IITM-CS5691 : Pattern Recognition and Machine Learning — Release Date: August 30, 2023
Assignment 1 — Due Date : Wednesday, 27 September 2023, 11:59 PM

Roll No: CS23E001 — Name: Shuvrajeet Das
Collaborators (if any):
References/sources (if any):

- Use LaTeX to write-up your solutions (in the solution blocks of the source LaTeX file of this assignment), and submit the resulting pdf files (one per question) at Crowdmark by the due date. (Note: **No late submissions** will be allowed, other than one-day late submission with 10% penalty or four-day late submission with 30% penalty! Instructions to join Crowdmark and submit your solution to each question within Crowdmark **TBA** later).
- For the programming question, please submit your code (rollno.ipynb file and rollno.py file in rollno.zip) directly in moodle, but provide your results/answers (including Jupyter notebook **with output**) in the pdf file you upload to Crowdmark.
- Collaboration is encouraged, but all write-ups must be done individually and independently, and mention your collaborator(s) if any. Same rules apply for codes written for any programming assignments (i.e., write your own code; we will run plagiarism checks on codes).
- If you have referred a book or any other online material or LLMs (Large Language Models like ChatGPT) for obtaining a solution, please cite the source. Again don't copy the source *as is* - you may use the source to understand the solution, but write-up the solution in your own words (this also means that you cannot copy-paste the solution from LLMs!). Please be advised that *the lesser your reliance on online materials or LLMs* for answering the questions, *the more your understanding* of the concepts will be and *the more prepared you will be for the course exams*.
- Points will be awarded based on how clear, concise and rigorous your solutions are, and how correct your answer is. The weightage of this assignment is 12% towards the overall course grade.

1. (8 points) [GETTING YOUR BASICS RIGHT!]

   (a) (5 points) Let a random vector X follow a bivariate Gaussian distribution with mean $\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ and covariance matrix $\Sigma = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, i.e., $X \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} a & b \\ c & d \end{bmatrix}\right)$. Then, use the pdf (probability density function) of X to:
   Find the distribution of (i) $X_2|X_1 = x_1$ and (ii) $X_1|X_2 = x_2$, and use them to (iii) find the permissible values of a, b, c, and d.
   (Hint: You can use the same approach of "completing the squares" seen in class).

(b) (2 points) Consider the function $f(x) = x_1^2 + x_2^2 + x_1x_2$, and a point $v = \begin{bmatrix}3\\5\end{bmatrix}$. Find the linear approximation of f around v (i.e., $L_v[f](y)$), and show that the graph of this approximation is a hyperplane in $\mathbb{R}^3$.

(c) (1 point) Which of these statements are true about two random variables X and Y defined on the same probability space?

(i) If X, Y are independent, then X, Y are uncorrelated ($Cov(X, Y) = 0$).

(ii) If X, Y are uncorrelated, then X, Y are independent.

(iii) If X, Y are uncorrelated and follow a bivariate normal distribution, then X, Y are independent.

(iv) None of the above.

Solution: The solution of question(a)

Given the conditions, $\mu = \begin{bmatrix}0\\0\end{bmatrix}$ and $\Sigma = \begin{bmatrix}a & b\\c & d\end{bmatrix}$

from this we can get, $\Sigma^{-1} = \frac{1}{ad-bc}\begin{bmatrix}d & -b\\-c & a\end{bmatrix}$ and $|\Sigma| = (ad - bc)$

The multivariate Gaussian distribution can defined as:-

$$f_{\mathbf{X}}(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}|\mathbf{\Sigma}|^{1/2}} \cdot \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

Putting all the necessary values we get:

$$= \frac{1}{(2\pi)(ad-bc)^{1/2}} \exp\left(-\frac{1}{2(ad-bc)}\begin{bmatrix}x_1\\x_2\end{bmatrix}^T\begin{bmatrix}d & -b\\-c & a\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix})\right)$$

$$= \frac{1}{(2\pi)(ad-bc)^{1/2}} \exp\left(-\frac{1}{2(ad-bc)}\begin{bmatrix}x_1d - x_2c & -x_1b + x_2a\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix}\right)$$

$$= \frac{1}{(2\pi)(ad-bc)^{1/2}} \exp\left(-\frac{1}{2(ad-bc)}(x_1^2d - x_1x_2c - x_1x_2b + x_2^2a)\right)$$

$$= \frac{1}{(2\pi)(ad-bc)^{1/2}} \exp\left(-\frac{1}{2(ad-bc)}(x_1^2d - x_1x_2(c+b) + x_2^2a)\right)$$

For question (1)$P(X_2|X_1 = x_1)$ is

$$= \frac{1}{(2\pi)(ad-bc)^{1/2}} \exp\left(-\frac{a}{2(ad-bc)}\left(\frac{x_1^2d}{a} - 2x_1x_2\left(\frac{c+b}{2a}\right) + x_2^2\right)\right)$$

Let's solve the inner part in the exponent,

$$= \frac{-a}{2(ad-bc)}\left(\frac{x_1^2 d}{a} - 2x_1x_2\left(\frac{c+b}{2a}\right) + x_2^2\right)$$
$$= \frac{-a}{2(ad-bc)}\left(\frac{x_1^2 d}{a} - \left(\frac{c+b}{2a}\right)^2 x_1^2 + \left(\frac{c+b}{2a}\right)^2 x_1^2 - 2x_1x_2\left(\frac{c+b}{2a}\right) + x_2^2\right)$$
$$= \frac{-a}{2(ad-bc)}\left(\frac{x_1^2 d}{a} - \left(\frac{c+b}{2a}\right)^2 x_1^2 + \left(\left(\frac{c+b}{2a}\right)x_1 - x_2\right)^2\right)$$
$$= \frac{-a}{2(ad-bc)}\left(\left(\frac{4ad-(c+b)^2}{4a^2}\right)x_1^2 + \left(\left(\frac{c+b}{2a}\right)x_1 - x_2\right)^2\right)$$
$$= \frac{-a}{2(ad-bc)}\left(\left(\frac{4ad-(c+b)^2}{4a^2}\right)x_1^2 + \frac{-a}{2(ad-bc)}\left(\left(\frac{c+b}{2a}\right)x_1 - x_2\right)^2\right.$$

