AI 3000 / CS 5500 : Reinforcement Learning

Assignment № 2

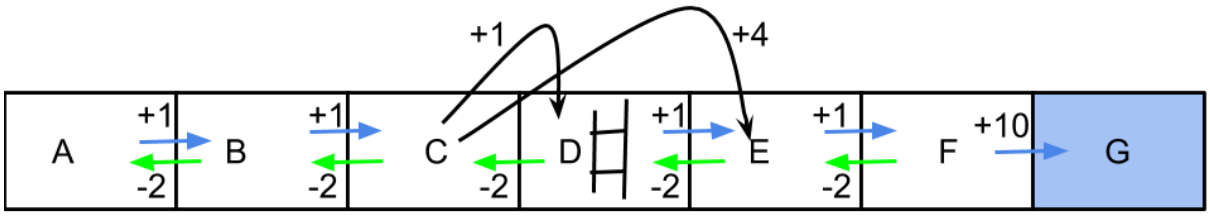
Due Date : 19/10/2021

Teaching Assistants : Chaitanya Devaguptapu and Deepayan Das

Easwar Subramanian, IIT Hyderabad 06/10/2021

Problem 1 : Model Free Prediction and Control

Consider the MDP shown below with states $\{A, B, C, D, E, F, G\}$. Normally, an agent can either move *left* or *right* in each state. However, in state C, the agent has the choice to either move *left* or *jump* forward as the state D of the MDP has an hurdle. There is no *right* action from state C. The *jump* action from state C will place the agent either in square D or in square E with probability 0.5 each. The rewards for each action at each state s is depicted in the figure below alongside the arrow. The terminal state is G and has a reward of zero. Assume a discount factor of $\gamma = 1$.



Consider the following samples of Markov chain trajectories with rewards to answer the questions below

- $A \xrightarrow{+1} B \xrightarrow{+1} C \xrightarrow{-2} B \xrightarrow{+1} C \xrightarrow{+1} D \xrightarrow{+1} E \xrightarrow{+1} F \xrightarrow{+10} G$
- $A \xrightarrow{+1} B \xrightarrow{+1} C \xrightarrow{+1} D \xrightarrow{+1} E \xrightarrow{+1} F \xrightarrow{+10} G$
- $A \xrightarrow{+1} B \xrightarrow{+1} C \xrightarrow{+4} E \xrightarrow{+1} F \xrightarrow{+10} G$
- $A \xrightarrow{+1} B \xrightarrow{+1} C \xrightarrow{+4} E \xrightarrow{-2} D \xrightarrow{+1} E \xrightarrow{+1} F \xrightarrow{+10} G$
- $A \xrightarrow{+1} B \xrightarrow{+1} C \xrightarrow{+4} E \xrightarrow{-2} D \xrightarrow{+1} E \xrightarrow{+1} F \xrightarrow{-2} E \xrightarrow{+1} F \xrightarrow{+10} G$

(a) Evaluate $V(s)$ using first visit Monte-Carlo method for all states s of the MDP. (2 Points)

(b) Which states are likely to have different value estimates if evaluated using every visit MC as compared to first visit MC ? Why ? (1 Point)

(c) Now consider a policy π_f that always move forward (using actions *right* or *jump*). Compute **true** values of $V^{\pi_f}(s)$ for all states of the MDP. (2 Points)

(d) Consider trajectories 2, 3 and 4 from the above list of rollouts. Compute $V^{\pi_f}(s)$ for all states of the MDP using maximum likelihood estimation (2 Points)
[**Hint** : A MLE (or certainity equivalence) estimate is based value estimation computed from sample trajectories. For example, to compute $V(B)$ we need to compute $V(C)$ and one need to calculate state transition probabilities to go from state C to D and E respectively using samples. Use the transition probabilities obtained to compute $V(C)$.]

