CS 446 / ECE 449 — Homework 1

yiminc2

Version 1.0

Instructions.

- Homework is due **Wednesday, September** 17^{th}**, 11:59 a.m**; you have **3** late days in total for **all Homeworks**.
- Everyone must submit **individually** at gradescope under `hw1`.
- The "written" submission at `hw1` **must be typed**, and submitted in any format gradescope accepts (to be safe, submit a PDF). You may use LaTeX, markdown, google docs, MS word, whatever you like; but it must be typed!
- When submitting at `hw1`, gradescope will ask you to **mark out boxes around each of your answers**; please do this precisely!
- Please make sure your NetID is clear and large on the first page of the homework.
- Your solution **must** be written in your own words. Please see the course webpage for full **academic integrity** information. You should cite any external reference you use.

1. K-NN (43 pt.)

(a) Assume that we have a dataset with two labels. The training set comes from the input space which is in the form of Figure 1. Each input sample is within the region enclosed by either of the circles. Each circle represents one class: all the samples that come from the left circle (blue one) are labeled as 0 and all the samples that come from the right circle (green one) are labeled as 1.

i. We know we have N total of training samples, but we do not know how many of them come from each class. We are given a single point from the green circle as well as access to a K-NN classifier for which we can choose arbitrary K and query the label of the K-NN classifier for the given point. What is the **minimum** number of queries required (in terms of N) to determine the number of training points from each class? Note that at each step, we can choose any K and query the K-NN classifier for the label of the test sample, i.e. from the green circle. Explain your answer. You may assume the number of green points is less than $N/2$. (6 pt.)
Hint: Can we do better than $\mathcal{O}(N)$?

ii. Is there any setting (the number of training samples, their label distribution, etc.) that leads to the wrong prediction for a given test point using 1-NN classifier? (4 pt.)

Figure 1: K-NN Training Dataset

(b) Consider a set of data points in $\mathbb{R}^3$ (see Table 1), now we want to use K-nearest neighbors (K-NN) to classify points in $\mathbb{R}^3$. Answer the questions below.

Index	x_1	x_2	x_3	**Label**
1	1	1	1	1
2	0	0	1	1
3	0	0	0	-1
4	1	0	0	-1
5	0	1	0	-1

Table 1: A set of data points

i. If $K = 1$, if you have a new data point $\boldsymbol{x} = (0.4, 0.4, 1.5)$, what would the label of $\boldsymbol{x}$ be? (Please use Euclidean distance as the distance metric for this question.) (3 pt.)

ii. When there is more noise in the training dataset, to reduce the overfitting problem, should we choose smaller K or larger K? Please explain your answer. (3 pt.)

(c) Assume an input space $S = \{\boldsymbol{x}^{(i)}\}_{i=1}^N$, all $\boldsymbol{x}^{(i)}$ being distinct. For each $\boldsymbol{x}^{(i)}$, the label $y^{(i)} \in \{1, 2\}$ is drawn independently with $\mathbb{P}(y^{(i)} = 1 \,|\, \boldsymbol{x}^{(i)}) = 0.9$. We form a training set $\mathcal{D} = \{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^N$. Given a test sample $(\boldsymbol{x}, y)$ where $\boldsymbol{x} \in S$ and y is drawn independently with the same conditional distribution, a 1-NN classifier predicts the label $\hat{y}$ of the $\boldsymbol{x}$. What would be the probability that we get the correct prediction for $\boldsymbol{x}$? (8 pt.)
Hint: Note that from the assumptions, we know that $\exists(\boldsymbol{x}^{(j)}, y^{(j)}) \in \mathcal{D}, \boldsymbol{x}^{(j)} = \boldsymbol{x}$.

(d) Consider the same setting of (c) but now suppose for each $\boldsymbol{x}^{(i)} \in S$ we have *three independent labeled replicas*

$$(\boldsymbol{x}^{(i)}, y^{(i,1)}),\ (\boldsymbol{x}^{(i)}, y^{(i,2)}),\ (\boldsymbol{x}^{(i)}, y^{(i,3)}),$$

where each $y^{(i,r)}$ is drawn independently with $\mathbb{P}(y^{(i,r)} = 1 \,|\, \boldsymbol{x}^{(i)}) = 0.9$. Let the training set be $\mathcal{D} = \{(\boldsymbol{x}^{(i)}, y^{(i,r)}) : r = 1, 2, 3\}_{i=1}^{N}$. Given a test sample $(\boldsymbol{x}, y)$ where $\boldsymbol{x} \in S$ and y is drawn independently with the same conditional distribution, what would be the probability that a 3-NN classifier makes the correct prediction for a test sample $\boldsymbol{x}$? How does this probability compare to using 1-NN? (8 pt.)

