SVKM'S NMIMS

MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT& ENGINEERING/

SCHOOL OF TECHNOLOGY MANAGEMENT

Academic Year: 2024-2025

Program/s: B TECH/MBA TECH — Year: _IV__ Semester: __VII____

Stream/s ; DS/AI/AIML/AIDS

Subject: ____Reinforcement Learning_____ — Time: __3hrs_ hrs (10:00am to 1:00pm)

Date: 05 / 12 / 2024 — No. of Pages: _9_

Marks: __100_____

Final Examination (2024-25)

Instructions: Candidates should read carefully the instructions printed on the question paper and on the cover of the Answer Book, which is provided for their use.

1) Question No. _1___ is compulsory.
2) Out of remaining questions, attempt any __4___ questions.
3) **In all ___5_ questions to be attempted.**
4) All questions carry equal marks.
5) **Answer to each new question to be started on a fresh page.**
6) **Figures in brackets on the right hand side indicate full marks.**
7) **Assume Suitable data if necessary.**

Q1		Answer briefly:	[20]
CO-1; SO- 1; BL-2	a.	Explain in not more than 5 sentences reinforcement learning framework with the help of suitable diagram using its core components such as Agent, Environment, Reward, Action, and State.	
CO- 2; SO- 1; BL-3	b.	Consider an MDP with two states $\{IN, END\}$. When in IN state, the agent can choose from a set of actions $A = \{STAY, QUIT\}$. If QUIT is selected, then the game will go to END state and agent receives a reward of 5 points. If STAY is selected, the agent will stay in IN state and obtain a reward of 2 points with probability ¾ or the agent will go to END state with probability of ¼ and obtain a reward of 4 points. Draw the state diagram for the given problem	
CO-3 ; SO-1 ; BL-3	c.	Compare Monte Carlo approach and Temporal Difference approach in Reinforcement Learning (provide 3 comparison points between the two approaches)	
CO- 4; SO- 1; BL-3	d.	Compare Value iteration and Policy iteration dynamic programming approaches in terms of computational steps, complexity and convergence.	

Q2 CO-2,3; SO-1; BL-4,5	a.	**Upper Confidence Bound (UCB)** equation is given by $$A_t = argmax_a \left[Q_t(a) + c\sqrt{\frac{\ln(t)}{N_t(a)}} \right]$$ where: $Q_t(a)$ is the estimated value of action 'a' at time step 't'. $N_t(a)$ is the number of times that action 'a' has been selected, prior to time 't'. Assume confidence value $c = 1$ i. Compute the estimated Q value using UCB for the following scenario. Let there be three machines with the following Q values after a total of 100 steps. $Q(M_1)$ =1.73, Q(M_2) =1.83, $Q(M_3)$ =1.89, $Q(M_4)$ =1.55 The number of times each machine is played out of 100 is also given. N_1=25, N_2=20, N_3=30, N_4=25 ii. Identify the machine that will be selected according to UCB for the next iteration. Justify your answer iii. Justify the following statement in 1-2 lines " UCB balances exploration and exploitation"	[10]
CO-2,3; SO-1; BL-4,5	b.	i. With help of a pseudo code, explain **SARSA** on-policy TD control algorithm. Consider the following $Q[S, A]$ table: Assume that α=0.1, and γ=0.5. ii. After the experience ⟨1,1,5,2,1⟩ for <S, A, R, S', A'>, which value of the table gets updated and what is its new value? iii. Justify why SARSA is called an on-policy TD control algorithm in 1-2 sentences. Hint: SARSA learning algorithm is given by the following equation: $$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\left[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)\right].$$	[10]

