AI 3000 / CS 5500 : Reinforcement Learning

Assignment № 1

Due Date : 27/09/2021

Teaching Assistants : Shantam Gulati and Megha Gupta

Easwar Subramanian, IIT Hyderabad 15/09/2021

Problem 1 : Markov Reward Process

Consider a fair four sided dice with faces marked as $\{'1', '2', '3', '4'\}$. The dice is tossed repeatedly and independently. By formulating a suitable Markov reward process (MRP) and using Bellman equation for MRP, find the expected number of tosses required for the pattern $'1234'$ to appear. Specifically, answer the following questions.

(a) Identify the states, transition probablities and terminal states (if any) of the MRP (3 Points)

(b) Construct a suitable reward function, discount factor and use the Bellman equation for MRP to find the 'average' number of tosses required for the pattern $'1234'$ to appear. (7 Points)

[**Explanation : For the target pattern to occur, four consequective tosses of the dice should result in different faces of the dice being on the top, in the specific order** $'1'$**,** $'2'$**,** $'3'$ **and** $'4'$]

Problem 2 : Finite Horizon MDP

Consider a dice game in which a player is eligible for a reward that is equal to $3x^2 + 5$ where x is the value of the face of the dice that comes on top. A player is allowed to roll the dice at most N times. At every time step, after having observed the outcome of the dice roll, the player can pick the eligible reward and quit the game or roll the dice one more time with no immediate reward. If not having stopped before, then, at terminal time N, the game ends and the player gets the reward corresponding to the outcome of dice roll at time N.

The goal of this problem is to model the game as an MDP and formulate a policy that helps the player decide, at any time step $n < N$, whether to continue or quit the game. As a specific case, let's consider a fair four sided dice for this game. It then follows that one can model the game as a finite horizon MDP (with horizon N) consisting of four states $\mathcal{S} = \{1, 2, 3, 4\}$ and two actions $\mathcal{A} = \{Continue, Quit\}$. One can assume that the discount factor (γ) is 1. For any $n \leq N$, denote $V^n(s)$ and $Q^n(s, a)$ as the state and action functions for state s and action a at time step n.

[**Hint :** A finite horizon MDP is solved backwards in time. One first computes the value of a state at terminal time and then use it to compute the value of a state at intermediate times. Note

that the value of a state at any intermediate time is equal to the best action value possible for that state at that time. The best action value for a state, at any time, is evaluated by considering all possible actions from that state at that time.

(a) Evaluate the value function $V^N(s)$ for each state s of the MDP. (1 Point)

(b) Compute $Q^{N-1}(s,a)$ for each state-action pair of the MDP. (2 Points)

(c) Evaluate the value function $V^{N-1}(s)$ for each state s of the MDP. (1 Point)

(d) For any time $2 < n \leq N$, express $V^{n-1}(s)$ recursively in terms of $V^n(s)$. (2 Points)

(e) For any time $2 < n \leq N$, express $Q^{n-1}(s, "Continue")$ in terms of $Q^n(s, "Continue")$. (2 Points)

(f) What is the optimal policy at any time n that lets a player decide whether to continue or quit based on current state s ? (2 Points)

(g) Is the optimal policy stationary or non-stationary ? Explain. (2 Points)

Problem 3 : Value Iteration

Let M be a MDP given by $< \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma >$ with $|\mathcal{S}| < \infty$ and $|\mathcal{A}| < \infty$ and $\gamma \in [0, 1)$. Let $\hat{M} =< \mathcal{S}, \mathcal{A}, \mathcal{P}, \hat{\mathcal{R}}, \gamma >$ be another MDP with a modified reward function $\hat{\mathcal{R}}$ such that

$$\left|\mathcal{R}(s,a,s') - \hat{\mathcal{R}}(s,a,s')\right| = \varepsilon.$$

Given a policy π, let V^π and $\hat{V}^\pi$ be value functions under policy π for MDPs M and $\hat{M}$ respectively.

(a) Derive an expression that relates $V^\pi(s)$ to $\hat{V}^\pi(s)$ for any state $s \in \mathcal{S}$ of the MDP. (5 Points)

(b) Derive an expression that relates the optimal value functions V_* and $\hat{V}_*$. (3 Points)

(c) Will M and $\hat{M}$ have the same optimal policy ? Explain briefly. (2 Points)

Problem 4 : Effect of Noise and Discounting

Consider the grid world problem shown in Figure 1. The grid has two terminal states with positive payoff (+1 and +10). The bottom row is a cliff where each state is a terminal state with negative payoff (-10). The greyed squares in the grid are walls. The agent starts from the yellow state S. As usual, the agent has four actions $\mathcal{A}$ = (Left, Right, Up, Down) to choose from any non-terminal state and the actions that take the agent off the grid leaves the state unchanged. Notice that, if agent follows the dashed path, it needs to be careful not to step into any terminal state at the bottom row that has negative payoff. There are four possible (optimal) paths that an agent can take.