(e) Suppose, using policy π_f, we collect infinitely many trajectories of the above MDP. If we compute the value function V^{π_f} using Monte Carlo and TD(0) evaluations, would the two methods converge to the same value function ? Justify your answer. (2 Points)

(f) Fill in the blank cells of the table below with the Q-values that result from applying the Q-learning update for the 4 transitions specified by the episode below. You may leave Q-values that are unaffected by the current update blank. Use learning rate $\alpha = 0.5$. Assume all Q-values are initialized to 0. (2 Points)

s	a	r	s	a	r	s	a	r	s	a	r	s
C	jump	4	E	right	1	F	left	-2	E	right	+1	F

	Q(C, left)	Q(C, jump)	Q(E, left)	Q(E, right)	Q(F, left)	Q(F, right)
Initial	0	0	0	0	0	0
Transition 1						
Transition 2						
Transition 3						
Transition 4						

(g) After running the Q-learning algorithm using the four transitions given above, construct a greedy policy using the current values of the Q-table in states C, E and F. (1 Point)

Problem 2 : On Learning Rates

In any TD based algorithm, the update rule is of the following form

$$V(s) \leftarrow V(s) + \alpha_t[r + \gamma V(s') - V(s)]$$

where α_t is the learning rate at the t-th time step. In here, the time step t refers to the t-th time we are updating the value of the state s. Among other conditions, the learning rate α_t has to

obey the Robbins-Monroe condition given by,

$$\sum_{t=0}^{\infty} \alpha_t = \infty$$
$$\sum_{t=0}^{\infty} \alpha_t^2 < \infty$$

for convergence to true $V(s)$. Other conditions being same, reason out if the following values for α_t would result in convergence. (5 Points)

(1) $\alpha_t = \frac{1}{t}$

(2) $\alpha_t = \frac{1}{t^2}$

(3) $\alpha_t = \frac{1}{t^{\frac{2}{3}}}$

(4) $\alpha_t = \frac{1}{t^{\frac{1}{2}}}$

Generalize the above result for $\alpha_t = \frac{1}{t^p}$ for any positive real number p (i.e. $p \in \mathbb{R}^+$)

Problem 3 : Q-Learning

Consider a single state state MDP with two actions. That is, $\mathcal{S} = \{s\}$ and $\mathcal{A} = \{a_1, a_2\}$. Assume the discount factor of the MDP γ and the horizon length to be 1. Both actions yield random rewards, with expected reward for each action being a constant $c \geq 0$. That is,

$$\mathbb{E}(r|a_1) = c \text{ and } \mathbb{E}(r|a_2) = c$$

where $r \sim \mathcal{R}^{a_i}$, $i \in \{1, 2\}$.

(a) What are the true values of $Q(s, a_1)$, $Q(s, a_2)$ and $V^*(s)$? (1 Point)

(b) Consider a collection of n prior samples of reward r obtained by choosing action a_1 or a_2 from state s. Denote $\hat{Q}(s, a_1)$ and $\hat{Q}(s, a_2)$ to be the sample estimates of action value functions $Q(s, a_1)$ and $Q(s, a_2)$, repectively. Let $\hat{\pi}$ be a greedy policy obtained with respect to the estimated $\hat{Q}(s, a_i)$, $i \in \{1, 2\}$. That is,

$$\hat{\pi}(s) = \arg\max_a \hat{Q}(s, a)$$

Prove that the estimated value of the policy $\hat{\pi}$, denoted by $\hat{V}^{\hat{\pi}}$, is a biased estimate of the optimal value function $V^*(s)$. (4 Points)
[**Note** : Assume that actions a_1 and a_2 have been choosen equal number of times.]

(c) Let us now consider that the first action a_1 always gives a constant reward of c whereas the second action a_2 gives a reward $c + \mathcal{N}(-0.2, 1)$ (normal distribution with mean -0.2 and unit variance). Which is the better action to take in expectation ? Would the TD control algorithms like Q-learning or SARSA control, trained using finite samples, always favor the action that is best in expectation ? Explain. (3 Points)

Problem 4 : Importance Sampling

Consider a single state MDP with finite action space, such that $|\mathcal{A}| = K$. Assume the discount factor of the MDP γ and the horizon length to be 1. For taking an action $a \in \mathcal{A}$, let $\mathcal{R}^a(r)$ denote the unknown distribution of reward r, bounded in the range $[0, 1]$. Suppose we have collected a dataset consisting of action-reward pairs $\{(a, r)\}$ by sampling $a \sim \pi_b$, where π_b is a stochastic behaviour policy and $r \sim \mathcal{R}^a$. Using this dateset, we now wish to estimate $V^\pi = \mathbb{E}_\pi[r|a \sim \pi]$ for some target policy π. We assume that π is fully supported on π_b.

