Solutions to the second edition of *Reinforcement Learning: An Introduction*

Farshid Asadi [1]

April 29, 2025

[1]This document is accompanied by an online GitHub repository, available at: github.com/farshidasadi47/reinforcement_learning_an_introduction_exercises

Chapter 1

Exercise 1.1. Since the agent is playing against itself, it learns to play against a good player. As a result, the policy will be different than the policy obtained by playing against random players. Such a policy may lead to more stalemate conditions. □

Exercise 1.2. The symmetric positions can be described by the same state, therefore reducing the number of states of the problem. Due to the reduced number of states, it is expected that the agent learns faster.

If other agent does not take advantage of the existing symmetry and treat them differently, the positions will have different values, so we should also treat them differently. □

Exercise 1.3. It will most likely learn worse. The reason is that in greedy learning there could be states, with possibly better values, that the agent never faces. The agent might get stuck in some locally optimal policy. □

Exercise 1.4. If learning from exploratory moves, we learn probabilities of a policy that sometimes chooses randomly, depending on the exploration rate. On the other hand, when not learning from exploratory moves, we learn probabilities related to optimal policy. I think the second approach would result in more wins since the learned values are learned via optimal policy. □

Exercise 1.5. We can learn the other player's policy and use it to predict its next moves when making decision. □

Chapter 2

Exercise 2.1. We have

$$\begin{aligned}\Pr[\text{greedy}] &= \Pr[\text{greedy}|\text{exploit}]\Pr[\text{exploit}] + \Pr[\text{greedy}|\text{explore}]\Pr[\text{explore}] \\ &= 1.0 \times 0.5 + 0.5 \times 0.5 = 0.75\end{aligned} \tag{2.1}$$

Exercise 2.2. We have the following action value estimation after each step. According to table 2.1, the ϵ case definitely happened in $A_4 = 2$ and $A_5 = 3$,

Value estimation \ Step	0	1	2	3	4	5
$Q_t(1)$	0	-1	-1	-1	-1	-1
$Q_t(2)$	0	0	1	$-1/2$	$1/3$	$1/3$
$Q_t(3)$	0	0	0	0	0	0
$Q_t(4)$	0	0	0	0	0	0

Table 2.1: Value estimations after each step.

since optimal action was not chosen. The ϵ case could have happened in $A_1 = 1$, $A_2 = 2$, and $A_3 = 2$. □

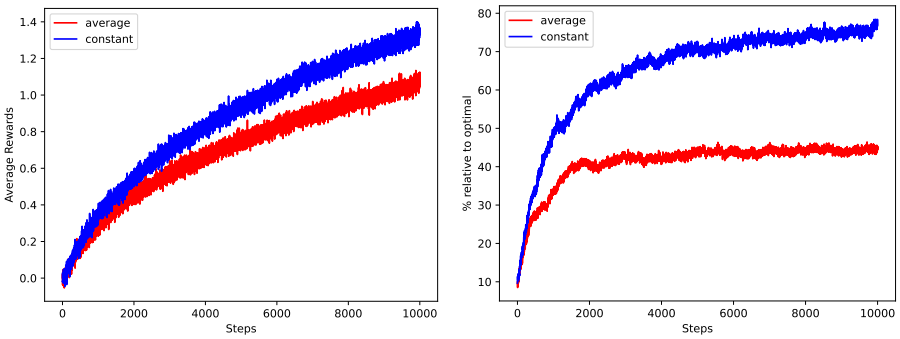
Exercise 2.3. In general we have

$$\begin{aligned}\Pr[\text{optimal}] &= \Pr[\text{optimal}|\text{exploit}]\Pr[\text{exploit}] + \Pr[\text{optimal}|\text{explore}]\Pr[\text{explore}] \\ &= \Pr[\text{optimal}|\text{exploit}](1-\epsilon) + \Pr[\text{optimal}|\text{explore}]\,\epsilon\end{aligned} \tag{2.2}$$

For greedy policy we have $\epsilon = 0$ and $\Pr[\text{optimal}|\text{exploit}] \approx 0.33$ (from figure 2.2 of the book), so we can write,

$$\Pr[\text{optimal}|\epsilon = 0] = 0.33 \times 1 = 0.33 \tag{2.3}$$

In ϵ greedy cases, the action value estimations will eventually converge to their true means, thus we can assume $\Pr[\text{optimal}|\text{exploit}, \epsilon > 0] = 1$ in the long run.



(a) Average reward (b) Reward relative to optimal

Figure 2.1: Performance of the simple bandit algorithm using averaging compared to a constant step size, exercise 2.5

Further, since there are 10 arms we have $\Pr[\text{optimal}|\text{explore}] = 0.1$. Using these, we can write,

$$\Pr[\text{optimal}|\epsilon = 0.01] = 1 \times 0.99 + 0.1 \times 0.01 = 0.991 \tag{2.4}$$

and

$$\Pr[\text{optimal}|\epsilon = 0.1] = 1 \times 0.9 + 0.1 \times 0.1 = 0.91 \tag{2.5}$$

Therefore ϵ greedy algorithm with $\epsilon = 0.01$ performs better than other two methods by at least 0.081 more probability of choosing optimal action. □

Exercise 2.4. We can write,

$$\begin{aligned}
Q_{n+1} &= Q_n + \alpha_n [R_n - Q_n] \\
&= \alpha_n R_n + (1-\alpha_n) Q_n \\
&= \alpha_n R_n + (1-\alpha_n) [\alpha_{n-1} R_{n-1} + (1-\alpha_{n-1}) Q_{n-1}] \\
&= \alpha_n R_n + (1-\alpha_n) \alpha_{n-1} R_{n-1} + (1-\alpha_n)(1-\alpha_{n-1}) Q_{n-1} \\
&= \alpha_n R_n + (1-\alpha_n) \alpha_{n-1} R_{n-1} + (1-\alpha_n)(1-\alpha_{n-1}) \alpha_{n-2} R_{n-2} + \dots \\
&\qquad + (1-\alpha_n)(1-\alpha_{n-1}) \dots (1-\alpha_2) \alpha_1 R_1 \\
&\qquad + (1-\alpha_n)(1-\alpha_{n-1}) \dots (1-\alpha_2)(1-\alpha_1) Q_1 \\
&= Q_1 \prod_{i=1}^{n} (1-\alpha_i) + \sum_{i=1}^{n} \alpha_i R_i \prod_{j=i+1}^{n} (1-\alpha_j)
\end{aligned} \tag{2.6}$$

□

Exercise 2.5. The result is given in figure 2.1. See the `repository` for the code. □

Exercise 2.6. The optimistic method tries all actions in the early stages. In the middle of this early period, the selection is quite random between unselected

actions. Thus, the reward would be most likely equal to the average reward for random selection. However, as more actions are selected, the average reward drops since the selection pool of remaining unselected actions gets smaller, thus the probability of selecting good actions randomly drops.

If the optimistic values get closer to zero there will be less spike. On the other hand, as the optimism increases, the magnitude of the spike will saturates as some value that depends on the problem details. □

Exercise 2.7. We can write

$$\bar{o}_n = \bar{o}_{n-1} + \alpha\left(1 - \bar{o}_{n-1}\right) = \alpha + (1-\alpha)\,\bar{o}_{n-1} \tag{2.7}$$

Also we can write

$$\begin{aligned} Q_{n+1} &= Q_n + \beta_n\left(R_n - Q_n\right) \\ &= \beta_n R_n + (1-\beta_n)\,Q_n \\ &= \frac{\alpha}{\bar{o}_n} + \left(1 - \frac{\alpha}{\bar{o}_n}\right) Q_n \end{aligned} \tag{2.8}$$

By multiplying both sides of equation (2.8) to $\bar{o}_n$, we can write

$$\bar{o}_n Q_{n+1} = \alpha R_n + \left(\bar{o}_n - \alpha\right) Q_n \tag{2.9}$$

Now, we can substitute $\bar{o}_n$ in the right hand side of equation (2.9) with equation (2.8) and write

$$\begin{aligned} \bar{o}_n Q_{n+1} &= \alpha R_n + \left(\alpha + (1-\alpha)\,\bar{o}_{n-1}\right) Q_n \\ &= \alpha R_n + (1-\alpha)\,\bar{o}_{n-1} Q_n \end{aligned} \tag{2.10}$$

By using equation (2.6) of book, i.e., $Q_{n+1} = (1-\alpha)^n Q_1 + \sum_{i=1}^{n} \alpha\,(1-\alpha)^{i} R_i$, and $\bar{o}_1 = 0$ we can simplify equation (2.10) into the following form

$$\begin{aligned} \bar{o}_n Q_{n+1} &= (1-\alpha)^n\,\bar{o}_1 Q_1 + \sum_{i=1}^{n} \alpha\,(1-\alpha)^{i} R_i \\ &= \sum_{i=1}^{n} \alpha\,(1-\alpha)^{i} R_i \end{aligned} \tag{2.11}$$

