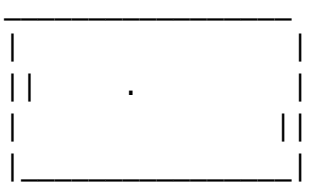
Birla Institute of Technology & Science - Pilani, Hyderabad Campus
First Semester 2020-2021
CS F441: Selected Topics (Reinformcent Learning)
Comprehensive Examination

Type: Open Book **Time: 2 hours** **Max Marks: 70** **Date: 15/05/2021**

Answer *all* questions. All parts of the same question should be answered together.

Q. **(Total Marks: 6)** Consider a small Atari pong game as shown below. The ball moves in 4x4 grid. A player can take three actions: {up, down, hold}.

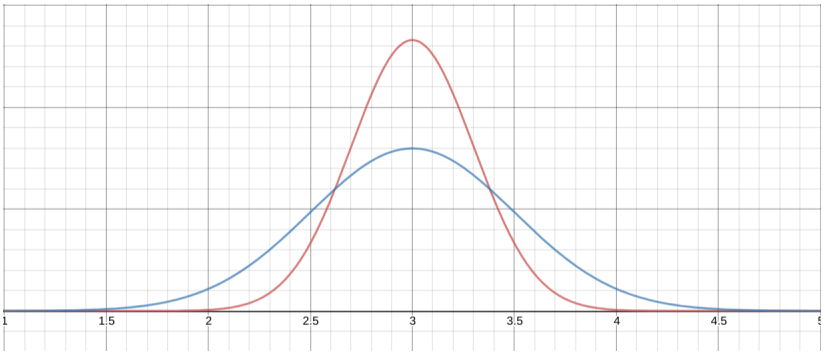


(a) To formulate this problem as MDP, please identify the different components of a state.
(b) What is the total number of states ?

Solution:

(a) Each state consists of four components: {player_pos, opponent_pos, ball_xpos, ball_ypos}.
(b) 4^4 (4 players position x 4 opponent positions x 4 x_axis pos x 4 y_axis pos)

Q. **(Total Marks: 4)** Given the reward distribution of two arms (red and blue) in the figure below, which arm will perform better for large number of experiments ?



Solution:

(a) Both arms will result into the same average reward since the mean for both the distributions is the same.

Q. (**Total Marks: 16)** Consider a multi-arm bandit with k=2 actions, denoted by A and B. The table shows the reward distribution from Actions A (2nd row) and B (3rd row) for 10 time steps. The 4th row shows the value of random variable R1 at each time stamp.

	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	t=9	t=10	
A	10	0	-1	0	0	-3	7	5	-18	10	
B	5	-7	3	5	-11	3	0	3	3	8	
R1	0.57	0.62	0.52	0.5	0.6	0.08	0.36	0.21	0.33	0.82	

Let us assume that you are using greedy action selection. The initial estimates of actions A and B are 0. Note that when the estimates of the two actions are the same, then the action at time stamp *i* will depend on the value of random variable R1 at time stamp *i.* If R1>=0.5, then Action B is selected or if R1<0.5 then Action A is selected.

Based on the table, **please show the estimate calculations** at each step and finally answer the following.

(a) How many times Action A is selected ?
(b) How many times Action B is selected ?
(c) How many times random variable R1 is being used ?
(d) What are the final average estimates of Action A and B after 10 time-stamps ?

Solution:

- Step 1: Both estimates are same. Action B is selected with reward 5 since R1=0.57.
- Step 2: Action B is selected with reward -7. The average estimate of Action B is (5-7)/2=-1.
- Step 3: Action A is selected. The average estimate of Action A is -1.
- Step 4: Both estimates are same. Action B is selected with reward 5 since R1=0.5. The average estimate of Action B is (5-7+5)/3=1.
- Step 5: Action B is selected with reward -11. The average estimate of B is (5-7+5-11)/4=-2.
- Step 6: Action A is selected. The average estimate of A is (-1-3)/2=-2.
- Step 7: Both estimates are same. Action A is selected with reward 7 since R1=0.36. The average estimate of A is (-1-3+7)/3=1.
- Step 8: Action A is selected with reward 5. The average estimate of A is (-1-3+7+5)/4=2.
- Step 9: Action A is selected with reward -18. The average estimate of A is (-1-3+7+5-18)/5=-2.

