Exercise Set 1

6.0 VU AKNUM Reinforcement Learning

April 23, 2020

Exercise (2.1) The probability is $\epsilon * 0.5 + (1 - \epsilon) * 1 = 0.75$, because the greedy action is taken with certainty in $1 - \epsilon$ times of the cases, and in case of explorative actions ϵ, it is taken with with a uniform probability of 50%.

Exercise (2.2) The ϵ case definitely occurred in $t = 4$, since $Q_4(2) = -0.5$ and $t = 5$ since $Q_5(2) > Q_5(3)$. It might have occurred in $t = 1$ since there is no clear optimal action initially and in $t = 2$, since the only non-empty entry in the value table is negative. It might also be the case that $t = 3$ was exploritative, since it always holds that $\epsilon > 0$.

Exercise (2.3) As $t \to \infty$, the 0.01-greedy method will be the best performing method, since it will end up choosing the optimal action in at least 99.1% of the cases and thus end up with a higher average reward than the 0.1-greedy method, which will choose the the optimal action in 91% of the cases.

Exercise (2.4) Since α_n is now not constant, the product of the $(1 - \alpha)$ terms will be different. Since we now cannot simply collapse $\prod_{i=1}^{n}(1-\alpha) = (1-\alpha)^n$ anymore, equation 2.6 changes into:

$$Q_{n+1} = Q_1 \prod_{i=1}^{n}(1 - \alpha_i) + \sum_{i=1}^{n} R_i \alpha_i \prod_{j=1}^{n-i}(1 - \alpha_j) \tag{1}$$

Thus, the weighting for reward R_n would be $\alpha_n \prod_{j=1}^{n}(1 - \alpha_j)$

Exercise (2.6) Since the estimates are optimistic, $R_t < Q_t(.) \forall t \in [1, k]$. Thus, each action is explored for the first k steps, giving a 10% probability of selecting the optimal action in this case. Then, at $t = 11$, the best performing action of the first full exploration will be chosen, which has a high probability of being the optimal action. Still, as $R_1 1 < Q_1 1(a^\star)$ (assuming sensible initial optimistic estimates and α), $a^\star$ will certainly not be chosen at $t = 12$. The second best performing action of the first exploration will be chosen. This will continue until $t = k * 2$. At this step, again, all optimistic estimations were reduced and $a^\star$ might be chosen again with a high probability. This is the spike at $t = 21$. But, since the optimism has by now been severely reduced, the osciliation will be less pronounced.

Exercise (3.2) There are obviously exceptions which the vanilla MDP framework can not represent. If a task violates the Markov property, in theory the MDP framework no longer applies. However, in practice, the MDP framework might still be able yield passable results. There exists the partially-observable MDP framework which aims to represent tasks violating the Markov property.

Exercise (3.4) As in the original table, one simply generates all combinations and discards all impossible transitions (where $p = 0$).

s	a	s'	r	$p(s', r\|s, a)$
high	search	high	$r_{\texttt{search}}$	α
high	search	low	$r_{\texttt{search}}$	$1 - \alpha$
low	search	high	-3	$1 - \beta$
low	search	low	$r_{\texttt{search}}$	β
high	wait	high	$r_{\texttt{wait}}$	1
low	wait	low	$r_{\texttt{wait}}$	1
low	recharge	high	0	1

Exercise (3.6) The discounted return for continuing tasks is:

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2}$$

When adapted for episodic tasks, the return is:

$$G_t = \sum_{k=0}^{T-t-1} \gamma^k R_{t+k+1} \tag{3}$$

In the specific case, we have $R_t = 0 \forall t \in [0, T)$ and $R_T = -1$. Thus, the return would be $\gamma^{T-t-1} R_T = -\gamma^{T-t-1}$.

Exercise (3.7) Since the agent receives very sparse rewards it will probably be stuck in exploring for a large number of steps. Once it reaches the exit, it recieves only a positive reward. Without discounting, the return will be the same for $T_1 \gg T_2$. Thus, there is no incentive to exit the maze quickly. This could be alleviated by introducing discounting.

It might be better to provide a clear signal to leave the maze, i.e. provide a reward of -1 for each step in which the agent did not exit the maze.

Exercise (3.8) The returns can be calculated using Equation 3, where $T = 5$. We thus set $G_5 = 0$. Then:

$G_5 = 0$
$G_4 = R_5 + \gamma G_5 = 2$
$G_3 = R_4 + \gamma G_4 = 4$
$G_2 = R_3 + \gamma G_3 = 8$
$G_1 = R_2 + \gamma G_2 = 6$
$G_0 = R_1 + \gamma G_1 = 2$

Exercise (3.9) Using equations 3.9 and 3.10, one can formulate:
$G_0 = R_1 + \gamma G_1 = R_1 + \gamma \sum_{k=0}^{\infty} R\gamma^k = R_1 + \gamma \sum_{k=0}^{\infty} \frac{R}{1-\gamma}$
With $\gamma = 0.9, R = 7$, this results in $G_0 = 65$ and $G_1 = 70$.

