AI 3000 / CS 5500 : Reinforcement Learning

Assignment № 1

Due Date : 05/09/2024

Couse Instructor : Easwar Subramanian 24/08/2024

Problem 1 : Markov Reward Process

A fair coin is tossed repeatedly and independently. By formulating a suitable Markov reward process and using Bellman equations, find the expected number of tosses required for the pattern HTH to appear. (6 Points)

Problem 2 : Markov Decision Process

A production facility has N machines. If a machine starts up correctly in the morning, it renders a daily revenue of $1\$$. A machine that does not start up correctly, needs to be repaired. A visit by a repair man costs $\frac{N}{2}\$$ per day and he repairs all broken machines on the same day. The repair cost is a lump-sum amount and does not depend on the number of machines that is repaired. A machine that has been repaired always starts up correctly the next day. The number of machines that start up correctly the next day depends on the number of properly working machines at present day and is governed by the probability distribution given in the table below, where m stands for the number of (presently) working machines and n stands for the number of ones that would start up correctly the next day. The goal for the facility manager is to maximize the profits (revenue - costs) earned.

m	$n=0$	$n=1$	$n=2$	$n=3$	$\cdots$	$n=N-1$	$n=N$
$m=1$	$\frac{1}{2}$	$\frac{1}{2}$	0	0	0	0	0
$m=2$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	0	0	0	0
$m=3$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	0	0	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$m=N-1$	$\frac{1}{N}$	$\frac{1}{N}$	$\frac{1}{N}$	$\frac{1}{N}$	$\frac{1}{N}$	$\frac{1}{N}$	0
$m=N$	$\frac{1}{N+1}$	$\frac{1}{N+1}$	$\frac{1}{N+1}$	$\frac{1}{N+1}$	$\frac{1}{N+1}$	$\frac{1}{N+1}$	$\frac{1}{N+1}$

(a) Formulate the above problem as a Markov decision process by ennumerating the state space, action space, rewards and transition probabilities. (3 Points)

(b) Would you use discounted or undiscounted setting for the above MDP formulation ? Justify the answer. (1 Point)

(c) Suppose the facility manager adopts the policy to never call the repair man. Calculate the value of the policy. For this sub-problem assume that the number of machines in the facility to be five. (3 Points)

(d) Perform one iteration the of policy iteration algorithm on the no-repair policy adopted by the facility manager to get an improved policy for the five machine scenario. (3 Points)

Problem 3 : 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 ? (3 Points)

(g) Is the optimal policy stationary or non-stationary ? Explain. (3 Points)

Problem 4 : 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