Answers to Exercises
Reinforcement Learning: Chapter 2

Exercise 2.1 In ε-greedy action selection, for the case of two actions and $\varepsilon = 0.5$, what is the probability that the greedy action is selected?

Answer: 0.75. There is a 0.5 chance of selecting the greedy action directly, plus a 0.25 chance of selecting it as one of the two actions when a random action is selected. □

Exercise 2.2: *Bandit example* This exercise appeared in the first printing slightly differently than intended. However, it is still a valid exercise, and below we give the answer first for the excercise as intended and then as it appeared in the first printing.

Exercise 2.2: *Bandit example as intended* 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?

Answer: The ε case definitely came up on steps 4 and 5, and could have come up on any of the steps.

To see this clearly, make a table with the estimates, the set of greedy of actions, and the data at each step:

t	Q_1	Q_2	Q_3	Q_4	$\{A_t^*\}$	A_t	ε-case?	R_t
1	0	0	0	0	$\{1,2,3,4\}$	1	maybe	-1
2	$\mathbf{-1}$	0	0	0	$\{2,3,4\}$	2	maybe	1
3	-1	$\mathbf{1}$	0	0	$\{2\}$	2	maybe	-2
4	-1	$\mathbf{-0.5}$	0	0	$\{3,4\}$	2	yes	2
5	-1	$\mathbf{0.3333}$	0	0	$\{2\}$	3	yes	0

The estimate that changed on each step is bolded. If the action taken is not in the greedy set, as on time steps 4 and 5, then the ε case must have come up. On steps 1–3, the greedy action was taken, but still it is possible that the ε case come up and the greedy action was taken by chance. Thus, the answer to the second question is *all the time steps.* □

Exercise 2.2: *Bandit example as first printed* 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?

Answer: The ε case definitely came up on steps 2 and 5, and could have come up on any of the steps.

To see this clearly, make a table with the estimates, the set of greedy of actions, and the data at each step:

t	Q_1	Q_2	Q_3	Q_4	$\{A^*_t\}$	A_t	ε-case?	R_t
1	0	0	0	0	$\{1,2,3,4\}$	1	maybe	1
2	**1**	0	0	0	$\{1\}$	2	yes	1
3	1	**1**	0	0	$\{1,2\}$	2	maybe	2
4	1	**1.5**	0	0	$\{2\}$	2	maybe	2
5	1	**1.6666**	0	0	$\{2\}$	3	yes	0

The estimate that changed on each step is bolded. If the action taken is not in the greedy set, as on time steps 2 and 5, then the ε case must have come up. On step 4, the sole greedy action was taken, but it is possible that the ε case come up and the greedy action was taken by chance. Thus, the answer to the second question is *all the time steps.* □

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 your answer quantitatively.

Answer: The cumulative performance measures are essentially the areas under the curves in Figure 2.2. In the near term, the $\varepsilon = 0.1$ method has a larger area, as shown in the figure. But in the long run, the $\varepsilon = 0.01$ method will reach and sustain a higher level. For example, it will eventually learn to select the best action 99.1% of the time, whereas the $\varepsilon = 0.1$ method will never select the best action more than 91% of the time, for a difference of 8.1%. Thus, in the long run the area under the $\varepsilon = 0.01$ curve will be greater. The greedy method will remain at essentially the same low level shown in the figure for a very long time.

In the long run, the $\varepsilon = 0.01$ method will be greater than the $\varepsilon = 0.1$ method by 8.1% of the difference between the value of the best action and the value of an average action. □

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?

Answer: As in the book, let us focus on the sequence of rewards, action values, and step-sizes corresponding to one particular action. Let us denote these by R_n, Q_n, and α_n. This greatly simplifies the notation. Then we can do a derivation similar to the one in (2.6):

$$\begin{aligned}
Q_{n+1} &= Q_n + \alpha_n\Big[R_n - Q_n\Big] \\
&= \alpha_n R_n + (1-\alpha_n)Q_n \\
&= \alpha_n R_n + (1-\alpha_n)\left[\alpha_{n-1}R_{n-1} + (1-\alpha_{n-1})Q_{n-1}\right] \\
&= \alpha_n R_n + (1-\alpha_n)\alpha_{n-1}R_{n-1} + (1-\alpha_n)(1-\alpha_{n-1})Q_{n-1} \\
&= \alpha_n R_n + (1-\alpha_n)\alpha_n R_{n-1} + (1-\alpha_n)(1-\alpha_{n-1})\alpha_{n-2}R_{n-2} + \\
&\qquad\qquad \cdots + (1-\alpha_n)(1-\alpha_{n-1})\cdots(1-\alpha_2)\alpha_1 R_1 \\
&\qquad + \ (1-\alpha_n)(1-\alpha_{n-1})\cdots(1-\alpha_1)Q_1 \\
&= \prod_{i=1}^{n}(1-\alpha_i)Q_1 + \sum_{n=1}^{n}\alpha_n R_n \prod_{i=n+1}^{n}(1-\alpha_i).
\end{aligned}$$

In other words, the weighting on R_n in Q_n is $\alpha_n \prod_{i=n+1}^{n}(1-\alpha_i)$. □

Exercise 2.6: *Mysterious Spikes* The results shown in Figure 2.3 should be quite reliable because they are averages over 2000 individual, randomly chosen 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?

