CSC 581D/ 485D: Topics in Artificial Intelligence: Reinforcement Learning (Spring 2022, Assignment 1)

Instructor: Kui Wu

Due: 23:55 pm, Jan 28, 2022

Each question has two marks: the first one for CSC 581D students, and the second one for CSC 485D students. A zero mark means optional.

1. (10, 10%) For an n-armed bandit problem, what is the probability selecting the greedy action under the ϵ-greedy algorithm?

2. (90, 90%) Set up the programming environment, following the guideline in Chapter 1 (page 11 to 16) of reference book "Deep reinforcement learning with Python" by Nimish Sanghi.

   (a) (50, 50%) Implement an OpenAI Gym environment for the following 5-armed bandit problem: Each arm is modeled as a two-stage Markov Decision Process (MDP) shown in Fig. 1. At State H(igh), the average reward of each arm $q_H^*(a)$ are $2, 4, 6, 8, 10$, respectively, and at State L(ow), the average reward of each arm $q_L^*(a)$ are $0, 1, 2, 3, 1$, respectively. The actual reward when selecting action (arm) a follows normal distribution with mean $q_H^*(a)$ and variance 1 if the arm is at State H and normal distribution with mean $q_L^*(a)$ and variance 1 if the arm is at State L. Initially, all arms are at the H state. We assume that non-selected arms do not change state, i.e., if an arm is not selected at a time step t, its state does not change at this time step. Your Gym environment, named `DynBandit`, does not need to implement an useful policy and only uses the random policy, i.e., a random arm is selected each step. To test *DynBandit*, plot the average reward of 100 independent runs over 10000 steps (i.e., 100 episode, each episode having 10000 steps) for the random policy.

   (b) (40, 0%) Assume that the agent knows the *structure* of the underlying MDP (Fig. 1), including the state transition probabilities. The agent also knows that the actural rewards of each arm, at each state, follow normal distribution of variance 1, but the agent has no knowledge regarding the actual values of $q_H^*(a)$, $q_L^*(a)$. Note that the environment only discloses the reward that the agent receives after each play. In other words, the agent does not know the current state of each arm even if the agent knows the *structure* of the MDP.

   Use the expectation–maximization (EM) algorithm to estimate the values of $q_H^*(a)$, $q_L^*(a)$ using the samples of the corresponding arm obtained so far. Set initial estimations of $q_H^*(a) = 1$, $q_L^*(a) = 0$ for all arms. Based on the estimated $q_H^*(a)$ and $q_L^*(a)$ of an arm, what is the the expected reward if this arm is to be played next time? Write out the formula for calculating the expected reward. **Hint**: you need to first calculate the stationary distribution of the MDP, because the agent does not know the current state of each arm.

Plot the average reward of 100 independent runs over 10000 steps for the following policies: (1). Greedy algorithm, i.e., selecting the arm that has the maximum expected reward; if multiple arms have the same maximum expected reward, randomly select one to break the tie; (2). ϵ-greedy algorithm with $\epsilon = 0.1$; (3). ϵ-greedy algorithm with $\epsilon = 0.15$.

Note: We will not teach the EM algorithm in class. As a graduate student, you should self-learn this classical algorithm if you do not know it. There are many online tutorials and sample code to explain the EM algorithm. **Hint**: Google "EM algorithm Gaussian mixture model."

(c) (0, 40%) Assume that the agent ignores the underlying Markov decision process and simply uses sample-average estimate of action values, with initial value estimate 0 for all arms. Plot the average reward of 100 independent runs over 10000 steps for the following policies: (1). Greedy algorithm; (2). ϵ-greedy algorithm with $\epsilon = 0.1$; (3). ϵ-greedy algorithm with $\epsilon = 0.15$.

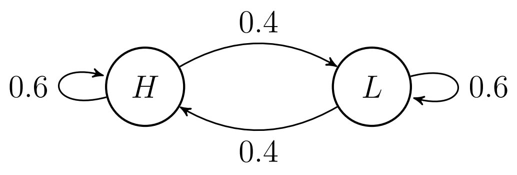

Figure 1: The Markov decision process for an arm.

Note: two files `Gym Programming Environement.ipynb` and `DynBandit.ipynb` are provided to help you finish the assignment. The first file includes a quick summary on how to build Gym environment; the second file includes the skeleton code of `DynBandit` and the RL agent.

Warning: Start your assignment as early as possible! Otherwise, you may not be able to finish it on time.

Deliverables: Zip your answers, including your answer for Q1, and your solution for Q2 (the plot, a Readme file, and your jupyter notebook that follows the Python code template provided with this Asssignment), in one file and submit the zipped file to Brightspace. The Readme file should tell TA (1) your session (CSC 581D or 485D) and (2) how to run your Python code on the jupyter notebook.

The End