INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

Stamp / Signature of the Invigilator

EXAMINATION (End Semester)	**SEMESTER (Spring 2023-2024)**

Roll Number										**Section**		**Name**	

Subject Number	C	S	6	0	0	5	0	**Subject Name**	*MACHINE LEARNING*

Department / Center of the Student		**Additional sheets**	

Important Instructions and Guidelines for Students

1. You must occupy your seat as per the Examination Schedule/Sitting Plan.
2. Do not keep mobile phones or any similar electronic gadgets with you even in the switched off mode.
3. Loose papers, class notes, books or any such materials must not be in your possession, even if they are irrelevant to the subject you are taking examination.
4. Data book, codes, graph papers, relevant standard tables/charts or any other materials are allowed only when instructed by the paper-setter.
5. Use of instrument box, pencil box and non-programmable calculator is allowed during the examination. However, exchange of these items or any other papers (including question papers) is not permitted.
6. Write on both sides of the answer script and do not tear off any page. **Use last page(s) of the answer script for rough work.** Report to the invigilator if the answer script has torn or distorted page(s).
7. It is your responsibility to ensure that you have signed the Attendance Sheet. Keep your Admit Card/Identity Card on the desk for checking by the invigilator.
8. You may leave the examination hall for wash room or for drinking water for a very short period. Record your absence from the Examination Hall in the register provided. Smoking and the consumption of any kind of beverages are strictly prohibited inside the Examination Hall.
9. Do not leave the Examination Hall without submitting your answer script to the invigilator. **In any case, you are not allowed to take away the answer script with you.** After the completion of the examination, do not leave the seat until the invigilators collect all the answer scripts.
10. During the examination, either inside or outside the Examination Hall, gathering information from any kind of sources or exchanging information with others or any such attempt will be treated as **'unfair means'**. Do not adopt unfair means and do not indulge in unseemly behavior.

Violation of any of the above instructions may lead to severe punishment.

Signature of the Student

To be filled in by the examiner

Question Number	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**Total**
Marks Obtained												

Marks obtained (in words)	**Signature of the Examiner**	**Signature of the Scrutineer**

Indian Institute of Technology Kharagpur
Department of Computer Science and Engineering

End-Semester Exam | **Machine Learning (CS60050)** | **Spring 2023-2024**

Date: 18-Apr-2024 (Thu, AN) | **Answer *all* questions.** | **Maximum Marks:** 100

— Write your answers at indicated places inside the question paper. —

— This page is kept blank intentionally. —

— The question paper starts from the next page. —

Q1. [Gradient Descent] **10 marks**

Let $\phi(x) : \mathbb{R} \to \mathbb{R}^d$, $\mathbf{w} \in \mathbb{R}^d$. Consider the following objective function (a.k.a. loss function).

$$Loss(x, y; \mathbf{w}) = \begin{cases} 1 - 2(\mathbf{w} \cdot \phi(x))y, & \text{if } (\mathbf{w} \cdot \phi(x))y \leq 0 \\ \big(1 - 2(\mathbf{w} \cdot \phi(x))y\big)^2, & \text{if } 0 < (\mathbf{w} \cdot \phi(x))y \leq 1 \\ 0, & \text{if } (\mathbf{w} \cdot \phi(x))y > 1 \end{cases} \quad \text{where } y \in \mathbb{R}.$$

(a) Compute the gradient $\nabla_{\mathbf{w}} Loss(x, y; \mathbf{w})$. **(2)**

Solution:

We apply the rules to compute the gradient for each case separately, leading to the following piece-wise function for the gradient.

$$\nabla_{\mathbf{w}} Loss(x, y; \mathbf{w}) = \begin{cases} -2\phi(x)y, & \text{if } (\mathbf{w} \cdot \phi(x))y \leq 0 \\ -4\big(1 - 2(\mathbf{w} \cdot \phi(x))y\big)\phi(x)y, & \text{if } 0 < (\mathbf{w} \cdot \phi(x))y \leq 1 \\ 0, & \text{if } (\mathbf{w} \cdot \phi(x))y > 1 \end{cases} \quad \text{where } y \in \mathbb{R}.$$

(b) Let the Gradient Descent update rule for some function $TrainLoss(\mathbf{w}) : \mathbb{R}^d \to \mathbb{R}$ is given as, $\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla_{\mathbf{w}} TrainLoss(\mathbf{w})$, where η is the step size. Let $d = 2$ and $\phi(x) = [1, x]$. Consider the following loss function, $TrainLoss(\mathbf{w}) = \frac{1}{2}\big(Loss(x_1, y_1; \mathbf{w}) + Loss(x_2, y_2; \mathbf{w})\big)$. Compute $\nabla_{\mathbf{w}} TrainLoss(\mathbf{w})$ for the values of $x_1 = -2$, $y_1 = 1$, $x_2 = -1$, $y_2 = -1$, $\mathbf{w} = \big[0, \frac{1}{2}\big]$. **(3)**

Solution:

To calculate $\nabla_{\mathbf{w}} Loss(x_1, y_1; \mathbf{w})$, first note that $\phi(x_1) = [1, -2]$. Since $\big(\mathbf{w} \cdot \phi(x_1)\big)y_1 = -1$, we consider Case 1 of the gradient expression equation (derived in Part (a)). Thus, we have,

$$\nabla_{\mathbf{w}} Loss(x_1, y_1; \mathbf{w}) = -2\phi(x_1)y_1 = [-2, 4]$$

To calculate $\nabla_{\mathbf{w}} Loss(x_2, y_2; \mathbf{w})$, similarly note that $\phi(x_2) = [1, -1]$. Since $\big(\mathbf{w} \cdot \phi(x_2)\big)y_2 = \frac{1}{2}$, we consider Case 2 of the gradient expression equation (derived in Part (a)). Thus, we have,

$$\nabla_{\mathbf{w}} Loss(x_2, y_2; \mathbf{w}) = -4\big(1 - 2(\mathbf{w} \cdot \phi(x_2))y_2\big)\phi(x_2)y_2 = [0, 0]$$

Combining the terms, we get,

$$\begin{aligned} \nabla_{\mathbf{w}} TrainLoss(\mathbf{w}) &= \frac{1}{2}\big(\nabla_{\mathbf{w}} Loss(x_1, y_1; \mathbf{w}) + \nabla_{\mathbf{w}} Loss(x_2, y_2; \mathbf{w})\big) \\ &= \frac{1}{2}\big([-2, 4] + [0, 0]\big) = \Big[-1, 2\Big] \end{aligned}$$

