Homework 3

Jerry Duncan

September 24, 2020

1 Problem 1

Part a For part a, we need to solve Exercise 4.1 in the book. Exercise 4.1 asks us to solve for $q_\pi(11, \textbf{down})$ and $q_\pi(7, \textbf{down})$ using an equiprobable policy π and v_∞.

Remember that q is calculated as follows for undiscounted, episodic tasks:

$$q_\pi(s, a) = \sum_{s',a} p(s', r|s, a)[r + v_\pi(s')]$$

When calculating $q_\pi(11, \textbf{down})$, $s' = T$ because it we can only go to a terminal state, and receive $r = -1$. When calculating $q_\pi(7, \textbf{down})$, $s' = 11$ because we can only go to state 11, and receive $r = -1$. That means that $p(T, -1|11, \textbf{down}) = 1$ and $p(11, -1|7, \textbf{down}) = 1$. This gives us two equations to setup, plug in, and solve.

$$q_\pi(11, \textbf{down}) = \sum_{T,\textbf{down}} p(T, -1|11, \textbf{down})[-1 + v_\pi(T)]$$

$$q_\pi(11, \textbf{down}) = 1 \cdot [-1 + 0] = -1$$

$$q_\pi(7, \textbf{down}) = \sum_{T,\textbf{down}} p(T, -1|7, \textbf{down})[-1 + v_\pi(11)]$$

$$q_\pi(7, \textbf{down}) = 1 \cdot [-1 + -14] = -15$$

Therefore the answers are $q_\pi(11, \textbf{down}) = -1$ and $q_\pi(7, \textbf{down}) = -15$.

Part b For part b, we're asked to solve Exercise 4.2. Exercise 4.2 asks us to consider adding a new state, 15. We are asked to find $v_k(15)$ for the equiprobable random policy.

Remember that v is calculated as follows for undiscounted, episodic tasks:

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)[r + v_\pi(s')]$$

We are given the following p's.

$$\begin{gathered} p(12, -1|15, \textbf{left}) = 1 \\ p(13, -1|15, \textbf{up}) = 1 \\ p(14, -1|15, \textbf{right}) = 1 \\ p(15, -1|15, \textbf{down}) = 1 \end{gathered}$$

We also know that $\pi(\textbf{left}|15) = \pi(\textbf{up}|15) = \pi(\textbf{right}|15) = \pi(\textbf{down}|15) = 0.25$.

Then we setup our equation and calculate:

$$\begin{gathered} v_k(15) = 0.25 \cdot 1 \cdot [-1 + v_\pi(12)] + 0.25 \cdot 1 \cdot [-1 + v_\pi(13)] \\ +0.25 \cdot 1 \cdot [-1 + v_\pi(14)] + 0.25 \cdot 1 \cdot [-1 + v_\pi(15)] \\ v_k(15) = -1 + 0.25 \cdot v_k(12) + 0.25 \cdot v_k(13) + 0.25 \cdot v_k(14) + 0.25 \cdot v_k(15) \\ \frac{3}{4} v_k(15) = -1 + 0.25 \cdot (-22 + -20 + -14) \\ \frac{3}{4} v_k(15) = -15 \\ v_k(15) = \frac{1}{3} \cdot -60 \\ v_k(15) = -20 \end{gathered}$$

Part c For part c, we're asked to consider changing Policy Iteration to use an ϵ-soft policy instead of a deterministic, greedy one.

For step 3, we need to change the determination of π to no longer be the deterministic, but instead to be stochastic by assigning probabilities as so:

$$\pi(a|s) \leftarrow \begin{cases} 1-\epsilon+\frac{\epsilon}{|A(s)|} & \text{if a } = A* \\ \frac{\epsilon}{|A(s)|} & \text{if a } \neq A* \end{cases}$$

Where A* is the best action given a state. We also need to change the check for stopping to look at the action that has the most probability and see if it has changed.

For step 2, we need to change how we calculate V to take into account the fact that there is a chance of taking different actions. That change might add something like $\sum_{a \in A \neq a_{argmax}} \frac{\epsilon}{|A(s)|} \cdot \ldots$ for all actions except the best, and $\sum_{a \in A \neq a_{argmax}} (1-\epsilon+\frac{\epsilon}{|A(s)|}) \cdot \ldots$ for the best action where $\ldots$ refers to the current V equation, $\sum_{s',r} p(s',r|s,\pi(s))[r+\gamma V(s')]$.

For step 1, we just need to change the initial π to randomly pick a best action for each state and assign its probability to $1-\epsilon+\frac{\epsilon}{|A(s)|}$ and assign the other's a probability of $\frac{\epsilon}{|A(s)|}$.