(a) Suppose the dataset consists of a single sample (a, r). Estimate V^π using importance sampling (IS). Is the obtained IS estimate of V^π is unbiased ? Explain. (2 Points)

(b) Compute

$$\mathbb{E}_{\pi_b}\left[\frac{\pi(a|\cdot)}{\pi_b(a|\cdot)}\right]$$

(1 Point)

(c) For the case that π_b is a uniformly random policy (all K actions are equiprobable) and π a deterministic policy, provide an expression for importance sampling ratio. (1 Point)

(d) For this sub-question, consider the special case when the reward r for choosing any action is identical, given by a deterministic constant r [i.e., $r = \mathcal{R}(a), \forall a \in \mathcal{A}$]. For a uniform behaviour policy π_b and a deterministic target policy π, calculate the variance of V^π estimated using importance sampling (IS) method. (5 Points)
[**Note :** Variance needs to be estimated under measure π_b]

(e) Derive an upper bound for the variance of the IS estimate of V^π for the general case when the reward distribution is bounded in the range $[0, 1]$. (3 Points)

(f) We now consider the case of multi-state (i.e $|\mathcal{S}| > 1$), multi-step MDP. We futher assume that $P(s_0)$ to be the initial start state distribution (i.e. $s_0 \sim P(s_0)$) where s_0 is the start state of the MDP. Let τ denote a trajectory (state-action sequence) given by, $(s_0, a_0, s_1, a_1, \cdots, s_t, a_t, \cdots)$ with actions $a_{0:\infty} \sim \pi_b$. Let P and Q be joint distributions, over the entire trajectory τ induced by the behaviour policy π_b and a target policy π, respectively. Provide a compact expression for the importance sampling weight $\frac{P(\tau)}{Q(\tau)}$. (3 Points)

[**Note :** A probablity distribution P is fully supported on another probablity distributions Q, if Q does not assign non-zero probablity to any outcome that is assigned non-zero probablity by P].

Problem 5 : Game of Tic-Tac-Toe

Consider a 3×3 Tic-Tac-Toe game. The aim of this problem is to implement a Tic-Tac-Toe agent using Q-learning. This is a two player game in which the opponent is part of the environment.

(a) Develop a Tic-Tac-Toe environment with the following methods. (5 Points)

(1) An **init** method that starts with an empty board position, assigns both player symbols ('X' or 'O') and determines who starts the game. For simplicity, you may assume that the agent always plays 'X' and the opponent plays 'O'.

(2) An **act** method that takes as input a move suggested by the agent. This method should check if the move is valid and place the 'X' in the appropriate board position.

(3) A **print** method that prints the current board position

(4) You are free add other methods inside the environment as you deem fit.

(b) Develop two opponents for the Q-learning agent to train against, namely, a random agent and safe agent (5 Points)

(1) A **random agent** picks a square among all available empty squares in a (uniform) random fashion

(2) A **safe agent** uses the following heuristic to choose a square. If there is a winning move for the safe agent, then the corresponding square is picked. Else, if there is a blocking move, the corresponding square is chosen. A blocking move obstructs an opponent from winning in his very next chance. If there are no winning or blocking moves, the safe agent behaves like the random agent.

(c) The Q-learning agent now has the task to learn to play Tic-Tac-Toe by playing several games against **safe** and **random** opponents. The training will be done using tabular Q-learning by playing 10,000 games. In each of these 10,000 games, a fair coin toss determines who makes the first move. After every 200 games, assess the efficacy of the learning by playing 100 games with the opponent using the full greedy policy with respect to the current Q-table. Record the number of wins in those 100 games. This way, one can study the progress of the training as a function of training epochs. Plot the training progress graph as suggested. In addition, after the training is completed (that is after 10,000 games of training is done), the trained agent's performance is ascertained by playing 1000 games with opponents and recording the total number of wins, draws and losses in those 1000 games. The training and testing process is described below. (10 Points)

(1) Training is done only against the random player. But the learnt Q-table is tested against both random and safe player.

(2) Training is done only against the safe player. But the learnt Q-table is tested against both random and safe player.

(3) In every game of training, we randomly select our opponent. The learnt Q-table is tested against both random and safe player.

(4) Among the three agents developed, which agent is best ? Why ?

(5) Is the Q-learning agent developed unbeatable against any possible opponent ? If not, suggest ways to improve the training process.

[**Note** : A useful diagnostic could be to keep count of how many times each state-action pair is visited and the latest Q value for each state-action pair. The idea is that, if a state-action pair

is visited more number of times, Q value for that state-action pair gets updated frequently and consequently it may be more close to the 'optimal' value. Although, it is not necessary to use the concept of afterstate discussed in the class, it may be useful to accelerate the training process]

ALL THE BEST