CMSC 389F: Lecture 5

Intro: We now know how to evaluate how good a deterministic policy is (just trace over the steps and rewards and discount). Today we'll look at figuring out how to evaluate how good certain states and actions are. Then, we'll show that if we know which actions are best, we can just take those actions.

Value Functions: It tell us how good certain states and actions are, following a certain policy. How good it is to be in this state is dependent on what I do after I leave that state. This is what the policy tells us, and explains why each value function is defined in terms of a specific policy. Remember that value is the total expected reward, not just the immediate reward.

There are two types of Value Functions:
Action Value Function; written as Q(state, action). It tells you how good is it to take a particular action from a particular state. We care about this when we want to decide what action to take, picking the action with highest action value. It is equivalent to the expected return (aka cumulative reward) after taking a particular action from a particular state, following policy pi.

The Action Value Function can be defined in two parts summed together:

1. The immediate reward of being in your current state s and taking action a
2. The discounted sum of state values of the successor states s' you can reach by taking action a, weighted by the probability you will go to s' by taking action a from state s

So, for each state s' that you may end up in by taking action a, you sum the state value for that state, and weigh it by the probability of you going to that state by taking action a. We weigh by probability so that the value of states that we are most likely to end up in are counted heavily.

$$q_\pi(s,a) = \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_\pi(s')$$

State Value Function: written as V(state). Shows how good it is to be in state s following policy pi. The state value tells us the cumulative reward you will receive after being in that state.

The state value is equivalent to the average of action values of all possible actions given by the policy, weighted by the probability of taking action a given by the policy you are following.

$$v_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) q_\pi(s,a)$$

As you can see, the state value function and value function are written in terms of each other!

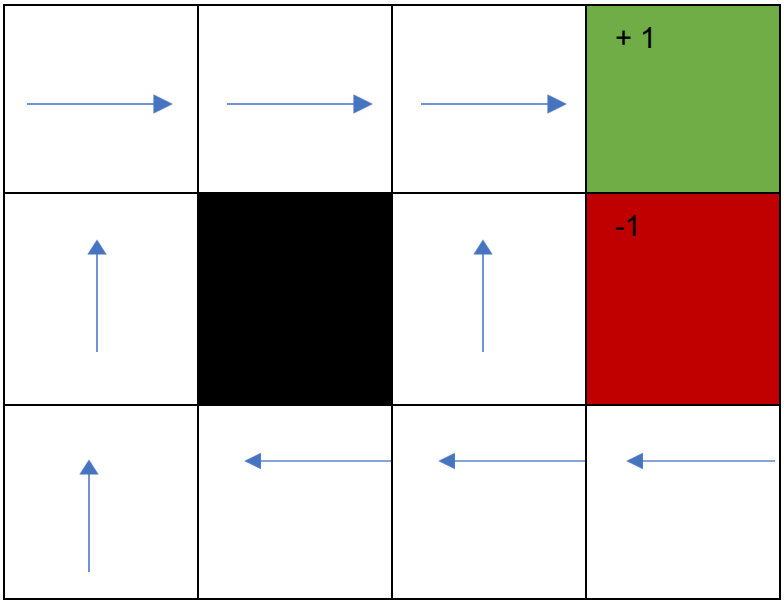
Value Function Example:

Environment Details:

- Reward: -0.04 for every action. Discount Factor (Gamma) of 1.
- If you take an action, there is an 80% chance of going in the desired direction, 10% chance of going left in relation to the desired direction, 10% chance of going right in relation to the desired direction. (Ex: if you are intending to move right, you have an 80% of going right, 10% chance of going up ,since up is left relative to moving right, and 10% chance of going down)
- Policy is deterministic: given a state, it returns a single action to take
- The black cell is a wall, it is not a state you can visit
- There are two terminal states: green state with reward +1 and a red state with -1 reward
- Each state is numbered, shown in first figure. These numbers do not represent rewards

7	8	9	+ 1
5		6	-1
1	2	3	4

The value functions defined next follow the policy below:



State Action Values

0.812	0.868	0.918	+ 1
0.762		0.660	- 1
0.705	0.655	0.611	0.388

Let's look at the state value for the state 9 to the left of the goal state, that has a value of .918. From our policy, we know the action we should take is to go right.

To verify this value, we can average over all the possible states we can go to from following the policy and taking the action of moving right. There are three states we can end up in: the goal state, with P = .8, the state with value .660 with P = .1, and bouncing back to the current state again, with value of .918 and P = .1.

$$v_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) q_\pi(s, a)$$

Following our equation for state value function (shown again above), our policy tells us to go right 100% of the time, so all that is left is to find Q(9, right). To find Q(9, right), we use our action value function:

$$q_\pi(s, a) = \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_\pi(s')$$

We know the immediate reward of taking any action is -0.04. Next, we must go through all possible states we could end up in, and sum their action values multiplied by their probabilities, which we listed above. Finally, we add our immediate reward of -0.04:
(1 * .8 + .660 * .1 + .918 * .1) - 0.04 = .918

So what did we do just there? We actually just defined the state value function in terms of the action value function!

Each possible action our policy can tell us to take has a q value from its starting state, along with a probability of taking that action given by our policy (in the case of a deterministic policy, there is only a single action to take, with a probability of 1).

In this case, our policy is deterministic, so it only tells us a single action to take, with a probability of 1. What we did just now was calculate the q value of taking the action given by the policy.

We went through all the states that the action of moving right could take us, and then weighed their values by the probability of ending up in those states by moving right.

So if our policy was stochastic, and told us to go left 50% of the time and right 50% of the time, the state value would equal q(s, left) * .5 + q(s,right) * .5. We just calculated q(s,right) to be equivalent to .918.

Lets see another example, this time for Q(9,down):
Q(9, down) = (.660 *8 + .868 * .1 + .918 * .1) - 0.04 = 0.675 (calculated using the state values)

Next Lecture:

- Define Optimal State and Action Value Functions
- While it may not be immediately clear how value functions can be useful when we are already given the state values (as in this example), we will see next lecture how we can come up with these values ourselves. This is done through a process called **Value Iteration.**