Simply filling the equation again we get, a product of two quantities,

$$\frac{1}{(2\pi)^{1/2}(ad-bc)^{1/2}}\exp\left(\frac{-a}{2(ad-bc)}\left(\left(\frac{c+b}{2a}\right)x_1 - x_2\right)^2\right)$$

and

$$\frac{1}{(2\pi)^{1/2}}\exp\left(\frac{-a}{2(ad-bc)}\left(\frac{4ad-(c+b)^2}{4a^2}\right)x_1^2\right)$$

For question (2)$P(X_1|X_2 = x_2)$ is

$$= \frac{1}{(2\pi)(ad-bc)^{1/2}}\exp\left(-\frac{d}{2(ad-bc)}\left(x_1^2 - 2x_1x_2\left(\frac{c+b}{2d}\right) + \frac{x_2^2 a}{d}\right)\right)$$

Let's solve the inner part in the exponent,

$$= -\frac{d}{2(ad-bc)}\left(x_1^2 - 2x_1x_2\left(\frac{c+b}{2d}\right) + \frac{x_2^2 a}{d}\right)$$
$$= \frac{-d}{2(ad-bc)}\left(x_1^2 - 2x_1x_2\left(\frac{c+b}{2d}\right) + \left(\frac{c+b}{2d}\right)^2 x_2^2 - \left(\frac{c+b}{2d}\right)^2 x_2^2 + \frac{x_2^2 a}{d}\right)$$
$$= \frac{-d}{2(ad-bc)}\left(\left(x_1 - \left(\frac{c+b}{2d}\right)x_2\right)^2 - \left(\frac{c+b}{2d}\right)^2 x_2^2 + \frac{x_2^2 a}{d}\right)$$
$$= \frac{-d}{2(ad-bc)}\left(\left(x_1 - \left(\frac{c+b}{2d}\right)x_2\right)^2 + \left(\frac{4ad-(c+b)^2}{4d^2}\right)x_2^2\right)$$
$$= \frac{-d}{2(ad-bc)}\left(x_1 - \left(\frac{c+b}{2d}\right)x_2\right)^2 + \frac{-d}{2(ad-bc)}\left(\frac{4ad-(c+b)^2}{4d^2}\right)x_2^2$$

Simply filling the equation again we get, a product of two quantities,

$$\frac{1}{(2\pi)^{1/2}(ad-bc)^{1/2}}\exp\left(\frac{-d}{2(ad-bc)}\left(x_1-\left(\frac{c+b}{2d}\right)x_2\right)^2\right)$$

and

$$\frac{1}{(2\pi)^{1/2}}\exp\left(\frac{-1}{2(ad-bc)}\left(\frac{4ad-(c+b)^2}{4d^2}\right)x_2^2\right)$$

we can say that $d>0$, $ad-bc>0$, $a>0$ also $a=d=1$ filling up the values we get,

$$P(X_1|X_2)=\frac{1}{(2\pi)^{1/2}(1-bc)^{1/2}}\exp\left(\frac{-1}{2(1-bc)}\left(x_1-\left(\frac{c+b}{2}\right)x_2\right)^2\right)$$

$$P(X_2)=\frac{1}{(2\pi)^{1/2}}\exp\left(\frac{-1}{2(1-bc)}\left(\frac{4-(c+b)^2}{4}\right)x_2^2\right)$$

$$P(X_2|X_1)=\frac{1}{(2\pi)^{1/2}(1-bc)^{1/2}}\exp\left(\frac{-1}{2(1-bc)}\left(\left(\frac{c+b}{2}\right)x_1-x_2\right)^2\right)$$

$$P(X_2)=\frac{1}{(2\pi)^{1/2}}\exp\left(\frac{-1}{2(1-bc)}\left(\frac{4-(c+b)^2}{4}\right)x_1^2\right)$$

we also get $1-bc>0$

The solution of question (b)

we know from the taylor series expnasion,

$$f(x)=f(a)+\frac{f'(a)}{1!}(x-a)+\frac{f''(a)}{2!}(x-a)^2+\frac{f'''(a)}{3!}(x-a)^3+\cdots$$

The linear approximation of a function at a given point is its first-order Taylor series expansion around that point. To find the linear approximation of the function $f(x_1,x_2)=x_1^2+x_2^2+x_1x_2$, we need to choose a point (a, b) around which we want to approximate the function. Let's choose the point (a, b).

