CS 234 Midterm - Winter 2018-19

**Do not turn this page until you are instructed to do so.

Instructions

Please answer the following questions to the best of your ability. Read all the questions first before answering. You have **80** minutes to complete the exam. The exam is closed-book and closed-internet. Additionally use of all electronic items during the exam is prohibited. However, you may use one one-sided letter-sized page of notes as reference. All of the intended answers can be written well within the space provided. Good luck!

Stanford University Honor Code

The following is a statement of the Stanford University Honor Code:

1. The Honor Code is an undertaking of the students, individually and collectively: (1) that they will not give or receive aid in examinations; that they will not give or receive unpermitted aid in class work, in the preparation of reports, or in any other work that is to be used by the instructor as the basis of grading; (2) that they will do their share and take an active part in seeing to it that others as well as themselves uphold the spirit and letter of the Honor Code.
2. The faculty on its part manifests its confidence in the honor of its students by refraining from proctoring examinations and from taking unusual and unreasonable precautions to prevent the forms of dishonesty mentioned above. The faculty will also avoid, as far as practicable, academic procedures that create temptations to violate the Honor Code.
3. While the faculty alone has the right and obligation to set academic requirements, the students and faculty will work together to establish optimal conditions for honorable academic work.

By signing your name below, you acknowledge that you have abided by the Stanford Honor Code while taking this exam.

Signature:

Name:

SUNet ID:

Grading (For Midterm Graders Only)

Question #	1	2	3	4	**Total**
Maximum Points	25	10	20	25	**80**
Student Grade					

Question 1 – True/False and Short Answers [25 pts]

(A) Circle True or False. Provide a one-sentence justification for each question. [2 pts each]

1. True False An optimal policy of an infinite-horizon MDP is always stationary (i.e. it does not vary with the time step).

2. True False When using a linear value function approximation, both Monte Carlo and TD(0) policy evaluations achieve the minimum mean squared error possible at convergence.

3. True False All discounted MDPs with finite states and action spaces with bounded reward (i.e. $|R(s, a)| \leq R_{\max}$ for some $R_{\max} \geq 0$) have bounded returns.

4. True False Monte Carlo policy evaluation does not require the environment to be Markovian.

5. True False Double Q-learning helps training by directly estimating the advantage function.

(B) For the next few questions answer in 2 - 3 sentences. [3 pts each]

1. Explain why Q-learning is off-policy and SARSA is on-policy.

2. What is the challenge with using DQN when the action space is continuous?

3. If you need an algorithm that will definitely reach a local optimum, should you use DQN or REINFORCE, and why? Assume that the state space is very high-dimensional and cannot be stored in memory.

4. State two advantages of using value function approximation compared to a tabular representation.

5. Suppose we gathered infinitely many trajectories for an MDP with finite state space. If we run tabular Monte Carlo and TD(0) policy evaluations, would the two methods converge to the same value function? Justify your answer.

Question 2 – Value Iteration [10 pts]

Suppose we have an MDP $(\mathcal{S}, \mathcal{A}, R, P, \gamma)$, where $\mathcal{S}$ is a finite state space and $\mathcal{A}$ is a finite action space. This MDP is defined such that $R(s, a) \geq 0$ for all state action pairs (s, a). Furthermore, suppose that for every state $s \in \mathcal{S}$, there is some action $a_s \in \mathcal{A}$ such that $P(s' = s \mid s, a_s) \geq p$, where $0 \leq p \leq 1$ is some constant probability.

Consider performing value iteration on this MDP. Let $V_t(s)$ be the value of state s after t iterations. We initialize to $V_0(s) = 0$ for all states s.

(a) Prove that for all states s and $t \geq 0$, $V_{t+1}(s) \geq p\gamma V_t(s)$. (5 pts)

(b) During a single iteration of the Value Iteration algorithm, we typically iterate over the states in $\mathcal{S}$ in some order to update $V_t(s)$ to $V_{t+1}(s)$ for all states s. Is it possible to do this iterative process in parallel? Explain why or why not. (5 pts)

Question 3 – Q-Learning and Function Approximation [20 pts]

Consider an episodic, deterministic chain MDP with $n = 7$ states assembled in a line.