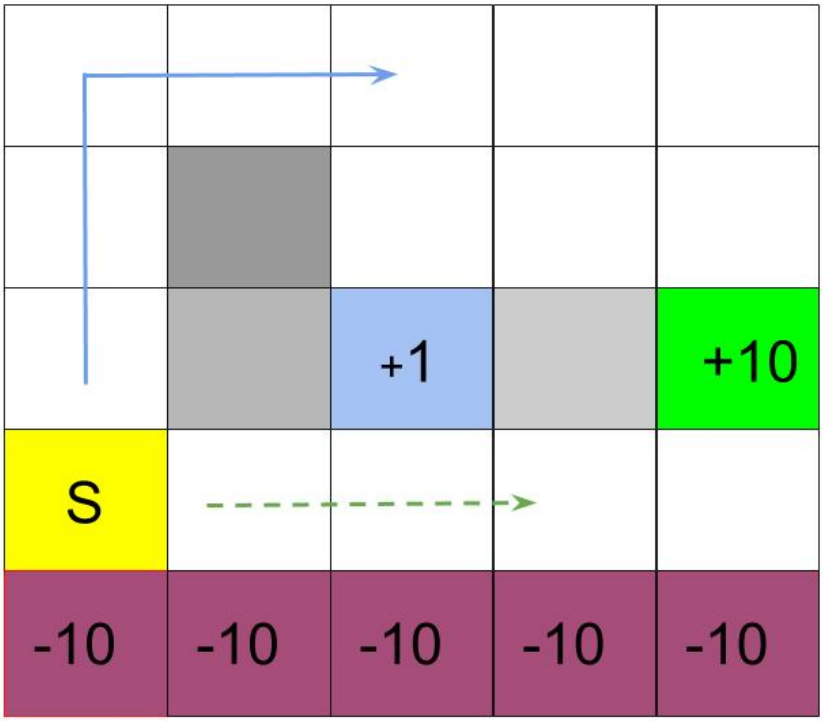

Figure 1: Modified Grid World

- Prefer the close exit (state with reward +1) but risk the cliff (dashed path to +1)
- Prefer the distant exit (state with reward +10) but risk the cliff (dashed path to +10)
- Prefer the close exit (state with reward +1) by avoiding the cliff (solid path to +1)
- Prefer the distant exit (state with reward +10) by avoiding the cliff (solid path to +10)

There are two free parameters to this problem. One is the discount factor γ and the other is the noise factor (η) in the environment. Noise makes the environment stochastic. For example, a noise of 0.2 would mean the action of the agent is successful only 80 % of the times. The rest 20 % of the time, the agent may end up in an unintended state after having chosen an action.

(a) Identify what values of γ and η lead to each of the optimal paths listed above with reasoning. If necessary, you could implement the value iteration algorithm on this environment and observe the optimal paths for various choices of γ and η. (10 Points)

[**Hint** : For the discount factor, try high and low γ values like 0.9 and 0.1 respectively. For noise, consider deterministic and stochastic environment with noise level η being 0 or 0.5 respectively]

Problem 5 : On Value Iteration Algorithm

Let M be an MDP given by $< \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma >$ with $|\mathcal{S}| < \infty$ and $|\mathcal{A}| < \infty$ and $\gamma \in [0, 1)$. We are given a policy π and the task is to evalute $V^{\pi}(s)$ for every state $s \in \mathcal{S}$ of the MDP.

To this end, we use the iterative policy evalution algorithm. It is the analog of the algorithm described in **slide 9 of Lecture 6** for the policy evaluation case. We start the iterative policy evaluation algorithm with an initial guess V_1 and let V_{k+1} be the $k+1$-th iterate of the value function corresponding to policy π. Our constraint on compute infrastructure does not allow us to wait for the successive iterates of the value function to converge to the true value function V^π given by $V^\pi = (I - \gamma P)^{-1} R$. Instead, we let the algorithm terminate at time step $k+1$ when the distance between the sucessive iterates given by $\|V_{k+1} - V_k\|_\infty \leq \varepsilon$ for a given $\varepsilon > 0$.

(a) Prove that the error estimate between the obtained value function estimate V_{k+1} and true value function V^π is given by

$$\|V_{k+1} - V^\pi\|_\infty \leq \frac{\varepsilon\gamma}{1-\gamma}$$

(5 Points)

(b) Prove that the iterative policy evaluation algorithm converges geometrically, i.e.

$$\|V_{k+1} - V^\pi\|_\infty \leq \gamma^k \|V_1 - V_\infty^\pi\|$$

(2 Points)

(c) Let v denote a value function and consider the Bellman optimallity operator given by,

$$L(v) = \max_{a \in \mathcal{A}} \left[\mathcal{R}^a + \gamma \mathcal{P}^a v\right].$$

Prove that the Bellman optimality operator ($\mathcal{L}$) satisfies the monotoniciity property. That is, for any two value functions u and v such that $u \leq v$ (this means, $u(s) \leq v(s)$ for all $s \in \mathcal{S}$), we have $\mathcal{L}(u) \leq \mathcal{L}(v)$ (3 Points)

Problem 6 : On Contractions

(a) Let P and Q be two contractions defined on a normed vector space $< \mathcal{V}, \|\cdot\| >$. Prove that the compositions $P \circ Q$ and $Q \circ P$ are contractions on the same normed vector space. (5 Points)

(b) What can be suitable contraction (or LIpschitz) coeffecients for the contractions $P \circ Q$ and $Q \circ P$? (1 point)

(c) Define operator $\mathcal{B}$ as $\mathcal{F} \circ \mathcal{L}$ where $\mathcal{L}$ is the Bellman optimality operator and $\mathcal{F}$ is any other suitable operator. For example, $\mathcal{F}$ could play the role of a function approximator to the Bellman backup $\mathcal{L}$. Under what conditions would the value iteration algorithm converge to a unique solution if operator $\mathcal{B}$ is used in place of $\mathcal{L}$ (in the value iteration algorithm) ? Explain your answer. (2 Points)

Problem 7 : Programming Value and Policy Iteration

Implement value and policy iteration algorithm and test it on **'Frozen Lake'** environment in openAI gym. **'Frozen Lake'** is a grid-world like environment available in gym. The purpose of this exercise is to to help you get hands on with using gym and to understand the implementation details of value and policy iteration algorithm(s)

This question will not be graded but **will** still come in handy for future assignments.

ALL THE BEST