IITM-CS5691 : Pattern Recognition and Machine Learning
Assignment 1

Release Date: March 9, 2021
Due Date : March 19, 23:59

Roll No: EE19B094
Name: Manvar Nisharg
Collaborators (if any):
References (if any): For plotting decision boundaries

- Use LaTeX to write-up your solutions (in the solution blocks of the source LaTeX file of this assignment), and submit the resulting single pdf file at GradeScope 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! Within GradeScope, indicate the page number where your solution to each question starts, else we won't be able to grade it! You can join GradeScope using course entry code **5VDNKV**).
- 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 in the pdf file you upload to GradeScope.
- 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 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.
- Points will be awarded based on how clear, concise and rigorous your solutions are, and how correct your code is. Overall points for this assignment would be **min**(your score including bonus points scored, 50).

1. (10 points) [GETTING YOUR BASICS RIGHT!]

   (a) (1 point) You have a jar of $1,000$ coins. 999 are fair coins, and the remaining coin will always land heads. You take a single coin out of the jar and flip it 10 times in a row, all of which land heads. What is the probability your next toss with the same coin will land heads? Explain your answer. How would you call this probability in Bayesian jargon?

   Solution: Basically we need to find P(H/10H). Using Bayes Rule, we can write P(H/10H) as,

$$\begin{aligned} P(H/10H) &= \frac{P(H \cap 10H)}{P(10H)} \\ &= \frac{P(11H)}{P(10H)} \end{aligned}$$

Now, as there are two types of coins present (say coin A is the unfair coin and B the fair one) we can write,

$$\begin{aligned} P(10H) &= P(10H, A) + P(10H, B) \\ &= P(10H/A) \cdot P(A) + P(10H/B) \cdot P(B) \end{aligned}$$

$$= 1^{10} \cdot \frac{1}{1000} + (\frac{1}{2})^{10} \cdot \frac{999}{1000}$$

$$= \frac{2023}{2^{10} \cdot 1000}$$

Similarly, for the other term,

$$P(11H) = P(11H/A) \cdot P(A) + P(11H/B) \cdot P(B)$$

$$= 1^{11} \cdot \frac{1}{1000} + (\frac{1}{2})^{11} \cdot \frac{999}{1000}$$

$$= \frac{3047}{2^{11} \cdot 1000}$$

Therefore, our final answer,

$$P(H/10H) = \frac{P(11H)}{P(10H)}$$

$$= \frac{\frac{3047}{2^{11} \cdot 1000}}{\frac{2023}{2^{10} \cdot 1000}}$$

$$\approx 0.753$$

(b) (3 points) 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. How can you argue that the stationary points so obtained are indeed global maxima of the likelihood function? Next, derive the bias

of the MLE of μ, σ^2.

Solution: Now, we know that the estimated values of μ, σ from MLE for a 1-D Gaussian distribution is,

$$\mu_n = \sum_{i=1}^{N} \frac{x_i}{N}$$
$$\sigma_n = \sum_{i=1}^{N} \frac{(x_i - \mu_n)^2}{N}$$

Also, as all x_i are i.i.d samples from the distribution, therefore $E[x_i] = \mu$ where μ is the mean of the distribution and $Var[x_i] = \sigma$ where σ is the variance of the distribution.

Now for calculating the bias of μ, σ, we calculate the expected values of the above quantities.
So, in case of μ_n

$$E[\mu_n] = E\left[\sum_{i=1}^{N} \frac{x_i}{N}\right]$$
$$= \sum_{i=1}^{N} E\left[\frac{x_i}{N}\right]$$
$$E[\mu_n] = N \cdot \frac{\mu}{N}$$
$$E[\mu_n] = \mu$$

Similarly for the bias of σ,

$$
\begin{aligned}
E[\sigma] &= E\left[\sum_{i=1}^{N} \frac{(x_i - \mu_n)^2}{N}\right] \\
&= E\left[\frac{1}{N}\sum_{i=1}^{N} ((x_i - \mu) - (\mu_n - \mu))^2\right] \\
&= E\left[\frac{1}{N}\sum_{i=1}^{N} ((x_i - \mu)^2 - 2(x_i - \mu)(\mu_n - \mu) + (\mu_n - \mu)^2)\right] \\
&= E\left[\frac{1}{N}\left(\sum_{i=1}^{N} (x_i - \mu)^2 - 2(\mu_n - \mu)\sum_{i=1}^{N} (x_i - \mu) + \sum_{i=1}^{N} (\mu_n - \mu)^2\right)\right] \\
&= E\left[\frac{1}{N}\left(\sum_{i=1}^{N} (x_i - \mu)^2 - 2N(\mu_n - \mu)(\mu_n - \mu) + N(\mu_n - \mu)^2\right)\right] \\
&= E\left[\frac{1}{N}\left(\sum_{i=1}^{N} (x_i - \mu)^2 - N(\mu_n - \mu)^2\right)\right] \\
&= \frac{1}{N}E\left[\sum_{i=1}^{N}(x_i - \mu)^2\right] - E[(\mu_n - \mu)^2] \\
&= \sigma^2 - Var(\mu_n) \\
&= \sigma^2 - Var\left[\frac{1}{N}\sum_{i=1}^{N} x_i\right] \\
&= \sigma^2 - \frac{1}{N^2}\sum_{i=1}^{N} Var(x_i) \\
&= \sigma^2 - \frac{1}{N^2}N\sigma \\
&= (1 - \frac{1}{N})\sigma
\end{aligned}
$$

(c) (2 points) Consider a hyperplane $\mathbb{H}$ in $\mathbb{R}^d$ passing through zero. Prove that $\mathbb{H}$ is a subspace of $\mathbb{R}^d$ and is of dimension $d - 1$.

Solution: We can write the equation of a hyperplane $\mathbb{H}$ in $\mathbb{R}^d$ and passing through zero

can be written as,

$$a_1x_1 + a_2x_2 + ... + a_dx_d = 0 \qquad (Where\ all\ a_is\ are\ not\ 0\ together)$$

$$\begin{bmatrix} a_1 & a_2 & \dots & a_d \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix} = 0$$

$$Ax = 0$$

This can equivalently, be written in set notation,

$$\mathbb{H} = \left[x \in \mathbb{R}^d \mid Ax = 0\right]$$

Now we see that the above equation is the definition of null space of A usually called $\mathcal{N}(A)$. We also know that every null space of a matrix is a subspace, therefore we can conclude that $\mathbb{H}$ is a subspace of $\mathbb{R}^d$.

Now, for dimension, we derived that, $\mathbb{H} = \mathcal{N}(A)$. Therefore we can say that nullity of the matrix A = dimension of $\mathbb{H}$.
Since, by definition all a_is are not zero together thus, rank of matrix A = 1. Now, from rank-nullity theorem, we have

$$rank(A) + nullity(A) = d$$
$$\therefore dim(\mathbb{H}) = d - 1$$

(d) (2 points) We saw a mixture of two 1D Gaussians ($N(\mu_1, \sigma_1^2)$ and $N(\mu_2, \sigma_2^2)$) in class with parameters π_1, π_2 for the mixing proportions. Is the likelihood of this model convex or not convex? Give proof to support your view.

Solution: Likelihood in the given case is,

$$\prod_{i=1}^{N} \left(\pi_1 N(x_i; \mu_1, \sigma_1^2) + \pi_2 N(x_i; \mu_2, \sigma_2^2)\right)$$

We shall prove likelihood is not convex by using proof by counterexample. Consider the

corner case of $\pi_2 = 0$. In that case our likelihood will become,

$$\prod_{i=1}^{N} \left(\pi_1 N(x_i; \mu_1, \sigma_1^2)\right)$$

which is clearly not convex. Thus we can say that likelihood for the given model is not convex.

(e) (2 points) Show that there always exists a solution for the system of equations, $A^T Ax = A^T b$, where $x \in \mathbb{R}^m$, $A \in \mathbb{R}^{n \times m}$ and $b \in \mathbb{R}^n$. Further, show that for some solution x^* of this system of equations, Ax^* is the projection of b onto the column space of A.

Solution: We know that the column space $C(A^T A)$ is the orthogonal complement of the left nullspace of $A^T A$ i.e.

$$C(A^T A) = \left(N(A^T A)\right)^{\perp} = (N(A))^{\perp} = C(A^T)$$

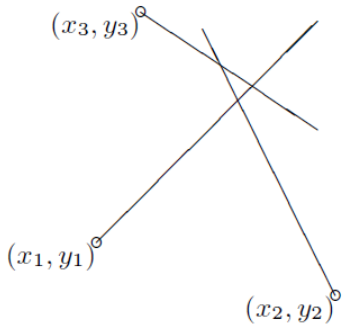
But, now as the column spaces of A^T and $A^T A$ are same, and $A^T b$ is the column space of A^T, we can surely say that $A^T Ax = A^T b$ is solvable.

