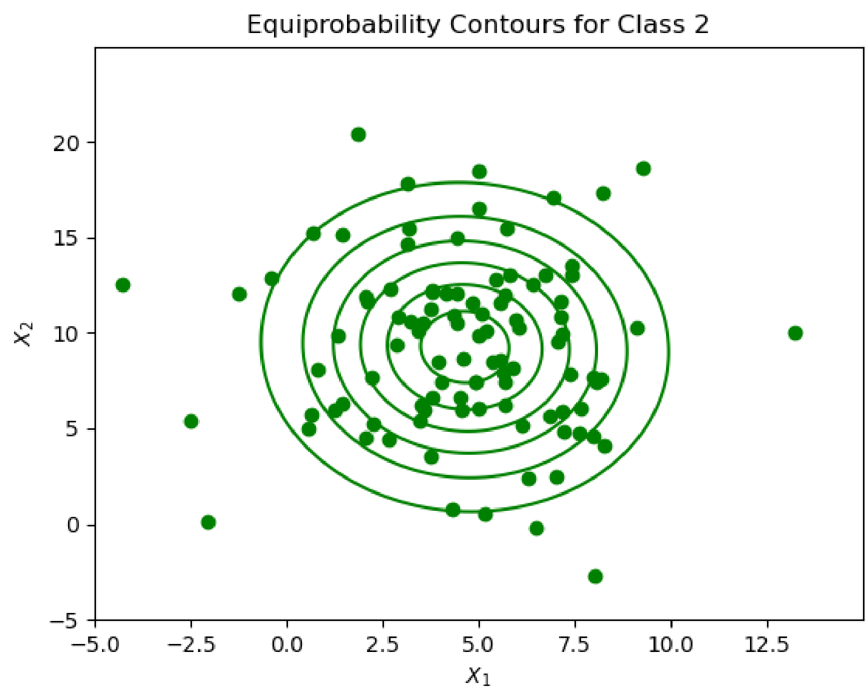
The code for this question can be found in Exercise_1_5.py. The output from the script containing each class's sample mean, covariance matrix, eigenvalues and eigenvectors can be seen below:

CLASS 1:

The sample mean is:
[3.6112466 7.01627777]
The sample covariance matrix is:
[[8.29526765 2.25241988]
[2.25241988 8.88987936]]
The eigenvalues of the covariance matrix are:
[6.32061706 10.86452995]
The corresponding eigenvectors are:
[[-0.75195045 0.65921963]
[0.65921963 0.75195045]]
CLASS 2:
The sample mean is:
[4.63708248 9.25835522]
The sample covariance matrix is:
[[7.48133482 -0.36215196]
[-0.36215196 19.70349746]]
The eigenvalues of the covariance matrix are:
[7.47061339 19.71421889]
The corresponding eigenvectors are:
[[-0.99956207 -0.02959182]
[-0.02959182 0.99956207]]

The generated equiprobability contours for the two classes can be seen below.





Due to the larger sample size, the contours for each class show characteristics that match the original class definitions better. Looking at the sample mean and covariance matrix for each class, it is much closer to the class's actual values in the larger sample.

Overall, a larger sample size allows for a better representation of the population. The variance in a small sample leads to a much more skewed sample mean and covariance matrix. Because of this, the contours representing the small sample are much different in their shape and orientation compared to the larger sample, which has contours that are more expected, given their true mean and covariance matrix.

Exercise 2: MED Classifier

1.

From Exercise 1, we know the means of the samples for each class.

$$\mu_1 = \begin{bmatrix} 4.29 \\ 4.88 \end{bmatrix} = z_1$$

$$\mu_2 = \begin{bmatrix} 4.69 \\ 11.16 \end{bmatrix} = z_2$$

We can now calculate the discriminant function and decision boundary:

$$\begin{gathered} -z_1^T x + \frac{1}{2} z_1^T z_1 \lesseqgtr -z_2^T x + \frac{1}{2} z_2^T z_2 \\ -\begin{bmatrix} 4.29 \\ 4.88 \end{bmatrix}^T \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 4.29 \\ 4.88 \end{bmatrix}^T \begin{bmatrix} 4.29 \\ 4.88 \end{bmatrix} \lesseqgtr -\begin{bmatrix} 4.69 \\ 11.16 \end{bmatrix}^T \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 4.69 \\ 11.16 \end{bmatrix}^T \begin{bmatrix} 4.69 \\ 11.16 \end{bmatrix} \\ -(4.29x_1 + 4.88x_2) + \frac{1}{2}(42.22) \lesseqgtr -(4.69x_1 + 11.16x_2) + \frac{1}{2}(146.54) \\ -4.29x_1 - 4.88x_2 + 21.11 \lesseqgtr -4.69x_1 - 11.16x_2 + 73.27 \end{gathered}$$

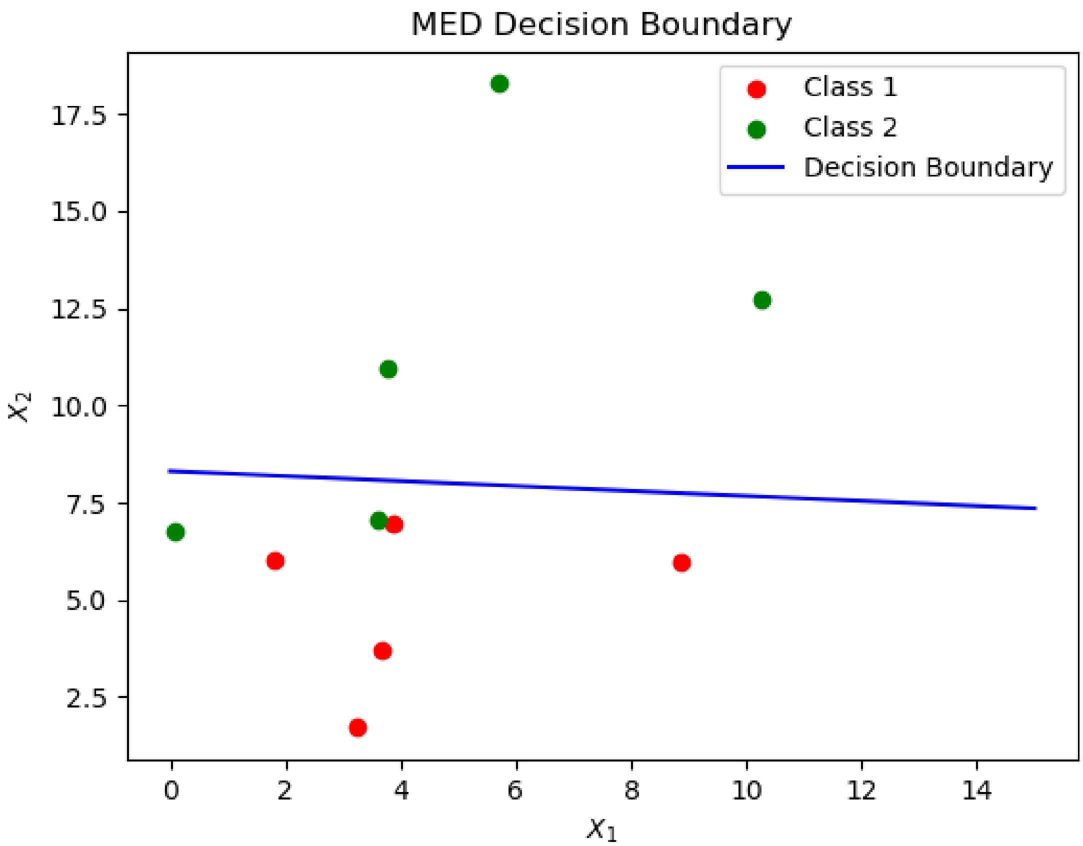
$$g(x) = g_1(x) - g_2(x) = 0$$

$$0.4x_1 + 6.28x_2 - 52.16 = 0$$

According to this equation, any point that gives a value of $0.4x_1 + 6.28x_2 - 52.16$ that is less than 0 belongs to class 1, while anything greater than 0 belongs to class 2. Using this, we can find the decision boundary.

$$\begin{aligned} x_2 &= \frac{-0.4x_1 + 52.16}{6.28} \\ &= -0.064x_1 + 8.31 \end{aligned}$$

The decision boundary is defined as: $x_2 = -0.064x_2 + 8.31$. We can now plot this decision boundary along with the points of each class, using the code from Exercise_2_1.py.