The possible actions are $a \in \{-1, 1\}$, and the transition function is deterministic such that $s' = s + a$. Note that as an exception, taking $a = -1$ from $s = 1$ keeps us in $s = 1$, and taking $a = 1$ from $s = 7$ keeps us in $s = 7$.

We have a special goal state, $g = 4$, such that taking any action **from** g ends the episode with a reward of $r = 0$. From all other states, any action incurs a reward of $r = -1$. We let $\gamma = 1$.

The chain MDP is pictured in Figure 1, with the goal state s_4 shaded in.

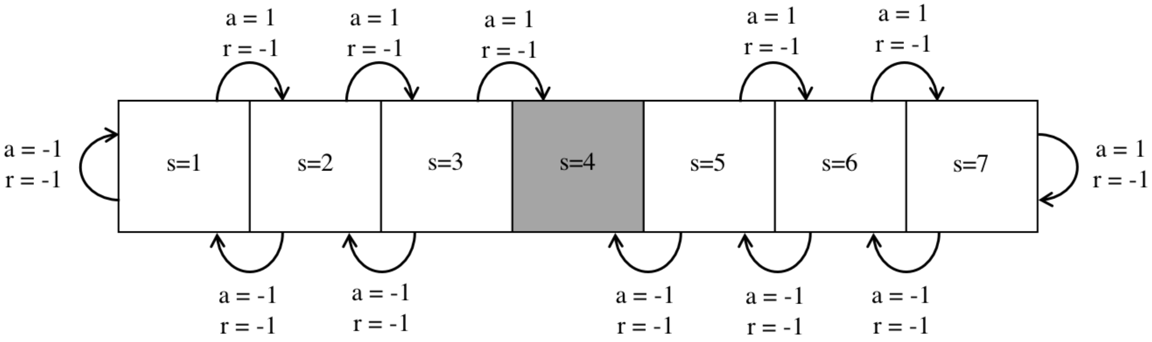

Figure 1: Chain MDP

By inspection, we see that $V^*(s) = -|s - 4|$.

(a) We would like to perform tabular Q-learning on this chain MDP. Suppose we observe the following 4 step trajectory (in the form $(state, action, reward)$):

$$(3, -1, -1), (2, 1, -1), (3, 1, -1), (4, 1, 0)$$

Suppose we initialize all Q values to 0. Use the tabular Q-learning update to give updated values for

$$Q(3, -1), Q(2, 1), Q(3, 1)$$

assuming we process the trajectory in the order given from left to right. Use the learning rate $\alpha = \frac{1}{2}$. (5 pts)

Now, we are interested in performing linear function approximation in conjunction with Q-learning. In particular, we have a weight vector

$$w = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} \in \mathbb{R}^3$$

Given some state s and action $a \in \{-1, 1\}$, the featurization of this state, action pair is: $\begin{bmatrix} s \\ a \\ 1 \end{bmatrix}$

To approximate the Q-values, we compute

$$\hat{q}(s, a; w) = w^T \begin{bmatrix} s \\ a \\ 1 \end{bmatrix} = w_0 * s + w_1 * a + w_2$$

Given the parameters w and a single sample (s, a, r, s'), the loss function we will minimize is

$$J(w) = (r + \gamma \max_{a'} \hat{q}(s', a'; w^-) - \hat{q}(s, a; w))^2$$

where $\hat{q}(s', a'; w^-)$ is a target network parametrized by fixed weights w^-.

