Homework 3

CSE 802 - Pattern Recognition and Analysis
Instructor: Dr. Arun Ross
Points: 150
Due Date: March 29, 2021

Note:

1. **You are permitted to discuss the following questions with others in the class. However, you *must* write up your *own* solutions to these questions. Any indication to the contrary will be considered an act of academic dishonesty. Copying from *any source* constitutes academic dishonesty.**
2. **A neatly typed report is expected (alternately, you can neatly handwrite the report and then scan it). The report, in PDF format, must be uploaded as a *separate* file in D2L by March 29, 12:40 pm. Late submissions will not be graded. In your submission, please include the names of individuals you discussed this homework with and the list of external resources (e.g., websites, other books, articles, etc.) that you used to complete the assignment (if any).**
3. **When solving equations or reducing expressions you must explicitly show every step in your computation and/or include the code that was used to perform the computation. Missing steps or code will lead to a deduction of points.**
4. **Code developed as part of this assignment must be (a) included as an appendix to your report or inline with your solution, and (b) archived in a single separate zip file and uploaded in D2L. Including the code without the outputs or including the outputs without the code will result in deduction of points.**

---

1. [10 points] Consider the multivariate normal density with mean $\boldsymbol{\mu} = (\mu_1, \mu_2, \dots \mu_d)^t$, $\sigma_{ij} = 0$, $\forall i \neq j$, and $\sigma_{ii} = {\sigma_i}^2$, that is, the covariance matrix is diagonal: $\boldsymbol{\Sigma} = \text{diag}\big(\sigma_1^2, \sigma_2^2, \dots \sigma_d^2\big)$. Show that,

$$p(x) = \frac{1}{\prod_{i=1}^{d}\left(\sqrt{2\pi}\sigma_i\right)} exp\left[-\frac{1}{2}\sum_{i=1}^{d}\left(\frac{x_i - \mu_i}{\sigma_i}\right)^2\right].$$

Answer:

The multivariate normal density is as follow:

$$p(x) = \frac{1}{(2\pi)^{\frac{d}{2}} |\sum|^{\frac{1}{2}})} exp\left[-\frac{1}{2}(x-\mu)^t \sum^{-1}(x-\mu)\right]$$

From here, we have

$$\sum^{-1} = \begin{bmatrix} \frac{1}{\sigma_1^2} & & & \\ & \frac{1}{\sigma_2^2} & & \\ & & ... & \\ & & & \frac{1}{\sigma_d^2} \end{bmatrix}$$

$$(x-\mu)^t = [(x_1-\mu_1),(x_2-\mu_2),...,(x_d-\mu_d)]$$

$$(x-\mu) = \begin{bmatrix} x_1-\mu_1 \\ x_2-\mu_2 \\ ... \\ x_d-\mu_d \end{bmatrix}$$

Thus, we have

$$(x-\mu)^t \sum^{-1}(x-\mu) = \frac{(x_1-\mu_1)^2}{\sigma_1^2} + \frac{(x_2-\mu_2)^2}{\sigma_2^2} + ... + \frac{(x_d-\mu_d)^2}{\sigma_d^2}$$
$$= \sum_{i=1}^{d}\left(\frac{x_i-\mu_i}{\sigma_i}\right)^2$$

Determinant of the diagonal matrix is the product of the element of that diagonal matrix, that is

$$|\sum| = \prod_{i=1}^{d}\sigma_i^2$$

Thus,

$$|\sum|^{\frac{1}{2}} = \prod_{i=1}^{d}\sigma_i$$

Hence,

$$p(x) = \frac{1}{\prod_{i=1}^{d}(\sqrt{2\pi}\sigma_i}) exp\left[-\frac{1}{2}\sum_{i=1}^{d}\left(\frac{x_i-\mu_i}{\sigma_i}\right)^2\right]$$

2. Consider the following bivariate density function for the random variable $\boldsymbol{x}$:
$p_{original}(\boldsymbol{x}) \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 20 & 10 \\ 10 & 30 \end{bmatrix}\right)$.

(a) [5 points] Computer and report the whitening transform, $\boldsymbol{A_w}$, of $\boldsymbol{x}$? (You can use matlab (or any other programming language) to compute the eigenvectors/values).

Answer:

$$A_w = \begin{bmatrix} -0.22882456, & -0.0874032 \\ 0.14142136 & -0.14142136 \end{bmatrix}$$

(b) [3 points] When the whitening transformation is applied to $\boldsymbol{x}$, what is the density function, $p_{transform}$ of the resulting random variable?