2.

We can perform the same operations on the 100 sample datasets for the two classes. Using the code from Exercise_2_2.py, we can determine the mean for each class. The means are:

$$\mu_1 = \begin{bmatrix} 3.61 \\ 7.02 \end{bmatrix} = z_1$$

$$\mu_2 = \begin{bmatrix} 4.64 \\ 9.26 \end{bmatrix} = z_2$$

The decision boundary was calculated using Python code. The equation was determined to be:

$$x_2 = -0.46x_1 - 10.02$$

This was verified using the following math:

$$-z_1^T x + \frac{1}{2} z_1^T z_1 \lessgtr -z_2^T x + \frac{1}{2} z_2^T z_2$$

$$-\begin{bmatrix} 3.61 \\ 7.02 \end{bmatrix}^T \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 3.61 \\ 7.02 \end{bmatrix}^T \begin{bmatrix} 3.61 \\ 7.02 \end{bmatrix} \lessgtr -\begin{bmatrix} 4.64 \\ 9.26 \end{bmatrix}^T \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 4.64 \\ 9.26 \end{bmatrix}^T \begin{bmatrix} 4.64 \\ 9.26 \end{bmatrix}$$

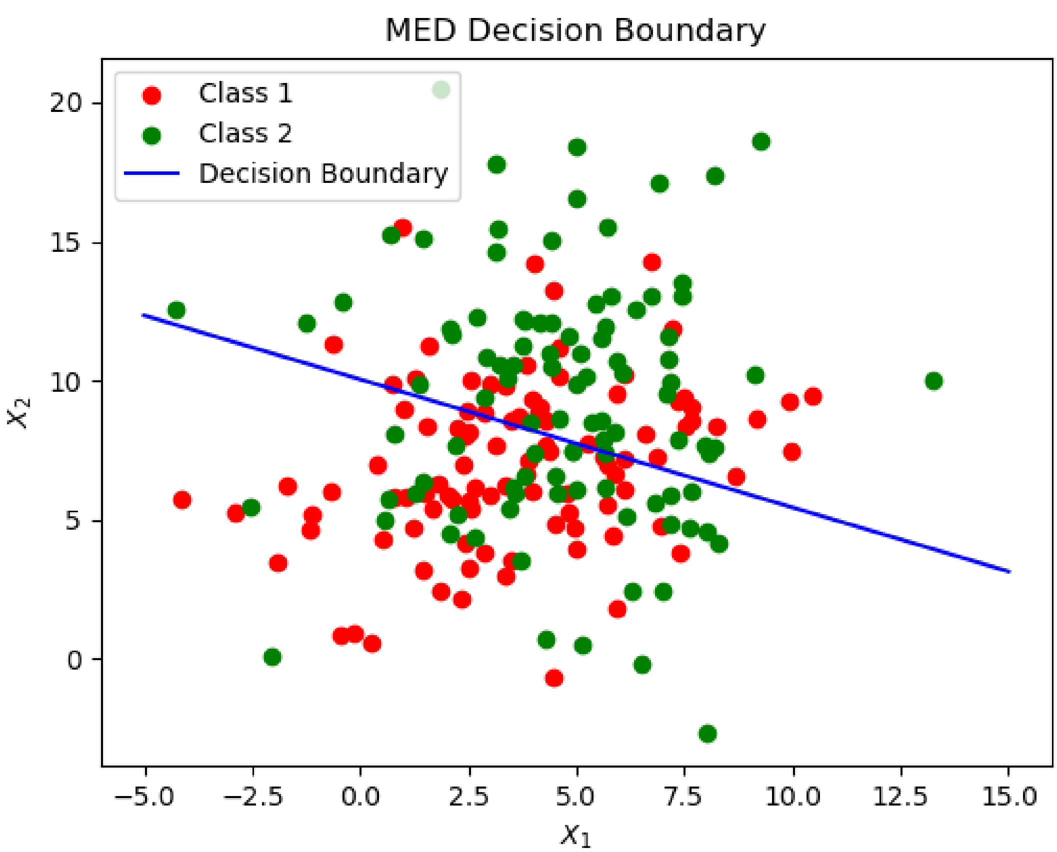
$$-(3.61x_1 + 7.02x_2) + \frac{1}{2}(62.31) \lessgtr -(4.64x_1 + 9.26x_2) + \frac{1}{2}(107.28)$$
$$-3.61x_1 - 7.02x_2 + 31.16 \lessgtr -4.64x_1 - 9.26x_2 + 53.67$$

