INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

Stamp / Signature of the Invigilator

EXAMINATION (End Semester)	**SEMESTER (Autumn 2024-2025)**

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

Subject Number	C	S	6	0	0	7	7	**Subject Name**	*REINFORCEMENT 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**	**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

Reinforcement Learning (CS60077) **Autumn Semester 2024-2025**

September 2024 **Mid-Semester Examination** Maximum Marks: 60

Instructions:

- Write your answers in the question paper itself. Be brief and precise. Answer *all* questions.
- Write the answers only in the respective spaces provided. The last two blank pages may be used for rough work or leftover answers.
- In case you may need more space/pages, please ask for additional sheets in the exam hall and attach the same with this booklet while submitting.
- If you use any theorem / result / formula covered in the class, just mention it, do not elaborate or prove unless you are explicitly asked to do so.
- Write all the proofs / derivations in mathematically / logically precise language. Unclear and/or dubious statements would be severely penalized.

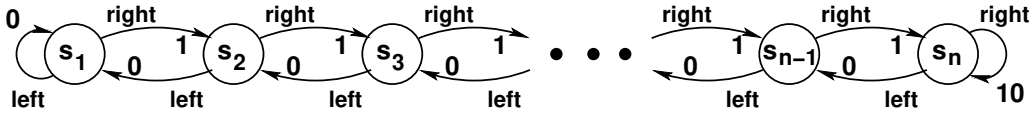
— Question Paper Starts from Next Page —

Q1. [Planning using Dynamic Programming] **16 marks**

Consider an MDP, $M = \langle S, A, P, R, \gamma \rangle$ with n states, that is, $S = \{s_1, s_2, \ldots, s_n\}$, and 2 actions, that is, $A = \{\texttt{left}, \texttt{right}\}$. Assume the discount factor as, $\gamma = 0.5$. The transition probabilities are given as,

$\mathbb{P}\big[S_{t+1} = s_1 \mid S_t = s_1, A_t = \texttt{left}\big] = 1$ and $\mathbb{P}\big[S_{t+1} = s_{k-1} \mid S_t = s_k, A_t = \texttt{left}\big] = 1 \;\; \big(2 \le k \le n\big)$

$\mathbb{P}\big[S_{t+1} = s_n \mid S_t = s_n, A_t = \texttt{right}\big] = 1$ and $\mathbb{P}\big[S_{t+1} = s_{k+1} \mid S_t = s_k, A_t = \texttt{right}\big] = 1 \;\; \big(1 \le k \le n-1\big)$.



The rewards are distributed as, $R(s_k, \texttt{left}) = 0 \;\big(1 \le k \le n\big)$, $R(s_k, \texttt{right}) = +1 \;\big(1 \le k \le n-1\big)$, and $R(s_n, \texttt{right}) = +10$. Answer the following questions.

(a) Figure out the optimal policy, that is, $\pi^*(s_k)$, for every k $(1 \le k \le n)$ for the MDP M. **(1)**

Solution:

$\pi^*(s_k) = \texttt{right}$, for every k $(1 \le k \le n)$.

(b) Compute the optimal value of state, s_n, that is, $V_*(s_n)$. Show your calculations. **(2)**

Solution:

Since $\pi^*(s_n) = \texttt{right}$, that is, $\mathbb{P}\left[A_t = \texttt{right} \mid S_t = s_n\right] = 1$, we have

$$\begin{aligned} V_*(s_n) &= \mathbb{E}_{\pi^*}\left[R(S_t, \texttt{right}) + \gamma V_{\pi^*}(S_{t+1}) \mid S_t = s_n\right] \\ &= R(S_n, \texttt{right}) + \gamma \cdot R(S_n, \texttt{right}) + \gamma^2 \cdot R(S_n, \texttt{right}) + \cdots \\ &= 10 + 0.5 \cdot 10 + 0.5^2 \cdot 10 + \cdots \;=\; 10 \times \frac{1}{1-0.5} \;=\; 20. \end{aligned}$$

(c) Compute the optimal value function, $V_*(s_k)$ as a function of n and k, for all k $(1 \le k \le n-1)$. Show your calculations. **(3)**

Solution:

Since $\pi^*(s_k) = \texttt{right}$, that is, $\mathbb{P}\left[A_t = \texttt{right} \mid S_t = s_k\right] = 1$ for every k $(1 \le k \le n)$, we have

$$\begin{aligned} V_*(s_{n-1}) &= R(s_{n-1}, \texttt{right}) + \gamma \cdot V_*(s_n) \;=\; 1 + 0.5 \cdot V_*(s_n) \\ V_*(s_{n-2}) &= R(s_{n-2}, \texttt{right}) + \gamma \cdot V_*(s_{n-1}) \;=\; 1 + 0.5 + 0.5^2 \cdot V_*(s_n) \\ V_*(s_{n-3}) &= R(s_{n-3}, \texttt{right}) + \gamma \cdot V_*(s_{n-2}) \;=\; 1 + 0.5 + 0.5^2 + 0.5^3 \cdot V_*(s_n) \\ \cdots \quad & \quad \cdots \quad \cdots \\ V_*(s_{n-k}) &= R(s_{n-k}, \texttt{right}) + \gamma \cdot V_*(s_{n-k+1}), \quad \text{for every } k\ (1 \le k \le n-1) \\ &= 1 + 0.5 + 0.5^2 + \cdots + 0.5^{k-1} + 0.5^k \cdot V_*(s_n) \\ &= \frac{1 - 0.5^k}{1 - 0.5} + 0.5^k \cdot 20 \;=\; 2 + 0.5^k \cdot 18 \end{aligned}$$

$$\therefore\ V_*(s_k) \;=\; 2 + 0.5^{(n-k)} \cdot 18, \quad \text{for all } k\ (1 \le k \le n-1)$$

(d) Suppose you wish to perform *value iteration* over the MDP M to figure out the value estimates of each state. You plan to start with a value estimate equal to 0 for every state. Calculate the value estimates of all states after the *first* and *second* iterations, respectively; that is, determine $V^1(s_k)$ and $V^2(s_k)$, for all k $(1 \leq k \leq n)$. **(2 + 3)**

Solution:

For $(j+1)$-th iteration, the update made (as per Bellman Optimality Equations) will be:

$$\begin{aligned}
V^{j+1}(s_n) &= \max\Big[\Big(R(s_n,\texttt{left})+\gamma\cdot V^j(s_{n-1})\Big),\ \Big(R(s_n,\texttt{right})+\gamma\cdot V^j(s_n)\Big)\Big] \\
&= \max\Big[\Big(0+0.5\cdot V^j(s_{n-1})\Big),\ \Big(10+0.5\cdot V^j(s_n)\Big)\Big] \\
V^{j+1}(s_1) &= \max\Big[\Big(R(s_1,\texttt{left})+\gamma\cdot V^j(s_1)\Big),\ \Big(R(s_1,\texttt{right})+\gamma\cdot V^j(s_2)\Big)\Big] \\
&= \max\Big[\Big(0+0.5\cdot V^j(s_1)\Big),\ \Big(1+0.5\cdot V^j(s_2)\Big)\Big] \\
V^{j+1}(s_{n-k}) &= \max\Big[\Big(R(s_{n-k},\texttt{left})+\gamma\cdot V^j(s_{n-k-1})\Big),\ \Big(R(s_{n-k},\texttt{right})+\gamma\cdot V^j(s_{n-k+1})\Big)\Big] \\
&= \max\Big[\Big(0+0.5\cdot V^j(s_{n-k-1})\Big),\ \Big(1+0.5\cdot V^j(s_{n-k+1})\Big)\Big], \quad \text{for all } k\ (1 \leq k \leq n-2)
\end{aligned}$$

Given that, $V^0(s_k) = 0$, for all k $(1 \leq k \leq n)$.

So, iteration-wise we get,

After First Iteration:

$$\begin{aligned}
V^1(s_n) &= \max[0,10] = 10, \\
V^1(s_{n-k}) &= \max[0,1] = 1, \quad \text{for all } k\ (1 \leq k \leq n-1)
\end{aligned}$$

After Second Iteration:

$$\begin{aligned}
V^2(s_n) &= \max[(0+0.5\cdot 1),(10+0.5\cdot 10)] = 15, \\
V^2(s_{n-1}) &= \max[(0+0.5\cdot 1),(1+0.5\cdot 10)] = 6, \\
V^2(s_{n-k}) &= \max[(0+0.5\cdot 1),(1+0.5\cdot 1)] = 1.5, \quad \text{for all } k\ (2 \leq k \leq n-1)
\end{aligned}$$

