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 $VisitCount(S_t, A_t)$.