$$g(x) = g_1(x) - g_2(x) = 0$$

$$1.03x_1 + 2.24x_2 - 22.51 = 0$$

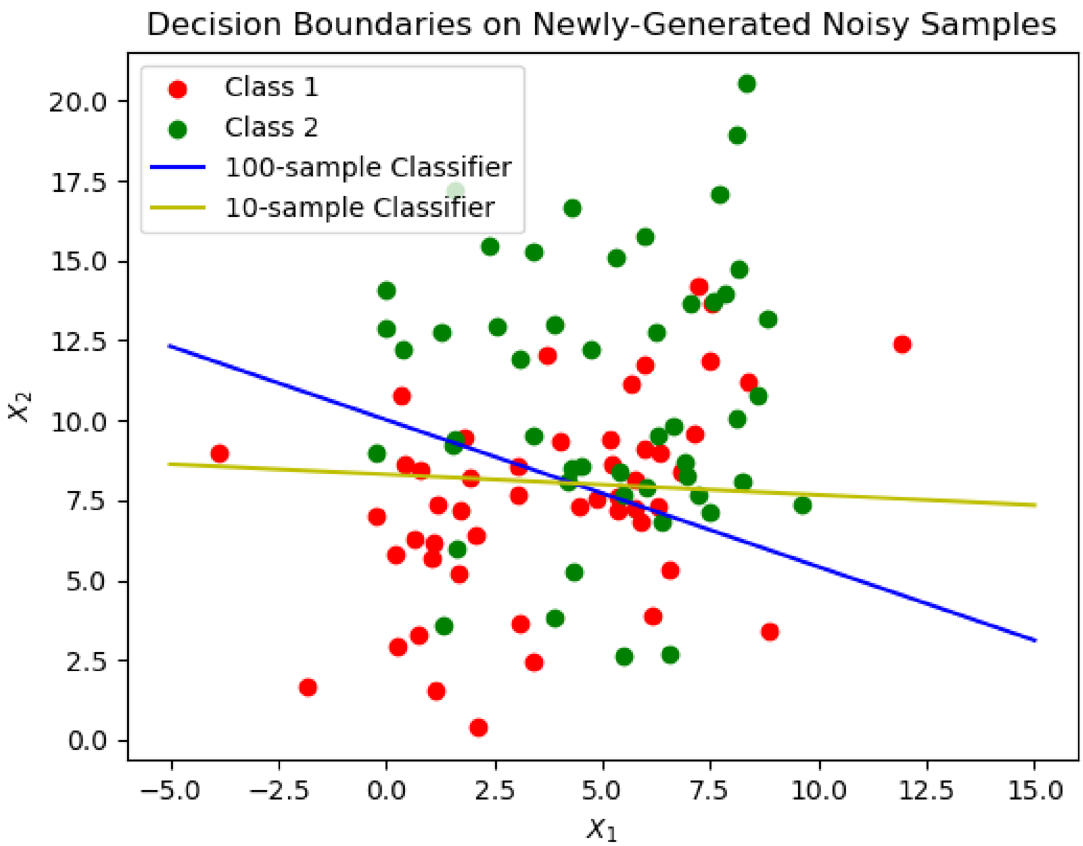
$$x_2 = -0.46x_1 + 10.04$$

The plot can be seen below, created using the code in Exercise_2_2.py.



3.

