CS6700 : Reinforcement Learning
Written Assignment #1

Intro to RL, Bandits, DP — Deadline: 23 Feb 2020, 11:55 pm

Name: Enter Name — **Roll number:** Enter Roll Number

- This is an individual assignment. Collaborations and discussions are strictly prohibited.
- Be precise with your explanations. Unnecessary verbosity will be penalized.
- Check the Moodle discussion forums regularly for updates regarding the assignment.
- Type your solutions in the provided LaTeXtemplate file.
- **Please start early.**

1. (2 marks) You have come across Median Elimination as an algorithm to get $(\epsilon, \delta)-$PAC bounds on the best arm in a bandit problem. At every round, half of the arms are removed by removing arms with return estimates below the median of all estimates. How would this work if we removed only one-fourth of the worst estimated arms instead? Attempt a derivation of the new sample complexity.

Solution:

2. (3 marks) Consider a bandit problem in which you know the set of expected payoffs for pulling various arms, but you do not know which arm maps to which expected payoff. For example, consider a 5 arm bandit problem and you know that the arms 1 through 5 have payoffs 3.1, 2.3, 4.6, 1.2, 0.9, but not necessarily in that order. Can you design a regret minimizing algorithm that will achieve better bounds than UCB? What makes you believe that it is possible? What parts of the analysis of UCB will you modify to achieve better bounds?

Solution: It is possible to design a regret minimizing algorithm that will achieve better bounds than UCB because we know the true payoffs of the arms. That means we already know the true payoff of the best arm which is:

$$q_*(a^*) = \max\{q_*(a) | a \epsilon A\} \tag{1}$$

Before we used to choose the arm with the highest upper bound on payoff, Now we just take the arm which is closest to the true highest payoff.

i.e. pick the action that maximises :

$$Q_j + \sqrt{\frac{2 \ln t}{N_j}} \, \Delta_j \tag{2}$$

where $\Delta_j = q_*(a^*) - q_*(j)$

3. (3 marks) Suppose you face a 2-armed bandit task whose true action values change randomly from time step to time step. Specifically, suppose that, for any time step, the true values of actions 1 and 2 are respectively 0.1 and 0.2 with probability 0.5 (case A), and 0.9 and 0.8 with probability 0.5 (case B).

   (a) (1 mark) If you are not able to tell which case you face at any step, what is the best expectation of success you can achieve and how should you behave to achieve it?

   Solution: If we are not able to tell which case we are facing at a step then we can hold separate estimates for case A and case B. So, the best we can do is pick the arm with the best average expectation

   $$Q_1 = (0.1 + 0.9)/2 = 0.5$$
   $$Q_2 = (0.2 + 0.8)/2 = 0.5$$

   From the average expectations computed, to get the best expected reward, we have to choose any one arm and continue picking it.

   (b) (2 marks) Now suppose that on each step you are told whether you are facing case A or case B (although you still don't know the true action values). This is an associative search task. What is the best expectation of success you can achieve in this task, and how should you behave to achieve it?

   Solution: If we know which case we are facing at each step, we can treat each case as a different bandit problem and learn their expected action values. In this case, the best expectation is

   $$Q = (0.2 + 0.9)/2 = 0.55$$

   We can achieve this expected value by picking the arm suggested by the respective bandit problem for each case.

4. (5 marks) Many tic-tac-toe positions appear different but are really the same because of symmetries.

(a) (2 marks) How might we amend the learning process described above to take advantage of this? In what ways would this change improve the learning process?

Solution:
We can take advantage of symmetries in tic-tac-toe by updating by considering all the symmetric states as one state. Thus reducing the state space many-fold. This would reduce the memory requirement and the time taken to learn the policy.

(b) (1 mark) Suppose the opponent did not take advantage of symmetries. In that case, should we? Is it true, then, that symmetrically equivalent positions should necessarily have the same value?

Solution:
For Instance if the opponent didn't take advantage of symmetries, the opponent would possibly be taking sub-optimal actions from one state (say S_1) compared to another state (say S_2) which is symmetric to S_1. So, here we can exploit the mistakes of opponent by not taking advantage of symmetry and get higher return.
Hence symmetrically equivalent positions need not have same value.

(c) (2 marks) Suppose, instead of playing against a random opponent, the reinforcement learning algorithm described above played against itself, with both sides learning. What do you think would happen in this case? Would it learn a different policy for selecting moves?

Solution:
By playing against itself the RL agent is learning from both sides. There is no difference for the RL agent whether side-1 wins or side-2. Initially both sides will be taking random steps and learning form it. As learning progressing both sides are trying to improve the policy against the other side. Eventually the agent learns to get to positions from where it cannot lose and keeps drawing against itself. And the policy will converge to optimal policy.

5. (1 mark) Ego-centric representations are based on an agent's current position in the world. In a sense the agent says, I don't care where I am, but I am only worried about the position of the objects in the world relative to me. You could think of the agent as being at the origin always. Comment on the suitability (advantages and disadvantages) of using an ego-centric representation in RL.

Solution: Ego-centric representation in RL is very good at remembering and dealing with cliff like situations which can occur at multiple places in the world, Whereas in normal version it would treat that as a new place.
However when we consider long term rewards this representation at times fail to identify bigger rewards. And also is very unstable in convergence.