Answer:

A whitening transform leads to a circular symmetric Gaussian, i.e.,

$$p(x) \sim N(A_w^t \mu, I)$$

We have $\mu = (0,0)^t$ and $\Sigma = I$, Thus the density function is as follows:

$$p(x) = \frac{1}{(2\pi)^{d/2}} \exp\left[-\frac{1}{2}x^t x\right]$$

(c) [5 points] Generate 10,000 bivariate *random* patterns from $p_{original}$ (if you are using matlab, then the *mvnrnd* function can be used to generate these patterns). Plot these patterns in a graph.

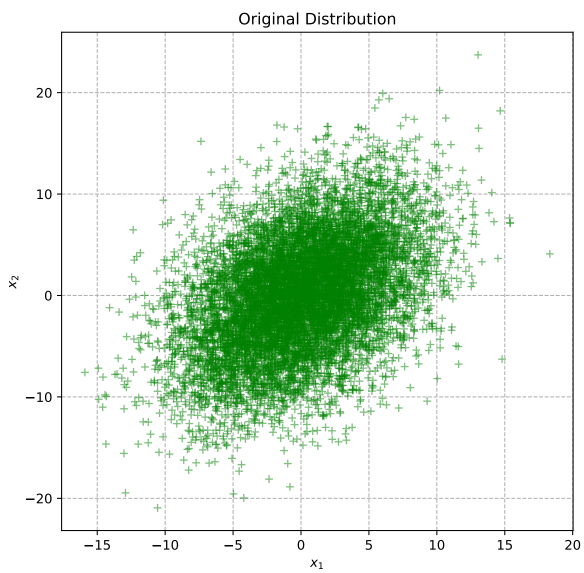

Figure 1: Distribution of 10,000 bivariate random patterns from P_{origin}

(d) [5 points] Apply the whitening transform, $\boldsymbol{A_w}$, to the 10,000 bivariate patterns generated above. Plot the transformed patterns in a separate graph.

Answer:

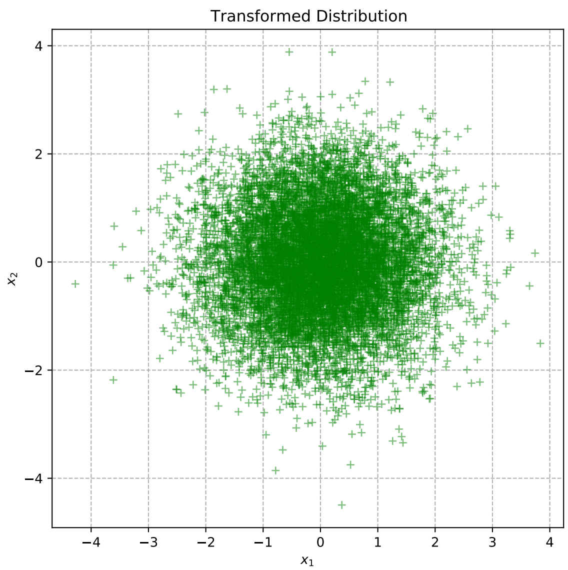

Figure 2: Distribution of 10,000 transformed random patterns from $P_{transform}$

(e) [2 points] Compare the patterns in 2c and 2d. **What do you observe?**.
Answer:
Plot in 2c shows that the patterns have positive correlated at the position of $\mu = (0,0)^t$, where spread along the x2 axis is higher than the spread along x1 axis. After applying whitening transformation, as shown in a plot of 2d, the transformed distribution results in a circularly symmetric Gaussian, having covariance matrix equal to the identity matrix.

3. [10 points] Consider the following class-conditional densities for a three-class problem involving two-dimensional features:

$$p(\boldsymbol{x}|\omega_1) \sim N\left((-1,-1)^t, I\right);$$
$$p(\boldsymbol{x}|\omega_2) \sim N\left((1,1)^t, I\right);$$
$$p(\boldsymbol{x}|\omega_3) \sim \frac{1}{2}N\left((0.5,0.5)^t, I\right) + \frac{1}{2}N\left((-0.5,-0.5)^t, I\right).$$

(Here, class ω_3 conforms to a **Gaussian Mixture Model (GMM)** with two components - one component is $N\left((0.5,0.5)^t, I\right)$ and the other component is $N\left((-0.5,-0.5)^t, I\right)$ - whose weights are equal (i.e., $\frac{1}{2}$)).

(a) In a 2D graph, mark the mean of ω_1, ω_2, and the two components of ω_3. In the same graph, mark the point $\boldsymbol{x} = (0.1, 0.1)^t$.