Let R(A) denote the column space of A. We know that $R(A)^{\perp} = N(A^T)$ i.e. the nullspace of A^T.
Now, as $A^T Ax^* = A^T b$ or equivalently $A^T(b - Ax^*) = 0$. Thus, $(b - Ax^*)$ is orthogonal to R(A) which implies that Ax^* is the orthogonal projection of b onto R(A).

2. (5 points) [OF SAILORS AND BEARINGS...] A sailor infers his location (x, y) by measuring the bearings of three buoys whose locations (x_n, y_n) are given on his chart. Let the true bearings of the buoys be θ_n (measured from north as explained here). Assuming that his measurement $\tilde{\theta}_n$ of each bearing is subject to Gaussian noise of small standard deviation σ, what is his inferred location, by maximum likelihood?

   The sailor's rule of thumb says that the boat's position can be taken to be the centre of the cocked hat, the triangle produced by the intersection of the three measured bearings as in the figure shown. Can you persuade him that the maximum likelihood answer is better?

Solution: If we assume the true position of the ship to be (x,y).
Now, if the measured buoy angle is θ_n then the error in the measurement is $\tan^{-1}\left(\frac{x_n - x}{y_n - y}\right) - \theta_n$
Now, as the error follows normal distribution with standard deviation σ. Therefore, the probability of that happening is $\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\tan^{-1}\left(\frac{x_n - x}{y_n - y}\right) - \theta_n}{2\sigma^2}}$



Now, therefore the likelihood given the values of measured buoy angle values $\theta = (\theta_1, \theta_2, \theta_3)$ is,

$$\mathcal{L}(x, y|\theta) = \prod_{n=1}^{3} \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\tan^{-1}\left(\frac{x_n - x}{y_n - y}\right) - \theta_n}{2\sigma^2}} \right)$$

Now, the log likelihood will be,

$$\mathcal{LL}(x, y|\theta) = -3\ln(\sqrt{2\pi})\sigma - \sum_{n=1}^{3} \frac{\tan^{-1}\left(\frac{x_n - x}{y_n - y}\right) - \theta_n}{2\sigma^2}$$

Thus, now we maximise this $\mathcal{LL}$ function, wrt x and y to get the ML position of the boat i.e.

$$(x, y) = \arg\max_{x,y} \left(\mathcal{LL}(x, y|\theta)\right)$$

3. (5 points) [REVEREND BAYES DECIDES]

   (a) (2 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 $L_{kj} = 1 - I_{kj}$, with I_{kj} being 1 for $k = j$ and 0 otherwise).

   Solution: Let us assume that we have a feature vector X, for which we need to select a class (say C_r) out of set of classes (say C). We mainly have two options, either to choose a specific class (C_r) or to select no class (i.e. Reject Option). Now, we take a decision, for which we have the least expected loss among all other options.Thus, we find the expected loss for both the options.

For the reject option, we have a fixed constant loss of ψ.

$$E[L]_{reject} = \psi$$

Now, in case we choose to select a particular class C_j the expected loss in that case would be,

$$E[L]_{C_j} = \sum_k P(X, C_k) L_{kj}$$

Now, for min loss, we select the class such that $E[L]_{C_j}$ is minimum among all the classes. Thus, we choose class C_r for which,

$$E[L]_{C_r} = min(E[L]_{C_j} \; ; \; where \; C_j \in C)$$

Thus, now our final decision is to either select the class C_r which has the least possible expected loss among all classes OR choose the reject option. Here too we choose the option which has the smaller expected loss. Thus for a given input vector X,

$$Decision = \begin{cases} \text{allocate to class } C_r & \text{if } E[L]_{C_r} <= \psi \\ \text{Reject option} & \text{if } E[L]_{C_r} > \psi \end{cases}$$

Now in case of 0-1 loss matrix, we have $L_{kj} = 1 - I_{kj}$, we can simplify our expression of $E[L]_{C_j}$,

$$\begin{aligned} E[L]_{C_j} &= \sum_{k \neq j} P(X, C_k) \\ &= 1 - P(X, C_j) \end{aligned}$$

Thus, again we select the class C_r for which our expression $E[L]_{C_j}$ is minimum or equivalently the class which has the highest $P(X, C_j)$ value i.e. the class C_r for which,

$$E[L]_{C_r} = 1 - max(P(X, C_j) \; ; \; where \; C_j \in C)$$

Now again, we have to choose between either choosing class C_r or selecting the reject option. We follow the same logic as before, i.e.

$$Decision = \begin{cases} \text{allocate to class } C_r & \text{if } E[L]_{C_r} <= \psi \\ \text{Reject option} & \text{if } E[L]_{C_r} > \psi \end{cases}$$

The only difference being that the expression of $E[L]_{C_r}$ is now simpler.

(b) (2 points) 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. All the three classes are equally likely to occur. The class densities are $P(x|C_1 = 1) \sim N(-2, 1)$, $P(x|C_2 = 2) \sim N(0, 1)$ and $P(x|C_3) \sim N(2, 1)$. Find the Bayes classifier $h(x)$.

Solution: Firstly, we need to find the posterior probabilities,

$$posterior \propto (prior) * (class\ conditional)$$

Also, as priors for all class are equal,

$$posterior \propto (class\ conditional)$$

Now, we find the loss corresponding to selecting class C_j denoted by $\mathcal{L}(C_j)$ for each class,

$$\mathcal{L}(C_j) = \sum_{i=1}^{3} L_{ij} * posterior$$

$$\begin{aligned} \mathcal{L}(C_1) &= 1 * \mathcal{N}(0, 1) + 2 * \mathcal{N}(2, 1) \\ \mathcal{L}(C_2) &= 1 * \mathcal{N}(-2, 1) + 1 * \mathcal{N}(2, 1) \\ \mathcal{L}(C_3) &= 2 * \mathcal{N}(-2, 1) + 1 * \mathcal{N}(0, 1) \end{aligned}$$

Now for every x we see which of the above losses are least and choose the corresponding class.

$$\therefore h(x) = \begin{cases} C_1 & -\infty < x \leqslant -1.009 \\ C_2 & -1.009 < x \leqslant 1.009 \\ C_3 & 1.009 < x < \infty \end{cases}$$

(c) (1 point) Consider two classes C_1 and C_2 with equal priors and with class conditional densities of a feature x given by Gaussian distributions with respective means μ_1 and μ_2, and same variance σ^2. Find equation of the decision boundary between these two classes.

Solution: According to Bayes classifier, our decision is based on the posterior probability as,

$$Decision = \begin{cases} C_1 & \text{if } P(C_1/X) \geqslant P(C_2/X) \\ C_2 & \text{if } P(C_1/X) < P(C_2/X) \end{cases}$$

Thus, now our decision boundary are the input vectors where $P(C_1/X) = P(C_2/X)$
Here, we are given prior and class conditional densities. We know that,

$$P(C_1/X) \propto P(C_1) \cdot P(X/C_1)$$
$$P(C_2/X) \propto P(C_2) \cdot P(X/C_2)$$

Now, in our case the priors for both the classes are equal hence,

$$P(C_1/X) \propto P(X/C_1)$$
$$P(C_2/X) \propto P(X/C_2)$$

Thus, our decision boundary is where both the class conditional densities are equal i.e. where,

$$\mathcal{N}(\mu_1, \sigma^2) = \mathcal{N}(\mu_2, \sigma^2)$$
$$\implies X = \frac{\mu_1 + \mu_2}{2}$$

4. (10 points) [DON'T MIX YOUR WORDS!]
Consider two documents D_1, D_2 and a background language model given by a Categorical distribution (i.e., assume $P(w|\theta)$ is known for every word w in the vocabulary V). We use the maximum likelihood method to estimate a unigram language model based on D_1, which will be denoted by θ_1 (i.e, $p(w \mid \theta_1)$ = "nos. of times word w occurred in D_1"/ $|D_1|$, where $|D_1|$ denotes the total number of words in D_1). Assume document D_2 is generated by sampling words from a two-component Categorical mixture model where one component is $p(w|\theta_1)$ and the other is $p(w|\theta)$. Let λ denote the probability that D_1 would be selected to generate a word in D_2. That makes $1 - \lambda$ the probability of selecting the background model. Let $D_2 = (w_1, w_2, ...w_k)$, where w_i is a word from the vocabulary V. Use the mixture model to fit D_2 and compute the ML estimate of λ using the EM (Expectation-Maximization) algorithm.

(a) (2 points) Given that each word w_i in document D_2 is generated independently from the mixture model, write down the log-likelihood of the whole document D_2. Is it easy to maximize this log-likelihood?