Hint: Note that from the assumptions, we know that $\exists(\boldsymbol{x}^{(j)}, y^{(j)}) \in \mathcal{D}, \boldsymbol{x}^{(j)} = \boldsymbol{x}$.

(e) In real-world scenarios, it is common for test data points to have missing features (e.g., due to sensor failure or incomplete measurements). Is it still possible to apply the K-NN algorithm in such situations? If so, describe give one of the ways of how this can be achieved. (5 pt.)

(f) **(Bonus question)** Assume an input space S to be a polytope with N vertices in $\mathbb{R}^d$. Further, assume that no pair of vertices are farther apart than 1 from one another. We are interested in training a 1-NN classifier on this input space such that the nearest neighbor used for the label prediction of each given point will be in a radius of r from that point. Show that at most N^{1/r^2+1} training samples are needed. For this purpose, you can use the following theorem:

(Special case of Approximate Caratheodory's theorem) For such an input space, assuming that the vertices are $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(N)}$, for each point $x \in S$ and each integer value p, there exist a subset of the vertices of the polytope $\boldsymbol{x}^{(j_1)}, \ldots, \boldsymbol{x}^{(j_p)} \in S$, such that $\|\boldsymbol{x} - \frac{1}{p}\sum_{i=1}^{p} \boldsymbol{x}^{(j_i)}\|_2 \leq \frac{1}{\sqrt{p}}$. (6 pt.)

Solution.

(a) i. What is the minimum number of training samples in the green circle?
Ans: $LogN$
As the two circles do not overlap, their respective data points are certainly closer to their neighbors within the same circle, leading the classifier to output the majority of labels ($K = N$). Given this fact and the following assumptions

- $output = \begin{cases} 0 & (blue) \\ 1 & (green) \end{cases}$
- The number of samples in the green circle is less than $N/2$

We can iteratively classify the given green-labeled data point, starting with $K = N/2$, dividing by 2 in each iteration. This way, we can effectively perform a Binary Search on the size of the green circle and minimize the number of queries to $logN$.

ii. Is there any setting that leads to the wrong prediction for a given test point using a 1-NN classifier?
Ans: Yes, if the data points are imbalanced between the two circles.
Assume the blue circle is dense while the green circle is sparse. The green input data may end up closer to a point in the blue circle and be classified as blue.

(b) i. What would be the label of x?
Ans: Label 1
Given $K = 1$, the data point with the index 2 is the closest to the input x with the Euclidean distance: Given $K = 1$, the data point with the index 2 (whose label is 1) is the closest to the input x with the Euclidean distance: Given $K = 1$, the data point with the index 2 is the closest to the input x with the Euclidean distance:

$$\sqrt{(0-0.4)^2 + (0-0.4)^2 + (1-1.5)^2} = \sqrt{0.57} \approx 0.7549834435$$

ii. Should we choose a smaller K or a larger K to reduce overfitting when the training data contains a lot of noise?
Ans: Larger K.
To avoid overfitting, a larger K value should be adopted, so the input is not affected by the labels of outliers. As the KNN algorithm classifies an input based on the most common labels among K nearest neighbors to the input, the more neighbors considered, the less say a single neighbor has. Thus, increasing the K value suppresses the negative impact of noise.

(c) What is the probability that we get the correct prediction for **x**?
Ans: 0.82
Given the input space $S = \{x^{(i)}\}_{i=1}^{N}$ and the training set domain $\mathcal{D} = \{x^{(i)}, y^{(i)}\}_{i=1}^{N}$ of the 1-NN classifier, we can extrapolate that the i^{th} training sample $x^{(i)}$ is the same as the input x. Therefore, the probability of the correct prediction requires that two independent draws of the labels (one for the real label y and the other for the classifier's prediction $\hat{y}$) from the given conditional distribution should match.

$$\begin{aligned}&\mathbb{P}(\hat{y} = y)\\&= \mathbb{P}(y = 1, \hat{y} = 1) + \mathbb{P}(y = 2, \hat{y} = 2)\\&= 0.9^2 + 0.1^2 = 0.82\end{aligned}$$

(d) What would be the probability that a 3-NN classifier makes the correct prediction for a test sample x? How does this probability compare to using 1-NN?
Ans: (1) 0.8776 (2) The probability is increased by 0.0576.
Given that the settings are the same and the value of K increased to 3, we can recalculate the probability of correct predictions as follows:

- 3-NN classifier's predictions ($\hat{y}$):

$$\begin{cases}\mathbb{P}(\hat{y} = 1) = \binom{3}{2} \cdot 0.9^2 \cdot 0.1^1 + \binom{3}{3} \cdot 0.9^3 \cdot 0.1^0 = 0.972\\ \mathbb{P}(\hat{y} = 2) = 1 - \mathbb{P}(\hat{y} = 1) = 0.028\end{cases}$$

- Updated probability of correct predictions:

$$\begin{aligned}&\mathbb{P}(\hat{y} = y)\\&= \mathbb{P}(y = 1, \hat{y} = 1) + \mathbb{P}(y = 2, \hat{y} = 2)\\&= 0.9 \times 0.972 + 0.1 \times 0.028 = 0.8776\end{aligned}$$

(e) Is it possible to apply the K-NN algorithm in situations where the features of data points are missing?
Ans: Yes, it is possible to apply the K-NN algorithm to impute missing features using K-NN imputation. The K-NN imputation first finds the K nearest neighbors and then replaces the missing values with the aggregation (i.e., mean, median, or mode) of the neighbors' values.

2. Perceptron Algorithm (33 Pt.)

(a) Consider a set of data points in $\mathbb{R}^3$ (the data samples follow what you have in 1(b), see Table 1 for details), now you want to use perceptron algorithm to correctly classify all the data points, answer the following questions below (**Please follow the hacked notation in the lecture 3 slides page 12**):

i. What is the vector $\mathbf{w} \in \mathbb{R}^{\mathbf{4}}$ after first iteration over the first data point? (2 pt)

ii. Compute the $\mathbf{w}$ after the algorithm is converged. (2 pt)

iii. If we switch the label of points 2 and 3, will the perceptron algorithm still be applicable? Prove your answer. (*Hint*: 3-d space is still visualizable :)) (4 pt)

iv. With the same set of training data, will the K-NN and perceptron algorithm always have the same results on the test set? If yes, prove it, otherwise give a counterexample. (3 pt)

(b) Assume:

i. $\exists \mathbf{w}^*$ such that $y^{(i)}(\mathbf{w}^{*\top})\mathbf{x}^{(i)} > 0,\ \forall(\mathbf{x}^{(i)}, y^{(i)}) \in \mathcal{D}$

ii. Rescale $\mathbf{w}^*$ and each data point such that $\|\mathbf{w}^*\| = 1$ and $\|\mathbf{x}^{(i)}\| \leq 1,\ \forall \mathbf{x}^{(i)} \in \mathcal{D}$

iii. Margin of a hyperplane γ is defined as $\gamma = \min |\mathbf{w}^{*\top}\mathbf{x}^{(i)}|, \forall \mathbf{x}^{(i)} \in \mathcal{D}$

Consider an adapted Perceptron Algorithm as illustrated in Algorithm 1.

Algorithm 1 Adapted Perceptron Algorithm

Input: Dataset $\mathcal{D}$.
Output: Weight vector $\mathbf{w}$
Initialization: $\mathbf{w} = [0, \cdots, 0]^T$.
while TRUE **do**
  changed = FALSE
  for i = 1 to N **do**
    if $\frac{y^{(i)}\mathbf{w}^T\mathbf{x}^{(i)}}{\|\mathbf{w}\|} \leq \frac{\gamma}{2}$ **then**
      $\mathbf{w} \leftarrow \mathbf{w} + y^{(i)}\mathbf{x}^{(i)}$
      changed = TRUE
    end if
  end for
  if Not Changed **then**
    Break
  end if
end while

Answer following questions:

i. (4 pts) Consistent with the lecture note, we use $\mathbf{w}_{new}$ to denote the weight $\mathbf{w}$ of perceptron after an update. Prove that

$$\mathbf{w}_{new}^{\top}\mathbf{w}^* \geq \mathbf{w}^{\top}\mathbf{w}^* + \gamma.$$

ii. (3 pts) Prove that when $a \geq 0$, $b \geq 0$, and $\gamma \geq 0$ if

$$a^2 \leq b^2 + b\gamma + 1,$$

then

$$a \leq b + \frac{1}{2b} + \frac{\gamma}{2}.$$

iii. (4 pts) Prove that

$$\|\mathbf{w}_{new}\| \leq \|\mathbf{w}\| + \frac{1}{2\|\mathbf{w}\|} + \frac{\gamma}{2}$$

iv. (3pt) We use M to denote the total number of updates that the adapted perceptron algorithm makes. Prove that if

$$M \geq \frac{2}{\gamma^2},$$

then

$$\exists t \leq M, \text{ s.t. after } t\text{-th updates, } \|\mathbf{w}\| \geq \frac{2}{\gamma}$$

v. (8pt) Prove that

$$M \leq \frac{8}{\gamma^2}$$

Hint: You might want to use the conclusion from the previous problem to solve the current one.