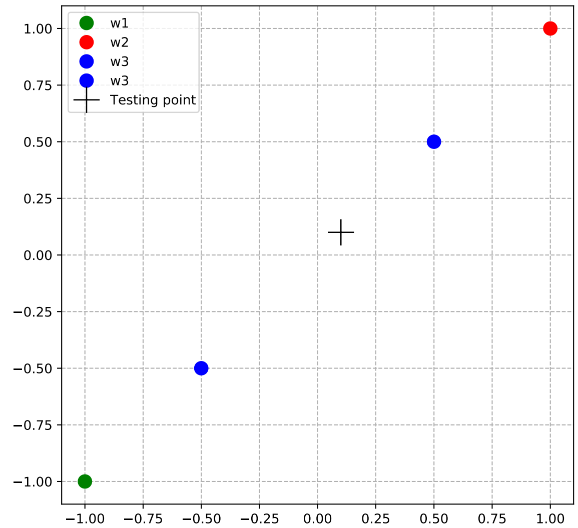

Figure 3: The means of ω_1, ω_2, the two components of ω_3, and the testing point $\boldsymbol{x} = (0.1, 0.1)^t$

(b) Assuming a 0-1 loss function and equal priors, determine the class to which you will assign the two-dimensional point $\boldsymbol{x} = (0.1, 0.1)^t$ based on the Bayes decision rule.

Answer:

Since the three-class problem has equal priors and 0-1 loss function is adopted. The three discriminate functions corresponding to the three classes can be written as follow:

$$g_1(x) = p(x|\omega_1) = exp\left[-\frac{1}{2}\left(x - \begin{bmatrix}-1\\-1\end{bmatrix}\right)^t \left(x - \begin{bmatrix}-1\\-1\end{bmatrix}\right)\right]$$

$$g_2(x) = p(x|\omega_2) = exp\left[-\frac{1}{2}\left(x - \begin{bmatrix}1\\1\end{bmatrix}\right)^t \left(x - \begin{bmatrix}1\\1\end{bmatrix}\right)\right]$$

$$\begin{aligned} g_3(x) &= p(x|\omega_3) \\ &= \frac{1}{2}\left(exp\left[-\frac{1}{2}\left(x - \begin{bmatrix}0.5\\0.5\end{bmatrix}\right)^t \left(x - \begin{bmatrix}0.5\\0.5\end{bmatrix}\right)\right]\right) + \frac{1}{2}\left(exp\left[-\frac{1}{2}\left(x - \begin{bmatrix}-0.5\\-0.5\end{bmatrix}\right)^t \left(x - \begin{bmatrix}-0.5\\-0.5\end{bmatrix}\right)\right]\right) \end{aligned}$$

Note that $(2\pi)^{d/2}$ and $\Sigma = I$ are the same for all g_i's

The Bayes decision rule will be as follow:

$$i^* = argmax_i\{g_i(x)\}_{i=1}^3$$

Where i^* represents the class that x will be assigned.

After solving using the developed code (q3.ipynb), the results are as follows:

$$g_1(x) = 0.480902$$
$$g_2(x) = 0.720327$$
$$g_3(x) = 0.962268$$

Thus, the point $x = (0.1, 0.1)^t$ will be assigned to class ω_3.

4. [10 points] Suppose we have two normal distributions, corresponding to two classes (ω_1 and ω_2) with the same covariances but different means: $N(\boldsymbol{\mu_1}, \boldsymbol{\Sigma})$ and $N(\boldsymbol{\mu_2}, \boldsymbol{\Sigma})$. In terms of their prior probabilities $P(\omega_1)$ and $P(\omega_2)$, state the condition under which the Bayes decision boundary will *not* pass anywhere *between* the two means. You can assume a 0-1 loss function.
Answer
With the hint from the book, the Bayes decision boundary does not pass between the two means (i.e., μ_1 and μ_2) if and only $w^t(\mu_1 - x_0)$ and $w^t(\mu_2 - x_0)$ have the same sign. This can be shown as follows:

$$w^t(\mu_1 - x_0) > 0 and w^t(\mu_2 - x_0) > 0$$

or

$$w^t(\mu_1 - x_0) < 0 and w^t(\mu_2 - x_0) < 0$$

These conditions are equivalents to (0-1 loss function is applied)

$$w^t(\mu_1 - x_0) = \frac{1}{2}(\mu_1 - \mu_2)^t \Sigma^{-1}(\mu_1 - \mu_2) - \ln[\frac{P(w_1)}{P(w_2)}]$$
$$w^t(\mu_2 - x_0) = -\frac{1}{2}(\mu_1 - \mu_2)^t \Sigma^{-1}(\mu_1 - \mu_2) - \ln[\frac{P(w_1)}{P(w_2)}]$$

