CS 747 (Autumn 2015): End-semester Examination

Instructor: Shivaram Kalyanakrishnan

9.30 a.m. – 12.30 p.m., November 21, 2015, LA 301

Total marks: 25

Note. Provide brief justifications and/or calculations along with each answer to illustrate how you arrived at the answer.

Question 1. Consider the following on-line learning method to estimate the expected value of a real-valued random variable X. We begin with an initial estimate μ_0, and then for $t = 1, 2, \dots,$

- Obtain x_t as an i.i.d. sample of X, and
- Revise our estimate as $\mu_t \leftarrow (1 - \alpha_t)\mu_{t-1} + \alpha_t x_t$.

α_t is the learning rate at step t. Assume X is normally distributed with mean μ and variance σ^2.

1a. If we choose $\alpha_t = \alpha$, where $\alpha \in (0, 1)$ is a constant, what is $\mathbb{E}[\mu_t]$? [2 marks]

1b. If we choose $\alpha_t = \frac{1}{t}$, what is $\mathbb{E}[\mu_t]$? [2 marks]

Question 2. Consider an MDP with three states, s_1, s_2, and s_3, of which the last is a terminal state. Episodes can start either at s_1 or at s_2. The MDP, together with the policy π being followed by an agent, is such that if at s_1, the agent necessarily goes to s_2, gaining a reward of 1 for the transition. If in s_2, the agent necessarily goes to s_3—but in this case, the reward can either be -3 or 4. No discounting is used to calculate values; that is, $\gamma = 1$.

In its first 40 episodes, the following are the number of times each trajectory has been encountered by the agent:

$s_1, \pi(s_1), 1, s_2, \pi(s_2), -3, s_3$	15 times,
$s_1, \pi(s_1), 1, s_2, \pi(s_2), 4, s_3$	5 times,
$s_2, \pi(s_2), -3, s_3$	10 times, and
$s_2, \pi(s_2), 4, s_3$	10 times.

2a. Based on this data, what are the Monte Carlo estimates of $V^\pi(s_1)$ and $V^\pi(s_2)$? [1 mark]

2b. To what estimates of $V^\pi(s_1)$ and $V^\pi(s_2)$ would TD(0) converge if run repeatedly over these trajectories? [1 mark]

Question 3. You are given access to an efficient planner, which, given an MDP M as input, computes an optimal policy $\pi^\star$ for M:

$$\pi^\star = \text{RUN–PLANNER}(M).$$

Your task is to construct a *learning* algorithm L that makes meaningful use of RUN-PLANNER as a subroutine. An agent that applies L while sequentially sampling an MDP (S, A, R, T, γ) must eventually start taking optimal actions with probability 1. Provide pseudocode for L. [3 marks]

Question 4. An agent interacts with a 3-state, 3-action MDP, in which the discount factor $\gamma = 1/2$. The agent initialises its estimate of the action value function as in the table below.

Q	a_1	a_2	a_3
s_1	5	6	9
s_2	0	-1	2
s_3	6	4	-3

The agent then encounters the following trajectory:

$$s_2, a_2, 4, s_2, a_3, 0, s_3, a_2 \text{ (4 and 0 are rewards).}$$

4a. If the agent was implementing Q-learning with a constant learning rate of $\alpha = 0.1$, what would the action value table be after making the first two learning updates? [2 marks]

4b. If the agent was implementing Sarsa with a constant learning rate of $\alpha = 0.1$, what would the action value table be after making the first two learning updates? [2 marks]

Question 5. Explain tile coding, describing its different configuration parameters and their effects. [3 marks]

Question 6. In an MDP with three states, s_1, s_2, and s_3, a policy π is such that $V^\pi(s_1) = 4$ $V^\pi(s_2) = 6$, and $V^\pi(s_3) = -5$. Under π, the steady-state probability of being in s_1 is 1/2, and the probabilities of being in s_2 and s_3 are each 1/4. A learning agent seeks to approximate $V^\pi(\cdot)$ as

$$V^\pi(\cdot) \approx w_1 f(\cdot) + w_2,$$

where $f(\cdot)$ is a state feature, and w_1 and w_2 are the weights to learn. Features for the three states are as follow: $f(s_1) = 2$, $f(s_2) = 1$, and $f(s_3) = -1$.