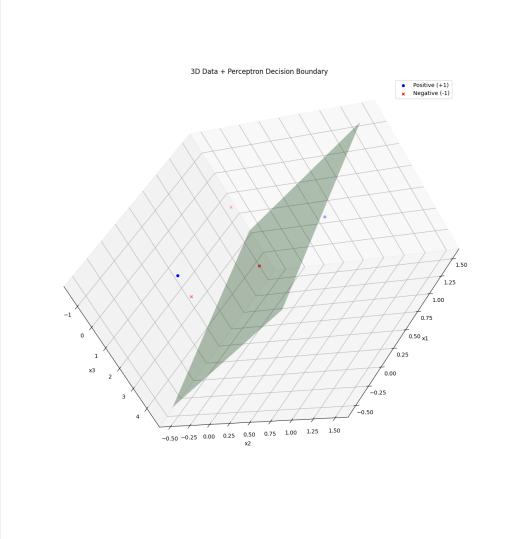
Solution.

(a) i. What is the vector $w \in \mathbb{R}^4$ after the first iteration over the first data point?

- $x^{(1)} = [1, 1, 1]$ $\quad y^{(1)} = 1$ (Label) $\quad w^{(1)} = [0, 0, 0, 0]$
- Augmented x: $\hat{x}^{(1)} = [1, 1, 1, 1]$
- $\because y^{(1)}(w^{(1)\intercal}\hat{x}^{(1)}) = 0 \quad \therefore w^{(1)} \leftarrow w^{(1)} + y^{(1)}\hat{x}^{(1)}$
- **Ans:** $w = [1, 1, 1, 1]$

ii. Compute the w after the algorithm converges.
Ans: w = [-2, 0, 0, 3]

iii. Is the perceptron algorithm still applicable after the labels for 2 and 3 are swapped?
Ans: No, the data becomes linearly inseparable as shown by the figure below.



iv. With the same set of training data, will the K-NN and perceptron algorithms always have the same results on the test set?
Ans: No.
The perceptron algorithm is a supervised learning method designed for linearly separable data, whereas the K-NN algorithm is a supervised, instance-based method that assigns labels based on the majority class of an input's nearest neighbors. When the data is not linearly separable, the perceptron may fail to find a suitable hyperplane and misclassify

points. In contrast, K-NN can often handle such cases more effectively by relying on local neighborhood information rather than a global linear boundary.

(b) i. Prove that $\mathbf{w}_{new}^{\top}\mathbf{w}^{*} \geq \mathbf{w}^{\top}\mathbf{w}^{*} + \gamma$.

$$\begin{aligned} w_{new}^{\intercal} w^{*} &= (w + yx)^{\intercal} w^{*} \\ &= w^{\intercal} w^{*} + y w^{*\intercal} x \\ &\because sign(y) = sign(w^{*\intercal} x) \\ &\because y w^{*T} x = |w^{*\intercal} x| \geq \gamma \\ &\geq w^{\intercal} w^{*} + \gamma \quad (\because y w^{*\intercal} x \geq \gamma) \end{aligned}$$

ii. Prove that when $a \geq 0$, $b \geq 0$, and $\gamma \geq 0$ if $a^2 \leq b^2 + b\gamma + 1$, then $a \leq b + \frac{1}{2b} + \frac{\gamma}{2}$.

$$\begin{aligned} a^2 &\leq b^2 + b\gamma + 1 \\ &\leq (b + \frac{\gamma}{2})^2 + 1 - \frac{\gamma^2}{4} \\ &\leq (b + \frac{\gamma}{2})^2 + 1 \\ &\because (b + \frac{1}{2b} + \frac{\gamma}{2})^2 = (b + \frac{\gamma}{2})^2 + \gamma(b + \frac{\gamma}{2}) + \frac{\gamma^2}{2} > (b + \frac{\gamma}{2})^2 + 1 \\ &\therefore a \leq b + \frac{1}{2b} + \frac{\gamma}{2} \end{aligned}$$

iii. Prove that $\|\mathbf{w}_{new}\| \leq \|\mathbf{w}\| + \frac{1}{2\|\mathbf{w}\|} + \frac{\gamma}{2}$.

$$\begin{aligned} \|w_{new}\|^2 &= \|w + yx\|^2 \\ &= \|w\|^2 + 2yw^{\intercal}x + \|yx\|^2 \end{aligned}$$

$\because$ The algorithm only updates when $yw^{\intercal}x \leq \frac{\gamma}{2}$ while $\|w\| = 1$

$\therefore$ $\|w_{new}\|^2 \leq \|w\|^2 + \gamma + 1$

Let $a = \|w_{new}\|$ and $b = \|w\|$.