(c) Perform two iterations of Gradient Descent to minimize the objective function, $TrainLoss(\mathbf{w}) = \frac{1}{2}\big(Loss(x_1, y_1; \mathbf{w}) + Loss(x_2, y_2; \mathbf{w})\big)$ with initial values for x_1, y_1, x_2, y_2 as given in Part (b) above. Show the calculations. Use initialization $\mathbf{w}^{(0)} = \big[0, \frac{1}{2}\big]$ and step size $\eta = \frac{1}{2}$ and derive what are the weight parameters, $\mathbf{w}^{(1)}$ and $\mathbf{w}^{(2)}$, after each of these two iterations. **(5)**

Solution:

Note that, we have already computed $\nabla_{\mathbf{w}} TrainLoss(\mathbf{w})$ at the initialization point $\mathbf{w}^{(0)}$ in Part (b).

$$\begin{aligned} \mathbf{w}^{(1)} &\leftarrow \mathbf{w}^{(0)} - \eta \nabla_{\mathbf{w}} TrainLoss(\mathbf{w}) \text{ at } \mathbf{w}^{(0)} \\ &= \Big[0, \frac{1}{2}\Big] - \Big(\frac{1}{2}\Big)\big[-1, 2\big] \quad = \quad \Big[\frac{1}{2}, -\frac{1}{2}\Big] \end{aligned}$$

Now we need to compute $\nabla_{\mathbf{w}} Loss(x_1, y_1; \mathbf{w})$ and $\nabla_{\mathbf{w}} Loss(x_2, y_2; \mathbf{w})$ at the new iteration (updated) weight $\mathbf{w}^{(1)}$. We repeat the process that we did for Part (b) by applying the piece-wise defined gradient (derived in Part (a)) to the two points, this time setting $\mathbf{w} \leftarrow \mathbf{w}^{(1)}$.

Since $\big(\mathbf{w}^{(1)} \cdot \phi(x_1)\big) y_1 = \frac{3}{2} > 1$, we consider Case 3 of the gradient expression equation (derived in Part (a)). Thus, we have,

$$\nabla_{\mathbf{w}} Loss(x_1, y_1; \mathbf{w}) = \big[0, 0\big]$$

Since $\big(\mathbf{w}^{(1)} \cdot \phi(x_2)\big) y_2 = -1 \leq 0$, we consider Case 1 of the gradient expression equation (derived in Part (a)). Thus, we have,

$$\nabla_{\mathbf{w}} Loss(x_2, y_2; \mathbf{w}) = -2\phi(x_2) y_2 = [2, -2]$$

Combining the terms, we get,

$$\begin{aligned} \nabla_{\mathbf{w}} TrainLoss(\mathbf{w}) \text{ at } \mathbf{w}^{(1)} &= \frac{1}{2}\big(\nabla_{\mathbf{w}} Loss(x_1, y_1; \mathbf{w}) + \nabla_{\mathbf{w}} Loss(x_2, y_2; \mathbf{w})\big) \\ &= \frac{1}{2}\Big(\big[0, 0\big] + [2, -2]\Big) = \big[1, -1\big] \end{aligned}$$

Hence,

$$\begin{aligned} \mathbf{w}^{(2)} &\leftarrow \mathbf{w}^{(1)} - \eta \nabla_{\mathbf{w}} TrainLoss(\mathbf{w}) \text{ at } \mathbf{w}^{(1)} \\ &= \Big[\frac{1}{2}, -\frac{1}{2}\Big] - \Big(\frac{1}{2}\Big)\big[1, -1\big] \quad = \quad \big[0, 0\big] \end{aligned}$$

Q2. [Learning Theory: VC-Dimension] **13 marks**

(a) Decision trees can split on data with binary features ($X = \{0,1\}^d$) or continuous features ($X = \mathbb{R}^d$). Assume that the nodes of a continuous decision tree have splitting rules that threshold the value of a single feature. Note that for continuous decision trees, multiple splits can be made on the same feature. For binary decision trees, only a single split can be made on a feature.

Consider the following two hypothesis classes and answer the questions asked below.

- $\mathscr{H}_1 = \{h : h \text{ is a decision tree for data with only binary features}\}$
- $\mathscr{H}_2 = \{h : h \text{ is a decision tree for data with only continuous features}\}$

(i) What is the VC-dimension of $\mathscr{H}_1$? Analytically derive/explain. **(3)**

(ii) What is the VC-dimension of $\mathscr{H}_2$? Analytically derive/explain. **(3)**

(iii) Are both of these $\mathscr{H}_1$ and $\mathscr{H}_2$ PAC learnable? Explain your answer. **(2)**

Solution:

(i) The VC-dimension is 2^d.

With d binary attributes, there are 2^d possible input values. The set of size 2^d with each possible input can be shattered by splitting on every feature to isolate each example to its own leaf. Then, any possible labeling can be realized as we can choose the label assigned to each leaf. This is sufficient for the set to be shattered.
There is no set of size $2^d + 1$, thus 2^d is the largest size set that can be shattered and $VCDim(\mathscr{H}_1) = 2^d$.

Note: *To be specific, there is no set of size $2^d + 1$ possible.*
On contrary, the following reasoning is actually wrong and should get only partial credit: Any larger set will have two examples with the same attribute values. The tree must then always predict the same label for these examples so the set cannot be shattered. As a set of size 2^d can be shattered but not a set of size $2^d + 1$, the VC dimension is 2^d.

(ii) The VC-dimension is ∞ (infinite).

The VC-dimension is infinite if for any size m, there is a set C that is shattered by $\mathscr{H}_2$. Consider a set C of size m, where all points have a unique value for attribute x_0 and all other attributes are 0. (This effectively reduces the problem into a single dimension $\mathbb{R}$). Then, repeatedly choosing nodes that create splits in between each of the points' attribute values of x_0 isolates each example to its own leaf, so $\mathscr{H}_2$ shatters C. This holds for a set C for arbitrary size m, so $VCDim(\mathscr{H}_2) = \infty$.

(iii) Any hypothesis class is PAC learnable if and only if it has a finite VC-dimension. Therefore, we can conclude that $\mathscr{H}_1$ is PAC-learnable but $\mathscr{H}_2$ is not.

(b) Let the VC-dimensions of two hypothesis classes, H_1 and H_2, be $VCDim(H_1) = d_1$ and $VCDim(H_2) = d_2$. Prove that, the VC-Dimension of the union of these hypothesis, i.e. $H = H_1 \cup H_2$, will be at most $(d_1 + d_2 + 1)$, i.e. $VCDim(H) \leq VCDim(H_1) + VCDim(H_2) + 1$. **(5)**

Solution:

By the definition of growth function on any N points for a hypothesis class $\mathscr{H}$, we know that, $m_{\mathscr{H}}(N) \leq \sum_{i=0}^{d_{VC}} \binom{N}{i}$, where $VCDim(\mathscr{H}) = d_{VC}$ is the VC-dimension of $\mathscr{H}$.