In a separate path, equation (2.7) can be iterated by itself in the following way

$$\begin{aligned} \bar{o}_n &= \alpha + (1-\alpha)\,\bar{o}_{n-1} \\ &= \alpha + (1-\alpha)\left[\alpha + (1-\alpha)\,\bar{o}_{n-2}\right] \\ &= \alpha + (1-\alpha)\,\alpha + (1-\alpha)^2\,\bar{o}_{n-2} \\ &= \alpha + \alpha(1-\alpha) + \cdots + \alpha(1-\alpha)^{n-1} + (1-\alpha)^n\,\bar{o}_0 \\ &= \alpha \sum_{i=1}^{n-1} (1-\alpha)^{i} \end{aligned} \tag{2.12}$$

Using equations (2.11) and (2.12) we can write

$$O_{n+1} = \frac{\sum_{i=1}^{n} \alpha (1-\alpha)^i R_i}{\alpha \sum_{i=1}^{n-1} (1-\alpha)^i} = \frac{\sum_{i=1}^{n} (1-\alpha)^i R_i}{\sum_{i=1}^{n-1} (1-\alpha)^i} \tag{2.13}$$

which is independent of initial action value estimates Q_1. □

Exercise 2.8. Till the 11-th step, actions are chosen from unselected actions, but in the 11-th step the estimators have some values, and action selection is based on those value estimates, thus it produces a spike. Immediately after the 11-th step, the term related to upper confidence bounds becomes less for the chosen action, and there will be some milder exploration, thus there is a mild reduction after the spike. If c is reduced there will be fewer exploratory behaviours therefore the spike is shorter. □

Exercise 2.9. The set of actions are $\{a_1, a_2\}$. We can write

$$\begin{aligned} \Pr(a_1) &= \frac{e_t^H(a_1)}{e_t^H(a_1) + e_t^H(a_2)} \\ &= \frac{e_t^H(a_1)}{e_t^H(a_1)} \frac{1}{1 + {}^{e(a)}/_{e(a)}} \\ &= \frac{1}{1 + e^{-(H(a) - H(a))}} \\ &= \frac{1}{1 + e^{-x}} \end{aligned} \tag{2.14}$$

where $x \doteq H_t(a_1) - H_t(a_2)$. Since there are only two actions, we have

$$\Pr(a_2) = 1 - \Pr(a_1) \tag{2.15}$$

Therefore *soft-max distribution* for two actions is the same as logistic function. □

Exercise 2.10. If we do not know which k-arm bandit we are facing, we have

$$\begin{aligned} \mathbb{E}[R_t | A_t = a] = \mathbb{E}[R_t | A_t = a, \text{case A}] \Pr[\text{case A}] \\ + \mathbb{E}[R_t | A_t = a, \text{case B}] \Pr[\text{case B}] \end{aligned} \tag{2.16}$$

Thus, we can write

$$\mathbb{E}[R_t | A_t = a_1] = 10 \times 0.5 + 90 \times 0.5 = 50 \tag{2.17}$$

$$\mathbb{E}[R_t | A_t = a_2] = 20 \times 0.5 + 80 \times 0.5 = 50 \tag{2.18}$$

Therefore, there is no difference in taking each of the actions.

If we know which case we are facing we have

$$\begin{aligned} \max \mathbb{E}[R_t] &= \max \mathbb{E}[R_t | \text{Case A}] \Pr[\text{case A}] + \max \mathbb{E}[R_t | \text{Case B}] \Pr[\text{case B}] \\ &= 20 \times 0.5 + 90 \times 0.5 = 55 \end{aligned} \tag{2.19}$$

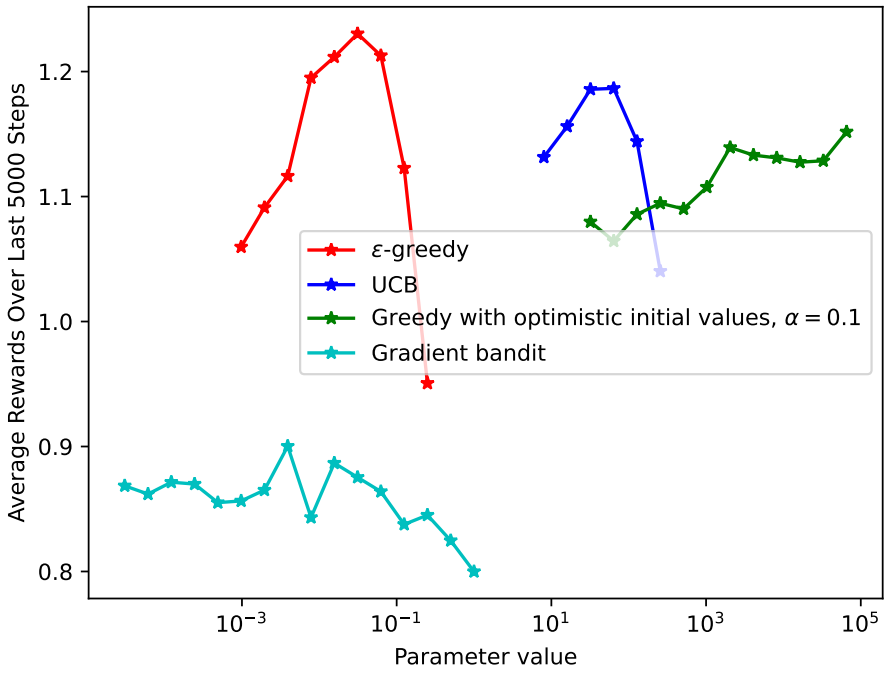


Figure 2.2: Sensitivity analysis of different approaches for nonstationary bandit problem, exercise 2.11

Exercise 2.11. The result is given in figure 2.2. As it can be seen the result is not fully replicating the figure given in the book, however it is showing the same behaviors. I think this is due to simulating multiple instances of bandits in parallel. See the `repository` for the code.

Chapter 3

Exercise 3.1.

- Planning schedules for elevators of a high-rise building. The action is whether or not to open the door of each elevator on the floors, whether or not to move, and the direction of movement for each elevator. The state can be the number of people waiting on each floor, their destination, and current occupants of each elevator, their destinations, and the current positions and direction of each elevator. The reward can be a negative number assigned to people waiting for elevators.
- Autonomous driving. The actions can be the velocity goal of the car and the direction to go. The state is the current velocity, direction, and position of the car. Also, the state includes the given destination of the car sensory information about the surroundings, and a map of the place. The reward includes negative values for each step that the car does not reach to destination. There should also be bigger negative rewards for colliding and aggressive driving.
- Investing in the stock market. The actions are whether or not to buy or sell stocks, which stocks to buy or sell, and how much to buy or sell. The state could be the history of stock market prices, compiled history of news around the world, and the available capital. The rewards are the gain or loss of the investors' capital in each step.

□

Exercise 3.2. Systems possessing infinite-dimensional states or exhibiting singularities in their dynamics can pose challenges when attempting to model them using MDPs. A prime illustration of this complexity can be found in the evolution of universe, where the interconnectedness of all its components defies straightforward MDP modeling. Furthermore, the enigmatic behavior of black holes introduces singularities that disrupt the celestial system's predictability. □

Exercise 3.3. I prefer the first option. Apart from the criteria explaining that the boundary of agent-environment is where the agent can arbitrarily make

changes, the boundary should also be where it makes the problem easier in terms of other aspects, e.g., measuring states, and implementing actions. Within the possible agent-environment boundary levels the ones that have simpler MDP representations are fundamentally more preferable. □

Exercise 3.4. The `state-action-next_state-reward` quadruplets that are not mentioned in table 3.1 have zero probability.

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

Table 3.1: Transition table.

□

Exercise 3.5. Every $s' \in \mathcal{S}$ should be converted to $s' \in \mathcal{S}^+$ to include terminal state in future states, that is,

$$\sum_{s\in\mathcal{S}}\sum_{r\in\mathcal{R}} p(s', r|s, a) = 1, \text{ for all } s \in \mathcal{S}, a \in \mathcal{A}(s) \tag{3.1}$$

Exercise 3.6. If the episodes end only when failure happens, this is similar to a continuous case and both receive the reward $-\gamma^{K-1}$ where K is the time of failures.

If the maximum length of episodes is set to T, then if a failure happens before episode length the reward will be $-\gamma^{K-1}, K \leq T$. On the other hand, if failure does not happen in the episode the reward will be 0. This episodic reward encourages not failing for the episode length and does not care if a failure happens after T steps. Thus in this case the reward is different than that of the continuous case, that is, $-\gamma^{K-1}$. □

Exercise 3.7. With the given reward, the agent's priority isn't the speed of escaping the maze. Agent's sole concern is successfully escaping the maze, irrespective of the time taken to achieve it. In order to effectively incentivize the agent to escape the maze as fast as possible, we should discount the reward, resulting in total expected reward γ^{K-1}. In this setup, the smaller the termination time K, the greater the reward. Consequently, this approach encourages the agent to exit the maze faster. □

Exercise 3.8. We use $G_t = R_{t+1} + \gamma G_{t+1}$ and since this is an episode, we have

$G_5 = 0$. Thus,

$$\begin{aligned} G_5 &= 0 \\ G_4 &= R_5 + \gamma G_5 = 2 \\ G_3 &= R_4 + \gamma G_4 = 4 \\ G_2 &= R_3 + \gamma G_3 = 8 \\ G_1 &= R_2 + \gamma G_2 = 6 \\ G_0 &= R1 + \gamma G_1 = 2 \end{aligned} \tag{3.2}$$