The linear approximation of $f(x_1,x_2)$ at the point (a, b) is given by:

$$L(x_1,x2)=f(a,b)+\frac{\partial f(a,b)}{\partial x_1}(x_1-a)+\frac{\partial f(a,b)}{\partial x_2}(x_2-b)$$

To find the linear approximation, we need to calculate the partial derivatives of $f(x_1,x_2)$ with respect to x and y. Here are the derivatives:

$$\frac{\partial f(x_1,x_2)}{\partial x_1}=2x_1+x_2\quad\frac{\partial f(x_1,x_2)}{\partial x_2}=2x_2+x_1$$

Now, we can plug these derivatives into the linear approximation formula:

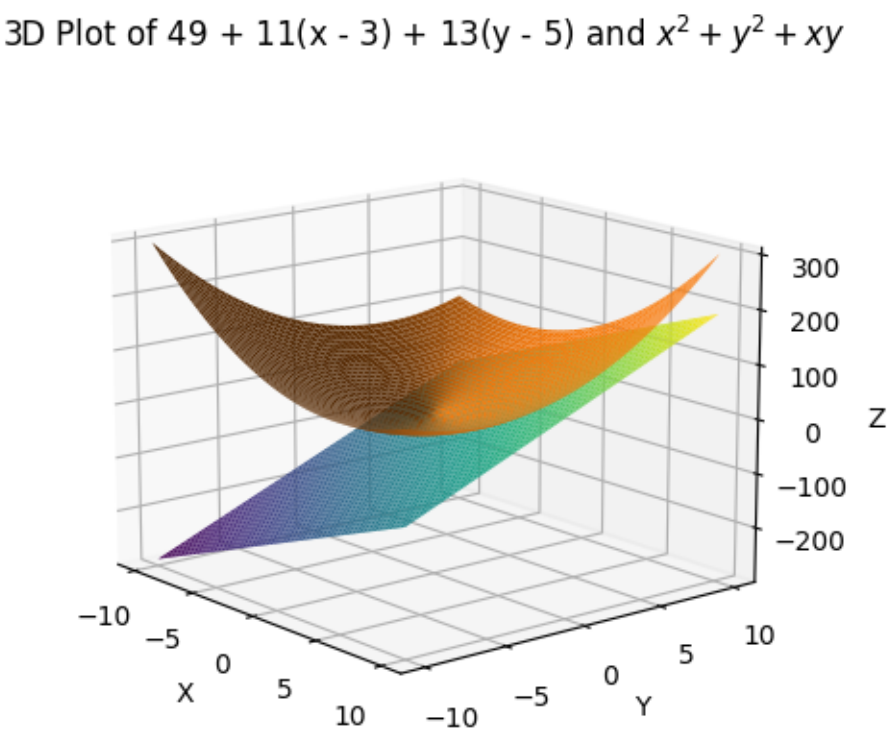
$$L(x_1, x_2) = (a^2 + b^2 + ab) + (2a + b)(x_1 - a) + (2b + a)(x_2 - b)$$

So, the linear approximation of $f(x_1, x_2) = x_1^2 + x_2^2 + x_1x_2$ around the point (a, b) is:

$$L(x_1, x_2) = a^2 + b^2 + ab + (2a + b)(x_1 - a) + (2b + a)(x_2 - b)$$

Filling up all the necessary values we get

$$L(x_1, x_2) = 49 + 11(x - 3) + 13(y - 5)$$



The solution of question (c)

The statement no (i) and (iii) are correct.

IITM-CS5691 : Pattern Recognition and Machine Learning
Assignment 1

Release Date: August 30, 2023
Due Date : September 14, 2023, 23:59

Roll No: CS23E001
Name: Shuvrajeet Das
Collaborators (if any):
References/sources (if any):

- Use LaTeX to write-up your solutions (in the solution blocks of the source LaTeX file of this assignment), and submit the resulting pdf files (one per question) at Crowdmark by the due date. (Note: **No late submissions** will be allowed, other than one-day late submission with 10% penalty or four-day late submission with 30% penalty! Instructions to join Crowdmark and submit your solution to each question within Crowdmark **TBA** later).
- For the programming question, please submit your code (rollno.ipynb file and rollno.py file in rollno.zip) directly in moodle, but provide your results/answers (including Jupyter notebook **with output**) in the pdf file you upload to Crowdmark.
- Collaboration is encouraged, but all write-ups must be done individually and independently, and mention your collaborator(s) if any. Same rules apply for codes written for any programming assignments (i.e., write your own code; we will run plagiarism checks on codes).
- If you have referred a book or any other online material or LLMs (Large Language Models like ChatGPT) for obtaining a solution, please cite the source. Again don't copy the source *as is* - you may use the source to understand the solution, but write-up the solution in your own words (this also means that you cannot copy-paste the solution from LLMs!). Please be advised that *the lesser your reliance on online materials or LLMs* for answering the questions, *the more your understanding* of the concepts will be and *the more prepared you will be for the course exams*.
- Points will be awarded based on how clear, concise and rigorous your solutions are, and how correct your answer is. The weightage of this assignment is 12% towards the overall course grade.

1. (8 points) [EXPLORING MAXIMUM LIKELIHOOD ESTIMATION]
Consider the i.i.d data $\mathbf{X} = \{x_i\}_{i=1}^n$, such that each $x_i \sim \mathcal{N}(\mu, \sigma^2)$. We have seen ML estimates of μ, σ^2 in class by setting the gradient to zero.
   (a) (4 points) How can you argue that the stationary points so obtained are indeed global maxima of the likelihood function?
   (b) (4 points) Derive the bias of the MLE of μ, σ^2.

