5. Policy and Value Functions
Policy and Value Functions

Start of transcript. Skip to the end.

The next thing that I need to introduce to you before we can

move to the algorithm-- you can see we're doing a lot

of introduction, but this, I promise, is the last one--

we need to talk about policy, OK?

So what is policy?

Policy is an optimal action that you can take in a state.

OK?

Video
Download video file

Transcripts
Download SubRip (.srt) file
Download Text (.txt) file

Definition of Optimal Policy

1/1 point (graded)
Given an MDP, and a utility function $U\left[s_0, s_1, \ldots, s_n\right]$ „ our goal is to find an optimal policy function that maximizes the expectation of the utility. Here, a **policy** is a function $\pi : S \rightarrow A$ that assigns an action $\pi\left(s\right)$ to any state s. We denote the optimal policy by π^*.

Which of the following options are correct about the optimal policy function?

- The optimal policy function would only depend on the state and action space but is independent of the reward structure.
- The optimal policy assigns an action at every state that maximizes the expected utility. ✔
- For any given state, the optimal policy function should always take an action that results in the best expected immediate reward for that state.
- A policy function specifies an action for every different sequence of states starting from the initial state until the current state.

Solution:

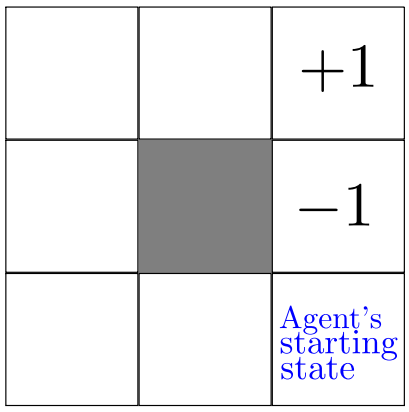
The goal of the optimal policy function is to maximize the expected discounted reward, even if this means taking actions that would lead to lower immediate next-step rewards from few states.

A policy function is a mapping from a state to an action (or a probability distribution over the set of actions). Under our current markovian setting, it does not matter what sequence of states where visited in the past to determine what the policy should be from a given state.

Submit You have used 1 of 2 attempts

Answers are displayed within the problem

MDP Example: Negative Living Reward



Recall the MDP example in the lecture. An AI agent navigates in the 3x3 grid depicted above, where the middle square is not accessible (and hence is greyed out).

The MDP is defined as follows. As before, every state s is defined by the current position of the agent in the grid. The actions are the 4 directions “up", “down",“left", “right".

Now, The transition probabilities from state s via action a to state s' is given by $T(s,a,s') = P(s'|s,a)$.

Reward structure:

As before, the agent receives a reward of $+1$ for arriving at the top right cell, and a reward of -1 for arriving in the cell immediately below it. It does not receive any non-zero reward at the other cells as illustrated in the following figure.

However, this time, the agent also receives a reward (or penalty) of -10 for every action that it takes, including any action that leads the agent into the $+1$ or -1 cells.

Transition Probabilities:

For simplicity, assume that all the transitions are deterministic. That is, given any state s, the outcome of all actions are deterministic: The next state reached is completely predictable.

For intance, taking the action “left" from the bottom right cell will always take the agent to the cell immediately to its left. Any action pointing off the grid would lead the agent to remain in its current cell.

Initial State:

Also, assume that the agent always starts off from the bottom right corner of the grid. It continues to take action until it reaches the top right corner, at which point it stops and does not act anymore.

Optimal policy - Numerical Example

2/2 points (graded)
Recall that in this setup, the agent receives a reward (or penalty) of -10 for every action that it takes, on top of the $+1$ and -1 when it reached the corresponding cells. Since the agent always starts at the state s_0, and the outcome of each action is deterministic, the discounted reward depends only on the action sequences and can be written as:

$$U[a_1, a_2, \ldots] = R(s_0, a_1) + \gamma R(s_1, a_2) + \gamma^2 R(s_2, a_3) + \cdots$$

where the sum is until the agent stops.

