CS479 - Programming Assignment 1: Bayes Classifier

Tim Kwist and Shane Melton

Due February 22, 2016 - Submitted February 22, 2016

Contents

1 Technical Discussion

Equations:
Multivariate Gaussian Density Case I
Discriminant:

$$g_i(x) = [\frac{1}{\sigma^2}\mu_i]^t x + -\frac{1}{2\sigma^2}\mu_i^t\mu_i + lnP(w_i) \tag{1}$$

Decision Boundary:

$$(\mu_{\mathbf{i}} - \mu_{\mathbf{0}})^t(\mathbf{x} - (\frac{1}{2}(\mu_{\mathbf{i}} + \mu_{\mathbf{j}}) - \frac{\sigma^2}{\|\mu_{\mathbf{i}} - \mu_{\mathbf{j}}\|^2} ln\frac{P(w_i)}{P(w_j)}(\mu_{\mathbf{i}} - \mu_{\mathbf{j}}))) = 0 \tag{2}$$

Minimum Distance Classifier:

$$g_i(x) = -\|\mathbf{x} - \mu_{\mathbf{i}}\|^2 \tag{3}$$

Multivariate Gaussian Density Case III
Discriminant:

$$g_i(x) = \mathbf{x}^t(-\frac{1}{2}\mathbf{\Sigma_i^{-1}})\mathbf{x} + (\Sigma_i^{-1}\mu_{\mathbf{i}})\mathbf{x} + -\frac{1}{2}\mu_{\mathbf{i}}^{\mathbf{t}}\mathbf{\Sigma^{-1}}\mu_{\mathbf{i}} - \frac{1}{2}ln|\Sigma_i| + lnP(w_i) \tag{4}$$

Decision Boundary:

$$g_i(x) = g_j(x) \tag{5}$$

Error Bounds

$$\kappa(\beta) = \frac{\beta(1-\beta)}{2}(\mu_{\mathbf{1}} - \mu_{\mathbf{2}})^t[(1-\beta)\mathbf{\Sigma_1} + \beta\mathbf{\Sigma_2}]^{-1}(\mu_{\mathbf{1}} - \mu_{\mathbf{2}}) + \frac{1}{2}ln\frac{|(1-\beta)\mathbf{\Sigma_1} + \beta\mathbf{\Sigma_2}|}{|\Sigma_1|^{1-\beta}|\Sigma_2|^{\beta}} \tag{6}$$

Chernoff Bound:

$$Minimize\ e^{-\kappa(\beta)} \tag{7}$$

Bhattacharrya Bound:

$$For\ e^{-\kappa(\beta)}\ Set\ \beta = 0.5 \tag{8}$$

For generating results, the Box-Muller transformation was performed, using C++ code from Everett F. Carter Jr.

In this programming assignment, Bayesian Minimum-Error techniques are used to develop a classifier to describe and differentiate samples generated from 2D Gaussian distributions based on various parameters. In classifying data based on Gaussian distributions, certain assumptions about the data can be useful to simplify the discriminant functions necessary to determine classification. In this assignment, we take advantage of the assumptions necessary to utilize Case I and Case III.

For Case I, the following assumption is necessary:

$$For\ each\ distribution\ i,\ \Sigma_i = \sigma^2 I \tag{9}$$

Given this assumption, we can utilize the discriminant from Equation 1 and the decision boundary from Equation 2.

For Case III, the following assumption is necessary:

$$For\ each\ distribution\ i,\ \Sigma_i = arbitrary \tag{10}$$

Given this assumption, we can utilize the discriminant from Equation 4 and the decision boundary from Equation 5.

Given the above definitions and equations, the parameters given for each distribution are as follows:

$$\mu_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \Sigma_1 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \mu_2 = \begin{bmatrix} 6 \\ 6 \end{bmatrix} \Sigma_2 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \tag{11}$$

$$\mu_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \Sigma_2 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \mu_2 = \begin{bmatrix} 6 \\ 6 \end{bmatrix} \Sigma_2 = \begin{bmatrix} 4 & 0 \\ 0 & 8 \end{bmatrix} \tag{12}$$

For the data sets in (11), the assumption for Case I is satisfied. Thus, discriminant (1) and decision boundary (2) are used to classify the samples from data set (11).

For the data sets in (12), the assumption for Case III is satisfied. Thus, discriminant (4) and decision boundary (5) are used to classify the samples from data set (12).