(e) Suppose you wish to perform *policy evaluation* over the MDP M to figure out the value estimates of each state for a given policy π, where $\pi(s_k, \texttt{left}) = \frac{1}{4}$ and $\pi(s_k, \texttt{right}) = \frac{3}{4}$ for every state s_k $(1 \leq k \leq n)$. You plan to start with a value estimate equal to 0 for every state. Calculate the value estimates of all states after the *first* and *second* iterations, respectively; that is, determine $V_\pi^1(s_k)$ and $V_\pi^2(s_k)$, for all k $(1 \leq k \leq n)$. **(2 + 3)**

Solution:

For $(j+1)$-th iteration, the update made (as per Bellman Expectation Equations) will be:

$$\begin{aligned}
V_\pi^{j+1}(s_n) &= \left[\pi(s_n, \texttt{left}) \cdot \left(R(s_n, \texttt{left}) + \gamma \cdot V_\pi^j(s_{n-1})\right) + \pi(s_n, \texttt{right}) \cdot \left(R(s_n, \texttt{right}) + \gamma \cdot V_\pi^j(s_n)\right)\right] \\
&= \left[\frac{1}{4} \cdot \left(0 + 0.5 \cdot V_\pi^j(s_{n-1})\right) + \frac{3}{4} \cdot \left(10 + 0.5 \cdot V_\pi^j(s_n)\right)\right] \\
V_\pi^{j+1}(s_1) &= \left[\pi(s_1, \texttt{left}) \cdot \left(R(s_1, \texttt{left}) + \gamma \cdot V_\pi^j(s_1)\right) + \pi(s_1, \texttt{right}) \cdot \left(R(s_1, \texttt{right}) + \gamma \cdot V_\pi^j(s_2)\right)\right] \\
&= \left[\frac{1}{4} \cdot \left(0 + 0.5 \cdot V_\pi^j(s_1)\right) + \frac{3}{4} \cdot \left(1 + 0.5 \cdot V_\pi^j(s_2)\right)\right] \\
V_\pi^{j+1}(s_{n-k}) &= \left[\pi(s_{n-k}, \texttt{left}) \cdot \left(R(s_{n-k}, \texttt{left}) + \gamma \cdot V_\pi^j(s_{n-k-1})\right)\right. \\
&\quad \left. + \pi(s_{n-k}, \texttt{right}) \cdot \left(R(s_{n-k}, \texttt{right}) + \gamma \cdot V_\pi^j(s_{n-k+1})\right)\right] \\
&= \left[\frac{1}{4} \cdot \left(0 + 0.5 \cdot V_\pi^j(s_{n-k-1})\right) + \frac{3}{4} \cdot \left(1 + 0.5 \cdot V_\pi^j(s_{n-k+1})\right)\right], \quad \text{for all } k \ (1 \leq k \leq n-2)
\end{aligned}$$

Given that, $V_\pi^0(s_k) = 0$, for all k $(1 \leq k \leq n)$.

So, iteration-wise we get,

After First Iteration:

$$\begin{aligned}
V^1(s_n) &= \left[\frac{1}{4} \cdot 0 + \frac{3}{4} \cdot 10\right] = 7.5, \\
V^1(s_{n-k}) &= \left[\frac{1}{4} \cdot 0 + \frac{3}{4} \cdot 1\right] = 0.75, \quad \text{for all } k \ (1 \leq k \leq n-1)
\end{aligned}$$

After Second Iteration:

$$\begin{aligned}
V^2(s_n) &= \left[\frac{1}{4} \cdot \left(0 + 0.5 \cdot 0.75\right) + \frac{3}{4} \cdot \left(10 + 0.5 \cdot 7.5\right)\right] = 10.40625, \\
V^2(s_{n-1}) &= \left[\frac{1}{4} \cdot \left(0 + 0.5 \cdot 0.75\right) + \frac{3}{4} \cdot \left(1 + 0.5 \cdot 7.5\right)\right] = 3.65625, \\
V^2(s_{n-k}) &= \left[\frac{1}{4} \cdot \left(0 + 0.5 \cdot 0.75\right) + \frac{3}{4} \cdot \left(1 + 0.5 \cdot 0.75\right)\right] = 1.125, \quad \text{for all } k \ (2 \leq k \leq n-1)
\end{aligned}$$