For the cases $\gamma = 0$ and $\gamma = 0.5$, what is the maximum discounted reward that the agent can accumulate by starting at the bottom right corner and taking actions until it reached the top right corner?

(Remember the negative reward -10 is applied to any action taken, including one that leads the agent to the -1 or $+1$ cells.)

For $\gamma = 0$:

Maximum discounted reward: -10 ✔ **Answer:** -10

If $\gamma = 0.5$:

Maximum discounted reward: -11 - 9/2 ✔ **Answer:** -15.5

Solution:

If $\gamma = 0$, then the discounted reward is the same as the reward received after the first step. The agent could receive a reward of -10 if it choose any one of the actions “Left", “Down", “Right". It would receive an additional reward of -1 if it selects to go up. So, the best discounted reward under this condition would be -10.

If $\gamma = 0.5$, then the best discounted reward would occur for the following sequence of actions starting from the initial state: “Up", “Up". It would recieve rewards of $-11, -9$ for these two actions respectively. The agent would reach the top right state and come to a complete halt after taking the above sequence of actions. The discounted reward amounts to $-11 + 0.5 * -9 = -15.5$

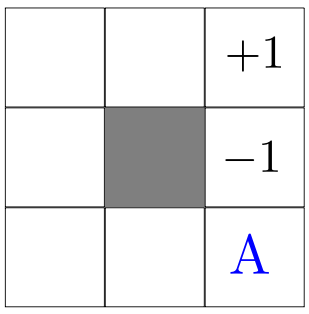
Submit You have used 2 of 3 attempts

Answers are displayed within the problem

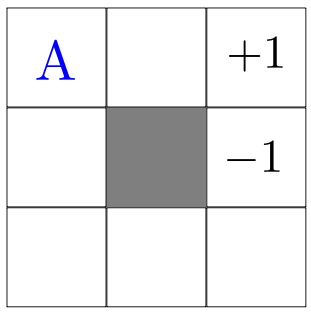
Value Function

1/1 point (graded)
As above, we are working with the 3×3 grid example with $+1$ reward at the top right corner and -1 at the cell below it. The agent also gets a reward of -10 for every action that it takes. The action outcomes are deterministic. The agent continues to act until it reaches the $+1$ cell, when it stops.

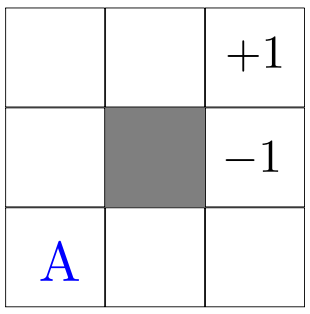
The following figures show states s_1, s_2, s_3, in which the letter “A" marks the current location of the agent.



s_1



s_2



s_3

A **value function** $V(s)$ of a given state s is the expected reward (i.e the expectation of the utility function) if the agent acts optimally starting at state s. In the given MDP, since the action outcome is deterministic, the expected reward simply equals the utility function.

Which of the following should hold true for a good value function $V(s)$ under the reward structure in the given MDP?

Note: You may want to watch the video on the next page before submitting this question.

- $V(s_1) < V(s_2) < V(s_3)$
- $V(s_3) < V(s_2) < V(s_1)$
- $V(s_3) < V(s_1) < V(s_2)$ ✔

Solution:

Note that the agent in states s_2 could reach the target in just 2 steps in both of these cases and for a total reward of $-10 * 2 + 1 = -19$.

The agent in s_1 is also at a distance of 2 steps away from the target but it has to pass through the cell with -1 reward before it can reach the target in 2 steps. Thus it can reach the target with for a best total reward of $-10 * 2 - 1 + 1 = -20$.

The agent in the state s_3 is at least 4 steps away from the target making it incur a minimum penalty of 40 before it can even reach the target.
Hence the right order of the value functions is

$$V(s_3) < V(s_1) < V(s_2).$$

Submit You have used 1 of 2 attempts

Answers are displayed within the problem

Discussion

Show Discussion

Topic: Unit 5 Reinforcement Learning (2 weeks) :Lecture 17. Reinforcement Learning 1 / 5. Policy and Value Functions