For each set of data, two classification rounds are made: First, with

$$P(w_1) = P(w_2) = 0.5$$

Second, with

$$P(w_1) = 0.2\ and\ P(w_2) = 0.8$$

In addition to classification, the Chernoff Bound (as described in Equation 6 and Equation 7) is found for each data set and is plotted against the Bhattacharyya bound (described in Equation 8).

Finally, the minimum-distance classifier (described in Equation 3) is used for data set 12. It is important to note that the minimum-distance classifier is ideally used only for Case I, and is used for Case III in this example specifically to compare the performance against the Bayes Classifier for Case III.

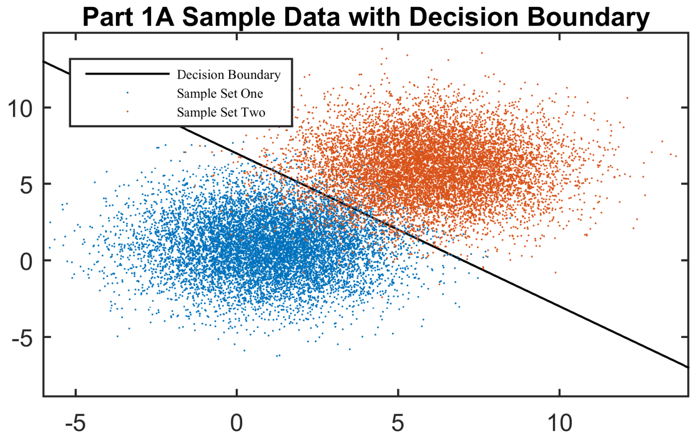

Figure 1: Using eq. 13 we plot the decision boundary with the sample data.

2 Results

After studying the technical details and math regarding Bayseian classifiers we implemented the theories with C++ and ran our program with the data sets generated by (11) and (12) to see how our implementation fared compared to the theory.

2.1 Part 1

Design a Bayes classifier to classify data generated by the parameters in (11) into two different sets.

As the technical details stated, this data set requires Case I, so to start off with the decision boundary was calculated using the discriminant (1) and can be seen below:

$$P(w_1) = 0.5 \; and \; P(w_2) = 0.5$$

$$g(x) = -x + 7 + \frac{ln(\frac{P(w_i)}{P(w_j)})}{1.25} \tag{13}$$

Plotting this boundary and the sample data (Fig. 1) starts to show places where the classifier will likely have difficulty classifying specific samples (i.e. samples near or around the decision boundary). After plotting the misclassified samples (Fig. 2) we can see that our predictions were correct and the misclassified samples congregate around the decision boundary. In total we find that 757 data samples are misclassified (351 from sample one and 406 from sample two).

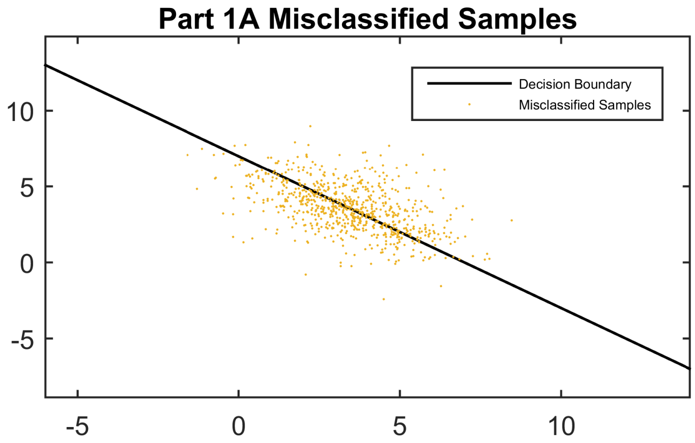

Figure 2: Misclassified sample data that is centered around the decision boundary

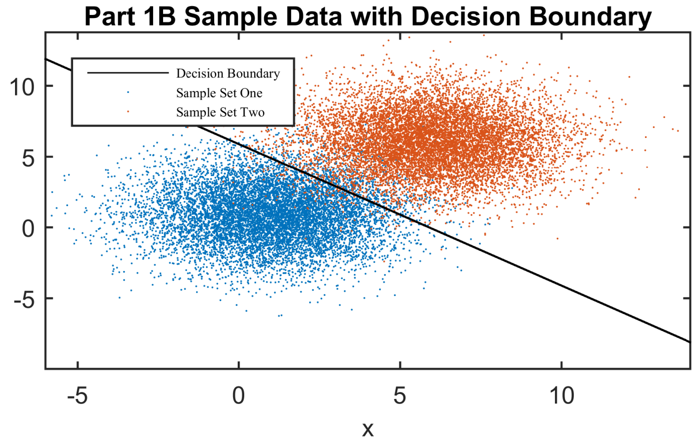

Figure 3: Re-plotting the decision boundary with the new probabilities.

