CS 188
Spring 2019

Midterm Review RL Solutions

Q1. Markov Decision Processes

Consider a simple MDP with two states, S_1 and S_2, two actions, A and B, a discount factor γ of 1/2, reward function R given by

$$R(s, a, s') = \begin{cases} 1 & \text{if } s' = S_1; \\ -1 & \text{if } s' = S_2; \end{cases}$$

and a transition function specified by the following table.

s	a	s'	$T(s,a,s')$
S_1	A	S_1	1/2
S_1	A	S_2	1/2
S_1	B	S_1	2/3
S_1	B	S_2	1/3
S_2	A	S_1	1/2
S_2	A	S_2	1/2
S_2	B	S_1	1/3
S_2	B	S_2	2/3

(a) Perform a single iteration of value iteration, filling in the resultant Q-values and state values in the following tables. Use the specified initial value function V_0, rather than starting from all zero state values. Only compute the entries not labeled "skip".

s	a	$Q_1(s,a)$
S_1	A	1.25
S_1	B	1.50
S_2	A	skip
S_2	B	skip

s	$V_0(s)$	$V_1(s)$
S_1	2	1.50
S_2	3	skip

$$Q_1(s) = \sum_{s'} T(s,a,s')[R(s,a,s') + \gamma V_0(s')]$$

$$V_1(s) = \max_a Q_1(s,a)$$

(b) Suppose that Q-learning with a learning rate α of 1/2 is being run, and the following episode is observed.

s_1	a_1	r_1	s_2	a_2	r_2	s_3
S_1	A	1	S_1	A	-1	S_2

Using the initial Q-values Q_0, fill in the following table to indicate the resultant progression of Q-values.

s	a	$Q_0(s,a)$	$Q_1(s,a)$	$Q_2(s,a)$
S_1	A	$-1/2$	1/4	$-1/8$
S_1	B	0	(0)	(0)
S_2	A	-1	(-1)	(-1)
S_2	B	1	(1)	(1)

Here is the only update for the first observed tuple (s_1, a_1, r_1, s_2):

$$Q_1(s_1, a_1) = Q_0(s_1, a_1) + \alpha(r_1 + \gamma \max_a Q_0(s_2, a) - Q_0(s_1, a_1)$$

There is another observed tuple, so there is another update to get to Q_2.

(c) Given an arbitrary MDP with state set S, transition function $T(s, a, s')$, discount factor γ, and reward function $R(s, a, s')$, and given a constant $\beta > 0$, consider a modified MDP (S, T, γ, R') with reward function $R'(s, a, s') = \beta \cdot R(s, a, s')$. Prove that the modified MDP (S, T, γ, R') has the same set of optimal policies as the original MDP (S, T, γ, R).

$V^\pi_{\text{modified}} = \beta \cdot V^\pi_{\text{original}}$ satisfies the Bellman equation

$$\begin{aligned}
\beta \cdot V^\pi_{\text{original}}(s) &= V^\pi_{\text{modified}}(s) \\
&= \sum_{s'} T(s, \pi(s), s')[R'(s, \pi(s), s') + \gamma \cdot V^\pi_{\text{modified}}(s')] \\
&= \sum_{s'} T(s, \pi(s), s')[\beta \cdot R(s, \pi(s), s') + \gamma \cdot \beta \cdot V^\pi_{\text{original}}(s')] \\
&= \beta \cdot \sum_{s'} T(s, \pi(s), s')[R(s, \pi(s), s') + \gamma \cdot V^\pi_{\text{original}}(s')] \\
&= \beta \cdot V^\pi_{\text{original}}(s).
\end{aligned}$$

It follows that for any state s, the set of policies π that maximize V^π_{original} is precisely the same set of policies that maximize V^π_{modified}.

Intuitively, you should understand that scaling the reward function does not affect the $\arg\max$ that ultimately determines the policy.

(d) Although in this class we have defined MDPs as having a reward function $R(s, a, s')$ that can depend on the initial state s and the action a in addition to the destination state s', MDPs are sometimes defined as having a reward function $R(s')$ that depends only on the destination state s'. Given an arbitrary MDP with state set S, transition function $T(s, a, s')$, discount factor γ, and reward function $R(s, a, s')$ that *does depend* on the initial state s and the action a, define an *equivalent* MDP with state set S', transition function $T'(s, a, s')$, discount factor γ', and reward function $R'(s')$ that depends only on the destination state s'.

By *equivalent*, it is meant that there should be a one-to-one mapping between state-action sequences in the original MDP and state-action sequences in the modified MDP (with the same value). **You do not need to give a proof of the equivalence.**

States: $S' = S \times A \times S$, where A is the set of actions.