Let the growth functions on any N points of the hypothesis classes, H_1, H_2 and H, are denoted by $m_{H_1}(N)$, $m_{H_2}(N)$ and $m_H(N)$, respectively. Since we have $H = H_1 \cup H_2$, we can write

$$m_H(N) \leq m_{H_1}(N) + m_{H_2}(N).$$

Taking $N = d_1 + d_2 + 2$, we can proceed as follows:

$$\begin{aligned}
m_H(N) \leq m_{H_1}(N) + m_{H_2}(N) \quad &\leq \quad \sum_{i=0}^{d_1} \binom{N}{i} + \sum_{i=0}^{d_2} \binom{N}{i} \\
&= \quad \sum_{i=0}^{d_1} \binom{d_1+d_2+2}{i} + \sum_{i=0}^{d_2} \binom{d_1+d_2+2}{i} \\
&= \quad \sum_{i=0}^{d_1} \binom{d_1+d_2+2}{i} + \sum_{i=0}^{d_2} \binom{d_1+d_2+2}{d_1+d_2+2-i} \\
&= \quad \sum_{i=0}^{d_1} \binom{d_1+d_2+2}{i} + \sum_{i=d_1+2}^{d_1+d_2+2} \binom{d_1+d_2+2}{i} \\
&= \quad \sum_{i=0}^{d_1+d_2+2} \binom{d_1+d_2+2}{i} - \binom{d_1+d_2+2}{d_1+1} \\
&= \quad 2^{d_1+d_2+2} - \binom{d_1+d_2+2}{d_1+1} \\
&< \quad 2^{d_1+d_2+2} \quad = \quad 2^N \\
\implies m_H(d_1+d_2+2) < 2^{d_1+d_2+2} \quad &\implies \quad (d_1+d_2+2) \text{ is a break point of } H \\
&\implies \quad VCDim(H) \leq d_1+d_2+1
\end{aligned}$$

Therefore, $VCDim(H) \leq VCDim(H_1) + VCDim(H_2) + 1$. *[Proved]*

Q3. [Kernels and Support Vector Machines] **10 marks**

(a) One of the most commonly used kernels in Support Vector Machines (SVM) is the Gaussian Radial Basis Function (RBF) kernel: $k(a,b) = \exp\big(-\frac{||a-b||^2}{2\sigma}\big)$. Suppose we have three points, z_1, z_2, and x. z_1 is geometrically very close to x, and z_2 is geometrically far away from x. What are the values of $k(z_1,x)$ and $k(z_2,x)$? Choose the correct option with suitable justification: **(2)**

(i) $k(z_1,x)$ will be close to 1 and $k(z_2,x)$ will be close to 0.

(ii) $k(z_1,x)$ will be close to 0 and $k(z_2,x)$ will be close to 1.

(iii) $k(z_1,x)$ will be close to c_1, $c_1 \gg 1$ and $k(z_2,x)$ will be close to c_2, $c_2 \ll 0$ ($c_1, c_2 \in \mathbb{R}$).

(iv) $k(z_1,x)$ will be close to c_1, $c_1 \ll 0$ and $k(z_2,x)$ will be close to c_2, $c_2 \gg 1$ ($c_1, c_2 \in \mathbb{R}$).

Solution: (i)

RBF kernel generates a 'bump' around the center x. For points z_1 close to the center of the bump, $k(z_1,x)$ will be close to 1, for points away from the center of the bump $k(z_2,x)$ will be close to 0.

(b) Suppose You are training a RBF soft-margin SVM classifier with the following parameters: ξ (slack penalty) and σ (where σ^2 is the variance of the RBF kernel) and found that it is overfitting. How should you tweak the parameters (only ξ and σ) to reduce overfitting? **(2)**

Solution:

Reduce ξ and / or increase σ

(c) Multiple kernels can be combined to produce new kernels. For example, $K(x,z) = K_1(x,z) + K_2(x,z)$ is a valid kernel combination. Suppose, the kernels K_1 and K_2 have the associated feature transformations Φ_1 and Φ_2, respectively. Derive what will be the corresponding feature transform for the following kernel combination: $K(x,z) = \alpha.K_1(x,z) + \beta.K_2(x,z)$, where $\alpha, \beta \in \mathbb{R}^+$ are positive constants. Here, derive $\Phi(x)$ as a feature vector with $\Phi_1(x)$ and $\Phi_2(x)$. **(2)**

Solution:

Note that,

$$\begin{aligned} K(x,z) &= \Phi(x).\Phi(z) = \big[\sqrt{\alpha}.\Phi_1(x), \sqrt{\beta}.\Phi_2(x)\big].\big[\sqrt{\alpha}.\Phi_1(z), \sqrt{\beta}.\Phi_2(z)\big]^T \\ &= \alpha.\Phi_1(x).\Phi_1(z) + \beta.\Phi_2(x).\Phi_2(z) = \alpha.K_1(x,z) + \beta.K_2(x,z) \end{aligned}$$

Therefore, $\Phi(x) = \big[\sqrt{\alpha}.\Phi_1(x), \sqrt{\beta}.\Phi_2(x)\big]$.

(d) A kernel function $K(x,z)$ measures the similarity between two instances x and z in a transformed space. For a feature transform $x \rightarrow \Phi(x)$ the kernel function is $K(x,z) = \Phi(x).\Phi(z)$. Consider the 2-dimensional input vectors $x = (x_1, x_2)$. Derive what will be the corresponding feature transform for each of the kernel function listed below.

(i) For $K(x,z) = 1 + x.z$, derive $\Phi(x)$ as a feature vector with x_1 and x_2. **(2)**

Solution:

Note that,

$$K(x,z) = \Phi(x).\Phi(z) = [1, x_1, x_2].[1, z_1, z_2]^T = 1 + x_1.z_1 + x_2.z_2 = 1 + x.z$$

Therefore, $\Phi(x) = [1, x_1, x_2]$.

(ii) For $K(x,z) = (1 + x.z)^2$, derive $\Phi(x)$ as a feature vector with x_1 and x_2. **(2)**

Solution:

Note that,

$$\begin{aligned} K(x,z) &= \Phi(x).\Phi(z) \\ &= [1, x_1^2, x_2^2, \sqrt{2}x_1x_2, \sqrt{2}x_1, \sqrt{2}x_2].[1, z_1^2, z_2^2, \sqrt{2}z_1z_2, \sqrt{2}z_1, \sqrt{2}z_2]^T \\ &= 1 + x_1^2z_1^2 + x_2^2z_2^2 + 2x_1x_2z_1z_2 + 2x_1z_1 + 2x_2z_2 \\ &= (1 + x_1.z_1 + x_2.z_2)^2 = (1 + x.z)^2 \end{aligned}$$

Therefore, $\Phi(x) = [1, x_1^2, x_2^2, \sqrt{2}x_1x_2, \sqrt{2}x_1, \sqrt{2}x_2]$.