	State 1	State 2
Action 1	1.5	2.5
Action 2	4	3

Q3 CO-1, 2; SO-1; BL-4,5	a.	Consider a finite, episodic and undiscounted Markov Decision Process (MDP) with states A and B apart from the terminal state. Assume the following two episodes are observed when a Monte-Carlo (MC) evaluation is being carried out. Assume discount factor $\gamma = 1$. **Episode1**: (A,+3) → (A,+2) → (B,−4) → (A,+4) → (B,−3) **Episode2**: (B,−2) → (A,+3) → (B,−3) For example, a sample such as, (B, −2) → (A,+3) → (B,−3), means that the episode starts at B then goes to A, then goes to B again and then terminates. On the way, the agent gets rewards of −2, +3 and −3, respectively. i. Estimate the state value of both A and B using **First Visit Monte-Carlo evaluation.** ii. Estimate the state value of both A and B using **Every Visit Monte-Carlo evaluation** iii. Construct a Markov model that best explains the observations given in the question. Hint: Return is given by $G_t = R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{T-t-1} R_T$	[10]
CO-2,3; SO-1; BL-4	b.	Suppose an agent wants to help Sam make an informed decision about whether to party or relax over the weekend. Sam prefers to party, but is worried about getting sick. Such a problem can be modeled as an MDP with two states, *healthy* and *sick*, and two actions, *relax*, and *party*. Thus, $S = \{healthy(he), sick(se)\}$ $A = \{relax(re), party(pa)\}$ The agent does not know the model (i.e. the transition probabilities from one state to another). It learns from the S, A, R, S' experiences, that are provided in the table below. Assume discount factor, $\gamma = 0.8$ and learning rate, $\alpha = 0.3$. Starting with all Q-values to be zero, compute the Q-values for the set of transitions mentioned in the table below:	[10]

S	A	R	S'	Q-values
he	re	7	he	
he	re	7	he	
he	pa	10	he	
he	pa	10	si	
si	pa	2	si	
si	re	0	si	
si	re	0	he	

Hint: **Q-learning algorithm** equation:

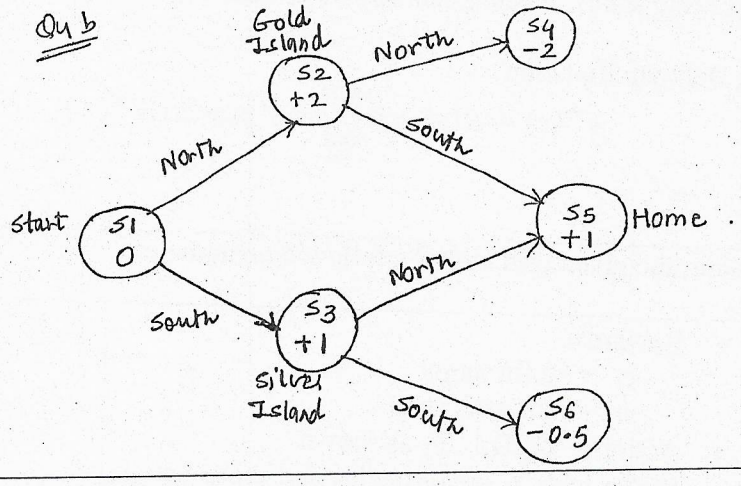
$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\left[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)\right].$$

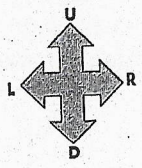
Q4 CO-2,3; SO-1; BL-4	a	For an action 'a', the n^{th} estimate for the action-value, 'Q_n', is given by the sum of all previous rewards obtained for that action, divided by the number of times that action has been selected (i.e. it's just the average value) as follows $$Q_n = \frac{R_1 + R_2 + R_3 + \cdots R_{n-1}}{n-1} = \frac{1}{n-1}\sum_{i=1}^{n-1} R_i$$ i. Mention in 2-3 sentences the limitations of above expression. ii. State the incremental implementation expression, where the new value Q_{n+1} is expressed in terms of the previous estimate Q_n and reward R_n. Consider a 2-armed bandit problem, where one has to choose one of two actions. Assume action a_1 yields a reward of $r = 2$ with probability $p = 0.4$ and 0 otherwise. If you take action a_2, you will receive a reward of $r = 0.8$ with probability $p = 0.8$ and 0 otherwise. The **2-armed bandit game** is played four times and Q-values are updated using the update rule $Q_{t+1}(a) = Q_t(a) + \eta(r - Q_t(a))$. Take $\eta = 0.2$. Assume the initial values of Q as zeros. In trial 1, you choose a_1 and obtain a reward of $r = 2$. In trial 2, you choose a_2 and obtain a reward of $r = 0.8$. In trial 3, Greedy action is exploited and the reward $r = 0$ is obtained In trial 4, a_2 is explored and the reward $r = 0.8$ is obtained. iii. Complete the table given below	[10]

Trial t	**Action selected (a_1 or a_2)**	**r**	**$Q(a_1)$**	**$Q(a_2)$**	**Next action selected**
0			0	0	a_1
1	a_1	2	$0 + 0.2(2 - 0) = 0.4$	-	a_2
2	a_2	0.8	0.4	$0 + 0.2(0.8 - 0) = 0.16$	
3					
4					