Transition function:

$$T'(s, a, s') = \begin{cases} T(s''', a, s'''') & \text{if } s = (s'', a', s''') \text{ and } s' = (s''', a, s''''); \\ 0 & \text{otherwise.} \end{cases}$$

Discount factor: $\gamma' = \gamma$

Reward function: $R'(s') = R(s, a, s'')$, where $s' = (s, a, s'')$.

Q2. Q-learning

Consider the following gridworld (rewards shown on left, state names shown on right).

Rewards

+10	+1

State names

A	B
G1	G2

From state A, the possible actions are right($\rightarrow$) and down($\downarrow$). From state B, the possible actions are left($\leftarrow$) and down($\downarrow$). For a numbered state (G1, G2), the only action is to exit. Upon exiting from a numbered square we collect the reward specified by the number on the square and enter the end-of-game absorbing state X. We also know that the discount factor $\gamma = 1$, and in this MDP all actions are **deterministic** and always succeed.

Consider the following episodes:

Episode 1 ($E1$)

s	a	s'	r
A	$\downarrow$	$G1$	0
$G1$	exit	X	10

Episode 2 ($E2$)

s	a	s'	r
B	$\downarrow$	$G2$	0
$G2$	exit	X	1

Episode 3 ($E3$)

s	a	s'	r
A	$\rightarrow$	B	0
B	$\downarrow$	$G2$	0
$G2$	exit	X	1

Episode 4 ($E4$)

s	a	s'	r
B	$\leftarrow$	A	0
A	$\downarrow$	$G1$	0
$G1$	exit	X	10

(a) Consider using temporal-difference learning to learn $V(s)$. When running TD-learning, all values are initialized to zero.
For which sequences of episodes, if repeated infinitely often, does $V(s)$ converge to $V^*(s)$ for all states s? (Assume appropriate learning rates such that all values converge.)
Write the correct sequence under "Other" if no correct sequences of episodes are listed.

☐ $E1, E2, E3, E4$ ☐ $E1, E2, E1, E2$ ☐ $E1, E2, E3, E1$ ■ $E4, E4, E4, E4$
☐ $E4, E3, E2, E1$ ☐ $E3, E4, E3, E4$ ☐ $E1, E2, E4, E1$
■ Other See explanation below

TD learning learns the value of the executed policy, which is $V^\pi(s)$. Therefore for $V^\pi(s)$ to converge to $V^*(s)$, it is necessary that the executing policy $\pi(s) = \pi^*(s)$.

Because there is no discounting since $\gamma = 1$, the optimal deterministic policy is $\pi^*(A) = \downarrow$ and $\pi^*(B) = \leftarrow$ ($\pi^*(G1)$ and $\pi^*(G2)$ are trivially exit because that is the only available action). Therefore episodes $E1$ and $E4$ act according to $\pi^*(s)$ while episodes $E2$ and $E3$ are sampled from a suboptimal policy.

From the above, TD learning using episode $E4$ (and optionally $E1$) will converge to $V^\pi(s) = V^*(s)$ for states A, B, $G1$. However, then we never visit $G2$, so $V(G2)$ will never converge. If we add either episode $E2$ or $E3$ to ensure that $V(G2)$ converges, then we are executing a suboptimal policy, which will then cause $V(B)$ to not converge. Therefore none of the listed sequences will learn a value function $V^\pi(s)$ that converges to $V^*(s)$ for all states s. An example of a correct sequence would be $E2, E4, E4, E4, ...$; sampling $E2$ first with the learning rate $\alpha = 1$ ensures $V^\pi(G2) = V^*(G2)$, and then executing $E4$ infinitely after ensures the values for states A, B, and $G1$ converge to the optimal values.

We also accepted the answer such that the value function $V(s)$ converges to $V^*(s)$ for states A and B (ignoring $G1$ and $G2$). TD learning using only episode $E4$ (and optionally $E1$) will converge to $V^\pi(s) = V^*(s)$ for states A and B, therefore the only correct listed option is $E4, E4, E4, E4$.

(b) Consider using Q-learning to learn $Q(s, a)$. When running Q-learning, all values are initialized to zero. For which sequences of episodes, if repeated infinitely often, does $Q(s, a)$ converge to $Q^*(s, a)$ for all state-action pairs (s, a)
(Assume appropriate learning rates such that all Q-values converge.)
Write the correct sequence under "Other" if no correct sequences of episodes are listed.

- ■ $E1, E2, E3, E4$
- ☐ $E1, E2, E1, E2$
- ☐ $E1, E2, E3, E1$
- ☐ $E4, E4, E4, E4$
- ■ $E4, E3, E2, E1$
- ■ $E3, E4, E3, E4$
- ☐ $E1, E2, E4, E1$
- ☐ Other ________

For $Q(s, a)$ to converge, we must visit all state action pairs for non-zero $Q^*(s, a)$ infinitely often. Therefore we must take the exit action in states $G1$ and $G2$, must take the down and right action in state A, and must take the left and down action in state B. Therefore the answers must include $E3$ and $E4$.