- Step 10: Both estimates are same. Action B is selected with reward 8. The average estimate of B is (5-7+5-11+8)/5=0.

(a) 5
(b) 5
(c) 4_i's
(d) -2 and 0.

Q. . (**Total Marks: 8)** Write a pseudocode for each actor-learner thread in asynchronous n-step Q-learning.

Replace n-step Q-learning with one-step Q-learning in the following algorithm:

Algorithm 1 Asynchronous one-step Q-learning - pseudocode for each actor-learner thread.

// Assume global shared θ, θ^-, and counter $T = 0$.
Initialize thread step counter $t \leftarrow 0$
Initialize target network weights $\theta^- \leftarrow \theta$
Initialize network gradients $d\theta \leftarrow 0$
Get initial state s
repeat
  Take action a with ϵ-greedy policy based on $Q(s, a; \theta)$
  Receive new state s' and reward r
  $y = \begin{cases} r & \text{for terminal } s' \\ r + \gamma \max_{a'} Q(s', a'; \theta^-) & \text{for non-terminal } s' \end{cases}$
  Accumulate gradients wrt θ: $d\theta \leftarrow d\theta + \frac{\partial (y - Q(s,a;\theta))^2}{\partial \theta}$
  $s = s'$
  $T \leftarrow T + 1$ and $t \leftarrow t + 1$
  if $T \mod I_{target} == 0$ **then**
    Update the target network $\theta^- \leftarrow \theta$
  end if
  if $t \mod I_{AsyncUpdate} == 0$ or s is terminal **then**
    Perform asynchronous update of θ using $d\theta$.
    Clear gradients $d\theta \leftarrow 0$.
  end if
until $T > T_{max}$

Q. . (**Total Marks: 12)** Let us consider a set of ***n*** MDPs, M_i = (S, A, T, R_i, γ) where i =1,2,…,*n*. These MDPs are identical but the reward functions are different. Let us also assume that there exists a policy $\pi: S \rightarrow A$ that is optimal for all MDPs M_i's. Assuming that all these MDPs implement continuing tasks, answer the following:

(a) Consider the MDP P = (S,A,T, R_1+ R_2+$R_{3...}$+R_n, γ) where the reward for each transition in P is the sum of rewards for the same transition under MDPs M_i's. Is π an optimal policy for MDP P as well ? Please answer yes and no and justify your answer with the complete proof.

(b) Consider the MDP Q = (S,A,T, $R_1 \times R_2 \times R_{3\ldots} \times R_n$, γ) where the reward for each transition in Q is the multiplication of rewards for the same transition under MDPs M_i's. Is π an optimal policy for MDP Q as well ? Please answer yes and no and justify your answer with the complete proof.

Solution:

(a) Yes.

$$v_\pi^P(s) = \mathrm{E}_\pi[r_0^P + \gamma r_1^P + \gamma^2 r_2^P \ldots]$$

$$v_\pi^P(s) = \mathrm{E}_\pi[(r_0^{M1} + r_0^{M2} + .. + r_0^{Mn}) + \gamma(r_1^{M1} + r_1^{M2} + \cdots + r_1^{Mn}) \ldots..]$$

$$v_\pi^P(s) = \mathrm{E}_\pi[(r_0^{M1} + \gamma r_1^{M1} + \gamma^2 r_2^{M1}) + (r_0^{M2} + \gamma r_1^{M2} + \gamma^2 r_2^{M2}) + \cdots]$$

$$v_\pi^P(s) = \mathrm{E}_\pi[(r_0^{M1} + \gamma r_1^{M1} + \gamma^2 r_2^{M1})] + \mathrm{E}_\pi[(r_0^{M2} + \gamma r_1^{M2} + \gamma^2 r_2^{M2})] + \cdots]$$

$$v_\pi^P(s) = v_\pi^{M1}(s) + v_\pi^{M2}(s) + \cdots + v_\pi^{Mn}(s)]$$

(b) Not Certain.

$$v_\pi^P(s) = \mathrm{E}_\pi[r_0^P + \gamma r_1^P + \gamma^2 r_2^P \ldots]$$

$$v_\pi^P(s) = \mathrm{E}_\pi[(r_0^{M1} \times r_0^{M2} \times .. \times r_0^{Mn}) + \gamma(r_1^{M1} \times r_1^{M2} \times \ldots \times r_1^{Mn}) \ldots..]$$

Type equation here.

$$v_\pi^P(s) \neq v_\pi^{M1}(s) + v_\pi^{M2}(s) + \cdots + v_\pi^{Mn}(s)]$$

Unlike above, this cannot be reduced to the sum of value functions, we cannot deduce whether π is an optimal policy for MDP Q.