6. (2 marks) Consider a general MDP with a discount factor of γ. For this case assume that the horizon is infinite. Let π be a policy and V^π be the corresponding value function. Now suppose we have a new MDP where the only difference is that all rewards have a constant k added to them. Derive the new value function V^π_{new} in terms of V^π, c and γ.

Solution:

$$V^\pi(s) = E_\pi[G_t|S_t = s]$$
$$V^\pi(s) = E_\pi[\sum_{j=1}^{j=\infty} \gamma^{j-1} R_{t+j}|S_t = s] \tag{3}$$
$$V^\pi_{new}(s) = E_\pi[G_t|S_t = s]$$
$$V^\pi_{new}(s) = E_\pi[\sum_{j=1}^{j=\infty} \gamma^{j-1}(R_{t+j} + k)|S_t = s]$$
$$V^\pi_{new}(s) = E_\pi[\sum_{j=1}^{j=\infty} \gamma^{j-1}(R_{t+j})|S_t = s] + (\sum_{j=1}^{j=\infty} \gamma^{j-1} k)$$
$$V^\pi_{new}(s) = V^\pi(s) + \frac{k}{1-\gamma} \tag{4}$$

7. (4 marks) An ϵ-soft policy for a MDP with state set $\mathcal{S}$ and action set $\mathcal{A}$ is any policy that satisfies

$$\forall a \in \mathcal{A}, \forall s \in \mathcal{S} : \pi(a|s) \geq \frac{\epsilon}{|\mathcal{A}|}$$

Design a stochastic gridworld where a deterministic policy will produce the same trajectories as a ϵ-soft policy in a deterministic gridworld. In other words, for every trajectory under the same policy, the probability of seeing it in each of the worlds is the same. By the same policy I mean that in the stochastic gridworld, you have a deterministic policy and in the deterministic gridworld, you use the same policy, except for ϵ fraction of the actions, which you choose uniformly randomly.

(a) (2 marks) Give the complete specification of the world.

Solution:

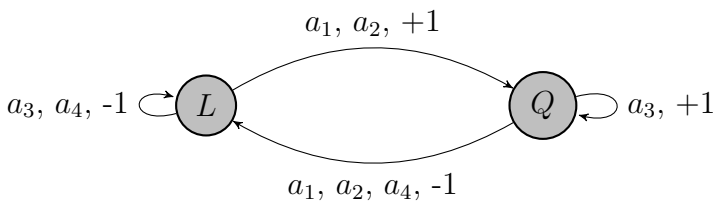
(b) (2 marks) Will SARSA on the two worlds converge to the same policy? Justify.

Solution:

8. (7 marks) You receive the following letter:
Dear Friend, Some time ago, I bought this old house, but found it to be haunted by ghostly sardonic laughter. As a result it is hardly habitable. There is hope, however, for by actual testing I have found that this haunting is subject to certain laws, obscure but infallible, and that the laughter can be affected by my playing the organ or burning incense. In each minute, the laughter occurs or not, it shows no degree. What it will do during the ensuing minute depends, in the following exact way, on what has been happening during the preceding minute: Whenever there is laughter, it will continue in the succeeding minute unless I play the organ, in which case it will stop. But continuing to play the organ does not keep the house quiet. I notice, however, that whenever I burn incense when the house is quiet and do not play the organ it remains quiet for the next minute. At this minute of writing, the laughter is going on. Please tell me what manipulations of incense and organ I should make to get that house quiet, and to keep it so.
Sincerely,
At Wits End

(a) (3 marks) Formulate this problem as an MDP (for the sake of uniformity, formulate it as a continuing discounted problem, with $\gamma = 0.9$. Let the reward be +1 on any transition into the silent state, and -1 on any transition into the laughing state.) Explicitly give the state set, action sets, state transition, and reward function.

Solution: State set, S = {Laughing (L), Quiet (Q) }
Action set, A = { Both Organ and Incense(say a_1), Only Organ(say a_2), Only incense (say a_3), No Organ or Incense (say a_4) }



(b) (2 marks) Starting with simple policy of **always** burning incense, and not playing organ, perform a couple of policy iterations.

Solution:

(c) (2 marks) Finally, what is your advice to "At Wits End"?

Solution: As the starting state is laughing, I would advice "At Wits End" to play organ till the laughing stops and then to burn incense continuously which keeps the house quiet forever.

9. (4 marks) Consider the task of controlling a system when the control actions are delayed. The control agent takes an action on observing the state at time t. The action is applied to the system at time $t + \tau$. The agent receives a reward at each time step.

(a) (2 marks)What is an appropriate notion of return for this task?

Solution: For the state at time "t", the control action is applied at time "$t+\tau$". So, the appropriate notion of return for this task is:

$$G_t = R_{t+\tau+1} + \gamma R_{t+\tau+2} + ...$$
$$G_t = \sum_{j=0}^{j=\infty} \gamma^j \; R_{t+\tau+j+1} \tag{5}$$

(b) (2 marks) Give the TD(0) backup equation for estimating the value function of a given policy.

Solution:

$$V^\pi(s_t) = V^\pi(s_t) + \alpha \left[r_{t+\tau+1} + \gamma \, V^\pi(s_{t+\tau+1}) - V^\pi(s_t)\right] \tag{6}$$