Q4. [Classifier Evaluation] **7 marks**

You wrote a spam filtering program by yourself and now you are testing your program on 100 emails among which you already knew that 20% emails are spams. However, upon running your program on those 100 email corpus, it predicted $\frac{1}{2}$ of the 'spam' emails as non-spam. Answer the following:

(a) In order to push the *Accuracy* $\geq 75\%$, how many 'non-spam' emails at most (maximum) you can afford to mis-predict as spams? **(2)**

Solution:

Suppose, the spam filtering program can afford to mis-predict at most M 'non-spam' emails as spams. As per the problem, among 100 total test emails, 20 are actual spams, and hence $TP = FN = 10$. So, we have, $FP = M$ and $TN = 80 - M$.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{10+(80-M)}{100} \geq \frac{3}{4} \implies M \leq 15$$

So, this spam filtering program can afford to mis-predict at most 15 'non-spam' emails as spams.

(b) With the derived setup in Part (a), i.e., when your *Accuracy* is exactly 75%, present the confusion matrix (in tabular form). **(2)**

Solution:

		(Actual)	**(Actual)**
	Confusion Matrix	**Spam Emails**	**Non-Spam Emails**
(Predicted)	**Spam Emails**	10 (TP)	15 (FP)
(Predicted)	**Non-Spam Emails**	10 (FN)	65 (TN)

Alternative Approach:

		(Actual)	**(Actual)**
	Confusion Matrix	**Non-Spam Emails**	**Spam Emails**
(Predicted)	**Non-Spam Emails**	65 (TP)	10 (FP)
(Predicted)	**Spam Emails**	15 (FN)	10 (TN)

(c) As per your confusion matrix that you presented in Part (b), calculate *Precision*, *Recall* and F_1-*score* of your spam filtering program. **(3)**

Solution:

$$\begin{aligned}
Precision &= \frac{TP}{TP+FP} = \frac{10}{10+15} = 0.4 \\
Recall &= \frac{TP}{TP+FN} = \frac{10}{10+10} = 0.5 \\
F_1\text{-}score &= 2 \times \frac{Precision \times Recall}{Precision + Recall} = 2 \times \frac{0.4 \times 0.5}{0.4+0.5} \approx 0.22
\end{aligned}$$

Alternative Approach:

$$\begin{aligned}
Precision &= \frac{TP}{TP+FP} = \frac{65}{65+10} \approx 0.87 \\
Recall &= \frac{TP}{TP+FN} = \frac{65}{65+15} \approx 0.81 \\
F_1\text{-}score &= 2 \times \frac{Precision \times Recall}{Precision + Recall} \approx 2 \times \frac{0.87 \times 0.81}{0.87+0.81} \approx 0.84
\end{aligned}$$

Q5. [Unsupervised Learning: K-Means Clustering] **12 marks**

Assume the following dataset (consisting of (x,y)-coordinates of six points in a 2-dimensional plane) is given: $(0,0)$, $(0,1)$, $(1,0)$, $(1,3)$, $(2,2)$, $(2,3)$. You need to run the K-Means algorithm with $K=2$ and $K=3$ to cluster these data points. Assume that, Euclidean distance is used as the distance function to compute distances between centroids and objects (points) in the dataset. Answer the following.

(a) During the initialization, assume the centroids of the K clusters are greedily taken to be the first K points given in the dataset. Execute iteratively the K-Means clustering algorithm (till termination) and show the stepwise outputs in detail (for cluster sizess, $K=2$ and $K=3$). In particular, indicate the set of points that come under each cluster after every iteration and also compute their centroid to be used for the next iteration. Indicate when and how you decided to terminate/stop. You need to answer for $K=2$ and $K=3$ cases separately. **(4 × 2)**

Solution:

- For $K=2$ (two cluster formation):

Iterations	Output	Cluster-1	Cluster-2
0 (init.)	Elements	–	–
	Centroid	$(0,0)$	$(0,1)$
1 (cont.)	Elements	$(0,0)$, $(1,0)$	$(0,1)$, $(1,3)$, $(2,2)$, $(2,3)$
	Centroid	$(0.5,0)$	$(1.25,2.25)$
2 (cont.)	Elements	$(0,0)$, $(0,1)$, $(1,0)$	$(1,3)$, $(2,2)$, $(2,3)$
	Centroid	$(0.33,0.33)$	$(1.67,2.67)$
3 (stop)	Elements	$(0,0)$, $(0,1)$, $(1,0)$	$(1,3)$, $(2,2)$, $(2,3)$
	Centroid	$(0.33,0.33)$	$(1.67,2.67)$

- For $K=3$ (three cluster formation):

Iterations	Output	Cluster-1	Cluster-2	Cluster-3
0 (init.)	Elements	–	–	–
	Centroid	$(0,0)$	$(0,1)$	$(1,0)$
1 (cont.)	Elements	$(0,0)$	$(0,1)$, $(1,3)$, $(2,2)$, $(2,3)$	$(1,0)$
	Centroid	$(0,0)$	$(1.25,2.25)$	$(1,0)$
2 (cont.)	Elements	$(0,0)$, $(0,1)$	$(1,3)$, $(2,2)$, $(2,3)$	$(1,0)$
	Centroid	$(0,0.5)$	$(1.67,2.67)$	$(1,0)$
3 (stop)	Elements	$(0,0)$, $(0,1)$	$(1,3)$, $(2,2)$, $(2,3)$	$(1,0)$
	Centroid	$(0,0.5)$	$(1.67,2.67)$	$(1,0)$

- Decision to stop / terminate:
  In both cases, we stop when the elements inside the clusters remain unchanged across iterations.

(b) Upon termination, compute the average silhouette coefficient (SC) of the overall clustering only for the two cluster case (i.e. with your formed clusters for $K = 2$). **(4)**

Note: For your convenience, pairwise distances between points are given in the following table.