□

Exercise 3.9. We have

$$\begin{aligned} G_1 &= 7 + \gamma 7 + \gamma^2 7 + \dots \\ &= 7 \sum_{k=0}^{\infty} \gamma^k = 7 \frac{1}{1-\gamma} = \frac{7}{0.1} = 70 \end{aligned} \tag{3.3}$$

and

$$G_0 = R_1 + \gamma G_1 = 2 + 0.9 \times 70 = 65 \tag{3.4}$$

□

Exercise 3.10. We want to prove

$$1 + \gamma + \gamma^2 + \gamma^3 + \dots = \frac{1}{1-\gamma} \tag{3.5}$$

with $\gamma \in [0, 1)$. We start by multiplying both side by $1 - \gamma$,

$$(1-\gamma)\left(1 + \gamma + \gamma^2 + \gamma^2 + \dots\right) = 1 \tag{3.6}$$

We can expand the left hand side as

$$\begin{aligned} \left(1 + \gamma + \gamma^2 + \gamma^3 + \dots\right) - \left(\gamma + \gamma^2 + \gamma^2 + \dots\right) = \\ 1 + \gamma - \gamma + \gamma^2 - \gamma^2 + \gamma^3 - \gamma^3 + \dots = 1 \end{aligned} \tag{3.7}$$

Thus the equality is proved. □

Exercise 3.11. We can write

$$\begin{aligned} r_\pi(s) = \mathbb{E}\left[R_{t+1} | S_t = s\right] &= \sum_{a \in \mathcal{A}(s)} \pi(a|s)\, r(s,a) \\ &= \sum_{a \in \mathcal{A}(s)} \pi(a|s) \sum_{r \in \mathcal{R}} r \sum_{s \in \mathcal{S}} p(s', r|s, a) \end{aligned} \tag{3.8}$$

□

Exercise 3.12. We have

$$\begin{aligned} v_\pi(s) &\doteq \mathbb{E}_\pi[G_t|S_t = s] \\ &= \sum_{a\in\mathcal{A}(s)} \pi(a|s)\, \mathbb{E}_\pi[G_t|S_t = s, A_t = a] \\ &= \sum_{a\in\mathcal{A}(s)} \pi(a|s)\, q_\pi(s,a) \end{aligned} \tag{3.9}$$

□

Exercise 3.13. We have

$$\begin{aligned} q_\pi(s,a) &\doteq \mathbb{E}_\pi[G_t|S_t = s, A_t = a] \\ &= \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1}|S_t = s, A_t = a] \\ &= \sum_{r\in\mathcal{R}}\sum_{s\in\mathcal{S}} p(s', r|s, a)\left[r + \gamma\mathbb{E}_\pi[G_{t+1}|S_{t+1} = s']\right] \\ &= \sum_{r\in\mathcal{R}}\sum_{s\in\mathcal{S}} p(s', r|s, a)\left[r + \gamma v_\pi(s')\right] \end{aligned} \tag{3.10}$$

□

Exercise 3.14. The dynamics of the MDP, that is $p(s', r|s, a)$, is deterministic. Thus we have

$$\begin{aligned} v_\pi(\text{center}) &= \sum_a \pi(a|s)\left[r + \gamma v_\pi(s')\right] \\ &= \frac{1}{4}(0 + 0.9\cdot 2.3) + \frac{1}{4}(0 + 0.9\cdot 0.4) \\ &\quad + \frac{1}{4}(0 - 0.9\cdot 0.4) + \frac{1}{4}(0 + 0.9\cdot 0.7) \\ &= 0.675 \approx 0.7 \end{aligned} \tag{3.11}$$

□

Exercise 3.15. We have

$$\begin{aligned} G'_t &= \sum_{k=0}^{\infty} \gamma^k (R_{t+k+1} + c) \\ &= \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} + \sum_{k=0}^{\infty} \gamma^k c \\ &= G_t + c\sum_{k=0}^{\infty} \gamma^k = G_t + \frac{c}{1-\gamma} \end{aligned} \tag{3.12}$$

By substitution equation (3.12) into definition of state value function, that is

$v_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s]$, we can write

$$\begin{aligned} v'_\pi(s) &= \mathbb{E}_\pi[G'_t|S_t = s] \\ &= \mathbb{E}_\pi[G_t|S_t = s] + \mathbb{E}_\pi\left[\frac{c}{1-\gamma}\right] \\ &= v_\pi(s) + \frac{c}{1-\gamma} \end{aligned} \tag{3.13}$$

Therefore, $v_c = \frac{c}{1-\gamma}$. □

Exercise 3.16. Yes, it does. The added constant adds a different amount to the expected return depending on the length of the episode. Therefore, two episodes with the same expected return but different episode lengths will have different expected returns if we add a constant to rewards. To express this more formally, let us assume that T_1 and T_2 are terminal times of two episodes such that $T_2 > T_1$ and $G_t^1 = G_t^2$, where $G_t \doteq \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k$. Now if we consider a modified reward $R'_t = R_t + c$ we can write

$$\begin{aligned} G_t^{1\prime} &= \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k + \sum_{k=t+1}^{T} \gamma^{k-t-1} c \\ &= G_t^1 + c \sum_{k=t+1}^{T} \gamma^{k-t-1} \\ &= G_t^1 + c \sum_{k=0}^{T-t-1} \gamma^k \end{aligned} \tag{3.14}$$

and

$$G_t^{2\prime} = G_t^2 + c \sum_{k=0}^{T-t-1} \gamma^k \tag{3.15}$$

Based on our assumption, that is $T_2 > T_1$, we have $\sum_{k=0}^{T-t-1} \gamma^k > \sum_{k=0}^{T-t-1} \gamma^k$. Therefore $G_t^{2\prime} > G_t^{1\prime}$ and we conclude that adding a constant to all rewards changes the relative values in episodic tasks. As a numerical example, the following reward series have the same expected return

$$G_0^1 = R_1 + R_2 = 1 + 1 = 2 \tag{3.16}$$

and

$$G_0^2 = R_1 + R_2 + R_3 + R_4 = 1 + 0 + 0 + 1 = 2 \tag{3.17}$$

Adding $c = 10$ to all rewards we can write

$$G_0^{1\prime} = R_1 + R_2 = 11 + 11 = 22 \tag{3.18}$$

and

$$G_0^{2\prime} = R_1 + R_2 + R_3 + R_4 = 11 + 10 + 10 + 11 = 42 \tag{3.19}$$

That changed relative expected returns. □

Exercise 3.17. We have

$$\begin{aligned} q_\pi(s,a) &= \mathbb{E}_\pi[G_t|S_t=s,A_t=a] \\ &= \mathbb{E}_\pi[R_{t+1}+\gamma G_{t+1}|S_t=s,A_t=a] \\ &= \sum_{s,r} p(s',r|s,a)\left[r+\gamma\mathbb{E}_\pi[G_{t+1}|S_{t+1}=s']\right] \\ &= \sum_{s,r} p(s',r|s,a)\left[r+\gamma\sum_a \pi(a'|s')\,\mathbb{E}_\pi[G_{t+1}|S_{t+1}=s',A_{t+1}=a']\right] \\ &= \sum_{s,r} p(s',r|s,a)\left[r+\gamma\sum_a \pi(a'|s')\,q_\pi(s',a')\right] \end{aligned} \tag{3.20}$$

Exercise 3.18. We have

$$\begin{aligned} v_\pi(s) &= \mathbb{E}_\pi[q_\pi(s,a)] \\ &= \sum_a \pi(a|s)\,q_\pi(s,a) \end{aligned} \tag{3.21}$$

□

Exercise 3.19. We have

$$\begin{aligned} q_\pi(s,a) &= \mathbb{E}[R_{t+1}+\gamma G_{t+1}] \\ &= \sum_{s,r} p(s',r|s,a)\left[r+\gamma v_\pi(s')\right] \end{aligned} \tag{3.22}$$

□

Exercise 3.20. An optimal policy is to use drivers outside green areas and putters inside green areas. Therefore, the optimal state value function $v_*(s)$ can be described by values of $q_*(s,\texttt{driver})$ outside of the green area and $v_{\texttt{putt}}$ inside the green area.

Exercise 3.21. The optimal action value function $q_*(s,\texttt{putter})$ can approximately be described as

Current regions	Value
green	-1
contour -2 of $v_{\texttt{putt}}$	-2
sand, contours -3 and -4 of $v_{\texttt{putt}}$	-3
contours -5 and -6 of $v_{\texttt{putt}}$	-4

Table 3.2: Value estimations after each step.