Plug the formula in (ii) in and we derive the result :

$$\|\mathbf{w}_{new}\| \leq \|\mathbf{w}\| + \frac{1}{2\|\mathbf{w}\|} + \frac{\gamma}{2}$$

iv. Prove that if $M \geq \frac{2}{\gamma^2}$, then $\exists t \leq M$ s.t. after t^{th} updates, $\|w\| \geq \frac{2}{\gamma}$

From (i), we can derive that after t updates

$w_t^\intercal w^* \geq w_{t-1}^\intercal w^* + t\gamma \geq t\gamma$

$\because$

$w_t^\intercal w^* \leq ||w_t|| \cdot ||w^*||$ (Cauchy-Schwarz Inequality)

$||w_t|| \cdot ||w^*|| = ||w_t|| \cdot 1$ (since $||w^*|| = 1$)

$t\gamma \leq w_t^\intercal w^* \leq \|w_t\|$

$\therefore$

Given that t $\leq M$ and that M $\geq \frac{2}{\gamma^2}$, if $\|w\| < \frac{2}{\gamma}$ and t = M

$M\gamma \leq w_M^\intercal w^* \leq \|w_M\| < \frac{2}{\gamma}$

$\Rightarrow M < \frac{2}{\gamma^2} \Rightarrow\!\!\!\Leftarrow M \geq \frac{2}{\gamma^2}$

$\because \quad \|w\| < \frac{2}{\gamma} \Rightarrow M < \frac{2}{\gamma^2}$ and $M \geq \frac{2}{\gamma^2}$ (*contradiction*)

$\therefore \quad \|w\| \geq \frac{2}{\gamma}$

v. Prove $M \leq \frac{8}{\gamma^2}$.

$M\gamma \leq w^T w^*$

$\leq |w^T w^*|$

$\leq ||w|| \cdot ||w^*||$ (Cauchy-Schwarz Inequality)

$\leq ||w|| \cdot 1$ (since $||w^*|| = 1$)

$\leq ||w||$

$\because$

$\|w_{t+1}\|^2 \leq \|w_t\|^2 + \gamma + 1$ (squared version used in (iii))

$\because \|w^*\| = 1$, $\|x\| \leq 1$, and $\gamma \leq \|w^*\|\|x\| \Rightarrow \gamma \leq 1$

$\therefore \gamma + 1 \leq 2$, and with $\|w_0\| = 0$

$\Rightarrow \|w_M\|^2 \leq 2M$

$\Rightarrow \|w_M\| \leq \sqrt{2M}$

$\therefore \quad M\gamma \leq w_M^\intercal w^* \leq \|w_M\| \leq \sqrt{2M}$

$\Rightarrow M \leq \frac{\|w\|^2}{\gamma}$

$\Rightarrow M\gamma \leq \sqrt{2M}$

$\Rightarrow M^2\gamma^2 \leq 2M$

$\Rightarrow M \leq \frac{2}{\gamma^2}$

$\Rightarrow M \leq \frac{8}{\gamma^2}$

3. MLE, MAP (40 Pt.)

(a) (10 Pt.)

i. (4 pt) Let $X \sim \text{Triangle}(a, b)$ where a is a given real number and b is an unknown parameter, $a < b$. We observe N draws $\mathcal{D} = \{x^{(i)}\}_{i=1}^{N}$. Find the MLE estimate $\widehat{b}$ of b. Here Triangle(a, b) is such a distribution where the PDF (probability density function) is

$$p(x|a,b) = \begin{cases} \frac{2(x-a)}{(b-a)^2} & \text{for } x \in [a,b]\,, \\ 0 & \text{for } x \notin [a,b]\,. \end{cases}$$

ii. For the discrete random variable Z, we have an unknown distribution $p(Z|X)$ where X is a discrete parameter. After drawing 100 samples, the numbers of observation (Z, X) are as follows:

	$X=1$	$X=2$	$X=3$
$Z=1$	18	7	6
$Z=2$	9	12	3
$Z=3$	10	2	9
$Z=4$	3	19	2

(*Hint:* You are supposed to estimate $P(Z \mid X)$ from the observed values and then use your estimated $P(Z \mid X)$ to solve the following questions)

A. Given that a measurement $Z = 3$ has been taken, what is the MLE for X? (3 pt)

B. Following (a), we have prior probabilities (obtained from elsewhere) as follows: $P(X = 1) = 0.2, P(X = 2) = 0.5, P(X = 3) = 0.3$. What's the MAP estimate for X? (3 pt)

(b) (15 Pt.)