Solution: Let z = 1 denote the component connected to selecting the word from D1 and z

= 2 from background model. So, the likelihood function for the document will be,

$$\begin{aligned}
\mathcal{L} &= \prod_{k=1}^{K} P(w_k;\lambda) \\
&= \prod_{k=1}^{K} \left[\sum_{z\in(1,2)} P(w_k, z;\lambda) \right] \\
\therefore \mathcal{LL} &= \sum_{k=1}^{K} \left[\log \left(\sum_{z\in(1,2)} P(w_k, z;\lambda) \right) \right] \\
&= \sum_{k=1}^{K} \left[\log \Big(P(w_k, z=1;\lambda) + P(w_k, z=2;\lambda) \Big) \right] \\
&= \sum_{k=1}^{K} \left[\log \Big(\lambda P(w_k|\theta_1) + (1-\lambda) P(w_k|\theta) \Big) \right]
\end{aligned}$$

As, we can see the log can't penetrate the sum term and thus can't be applied on the probability density functions individually. This makes differentiating and solving for the parameters very hard.

(b) (4 points) Write down the E-step and M-step updating formulas for estimating λ. Show your derivation of these formulas.

Solution: Let, us rewrite the $\mathcal{LL}$ value in a different way,

$$\mathcal{LL} = \sum_{k=1}^{K} \left[\log \left(\sum_{z\in(1,2)} Q(z) \frac{P(w_k, z;\lambda)}{Q(z)} \right) \right]$$

Thus, now using Jensen's inequality,

$$\mathcal{LL} \geqslant \sum_{k=1}^{K} \left[\sum_{z\in(1,2)} Q(z) \log \left(\frac{P(w_k, z;\lambda)}{Q(z)} \right) \right]$$

Now, for ELBO we must have $\frac{P(w_k,z;\lambda)}{Q(z)} = const.$

$$\therefore Q(z) \propto P(w_k, z; \lambda_t)$$
$$\text{Also, } \sum Q(z) = 1$$
$$\therefore Q(z) = P(z|w_k; \lambda_t)$$

Thus, in the E-step we maximise the value of $Q_{w_k}(z)$ by equating it to the posterior probability i.e.

$$Q_{w_k}(z) = P(z|w_k; \lambda_t)$$

$$\therefore Q_{w_k}(z=1) = \frac{\lambda_t P(w_k|\theta_1)}{\lambda_t P(w_k|\theta_1) + (1-\lambda_t)P(w_k|\theta)}$$
$$Q_{w_k}(z=2) = \frac{(1-\lambda_t)P(w_k|\theta)}{\lambda_t P(w_k|\theta_1) + (1-\lambda_t)P(w_k|\theta)}$$

Now, in the M-step we use the $Q_{w_k}(z)$ found above and use it to find the next set of parameters values.

$$\therefore \lambda_{t+1} = \arg\max_{\lambda} \left[\sum_{k=1}^{K} \left[\sum_{z \in (1,2)} P(z|w_k; \lambda_t) \log \left(\frac{P(w_k, z; \lambda)}{P(z|w_k; \lambda_t)} \right) \right] \right]$$

(c) (4 points) In the previous parts of the question, we assume that the background language model $P(w|\theta)$ is known. How will your E-step and M-step change if you do not know the parameter θ and only know θ_1? Show your derivation.

Solution: In this case, we can consider θ as another parameter whose value we need to calculate. We can use the same EM-Algorithm to find values of both θ as well as λ.

Now as θ is also a unknown, the E step changes slightly to,

$$Q_{w_k}(z) = P(z|w_k; \lambda_t, \theta_t)$$

$$\therefore Q_{w_k}(z=1) = \frac{\lambda_t P(w_k|\theta_1)}{\lambda_t P(w_k|\theta_1) \ + \ (1-\lambda_t) P(w_k|\theta_t)}$$
$$Q_{w_k}(z=2) = \frac{(1-\lambda_t) P(w_k|\theta_t)}{\lambda_t P(w_k|\theta_1) \ + \ (1-\lambda_t) P(w_k|\theta_t)}$$

And, in the M-step, we will now have to differentiate the equation wrt two variables and solve them to get the corresponding values.

$$\therefore \lambda_{t+1}, \theta_{t+1} = \arg\max_{\lambda,\theta} \left[\sum_{k=1}^{K} \left[\sum_{z \in (1,2)} P(z|w_k; \lambda_t, \theta_t) \log \left(\frac{P(w_k, z; \lambda, \theta)}{P(z|w_k; \lambda_t, \theta_t)} \right) \right] \right]$$

(d) (3 points) [BONUS] The previous parts of the question deal with MLE based density estimation. If you were to employ a Bayesian estimation method to infer λ, how will you proceed? That is, what prior would you choose for λ, and what is the the formula for the posterior? Is this posterior easily computable (i.e., has a closed-form expression or can be computed efficiently)? You can assume that both $P(w|\theta_1)$ and $P(w|\theta)$ are known and only λ is not known.

Solution:

5. (10 points) [DENSITY ESTIMATION - THE ONE RING TO RULE THEM ALL!] With density estimation ring already in your finger, you have all you need to master simple linear regression (even before seeing regression formally in class). Simple linear regression is a model that assumes a linear relationship between an input (aka independent) variable x and an output (aka dependent) variable y. Let us assume that the available set of observations, $\mathbb{D} = \{x_i, y_i\}_{i=1}^n$, are iid samples from the following model that captures the relationship between y and x:

$$y_i = w_0 + w_1 x_i + \epsilon_i; \qquad \epsilon_i \sim \mathcal{N}(0, \sigma^2)$$

In this model, note that x_i is not a random variable, whereas ϵ_i and hence y_i are random variables, with ϵ_i being modeled as a Gaussian noise that is independent of each other and doesn't depend on x_i. Value of σ is assumed to be known for simplicity.

We would like to learn the parameters $\theta = \{w_0, w_1\}$ of the model, i.e., we would like to use MLE to estimate the exact parameter values or Bayesian methods to infer the (posterior) probability distribution over the parameter values.

(a) (2 points) Compute the probability distribution $P(y_i|x_i, \theta)$, and use it to write down the log likelihood of the model.

Solution: When we get the output y_i for a specific value of x_i, it is equivalent to saying ϵ_i took the value $y_i - w_0 - w_1 x_i$. Thus,

$$P(y_i|x_i, \theta) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - w_0 - w_1 x_i)^2}{2\sigma^2}}$$

Thus, the log likelihood of the model will be,

$$\begin{aligned}
\mathcal{L} &= \prod_{i=1}^{N} P(y_1|x_1, \theta) \\
&= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - w_{0n} - w_{1n} x_i)^2}{2\sigma^2}} \\
\therefore \mathcal{LL} &= \sum_{i=1}^{N} \log(P(y_i|x_i, \theta) \\
&= -\frac{N}{2}\log(2\pi) - N\log(\sigma) - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - w_{0n} - w_{1n} x_i)^2
\end{aligned}$$

(b) (3 points) Derive the ML estimates for w_0 and w_1 by optimizing the above log likelihood.

Solution: Now, for estimating the values of w_{0n} and w_{1n}, we differentiate the $\mathcal{LL}$ w.r.t w_{0n} and w_{1n} and equate them to 0.

$$\begin{aligned}
\frac{\partial(\mathcal{LL})}{\partial w_{0n}} &= -2\sum_{i=1}^{N}(y_i - w_{0n} - w_{1n} x_i) = 0 \\
\therefore -2N\overline{y} + 2w_{1n}N\overline{x} + 2Nw_{0n} &= 0 \\
\therefore w_{0n} &= \overline{y} - w_{1n}\overline{x}
\end{aligned}$$

Now,

$$\frac{\partial(\mathcal{LL})}{\partial w_{1n}} = -2\sum_{i=1}^{N}(y_i - w_{0n} - w_{1n}x_i)x_i = 0$$

$$-2\sum_{i=1}^{N} x_i y_i + 2w_{1n}\sum_{i=1}^{N} {x_i}^2 + 2w_{0n}\sum_{i=1}^{N} x_i = 0$$

$$-2\sum_{i=1}^{N} x_i y_i + 2w_{1n}\sum_{i=1}^{N} {x_i}^2 + (\overline{y} - w_{1n}\overline{x})\sum_{i=1}^{N} x_i = 0$$

$$2w_{1n}(\sum_{i=1}^{N} {x_i}^2 - \overline{x}\sum_{i=1}^{N} x_i) + 2\overline{y}\sum_{i=1}^{N} x_i - 2\sum_{i=1}^{N} x_i y_i = 0$$

$$\therefore w_{1n} = \frac{\sum_{i=1}^{N} x_i y_i - \overline{xy}}{\sum_{i=1}^{N} x_i^2 - N\overline{x}^2}$$

$$\boxed{w_{1n} = \frac{\sum_{i=1}^{N}(y_i - \overline{y})(x_i - \overline{x})}{\sum_{i=1}^{N}(x_i - \overline{x})^2}}$$

$$\boxed{\therefore w_{0n} = \overline{y} - w_{1n}\overline{x}}$$

(c) (2 points) If σ is also not known before, derive the ML estimate for σ.