Pairwise Distance	**Data Points $\mathbf{P_i} = (\mathbf{x}, \mathbf{y})$**					
	$\mathbf{P_1} = (\mathbf{0},\mathbf{0})$	$\mathbf{P_2} = (\mathbf{0},\mathbf{1})$	$\mathbf{P_3} = (\mathbf{1},\mathbf{0})$	$\mathbf{P_4} = (\mathbf{1},\mathbf{3})$	$\mathbf{P_5} = (\mathbf{2},\mathbf{2})$	$\mathbf{P_6} = (\mathbf{2},\mathbf{3})$
$\mathbf{P_1} = (\mathbf{0},\mathbf{0})$	0.000					
$\mathbf{P_2} = (\mathbf{0},\mathbf{1})$	1.000	0.000				
$\mathbf{P_3} = (\mathbf{1},\mathbf{0})$	1.000	1.414	0.000			
$\mathbf{P_4} = (\mathbf{1},\mathbf{3})$	3.162	2.236	3.000	0.000		
$\mathbf{P_5} = (\mathbf{2},\mathbf{2})$	2.828	2.236	2.236	1.414	0.00	
$\mathbf{P_6} = (\mathbf{2},\mathbf{3})$	3.606	2.828	3.162	1.000	1.000	0.00

Solution:

The silhouette coefficient (SC) for each of the points are computed as:

$$P_1\ (0,0) \quad : \quad SC = 1 - \frac{a}{b} = 1 - \frac{\left(\frac{1+1}{2}\right)}{\left(\frac{3.162+2.828+3.606}{3}\right)} = 0.687$$

$$P_2\ (0,1) \quad : \quad SC = 1 - \frac{a}{b} = 1 - \frac{\left(\frac{1+1.414}{2}\right)}{\left(\frac{2.236+2.236+2.828}{3}\right)} = 0.504$$

$$P_3\ (1,0) \quad : \quad SC = 1 - \frac{a}{b} = 1 - \frac{\left(\frac{1+1.414}{2}\right)}{\left(\frac{3+2.236+3.162}{3}\right)} = 0.569$$

$$P_4\ (1,3) \quad : \quad SC = 1 - \frac{a}{b} = 1 - \frac{\left(\frac{1.414+1}{2}\right)}{\left(\frac{3.162+2.236+3}{3}\right)} = 0.569$$

$$P_5\ (2,2) \quad : \quad SC = 1 - \frac{a}{b} = 1 - \frac{\left(\frac{1.414+1}{2}\right)}{\left(\frac{2.828+2.236+2.236}{3}\right)} = 0.504$$

$$P_6\ (2,3) \quad : \quad SC = 1 - \frac{a}{b} = 1 - \frac{\left(\frac{1+1}{2}\right)}{\left(\frac{3.606+2.828+3.162}{3}\right)} = 0.687$$

$$\text{Cluster-1} \quad : \quad \Big(\{P_1, P_2, P_3\}\Big) \quad \text{Average-SC} = \frac{0.687 + 0.504 + 0.569}{3} = 0.587$$

$$\text{Cluster-2} \quad : \quad \Big(\{P_4, P_5, P_6\}\Big) \quad \text{Average-SC} = \frac{0.569 + 0.504 + 0.687}{3} = 0.587$$

$$\text{Overall} \quad : \quad \text{Average-SC} = \frac{0.587 + 0.587}{2} = 0.587$$

Q6. [Ensemble Learning: Boosting] **8 marks**

In this problem, we study how boosting algorithm performs on a very simple classification problem. We are given with *seven* training points (P_i) in a 2-dimensional plane ((x,y)-valued) positioned as $P_i = (i,i)$, for all $1 \leq i \leq 7$ ($i \in \mathbb{N}$), and their corresponding 2-class ($+/-$) labels are given as $+$, $+$, $+$, $-$, $-$, $-$, $+$, respectively. We shall use decision stumps as our weak learner / hypothesis. Decision stump classifier chooses a constant value c and classifies all points where $x \geq c$ as one class and other points where $x < c$ as the other class (or vice versa). Answer the following.

(a) What is the initial weight assigned to each data point? **(1)**

Solution:

Since $Weight(P_i)$ are equal for all P_i and $\sum_{i=1}^{7} Weight(P_i) = 1$,

therefore we get, $Weight(P_i) = \frac{1}{7}$, $\forall i \in [1,7]$.

(b) How many different decision stumps are possible for the data points given? **(2)**

Solution:

(7 separators / stumps) $\times$ (2 different class organizations for each)

= 14 different decision stumps are possible.

Alternative Approach:

(7 separators / stumps) $\times$ (2 different class organizations for each) $\times$ (2 dimensions)

= 28 different decision stumps are possible.

(though, this is not exactly correct as per the problem statement given, which mentions that the decision stumps are vertical lines having the form $x = c$ – however, full marks has been given for this too!)

(c) Let us chose one such decision stump as follows: $x \geq 3.5$ region is classified as '$-$' zone and $x < 3.5$ region is classified as '$+$' zone. What will be weights of all the data points after boosting is performed? Present your approach and calculations in details. **(4)**

Solution:

$$\varepsilon_t = \frac{1}{7} \qquad \text{and} \qquad \alpha_t = \frac{1}{2}\ln\left(\frac{1-\frac{1}{7}}{\frac{1}{7}}\right) = \frac{1}{2}\ln 6 = \ln(\sqrt{6})$$

Also, normalization factor, $Z = 2\sqrt{\varepsilon_t(1-\varepsilon_t)} = 2\sqrt{\frac{1}{7}\times\frac{6}{7}} = \frac{2\sqrt{6}}{7}$.

So, for correctly classified data-points (P_i where $i \in [1,6]$), the weight will decrease and become:

$$Weight(P_i) = \frac{\varepsilon_t . e^{-\alpha_t}}{Z} = \frac{1}{12}, \quad \forall i\ (1 \leq i \leq 6)$$

So, for wrongly classified data-point (P_7), the weight will increase and become:

$$Weight(P_7) = \frac{\varepsilon_t . e^{\alpha_t}}{Z} = \frac{1}{2}$$

(d) Can we perfectly classify all the training examples given in this problem by only applying boosting algorithm (AdaBoost)? Justify your claim. **(1)**

Solution:

No! The dataset given in this problem is *not* linearly separable.

Q7. [Principal Component Analysis] **6 marks**

You are provided with a dataset containing study-hours in a day and cgpa obtained for five students in $(hours, cgpa)$ format as: $(7,7)$, $(9,8)$, $(10,9)$, $(5,7)$ and $(4,4)$. You want to reduce this dataset into one dimension. Calculate the first principal component. Show your calculations in details. **(6)**

Solution:

The mean of the given data points is: $\left(\frac{7+9+10+5+4}{5} \frac{7+8+9+7+4}{5}\right) = (7,7)$.