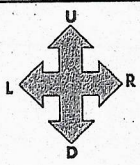
CO-2,3; SO-1; BL-4,5	b	Pirate Ship is anchored in an Island. It has to reach its home safely and it can only take two directions, namely, North and South. The Possible routes the ship can take are- • Go to Gold Island, collect gold and then go to Home (Move North, then South) • Go to Silver Island, collect silver and then go to Home (Move South and North) • Go to home There are two dangers in the first two routes - (a) There is a Bermuda Triangle to the North of Gold Island. (b) There is a Prison to the south of Silver Island. There is uncertainty in directions which means, there are probabilities involved whenever a direction is decided. Probability of moving in the intended direction is 0.8. Example: When the ship wants to move towards North, the probability of the ship moving towards North is 0.8 and there is a probability of 0.2 for South. So how does the Ship reach home with maximum possible benefits? Assume randomly initialize the policy as moving in the north direction for all states. Assume initial values for all state to be zero. Apply **Dynamic Programming based Policy Iteration Algorithm** for one iterations to obtain policy evaluation and policy improvement table. Identify the change policy after the **one iteration.** The state diagram for this scenario is shown below. Also the table for the initial value, random policy and the reward for each state is given below.	[10]

States	S1	S2	S3	S4	S5	S6
Reward	0	1	2	-2	1	-0.5
Values	0	0	0	0	0	0
Initial Random Policy	N	N	N	-	-	-



		Hint: Policy iteration expressions expression Policy evaluation $$V_\pi(s) = r(s) + \gamma \sum P\big((s', r)|(s, \pi(s))\big)\ V(s')$$ Policy improvement $$\pi(s) = argmax_a \sum_{s' \in S} p(s', r|s, a)\, V_\pi(s')$$	
Q5 CO-1,3; SO-1; BL-3,4	a	Consider discount factor $\gamma = 0.8$ and rewards $R_1 = 3, R_2 = -1$, $R_3 = 2,\ R_4 = 4\ and\ R_5 = 5$ with T=5. Compute $G_0, G_1, G_2, G_3, G_4, G_5$ if return is given by $G_t = R_{t+1} + \gamma G_{t+1}$. State with reason whether the agent is *myopic* or *farsighted*	[05]
CO-2,3; SO-1; BL-4	b	Consider the grid world shown below: 2.0 4.4 1.1 0.5 3.0 2.3 1.9 1.6 0.4 0.7 1.1 3.2 4.9 2.9 2.0 1.7  All four actions (right, left, up and down) have probabilities (0.2, 0.4,0.3 and 0.1) respectively. In the figure above, the value function, V^π at each state in the grid is provided, for the case where discount factor $\gamma = 0.85$. Show numerically that the Bellman equation given below applies for the state of the 2nd row and 2nd column valued at 2.3, with respect to its four neighboring states. Assume that R(s) at all states is zero. Hint: Bellman Equation $$V^\pi(s) = R(s) + \gamma \sum_{s'} P(s'|s, \pi(s)) V^\pi(s')$$	[05]
CO-2,3; SO-1; BL-4	c	Consider the given grid with the following parameters • Rewards: o +50 for target o -1 for each state • Actions= {U,D,L,R} as shown • States with X are obstacles and those action would be avoided by the agent	[10]

- Initial random policy for each state is mentioned.
- Initial Q value is equal to zero



Apply Exploring Start Monte Carlo Algorithm for an Episode with T=5 and initial random state-action as (s10,R)

i. Generate state-action sequence for this episode for the given policy
ii. Discounted return given by equation $G = \gamma G + R_{t+1}$. For discount factor $\gamma = 0.8$, compute discounted return list for the given episode
iii. Obtain the average return list and update the Q value
iv. Update the policy in the grid