6a. If the agent performs Monte Carlo policy evaluation with this linear generalisation scheme, to what values will w_1 and w_2 converge? [3 marks]

6b. How would your answer change if the agent uses linear TD(0) instead for policy evaluation? [1 mark]

Question 7. Consider the following zero sum matrix game, with two players A (row) and B (column). A can take actions a_1 and a_2, while B can take actions b_1 and b_2. Each entry in the matrix shows A's reward when A and B play actions from the corresponding row and column (B gets the negative of the same reward).

	b_1	b_2
a_1	1	-1
a_2	-2	0

7a. What is A's minimax strategy? What is the least reward that A can possibly get by playing its minimax strategy? [2 marks]

7b. What is B's minimax strategy? What is the least reward that B can possibly get by playing its minimax strategy? [2 marks]

7c. Suppose A plays a strategy in which a_1 is picked with probability 0.1, and a_2 is picked with probability 0.9. What is the maximum possible reward B can get? [1 mark]

Solutions

1a. Unrolling the recursion, we get for $t = 1, 2, \ldots$:

$$\mu_t = (1-\alpha)^t \mu_0 + \sum_{\tau=1}^{t} \alpha(1-\alpha)^{\tau-1} x_\tau.$$

Thus $\mathbb{E}[\mu_t] = (1-\alpha)^t \mu_0 + \sum_{\tau=1}^{t} \alpha(1-\alpha)^{\tau-1} \mathbb{E}[x_\tau] = (1-\alpha)^t \mu_0 + (1-(1-\alpha)^t)\mu$.

1b. Unrolling the recursion, we get for $t = 1, 2, \ldots$:

$$\mu_t = \frac{1}{t} \sum_{\tau=1}^{t} x_\tau,$$

and therefore $\mathbb{E}[\mu_t] = \mu$.

2a. There are 20 trajectories passing through s_1, and the total reward accrued over them, subsequent to passing s_1, is $(1 + -3) \times 15 + (1 + 4) \times 5 = -5$. Hence, the Monte Carlo estimate of $V^\pi(s_1)$ is $-5/20 = -1/4$. Similarly, there are 40 trajectories passing through s_2, and the total reward accrued over them, subsequent to passing s_2, is $-3 \times 15 + 4 \times 5 + -3 \times 10 + 4 \times 10 = -15$. The resulting Monte Carlo estimate of $V^\pi(s_2)$ is therefore $-15/40 = -5/8$. Marks are awarded only if both answers are correct.

2b. The batch TD estimates $V^\pi_{\text{TD-est}}(\cdot)$ must satisfy

$$V^\pi_{\text{TD-est}}(s_1) = R(s_1, \pi(s_1), s_2) + V^\pi_{\text{TD-est}}(s_2), \text{ and } V^\pi_{\text{TD-est}}(s_2) = R(s_2, \pi(s_2), s_3) + V^\pi_{\text{TD-est}}(s_3),$$

where $R(s_1, \pi(s_1), s_2)$ and $R(s_2, \pi(s_2), s_3)$ are empirical averages of the corresponding rewards. Since $R(s_1, \pi(s_1), s_2) = 1$, $R(s_2, \pi(s_2), s_3) = -3/8$, and $V^\pi_{\text{TD-est}}(s_3) = V^\pi(s_3) = 0$, we obtain $V^\pi_{\text{TD-est}}(s_2) = 5/8$ and $V^\pi_{\text{TD-est}}(s_2) = -3/8$. Marks are awarded only if both answers are correct.

3. The most natural use of RUN-PLANNER is to keep at hand a policy that is optimal with respect to the current estimate of the environmental dynamics and rewards. In order to get enough evidence for building this model, while also being able to eventually acting optimally, a GLIE policy must be followed. Here is a sketch of the algorithm to employ; several variants are possible.

1. Let $\hat{T}$ and $\hat{R}$ be running estimates of the true transition and reward functions. A uniform random policy may be followed until each possible state-action pair is visited at least once, and at which point $\hat{T}$ and $\hat{R}$ become well-defined. The following steps are repeated subsequently.

2. At each step t, let $\pi_t = \text{RUN-PLANNER}(S, A, \hat{R}, \hat{T}, \gamma)$. An action a_t is picked uniformly at random with probability $1/t$, and as $\pi_t(s_t)$ with probability $1 - 1/t$.

3. Reward r_t and next state s_{t+1} are obtained and used to update $\hat{T}$ and $\hat{R}$.

4. t is incremented, and we go to 2.