□

Exercise 3.22. Considering $\gamma \in [0,)$, we can write

$$\begin{aligned} v_{\texttt{left}} &= 1\gamma^0 + 0\gamma^1 + 1\gamma^2 + 0\gamma^3 + \dots \\ &= \sum_{k=0}^{\infty} \gamma^{2k} = \sum_{k=0}^{\infty} \left(\gamma^2\right)^k \\ &= \frac{1}{1-\gamma^2} \end{aligned} \tag{3.23}$$

and

$$\begin{aligned} v_{\texttt{right}} &= 0\gamma^0 + 2\gamma^1 + 0\gamma^2 + 2\gamma^3 + \dots \\ &= 2\sum_{k=0}^{\infty} \gamma^{2k+1} = 2\gamma \sum_{k=0}^{\infty} \gamma^{2k} = 2\gamma \sum_{k=0}^{\infty} \left(\gamma^2\right)^k \\ &= \frac{2\gamma}{1-\gamma^2} \end{aligned} \tag{3.24}$$

If $\gamma = 0$, we have $v_{\texttt{left}} = 1$ and $v_{\texttt{right}} = 0$, so $\pi_{\texttt{left}}$ is better. If $\gamma = 0.5$, we have $v_{\texttt{left}} = 4/3 \approx 1.33$ and $v_{\texttt{right}} = 4/3 \approx 1.33$, so both policies are the same. If $\gamma = 0.9$, we have $v_{\texttt{left}} = 100/19 \approx 5.26$ and $v_{\texttt{right}} = 180/19 \approx 9.47$, so $\pi_{\texttt{right}}$ is better. □

Exercise 3.23. We have

$$\begin{aligned} q_*(\texttt{h}, \texttt{s}) &= p(\texttt{h}|\texttt{h}, \texttt{s}) \Big[r(\texttt{h}, \texttt{h}, \texttt{s}) + \gamma \max_{a \in \mathcal{A}(\texttt{h})} q_*(\texttt{h}, a') \Big] \\ &\quad + p(\texttt{l}|\texttt{h}, \texttt{s}) \Big[r(\texttt{l}, \texttt{h}, \texttt{s}) + \gamma \max_{a \in \mathcal{A}(\texttt{l})} q_*(\texttt{l}, a') \Big] \\ &= \alpha \Big[r_{\texttt{s}} + \gamma \max_{a \in \mathcal{A}(\texttt{h})} q_*(\texttt{h}, a') \Big] + (1-\alpha) \Big[r_{\texttt{s}} + \gamma \max_{a \in \mathcal{A}(\texttt{l})} q_*(\texttt{l}, a') \Big] \end{aligned} \tag{3.25}$$

and

$$\begin{aligned} q_*(\texttt{h}, \texttt{w}) &= p(\texttt{h}|\texttt{h}, \texttt{w}) \Big[r(\texttt{h}, \texttt{h}, \texttt{w}) + \gamma \max_{a \in \mathcal{A}(\texttt{h})} q_*(\texttt{h}, a') \Big] \\ &\quad + p(\texttt{l}|\texttt{h}, \texttt{w}) \Big[r(\texttt{l}, \texttt{h}, \texttt{w}) + \gamma \max_{a \in \mathcal{A}(\texttt{l})} q_*(\texttt{l}, a') \Big] \\ &= r_{\texttt{w}} + \gamma \max_{a \in \mathcal{A}(\texttt{h})} q_*(\texttt{h}, a') \end{aligned} \tag{3.26}$$

and

$$\begin{aligned} q_*(\texttt{l}, \texttt{s}) &= p(\texttt{h}|\texttt{l}, \texttt{s}) \Big[r(\texttt{h}, \texttt{l}, \texttt{s}) + \gamma \max_{a \in \mathcal{A}(\texttt{h})} q_*(\texttt{h}, a') \Big] \\ &\quad + p(\texttt{l}|\texttt{l}, \texttt{s}) \Big[r(\texttt{l}, \texttt{l}, \texttt{s}) + \gamma \max_{a \in \mathcal{A}(\texttt{l})} q_*(\texttt{l}, a') \Big] \\ &= (1-\beta) \Big[-3 + \gamma \max_{a \in \mathcal{A}(\texttt{h})} q_*(\texttt{h}, a') \Big] + \beta \Big[r_{\texttt{s}} + \gamma \max_{a \in \mathcal{A}(\texttt{l})} q_*(\texttt{l}, a') \Big] \end{aligned} \tag{3.27}$$

and

$$\begin{aligned} q_*(\mathtt{l},\mathtt{w}) &= p(\mathtt{h}|\mathtt{l},\mathtt{w})\left[r(\mathtt{h},\mathtt{l},\mathtt{w}) + \gamma \max_{a\in\mathcal{A}(\mathtt{h})} q_*(\mathtt{h},a')\right] \\ &\quad + p(\mathtt{l}|\mathtt{l},\mathtt{w})\left[r(\mathtt{l},\mathtt{l},\mathtt{w}) + \gamma \max_{a\in\mathcal{A}(\mathtt{l})} q_*(\mathtt{l},a')\right] \\ &= r_{\mathtt{w}} + \gamma \max_{a\in\mathcal{A}(\mathtt{l})} q_*(\mathtt{l},a') \end{aligned} \tag{3.28}$$

and

$$\begin{aligned} q_*(\mathtt{l},\mathtt{r}) &= p(\mathtt{h}|\mathtt{l},\mathtt{r})\left[r(\mathtt{h},\mathtt{l},\mathtt{r}) + \gamma \max_{a\in\mathcal{A}(\mathtt{h})} q_*(\mathtt{h},a')\right] \\ &\quad + p(\mathtt{l}|\mathtt{l},\mathtt{r})\left[r(\mathtt{l},\mathtt{l},\mathtt{r}) + \gamma \max_{a\in\mathcal{A}(\mathtt{l})} q_*(\mathtt{l},a')\right] \\ &= \gamma \max_{a\in\mathcal{A}(\mathtt{l})} q_*(\mathtt{l},a') \end{aligned} \tag{3.29}$$

□

Exercise 3.24. For $\gamma \in [0,1)$, we have

$$\begin{aligned} v_*(A) &= 10\gamma^0 + 0\gamma^1 + 0\gamma^2 + 0\gamma^3 + 0\gamma^4 \\ &\quad + 10\gamma^5 + 0\gamma^6 + 0\gamma^7 + 0\gamma^8 + 0\gamma^9 + 10\gamma^{10} + \dots \\ &= 10\sum_{k=0}^{\infty}\gamma^{5k} = 10\sum_{k=0}^{\infty}\left(\gamma^5\right)^k \\ &= \frac{10}{1-\gamma^5} \end{aligned} \tag{3.30}$$

Substituting $\gamma = 0.9$, we obtain $v_*(A) \approx 24.419$. □

Exercise 3.25.

$$v_*(s) = \max_{a\in\mathcal{A}(s)} q_*(s,a) \tag{3.31}$$

□

Exercise 3.26.

$$q_*(s,a) = \sum_{s',r} p(s',r|s,a)\Big[r + \gamma v_*(s')\Big] \tag{3.32}$$

□

Exercise 3.27.

$$\pi_*(s) = \arg\max_{a\in\mathcal{A}(s)} q_*(s,a) \tag{3.33}$$

□

Exercise 3.28.

$$\pi_*(s) = \arg\max_{a \in \mathcal{A}(s)} \left\{ \sum_{s,r} p(s', r|s, a) \Big[r + \gamma v_*(s') \Big] \right\} \tag{3.34}$$

□

Exercise 3.29. Second line of derivations use the fact that sum of probabilities is equal to one.

$$\begin{aligned} v_\pi(s) &= \sum_a \pi(a|s) \Big[r(s,a) + \gamma \sum_s p(s'|s,a) v_\pi(s') \Big] \\ &= \sum_a \pi(a|s) \sum_s p(s'|s,a) \Big[r(s,a) + \gamma v_\pi(s') \Big] \end{aligned} \tag{3.35}$$

$$\begin{aligned} v_*(s) &= \max_a \left\{ r(s,a) + \gamma \sum_s p(s'|s,a) v_*(s') \right\} \\ &= \max_a \sum_s p(s'|s,a) \Big[r(s,a) + \gamma v_*(s') \Big] \end{aligned} \tag{3.36}$$

$$\begin{aligned} q_\pi(s,a) &= \sum_a \pi(a|s) \Big[r(s,a) + \gamma \sum_s p(s'|s,a) \sum_a \pi(a'|s') q_\pi(s', a') \Big] \\ &= \sum_a \pi(a|s) \sum_s p(s'|s,a) \Big[r(s,a) + \gamma \sum_a \pi(a'|s') q_\pi(s', a') \Big] \end{aligned} \tag{3.37}$$

$$\begin{aligned} q_*(s,a) &= r(s,a) + \gamma \sum_s p(s'|s,a) \max_a q_*(s', a') \\ &= \sum_s p(s'|s,a) \Big[r(s,a) + \gamma \max_a q_*(s', a') \Big] \end{aligned} \tag{3.38}$$

Chapter 4

Exercise 4.1.

$$q_\pi(11, \text{down}) = 1 \cdot (-1 + v_\pi(s_{\text{Terminal}})) = -1 + 0 = -1 \tag{4.1}$$

$$q_\pi(7, \text{down}) = 1 \cdot (-1 + v_\pi(11)) = -1 - 14 = -15 \tag{4.2}$$

□

Exercise 4.2. For unchanged policy,

$$\begin{aligned} v_\pi(15) &= \frac{1}{4}(-1 + v_\pi(12)) + \frac{1}{4}(-1 + v_\pi(13)) \\ &\quad + \frac{1}{4}(-1 + v_\pi(14)) + \frac{1}{4}(-1 + v_\pi(15)) \\ &= -\frac{60}{4} + \frac{1}{4}v_\pi(15) \end{aligned} \tag{4.3}$$

Therefore $v_\pi(15) = -20$.