Consider this case, where

$$w^t(\mu_1 - x_0) > 0 and w^t(\mu_2 - x_0) > 0$$

This implies

$$(\mu_1 - \mu_2)^t \Sigma^{-1}(\mu_1 - \mu_2) > 2\ln[\frac{P(w_1)}{P(w_2)}]$$

and

$$(\mu_1 - \mu_2)^t \Sigma^{-1}(\mu_1 - \mu_2) < -2\ln[\frac{P(w_1)}{P(w_2)}]$$

Similarly to the case where

$$w^t(\mu_1 - x_0) < 0 and w^t(\mu_2 - x_0) < 0$$

This implies

$$(\mu_1 - \mu_2)^t \Sigma^{-1}(\mu_1 - \mu_2) < 2\ln[\frac{P(w_1)}{P(w_2)}]$$

and

$$(\mu_1 - \mu_2)^t \Sigma^{-1}(\mu_1 - \mu_2) > -2\ln[\frac{P(w_1)}{P(w_2)}]$$

As demonstrated above, we can conclude that there are two cases (in terms of their prior probabilities) where the Bayes decision boundary does not pass between the two means as follows:
Case1: $P(w_1) > P(w_2)$
The condition is $(\mu_1 - \mu_2)^t \Sigma^{-1}(\mu_1 - \mu_2) < 2\ln[\frac{P(w_1)}{P(w_2)}]$ and this ensures $w^t(\mu_1 - x_0) < 0$ and $w^t(\mu_2 - x_0) < 0$
Case2: $P(w_1) <= P(w_2)$

The condition is $(\mu_1-\mu_2)^t\Sigma^{-1}(\mu_1-\mu_2) < 2\ln[\frac{P(w_1)}{P(w_2)}]$ and this ensures $w^t(\mu_1-x_0) > 0$ and $w^t(\mu_2-x_0) > 0$

5. Consider a two-category classification problem involving two-dimensional feature vectors of the form $\boldsymbol{x} = (x_1, x_2)^t$. The two categories are ω_1 and ω_2, and

$$p(\boldsymbol{x} \mid \omega_1) \sim N\left((4,4)^t, I\right),$$

$$p(\boldsymbol{x} \mid \omega_2) \sim N\left((0,0)^t, I\right),$$

$$P(\omega_1) = P(\omega_2) = \frac{1}{2}.$$

(a) [5 points] Calculate the Bayes decision boundary and write down the Bayes decision rule assuming a 0-1 loss function. The Bayes decision rule has to be written in terms of the Bayes decision boundary.

Answer:

The decision boundary is found by equating the posterior probabilities. But since the problem requires to write down the decision rule as well, we will proceed as follows. Assign x to class ω_1, if:

$$P(\omega_1|x) > P(\omega_2|x)$$
$$p(x|\omega_1) > p(x|\omega_2)$$
$$\frac{-1}{2}(x-\mu_1)^t \sum_1^{-1}(x-\mu_2) > \frac{-1}{2}(x-\mu_2)^t \sum_2^{-1}(x-\mu_2)$$
$$\frac{-1}{2}\left(x-\begin{bmatrix}4\\4\end{bmatrix}\right)^t \sum_1^{-1}\left(x-\begin{bmatrix}4\\4\end{bmatrix}\right) > \frac{-1}{2}\left(x-\begin{bmatrix}0\\0\end{bmatrix}\right)^t \sum_2^{-1}\left(x-\begin{bmatrix}0\\0\end{bmatrix}\right)$$
$$\frac{-1}{2}[x_1-4, x_2-4]\left[\begin{bmatrix}x_1-4\\x_2-4\end{bmatrix}\right] > \frac{-1}{2}[x_1, x_2]\left[\begin{bmatrix}x_1\\x_2\end{bmatrix}\right]$$
$$\frac{-1}{2}\left[(x_1-4)^2+(x_2-4)^2\right] > \frac{-1}{2}\left(x_1^2+x_2^2\right)$$
$$x_1^2-8x_1+16+x_2^2-8x_2+16 < x_1^2+x_2^2$$
$$-8(x_1+x_2)+32 < 0$$
$$-8(x_1+x_2) < -32$$
$$x_1+x_2 > 4$$

Thus, the decision rule is:
Decide ω_1 if $x_1+x_2 > 4$
Decide ω_2 if $x_1+x_2 < 4$
So the equation for the decision boundary is $x_1+x_2-4=0$.

(b) [10 points] What are the Bhattacharyya and Chernoff theoretical bounds on the probability of misclassification, P(error)?