Q.) . (**Total Marks: 12)** Consider a 4x4 gridworld as shown in the figure where the start state is 1 and the end state is 9. When the end-state is reached, the reward is +5 else for other transitions, the reward is -1. In each states, four actions (up, down, left and right) are possible, however, at the edge states, you cannot take the action that takes you outside the grid (for example, in state 7, you cannot take two actions: left and up).

7	8	9 (end)

4	5	6
1 (start)	2	3

Let us assume Q-learning is being used for this MDP and a Q-table after t-steps is shown below.

Q(1,up) = 2	Q(1,down) = -	Q(1,left) = -	Q(1,right) = 4
Q(2,up) = 3	Q(2,down) = -	Q(2,left) = 5	Q(2,right) = 2
Q(3,up) = 6	Q(3,down) = -	Q(3,left) = 1	Q(3,right) = -
Q(4,up) = 5	Q(4,down) = 3	Q(4,left) = -	Q(4,right) = 2
Q(5,up) = 3	Q(5,down) = 2	Q(5,left) = 1	Q(5,right) = 4
Q(6,up) = 6	Q(6,down) = 2	Q(6,left) = 2	Q(6,right) = -
Q(7,up) = -	Q(7,down) = 1	Q(7,left) = -	Q(7,right) = 4
Q(8,up) = -	Q(8,down) = 1	Q(8,left) = 2	Q(8,right) = 6

Further assume that the $\in -greedy$ exploration policy is being used with $\in= 0.3$. The player will be taking a random action (other than the greedy action) if the random number is less than 0.3. Let us assume that at time stamp t, the player is at state 7. The three next random numbers are 0.2, 0.12., 0.5. Make the next three updates following Q-learning with $\alpha = 0.1$ and $\gamma = 0.9$. What are the value Q-learning values of these updates ?

Solution:

- At time stamp t+1, since random number is 0.2, player will not make greedy action i.e., Q(7,right). It will take a random action and since only one action is possible It will take Q(7,down) and reach to State 4. Make the Q-learning update for Q(7,down).
- At time stamp t+2, since random number is 0.12, player will not make greedy action i.e.,g Q(4,up). It will take a random action, either down or right. Show the Q-learning updates for Q(4,down) and Q(4,right).

- Assuming it goes down to 1. At time stamp t+3, since random number is 0.5, make a greedy update at state 1 and go to right. Make a Q-learning update for Q(1,right).

- Assuming it goes right to 5. At time stamp t+3, since random number is 0.5, make a greedy update at state 5 and go to right to state 6. Make a Q-learning update for Q(5,right).

Q) . (**Total Marks: 12)** The function approximator is used for TD(λ) updates. Let us assume that we use a linear function approximator given by **w** $\frac{\partial \ln(\pi(s,a,\theta))}{\partial\theta}$+**v** $\emptyset(s)$ where **w** and **v** are two weight vectors. Given this function approximator, write down TD(λ) updates equations for any time step *t* including TD

error updates and eligibility traces updates. Assume that $\mathbf{z^w}$ is the eligibility trace vector for **w** and $\mathbf{z^v}$ is the eligibility trace vector for **v**. Write the updates for two eligibility traces separately.

Solution:

$$\delta \leftarrow R_t + \gamma \left(\mathbf{w} \frac{\partial \ln\big(\pi(S_{t+1}, A_{t+1}, \theta)\big)}{\partial \theta} + \mathbf{v}\, \emptyset(S_{t+1}) \right) - \mathbf{w} \frac{\partial \ln\big(\pi(S_t, A_t, \theta)\big)}{\partial \theta} - \mathbf{v}\, \emptyset(S_t)$$

$$\boldsymbol{z_t^w} \leftarrow \gamma\lambda \boldsymbol{z_{t-1}^w} + \frac{\partial \ln\big(\pi(S_t, A_t, \theta)\big)}{\partial \theta}$$

$$\boldsymbol{z_t^v} \leftarrow \gamma\lambda \boldsymbol{z_{t-1}^v} + \emptyset(S_t)$$