Solution: If, σ is also not a known constant then, $\mathcal{LL}$ becomes,

$$\mathcal{LL} = -\frac{N}{2}\log(2\pi) - N\log(\sigma_n) - \frac{1}{2\sigma_n^2}\sum_{i=1}^{N}(y_i - w_{0n} - w_{1n}x_i)^2$$

Then, $$\frac{\partial(\mathcal{LL})}{\partial\sigma_n} = -\frac{N}{\sigma_n} + \frac{1}{\sigma_n^3}\sum_{i=1}^{N}(y_i - w_{0n} - w_{1n}x_i)^2 \quad = 0$$

$$\boxed{\therefore \sigma_n^2 = \frac{1}{N}\sum_{i=1}^{N}(y_i - w_{0n} - w_{1n}x_i)^2}$$

(d) (3 points) For Bayesian inference, assume that the parameters w_0, w_1 are independent of each other and follow the distributions $\mathcal{N}(\mu_0, \sigma_0^2)$ and $\mathcal{N}(\mu_1, \sigma_1^2)$ respectively. Compute the posterior

distributions for each parameter. How does the mode of this posterior (i.e., MAP estimate) relate to the MLE of w_0 and w_1 derived above?

Solution: Now, for Bayesian inference, the priors for w_0 and w_1 are $\mathcal{N}(\mu_0, \sigma_0)$ and $\mathcal{N}(\mu_1, \sigma_1)$ respectively. Thus, for posterior,

$$\texttt{posterior} = \texttt{likelihood} * \texttt{prior}$$

$$\therefore P(w_{0n}, w_{1n}|D, \sigma_0, \mu_0, \sigma_1, \mu_1) =$$

$$\left(\prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - w_{0n} - w_{1n}x_i)^2}{2\sigma^2}}\right)\left(\frac{1}{\sqrt{2\pi\sigma_0^2}} e^{-\frac{(w_{0n}-\mu_0)^2}{2\sigma_0^2}} \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{(w_{1n}-\mu_1)^2}{2\sigma_1^2}}\right)$$

Thus, for posterior of each variable we integrate the posterior w.r.t other variable i.e.

$$P(w_{0n}|D, \sigma_0, \mu_0, \sigma_1, \mu_1) = \int_{-\infty}^{\infty} P(w_{0n}, w_{1n}|D, \sigma_0, \mu_0, \sigma_1, \mu_1)\, d(w_{1n})$$

$$P(w_{1n}|D, \sigma_0, \mu_0, \sigma_1, \mu_1) = \int_{-\infty}^{\infty} P(w_{0n}, w_{1n}|D, \sigma_0, \mu_0, \sigma_1, \mu_1)\, d(w_{0n})$$

Now, for the values of w_{0n} and w_{1n} we take log of posterior and differentiate it w.r.t both w_{0n} and w_{1n} and equate them to 0 we get two linear equations.

$$\frac{1}{\sigma^2}\sum y_i - w_{0n}\left(\frac{N}{\sigma^2} - \frac{1}{\sigma_0^2}\right) - \frac{w_{1n}}{\sigma^2}\sum x_i + \frac{\mu_0}{\sigma_0^2} = 0$$

$$\frac{1}{\sigma^2}\sum y_i x_i - \frac{w_{0n}}{\sigma^2}\sum x_i - w_{1n}\left(\frac{\sum x_i^2}{\sigma^2} + \frac{1}{\sigma_1^2}\right) + \frac{\mu_1}{\sigma_1^2} = 0$$

Solving for w_{0n} and w_{1n} gives us the desired values.

Also, now as $\sigma_1, \sigma_2 \to \infty$, the prior becomes non-informative as it equal for all values. In this case, the parameters of MLE become same as MAP.

6. (10 points) [LET'S ROLL UP YOUR CODING SLEEVES...] **Learning Binary Bayes Classifiers from data via Density Estimation**
   Derive Bayes classifiers under assumptions below and employing maximum likelihood approach to estimate class prior/conditional densities, and return the results on a test set.

   1. **BayesA** Assume $X|Y = -1 \sim \mathcal{N}(\mu_-, I)$ and $X|Y = 1 \sim \mathcal{N}(\mu_+, I)$
   2. **BayesB** Assume $X|Y = -1 \sim \mathcal{N}(\mu_-, \Sigma)$ and $X|Y = 1 \sim \mathcal{N}(\mu_+, \Sigma)$

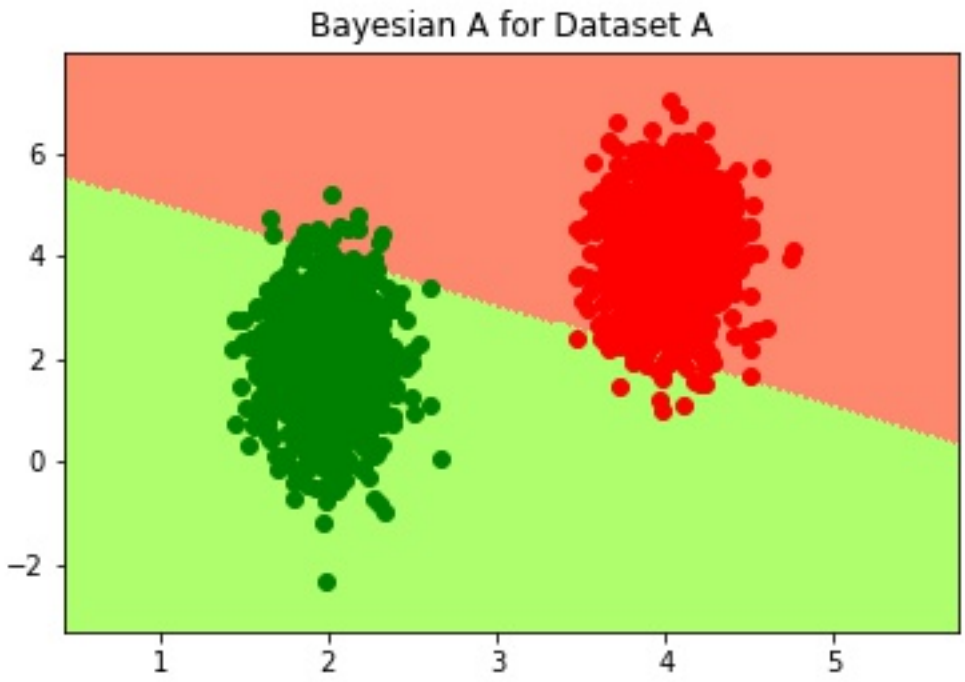
3. **BayesC** Assume $X|Y=-1 \sim \mathcal{N}(\mu_-, \Sigma_-)$ and $X|Y=1 \sim \mathcal{N}(\mu_+, \Sigma_+)$

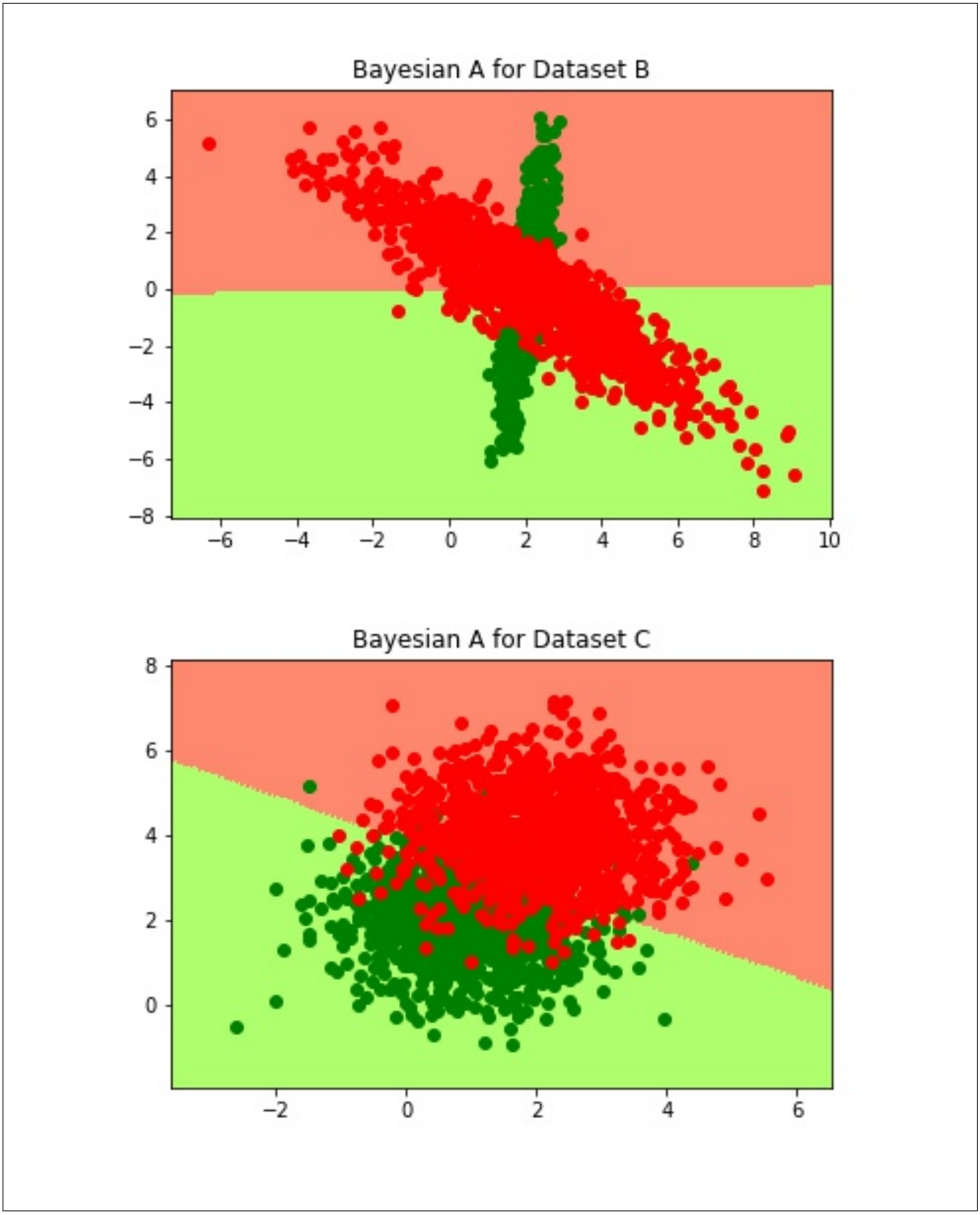
Please see this folder for the template .ipynb file containing the helper functions, and you've to add the missing code to this file (specifically, three functions function_for_A, function_for_B and function_for_C, and associated plotting/ROC code snippets) to implement the above three algorithms for the three datasets given in the same folder.