Answer:

After writing the code (q5.ipynb) to find the bounds on the probability of misclassification. The results obtained as follows:

At $\beta^* = 0.50$, the Chernoff bound on the error is $P(error) \leqslant 0.00915782$.
And at $\beta = \frac{1}{2}$, the Bhattacharrya bound on the error is $P(error) \leqslant 0.00915782$.

(c) [5 points] Generate $n = 25$ test patterns from *each* of the two class-conditional densities and plot them in a two-dimensional feature space using different markers for the two categories (if you are using matlab, then the *mvnrnd* function can be used to generate these patterns). Draw the Bayes decision boundary on this plot for visualization purposes.

Answer:

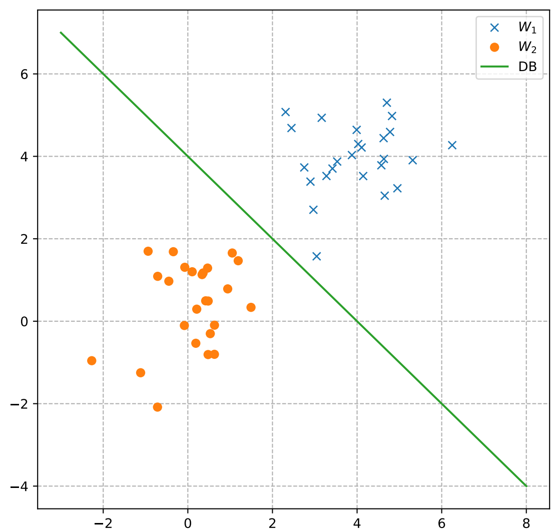

Figure 4: 25 test patterns generated from each of the two-class conditional densities and the Bayes decision boundary.

(d) [5 points] What is the confusion matrix and empirical error rate when classifying the generated patterns using the Bayes decision rule?

Answer:

The empirical error rate is 0.0

The confusion matrix C is $C = \begin{bmatrix} 25 & 0 \\ 0 & 25 \end{bmatrix}$

Note that $rows$ represents the true label and the $columns$ represents the predicted label.

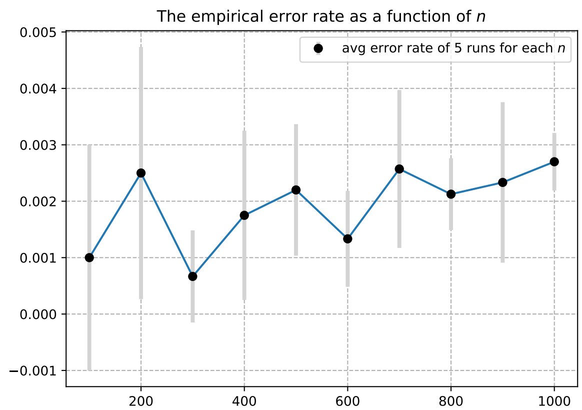

Figure 5: The empirical error rate as a function of n, where each n, the test patterns are generated 5 times.

(e) [10 points] Can the empirical error rate exceed the theoretical bounds on the probability of misclassification? Why or why not? (Hint: To help answer this question, do the following. Compute the empirical error rate by increasing the value of n from 100 to 1000 in steps of 100. Plot the empirical error rate as a function of n. What do you observe? For each value of n, you may also want to generate the test patterns multiple times (say 3 times), to determine how the empirical error rate changes for the same value of n. *Note that n is the number of test patterns from each class.*)

Answer:
The empirical error rate as a function of n is shown in Figure 5, where n is the number of test patterns from each class. We see that the empirical error rate has fluctuated. However, the obtained error rate from multiple runs is not exceeded the theoretical bounds, found in the previous questions. From this experiment, one could say that the theoretical bounds provide us the upper bound on the error.

6. [15 points] Consider a set of n i.i.d. samples (one-dimensional training patterns), $D = \{x_1, x_2, \dots x_n\}$, that are drawn from the following distribution (Rayleigh distribution):

$$p(x|\theta) = 2\theta x e^{-\theta x^2}, \quad x \geq 0, \theta > 0.$$

(a) Derive the maximum likelihood estimate (MLE) of θ, i.e., $\widehat{\theta}_{mle}$.
Answer:

$$
\begin{aligned}
p(x|\theta) &= 2\theta x_k e^{-\theta x_k^2} \\
\ln p(x|\theta) &= 2\ln\theta + \ln x_k - \theta x_k^2 \\
\frac{\partial \ln}{\partial \theta} p(x|\theta) &= \frac{1}{\theta} - x_k^2 \\
\sum_{k=1}^{n} \frac{\partial \ln}{\partial \theta} p(x|\theta) &= \sum_{k=1}^{n}\left[\frac{1}{\theta} - x_k^2\right] \\
\frac{n}{\theta} - \sum_{k=1}^{n} x_k^2 &= 0 \\
\frac{n}{\theta} &= \sum_{k=1}^{n} x_k^2 \\
\theta &= \frac{n}{\sum_{k=1}^{n} x_k^2}
\end{aligned}
$$

(b) Consider a set of 1000 training patterns that can be accessed **here**. Plot the normalized histogram of the training patterns. In the same graph, plot the distribution, $p(x)$, after estimating $\widehat{\theta}_{mle}$ from these training patterns.