Exercise (3.11) Following the stochastic policy $\pi(a|s)$, one gets:
$\mathbb{E}[R_{t+1}|S_t = s] = \sum_{a\in\mathcal{A}} \pi(a|s) \sum_{s\in\mathcal{S},r\in\mathcal{R}} p(s', r|s, a)r$

Exercise (3.14) Assume $v_\pi(s) = 0.7, v_\pi(s'_1) = 2.3, v_\pi(s'_2) = 0.4, v_\pi(s'_3) = -0.4, v_\pi(s'_4) = 0.7$ and $\gamma = 0.9$. The reward of all successor states r is 0.
Then, using equation 3.14: $\sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)[r + \gamma v_\pi(s')] =$
$\frac{\gamma v_\pi(s'_1)}{4} + \frac{\gamma v_\pi(s'_2)}{4} + \frac{\gamma v_\pi(s'_3)}{4} + \frac{\gamma v_\pi(s'_4)}{4} \approx 0.7$

Exercise (3.15) Let $c \in \mathbb{R}$ be a constant and let $R' = R+c$ denote the updated rewards. Then by definition, the value function using the updated rewards is:
$v_\pi(s) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R'_{t+k+1} \;\middle|\; S_t = s\right] =$
$\mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k (R_{t+k+1} + c) \;\middle|\; S_t = s\right] =$
$\mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} + \gamma^k c \;\middle|\; S_t = s\right]$.
We can denote $v_c = \gamma^k c$. It is clear that v_c is independent of the state and policy. Therefore, the relative values are not affected. □

Exercise (3.16) Here the case is different, since there is a terminal reward. If we consider the maze running case, where one might introduce negative rewards to encourage completing the maze. If c is set s.t. the rewards become positive, this effect is severely reduced and the incentive for the agent changes to staying in the maze as long as possible.

Exercise (3.17) The Bellman equation for action values is similar to the Bellman equation for state values. Equation 3.14 needs to be adapted s.t. we sum over the policy actions one step into the future:
$q_\pi(s, a) = \sum_{s',r} p(s', r|s, a)[r + \gamma \sum_{a'} q_\pi(s', a')]$

Exercise (3.18) The state value function can be specified depending on the action value function using an expectation:
$v_\pi(s) = \mathbb{E}_\pi[q_\pi(s, a) \mid S_t = s]$
Then, eliminating the expectation is easy using the policy π:
$v_\pi(s) = \sum_a \pi(a|s) q_\pi(s, a)$
This makes it appearant that the state value function is simply the weighted sum of all possible action value functions given a state.

Exercise (3.19) Defining the action value function using the state value function is similar to the inverse case. One has to take into account that the state value function is now concerned with the succeeding state and thus has to be discounted.
$q_\pi(s, a) = \mathbb{E}[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S_t = s, A_t = a]$
The expectation can be eliminated using the transition dynamics:
$q_\pi(s, a) = \sum_{s',r} p(s', r|s, a)[r + \gamma v_\pi(s')]$
It can be seen that the action value function is just a weighted sum over the rewards and values of the possible next states given the current state and action.

Exercise (3.22) In this case, one can simply calculate the value functions or interpret γ.

Since $\gamma = 0$ represents the myopic case, π_{left} would be optimal.

For $\gamma = 0.9$, the reward of the second decision will be nearly weighted the same as the first decision. Since $v_{\pi_{\text{right}}} = 1.8 > v_{\pi_{\texttt{left}}} = 1$, π_{right} is optimal.

In the case $\gamma = 0.5$, $v_{\pi_{\text{right}}} = 1 = v_{\pi_{\texttt{left}}} = 1$, thus both policies are optimal.

Exercise (3.24) The optimal value function is given by equation 3.19:

$v_*(A) = \max_a \mathbb{E}_{\pi_*}[G_t \mid S_t = A, A_t = a] = \max_a \sum_{s',r} p(s', r|s, a)[r + \gamma v_*(s')]$

The optimal policy assigns the maximum probability to $a = \texttt{up}$ and the transition dynamic $p(A', 10|A, \texttt{up}) = 1$ and 0 otherwise. Now, for state A', the optimal policy would assign the highest probability to `up` for four consecutive times, until state A is reached. The transition dynamics are 1 for $r = 0$ and 0 otherwise. For this infite series we have a constant reward of 10 if $t \mod 5 = 0$ and 0 otherwise. Using equation 3.10, we end up with:

$G_t = \sum_{k=0}^{\infty} \gamma^{5k} 10 = \frac{10}{1-\gamma^5} = 24.419$