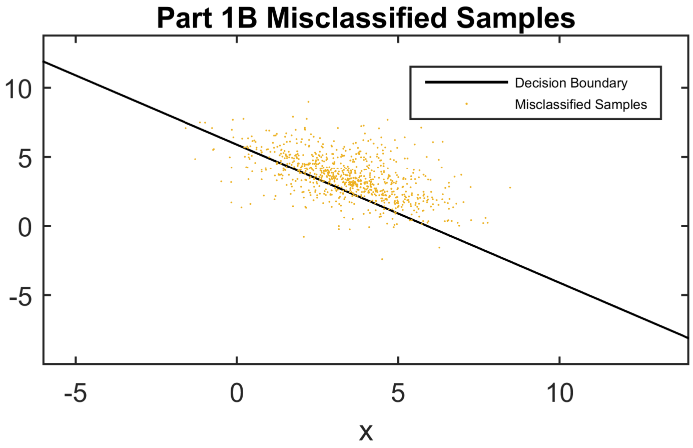

Figure 4: The misclassified data using the new decision boundary.

Next we experimented with the probabilities of each data set to see its effect on the classification accuracy.

$$P(w_1) = 0.2 \; and \; P(w_2) = 0.8$$

We re-plotted the decision boundary (Fig. 3) and the misclassified samples (Fig. 4) with this new information and found that number of misclassified samples increased to 832 (559 from sample one and 273 from sample two). We can see from the graph that the decision boundary moved away from sample set two in order to account for the higher probability of sample two (a now 80% probability).

Finally we calculated and graphed the error bound for this data set. The error boundary equation can be seen in equation (14) and by minimizing this bound we found the Chernoff bound to be at $\beta = 0.5$ with a value of 0.439369. We also calculated the Bhattacharyya bound to be the same(using $\beta = 0.5$) at 0.439369. A graph of this error bound can be seen in Figure 5.

$$e^{-(\beta(1-\beta))(\frac{25}{2})} \tag{14}$$

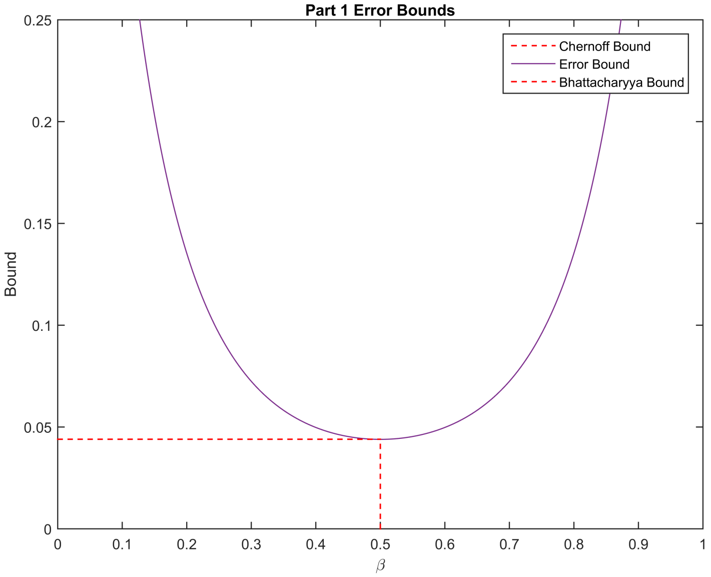

Figure 5: A graph of the Chernoff and Bhattacharyya error bounds

2.2 Part 2

Our next task was to use data set (12) and use Case III of the Bayes classifier on the data set and determine its accuracy. Again, we calculated the decision boundary (eq. 15), this time using equation 4 and 5 for Case III, to find an equation for an ellipse.

$$3x_2^2 + 4x_2 = -2x_1^2 - 16x_1 + 101.04 \tag{15}$$

Plotting this boundary with the sample data set (Fig. 6) again shows us likely places for misclassified samples to appear. We also can see that one sample set is partially within the other, so we can expect to see a larger number of misclassified samples due to the overlap. This is confirmed once we plot the misclassified data (Fig. 7) and see that there are a much greater number of misclassified samples which do, in fact, cluster near the decision boundary. In total we find that 1975 data samples are misclassified (855 from sample one and 1120 from sample two).