For changed policy,

$$\begin{aligned} v_\pi(13) &= \frac{1}{4}(-1 + v_\pi(12)) + \frac{1}{4}(-1 + v_\pi(9)) \\ &\quad + \frac{1}{4}(-1 + v_\pi(14)) + \frac{1}{4}(-1 + v_\pi(15)) \\ &= -\frac{60}{4} + \frac{1}{4}v_\pi(15) \end{aligned} \tag{4.4}$$

and

$$\begin{aligned} v_\pi(15) &= \frac{1}{4}(-1 + v_\pi(12)) + \frac{1}{4}(-1 + v_\pi(13)) \\ &\quad + \frac{1}{4}(-1 + v_\pi(14)) + \frac{1}{4}(-1 + v_\pi(15)) \\ &= -\frac{40}{4} + \frac{1}{4}v_\pi(13) + \frac{1}{4}v_\pi(15) \end{aligned} \tag{4.5}$$

Combining above equations we can write

$$\begin{cases} v_\pi(15) - 4 \cdot v_\pi(13) = 60 \\ -3 \cdot v_\pi(15) + v_\pi(13) = 40 \end{cases} \tag{4.6}$$

Solving th system of equations we can write $v_\pi(13) = -20$ and $v_\pi(15) = -20$. □

Exercise 4.3.

$$\begin{aligned} q_\pi(s,a) &= \mathbb{E}_\pi \left[G_t | S_t = s, A_t = a \right] \\ &= \mathbb{E}_\pi \left[R_{t+1} + \gamma G_{t+1} | S_t = s, A_t = a \right] \\ &= \sum_{s,r} p(s', r|s, a) \Big[r + \gamma \mathbb{E}_\pi \left[G_{t+1} | S_{t+1} = s' \right] \Big] \\ &= \sum_{s,r} p(s', r|s, a) \Big[r + \gamma \sum_a \pi(a'|s') \mathbb{E}_\pi \left[G_{t+1} | S_{t+1} = s', A_{t+1} = a' \right] \Big] \\ &= \sum_{s,r} p(s', r|s, a) \Big[r + \gamma \sum_a \pi(a'|s') q_\pi(s', a') \Big] \end{aligned} \tag{4.7}$$

and

$$\begin{aligned} q_{k+1}(s,a) &= \mathbb{E}_\pi \left[R_{t+1} + \gamma \sum_A \pi(A_{t+1}|S_{t+1}) q_k(S_{t+1}, A_{t+1}) | S_t = s, A_t = a \right] \\ &= \sum_{s,r} p(s', r|s, a) \Big[r + \gamma \sum_a \pi(a'|s') q_k(s', a') \Big] \end{aligned} \tag{4.8}$$

□

Exercise 4.4. We can modify the algorithm to assess convergence using the state value function. This adjustment involves storing state values in an array, e.g., V_{old}, prior to policy evaluation and comparing these values with the updated state values, that is V, following policy evaluation. If the maximum absolute difference between the previous state values and the updated values falls below the threshold Δ, it signifies that the algorithm is converged. □

Exercise 4.5. The algorithm is given in Policy Iteration using action value function algorithm. □

Exercise 4.6. In step 1, we should have probabilistic policy $\pi(a|s)$ such that $\pi(a|s) \geq \epsilon/|\mathcal{A}(s)|$ for all $a \in \mathcal{A}(a)$. In step 2, we will have

$$V(s) \leftarrow \sum_a \pi(a|s) \sum_{s,a} p(s', r|s, a)[r + \gamma V(s')].$$

Algorithm 1 Policy Iteration using action value function $Q(s, a)$

1: Initialization
$Q(s, a) \in \mathbb{R}$ and $\pi(s) \in \mathcal{A}(s)$ arbitrarily for all $s \in \mathcal{S}$ and $a \in \mathcal{A}(s)$; $Q(\textit{terminal}, a) \doteq 0$

2: Policy Evaluation
Loop:
$\Delta \leftarrow 0$
Loop for each $s \in \mathcal{S}$ and $a \in \mathcal{A}(a)$:
$q \leftarrow Q(s, a)$
$Q(s, a) \leftarrow \sum_{s,r} p(s', r|s, a)[r + \gamma Q(s', \pi(s'))]$
$\Delta \leftarrow \max(\Delta, |q - Q(s, a|)$
until $\Delta < \theta$ (a small positive number determining the accuracy of estimation)

3: Policy Improvement
policy-stable $\leftarrow$ *true*
For each $s \in \mathcal{S}$:
old-action $\leftarrow \pi(s)$
$\pi(s) \leftarrow \arg\max_a Q(s, a)$
If *old-action* $\neq \pi(s)$, then *policy-stable* $\leftarrow$ *false*
If *policy-stable*, then stop and return $Q \approx q_*$ and $\pi \approx \pi_*$; else go to 2

In step 3, we should have

$$\begin{aligned} a_* &\leftarrow \arg\max_a \sum_{s,r} p(s', r|s, a)[r + \gamma V(s')] \\ \pi(a_*|s) &\leftarrow 1 - \epsilon\frac{|\mathcal{A}| - 1}{|\mathcal{A}|} \text{ and } \pi(a|s) = \frac{\epsilon}{|\mathcal{A}|} \text{ for } a \in \mathcal{A}(s) \neq a_* \end{aligned} \tag{4.9}$$

□

Exercise 4.7. The result is given in figure 4.1. See the `repository` for the code.

Exercise 4.8. It's evident that a gambler's likelihood of winning increases with a larger amount of money. Given a win probability of $p_h = 0.4$, when the gambler possesses \$50 dollars, staking all \$50 dollars results in an exact win probability of $p_{\text{win}} = p_h = 0.4$. Betting any amount lower would yield a winning chance of at most $p_{\text{win}} = (1 - p_h) \cdot p_h \cdot p_h$, which is less than 0.4. Therefore, the optimal strategy is to wager all available funds.

Since the odds are against us, it's logical to bet as few as possible. For sums of money less than \$50, the most effective approach is to reach \$50 using the fewest possible bets. For sums exceeding \$50, consider the case of having \$51. Betting \$1, even in the event of a loss, maintains a winning chance of 0.4. Thus, the best course of action remains to reach \$50 in as few bets as possible.

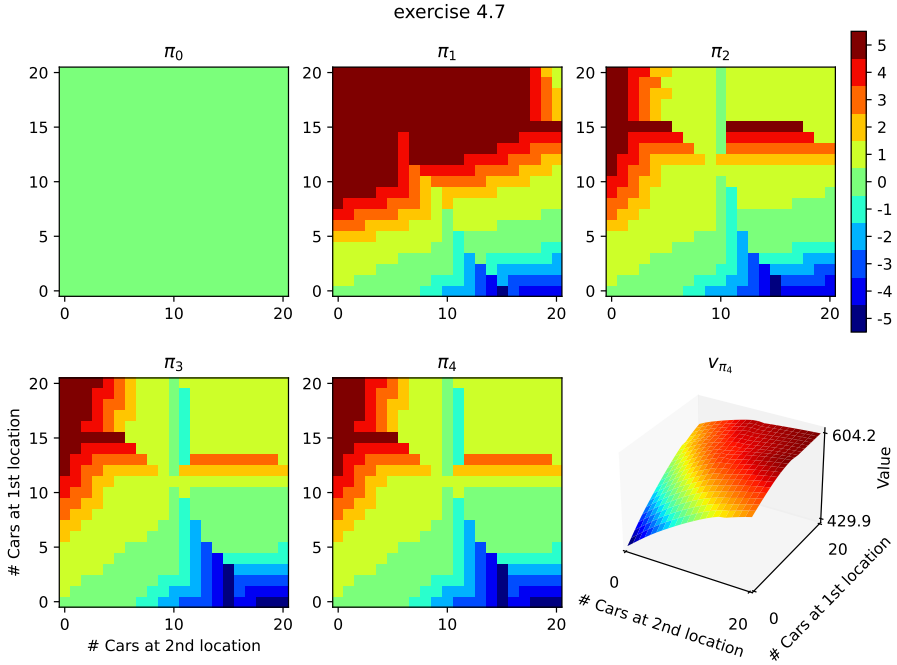
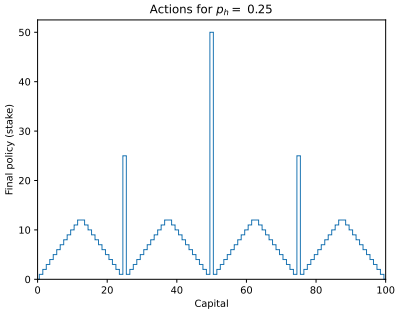


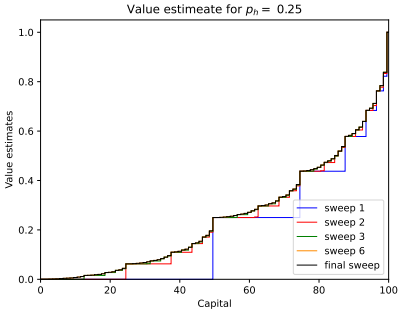
Figure 4.1: Sensitivity analysis of different approaches for nonstationary bandit problem, exercise 4.7

Exercise 4.9. The result is given in figure 4.2. See the `repository` for the code.