Answer:

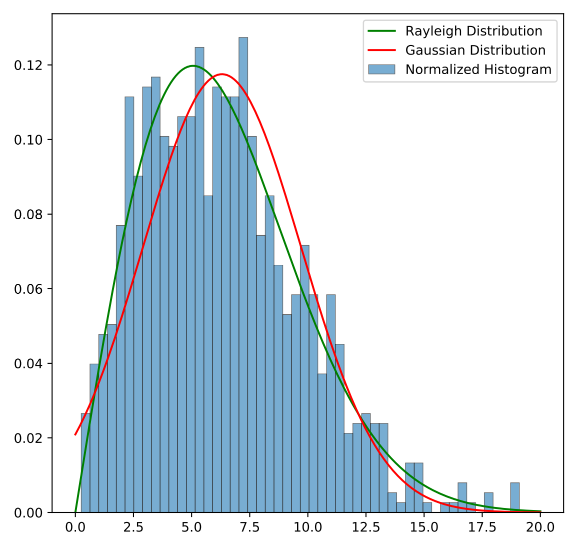

Figure 6: The distribution of Rayleigh and Gaussian

(c) Using the same training patterns, determine the MLE estimates for the mean and variance of a Gaussian distribution. In this case, you can use the MLE formulae directly. Plot the resulting Gaussian distribution on the same graph as above.

(d) Comment on which of the two distributions better “fit" the training data.

Answer:

As shown in Figure 7, the two distributions are slightly different. We see that both the Rayleigh distribution and the Gaussian distribution fit the training data well. Thus, it quite challenges here to completely conclude which of the two distributions is better fit the training data.

7. [30 points] Consider a two-category (ω_1 and ω_2) classification problem with equal priors. Each feature is a two-dimensional vector $\boldsymbol{x} = (x_1, x_2)^t$. The *true* class-conditional densities are:
$p(\boldsymbol{x}|\omega_1) \sim N(\boldsymbol{\mu_1} = [0,0]^t, \Sigma_1 = I)$,
$p(\boldsymbol{x}|\omega_2) \sim N(\boldsymbol{\mu_2} = [5,5]^t, \Sigma_2 = I)$.

Generate n=50 bivariate *random* training samples from each of the two densities.

(a) Write a program to find the values for the maximum likelihood estimates of $\boldsymbol{\mu_1}$, $\boldsymbol{\mu_2}$, Σ_1, and Σ_2 using these training samples (see page 89, use equations (18) and (19)).

Answer:

$\mu_1 = (0.13770049, -0.01653479)^t$
$\mu_2 = (5.3156104, 4.98997916)^t$

$$\Sigma_1 = \begin{bmatrix} 0.65914663 & -0.17948653 \\ -0.17948653 & 0.89596215 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 0.80115575 & -0.04999757 \\ -0.04999757 & 0.88305507 \end{bmatrix}$$

(b) Compute the Bayes decision boundary using the *estimated* parameters and plot it along with the training samples. What is the empirical error rate on the training samples?

Answer:

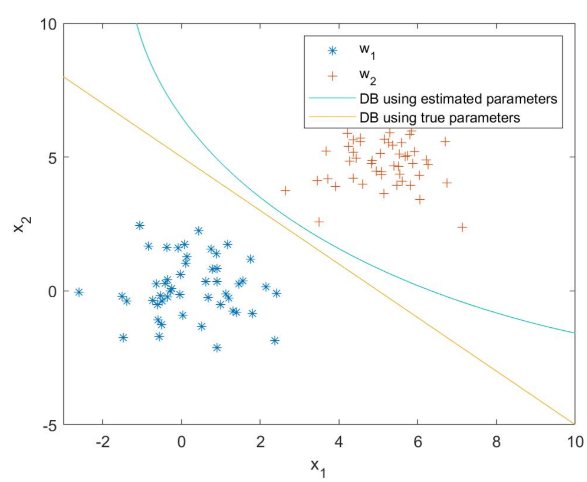

Figure 7: The Bayes decision boundaries, N = 50

Since both classes are Gaussian, and equal priors,

$$g_i(x) = -\frac{1}{2}\ln(|\Sigma_i|) - \frac{1}{2}(x-\mu_i)^t\Sigma^{-1}(x-\mu_i)$$

