Chapter 5 solutions **Hanna Gábor**

1. *Consider the diagrams on the right in Figure 5.1. Why does the estimated value function jump up for the last two rows in the rear? Why does it drop off for the whole last row on the left? Why are the frontmost values higher in the upper diagrams than in the lower?*

   In the last two rows we already have 20 or 21 points and we stick with it. We win if the dealer goes bust or sticks before reaching 20, so we have a huge probability of winning.

   In the last row on the left, the dealer has an ace and that makes is easier to reach a high score for the dealer. If he goes over 21 with counting the ace as 11, he still has a chance with counting it as 1.

   The frontmost values in the upper diagram are higher than in the lower, because having a usable ace increases your chances, as mentioned above.

2. *Suppose every-visit MC was used instead of first-visit MC on the blackjack task. Would you expect the results to be very different? Why or why not?*

   I would expect them to be exactly the same. We can't reach the same state twice in a game, so the two methods are the same.

3. *What is the backup diagram for Monte Carlo estimation of q_π?*

   It's the same as for v_π on page 95, except for it starts from a state-action pair.

4. *The pseudocode for Monte Carlo ES is inefficient because, for each state–action pair, it maintains a list of all returns and repeatedly calculates their mean. It would be more efficient to use techniques similar to those explained in Section 2.4 to maintain just the mean and a count (for each state–action pair) and update them incrementally. Describe how the pseudocode would be altered to achieve this.*

   We can get rid of *Returns* and introduce $VisitCounts$ that stores an integer for each state-action pair. Instead of appending G, we would increase the corresponding element in $VisitCount$ by one and update $Q(S_t, A_t)$ to $(Q(S_t, A_t) \cdot (n-1) + G)/n$, where n is $VisitCount(S_t, A_t)$.

5. *Consider an MDP with a single nonterminal state and a single action that transitions back to the nonterminal state with probability p and transitions to the terminal state with probability 1 - p. Let the reward be +1 on all transitions, and let $\gamma = 1$. Suppose you observe one episode that lasts 10 steps, with a return of 10. What are the first-visit and every-visit estimators of the value of the nonterminal state?*

   If the estimator takes the average return for the corresponding state, then the first-visit estimation of the value of the nonterminal state is 10, the every-visit estimation is $(10 + 9 + ... + 1)/9 = 5.5$.

6. *What is the equation analogous to (5.6) for action values Q(s, a) instead of state values V(s), again given returns generated using b?*

$$Q(s,a) \doteq \frac{\sum\limits_{t \in \mathcal{T}(s)} \rho_{t+1:T(t)-1} G_t}{|\mathcal{T}(s)|}$$

7. *In learning curves such as those shown in Figure 5.3 error generally decreases with training, as indeed happened for the ordinary importance-sampling method. But for the weighted importance-sampling method error first increased and then decreased. Why do you think this happened?*

   With high probability, the importance sampling ratio will be 0 for the trajectory and thus the estimation will be 0 for the first few steps. The first time we choose a path that meets the target policy, we will win or lose with high probability, so the estimated value will be 1 or -1 until the next time we choose such a path. Altogether, after the initial zero estimations, we will most probably start with an estimation of 1 or -1, and will maintain that estimation for some time.

8. *The results with Example 5.5 and shown in Figure 5.4 used a first-visit MC method. Suppose that instead an every-visit MC method was used on the same problem. Would the variance of the estimator still be infinite? Why or why not?*

   Yes, the variance of the estimator would still be infinite:

$$
\begin{aligned}
\mathbb{E}_b\left(\left(\frac{1}{T}\sum_{k=0}^{T-1}\prod_{t=T-1-k}^{T-1}\frac{\pi(A_t|S_t)}{b(A_t|S_t)}G_0\right)^2\right) &= \sum_{l=0}^{\infty}\left(0.1\cdot 0.9^l\cdot\frac{1}{2^{l+1}}\cdot\frac{1}{l+1}\sum_{m=1}^{l}2^{2m}\right)\\
&= 0.1\sum_{l=0}^{\infty}\left(0.9^l\cdot\frac{1}{2^{l+1}}\cdot\frac{1}{l+1}\frac{4^{l+1}-4}{3}\right)\\
&= 0.1\sum_{l=0}^{\infty}\left(0.9^l\frac{4^{l+1}-4}{3(l+1)2^{l+1}}\right)\\
&= 0.1\sum_{l=0}^{\infty}\left(0.9^l\frac{2^{2(l+1)}}{3(l+1)2^{l+1}}-0.9^l\frac{4}{3(l+1)2^{l+1}}\right)\\
&= 0.1\sum_{l=0}^{\infty}\left(0.9^l\frac{2^{l+1}}{3(l+1)}-0.9^l\frac{4}{3(l+1)2^{l+1}}\right)\\
&= 0.1\sum_{l=0}^{\infty}\left(\frac{2\cdot 1.8^l}{3(l+1)}-\frac{2\cdot 0.45^l}{3(l+1)}\right)\\
&= 0.2\sum_{l=0}^{\infty}\left(\frac{1.8^l-0.45^l}{3(l+1)}\right)\\
&\geq 0.2\sum_{l=0}^{\infty}\left(\frac{1.8^l-1}{3(l+1)}\right)=\infty
\end{aligned}
$$