For each of the steps, the main gist is changing them to allow for probabilistic policies, which they currently do not. In short, for step 1 we initialize a probabilistic policy, for step 2 we change how we calculate V by weighting them by the probability we take that action, and for step 3 we change how we calculate π to allow for $\frac{\epsilon}{|A(s)|}$) to be assigned to all actions that aren't the best, and the rest to be assigned to the best action.

2 Problem 2

Part a For part a, we want to calculate $V_1(A)$ and $V_1(B)$. Remember the formula to calculate V_{k+1}.

$$V_{k+1} = \underset{a}{max} \sum_{s',r} p(s',r|s,a)[r+\gamma V_k(s')]$$

We're given the following: $\gamma = 0.5$, $V_0(A) = 2$, and $V_0(B) = -2$.

$$V_1(A) = max \begin{cases} a_1 & 0.5 \cdot [2+0.5\cdot 2] + 0.5 \cdot [2+0.5\cdot -2] \\ a_2 & 0.8 \cdot [9+0.5\cdot 2] + 0.2 \cdot [2+0.5\cdot -2] \end{cases}$$

$$V_1(A) = max \begin{cases} a_1 & 0.5\cdot 3 + 0.5 \cdot 1 = 1.5 + 0.5 = 2 \\ a_2 & 0.8 \cdot 10 + 0.2 \cdot 1 = 8 + 0.2 = 8.2 \end{cases}$$

$$V_1(B) = max \begin{cases} a_1 & 0.7 \cdot [-1 + 0.5 \cdot 2] + 0.3 \cdot [-1 + 0.5 \cdot -2] \\ a_2 & 0.8 \cdot [-1 + 0.5 \cdot 2] + 0.2 \cdot [-1 + 0.5 \cdot -2] \end{cases}$$

$$V_1(B) = max \begin{cases} a_1 & 0.7 \cdot 0 + 0.3 \cdot -2 = 0 + -0.6 = -0.6 \\ a_2 & 0.8 \cdot 0 + 0.2 \cdot -2 = 0 + -0.4 = -0.4 \end{cases}$$

So $V_1(A) = 8.2$ and $V_1(B) = -0.4$.

Part b For part b we need to calculate the policy we select at A, given the above results.

Remember how we calculate π:

$$\pi_1(a|s) = \underset{a}{argmax} \sum_{s',r} p(s', r|s, a)[r + \gamma V_1(s')]$$

$$\pi_1(A) = \underset{a}{argmax} \begin{cases} a_1 & 0.5 \cdot [2 + 0.5 \cdot 8.2] + 0.5 \cdot [2 + 0.5 \cdot -0.4] \\ a_2 & 0.8 \cdot [9 + 0.5 \cdot 8.2] + 0.2 \cdot [2 + 0.5 \cdot -0.4] \end{cases}$$

$$\pi_1(A) = \underset{a}{argmax} \begin{cases} a_1 & 0.5 \cdot 6.1 + 0.5 \cdot 1.8 = 3.05 + 0.9 = 3.95 \\ a_2 & 0.8 \cdot 13.1 + 0.2 \cdot 1.8 = 10.48 + 0.36 = 10.84 \end{cases}$$

From these values, we can determine that $\pi_1(A) = a_2$.

It is not necessarily true that this is the optimal policy. In order to determine the optimal policy, V must converge as we go through the value iteration loop — that is, $|V_k - V_{k+1}| < \theta$. In our case, $V_0(A) = 2$ and $V_1(A) = 8.2$, the difference between them being 6.2 is a very large change and indicates that it is unlikely that we have converged on the correct $V(A)$. Because of this, the policy $\pi_1(A)$ we have deduced from value iteration step is not guaranteed to be the optimal policy, only better or equivalent to the previous policy. Because of that fact, we cannot say that this policy is necessarily the optimal policy.

Part c For part c we need to calculate $V_1(A)$ and $V_1(B)$ using in-place updates. That means that only one of them will change from a. For the purposes of answering this question, I am calculating $V_1(A)$ first, meaning it won't change from **part a**.

So we need to recalculate $V_1(B)$ using $V_1(A)$ instead of $V_0(A)$.

$$V_1(B) = max \begin{cases} a_1 & 0.7 \cdot [-1 + 0.5 \cdot 8.2] + 0.3 \cdot [-1 + 0.5 \cdot -2] \\ a_2 & 0.8 \cdot [-1 + 0.5 \cdot 8.2] + 0.2 \cdot [-1 + 0.5 \cdot -2] \end{cases}$$

$$V_1(B) = max \begin{cases} a_1 & 0.7 \cdot 3.1 + 0.3 \cdot -2 = 2.17 + -0.6 = 1.57 \\ a_2 & 0.8 \cdot 3.1 + 0.2 \cdot -2 = 2.48 + -0.4 = 2.08 \end{cases}$$

Using in-place updates, updating $V_1(A)$ first, $V_1(B) = 2.08$.