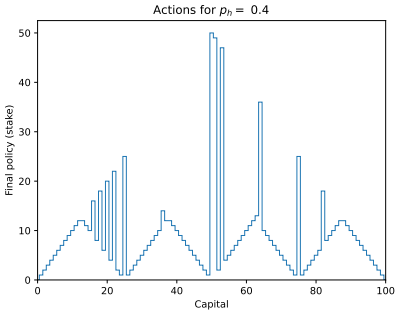
Exercise 4.10. The algorithm is given in Value Iteration using action value function algorithm. □



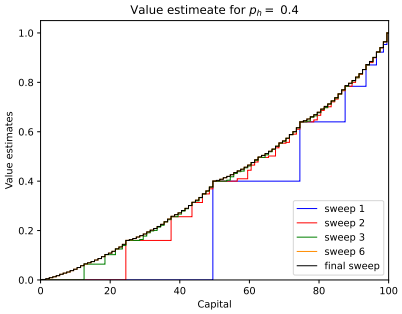
(a) Policy for $p_h = 0.25$



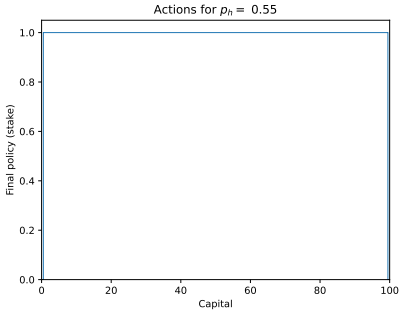
(b) Value estimate for $p_h = 0.25$



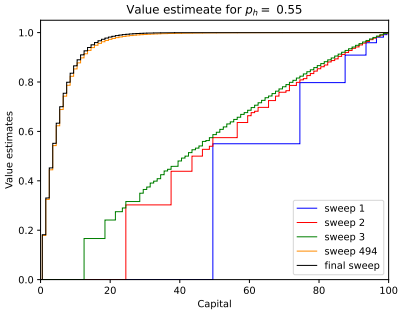
(c) Policy for $p_h = 0.40$



(d) Value estimate for $p_h = 0.40$



(e) Policy for $p_h = 0.55$



(f) Value estimate for $p_h = 0.55$

Figure 4.2: The solution to the gambler's problem for different p_h, exercise 4.9

Algorithm 2 Value Iteration using action value function $Q(s, a)$

Initialization
$Q(s, a) \in \mathbb{R}$ arbitrarily for all $s \in \mathcal{S}$ and $a \in \mathcal{A}(s)$; $Q(\textit{terminal}, a) \doteq 0$

Loop:
  $\Delta \leftarrow 0$
  Loop for each $s \in \mathcal{S}$ and $a \in \mathcal{A}(a)$:
    $q \leftarrow Q(s, a)$
    $Q(s, a) \leftarrow \sum_{s,r} p(s', r|s, a)[r + \gamma \max_a Q(s', a')]$
    $\Delta \leftarrow \max(\Delta, |q - Q(s, a|)$
until $\Delta < \theta$ (a small positive number determining the accuracy of estimation)
Output deterministic policy, $\pi \approx \pi_*$, such that
  $\pi(s) = \arg\max_a Q(s, a)$

Chapter 5

Exercise 5.1. The policy of sticking if the player's sum is 20 or 21 has a high probability of winning when the player's sum is greater than 19, therefore there is a jump in states' value for the last two rows.

This row means the dealer has an ace. So he has the chance to count it as 1 or 11 if needed, thus reducing the winning chance of the player.

In those states the policy is to hit. When the player has a usable ace and he hits, he still has a chance to count his ace as 1, therefore the chance of losing is less when having a usable ace and its value is higher.

Exercise 5.2. The final values will be the same since both methods converge to the same values.

In this specific game visiting the same state can happen if switching from usable ace to 1. The probability of such an event is not high so there won't be a big difference in the early episodes too.

Exercise 5.3. The backup diagram is almost the same as Monte Carlo policy evaluation for state value function, the only difference is that for action values, it will start with a full black dot representing state value pairs, assuming that we are using exploring starts.

Exercise 5.4. We need to make the following changes. In the initialization, we remove $Returns(s, a)$ and add the following line,

$$N(s, a) \leftarrow 0 \text{ for all } s \in \mathcal{S}, a \in \mathcal{A}(s)$$

Also, need to remove the two lines after "Unless" condition and change them to

$$N(S_t, A_t) \leftarrow N(S_t, A_t) + 1$$
$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{1}{N(S_t, A_t)} \left(G - Q(S_t, A_t)\right)$$

□

Exercise 5.5. The data of this episode can be summarized as in table 5.1. Based

Step	0	1	2	3	4	5	6	7	8	9	10
State	S	S	S	S	S	S	S	S	S	S	T
R_t	1	1	1	1	1	1	1	1	1	1	0
G_t	10	9	8	7	6	5	4	3	2	1	0

Table 5.1: Value estimations after each step.

on this data we can write,

$$\begin{aligned} v^{\text{first-visit}}(S) &= \frac{10}{1} = 10 \\ v_{\text{every-visit}}(S) &= \frac{10+9+8+7+6+5+4+3+2+1}{10} = 5.5 \end{aligned} \tag{5.1}$$

□

Exercise 5.6. *Ordinary importance sampling*,

$$Q_\pi(s,a) \doteq \frac{\sum_{t\in\mathcal{T}(s,a)} \rho'_{t:T(s,a)-1} G_t}{|\mathcal{T}(s,a)|} \tag{5.2}$$

and *weighted importance sampling*,

$$Q_\pi(s,a) \doteq \frac{\sum_{t\in\mathcal{T}(s,a)} \rho'_{t:T(s,a)-1} G_t}{\sum_{t\in\mathcal{T}(s,a)} \rho'_{t:T(s,a)-1}} \tag{5.3}$$

where

$$\rho'_{t:T-1} \doteq \frac{p(S_{t+1}|s,a)\prod_{k=t+1}^{T-1}\pi(A_k|S_k)p(S_{k+1}|S_k,A_k)}{p(S_{t+1}|s,a)\prod_{k=t+1}^{T-1}b(A_k|S_k)p(S_{k+1}|S_k,A_k)} = \frac{\prod_{k=t+1}^{T-1}\pi(A_k|S_k)}{\prod_{k=t+1}^{T-1}b(A_k|S_k)} \tag{5.4}$$

□

Exercise 5.7. Initially, it is unlikely that we see episodes with high *importance-sampling ratio*, therefore due to having very small numbers in the denominator of *weighted importance sampling* we may see an increase in error. However, as the number of episodes increases the estimated values converge to the real one and the error decreases. □

Exercise 5.8. In *every-visit* case, we need to consider all occurrences of state s

in each episode in the calculation of state value. So we have

$$\begin{aligned}
V(s) = & \frac{1}{2} \cdot 0.1 \left(\frac{1}{0.5}\right)^2 \\
& + \frac{1}{2} \cdot 0.9 \cdot \frac{1}{2} \cdot 0.1 \left[\left(\frac{1}{0.5}\frac{1}{0.5}\right)^2 + \left(\frac{1}{0.5}\right)^2\right] \\
& + \frac{1}{2} \cdot 0.9 \cdot \frac{1}{2} \cdot 0.9 \cdot \frac{1}{2} \cdot 0.1 \left[\left(\frac{1}{0.5}\frac{1}{0.5}\frac{1}{0.5}\right)^2 + \left(\frac{1}{0.5}\frac{1}{0.5}\right)^2 + \left(\frac{1}{0.5}\right)^2\right] \\
& + \dots \\
= & \, 0.1 \sum_{k=0}^{\infty} 0.9^k \cdot 2^k \cdot 2 + 0.1 \cdot 0.9 \sum_{k=0}^{\infty} 0.9^k \cdot 2^k + \dots \\
= & \, 0.2 \sum_{k=0}^{\infty} 1.8^k + 0.9 \sum_{k=0}^{\infty} 1.8^k + \dots = \infty + \infty + \dots = \infty
\end{aligned} \tag{5.5}$$

□

Exercise 5.9. In the initialization, instead of using *Returns* array, we should have

$$N(s) \leftarrow 0, \text{ for all } s \in \mathcal{S} \tag{5.6}$$

Further, in the section following "Unless ..." we should modify the algorithm as follows,

$$\begin{aligned}
& N(S_t) \leftarrow N(S_t) + 1 \\
& V(S_t) \leftarrow V(S_t) + \frac{1}{N(S_t)}[G_t - V(S_t)]
\end{aligned} \tag{5.7}$$

□

Exercise 5.10. We can write

$$\begin{aligned}
C_{n+1} & \doteq \sum_{k=1}^{n+1} W_k \\
& = \sum_{k=1}^{n} W_k + W_{n+1} \\
& = C_n + W_{n+1}
\end{aligned} \tag{5.8}$$

where $C_0 = 0$. We can also write, for $n \geq 1$,

$$\begin{aligned} V_{n+1} &\doteq \frac{\sum_{k=1}^{n} W_k G_k}{\sum_{k=1}^{n} W_k} \\ &= \frac{W_n G_n + \sum_{k=1}^{n-1} W_k G_k}{C_n} \\ &= \frac{W_n G_n + C_{n-1} V_n}{C_n} \\ &= \frac{W_n G_n + C_{n-1} V_n + W_n V_n - W_n V_n}{C_n} \\ &= \frac{W_n G_n + C_n V_n - W_n V_n}{C_n} \\ V_{n+1} &= V_n + \frac{W_n}{C_n}[G_n - V_n] \end{aligned} \tag{5.9}$$