Using the code in Exercise_2_3.py, 50 new samples were generated per class. White noise was then added to each point with 0 mean and identity covariance matrix. It was determined that the accuracy for the 10-sample classifier on the 100 samples generated was 65%, while the accuracy for the 100-sample classifier was 70%. A plot showing the newly-generated points as well as the decision boundaries for each classifier can be seen below:



4.

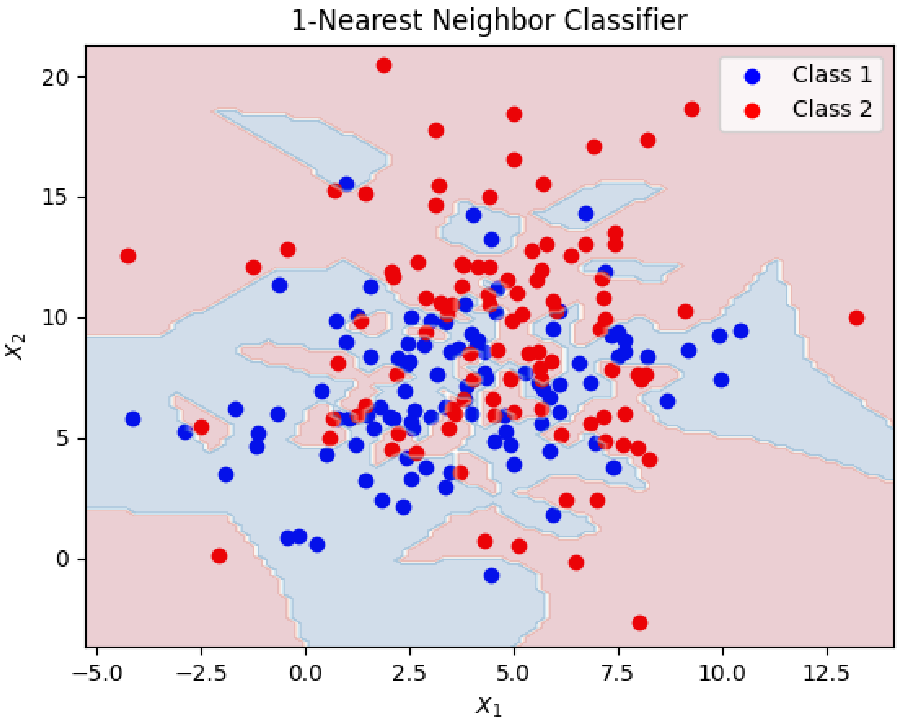
The classifier trained on 100 samples had a higher accuracy on the newly-generated data than the classifier trained on 10 samples. This is because having a larger number of samples means that the calculated mean for each class would be closer to the actual mean than with a smaller number of samples. That is, the mean calculated using the 50 samples per class was much truer to the actual means defined in Exercise 1 than the mean calculated using the 5 samples per class. With only 5 samples representing each class, the classifier is more likely to be affected by outliers and other noise in the data.