Next we experimented with the probabilities of each data set to see its effect on the classification accuracy.

$$P(w_1) = 0.2 \; and \; P(w_2) = 0.8$$

We re-plotted the decision boundary (Fig. 8) and the misclassified samples (Fig. 9) with this new information and found that number of misclassified

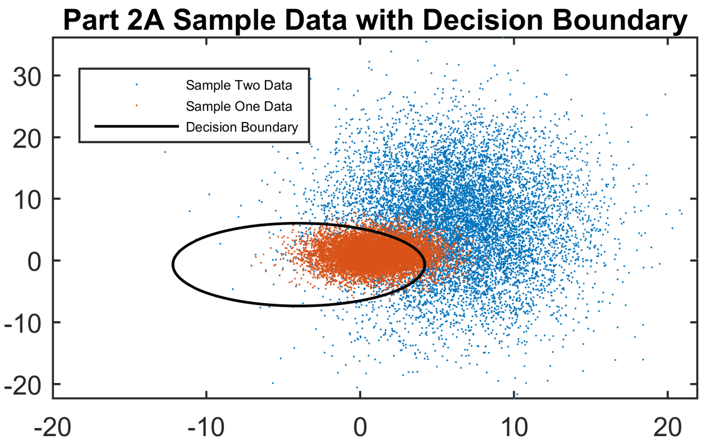

Figure 6: Using eq. 15 we plot the decision boundary with the sample data set.

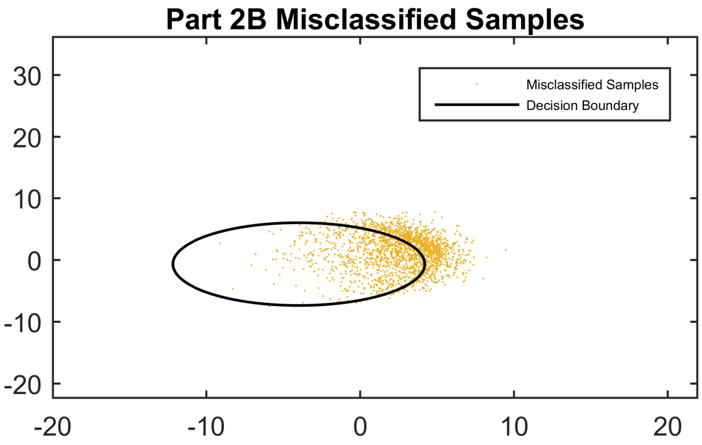

Figure 7: Misclassified sample data that clusters near the decision boundary.

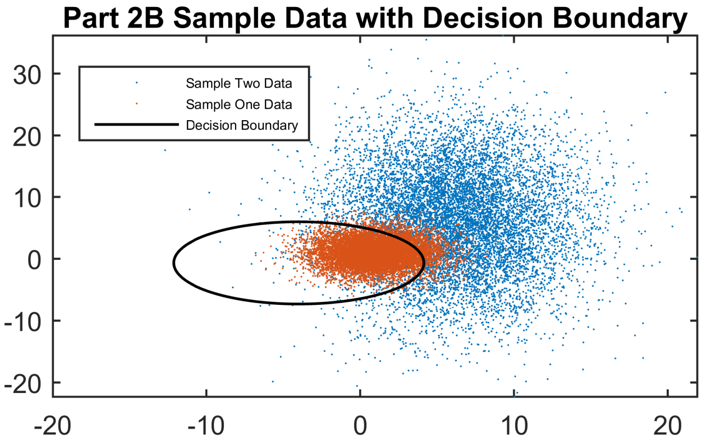

Figure 8: Re-plotting the decision boundary with the new probabilities.

samples increased to 2348 (1494 from sample one and 854 from sample two). We can see from the graph that the decision boundary moved away from sample set two in order to account for the higher probability of sample two (a now 80% probability).

Finally we calculated and graphed the error bound for this data set. The error boundary equation can be seen in equation (14) and by minimizing this bound we found the Chernoff bound to be at $\beta = 0.381991$ with a value of 0.20013 We also calculated the Bhattacharyya bound to be the same(using $\beta = 0.5$) at 0.21748. A graph of this error bound can be seen in Figure 10.

$$e^{-(\frac{1}{2}(1-x)x(\frac{25}{6x+2}\frac{25}{2x+2})+\frac{1}{2}ln(\frac{(2+2x)(2+6x)}{4^{1-x}32^{x}}))} \tag{16}$$