Solution: The solution of question (a)

The normal distribution can be written as $f(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$

Now, we can describe, $\mathcal{L}(X)$ as

$$\begin{aligned}\mathcal{L}(X) &= \prod_{i=1}^{n} f(x_i, \mu, \sigma) \\ &= \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} exp\left(\frac{1}{2\sigma^2}(x_i - \mu)^2\right) \\ &= \left(\frac{1}{\sigma\sqrt{2\pi}}\right)^n exp\left(\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2\right)\end{aligned}$$

Now for log-likelihood $l(x)$ to be defined by $log(\mathcal{L}(x))$

$$\begin{aligned}l(x) &= log(\mathcal{L}(x)) \\ &= nlog(\sigma) - nlog(\sqrt{2\pi}) = \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2\end{aligned}$$

Differentiating with μ and σ,

$$\begin{aligned}\frac{\partial l}{\partial \mu} &= \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2(-1) = 0 \\ &\implies \sum_{i=1}^{n} x_i - \sum_{i=1}^{n}\mu = 0 \\ &\implies \sum_{i=1}^{n} x_i - n\mu = 0 \\ &\implies \mu = \frac{\sum_{i=1}^{n} x_i}{n}\end{aligned}$$

$$\begin{aligned}\frac{\partial l}{\partial \sigma} &= \frac{n}{\sigma} - (-2)\frac{1}{2\sigma^3}\sum_{i=1}^{n}(x_i - \mu)^2 = 0 \\ &\implies -n\sigma^2 + \sum_{i=1}^{n}(x_i - \mu)^2 = 0 \\ &\implies \sum_{i=1}^{n}(x_i - \mu)^2 = n\sigma^2 \\ &\implies \sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2\end{aligned}$$

Thus the problem can be optimized as a minimization problem as least square forming a convex function to get a global minima

The solution of question (b)

Mathematically, the bias (B) of an estimator $\hat{\theta}$ estimating a parameter θ is defined as:

$$B(\hat{\theta}) = E[\hat{\theta}] - \theta$$

To derive the bias of the Maximum Likelihood Estimators (MLE) for μ and σ^2, we need to calculate the expected values of the MLEs and compare them to the true values of μ and σ^2.

Let's start with the MLE for μ, denoted as $\hat{\mu}$. The bias, $B(\hat{\mu})$, is defined as:

$$B(\hat{\mu}) = E[\hat{\mu}] - \mu$$

To calculate the expected value of the MLE for μ, we consider that $\hat{\mu}$ follows a normal distribution with mean μ and variance $\frac{\sigma^2}{n}$ (this is a known property of the MLE for the mean of a normal distribution):

$$E[\hat{\mu}] = \mu$$

Therefore, the bias of the MLE for μ is:

$$B(\hat{\mu}) = \mu - \mu = 0$$

Now, let's derive the bias for the MLE of σ^2, denoted as $\hat{\sigma}^2$. The bias, $B(\hat{\sigma}^2)$, is defined as:

$$B(\hat{\sigma}^2) = E[\hat{\sigma}^2] - \sigma^2$$

The MLE for σ^2 is:

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu})^2$$

To calculate the expected value of $\hat{\sigma}^2$, we can use the properties of sample variances for a normal distribution:

$$E[\hat{\sigma}^2] = \frac{n-1}{n}\sigma^2$$

Therefore, the bias of the MLE for σ^2 is:

$$B(\hat{\sigma}^2) = \frac{n-1}{n}\sigma^2 - \sigma^2 = \frac{n-1}{n}\sigma^2 - \frac{n}{n}\sigma^2 = \left(\frac{n-1}{n} - 1\right)\sigma^2 = \left(\frac{-1}{n}\right)\sigma^2$$

So, the bias of the MLE for σ^2 is $-\frac{\sigma^2}{n}$.

IITM-CS5691 : Pattern Recognition and Machine Learning
Assignment 1

Release Date: August 30, 2023
Due Date : September 14, 2023, 23:59

Roll No: CS23E001
Name: Shuvrajeet Das
Collaborators (if any):
References/sources (if any):

- Use LaTeX to write-up your solutions (in the solution blocks of the source LaTeX file of this assignment), and submit the resulting pdf files (one per question) at Crowdmark by the due date. (Note: **No late submissions** will be allowed, other than one-day late submission with 10% penalty or four-day late submission with 30% penalty! Instructions to join Crowdmark and submit your solution to each question within Crowdmark **TBA** later).
- For the programming question, please submit your code (rollno.ipynb file and rollno.py file in rollno.zip) directly in moodle, but provide your results/answers (including Jupyter notebook **with output**) in the pdf file you upload to Crowdmark.
- Collaboration is encouraged, but all write-ups must be done individually and independently, and mention your collaborator(s) if any. Same rules apply for codes written for any programming assignments (i.e., write your own code; we will run plagiarism checks on codes).
- If you have referred a book or any other online material or LLMs (Large Language Models like ChatGPT) for obtaining a solution, please cite the source. Again don't copy the source *as is* - you may use the source to understand the solution, but write-up the solution in your own words (this also means that you cannot copy-paste the solution from LLMs!). Please be advised that *the lesser your reliance on online materials or LLMs* for answering the questions, *the more your understanding* of the concepts will be and *the more prepared you will be for the course exams*.
- Points will be awarded based on how clear, concise and rigorous your solutions are, and how correct your answer is. The weightage of this assignment is 12% towards the overall course grade.

1. (8 points) [BAYESIAN DECISION THEORY]

(a) (4 points) [Optimal Classifier by Pen/Paper] Let L be the loss matrix defined by $L = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix}$, where L_{ij} indicates the loss for an input x with i being the true class and j the predicted class. Given the data:

x	-2.8	1.5	0.4	-0.3	-0.7	0.9	1.8	0.8	-2.4	-1.3	1.1	2.5	2.6	-3.3
y	1	3	2	2	1	3	3	2	1	1	2	3	3	1

find the optimal Bayes classifier $h(x)$, and provide its decision boundaries/regions. Assume that the class conditionals are Gaussian distributions with a known variance of 1 and unknown means (to be estimated from the data).