i. The probability mass function of a distribution is as follows:

$$P(k|\lambda) := \frac{\lambda^k e^{-\lambda}}{k!}, \quad k \in \mathbb{Z}.$$

Let $K = \{k^{(i)}\}_{i=1}^{N}$ be an i.i.d sample drawn from this distribution with parameter λ. Derive the MLE estimate $\widehat{\lambda}^{\text{MLE}}$ of λ based on this sample K. (6 pt)

ii. Following (i), we have a Gamma distribution

$$p(\lambda) := \frac{\lambda^{\alpha-1} e^{-\lambda/\beta}}{\Gamma(\alpha)\beta^{\alpha}}$$

as a prior for λ, where $\Gamma(\cdot)$ is the Gamma function, $\alpha > 1$ and $\beta > 0$. Derive the MAP estimate $\widehat{\lambda}^{\text{MAP}}$ of λ. (6 pt)

iii. Following (i),(ii) what happens to $\widehat{\lambda}^{\text{MAP}}$ when the sample size N goes to infinity? (Hint : Consider how do they relate to $\widehat{\lambda}^{\text{MLE}}$.) (3 pt)

(c) (15 Pt.)

i. Suppose we have N independent sonar measurements $Z = \{z^{(i)}\}_{i=1}^{N}$ of the 1-d position x, and the sensor error may be modelled as $p(z^{(i)}|x) = \frac{1}{\sigma_i\sqrt{2\pi}} e^{-\frac{(z^{(i)}-x)^2}{2\sigma_i^2}}$ for $i = 1, 2, ..., N$. Derive the MLE estimate $\widehat{x}^{\text{MLE}}$ of x based on this sample Z. (6 pt)

ii. Following (3.1), we have a Gaussian prior $x \sim \mathcal{N}(\theta_0, \sigma_0^2)$. Derive the MAP estimate $\widehat{x}^{\text{MAP}}$ of x. (6 pt)

iii. Following (i),(ii) what happens to $\widehat{x}^{\text{MAP}}$ when the sample size N goes to infinity? (Hint: consider how do they relate to $\widehat{x}^{\text{MLE}}$.) (3 pt)

Solution.

(a) i. Find the MLE estimate $\hat{b}$ of b.
Ans: $\hat{b}_{MLE} = x_{max} = \max_i x^{(i)}$

$$\begin{aligned} L(b) &= \prod_{i=1}^{N} p(x^{(i)} \mid a, b) \\ &= \prod_{i=1}^{N} \frac{2(x^{(i)} - a)}{(b-a)^2} \\ &= (b-a)^{-2N} \prod_{i=1}^{N} 2(x^{(i)} - a) \end{aligned}$$

$$\ln L(b) = -2N \ln(b-a) + \sum_{i=1}^{N} \ln 2(x^{(i)} - a) \quad \text{(Monotonicity of Logarithm)}$$

$$\frac{\partial \ln L(b)}{\partial b} = -\frac{2N}{(b-a)}$$

$\because \quad \frac{\partial \ln L(b)}{\partial b} < 0 \Rightarrow$ L(b) decreases with b

$\therefore$maximum over b is the smallest feasible b:

$$\hat{b}_{MLE} = x_{max} = \max_i x^{(i)}$$

ii. A. Given that a measurement Z = 3 has been taken, what is the MLE for X?
Ans: 3
Column Totals: $n_{X=1} = 40 \quad n_{X=2} = 40 \quad n_{X=3} = 20$

$$\begin{cases} \hat{P}(Z = 3 \mid X = 1) = \frac{10}{40} = 0.25 \\ \hat{P}(Z = 3 \mid X = 2) = \frac{2}{40} = 0.05 \\ \hat{P}(Z = 3 \mid X = 3) = \frac{9}{20} = 0.45 \end{cases}$$

$\because$ MLE ignores priors

$\therefore \quad X_{MLE} = 3$

B. Following (a), we have prior probabilities (obtained from elsewhere) as follows: P (X = 1) = 0.2, P (X = 2) = 0.5, P (X = 3) = 0.3. What's the MAP estimate for X?
Ans: 3

$$\begin{cases} P(Z = 3, X = 1) = 0.25 \times 0.20 = 0.05 \\ P(Z = 3, X = 2) = 0.05 \times 0.50 = 0.025 \\ P(Z = 3, X = 3) = 0.45 \times 0.30 = 0.135 \end{cases}$$

Normalize and pick the highest as the MAP.