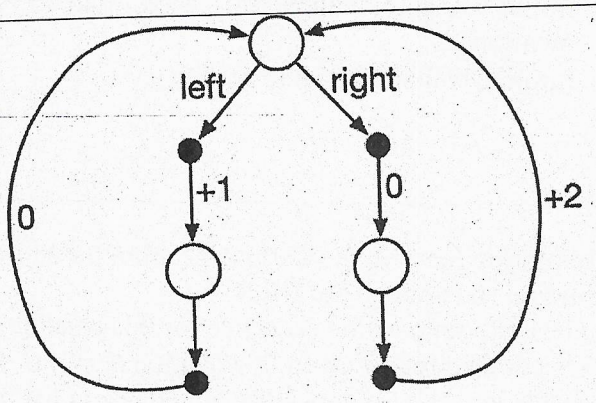
Target	← **s1**	← **s2**	← **s3**	← **s4**
↑ **s5**	↑ **s6**	X	↑ **s7**	↑ **s8**
↑ **s9**	↑ **s10**	← **s11**	X	↑ **s12**
↑ **s13**	X	↑ **s14**	← **s15**	↑ **s16**
← **s17**	← **s18**	↑ **s19**	← **s20**	**Start** ↑ **s21**

Q6
CO-3;
SO-1;
BL-4

a. Consider the **Markov Decision Process (MDP)** depicted in the figure, where the agent begins in state S_0 .
There are two actions available at S_0 : moving **left** and moving **right.**
The rewards associated with these actions are deterministic, as shown in the figure.
Let there be two deterministic policies:

i. π_{left}: Always move left.
ii. π_{right}: Always move right.

[05]

		 a. Calculate the total expected reward for both policies across the **horizon of time step two**, considering the discount factors γ =0 and 0.9. b. Show the calculations for both π_{left} and π_{right} c. Identify which policy is optimal in each scenario. Hint: $Expected\ Return\ G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + ..$	
CO-3; SO-1; BL-2	b.	Explain the concept of Markov Decision Process (MDP) and describe its key components in not more than 5 sentences. State the significance of discount factor γ to estimate returns in long term decision making	[05]
CO-3; SO-1; BL-2	c.	With reference to policy gradient algorithm, answer the following questions- i. "Policy is modeled with a parameterized function and the actions are selected without calculating the value function". Justify this statement. ii. Explain the following statement in a maximum of 2-3 sentences: Value function based methods are oriented towards finding deterministic policies whereas policy search methods are geared towards finding stochastic policies. iii. Why are value-based approaches computationally expensive in continuous spaces? How does Policy-based approach overcome this issue? iv. In general, why does the Actor-Critic method improve the convergence to a solution compared to Monte Carlo REINFORCE method?	[10]
Q7 CO-3; SO-1; BL-4	a	Sitting at your hall on a rainy evening, you hear an episode of experience as follows: At the first step you saw a lightning. At the second step you hear a thunder with a drizzle of rain. At the third step you saw only a drizzle of rain. Then you had a powercut, worth −1 reward, and the episode terminates on the fourth step. All other rewards were zero. The experiment is undiscounted (i.e., $\gamma = 1$).	[10]

		We may represent the state s that you witnessed by a vector of three binary features, light(s) ∈ {0,1}, thunder(s) ∈ {0,1} and drizzle(s) ∈ {0,1}. So, the sequence of feature vectors corresponding to the four steps of this episode can be expressed as, $[1,0,0]^T$, $[0,1,1]^T$, $[0,0,1]^T$ and $[0,0,0]^T$. i. Approximate the state-value function by a linear combination of these features with two parameters: $\varphi(s) = [l \times light(s) + t \times thunder(s) + d \times drizzle(s)]$ If $light\ l = -3$, $thunder\ t = -2$ and $drizzle\ d = 1$, then write down the sequence of approximate values corresponding to this episode. ii. Write down the sequence of λ-returns G_t^λ $(1 \le t \le 3)$ corresponding to this episode, for λ =0.5 and $l = -3, t = -2, d = 1$. Clearly show the detailed evaluations. Hint: The n-step return and λ-return G_t^λ are given as, $G_t^{(n)} = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots + \gamma^{n-1} R_{t+n} + \gamma^n V(s_{t+n})$ $G_t^\lambda = (1-\lambda) \sum_{n=1}^{\infty} \lambda^{n-1} . G_t^{(n)}$ where, $G_t^{(\infty)} = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots + \gamma^{T-1} R_T$, with multiplication factor being λ^{T-t-1}	
CO-1; SO-1; BL-2	b	With reference to Reinforcement Learning, answer the following briefly in not more than 2-3 sentences each i. Concept of exploration and exploitation ii. Difference between model free and model based Reinforcement Learning iii. Concept of Immediate Reinforcement Learning and Full Reinforcement Learning iv. On-Policy learning and Off-Policy Learning	[10]