4a. Let the table given correspond to $Q_0(\cdot,\cdot)$. We get

$$Q_1(s_2,a_2) = Q_0(s_2,a_2)(1-\alpha) + \alpha(4 + \gamma Q_0(s_2,a_3)) = -1 \times 0.9 + 0.1 \times (4 + 0.5 \times 2) = 0.4.$$

If $s \neq s_2$ or $a \neq a_2$, $Q_1(s,a) = Q_0(s,a)$. From the second update, we get

$$Q_2(s_2,a_3) = Q_1(s_2,a_3)(1-\alpha) + \alpha(0 + \gamma Q_1(s_3,a_1)) = 2 \times 0.9 + 0.1 \times (0 + 0.5 \times 6) = 2.1,$$

and again, if $s \neq s_2$ or $a \neq a_3$, $Q_2(s,a) = Q_1(s,a)$. Thus, $Q_2(\cdot,\cdot)$ is as follows.

Q	a_1	a_2	a_3
s_1	5	6	9
s_2	0	-0.4	2.1
s_3	6	4	-3

4b. Our working is similar to 4a, but because we perform Sarsa updates, we have

$$Q_1(s_2,a_2) = Q_0(s_2,a_2)(1-\alpha) + \alpha(4 + \gamma Q_0(s_2,a_3)), \text{ and}$$

$$Q_2(s_2,a_3) = Q_1(s_2,a_3)(1-\alpha) + \alpha(0 + \gamma Q_1(s_3,a_2)).$$

The table for $Q_2(\cdot,\cdot)$ is as follows.

Q	a_1	a_2	a_3
s_1	5	6	9
s_2	0	-0.4	2
s_3	6	4	-3

5. Section 8.3.2 in the textbook by Sutton and Barto (1998) has a thorough description of tile coding. After defining the concept, the main configuration parameters that the answer must discuss are the tile width (which influences generalisation), the number of tilings and the offset (which influence the resolution), and also the resulting demands on computation and memory.

6a. Under the function approximation scheme provided, we approximate $V^\pi(s_1)$ by $2w_1 + w_2$, $V^\pi(s_2)$ by $w_1 + w_2$, and $V^\pi(s_3)$ by $-w_1 + w_2$. Under Monte Carlo (TD(1)) updating, (w_1, w_2) will converge to $(w_1^\star, w_2^\star)$, which minimises the the squared difference between the true and approximated values, weighted by the stationary probabilities.

$$(w_1^\star, w_2^\star) = \operatorname*{argmin}_{(w_1,w_2)\in\mathbb{R}^2} \frac{1}{2}(4 - 2w_1 - w_2)^2 + \frac{1}{4}(6 - w_1 - w_2)^2 + \frac{1}{4}(-5 + w_1 - w_2)^2 .$$

By setting first derivatives w.r.t. w_1 and w_2 to zero, we find that $w_1^\star = 3$ and $w_2^\star = -3/4$.

6b. TD(0) with linear function approximation also converges, but its fixed point need not be equal to $(w_1^\star, w_2^\star)$. However, there do exist bounds on the distance between the fixed point of TD(0) and $(w_1^\star, w_2^\star)$.

7a. Let A's strategy be to play a_1 with probability p, and a_2 with probability $1-p$. Let B's strategy be to play b_1 with probability q, and b_2 with probability $1-q$. Clearly A's expected reward is

$$R(A) = -R(B) = pq(1) + p(1-q)(-1) + (1-p)q(-2) + (1-p)(1-q)(0),$$

which may be rewritten as

$$R(A) = -R(B) = -p - 2q + 4pq = -p - 2q(1-2p) = p(-1+4q) - 2q.$$

If $p < 1/2$, B can set $q = 1$ to restrict $R(A)$ to $3p-2$, and if $p \geq 1/2$, B can set $q = 0$ to restrict A's reward to $-p$. Clearly the highest reward A can hope for, assuming B acts adversarially, is by setting $p = 1/2$, which yields $R(A) = -1/2$.

7b. A symmetric argument holds. If $q < 1/4$, A can set $p = 0$ to restrict $R(B)$ to $2q$, and if $q \geq 1/4$, A can set $p = 1$ to restrict $R(B)$ to $1 - 2q$. B must set $q = 1/4$ to obtain a minimax reward of $1/2$.

7c. Since $p = 0.1$, $R(B) = 0.1 + 1.6q$. B can maximise this reward by setting $q = 1$, which gives it a reward of 1.7.