Answer: Because the initial values are optimistic, whatever actions are selected on the very first plays will disappoint, and will not be reselected until all the other actions have been tried at least once. Thus, the first 10 plays will just be a sweep through the 10 actions in some random order. The % optimal action on these 10 plays will thus be at the 10% chance level. The first opportunity to make a better than chance action selection will be on the 11th play. This is the first spike in the graph. This is where the action that did best in the previous 10 plays is selected again. This action naturally has a greater than chance probability of being the optimal action. However, even if it is optimal, it still disappoints because its action value estimate is still optimistic. Remember, these action-value estimates are computed with a constant step-size parameter and shift only gradually from their initial values to the level of the observed rewards. Thus, although the action on the 11th play was the one that did best on the first 10 plays, on the 11th play its value will be pushed *down*, closer to its real value rather its optimistic value. On the 12th play the action that did second best in the first 10 plays will be selected, and on the 13th play the action that did third best will be selected. And so on. By trial 21 we can expect a second spike, but it will be less well defined because the values are starting to become more accurate and less optimistic. □

Exercise 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 (see the analysis leading to (2.6)). However, sample averages are not a completely satisfactory solution because they may perform poorly on nonstationary problems. Is it possible to avoid the bias of constant step sizes while retaining their advantages on nonstationary problems? One way is to use a step size of

$$\beta_n \doteq \alpha/\bar{o}_n, \tag{2.1}$$

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:

$$\bar{o}_n \doteq \bar{o}_{n-1} + \alpha(1 - \bar{o}_{n-1}), \quad \text{for } n \geq 0, \quad \text{with } \bar{o}_0 \doteq 0. \tag{2.2}$$

Carry out an analysis like that in (2.6) to show that Q_n is an exponential recency-weighted average *without initial bias.*

Answer: TBD □

Exercise 2.8: *UCB Spikes* In Figure 2.4 the UCB algorithm shows a distinct 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 the subsequent steps. Hint: if $c = 1$, then the spike is less prominent.

Answer: There are two things to explain here: why the jump up at step 11, and why the smaller jump down on step 12. Together these give us the spike. The jump up at step 11 is due to that being the first step on which the action selected depends on the received rewards—the first ten actions being random without replacement. The action taken on step 11 will be that which got the greatest reward on its single first pick. From the graph it is clear that this is not necessarily the best action (the average reward eventually exceeds this height) but it is far better than a random action. This is why performance jumps up at step 11.

At step 12, the second part of the UCB equation (2.10) kicks in (because now the actions vary in the number of times they have been tried). The one action that did well in the first 10 steps, and was repeated at step 11 will now be at a disadvantage because it has been selected twice while the others only once. If c is large, then this effect dominates and the action that performed best in the first 10 steps is ruled out on step 12. In this case the action selected on step 12 is the action that performed second best in the first 10 steps. On average, this action is less good than the one that performed best in the first 10 steps, so there is a smaller jump down on step 12, giving us the complete spike.

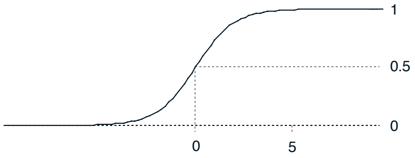
If is smaller (e.g., $c = 1$), then the jump up at step 11 is just the same. However, the jump down at step 12 is smaller because the advantage that comes from having been selected few times is smaller. It is now possible that the action that performed best in steps 1-10, and which was selected again in step 11, will be repeated in step 12. Thus, there is a smaller jump down, although long-term performance may be worse because of the reduced exploration. □

Exercise 2.9 Show that in the case of two actions, the soft-max distribution is the same as that given by the logistic, or sigmoid, function often used in statistics and artificial neural networks.

Answer: The logistic function is

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

where, x is the total evidence (e.g., a weighted sum) in favor of a positive decision, and $\sigma(x)$ is the probability of a positive decision (or, e.g., the output of an artificial neuron). The logistic function looks like this:



In the case of two actions, with two estimated action values, call then $Q(a)$ and $Q(b)$, the soft-max probability of selecting a is

$$\frac{e^{Q(a)}}{e^{Q(a)}+e^{Q(b)}} = \frac{1}{1+\frac{e^{Q(b)}}{e^{Q(a)}}} = \frac{1}{1+e^{(Q(b)-Q(a))}} = \sigma(Q(a)-Q(b)).$$

Thus, the Gibbs distribution is making a stochastic decision equivalent in the case of two actions to the logistic in the difference of the action values. Only the difference in the values values matters. □

Exercise 2.10 Suppose you face a 2-armed bandit task whose true action values change randomly from time step to time step. Specifically, suppose that, for any time step, the true values of actions 1 and 2 are respectively 0.1 and 0.2 with probability 0.5 (case A), and 0.9 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?

Answer: If you can't tell which bandit problem you are facing, then action 1 results in expected reward of $0.5 \cdot 0.1 + 0.5 \cdot 0.9 = 0.5$ and action 2 results in expected reward of $0.5 \cdot 0.2 + 0.5 \cdot 0.8 = 0.5$. Thus, you would receive an expected reward of 0.5 no matter how you select actions.

If you are told which of the two cases you are facing on each play, then you can learn the right action for each case: action 2 in case A, and action 1 in case B. If you behaved in this way, then you would obtain an expected reward of $0.5 \cdot 0.2 + 0.5 \cdot 0.9 = 0.55$. □