9. *Modify the algorithm for first-visit MC policy evaluation (Section 5.1) to use the incremental implementation for sample averages described in Section 2.4.*

Algorithm 1 Incremental implementation of first-visit MC policy evaluation

```
Input: a policy $\pi$ to be evaluated

Initialize:
$V(s) \in \mathbb{R}$, arbitrarily, for all $s \in S$
$Seen(s) \leftarrow 0$, for all $s \in S$

while true do
    Generate an episode following $\pi : S_0, A_0, R_1, S_1, \ldots, R_T$
    $G \leftarrow 0$
    for $t = T-1, T-2, \ldots, 0$ do
        $G \leftarrow \gamma G_t + R_{t+1}$
        if $S_t$ does not appear in $S_0, \ldots, S_{t-1}$ then
            $Seen(S_t) \leftarrow Seen(S_t) + 1$
            $n \leftarrow Seen(S_t)$
            $V(S_t) \leftarrow \frac{(n-1)V(S_t)+G}{n}$
        end if
    end for
end while
```

10. *Derive the weighted-average update rule (5.8) from (5.7). Follow the pattern of the derivation of the unweighted rule (2.3).*

$$\begin{aligned}
V_{n+1} = \frac{\sum\limits_{k=1}^{n} W_k G_k}{C_n} &= \frac{1}{C_n}\left(W_n G_n + \sum_{k=1}^{n-1} W_k G_k\right) \\
&= \frac{1}{C_n}\left(W_n G_n + C_{n-1}\frac{1}{C_{n-1}}\sum_{k=1}^{n-1} W_k G_k\right) \\
&= \frac{1}{C_n}\left(W_n G_n + C_{n-1} V_n\right) \\
&= \frac{1}{C_n}\left(W_n G_n + C_n V_n - W_n V_n\right) \\
&= V_n + \frac{W_n}{C_n}(G_n - V_n)
\end{aligned}$$

11. *In the boxed algorithm for off-policy MC control, you may have been expecting the W update to have involved the importance-sampling ratio $\frac{\pi(A_t|S_t)}{b(At|St)}$, but instead it involves $\frac{1}{b(A_t|S_t)}$. Why is this nevertheless correct?*

    π is deterministic, so $\pi(A_t|S_t)$ is either 0 or 1. We quit the inner loop if it is 0, so if the update happens, then $\pi(A_t|S_t) = 1$.

12. *Racetrack (programming)*

    I tried Monte Carlo control with the ϵ-soft greedy policy as behavior policy, but it did not work out (yet). Anyway, the code can be found here.

13. *Show the steps to derive (5.14) from (5.12).*

$$
\begin{aligned}
\mathbb{E}(\rho_{t:T-1}R_{t+1}) &= \mathbb{E}\left(\prod_{k=t}^{T-1}\frac{\pi(A_k|S_k)}{b(A_k|S_k)}R_{t+1}\right) \\
&= \sum_{a_t,\dots a_{T-1}}\left(\prod_{k=t}^{T-1}b(a_k|S_k)\prod_{l=t}^{T-1}\frac{\pi(a_l|S_l)}{b(a_l|S_l)}r(S_t,a_t)\right) \\
&= \sum_{a_t,\dots a_{T-1}}\left(\prod_{k=t}^{T-1}b(a_k|S_k)\frac{\pi(a_k|S_k)}{b(a_k|S_k)}r(S_t,a_t)\right) \\
&= \sum_{a_t,\dots a_{T-1}}\left(\prod_{k=t}^{T-1}\pi(a_k|S_k)r(S_t,a_t)\right) \\
&= \sum_{a_t,\dots a_{T-1}}\left(\pi(a_t|S_t)r(S_t,a_t)\prod_{k=t+1}^{T-1}\pi(a_k|S_k)\right) \\
&= \sum_{a_t}\pi(a_t|S_t)r(S_t,a_t)\sum_{a_{t+1},\dots a_{T-1}}\prod_{k=t+1}^{T-1}\pi(a_k|S_k) \\
&= \sum_{a_t}\pi(a_t|S_t)r(S_t,a_t)\cdot 1 = \\
&= \sum_{a_t}b(a_t|S_t)\frac{\pi(a_t|S_t)}{b(a_t|S_t)}r(S_t,a_t) = \\
&= \mathbb{E}(\rho_{t:t}R_{t+1})
\end{aligned}
$$