(b) (4 points) Consider a classification problem in which the loss incurred on mis-classifying an input vector from class C_k as C_j is given by loss matrix entry L_{kj}, and for which the loss incurred in selecting the reject option is ψ. Find the decision criterion that will give minimum expected loss, and then simplify it for the case of 0-1 loss (i.e., when $L_{kj} = \mathbb{1}_{k \neq j}$).

Solution: The solution of question (a):
To find the optimal Bayes classifier $h(x)$ and its decision boundaries/regions, we need to estimate the class means for the Gaussian distributions for each class using the given data and then apply Bayes' rule to classify new data points. Bayes' rule states:

$$h(x) = \underset{c \in \mathcal{C}}{\operatorname{argmax}}\, P(C = c | X = x)$$

The class means for each class based on the data:

For Class 1 (y = 1):

$$\mu_1 = \frac{\sum_{i=1}^{N} x_i}{N} = \frac{-2.8 - 0.7 - 2.4 - 1.3 - 3.3}{5} = -2.1$$

For Class 2 (y = 2):

$$\mu_2 = \frac{\sum_{i=1}^{N} x_i}{N} = \frac{0.4 - 0.3 + 0.8 + 1.1}{4} = 0.5$$

For Class 3 (y = 3):

$$\mu_3 = \frac{\sum_{i=1}^{N} x_i}{N} = \frac{1.5 + 0.9 + 1.8 + 2.5 + 2.6}{5} = 1.86$$

The decision boundaries are where the posterior probabilities are equal for two neighboring classes. Let's calculate the posterior probabilities for each class:

For Class 1:

$$P(C = 1|X = x) \propto \exp\left(-\frac{(x - \mu_1)^2}{2\sigma^2}\right)$$

For Class 2:

$$P(C = 2|X = x) \propto \exp\left(-\frac{(x - \mu_2)^2}{2\sigma^2}\right)$$

For Class 3:

$$P(C = 3|X = x) \propto \exp\left(-\frac{(x - \mu_3)^2}{2\sigma^2}\right)$$

Assuming $\sigma^2 = 1$ (known variance), we can compare these probabilities for each class at each data point x. The class with the highest probability will be the classification.

The Bayes classifier minimizes the expected loss, which can be expressed as:

$$\mathcal{E}(L(y, h(x))) = \sum_{i,j} P(C = i, h(x) = j) \cdot L(i, j)$$

1. Decision boundary between Class 1 (y = 1) and Class 2 (y = 2):

At each data point x, calculate the expected loss for both classes and compare them. The boundary occurs where the expected losses are equal:

$$\mathcal{E}(L(y = 1, h(x))) = \mathcal{E}(L(y = 2, h(x)))$$

The values of x where these equations hold are the decision boundary between Class 1 and Class 2.

2. Decision boundary between Class 2 (y = 2) and Class 3 (y = 3): Similar to the previous step, compare the expected losses between Class 2 and Class 3 to find the boundary:

$$\mathcal{E}(L(y = 2, h(x))) = \mathcal{E}(L(y = 3, h(x)))$$

The values of x where these equations hold are the decision boundary between Class 2 and Class 3.

3. Decision boundary between Class 1 (y = 1) and Class 3 (y = 3): Compare the expected losses between Class 1 and Class 3 to find the boundary:

$$\mathcal{E}(L(y = 1, h(x))) = \mathcal{E}(L(y = 3, h(x)))$$

The values of x where these equations hold are the decision boundary between Class 1 and Class 3.

Now, calculate these boundaries using the estimated means and the loss matrix L. The boundaries occur where the expected losses for the respective classes are equal at each data point x.

The solution of question (b)

The decision criterion that minimizes the expected loss is to choose the class C_j that minimizes the expected risk:

$$\text{Decision: } j = \arg\min_j \sum_k P(C_k|\text{input})L_{kj} + P(\text{reject}|\text{input})\psi$$

Here, $P(C_k|\text{input})$ is the posterior probability of the true class C_k given the input data, and $P(\text{reject}|\text{input}$ is the probability of selecting the reject option given the input data.

Now, let's simplify this decision criterion for the case of 0-1 loss, which means L_{kj} is 0 if $k = j$ (correct classification) and 1 if $k \neq j$ (incorrect classification), and ψ is a constant representing the loss for rejecting.

In this case, the decision criterion becomes:

$$\text{Decision: } j = \arg\min_j \sum_k P(C_k|\text{input}) \cdot \mathbb{I}(k \neq j) + P(\text{reject}|\text{input})\psi \tag{1}$$

Here, $\mathbb{I}(k \neq j)$ is an indicator function that equals 1 when $k \neq j$ (incorrect classification) and 0 when $k = j$ (correct classification).

So, for the case of 0-1 loss, the decision criterion becomes select the class with the highest posterior probability, and if the probability of all classes is below a certain threshold (determined by ψ), then choose the reject option.

IITM-CS5691 : Pattern Recognition and Machine Learning
Assignment 1

Release Date: August 30, 2023
Due Date : September 14, 2023, 23:59

Roll No: CS23E001
Name: Shuvrajeet Das
Collaborators (if any):
References/sources (if any):

- Use LaTeX to write-up your solutions (in the solution blocks of the source LaTeX file of this assignment), and submit the resulting pdf files (one per question) at Crowdmark by the due date. (Note: **No late submissions** will be allowed, other than one-day late submission with 10% penalty or four-day late submission with 30% penalty! Instructions to join Crowdmark and submit your solution to each question within Crowdmark **TBA** later).
- For the programming question, please submit your code (rollno.ipynb file and rollno.py file in rollno.zip) directly in moodle, but provide your results/answers (including Jupyter notebook **with output**) in the pdf file you upload to Crowdmark.
- Collaboration is encouraged, but all write-ups must be done individually and independently, and mention your collaborator(s) if any. Same rules apply for codes written for any programming assignments (i.e., write your own code; we will run plagiarism checks on codes).
- If you have referred a book or any other online material or LLMs (Large Language Models like ChatGPT) for obtaining a solution, please cite the source. Again don't copy the source *as is* - you may use the source to understand the solution, but write-up the solution in your own words (this also means that you cannot copy-paste the solution from LLMs!). Please be advised that *the lesser your reliance on online materials or LLMs* for answering the questions, *the more your understanding* of the concepts will be and *the more prepared you will be for the course exams*.
- Points will be awarded based on how clear, concise and rigorous your solutions are, and how correct your answer is. The weightage of this assignment is 12% towards the overall course grade.