$$P(Z) = 0.05 + 0.025 + 0.135 = 0.21$$

$$\begin{cases} P(X = 1 \mid Z = 3) = 0.050 \approx 0.2380952381 \\ P(X = 2 \mid Z = 3) = 0.025 \approx 0.119047619 \\ P(X = 3 \mid Z = 3) = 0.135 \approx 0.6428571429 \end{cases}$$

$$\Rightarrow X_{MAP} = 3$$

(b) i. Derive the MLE estimate $\hat{\lambda}$ of λ
Ans: $\hat{\lambda}_{MLE} = \bar{k} \quad (\bar{k} = \frac{1}{N}\sum_{i=1}^{N} k^{(i)})$

$$
\begin{aligned}
L(\lambda) &= \prod_{i=1}^{N} P(k^{(i)} \mid \lambda) \\
&= \frac{\lambda^{k^{(1)}} e^{-\lambda}}{k^{(1)}!} \cdot \frac{\lambda^{k^{(2)}} e^{-\lambda}}{k^{(2)}!} \cdot \frac{\lambda^{k^{(3)}} e^{-\lambda}}{k^{(3)}!} \cdots \\
&= e^{-\lambda N} \cdot \lambda^{\bar{k}N} \cdot \frac{1}{k^{(1)}!k^{(2)}!k^{(3)}! \ldots} \quad (\bar{k} = \frac{1}{N}\sum_{i=1}^{N} k^{(i)}) \\
\text{Let } C &= \frac{1}{k^{(1)}!k^{(2)}!k^{(3)}! \ldots} \\
&= e^{-\lambda N} \cdot \lambda^{\bar{k}N} \cdot C \\
\ln L(\lambda) &= -\lambda N + \bar{k}N \ln \lambda + \ln C \\
\frac{\partial \ln L(\lambda)}{\partial \lambda} &= -N + \frac{\bar{k}N}{\lambda} \\
&\because \quad \text{Extrema occur where derivatives} = 0 \\
&\therefore \quad -N + \frac{\bar{k}N}{\hat{\lambda}_{MLE}} = 0 \\
&\Rightarrow \hat{\lambda}_{MLE} = \bar{k}
\end{aligned}
$$

ii. Derive the MAP estimate $\hat{\lambda}_{MAP}$ of λ.
Ans: $\hat{\lambda}_{MAP} = \frac{\alpha N\bar{k}-1}{N+1/\beta}$

The relationship between posteriors and priors.

$$
\begin{aligned}
&p(\lambda \mid k) \propto p(k \mid \lambda)p(\lambda) \quad \text{(x is the input data)} \\
&p(\lambda \mid k) \propto L(\lambda)p(\lambda)
\end{aligned}
$$

Likelihood

$$
\begin{aligned}
L(\lambda) &= \prod_{i=1}^{N} \frac{\lambda^{k^{(i)}} e^{-\lambda}}{k^{(i)}!} \\
&\propto \lambda^{N\bar{k}} e^{-N\lambda} \quad (\bar{k} = \frac{1}{N}\sum_{i=1}^{N} k^{(i)})
\end{aligned}
$$

Posterior kernel

$$
\begin{aligned}
p(\lambda \mid k) &\propto L(\lambda)p(\lambda) \\
&= \lambda^{\alpha+N\bar{k}-1} e^{-(N+1/\beta)\lambda}
\end{aligned}
$$

Log-form and derivative (set to 0)

$$
\begin{aligned}
l(\lambda) &= \ln L(\lambda)p(\lambda) = (a + N\bar{k} - 1)\ln\lambda - (N + 1/\beta)\lambda \\
l'(\lambda) &= \frac{(a + N\bar{k} - 1)}{\lambda} - (N + 1/\beta) = 0 \\
&\Rightarrow (a + N\bar{k} - 1) = (N + 1/\beta)\hat{\lambda}_{MAP} \\
&\Rightarrow \hat{\lambda}_{MAP} = \frac{\alpha N\bar{k} - 1}{N + 1/\beta}
\end{aligned}
$$

iii. What happens as $N \to \infty$?
Ans: As $N \to \infty$, $\hat{\lambda}_{MLE} \to \hat{\lambda}_{MAP} \to$ true population mean

$$
\begin{aligned}
\hat{\lambda}_{MAP} - \hat{\lambda}_{MLE} &= \frac{N}{N + \frac{1}{\beta}}\hat{k} + \frac{\alpha - 1}{N + \frac{1}{\beta}} - \hat{k} \\
&= \frac{\alpha - 1 - (\frac{1}{\beta})\hat{k}}{N + \frac{1}{\beta}} \\
&= 0 \quad (N \to \infty)
\end{aligned}
$$

Therefore, the difference between $\hat{\lambda}_{MAP}$ and $\hat{\lambda}_{MLE}$ shrinks to 0 as $N \to \infty$, which indicates that the sample mean $\hat{k}$ approaches the true population mean as the sample size N approaches infinity.