The covariance matix can be constructed as:

$$\begin{aligned} CoVar(x,x) = Var(x) &= \frac{\left[(7-7)^2+(9-7)^2+(10-7)^2+(5-7)^2+(4-7)^2\right]}{5} = \frac{26}{5} \\ CoVar(x,y) = CoVar(y,x) &= \frac{\left[(7-7)\times(7-7)+(9-7)\times(8-7)+(10-7)\times(9-7)+(5-7)\times(7-7)+(4-7)\times(4-7)\right]}{5} = \frac{17}{5} \\ CoVar(y,y) = Var(y) &= \frac{\left[(7-7)^2+(8-7)^2+(9-7)^2+(7-7)^2+(4-7)^2\right]}{4} = \frac{14}{5} \end{aligned}$$

$$\therefore CoVar = \begin{bmatrix} CoVar(x,x) & CoVar(x,y) \\ CoVar(y,x) & CoVar(y,y) \end{bmatrix} = \begin{bmatrix} \frac{26}{5} & \frac{17}{5} \\ \frac{17}{5} & \frac{14}{5} \end{bmatrix}.$$

To compute eigenvalues, we make $\left| CoVar - \lambda I \right| = 0$, which gives:

$$\left(\frac{26}{5}-\lambda\right)\cdot\left(\frac{14}{5}-\lambda\right)-\frac{289}{25}=0 \implies \lambda^2-8\lambda+3=0 \implies \lambda = 4\pm\sqrt{13} = 7.61 \text{ or } 0.39$$

The corresponding eigenvectors with respect to the eigenvalues are computed as,

$$\begin{bmatrix} \frac{26}{5} & \frac{17}{5} \\ \frac{17}{5} & \frac{14}{5} \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = 4\pm\sqrt{13}. \begin{bmatrix} x \\ y \end{bmatrix} \implies \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{6\pm5\sqrt{13}}{17} \\ 1 \end{bmatrix} = \begin{bmatrix} 1.41 \\ 1 \end{bmatrix} \text{ or } \begin{bmatrix} -0.71 \\ 1 \end{bmatrix}$$

The principal component is the eigenvector corresponding to the highest eigenvalue $(4+\sqrt{13})$, which is calculated as: $\left[\frac{6+5\sqrt{13}}{17}, 1\right]^T = \left[1.41, 1\right]^T$.

Alternative Approach:

Since the mean of the given data points, $X = \begin{bmatrix} 7 & 7 \\ 9 & 8 \\ 10 & 9 \\ 5 & 7 \\ 4 & 4 \end{bmatrix}$ is $(7,7)$, we can center the given points with respect to mean as, $\hat{X} = \begin{bmatrix} 0 & 0 \\ 2 & 1 \\ 3 & 2 \\ -2 & 0 \\ -3 & -3 \end{bmatrix}$.

Now, $\hat{X}^T.\hat{X} = \begin{bmatrix} 26 & 17 \\ 17 & 14 \end{bmatrix}$.

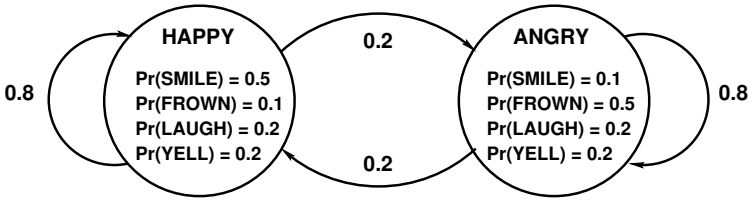
(Divide by 5 if you want the sample covariance matrix, but we do not care about the magnitude here.)

Its eigenvectors are $\left[\frac{6\pm5\sqrt{13}}{17}, 1\right]^T$ for eigenvalues $(20\pm5\sqrt{13})$. However, $\left[\frac{6+5\sqrt{13}}{17}, 1\right]^T$ eigenvector is chosen to be the principal component (as this corresponds to the highest eigenvalue).

Q8. [Hidden Markov Model] **8 marks**

Suppose, you live a very simple life and have only two emotional states, ANGRY and HAPPY. Some days you are ANGRY and some days you remain HAPPY. However, you hide your emotional state and others can only observe this from whether you SMILE, FROWN, LAUGH or YELL.

Suppose, you start on Day-1 at a HAPPY state and there is one transition per day. So, your emotional model (states and transitions with probabilities) and the probabilities of the observations at every state are given in the figure (right).



Let $Q(t)$ and $O(t)$ denote the state and observations at Day-t, respectively. Answer the following.

(a) Calculate: $Pr[Q(2) = HAPPY] = ?$ **(1)**

Solution:

If $Q(1) = HAPPY$ (i.e. you start from HAPPY state), then on Day-2, you can reach to HAPPY state by direct transition having 0.8 probability value.

Hence, $Pr[Q(2) = HAPPY] = 0.8$.

(b) Calculate: $Pr[O(2) = FROWN] = ?$ **(2)**

Solution:

$O(2) = FROWN$ can occur in Day-2 by either by staying in HAPPY state or by transiting to ANGRY state.

Hence, $Pr[O(2) = FROWN] = 0.8 \times 0.1 + 0.2 \times 0.5 = 0.18$.

(c) Calculate: $Pr[Q(2) = HAPPY \mid O(2) = FROWN] = ?$ **(2)**

Solution:

$$\begin{aligned} & Pr[Q(2) = HAPPY \mid O(2) = FROWN] \\ = & \frac{Pr[O(2) = FROWN \mid Q(2) = HAPPY] \cdot Pr[Q(2) = HAPPY]}{Pr[O(2) = FROWN]} \\ = & \frac{0.1 \times 0.8}{0.18} \approx 0.444 \end{aligned}$$

(d) Calculate: $Pr[O(100) = YELL] = ?$ **(2)**

Solution:

$$\begin{aligned} & Pr[O(100) = YELL] \\ = \; & Pr[O(100) = YELL \mid Q(100) = HAPPY] \cdot Pr[Q(100) = HAPPY] \\ & + Pr[O(100) = YELL \mid Q(100) = ANGRY] \times Pr[Q(100) = ANGRY] \\ = \; & 0.2 \times Pr[Q(100) = HAPPY] + 0.2 \times Pr[Q(100) = ANGRY] \\ = \; & 0.2 \times \big(Pr[Q(100) = HAPPY] + Pr[Q(100) = ANGRY]\big) \quad = \quad 0.2 \times 1 \quad = \quad 0.2 \end{aligned}$$

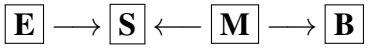
(e) Assume $O(1) = FROWN, O(2) = FROWN, O(3) = FROWN, O(4) = FROWN, O(5) = FROWN$, what is the most likely sequence of states? Just write the state sequence (no need to elaborate). **(1)**

Solution:

HAPPY, ANGRY, ANGRY, ANGRY, ANGRY

Q9. [Bayesian Networks] **8 marks**

A smell of Sulphur (S) can be caused either by rotten Eggs (E) or as a sign of the doom brought by the Mayan Apocalypse (M). The Mayan Apocalypse also causes the oceans to Boil (B). The Bayesian network and corresponding conditional probability tables for this situation are shown on right. Calculate the following.