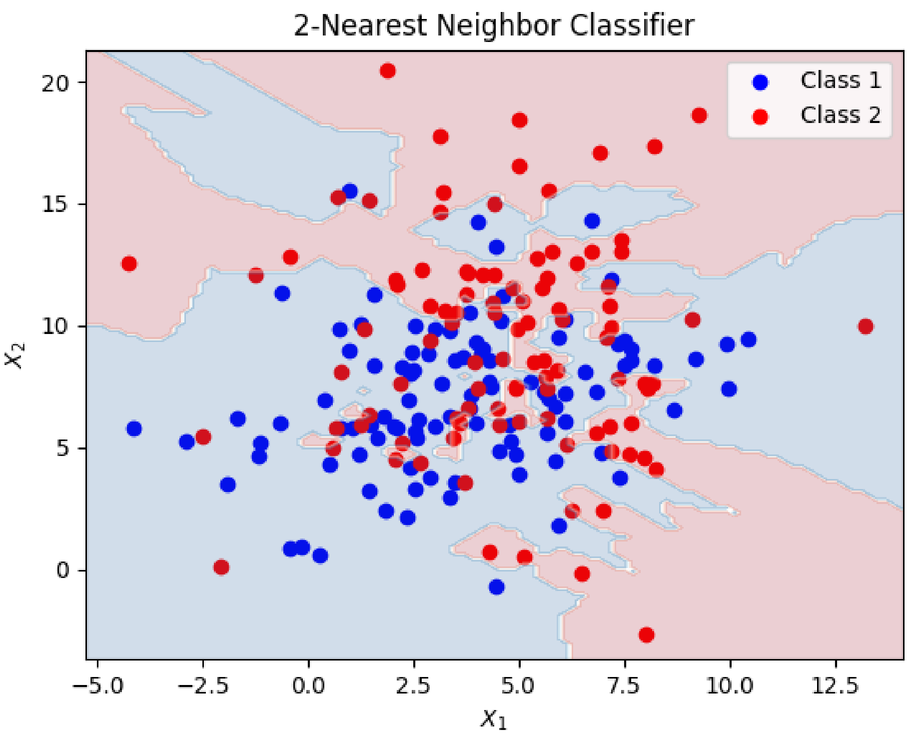
However, when comparing the actual equations, we see that both classifiers have a similar accuracy and shape. This is likely due to the high amount of overlap between the data in each class. Because the data is not very linearly separable, the overall accuracy of the data is not as impressive as if the data were more linearly separable.