Q2. [Model-Free Prediction and Control] **20 marks**

Suppose that, instead of knowing the model of the MDP M as explicitly presented in the question **Q1**, you only know the states, $S = \{s_1, s_2, \ldots, s_n\}$, and the actions, $A = \{\texttt{left}, \texttt{right}\}$ (both actions are applicable from all states). You have also observed the following trajectory of just one episode:

$$
\begin{array}{ccccccccccccccccc}
s_1 & \xrightarrow[0]{\texttt{left}} & s_1 & \xrightarrow[+1]{\texttt{right}} & s_2 & \xrightarrow[0]{\texttt{left}} & s_1 & \xrightarrow[+1]{\texttt{right}} & s_2 & \xrightarrow[+1]{\texttt{right}} & s_3 & \xrightarrow[0]{\texttt{left}} & s_2 & \xrightarrow[+1]{\texttt{right}} & s_3 & \xrightarrow[+1]{\texttt{right}} & s_4 & \cdots \\
 & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & & \\
\cdots & & s_{k+1} & \xrightarrow[0]{\texttt{left}} & s_k & \xrightarrow[+1]{\texttt{right}} & s_{k+1} & \xrightarrow[+1]{\texttt{right}} & s_{k+2} & \xrightarrow[0]{\texttt{left}} & s_{k+1} & \xrightarrow[+1]{\texttt{right}} & s_{k+2} & \xrightarrow[+1]{\texttt{right}} & s_{k+3} & & \cdots \\
 & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & & \\
 & \cdots & & s_{n-1} & \xrightarrow[0]{\texttt{left}} & s_{n-2} & \xrightarrow[+1]{\texttt{right}} & s_{n-1} & \xrightarrow[+1]{\texttt{right}} & s_n & \xrightarrow[0]{\texttt{left}} & s_{n-1} & \xrightarrow[+1]{\texttt{right}} & s_n & \xrightarrow[+10]{\texttt{right}} & s_n & &
\end{array}
$$

As shown above, the episode starts from state s_1 with a `left` action fetching a reward of 0 and staying in state s_1, followed by a `right` action fetching a reward of $+1$ and moving to state s_2 from s_1. Then, it continues with a repeated sequence of three actions, `left`, `right`, `right`, one after another for the next $(n-1)$ times, and finally terminates the episode at state s_n. Answer the following questions.

(a) Given this only episode, calculate the value function estimates, $V(s_k)$, as a function of n and k for every state $s_k \in S$ $(1 \le k \le n)$ obtained using: **(i)** Every-Visit Monte Carlo and **(ii)** First-Visit Monte Carlo methods. **(4 + 3)**

Solution: (taking $\gamma = 1$, however γ may be taken as any value or simply be kept symbolically)

(i) Every-Visit Monte-Carlo:

Every-Visit Monte Carlo averages the returns starting from each occurrence of each of the states across the episode. Therefore, Every-Visit Monte Carlo value function estimate would converge to:

$$
\begin{aligned}
V(s_1) &= \Big[\Big((0+1)+(0+1+1)\cdot(n-2)+(0+1+10)\Big) \\
&\quad +\Big((1)+(0+1+1)\cdot(n-2)+(0+1+10)\Big) \\
&\quad +\Big((1+1)+(0+1+1)\cdot(n-3)+(0+1+10)\Big)\Big]/3 \;=\; \frac{6n+23}{3} \\
V(s_k) &= \Big[\Big((0+1+1)\cdot(n-k)+(0+1+10)\Big) \\
&\quad +\Big((1)+(0+1+1)\cdot(n-k-1)+(0+1+10)\Big) \\
&\quad +\Big((1+1)+(0+1+1)\cdot(n-k-2)+(0+1+10)\Big)\Big]/3 \\
&= \frac{6n-6k+30}{3} \;=\; 2n-2k+10, \quad \text{for all } k\ (2 \le k \le n-1) \\
V(s_n) &= \Big[\Big(0+1+10\Big)+\Big(10\Big)\Big]/2 \;=\; 10.5
\end{aligned}
$$

(ii) First-Visit Monte Carlo:

First-Visit Monte Carlo averages the returns starting from the first occurrence of each of the states across the episode. Therefore, First-Visit Monte Carlo value function estimate would converge to:

$$V(s_1) = \frac{(0+1)+(0+1+1)\cdot(n-2)+(0+1+10)}{1} = 2n+8$$

$$V(s_k) = \frac{(0+1+1)\cdot(n-k)+(0+1+10)}{1} = 2n-2k+11, \quad \text{for all } k\ (2 \le k \le n)$$