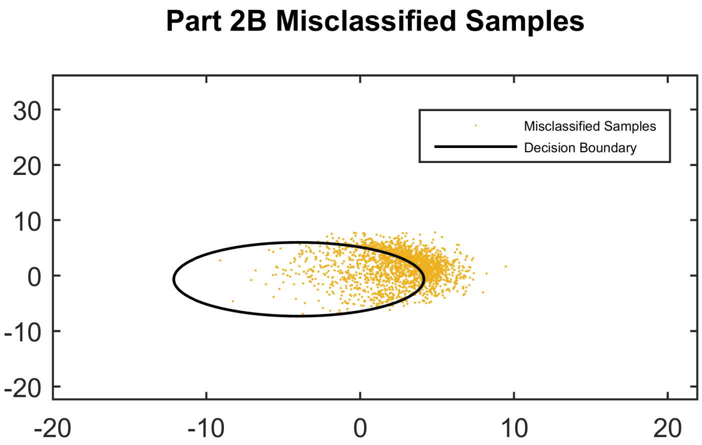

Figure 9: The misclassified data using the new decision boundary.

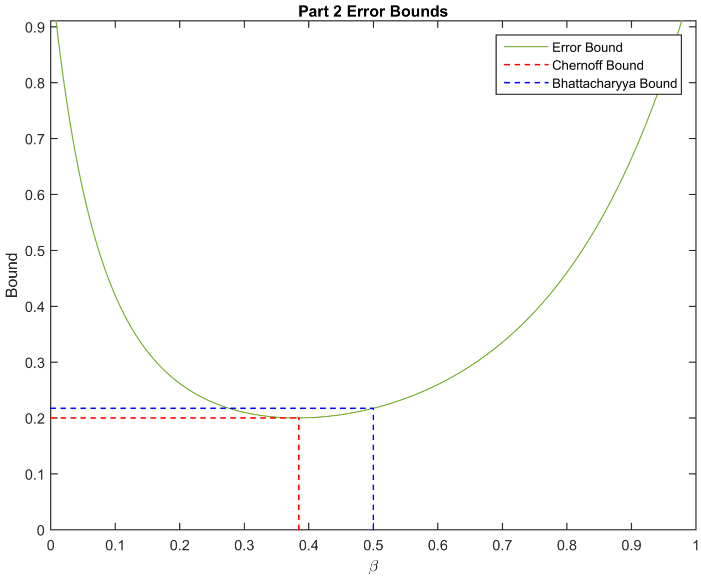

Figure 10: A graph of the Chernoff and Bhattacharyya error bounds

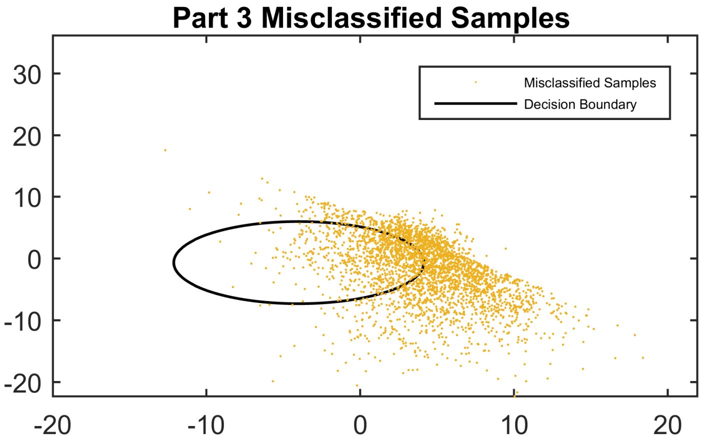

Figure 11: Misclassified samples using the minimum distance classifier (3222 misclassified samples)

2.3 Part 3

Next we decided to compare the number of misclassified samples from Part2A using the Bayes classifier (Fig. 7) to the number of misclassified samples that come from using a minimum distance classifier (Fig. 11). As we can see the Bayes classifier is far superior to the minimum distance classifier which caused an additional 1247 samples to be misclassified (totaling 3222 misclassified samples for the minimum distance classifier).

3 Division of Work

Shane:

- Programming of Case III, data set 12 discriminant
- Find mathematical decision boundary for Case I, data set 11
- Plot data, decision boundaries, and error bounds
- Write up Results section of report

Tim:

- Programming of Case I, data set 11 discriminant
- Find mathematical decision boundary for Case III, data set 12
- Programming of code to find error bounds and Minimum-Distance Classifier
- Write up Technical Discussion section of report

4 Program Listings

Source will be sent by email and can be found on Github at the following URL:
`https://github.com/timkwist/CS479`

Eigen library can be found at the following URL:
`http://eigen.tuxfamily.org/index.php?title=Main_Page`

Box-Muller Transformation C++ Code:
`ftp://ftp.taygeta.com/pub/c/boxmuller.c`