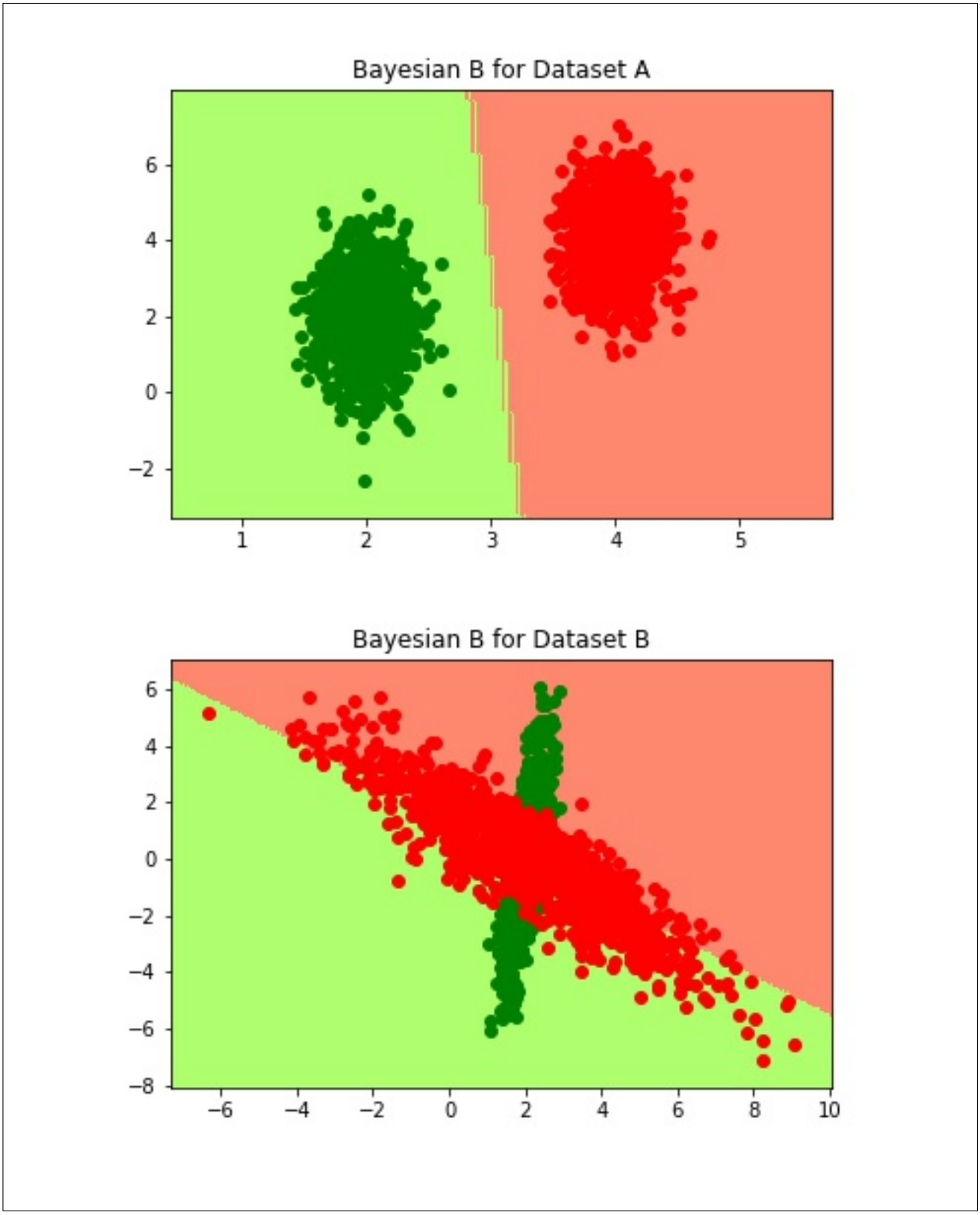
Please provide your results/answers in the pdf file you upload to GradeScope, but please submit your code separately in this moodle link. The code submitted should be a rollno.zip file containing two files: rollno.ipynb file (including your code as well as the exact same results/plots uploaded to Gradescope) and the associated rollno.py file.

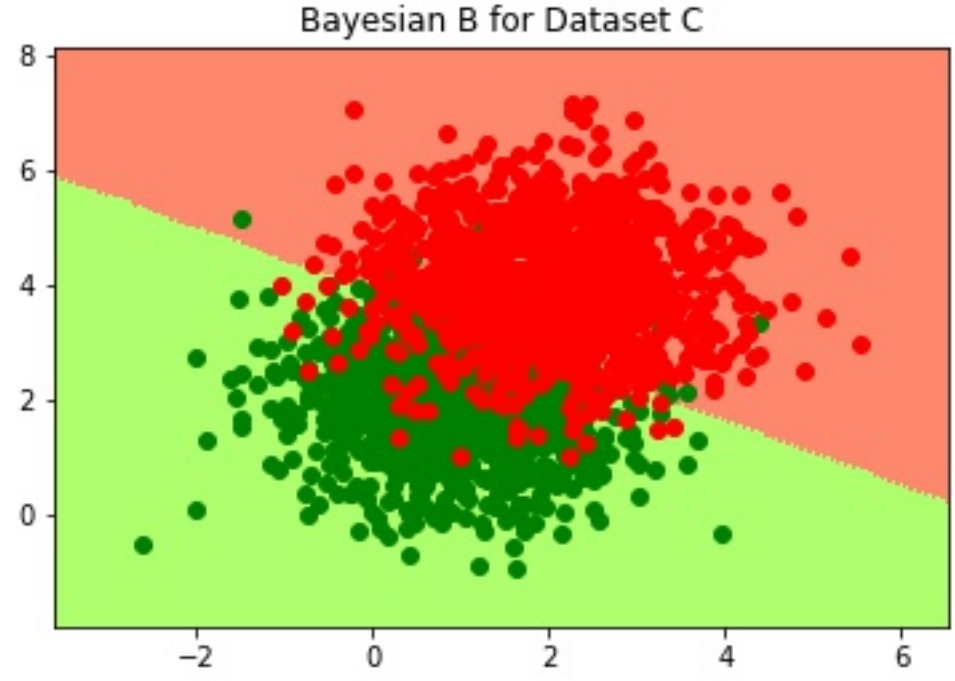
(a) (3 points) Plot all the classifiers (3 classification algorithms on 3 datasets = 9 plots) on a 2D plot, Add the training data points also on the plots. (Color the positively classified area light green, and negatively classified area light red as in Fig 4.5 in Bishop's book).