(b) From the one-step state transitions and sample rewards seen in the given episodic data, construct an MRP (*ignoring the actions* mentioned during state transitions) that TD(0) (that is, one-step temporal difference learning) essentially builds with the estimated transition probabilities, $\mathbb{P}\big[S_{t+1} = s_j \mid S_t = s_i\big]$, and the reward function, $R(s_k)$, for all i, j, k $(1 \le i, j, k \le n)$. **(3 + 2)**

Solution:

TD(0) essentially constructs an MRP, $M' = \langle S, P, R, \gamma \rangle$, with the transition probabilities, $P : S \times S \to \mathbb{R}^{[0,1]}$, and the reward function, $R : S \to \mathbb{R}$, following certainty equivalence estimate from the one-step transitions and sample rewards seen in the data.

The transition probabilities (P) would be estimated as:

$$\mathbb{P}\big[S_{t+1} = s_1 \mid S_t = s_1\big] = \frac{1}{3}, \quad \mathbb{P}\big[S_{t+1} = s_2 \mid S_t = s_1\big] = \tfrac{2}{3}, \quad \mathbb{P}\big[S_{t+1} = s_j \mid S_t = s_1\big] = 0, \ \big(2 \le j \le n\big)$$

$$\mathbb{P}\big[S_{t+1} = s_{k-1} \mid S_t = s_k\big] = \frac{1}{3}, \quad \mathbb{P}\big[S_{t+1} = s_{k+1} \mid S_t = s_k\big] = \tfrac{2}{3}, \quad \mathbb{P}\big[S_{t+1} = s_j \mid S_t = s_k\big] = 0, \ \begin{pmatrix} 2 \le k \le n-1, \\ 1 \le j \le n,\ j \ne k \pm 1 \end{pmatrix}$$

$$\mathbb{P}\big[S_{t+1} = s_{n-1} \mid S_t = s_n\big] = \frac{1}{2}, \quad \mathbb{P}\big[S_{t+1} = s_n \mid S_t = s_n\big] = \tfrac{1}{2}, \quad \mathbb{P}\big[S_{t+1} = s_j \mid S_t = s_n\big] = 0, \ \big(1 \le j \le n-2\big)$$

The reward function (R) would be estimated as:

$$R(s_k) = \frac{(0+1+1)}{3} = \frac{2}{3}, \text{ for all } k\ (1 \le k \le n-1) \quad \text{and} \quad R(s_n) = \frac{(0+10)}{2} = 5$$

(c) From the given episodic data spanning across multiple timesteps, calculate the estimated update values after first four timesteps, that is, determine $Q_1(s_1, \texttt{left})$, $Q_2(s_1, \texttt{right})$, $Q_3(s_2, \texttt{left})$ and $Q_4(s_1, \texttt{right})$, using *one-step SARSA* approach. Assume all the *initial* Q-value estimates of the states (prior to any update) as 1; the discount factor as $\gamma = 0.5$; and the step size as $\alpha = 0.1$. Here, $Q_t(s_k, \cdot)$ denotes the estimated Q-value of state $s_k \in S$ after t-th timestep. **(2 × 4)**

Solution:

Given transitions of the form, $s_i \xrightarrow[r_i]{a_i} s_j \xrightarrow[r_j]{a_j} s_k$, the SARSA update equation is:

$$Q(s_i, a_i) \leftarrow Q(s_i, a_i) + \alpha \Big[r_i + \gamma \cdot Q(s_j, a_j) - Q(s_i, a_i) \Big]$$

The updated Q-values following SARSA approach will be:

Update after First Timestep:

$$\begin{aligned} Q_1(s_1, \texttt{left}) &= Q_0(s_1, \texttt{left}) + \alpha \cdot \Big[R(s_1, \texttt{left}) + \gamma \cdot Q_0(s_1, \texttt{right}) - Q_0(s_1, \texttt{left}) \Big] \\ &= 1 + 0.1 \cdot \Big[0 + 0.5 \cdot 1 - 1 \Big] \;=\; 0.95 \end{aligned}$$