The decision boundary between the two classes is found by equating the discriminant functions. Thus,

$$g_1(x) = g_2(x)$$
$$-\frac{1}{2}\ln(|\Sigma_1|) - \frac{1}{2}(x-\mu_1)^t\Sigma^{-1}(x-\mu_1) = -\frac{1}{2}\ln(|\Sigma_2|) - \frac{1}{2}(x-\mu_2)^t\Sigma^{-1}(x-\mu_2)$$

Substitute with the obtained estimated parameters from a.) using developed Matlab code

The empirical error rate is 0.0.

(c) Compute the Bayes decision boundary using the *true* parameters and plot it on the same graph. What is the empirical error rate on the training samples?

Answer: A pattern x is assigned to the class w_1 if

$$g_1(x) > g_2(x),$$

where the discriminant function for each class w_i is given as follows:

$$g_i(x) = -\frac{1}{2}\ln(|\Sigma_i|) - \frac{1}{2}(x-\mu_i)^t\Sigma^{-1}(x-\mu_i)$$

The decision rule obtained is as follows:
Decide w_1 if $x_1 + x_2 - 5 < 0$
Decide w_2 if $x_1 + x_2 - 5 > 0$
Thus, the equation for the decision boundary is $x_1 + x_2 - 5 = 0$
The empirical error rate is 0.0.

(d) Repeat (a) - (c) after generating n=500 and n=50,000 random training samples from each of the two densities. How do the estimated parameters and the empirical error rate change in (a) and (b) when the number of representative training samples increases?

Answer:

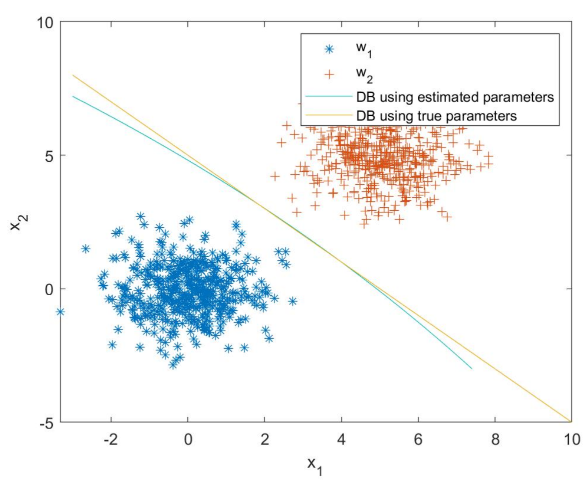

Figure 8: The Bayes decision boundaries, N = 500

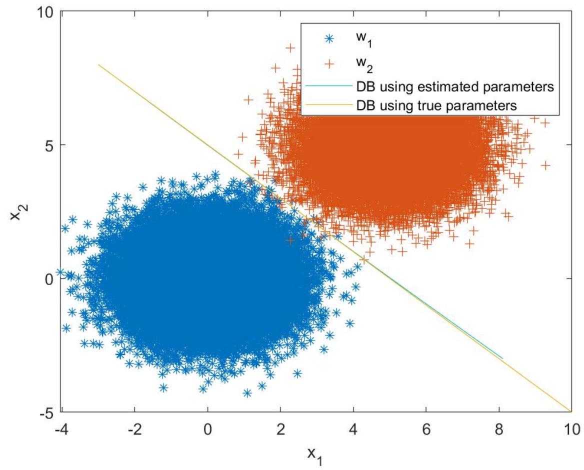

Figure 9: The Bayes decision boundaries, N = 50,000

N = 500
$\mu_1 = (-0.01314941, -0.08255046)^t$
$\mu_2 = (5.05406512, 4.92452505)^t$
$\Sigma_1 = \begin{bmatrix} 1.00680125 & -0.07324719 \\ -0.07324719 & 1.0083165 \end{bmatrix}$
$\Sigma_2 = \begin{bmatrix} 0.9551595 & 0.02581498 \\ 0.02581498 & 0.93461263 \end{bmatrix}$
The empirical error rate (estimated parameters) is 0.0
The empirical error rate is (true parameters) 0.0

N = 50,000
$\mu_1 = (-0.00324084, -0.00136667)^t$
$\mu_2 = (5.00267322, 4.99446753)^t$
$\Sigma_1 = \begin{bmatrix} 0.9905562 & 0.00561943 \\ 0.00561943 & 0.99628715 \end{bmatrix}$
$\Sigma_2 = \begin{bmatrix} 0.99510007 & -0.00224908 \\ -0.00224908 & 0.9900453 \end{bmatrix}$
The empirical error rate (estimated parameters) is 0.00042
The empirical error rate (true parameters) is 0.00042