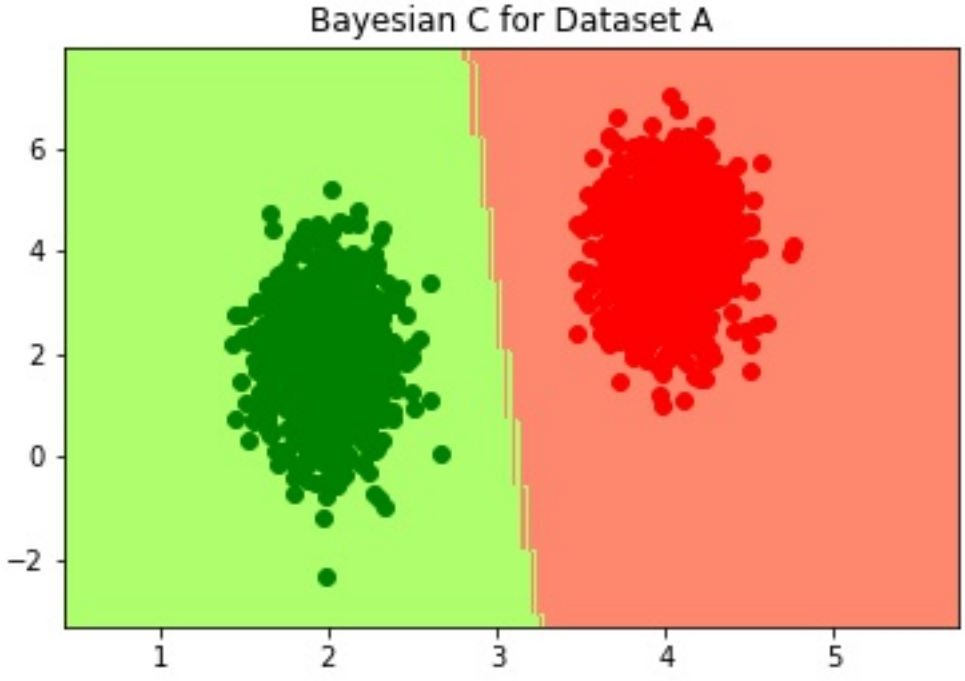
Solution:

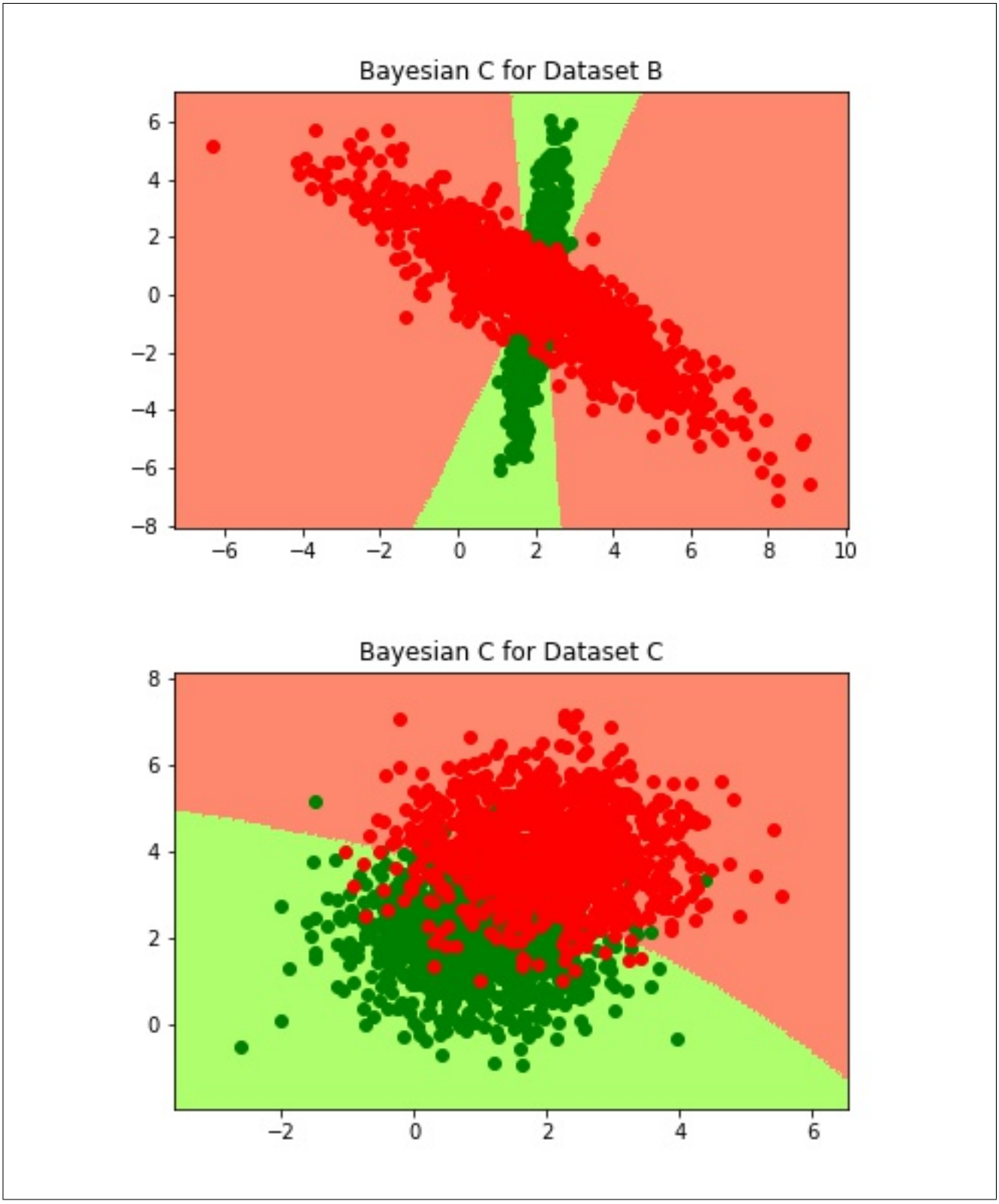



Bayesian A for Dataset B
6
4
2
0
−2
−4
−6
−8
−6
−4
−2
0
2
4
6
8
10
Bayesian A for Dataset C
8
6
4
2
0
−2
0
2
4
6


Bayesian B for Dataset A
6
4
2
0
−2
1
2
3
4
5
Bayesian B for Dataset B
6
4
2
0
−2
−4
−6
−8
−6
−4
−2
0
2
4
6
8
10

