A4Q1: Manual Value Iteration

Bodhi Nguyen

January 27th, 2021

1 Problem

Consider the simple MDP with three states (one terminal) and two actions as described in assingment 4. Perform two iterations of value iteration and then argue why any more iterations will not yield a different policy.

2 Manual Value Iteration

In each step of value iteration, we use the value function from the previous step and the transition probabilities to calculate the action-value function for the next iteration. We then maximize over actions to get the new policy for the next iteration.

We start with $v_0(s_1) = 10.0$, $v_0(s_2) = 6.0$, $v_0(s_3) = 0.0$.

In general,

$$Q_k(s,a) = R(s,a) + \gamma * \sum_{s' \in S} P(s', a, s) * V_{k-1}(s')$$

$$V_k(s) = \max_{a \in A}(Q_k(s,a))$$

So,

$$q_1(s_1, a_1) = 8.0 + 1 * (0.2 * 10.0 + 0.6 * 1.0 + 0.2 * 0) = 10.6$$
$$q_1(s_1, a_2) = 10.0 + 1 * (0.1 * 10.0 + 0.2 * 1.0 + 0.7 * 0) = 11.2$$
$$q_1(s_2, a_1) = 1.0 + 1 * (0.3 * 10.0 + 0.3 * 1.0 + 0.4 * 0) = 4.3$$
$$q_1(s_2, a_2) = -1.0 + 1 * (0.5 * 10.0 + 0.3 * 1.0 + 0.2 * 0) = 4.3$$

$$V_1(s_1) = 11.2, V_1(s_2) = 4.3, V_1(s_3) = 0.0$$

$$q_2(s_1, a_1) = 8.0 + 1 * (0.2 * 11.2 + 0.6 * 4.3 + 0.2 * 0) = 12.82$$
$$q_2(s_1, a_2) = 10.0 + 1 * (0.1 * 11.2 + 0.2 * 4.3 + 0.7 * 0) = 11.98$$
$$q_2(s_2, a_1) = 1.0 + 1 * (0.3 * 11.2 + 0.3 * 4.3 + 0.4 * 0) = 5.65$$
$$q_2(s_2, a_2) = -1.0 + 1 * (0.5 * 11.2 + 0.3 * 4.3 + 0.2 * 0) = 5.89$$

$$V_2(s_1) = 12.82, V_2(s_2) = 5.89, V_1(s_3) = 0.0$$

From here, the optimal policy remains to take action 1 in state 1 and action 2 in state 2. This is because we can see on further iterations, since $\gamma = 1$ we never discount the future, so further updates into the future will increase the value of each action in general. However, we know the difference in the action-value functions in state 1 between action 1 and action 2 is equal to

$$-2.0 + (0.2 - 0.1) * V_{k-1}(s_1) + (0.6 - 0.2) * V_{k-1}(s_2)$$

Similarly, the difference between the second action and first action's action-value function in the second state is

$$-2.0 + (0.2) * V_{k-1}(s_1)$$

which is greater than zero when the value of the first state is greater than 10.

After the second iteration the difference for state 1 is greater than zero, so action 1 gets chosen. Similarly the second action for state 2 gets chosen. The action-value function for (s_1, a_1) in the next iteration is thus equal to

$$q_k(s_1, a_1) = 8.0 + (0.2 * q_{k-1}(s_1, a_1) + 0.6 * q_{k-1}(s_2, a_2))$$

which is weakly increasing w.r.t to the action value functions in the previous iterations whenever $q_{k-1}(s_1, a_1) >= 10$ and the $q_{k-1}(s_2, a_2) > 0$. So we know in iteration 3, action 1 gets chosen for state 1 and the action-value function for state 1 action one is weakly increasing. This in turn implies that action 2 will be chosen by state 2. The last thing we need a guarantee of is that $q_k(s_2, a_2) > 0$ on each iteration, which is true since $0.5 * V_{k-1}(s_1)$ term offsets the -1.0 that appears as long as $V_{k+1} > 2$. This also guarantees that the action value function for state 2 action 2 will be greater than 5.41 going forward (not counting the contribution from state 2's value). This is enough to guarantee that action 1 gets chosen in each iteration.

In summary, on each iteration going forward, we know that in the previous iteration, action 1 was chosen for state 1 and action 2 was chosen for state 2. The Q function for state 1 action 1 has weakly increased in this iteration, which implies that the second action for state 2 gets chosen. We also know action 1 will be chosen in this iteration since the Q function for state 2 action 2 in the previous iteration is guaranteed to be above 5.41 by the weakly increasing property of the Q for state 1 action 1. So the policy stays the same going forward.