From the obtained results, we can see that when the number of training samples increase, the estimated parameters (i.e., $\boldsymbol{\mu_1}$, $\boldsymbol{\mu_2}$, Σ_1, and Σ_2) are closer to the true parameters, leading to a decision boundary (DB) that is close to the DB obtained from the true parameters. Interestingly, for n = 50 and n=500, the empirical error rates are the same on the training samples obtained from the Bayes decision boundary using the estimated parameters. For n=50,000, the empirical error rate increase to 0.00042.

8. [20 points] The **iris (flower) dataset** consists of 150 4-dimensional patterns (i.e., feature vectors) belonging to three classes (setosa=1, versicolor=2, and virginica=3). There are 50 patterns per class. The 4 features correspond to sepal length in cm (x_1), sepal width in cm (x_2), petal length in cm (x_3),

and petal width in cm (x_4). Note that the class labels are indicated at the end of every pattern.

Assume that each class can be modeled by a multivariate Gaussian density, i.e., $p(\boldsymbol{x}|\omega_i) \sim N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, $i = 1, 2, 3$. Write a program to design a Bayes classifier and test it by following the steps below:

(a) Train the classifier: Using the first 25 patterns of each class (training data), compute $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$, $i = 1, 2, 3$. **Report** these values.

Answer:

$\mu_1 = (5.028, 3.48, 1.46, 0.248)^t$

$$\Sigma_1 = \begin{bmatrix} 0.154016 & 0.11336 & 0.02312 & 0.018656 \\ 0.11336 & 0.1304 & 0.006 & 0.02136 \\ 0.02312 & 0.006 & 0.0376 & 0.00632 \\ 0.018656 & 0.02136 & 0.00632 & 0.010496 \end{bmatrix}$$

$\mu_2 = (6.012, 2.776, 4.312, 1.344)^t$

$$\Sigma_2 = \begin{bmatrix} 0.288256 & 0.105088 & 0.179056 & 0.049872 \\ 0.105088 & 0.119424 & 0.085088 & 0.044656 \\ 0.179056 & 0.085088 & 0.189056 & 0.061472 \\ 0.049872 & 0.044656 & 0.061472 & 0.040864 \end{bmatrix}$$

$\mu_3 = (6.576, 2.928, 5.64, 2.044)^t$

$$\Sigma_3 = \begin{bmatrix} 0.503424 & 0.116672 & 0.41496 & 0.059456 \\ 0.116672 & 0.125216 & 0.09488 & 0.057968 \\ 0.41496 & 0.09488 & 0.4008 & 0.06464 \\ 0.059456 & 0.057968 & 0.06464 & 0.062464 \end{bmatrix}$$

(b) Design the Bayes classifier: Assuming that the three classes are equally probable and a 0-1 loss function, write a program that inputs a 4-dimensional pattern $\boldsymbol{x}$ and assigns it to one of the three classes based on the maximum posterior rule, i.e., assign $\boldsymbol{x}$ to ω_j if,

$$j = \arg \max_{i=1,2,3} \{P(\omega_i|\boldsymbol{x})\}.$$

Answer:

The code for this Bayes classifier is at q8.ipynb

(c) Test the classifier: Classify the remaining 25 patterns of each class (test data) using the Bayes classifier constructed above and report the confusion matrix for this three-class problem. What is the empirical error rate on the test set?

Answer:

The confusion matrix C is $\begin{bmatrix} 25 & 0 & 0 \\ 0 & 24 & 1 \\ 0 & 1 & 24 \end{bmatrix}$

Note that *rows* represents the true label and the *columns* represents the predicted label.

The empirical error rate on the test set is 0.02666666666666667

A Appendices

1. Python code in Jupyter Notebook (with outputs) for Q2 is located at notebooks/q2.ipynb

   2. Python code in Jupyter Notebook (with outputs) for Q3 is located at notebooks/q3.ipynb
   3. Python code in Jupyter Notebook (with outputs) for Q5 is located at notebooks/q5.ipynb
   4. Python code in Jupyter Notebook (with outputs) for Q6 is located at notebooks/q6.ipynb
   5. Python code in Jupyter Notebook (with outputs) for Q7 is located at notebooks/q7.ipynb
   6. Matlab code (with outputs) for Q8 is located at notebooks/q7.m

7. https://www.cifasis-conicet.gov.ar/granitto/ECI2014/book$_{D}uda.pdf$