1. (8 points) [REVEREND BAYES DECIDES FURTHER!]

   (a) (2 points) For a two-class optimal Bayes classifer h, the decision region is given by: $R_i = \{x \in \mathbb{R} : h(x) = C_i\}$. Is R_1 always a single interval (based on a single cutoff separating the C_1 and C_2 class) or can R_1 be composed of more than one discontiguous interval? If yes for latter, give an example by plotting the pdfs $p(x, C_1)$ and $p(x, C_2)$ against x.

   (b) (2 points) For a binary classifer h, let $L = \begin{bmatrix} p & q \\ r & s \end{bmatrix}$ be the loss matrix; and $C_{\text{train}} = \begin{bmatrix} 100 & 10 \\ 20 & 120 \end{bmatrix}$, and $C_{\text{test}} = \begin{bmatrix} 90 & 45 \\ 30 & 85 \end{bmatrix}$ be the confusion matrix when h is applied on the training and test data respectively. All three matrices have ground-truth classes t along the rows and predictions h along the columns in the same order for the two classes. Express your estimate of the expected loss of h in terms of p to s above.

   (c) (4 points) Consider the dataset introduced in the table below, where the task is to predict whether a person is ill. We use a representation based on three features per subject to describe an individual person. These features are "running nose (N)", "coughing (C)", and "reddened skin (R)", each of which can take the value true ('+') or false ('–'). (i) Classify the data point (d_7 : N = −, C = +, R = −) using a Naive Bayes classifier. As part of your solution, also write down the (ii) Naive Bayes assumption and (iii) Naive Bayes classifier, along with (iv) which distribution's MLE formula you used to estimate the class conditionals.

Training Example	N (running nose)	C (coughing)	R (reddened skin)	**Classification**
d_1	+	+	+	positive (ill)
d_2	+	+	–	positive (ill)
d_3	–	–	+	positive (ill)
d_4	+	–	–	negative (healthy)
d_5	–	–	–	negative (healthy)
d_6	–	+	+	negative (healthy)

Solution:

The solution of question (a)

In a two-class optimal Bayes classifier, the decision regions are typically separated by a single cutoff point, resulting in a single interval for each region. This is often the case when the class-conditional probability density functions (pdfs) are well-behaved and exhibit a clear separation between the two classes.

However, there can be scenarios where the decision region R1 may be composed of more than one discontiguous interval if the class-conditional pdfs overlap in such a way that it's not possible to define a single cutoff point to separate the classes. This situation is more common when the class-conditional pdfs overlap significantly.

Let's illustrate this with an example by plotting the pdfs $p(x, C_1)$ and $p(x, C_2)$ against x:

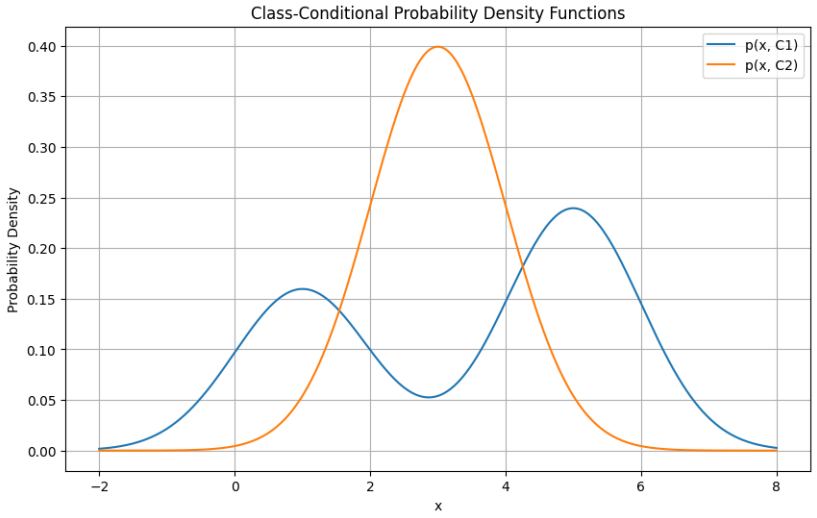
Suppose we have two classes, C1 and C2, and their class-conditional pdfs are given as follows:

For Class C1: $p(x, C1) = 0.4 * N(1, 1) + 0.6 * N(5, 1)$

For Class C2: $p(x, C2) = N(3, 1)$

Here, $N(\mu, \sigma^2)$ represents the normal distribution. In this example, the pdf for Class C_1 is a mixture of two Gaussian distributions with different means and weights.

Let's plot these pdfs:



In this plot, you can see that the two class-conditional pdfs overlap significantly, and there isn't a single cutoff point that cleanly separates the two classes. Therefore, in such cases, the decision region R1 for Class C1 would consist of multiple discontiguous intervals.

