Reinforcement Learning: Chapter 2 Exercises

Name: Eli Andrew

(a) **Exercise 2.1** In ϵ-greedy action selection, for the case of two actions and $\epsilon = 0.5$, what is the probability that the greedy action is selected.

- There is a $1 - \epsilon$ probability that the greedy action is selected which is 0.5.

(b) **Exercise 2.2: Bandit example** Consider a k-armed bandit problem with $k = 4$ actions, denoted 1, 2, 3, and 4. Consider applying to this problem a bandit algorithm using ϵ-greedy action selection, sample-average action-value estimates, and initial estimates of $Q_1(a) = 0$, for all a. Suppose the initial sequence of actions and rewards is $A_1 = 1$, $R_1 = -1$, $A_2 = 2$, $R_2 = 1$, $A_3 = 2$, $R_3 = -2$, $A_4 = 2$, $R_4 = 2$, $A_5 = 3$, $R_5 = 0$. On some of these time steps the ϵ case may have occurred, causing an action to be selected at random. On which time steps did this definitely occur? On which time steps could this possibly have occurred?

- The ϵ case may have occurred on time steps 1, 2, and 3 since there were multiple "greedy" actions that could have been selected. The ϵ case definitely occurred on time step 4 because there was a single "greedy" action to select and it was not taken.

(c) **Exercise 2.3** In the comparison shown in Figure 2.2, which method will perform best in the long run in terms of cumulative reward and probability of selecting the best action? How much better will it be? Express you answer quantatatively.

- In the long run I would expect the $\epsilon = 0.01$ method to perfrom best in terms of cumulative reward and probability of selecting the best action since it will eventually explore enough to know the best moves and then will exploit them more often than the $\epsilon = 0.1$ method.

(d) **Exercise 2.4** If the step-size parameters, α_n, are not constant, then the estimate Q_n is a weighted average of previously received rewards with a weighting different from that given by (2.6). What is the weighting on each prior reward for the general case, analogous to (2.6), in terms of the sequence of step-size parameters.

-

(e) **Exercise 2.5 (programming)** Skipped.

(f) **Exercise 2.6: Mysterious Spikes** The results shown in Figure 2.3 should be quite reliable because they are averages over 2000 individual, randomly chose 10-armed bandit tasks. Why, then, are there oscillations and spikes in the early part of the curve for the optimistic method? In other words, what might make this method perform particularly better or worse, on average, on particular early steps?

- The method would perform better or worse on average on certain early steps because it is exploring the whole state space and will therefore swing between the high and low rewards present in the space.

(g) **Exericse 2.7: Unbiased Constant-Step-Size Trick** In most of this chapter we have used sample averages to estimate action values because sample averages do not produce the initial bias that constant step sizes do. However, sample averages are not a completely satisfactory solution because they may perform poorly on nonstationary problems. It is possible to avoid the bias of constant step sizes while retaining their advatages on nonstationary problems? One way is to use a step size of $\beta_n = \frac{\alpha}{\bar{o}_n}$ to process the nth reward for a particular action, where $alpha > 0$ is a conventional constant step size, and $\bar{o}_n$ is a trace of one that starts at 0: $\bar{o}_n = \bar{o}_{n-1} + \alpha(1 - \bar{o}_{n-1})$, for $n \geq 0$, with $\bar{o}_0 = 0$. Carry out an analysis like that is (2.6) to show that Q_n is an exponential recency-weighted average *without initial bias*.

(h) **Exercise 2.8: UCB Spikes** In Figure 2.4 the UCB algorithm shows a ditinct spike in performance on the 11th step. Why is this? Note that for your answer to be fully satisfactory it must explain both why the reward increases on the 11th step and why it decreases on subsequent steps. Hint: If $c = 1$, then the spike is less prominent.

- The spike on the 11th step is due to the fact that $N_t(a)$ should equal 1 for all a since we would have explored all 10 actions at this step. So, on step 11 the average reward is expected to be very high since we have an understanding of the space, however, after step 11 we lower our expectations because we try the max action again and receive a lower than expected reward.

(i) **Exercise 2.10** Suppose you face a 2-armed bandit task whose true action values change randomly from time step to time step.

(j) Specifically, suppose that, for any time step, the true values of action 1 and 2 are respectively 0.1 and 0.2 with probability 0.5 (case A), and 0.9 and and 0.8 with probability 0.5 (case B). If you are not able to tell which case you face at any step, what is the best expectation of success you can achieve and how should you behave to achieve it? Now suppose that on each step you are told whether you are facing case A or case B (although you still don't know the true action values). This is an associative search task. What is the best expectation of success you can achieve in this task, and how should you behave to achieve it?

- The expectation of each case is given as: $E[A] = 0.5(0.1) + 0.5(0.2) = 0.15$ and $E[B] = 0.5(0.9) + 0.5(0.8) = 0.85$. The best we can do here is therefore $0.5(0.15)+0.5(0.85) = 0.5$. If we knew the optimal arm to select then the expected value would incrase to $0.5(0.2) + 0.5(0.9) = 0.55$.

(k) **Exercise 2.11 (programming)** Skipped.