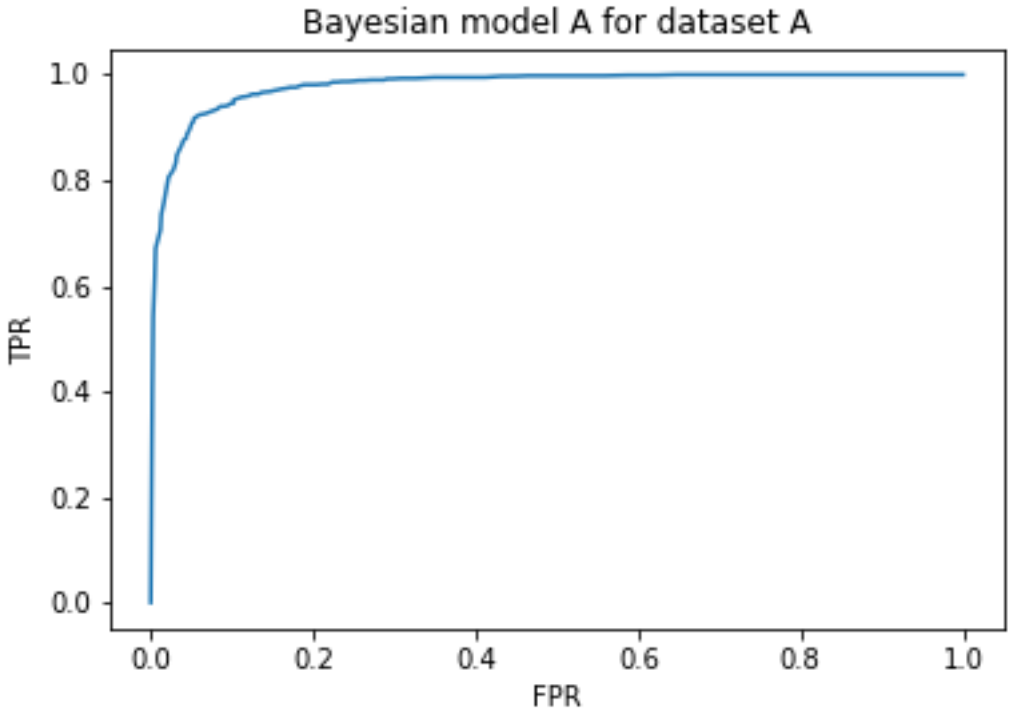

Bayesian B for Dataset C
8
6
4
2
0
−2
0
2
4
6

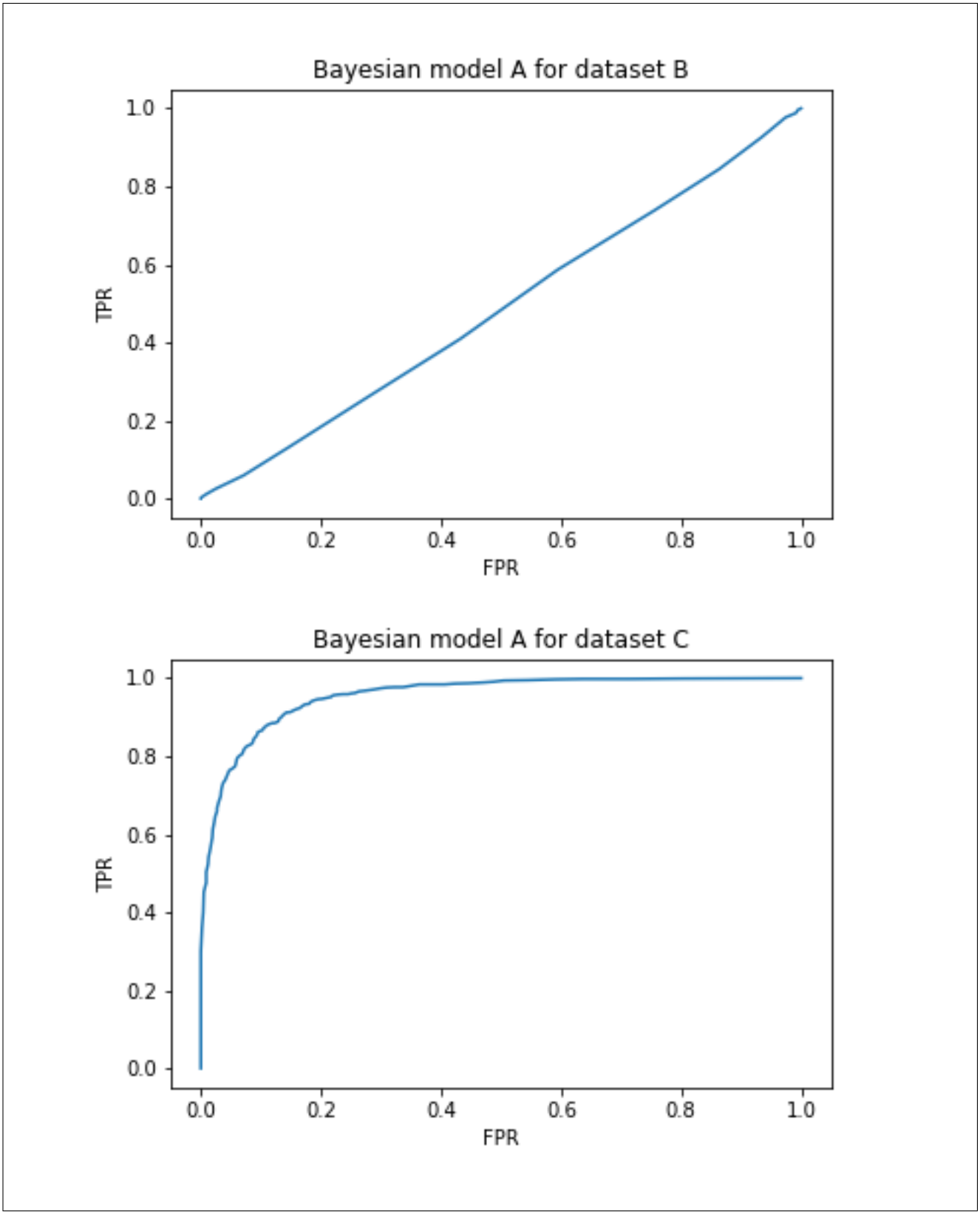

Bayesian C for Dataset A
6
4
2
0
−2
1
2
3
4
5

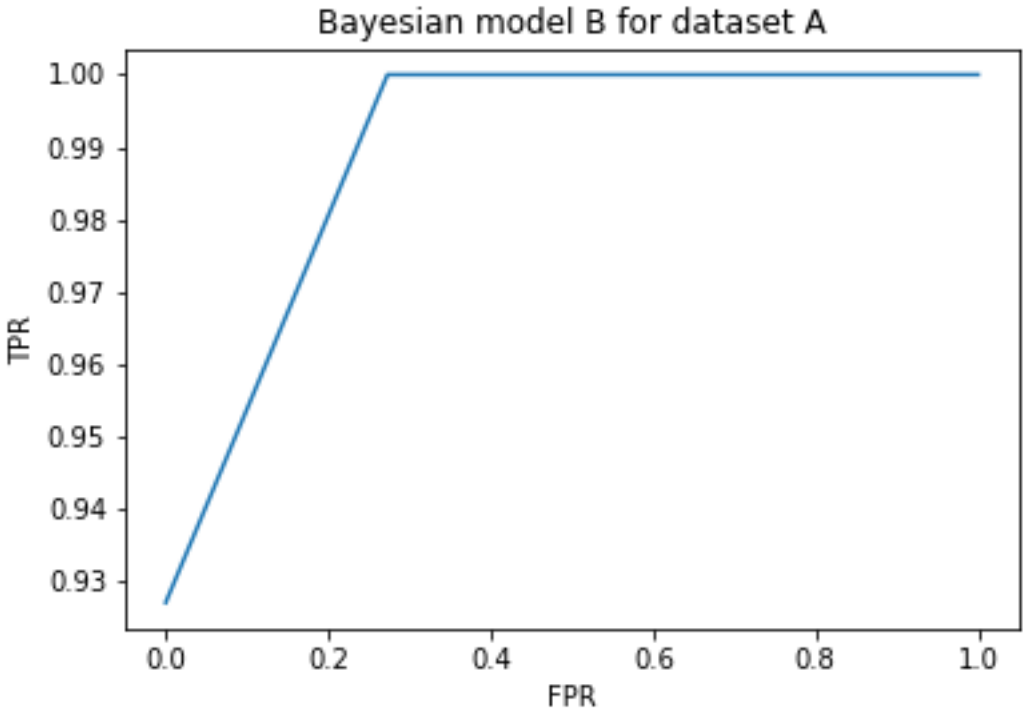

Bayesian C for Dataset B
6
4
2
0
−2
−4
−6
−8
−6
−4
−2
0
2
4
6
8
10
Bayesian C for Dataset C
8
6
4
2
0
−2
0
2
4
6

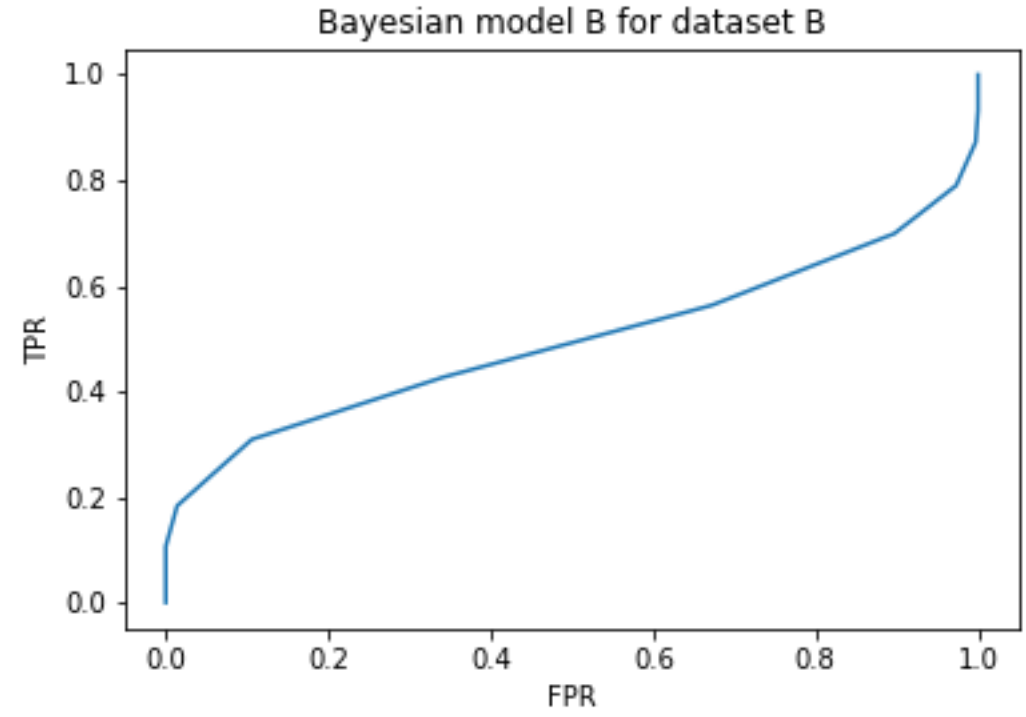
(b) (3 points) Give the ROC curves for all the classifiers. Note that a ROC curve plots the FPR (False Positive Rate) on the x-axis and TPR (True Positive Rate) on the y-axis. (9 plots)