(c) i. Derive the MLE estimate $\hat{x}_{MLE}$.
Ans: $\hat{x}_{MLE} = \bar{z} \quad (\bar{z} = \frac{1}{N}\sum_{i=1}^{N} z^{(i)})$

$$
\begin{aligned}
L(x) &= \prod_{i=1}^{N} \frac{1}{\sigma_i\sqrt{2\pi}} e^{-\frac{(z^{(i)} - x)^2}{2\sigma_i^2}} \\
\ln L(x) &= -\sum \ln(\sigma_i\sqrt{2\pi}) - \sum_{i=1}^{N} \frac{(z^{(i)} - x)^2}{2\sigma_i^2} \\
\frac{d\ln L(x)}{dx} &= \sum_{i=1}^{N} \frac{z^{(i)} - x}{\sigma_i^2} \\
\text{Let } \frac{d\ln L(x)}{dx} &= 0 \\
N\hat{x}_{MLE} &= N\bar{z} \quad (\bar{z} = \frac{1}{N}\sum_{i=1}^{N} z^{(i)}) \\
\hat{x}_{MLE} &= \bar{z}
\end{aligned}
$$

ii. Derive the MAP estimate $\hat{x}_{MAP}$.
Ans: $\Rightarrow \hat{x}_{MAP} = \frac{\frac{\theta_0}{\sigma_0^2} + \sum_{i=1}^{N} \frac{z^{(i)}}{\sigma_i^2}}{\frac{1}{\sigma_0^2} + \sum_{i=1}^{N} \frac{1}{\sigma_i^2}}$

Likelihood

$$p(Z \mid x) \propto e^{-\frac{1}{2}\sum_{i=1}^{N} \frac{(z^{(i)} - x)^2}{\sigma_i^2}}$$

Prior

$$p(x) \propto e^{\frac{(x - \theta_0)^2}{2\sigma_0^2}}$$

Posterior and log-posterior

$$
\begin{aligned}
p(x \mid Z) &\propto p(x)p(Z \mid x) \\
&= e^{\frac{(x-\theta_0)^2}{2\sigma_0^2} + \frac{1}{2}\sum_{i=1}^{N} \frac{(z^{(i)}-x)^2}{\sigma_i^2}} \\
l(x) &= -\frac{1}{2}\left[\frac{(x-\theta_0)^2}{2\sigma_0^2} + \sum_{i=1}^{N} \frac{(z^{(i)}-x)^2}{\sigma_i^2}\right]
\end{aligned}
$$

Differentiate and set to zero:

$$
\begin{aligned}
l'(x) &= \frac{x-\theta_0}{\sigma_0^2} + \sum_{i=1}^{N} \frac{z^{(i)}-x}{\sigma_i^2} = 0 \\
&\Rightarrow \hat{x}_{MAP}\left(\frac{1}{\sigma_0^2} + \sum_{i=1}^{N} \frac{1}{\sigma_i^2}\right) = \frac{\theta_0}{\sigma_0^2} + \sum_{i=1}^{N} \frac{z^{(i)}}{\sigma_i^2} \\
&\Rightarrow \hat{x}_{MAP} = \frac{\frac{\theta_0}{\sigma_0^2} + \sum_{i=1}^{N} \frac{z^{(i)}}{\sigma_i^2}}{\frac{1}{\sigma_0^2} + \sum_{i=1}^{N} \frac{1}{\sigma_i^2}}
\end{aligned}
$$

iii. What happens to $\hat{x}_{MAP}$ when $N \to \infty$?
Ans: As $N \to \infty$, $\hat{x}_{MAP} \to \hat{x}_{MLE}$
Rewrite

$$
\begin{aligned}
&\text{Let } W = \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \\
&\hat{x}_{MLE} = \frac{\sum_{i=1}^{N} z^{(i)}/\sigma_i^2}{1/\sigma_i^2} \\
&\hat{x}_{MAP} = \frac{W\hat{x}_{MLE} + \frac{\theta_0}{\sigma_0^2}}{W + \frac{1}{\sigma_0^2}} = \frac{W}{W + \frac{1}{\sigma_0^2}}\hat{x}_{MLE} + \frac{\frac{1}{\sigma_0^2}}{W + \frac{1}{\sigma_0^2}}\theta_0 \\
&\text{As } N \to \infty \Rightarrow W \to \infty \Rightarrow \frac{W}{W + \frac{1}{\sigma_0^2}} \to 1 \\
\therefore \quad &\hat{x}_{MAP} \to \hat{x}_{MLE} \quad (N \to \infty)
\end{aligned}
$$