$Pr(E)$	$=$	0.4	$Pr(S \mid E, M)$	$=$	1.0
$Pr(M)$	$=$	0.1	$Pr(S \mid E, \neg M)$	$=$	0.8
$Pr(B \mid M)$	$=$	1.0	$Pr(S \mid \neg E, M)$	$=$	0.3
$Pr(B \mid \neg M)$	$=$	0.1	$Pr(S \mid \neg E, \neg M)$	$=$	0.1

(a) What is the joint probability when none of these four events (E, S, M or B) occur? **(1.5)**

Solution:

By expanding the joint according to the chain rule of conditional probability:

$$\begin{aligned} Pr(\neg E, \neg S, \neg M, \neg B) \quad &= \quad Pr(\neg E) \cdot Pr(\neg M) \cdot Pr(\neg S \mid \neg E, \neg M) \cdot Pr(\neg B \mid \neg M) \\ &= \quad (1 - 0.4) \times (1 - 0.1) \times (1 - 0.1) \times (1 - 0.1) \quad = \quad 0.4374 \end{aligned}$$

(d) What is the probability that rotten eggs are present, given that Mayan Apocalypse is occurring? **(1.5)**

Solution:

Here, we have $E \perp\!\!\!\perp M$ (E is independent of M), which can be inferred from the Bayes' net.

$$Pr(E \mid M) \quad = \quad Pr(E) \quad = \quad 0.4$$

(c) What is the probability that the oceans boil? **(1.5)**

Solution:

By marginalizing out E according to the law of total probability:

$$\begin{aligned} Pr(B) &= Pr(B \mid M) \cdot Pr(M) + Pr(B \mid \neg M) \cdot Pr(\neg M) \\ &= 1.0 \times 0.1 + 0.1 \times (1 - 0.1) \quad = \quad 0.19 \end{aligned}$$

(d) What is the probability that Mayan Apocalypse is occurring, given that the oceans are boiling? **(1.5)**

Solution:

By the definition of conditional probability:

$$Pr(M \mid B) = \frac{Pr(B \mid M) \cdot Pr(M)}{Pr(B)} = \frac{1.0 \times 0.1}{0.19} \approx 0.5263$$

(e) What is the probability that Mayan Apocalypse is occurring, given that there is a smell of sulphur, the oceans are boiling, and there are rotten eggs? **(2)**

Solution:

By Bayes' probability rule:

$$\begin{aligned} Pr(M \mid S,B,E) &= \frac{Pr(M,S,B,E)}{Pr(S,B,E)} = \frac{Pr(M,S,B,E)}{Pr(M,S,B,E) + Pr(\neg M,S,B,E)} \\ &= \frac{Pr(E) \cdot Pr(M) \cdot Pr(S \mid E,M) \cdot Pr(B \mid M)}{Pr(E) \cdot Pr(M) \cdot Pr(S \mid E,M) \cdot Pr(B \mid M) + Pr(E) \cdot Pr(\neg M) \cdot Pr(S \mid E,\neg M) \cdot Pr(B \mid \neg M)} \\ &= \frac{0.4 \times 0.1 \times 1.0 \times 1.0}{0.4 \times 0.1 \times 1.0 \times 1.0 + 0.4 \times 0.9 \times 0.8 \times 0.1} \approx 0.5814 \end{aligned}$$

Q10. [Expectation-Maximization (EM) Algorithm] **10 marks**

Consider again the Bayesian Network shown in **Q9** (re-drawn on the right). You must train this network from partly observed data, using EM algorithm and given $K = 6$ training examples shown on the right where only one example contains an unobserved value (marked with ?), namely, s_4. You will be asked to simulate a few steps of EM algorithm by hand. Notation-wise, e_k, s_k, m_k, and b_k indicate the values of E, S, M and B, respectively, as seen in the k^{th} example (row). For example, $e_1 = 1$, $s_1 = 0$, $m_1 = 1$, and $b_1 = 1$.

K	E	S	M	B
$k = 1$	1	0	1	1
$k = 2$	0	1	1	1
$k = 3$	1	1	1	1
$k = 4$	1	?	0	0
$k = 5$	0	0	0	1
$k = 6$	1	1	0	1

Answer the following.

(a) Given that *all variables are Boolean*, how many basic parameters need to be estimated for the given Bayes Network? Write down the list of these parameters.

For example, one parameter will be $\theta(s \mid 10)$, which stands for $Pr(S \mid E, \neg M)$. **(1)**

Solution:

We need to estimate 8 parameters, which are given as follows:

$$\begin{aligned} \theta(e) &= Pr(E) & \theta(s \mid 11) &= Pr(S \mid E, M) \\ \theta(m) &= Pr(M) & \theta(s \mid 10) &= Pr(S \mid E, \neg M) \\ \theta(b \mid 1) &= Pr(B \mid M) & \theta(s \mid 01) &= Pr(S \mid \neg E, M) \\ \theta(b \mid 0) &= Pr(B \mid \neg M) & \theta(s \mid 00) &= Pr(S \mid \neg E, \neg M) \end{aligned}$$

(b) Simulate the first E-step of the EM algorithm. Before we start, we initialize all the basic parameters (which you identified in Part (a)) as 0.6, and then proceed to execute the E-step. What expectation values will get calculated in this E-step? Calculate: $\mathbb{E}(s_4 = 1 \mid e_4, m_4, b_4; \theta)$.

Note that, only one example ($k = 4$) contains an unobserved variable ($s_4 = ?$), but $e_4 = 1$ and $m_4 = b_4 = 0$. **(2.5)**

Solution:

$$\begin{aligned} & \mathbb{E}(s_4 = 1 \mid e_4, m_4, b_4; \theta) \\ = & \frac{Pr(s_4 = 1, e_4, m_4, b_4 \mid \theta)}{Pr(s_4 = 1, e_4, m_4, b_4 \mid \theta) + Pr(s_4 = 0, e_4, m_4, b_4 \mid \theta)} \\ = & \frac{\theta(s_4 = 1 \mid e_4, m_4) \cdot \theta(b_4 \mid m_4) \cdot \theta(e_4) \cdot \theta(m_4)}{\theta(s_4 = 1 \mid e_4, m_4) \cdot \theta(b_4 \mid m_4) \cdot \theta(e_4) \cdot \theta(m_4) + \theta(s_4 = 0 \mid e_4, m_4) \cdot \theta(b_4 \mid m_4) \cdot \theta(e_4) \cdot \theta(m_4)} \\ = & \frac{0.6 \times 0.4 \times 0.6 \times 0.4}{1 \times 0.4 \times 0.6 \times 0.4} = 0.6 \end{aligned}$$

(c) Next, simulate the first M-step of the EM algorithm. What will be the estimated values of all the model parameters (which you identified in Part (a)) that we obtain in this M-step? **(4)**

Solution:

8 parameters will get the updated values as follows:

$$\theta(e) = Pr(E) = \frac{\#\{E=1\}}{\#K} = \frac{4}{6} \approx 0.67$$

$$\theta(m) = Pr(M) = \frac{\#\{M=1\}}{\#K} = \frac{3}{6} = 0.5$$

$$\theta(b \mid 1) = Pr(B \mid M) = \frac{\#\{B=1, M=1\}}{\#\{M=1\}} = \frac{3}{3} = 1.0$$

$$\theta(b \mid 0) = Pr(B \mid \neg M) = \frac{\#\{B=1, M=0\}}{\#\{M=0\}} = \frac{2}{3} \approx 0.67$$

$$\theta(s \mid 11) = Pr(S \mid E, M) = \frac{\#\{E=1, M=1\} \cdot \mathbb{E}[S=1]}{\#\{E=1, M=1\}} = \frac{1 \times 1.0 + 1 \times 0.0}{2} = 0.5$$

$$\theta(s \mid 10) = Pr(S \mid E, \neg M) = \frac{\#\{E=1, M=0\} \cdot \mathbb{E}[S=1]}{\#\{E=1, M=0\}} = \frac{1 \times 0.6 + 1 \times 1.0}{2} = 0.8$$

$$\theta(s \mid 01) = Pr(S \mid \neg E, M) = \frac{\#\{E=0, M=1\} \cdot \mathbb{E}[S=1]}{\#\{E=0, M=1\}} = \frac{1 \times 1.0}{1} = 1.0$$

$$\theta(s \mid 00) = Pr(S \mid \neg E, \neg M) = \frac{\#\{E=0, M=0\} \cdot \mathbb{E}[S=1]}{\#\{E=0, M=0\}} = \frac{1 \times 0.0}{1} = 0.0$$

(d) Lastly, simulate (again) the second E-step of the EM algorithm. What expectation values will get calculated in this E-step? Calculate: $\mathbb{E}(s_4 = 1 \mid e_4, m_4, b_4; \theta)$. **(2.5)**

Solution:

$$\begin{aligned}
& \mathbb{E}(s_4 = 1 \mid e_4, m_4, b_4; \theta) \\
= & \frac{Pr(s_4 = 1, e_4, m_4, b_4 \mid \theta)}{Pr(s_4 = 1, e_4, m_4, b_4 \mid \theta) + Pr(s_4 = 0, e_4, m_4, b_4 \mid \theta)} \\
= & \frac{\theta(s_4 = 1 \mid e_4, m_4) \cdot \theta(b_4 \mid m_4) \cdot \theta(e_4) \cdot \theta(m_4)}{\theta(s_4 = 1 \mid e_4, m_4) \cdot \theta(b_4 \mid m_4) \cdot \theta(e_4) \cdot \theta(m_4) + \theta(s_4 = 0 \mid e_4, m_4) \cdot \theta(b_4 \mid m_4) \cdot \theta(e_4) \cdot \theta(m_4)} \\
= & \frac{0.8 \times 0.33 \times 0.67 \times 0.5}{1 \times 0.33 \times 0.67 \times 0.5} = 0.8
\end{aligned}$$

Q11. [Reinforcement Learning] **8 marks**

Consider a simple MDP with 3 states s_1, s_2, s_3 and 2 actions a_1, a_2. The transition probabilities and expected rewards are given in the following table. Assume discount factor $\gamma = 1$.

(from) State	Action	Reward	Transition Probability (to State)		
			s_1	s_2	s_3
s_1	a_1	8	0.2	0.6	0.2
	a_2	10	0.1	0.2	0.7
s_2	a_1	1	0.3	0.3	0.4
	a_2	-1	0.5	0.3	0.2
s_3	a_1	0	0	0	1.0
	a_2	0	0	0	1.0

Your task is to determine an optimal deterministic policy by manually working out simply the first two iterations of *value iteration* algorithm. Initialize the value function for each state to be it's max (over actions) reward, i.e., we initialize the value function to be $v_0(s_1) = 10,\ v_0(s_2) = 1,\ v_0(s_3) = 0$. Considering all states and actions, calculate $q_k(\cdot,\cdot)$ and $v_k(\cdot)$ from $v_{k-1}(\cdot)$ using the value iteration update, and then calculate the greedy policy $\pi_k(\cdot)$ from $q_k(\cdot,\cdot)$ for two iterations (i.e. for $k = 1, 2$). **(8)**

Solution:

Value iteration algorithm follows Bellman's optimality equation for iterative updates:

$$q_k(s,a) = R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a v_{k-1}(s')$$

$$v_k(s) = \max_{a \in A} q_k(s,a) = \max_{a \in A} \left[R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a v_{k-1}(s') \right]$$

Following this, we present the updates for two value iterations as follows:

For $k = 1$,

$$\begin{aligned} q_1(s_1,a_1) &= 8 + 0.2 \times 10.0 + 0.6 \times 1.0 + 0.2 \times 0 = 10.6 \\ q_1(s_1,a_2) &= 10 + 0.1 \times 10.0 + 0.2 \times 1.0 + 0.7 \times 0 = 11.2 \\ \therefore\ v_1(s_1) &= \max\left[10.6, 11.2\right] = 11.2 \quad \text{and} \quad \pi_1(s_1) = a_2 \end{aligned}$$

$$\begin{aligned} q_1(s_2,a_1) &= 1 + 0.3 \times 10.0 + 0.3 \times 1.0 + 0.4 \times 0 = 4.3 \\ q_1(s_2,a_2) &= -1 + 0.5 \times 10.0 + 0.3 \times 1.0 + 0.2 \times 0 = 4.3 \\ \therefore\ v_1(s_2) &= \max\left[4.3, 4.3\right] = 4.3 \quad \text{and} \quad \pi_1(s_2) = a_1 \text{ or } a_2 \end{aligned}$$

For $k = 2$,

$$\begin{aligned} q_2(s_1,a_1) &= 8 + 0.2 \times 11.2 + 0.6 \times 4.3 + 0.2 \times 0 = 12.82 \\ q_2(s_1,a_2) &= 10 + 0.1 \times 11.2 + 0.2 \times 4.3 + 0.7 \times 0 = 11.98 \\ \therefore\ v_2(s_1) &= \max\left[12.82, 11.98\right] = 12.82 \quad \text{and} \quad \pi_2(s_1) = a_1 \end{aligned}$$

$$\begin{aligned} q_2(s_2,a_1) &= 1 + 0.3 \times 11.2 + 0.3 \times 4.3 + 0.4 \times 0 = 5.65 \\ q_2(s_2,a_2) &= -1 + 0.5 \times 11.2 + 0.3 \times 4.3 + 0.2 \times 0 = 5.89 \\ \therefore\ v_2(s_2) &= \max\left[5.65, 5.89\right] = 5.89 \quad \text{and} \quad \pi_2(s_2) = a_2 \end{aligned}$$

— The question paper ends here. —