Update after Second Timestep:

$$\begin{aligned} Q_2(s_1, \texttt{right}) &= Q_1(s_1, \texttt{right}) + \alpha \cdot \Big[R(s_1, \texttt{right}) + \gamma \cdot Q_1(s_2, \texttt{left}) - Q_1(s_1, \texttt{right}) \Big] \\ &= 1 + 0.1 \cdot \Big[1 + 0.5 \cdot 1 - 1 \Big] \;=\; 1.05 \end{aligned}$$

Update after Third Timestep:

$$\begin{aligned} Q_3(s_2, \texttt{left}) &= Q_2(s_2, \texttt{left}) + \alpha \cdot \Big[R(s_2, \texttt{left}) + \gamma \cdot Q_2(s_1, \texttt{right}) - Q_2(s_2, \texttt{left}) \Big] \\ &= 1 + 0.1 \cdot \Big[0 + 0.5 \cdot 1.05 - 1 \Big] \;=\; 0.9525 \end{aligned}$$

Update after Fourth Timestep:

$$\begin{aligned} Q_4(s_1, \texttt{right}) &= Q_3(s_1, \texttt{right}) + \alpha \cdot \Big[R(s_1, \texttt{right}) + \gamma \cdot Q_3(s_2, \texttt{right}) - Q_3(s_1, \texttt{right}) \Big] \\ &= 1.05 + 0.1 \cdot \Big[1 + 0.5 \cdot 1 - 1.05 \Big] \;=\; 1.095 \end{aligned}$$

Q3. [Q-Learning and Value Function Approximation] **14 marks**

Again consider that, instead of knowing the model of the MDP M as explicitly presented in the question **Q1**, you only know the states, $S = \{s_1, s_2, \ldots, s_n\}$, and the actions, $A = \{\texttt{left}, \texttt{right}\}$ (both are applicable from all states). You have also observed the same trajectory of just one episode presented in the question **Q2**. Assume all the *initial* Q-value estimates of the states (prior to any update) as 1; the discount factor as $\gamma = 0.5$; and the step size as $\alpha = 0.1$. Here, $Q(s_k, \cdot)$ denotes the estimated Q-value of state $s_k \in S$.

(a) From the given episodic data spanning across multiple timesteps, calculate the estimated update values after first four timesteps, that is, determine $Q(s_1, \texttt{left})$, $Q(s_1, \texttt{right})$, $Q(s_2, \texttt{left})$ and $Q(s_1, \texttt{right})$, using *tabular Q-learning* approach. **(2 × 4)**

Solution:

Given transitions of the form, $s_i \xrightarrow[r_i]{a_i} s_j \xrightarrow[r_j]{a_j} s_k$, the Q-learning update equation is:

$$Q(s_i, a_i) \leftarrow Q(s_i, a_i) + \alpha \left[r_i + \gamma \cdot \max_{a_j \in \{\texttt{left}, \texttt{right}\}} Q(s_j, a_j) - Q(s_i, a_i) \right]$$

The updated Q-values following Q-learning approach will be:

Update after First Timestep:

$$\begin{aligned} Q(s_1, \texttt{left}) &\leftarrow Q(s_1, \texttt{left}) + \alpha \cdot \left[R(s_1, \texttt{left}) + \gamma \cdot \max\{Q(s_1, \texttt{left}), Q(s_1, \texttt{right})\} - Q(s_1, \texttt{left}) \right] \\ &= 1 + 0.1 \cdot \left[0 + 0.5 \cdot \max\{1, 1\} - 1 \right] = 0.95 \end{aligned}$$

Update after Second Timestep:

$$\begin{aligned} Q(s_1, \texttt{right}) &\leftarrow Q(s_1, \texttt{right}) + \alpha \cdot \left[R(s_1, \texttt{right}) + \gamma \cdot \max\{Q(s_2, \texttt{left}), Q(s_2, \texttt{right})\} - Q(s_1, \texttt{right}) \right] \\ &= 1 + 0.1 \cdot \left[1 + 0.5 \cdot \max\{1, 1\} - 1 \right] = 1.05 \end{aligned}$$

Update after Third Timestep:

$$\begin{aligned} Q(s_2, \texttt{left}) &\leftarrow Q(s_2, \texttt{left}) + \alpha \cdot \left[R(s_2, \texttt{left}) + \gamma \cdot \max\{Q(s_1, \texttt{left}), Q(s_1, \texttt{right})\} - Q(s_2, \texttt{left}) \right] \\ &= 1 + 0.1 \cdot \left[0 + 0.5 \cdot \max\{0.95, 1.05\} - 1 \right] = 0.9525 \end{aligned}$$