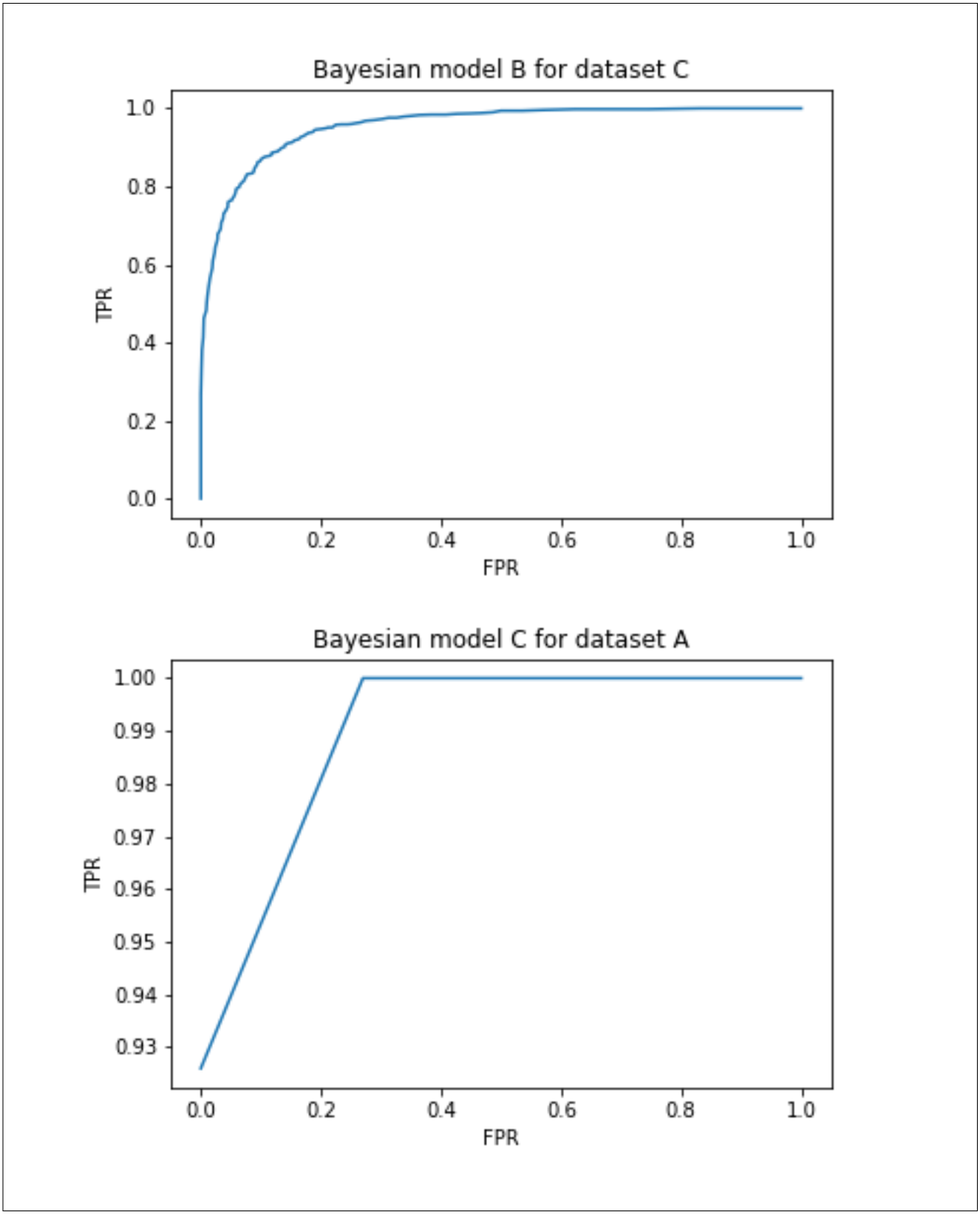
Solution:

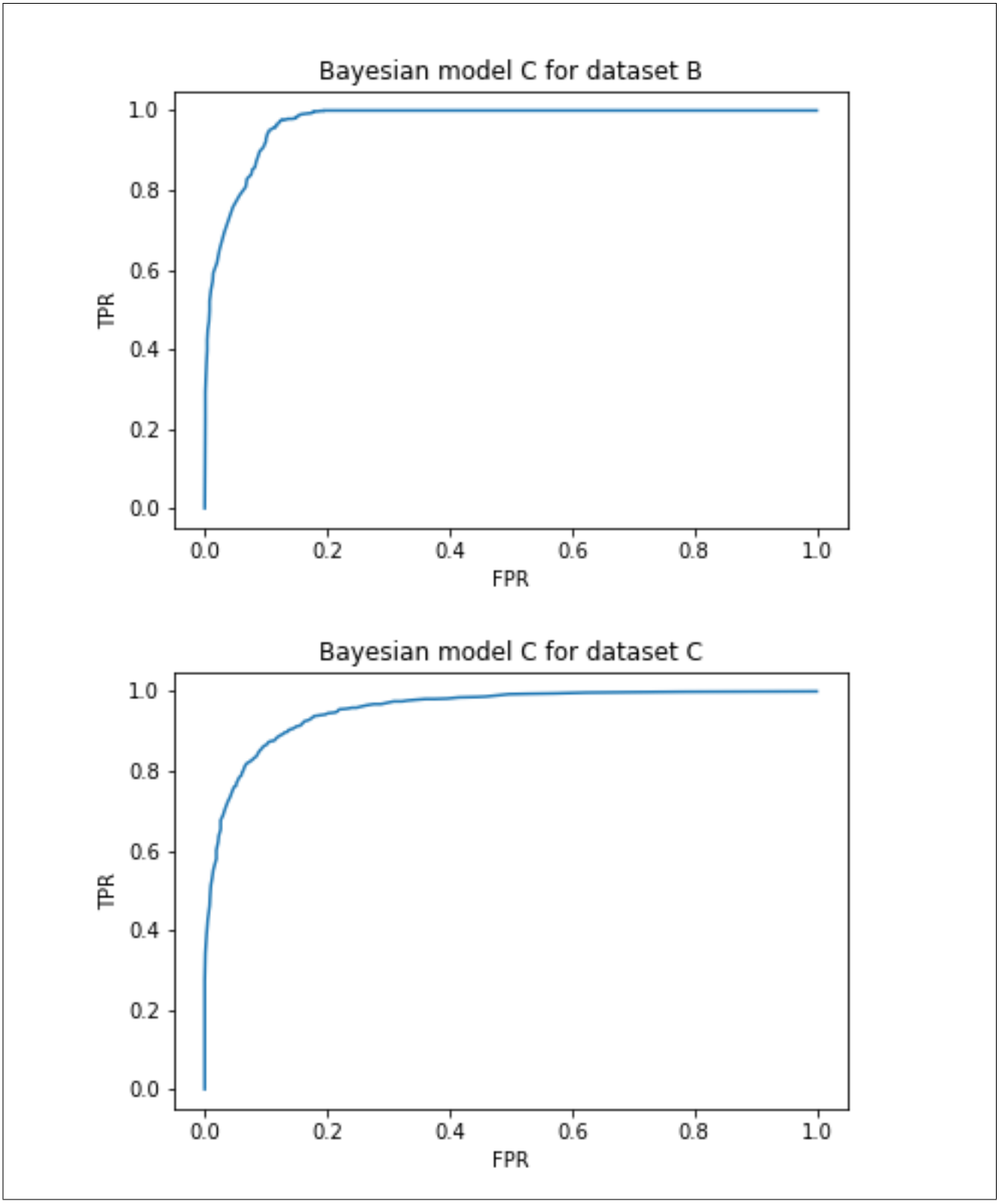



Bayesian model A for dataset B
TPR
FPR
0.0
0.2
0.4
0.6
0.8
1.0
Bayesian model A for dataset C
TPR
FPR
0.0
0.2
0.4
0.6
0.8
1.0


Bayesian model B for dataset A
TPR
FPR
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0.0
0.2
0.4
0.6
0.8
1.0


Bayesian model B for dataset B
TPR
FPR
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0


Bayesian model B for dataset C
TPR
FPR
0.0
0.2
0.4
0.6
0.8
1.0
Bayesian model C for dataset A
TPR
FPR
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0.0
0.2
0.4
0.6
0.8
1.0


Bayesian model C for dataset B
TPR
FPR
0.0
0.2
0.4
0.6
0.8
1.0
Bayesian model C for dataset C
TPR
FPR
0.0
0.2
0.4
0.6
0.8
1.0

(c) (2 points) Provide the error rates for the above classifiers (three classifiers on the three datasets as 3×3 table, with appropriately named rows and columns).

Solution: Following are the error rates for various combinations of bayesian model and dataset.

Classifier \ Dataset	A	B	C
A	9.8%	50.84%	11.75%
B	22.85%	50.00%	11.65%
C	22.55%	7.45%	11.80%

(d) (2 points) Summarise and explain your observations based on your plots and the assumptions given in the problem. Also briefly comment whether a non-parametric density estimation approach could have been used to solve this problem, and if so, what the associated pros/cons are compared to the parametric MLE based approach you have implemented.

Solution: We see that as model C is more general in the sense that it has more parameters to optimize, it is consistently pretty accurate than model A and model B. This can be particularly seen in the data-set B in which as the variances vary greatly between the two types of data-points, the assumption of the variances being equal in A and B lead to big error rates.