(b) Suppose we currently have a weight vectors

$$w = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}, \quad w^- = \begin{bmatrix} 1 \\ -1 \\ -2 \end{bmatrix}$$

and we observe a sample $(s = 2, a = -1, r = -1, s' = 1)$
Perform a single gradient update to the parameters w given this sample. Use the learning rate $\alpha = \frac{1}{4}$. Write out the gradient $\nabla_w J(w)$ as well as the new parameters w'. Show all work. (10 pts)

(c) The optimal Q function $Q^*(s,a)$ is exactly representable by some neural network architecture $\mathcal{N}$. Suppose we perform Q-learning on this MDP using the architecture $\mathcal{N}$ to represent the Q-values. Suppose we randomly initialize the weights of a neural net with architecture $\mathcal{N}$ and collect infinitely many samples using an exploration strategy that is greedy in the limit of infinite exploration (GLIE). Are we guaranteed to converge to the optimal Q function $Q^*(s,a)$? Explain your answer. (5 pts)

Question 4 – Certainty Equivalence Estimate [25 pts]

Consider the discounted ($\gamma < 1$), infinite-horizon MDP $M = (S, A, P, R, \gamma)$, where we do not know the true reward function $R(s, a) \in \mathbb{R}$ and state transition probabilities $P(s'|s, a)$. With a slight abuse of notation, we will also write $P(s, a) \in \mathbb{R}^{|S|}$ to denote the vector of transition probabilities of size $|S|$, whose values sum up to 1. For a given policy π, V_M^π denotes the value function of π in M.

Fortunately, we are given estimates of R and P, namely, $\widehat{R}$ and $\widehat{P}$, respectively, with the following properties:

$$\max_{s,a} |\widehat{R}(s,a) - R(s,a)| < \epsilon_R$$

$$\max_{s,a} \|\widehat{P}(s,a) - P(s,a)\|_1 < \epsilon_P$$

$$\text{where } \|\cdot\|_1 \text{ is the } L_1 \text{ norm for } \mathbf{x} \in \mathbb{R}^n\text{: } \|\mathbf{x}\| = \sum_{i=1}^{n} |x_i|$$

We can then define the *approximate MDP* $\widehat{M} = (S, A, \widehat{P}, \widehat{R}, \gamma)$ and let $V_{\widehat{M}}^\pi$ be the value function of policy π in $\widehat{M}$. For simplicity, assume that both reward functions are bounded within $[0, R_{\max}]$.

(a) Show that for all policies π and states s, the value function is bounded as follows:

$$0 \leq V^\pi(s) \leq \frac{R_{\max}}{1-\gamma} \tag{1}$$

(5 pts)

(b) Let V be the value function of an arbitrary policy. Show that for all state-action pairs (s, a), the following holds:

$$\sum_{s' \in S} \left[V(s') \left(\widehat{P}(s'|s,a) - P(s'|s,a) \right) \right] \leq \epsilon_P \frac{R_{\max}}{2(1-\gamma)} \tag{2}$$

(7 pts)

(c) Using part (b), show that for any policy π, the error in state value is bounded as follows:

$$\|V^{\pi}_{\widehat{M}} - V^{\pi}_{M}\|_{\infty} \leq \frac{\epsilon_R}{1-\gamma} + \gamma \epsilon_P \frac{R_{\max}}{2(1-\gamma)^2} \tag{3}$$

where $\|\cdot\|_{\infty}$ is the L_{∞} norm: $\|V^{\pi}_{\widehat{M}} - V^{\pi}_{M}\|_{\infty} = \max_s |V^{\pi}_{\widehat{M}}(s) - V^{\pi}_{M}(s)|$. (8 pts)

(d) Let π^* and $\widehat{\pi}^*$ be (deterministic) optimal policies in M and $\widehat{M}$, respectively. Using eq. (3), show that the following bound holds for all $s \in S$:

$$V_M^{\pi^*} - V_M^{\widehat{\pi}^*} \leq 2\left(\frac{\epsilon_R}{1-\gamma} + \gamma\epsilon_P \frac{R_{\max}}{2(1-\gamma)^2}\right) \tag{4}$$

(5 pts)

Note: What this means is that if we use a policy that is optimal in $\widehat{M}$, the amount we lose in value compared to the true optimal policy is bounded by the error in our approximation. In other words, the better the approximation, the closer we are to the true optimal policy!