Update after Fourth Timestep:

$$\begin{aligned} Q(s_1, \texttt{right}) &\leftarrow Q(s_1, \texttt{right}) + \alpha \cdot \left[R(s_1, \texttt{right}) + \gamma \cdot \max\{Q(s_2, \texttt{left}), Q(s_2, \texttt{right})\} - Q(s_1, \texttt{right}) \right] \\ &= 1.05 + 0.1 \cdot \left[1 + 0.5 \cdot \max\{0.9525, 1\} - 1.05 \right] = 1.095 \end{aligned}$$

(b) Now, we are interested in performing linear function approximation in conjunction with Q-learning. In particular, we have a weight vector, $w = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} \in \mathbb{R}^3$. Given some state $s_k \in S$ and action $a_k \in \{\texttt{left}, \texttt{right}\}$, the featurization of this state-action pair is given as, $\varphi(s_k, \texttt{left}) = \begin{bmatrix} 1 \\ k \\ -1 \end{bmatrix}$ and $\varphi(s_k, \texttt{right}) = \begin{bmatrix} 1 \\ k \\ 1 \end{bmatrix}$, for all k $(1 \leq k \leq n)$. Approximate Q-values are computed as, $\widehat{Q}(s_k, \texttt{left}; w) = w^\intercal \cdot \varphi = w_0 + w_1 \cdot k - w_2$ and $\widehat{Q}(s_k, \texttt{right}; w) = w^\intercal \cdot \varphi = w_0 + w_1 \cdot k + w_2$. Given the parameters w and sample transitions of the form, $s_i \xrightarrow[r_i]{a_i} s_j \xrightarrow[r_j]{a_j} s_k$, the loss function to be minimized here is, $J(w) = \left[r_i + \gamma \cdot \max\limits_{a_j \in \{\texttt{left}, \texttt{right}\}} \widehat{Q}(s_j, a_j; w^-) - \widehat{Q}(s_i, a_i; w)\right]^2$, where $\widehat{Q}(s_j, a_j; w^-)$ is a target network parametrized by fixed weights w^- and $1 \leq i, j \leq n$.

Suppose, you currently have weight vectors $w = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}$ and $w^- = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$, and you observe a sample transition, $s_1 \xrightarrow[+1]{\texttt{right}} s_2 \xrightarrow[0]{\texttt{left}} s_1$. Perform a single gradient update to the parameters w given this sample. Assume the discount factor as $\gamma = 0.5$ and the step size as $\alpha = 0.1$. Write out the gradient $\nabla_w J(w)$ as well as the new parameters w'. Show all computations clearly. **(6)**

Solution:

The gradient of $J(w)$ yields,

$$\begin{aligned} \nabla_w J(w) &= (-2) \cdot \left[r_i + \gamma \cdot \max_{a_j \in \{\texttt{left}, \texttt{right}\}} \widehat{Q}(s_j, a_j; w^-) - \widehat{Q}(s_i, a_i; w)\right] \cdot \nabla_w \widehat{Q}(s_i, a_i; w) \\ &= (-2) \cdot \left[r_i + \gamma \cdot \max\left\{(1 + w_1^- \cdot j - w_2^-), (1 + w_1^- \cdot j + w_2^-)\right\} - (1 + w_1 \cdot i \pm w_2)\right] \cdot \begin{bmatrix} 1 \\ i \\ \pm 1 \end{bmatrix} \end{aligned}$$

Using this, the parameter update with the sample transition, $s_1 \xrightarrow[+1]{\texttt{right}} s_2 \xrightarrow[0]{\texttt{left}} s_1$, is:

$$\begin{aligned} w' &\leftarrow w - \frac{1}{2}\alpha \nabla_w J(w) \\ &= w + 0.1 \cdot \left[r_1 + 0.5 \cdot \max\left\{(w_0^- + w_1^- \cdot 2 - w_2^-), (w_0^- + w_1^- \cdot 2 + w_2^-)\right\} - (w_0 + w_1 \cdot 1 + w_2)\right] \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} + 0.1 \cdot \left[1 + 0.5 \cdot \max\left\{(1 + 0 \cdot 2 + 1), (1 + 0 \cdot 2 - 1)\right\} - (-1 + 1 \cdot 1 + 1)\right] \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} + 0.1 \cdot 1 \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -0.9 \\ 1.1 \\ 1.1 \end{bmatrix} \end{aligned}$$