□

Exercise 5.11. In the off-policy MC control algorithm, the target policy is deterministic and greedy, leading to $\pi(A_t|S_t) = 1$. As a result, $\frac{\pi(A|S)}{b(A|S)}$ is equal to $\frac{1}{b(A|S)}$, and there is no difference in their usage. □

Exercise 5.13. Do to independence of each time step in episode, we can write

$$\begin{aligned} \mathbb{E}[\rho_{t:T-1} R_{t+k}] &= \mathbb{E}[\rho_{t:t+k-1} R_{t+k} \rho_{t+k:T-1}] \\ &= \mathbb{E}[\rho_{t:t+k-1} R_{t+k}] \mathbb{E}[\rho_{t+k:T-1}] \\ &= \mathbb{E}[\rho_{t:t+k-1} R_{t+k}] \mathbb{E}[\frac{\pi(A_{t+k}|S_{t+k})}{b(A_{t+k}|S_{t+k})} \cdots \frac{\pi(A_{T-1}|S_{T-1})}{b(A_{T-1}|S_{T-1})}] \\ &= \mathbb{E}[\rho_{t:t+k-1} R_{t+k}] \mathbb{E}[\frac{\pi(A_{t+k}|S_{t+k})}{b(A_{t+k}|S_{t+k})}] \ldots \mathbb{E}[\frac{\pi(A_{T-1}|S_{T-1})}{b(A_{T-1}|S_{T-1})}] \\ &= \mathbb{E}[\rho_{t:t+k-1} R_{t+k}] \end{aligned} \tag{5.10}$$

Setting $k = 1$ in equation (5.10) we can write

$$\mathbb{E}[\rho_{t:T-1} R_{t+1}] = \mathbb{E}[\rho_{t:t} R_{t+1}] \tag{5.11}$$

□

Exercise 5.14. The *discounting-aware weighted importance sampling* for the action-value function can be written as follows

$$Q_\pi(s,a) \doteq \frac{\sum_{t \in \mathcal{T}(s,a)} \left((1-\gamma) \sum_{h=t+1}^{T(t)-1} \gamma^{h-t-1} \rho'_{t:h-1} \bar{G}_{t:h} + \gamma^{T(t)-t-1} \rho'_{t:T(t)-1} \bar{G}_{t:T(t)} \right)}{\sum_{t \in \mathcal{T}(s,a)} \left((1-\gamma) \sum_{h=t+1}^{T(t)-1} \gamma^{h-t-1} \rho'_{t:h-1} + \gamma^{T(t)-t-1} \rho'_{t:T(t)-1} \right)} \tag{5.12}$$

where

$$\rho'_{t:x} \doteq \frac{p(S_{t+1}|s,a) \prod_{k=t+1}^{x} \pi(A_k|S_k) p(S_{k+1}|S_k, A_k)}{p(S_{t+1}|s,a) \prod_{k=t+1}^{x} b(A_k|S_k) p(S_{k+1}|S_k, A_k)} = \frac{\prod_{k=t+1}^{x} \pi(A_k|S_k)}{\prod_{k=t+1}^{x} b(A_k|S_k)} \tag{5.13}$$

and

$$\bar{G}_{t:x} \doteq R_{t+1} + R_{t+2} + \cdots + R_x. \tag{5.14}$$

with $\rho'_{t:t} = 1$ for any t. To obtain action-value estimates, it is necessary to derive recursive expressions for the equations equations (5.12) to (5.14). To this end, equations (5.13) and (5.14) can be written as

$$\rho'_{t:x} = \frac{\pi(A_{t+1}|S_{t+1})}{b(A_{t+1}|S_{t+1})}\rho'_{t+1:x} = \rho'_{t:t+1}\rho'_{t+1:x} \tag{5.15}$$

$$\bar{G}_{t:x} = R_{t+1} + \bar{G}_{t+1:x} \tag{5.16}$$

with $\rho'_{t:t+1} = \frac{\pi(A|S)}{b(A|S)}$, $\rho'_{T-1:T} = 1$, and $\bar{G}_{T-1:T} = R_T$. We can define α_t and express it in recursively as

$$\begin{aligned}
\alpha_t &\doteq \sum_{h=t+1}^{T(t)-1} \gamma^{h-t-1}\rho'_{t:h-1} \\
&= \rho'_{t:t} + \sum_{h=t+2}^{T(t)-1} \gamma^{h-t-1}\rho'_{t:h-1} \\
&= 1 + \gamma\rho'_{t:t+1} \sum_{h=t+2}^{T(t)-1} \gamma^{h-t-2}\rho'_{t+1:h-1} \\
&= 1 + \gamma\frac{\pi(A_{t+1}|S_{t+1})}{b(A_{t+1}|S_{t+1})}\alpha_{t+1}
\end{aligned} \tag{5.17}$$

where $\alpha_{T(t)-1} = 1$ for all $t \in \mathcal{T}(s,a)$. Similarly we can define β_t and express it recursively as

$$\begin{aligned}
\beta_t &= \sum_{h=t+1}^{T(t)-1} \gamma^{h-t-1}\rho'_{t:h-1}\bar{G}_{t:h} \\
&= \rho'_{t:t}\bar{G}_{t:t+1} + \sum_{h=t+2}^{T(t)-1} \gamma^{h-t-1}\rho'_{t:h-1}\bar{G}_{t:h} \\
&= R_{t+1} + \gamma\rho'_{t:t+1} \sum_{h=t+2}^{T(t)-1} \gamma^{h-t-2}\rho'_{t+1:h-1}\left(R_{t+1} + \bar{G}_{t+1:h}\right) \\
&= R_{t+1} + \gamma\rho'_{t:t+1}\left(R_{t+1}\sum_{h=t+2}^{T(t)-1} \gamma^{h-t-2}\rho'_{t+1:h-1} + \sum_{h=t+2}^{T(t)-1} \gamma^{h-t-2}\rho'_{t+1:h-1}\bar{G}_{t+1:h}\right) \\
&= R_{t+1} + \gamma\frac{\pi(A_{t+1}|S_{t+1})}{b(A_{t+1}|S_{t+1})}\left(R_{t+1}\alpha_{t+1} + \beta_{t+1}\right)
\end{aligned} \tag{5.18}$$

where $\beta_{T(t)-1} = R_{T(t)}$ for all $t \in \mathcal{T}(s,a)$. We can also define W_t and express is recursively as

$$\begin{aligned} W'_t &= \gamma^{T(t)-t-1}\rho'_{t:T(t)-1} \\ &= \gamma\frac{\pi(A_{t+1}|S_{t+1})}{b(A_{t+1}|S_{t+1})}\gamma^{T(t)-t-2}\rho'_{t+1:T(t)-1} \\ &= \gamma\frac{\pi(A_{t+1}|S_{t+1})}{b(A_{t+1}|S_{t+1})}W'_{t+1} \end{aligned} \tag{5.19}$$

where $W'_{T(t)-1} = 1$. To facilitate recursive calculation of action-value estimation, we can define

$$\begin{aligned} X_k &\doteq (1-\gamma)\sum_{h=t+1}^{T(t)-1}\gamma^{h-t-1}\rho'_{t:h-1}\bar{G}_{t:h} + \gamma^{T(t)-t-1}\rho'_{t:T(t)-1}\bar{G}_{t:T(t)} \\ &= (1-\gamma)\beta_t + W'_t\bar{G}_{t:T(t)} \end{aligned} \tag{5.20}$$

and

$$\begin{aligned} Y_k &\doteq (1-\gamma)\sum_{h=t+1}^{T(t)-1}\gamma^{h-t-1}\rho'_{t:h-1} + \gamma^{T(t)-t-1}\rho'_{t:T(t)-1} \\ &= (1-\gamma)\alpha_t + W'_t \end{aligned} \tag{5.21}$$

where k is the index of t in the set $\mathcal{T}(s,a)$. Further, we can also define C_n and express it in recursive form as

$$\begin{aligned} C_n &\doteq \sum_{k=1}^{n} Y_k \\ &= Y_n + \sum_{k=1}^{n-1} Y_k \\ &= Y_n + C_{n-1} \end{aligned} \tag{5.22}$$

Now using X_k, Y_k, and C_n, the action-value estimation from equation (5.12)

Algorithm 3 Off-policy MC control using discounting-aware weighted importance sampling

Initialize, for all $s \in \mathcal{S}, a \in \mathcal{A}(s)$:
$Q(s,a) \in \mathbb{R}$ (arbitrarily)
$C(s,a) \leftarrow 0$
$\pi(s) \leftarrow \arg\max_a Q(s,a)$ (with ties broken consistently)