The solution of question (b)

The expected loss for the train is,
$\mathcal{E}_{train} = 100.p + 10.q + 20.r + 120.s$

The expected loss for the train is,
$\mathcal{E}_{test} = 90.p + 45.q + 30.r + 85.s$

The solution to question (c)

Let's assume the dataset $D = \{\{1, 1, 1\}, \{1, 1, 0\}, \{0, 0, 1\}, \{1, 0, 0\}, \{0, 0, 0\}, \{0, 1, 1\}\}$ and
$Y = \{\{1, 1, 1, 0, 0, 0\}$ where '+' denotes 1 and '-' denotes 0
we know posterior $\propto$ likelihood x prior,
Assuming the priors for each class 0 is 0.5 and class 1 is 0.5 (since both classes are in the same quantities)

Now, likedlihood is,

$$\mathcal{L}_n(X_1, X_2, X_3, X_4, X_5, X_6, \theta) = \prod_{i=1}^{6} p_\theta(x_i)$$

i.e. for class 1, the likelihood is 0.5, and for class 0 is 0.5

Now from the Bayes rule, the desired class is argmax of the posterior of the 2 classes but in this, the posterior probability is the same so we can choose any 1 or 0.

(i) The classified point is 1 or 0 for anyone whom you chose.

(ii) The Naive Bayes assumption is that prior is $\frac{N_{samples=C_i}}{N_{samples}}$ where $i \in \{0, 1\}$

(iii) The Naive Bayes Classifier $P(\theta|X) = \frac{P(X|\theta)P(\theta)}{P(X)}$

(iv) The Distribution is assumed to be Bernoulli.

IITM-CS5691 : Pattern Recognition and Machine Learning
Assignment 1

Release Date: August 30, 2023
Due Date : September 14, 2023, 23:59

Roll No: CS23E001
Name: Shuvrajeet Das
Collaborators (if any):
References/sources (if any):

- Use LaTeX to write-up your solutions (in the solution blocks of the source LaTeX file of this assignment), and submit the resulting pdf files (one per question) at Crowdmark by the due date. (Note: **No late submissions** will be allowed, other than one-day late submission with 10% penalty or four-day late submission with 30% penalty! Instructions to join Crowdmark and submit your solution to each question within Crowdmark **TBA** later).
- For the programming question, please submit your code (rollno.ipynb file and rollno.py file in rollno.zip) directly in moodle, but provide your results/answers (including Jupyter notebook **with output**) in the pdf file you upload to Crowdmark.
- Collaboration is encouraged, but all write-ups must be done individually and independently, and mention your collaborator(s) if any. Same rules apply for codes written for any programming assignments (i.e., write your own code; we will run plagiarism checks on codes).
- If you have referred a book or any other online material or LLMs (Large Language Models like ChatGPT) for obtaining a solution, please cite the source. Again don't copy the source *as is* - you may use the source to understand the solution, but write-up the solution in your own words (this also means that you cannot copy-paste the solution from LLMs!). Please be advised that *the lesser your reliance on online materials or LLMs* for answering the questions, *the more your understanding* of the concepts will be and *the more prepared you will be for the course exams*.
- Points will be awarded based on how clear, concise and rigorous your solutions are, and how correct your answer is. The weightage of this assignment is 12% towards the overall course grade.

1. (16 points) [LET'S ROLL UP YOUR CODING SLEEVES...] (Note: You should follow instructions in the preamble on how to submit notebook with output/results, as well as the code source files, to get full credit for this programming question.)
You are supposed to build Bayesian classifiers that model each class using multivariate Gaussian density functions for the datasets assigned to you (under assumptions below and employing MLE approach to estimate class prior/conditional densities). This assignment is focused on handling and analyzing data using interpretable classification models, rather than aiming solely for the best classification accuracy.

   Build Bayesian models for the given case numbers (you may refer to the Chapter 2 of the book "Pattern Classification" by David G. Stork, Peter E. Hart, and Richard O. Duda):

   Case 1: Bayes classifier with the same Covariance matrix for all classes.

   Case 2: Bayes classifier with different Covariance matrix across classes.

   Case 3: Naive Bayes classifier with the Covariance matrix $S = \sigma^2\mathbf{I}$ same for all classes.

   Case 4: Naive Bayes classifier with S of the above form, but being different across classes.

   Refer to the provided dataset for each group, which can be found here. Each dataset includes 2D feature vectors and their corresponding class labels. There are two different datasets available:

   1. Linearly separable data.
   2. Non-linearly separable data.

   There are 41 folders in each dataset, but you need to look at only one folder – **the folder number assigned to you** being $\texttt{RollNo}\%41 + 1$.

   Plots/answers Required: For your assignment, you need to provide the following plots/answers (refer to the "Sample Plots" folder: link):

   (a) (4 points) The plot of Gaussian pdf for all classes is estimated using the train data (train.txt). (4 Cases $\times$ 2 Datasets = 8 plots in one page)

   (b) (4 points) The classifiers, specifically their decision boundary/surface as a 2D plot along with training points marked in the plot (again 8 plots in one page).

   (c) (1 point) Report the error rates for the above classifiers (four classifiers on the two datasets as a 4×2 table, with appropriately named rows and columns).

   (d) (1 point) Answer briefly on whether we can use the most general "Case 2" for all datasets? If not, answer when a simpler model like "Case 1" is preferable over "Case 2"?

   (e) (6 points) Ensure that the properly running code files that generate the above plots, etc., are submitted according to the detailed instructions in the preamble.

   (Not)Allowed Libraries: You are not allowed to use any inbuilt functions for building the model or classification using the model. However, you can use inbuilt functions/libraries for plotting and other purposes.