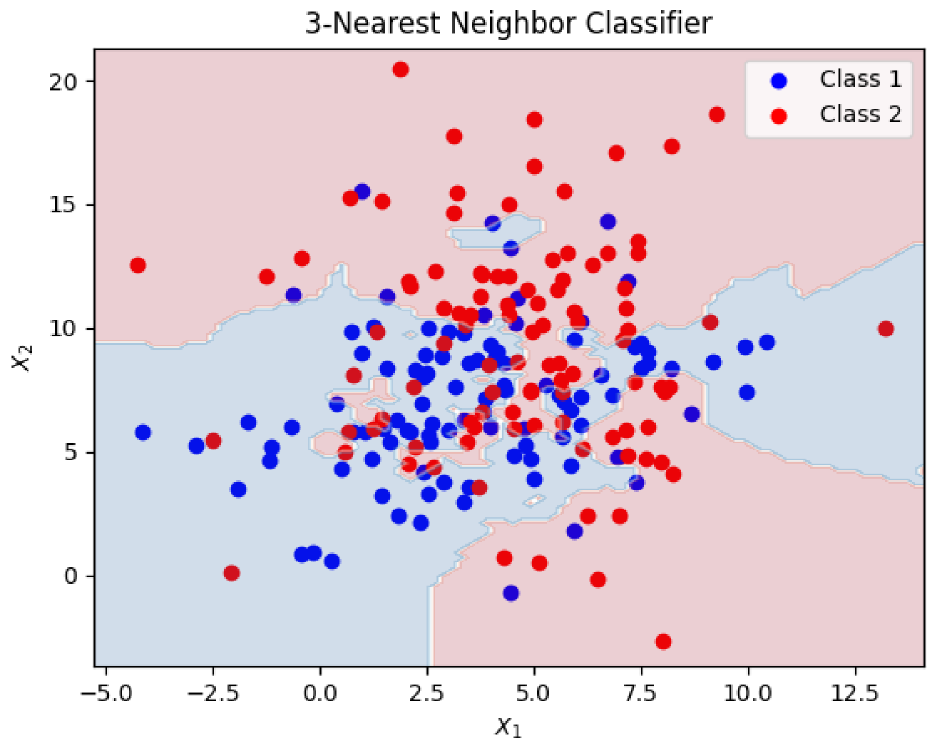
Exercise 3: Nearest Neighbor Classifier

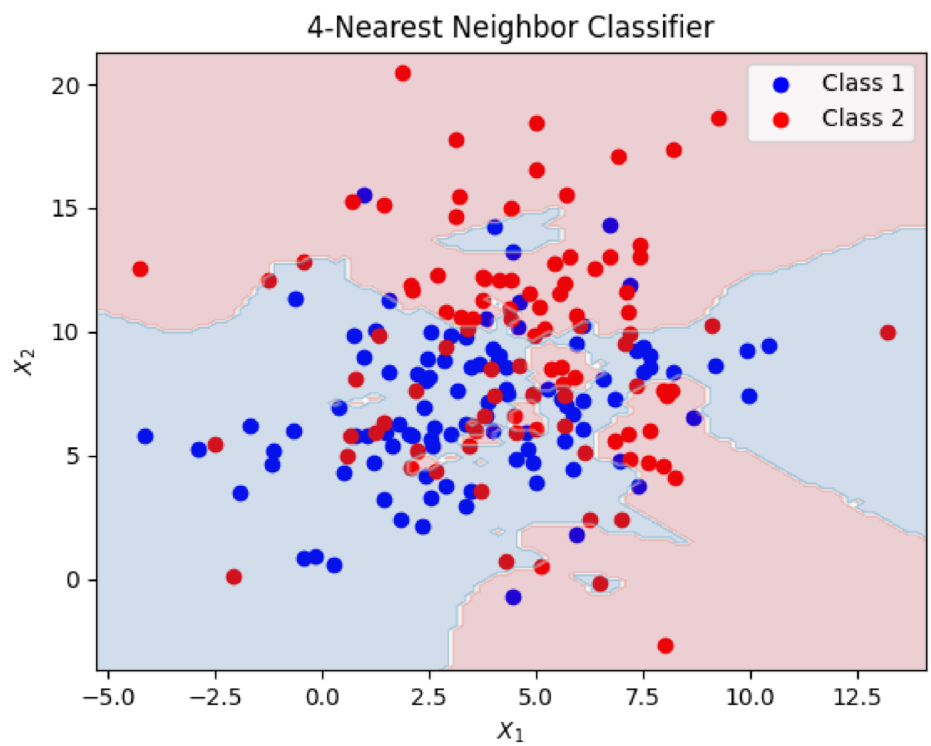
1.

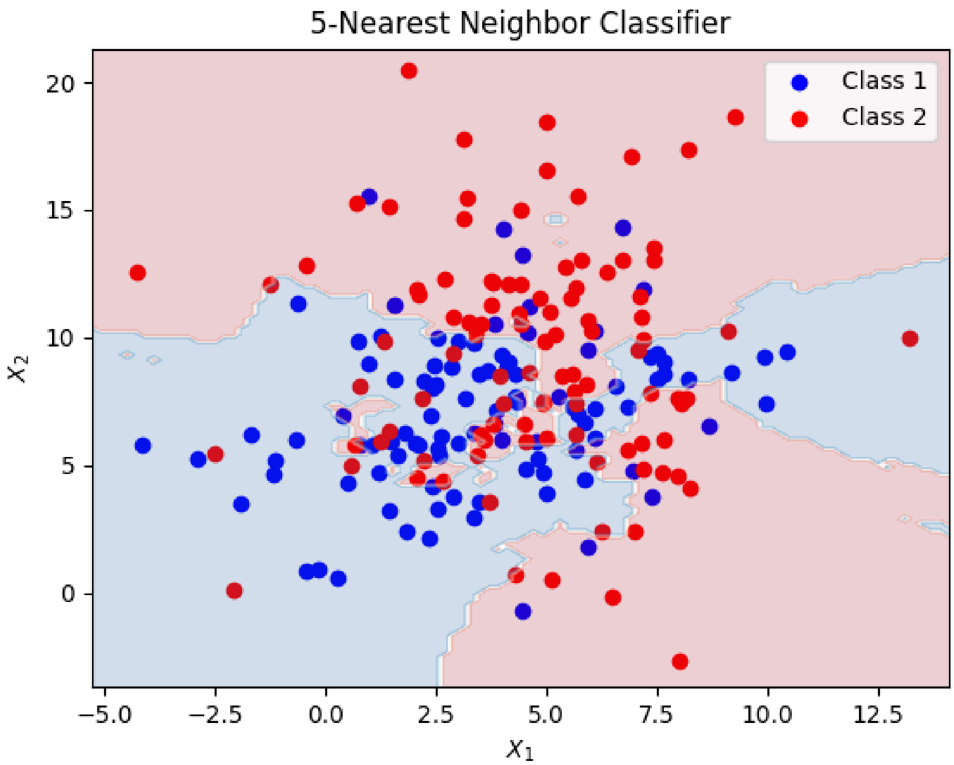
The plots containing the classification boundaries for each value of k can be seen below. The plots were generated using the code found in Exercise_3_1.py.