Note: Alternatively, the parameter update could also be written as: $w' \leftarrow w - \alpha \nabla_w J(w)$.
This is also fine, and the subsequent answer obtained will be:

$$w' \leftarrow \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} + 0.1 \cdot 2 \cdot 1 \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -0.8 \\ 1.2 \\ 1.2 \end{bmatrix}$$

Q4. [Properties of MDP and Optimal Policy] **10 marks**

Suppose we have an infinite-horizon, discounted MDP, $M = \langle S, A, P, R, \gamma \rangle$, with a finite state-action space, that is, $|S \times A| < \infty$, and $0 \leq \gamma < 1$. In the questions that follow, let $Q, Q' : S \times A \to \mathbb{R}$ be any two arbitrary action-value functions and consider any fixed state $s \in S$. Without loss of generality, you may assume that $Q(s,a) \geq Q'(s,a)$, for all $(s,a) \in S \times A$.

(a) Prove or disprove: $|\max_{a \in A} Q(s,a) - \max_{a' \in A} Q'(s,a')| \leq \max_{a \in A} |Q(s,a) - Q'(s,a)|$. **(3)**

Solution:

This inequality is true.

We can start by simply ignoring the absolute value signs on the left-hand side.

Let $a^* = \arg\max_{a \in A} Q(s,a)$. Then,

$$\begin{aligned} \max_{a \in A} Q(s,a) - \max_{a' \in A} Q'(s,a') &= Q(s,a^*) - \max_{a' \in A} Q'(s,a') \\ &\leq Q(s,a^*) - Q'(s,a^*) \\ &\leq \max_{a \in A} \big(Q(s,a) - Q'(s,a)\big) \\ &\leq \max_{a \in A} |Q(s,a) - Q'(s,a)| \end{aligned}$$

Now, take absolute values on both sides of the inequality (the left-hand side is already non-negative) to complete the proof.

(b) Prove or disprove: $|\min_{a\in A} Q(s,a) - \min_{a'\in A} Q'(s,a')| \leq \max_{a\in A} |Q(s,a) - Q'(s,a)|$. **(3)**

Solution:

This inequality is true.

We can start by simply ignoring the absolute value signs on the left-hand side.

Let $a^* = \arg\min_{a'\in A} Q'(s,a')$. Then,

$$\begin{aligned}
\min_{a\in A} Q(s,a) - \min_{a'\in A} Q'(s,a') &= \min_{a\in A} Q(s,a) - Q'(s,a^*) \\
&\leq Q(s,a^*) - Q'(s,a^*) \\
&\leq \max_{a\in A} \big(Q(s,a) - Q'(s,a)\big) \\
&\leq \max_{a\in A} |Q(s,a) - Q'(s,a)|
\end{aligned}$$

Now, take absolute values on both sides of the inequality (the left-hand side is already non-negative) to complete the proof.

(c) Prove or disprove: $\left|\frac{1}{|A|}\sum_{a\in A} Q(s,a) - \frac{1}{|A|}\sum_{a'\in A} Q'(s,a')\right| \leq \max_{a\in A} |Q(s,a) - Q'(s,a)|$. **(4)**

Solution:

This inequality is true.

We can start by simply ignoring the absolute value signs on the left-hand side.

$$\begin{aligned}
\frac{1}{|A|}\sum_{a\in A} Q(s,a) - \frac{1}{|A|}\sum_{a'\in A} Q'(s,a') &= \frac{1}{|A|}\sum_{a\in A}\Big(Q(s,a) - Q'(s,a)\Big) \\
&\leq \frac{1}{|A|}\sum_{a\in A} |Q(s,a) - Q'(s,a)| \\
&\leq \frac{1}{|A|}\sum_{a\in A} \max_{a'\in A} |Q(s,a') - Q'(s,a')| \\
&\leq \frac{1}{|A|}\cdot |A| \cdot \max_{a'\in A} |Q(s,a') - Q'(s,a')| \\
&\leq \max_{a\in A} |Q(s,a) - Q'(s,a)|
\end{aligned}$$

Now, take absolute values on both sides of the inequality (the left-hand side is already non-negative) to complete the proof.

— Question Paper Ends Here —