Solution: The solution of question (a)

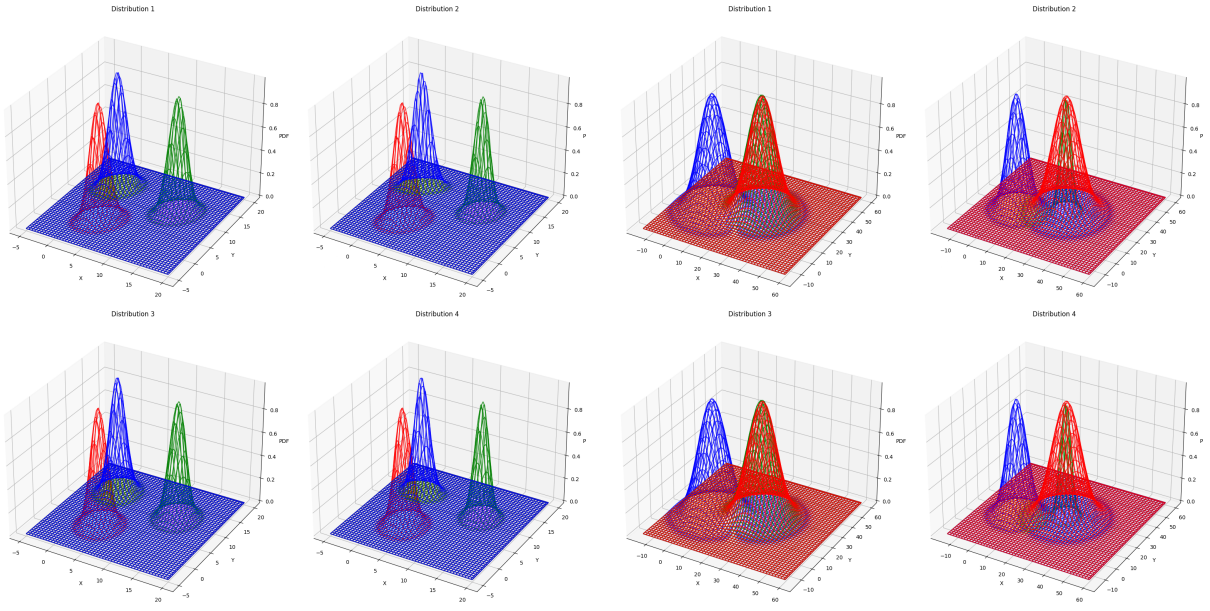


Figure 1: Figures generated Left 2 for linear data and Right 2 for non linear data

Description The first 2x2 plot is for the linearly separable data and the next 2x2 is for Non-Linearly separable data. The 8 graphs are produced after running the 4 classifier algorithm over the dataset(train.txt) for getting the plot of Gaussian pdf for all classes.

The solution of question (b)

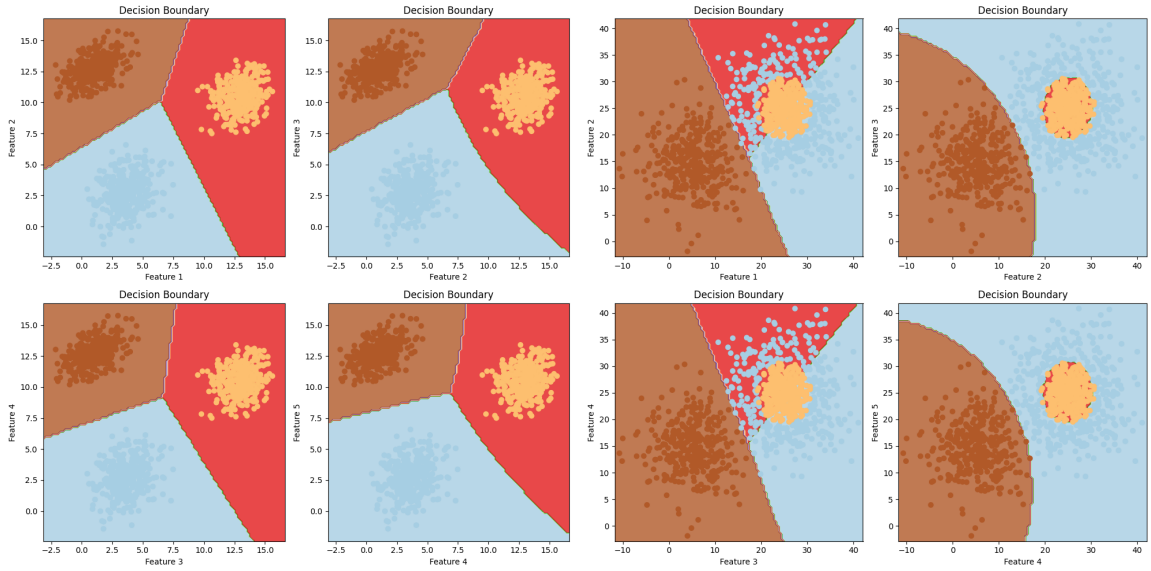


Figure 2: Figures generated Left 2 for linear data and Right 2 for non linear data

Description The first upper 2x2 plot is for the linearly separable data and the next lower 2x2 is for Non-Linearly separable data. The 8 graphs are produced after running the 4 classifier algorithm over the dataset(train.txt) for getting the classifier's decision boundary.

The solution of question (c)

Model	**Correct**	**Wrong**
Classifier 1	1050	0
Classifier 2	1050	0
Classifier 3	1050	0
Classifier 4	1050	0
Classifier 1	747	303
Classifier 2	1033	17
Classifier 3	748	302
Classifier 4	1030	20

The solution of question (d)

Yes we can use the most general case "Case 2" for all datasets.

The solution of question (e)

Reffer codes.