Loop forever (for each episode):
$b \leftarrow$ any soft policy
Generate an episode using b: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
$G \leftarrow 0, W' \leftarrow 1, \alpha \leftarrow 1, \beta \leftarrow R_T$
Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:
$G \leftarrow R_{t+1} + \gamma G$
$X \leftarrow (1-\gamma)\beta + W'G$
$Y \leftarrow (1-\gamma)\alpha + W'$
$C(S_t, A_t) \leftarrow C(s,a) + Y$
$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{1}{C(S,A)}[X - YQ(S_t, A_t)]$
$\pi(S_t) \leftarrow \arg\max_a Q(S_t, a)$ (with ties broken consistently)
If $A_t \neq \pi(S_t)$ then exit inner Loop (proceed to next episode)
$\beta \leftarrow R_t + \gamma \frac{1}{b(S,A)}(R_t\alpha + \beta)$
$\alpha \leftarrow 1 + \gamma \frac{1}{b(S,A)}\alpha$
$W' \leftarrow \gamma \frac{1}{b(S,A)} W'$

can be expressed recursively as

$$\begin{aligned} Q_{n+1} &= \frac{\sum_{k=1}^{n} X_k}{\sum_{k=1}^{n} Y_k} \\ &= \frac{X_n + \sum_{k=1}^{n-1} X_k}{\sum_{k=1}^{n} Y_k} \\ &= \frac{X_n + Q_n \sum_{k=1}^{n-1} Y_k}{\sum_{k=1}^{n} Y_k} \\ &= \frac{X_n + Q_n \sum_{k=1}^{n} Y_k - Q_n Y_n}{\sum_{k=1}^{n} Y_k} \\ &= Q_n + \frac{X_n - Q_n Y_n}{\sum_{k=1}^{n} Y_k} \\ &= Q_n + \frac{X_n - Q_n Y_n}{C_n} \end{aligned} \tag{5.23}$$

Using equations (5.13) to (5.23), we can write algorithm 3 to calculate *discounting-aware weighted importance sampling off-policy action-value estimation* as written in equation (5.12). □

Exercise 5.15. This is the same as the answer to exercise 5.6.

Ordinary importance sampling,

$$Q_\pi(s,a) \doteq \frac{\sum_{t\in\mathcal{T}(s,a)} \rho'_{t:T(s,a)-1} G_t}{|\mathcal{T}(s,a)|} \tag{5.24}$$

and *weighted importance sampling*,

$$Q_\pi(s,a) \doteq \frac{\sum_{t\in\mathcal{T}(s,a)} \rho'_{t:T(s,a)-1} G_t}{\sum_{t\in\mathcal{T}(s,a)} \rho'_{t:T(s,a)-1}} \tag{5.25}$$

where

$$\rho'_{t:T-1} \doteq \frac{p(S_{t+1}|s,a)\prod_{k=t+1}^{T-1}\pi(A_k|S_k)p(S_{k+1}|S_k,A_k)}{p(S_{t+1}|s,a)\prod_{k=t+1}^{T-1}b(A_k|S_k)p(S_{k+1}|S_k,A_k)} = \frac{\prod_{k=t+1}^{T-1}\pi(A_k|S_k)}{\prod_{k=t+1}^{T-1}b(A_k|S_k)} \tag{5.26}$$

□

Chapter 6

Exercise 6.1. Let us define

$$\delta_t \doteq R_{t+1} + V_t(S_{t+1}) - V_t(S_t) \tag{6.1}$$

We can write

$$\begin{aligned} G_t - V_t(S_t) &= R_{t+1} + \gamma G_{t+1} - V_t(S_t) + \gamma V_t(S_{t+1}) - \gamma V_t(S_{t+1}) \\ &= \delta_t + \gamma[G_{t+1} - V_t(S_{t+1}) + V_{t+1}(S_{t+1}) - V_{t+1}(S_{t+1})] \\ &= \delta_t + \gamma[G_{t+1} - V_{t+1}(S_{t+1})] + \gamma[V_{t+1}(S_{t+1}) - V_t(S_{t+1})] \\ &= \delta_t + \gamma\delta_{t+1} + \gamma^2[G_{t+2} - V_{t+2}(S_{t+2})] \\ &\quad + \gamma[V_{t+1}(S_{t+1}) - V_t(S_{t+1})] + \gamma^2[V_{t+2}(S_{t+2}) - V_{t+1}(S_{t+2})] \\ &= \sum_{k=t}^{T-1} \gamma^{k-t}\delta_k + \sum_{k=t}^{T-1} \gamma^{k-t+1}(V_{k+1}(S_{k+1}) - V_k(S_{k+1})) \end{aligned} \tag{6.2}$$

Exercise 6.2. In scenarios where we have reliable value estimates for some states and want to learn value estimates for other states, temporal difference learning can outperform Monte Carlo learning. This is due to utilizing existing estimates, a.k.a bootstrapping, in the process of temporal difference learning. For example, while commuting from a new house, it is very likely that we visit states similar to those when commuting from the old house. Temporal difference learning takes advantage of prior knowledge, resulting in more efficient learning. □

Exercise 6.3. In the first episode, all estimates are equal and the reward occurs only when transitioning from E to the right terminal state. Therefore, a change in the estimation of $V(.)$ only happens in the state before termination. Given that only $V(\mathsf{A})$ has changed, the episode ended in a left terminal state. We can express the transition from A to the left terminal state as follows,

$$\begin{aligned} V(\mathsf{A}) &= V(\mathsf{A}) + \alpha\delta_T \\ &= V(\mathsf{A}) + [R_T + V(\textit{left terminal}) - V(\mathsf{A})] \\ &= (1-\alpha)V(\mathsf{A}) \\ &= \frac{1-\alpha}{2} = 0.45 \end{aligned} \tag{6.3}$$

Therefore the change in estimation of $V(\mathsf{A})$ is $\alpha V(\mathsf{A}) = \frac{\alpha}{2} = 0.05$. □

Exercise 6.4. The graph suggests a direct relationship between α and the convergence rate in temporal difference learning, while an inverse relation exists between α and the final mean error. In contrast, there seems to be a potential convex quadratic relationship between α and the rate of convergence observed in Monte Carlo methods while the final mean error is unaffected. Based on the data presented, it is improbable that there exists a specific α value where the performance becomes equal in both methods. □

Exercise 6.5. Larger values of α result in a larger correction in the estimation of values. As value estimates converge to the real values, from some episodes onward, the large corrections cause oscillations around true values leading to an increase in the estimation error. Such an increase can happen for every α, although it may happen in significantly later episodes for smaller α values.

Although appropriate initial values can accelerate the convergence of the algorithm, their impact on converged values diminishes as episodes increase (as shown in the second chapter of the book). Therefore initial values cannot influence the occurrence of the phenomenon described in the question. □

Exercise 6.6. The first way is to Monte Carlo method to estimate the values.

The second way is to use dynamic programming to calculate values analytically. This is feasible since the system is simple, dynamics is known, and number of states are low.

In the book, the dynamic programming approach is used, since it only requires to solve the linear system of equations given in equation (6.4).

$$\begin{aligned}
V(\mathsf{A}) &= \frac{1}{2}[0 + V(\mathsf{B})] + \frac{1}{2}[0 + 0] = \frac{V(\mathsf{B})}{2} \\
V(\mathsf{B}) &= \frac{1}{2}[0 + V(\mathsf{A})] + \frac{1}{2}[0 + V(\mathsf{C})] = \frac{V(\mathsf{A}) + V(\mathsf{C})}{2} \\
V(\mathsf{C}) &= \frac{1}{2}[0 + V(\mathsf{B})] + \frac{1}{2}[0 + V(\mathsf{D})] = \frac{V(\mathsf{B}) + V(\mathsf{D})}{2} \\
V(\mathsf{D}) &= \frac{1}{2}[0 + V(\mathsf{C})] + \frac{1}{2}[0 + V(\mathsf{E})] = \frac{V(\mathsf{C}) + V(\mathsf{E})}{2} \\
V(\mathsf{E}) &= \frac{1}{2}[0 + V(\mathsf{D})] + \frac{1}{2}[1 + 0] = \frac{V(\mathsf{D}) + 1}{2}
\end{aligned} \tag{6.4}$$

□

Exercise 6.7. If the transition is produced from behavior policy, we need to use importance sampling to adjust the expected reward to be in accordance with the distribution of the target policy. Thus, we have

$$V(S_t) = V(t) + \alpha[\rho_{t:t}(R_{t+1} + \gamma V(S_{t+1})) - V(S_t)] \tag{6.5}$$

□

Exercise 6.8. We can write

$$\begin{aligned} G_t - Q_t(S_t, A_t) &= R_{t+1} + \gamma G_{t+1} - Q(S_t, A_t) + \gamma Q(S_{t+1}, A_{t+1}) - \gamma Q(S_{t+1}, A_{t+1}) \\ &= \delta_t + \gamma[G_{t+1} - Q(S_{t+1}, A_{t+1})] \\ &= \delta_t + \gamma\delta_{t+1} + \gamma^2[G_{t+2} - Q(S_{t+2}, A_{t+2})] \\ &= \delta_t + \gamma\delta_{t+1}b + \gamma^2\delta_{t+2} + \cdots + \gamma^{T-t-1}\delta_{T-1} + \gamma^{T-1}[G_T - Q(S_T, .)] \\ &= \delta_t + \gamma\delta_{t+1}b + \gamma^2\delta_{t+2} + \cdots + \gamma^{T-t-1}\delta_{T-1} + \gamma^{T-1}[0 - 0] \\ &= \sum_{k=t}^{T-1} \gamma^{k-t}\delta_k \end{aligned} \tag{6.6}$$

□

Exercise 6.11. asd