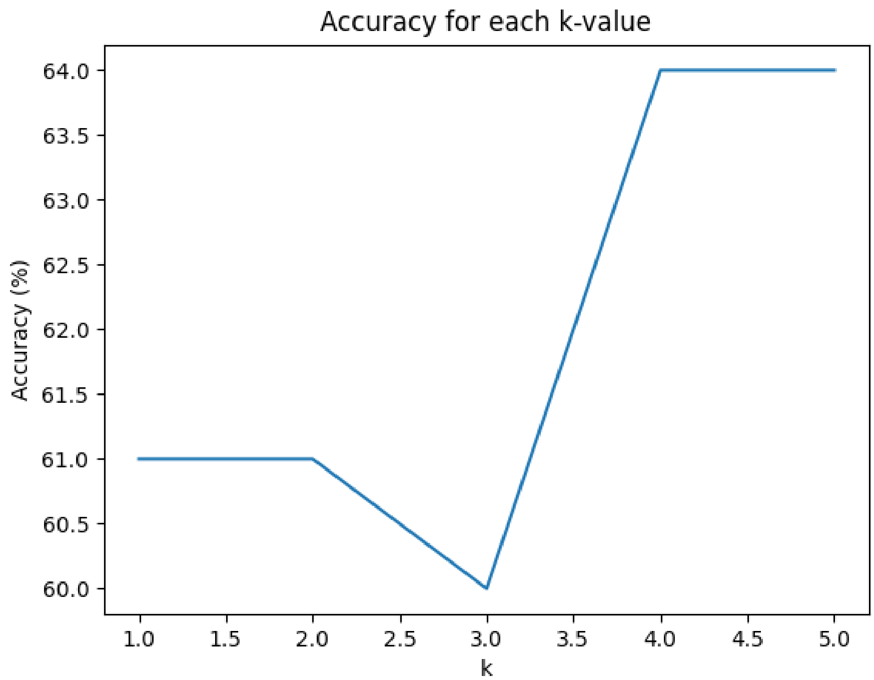

3-Nearest Neighbor Classifier
Class 1
Class 2
X_2
X_1
20
15
10
5
0
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
12.5


4-Nearest Neighbor Classifier
Class 1
Class 2
X_2
X_1
20
15
10
5
0
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
12.5



2.

The noisy data was classified using the kNN classifiers created above. The accuracies of each k-value were then recorded and the following plot shows the accuracy of the classifier as k increases.



3.

The k-values of 4 and 5 seem to be producing the best results. However, there is not a significant increase in accuracy compared to other k-values, with accuracies of 64% for k-values of 4 and 5. Meanwhile, the accuracy for k-values of 1 and 2 are 61% and the accuracy for the k-value of 3 is 60%.

Nonetheless, it makes sense that larger values of k leads to higher accuracies as the classifier becomes less sensitive to noise with a higher value of k. This allows outliers to have a smaller effect on the final decision boundary. However, looking at the graphs of the decision boundary as k increases, we see that the decision boundary smooths out as k increases. This could negatively affect the accuracy, which could explain why there is a decrease in accuracy with a k-value of 3.

4.

The accuracy of the kNN classifier is slightly lower than the accuracy of the MED classifier. This is likely because of the large amount of overlap between the data points in the set. Because of the large amount of overlap, the kNN classifier has difficulty distinguishing between the two classes for points that are contained in the overlapping areas. However, since the two classes were defined as having relatively different means, the MED classifier would work better as points closer to a particular class's mean would be more likely to be contained in that class.

The kNN classifier was also much slower in comparison to the MED classifier, as each new point would have to be compared to every other point in the 100-sample dataset. This would negatively affect the performance of the classifier exponentially as the number of points in the dataset increases.