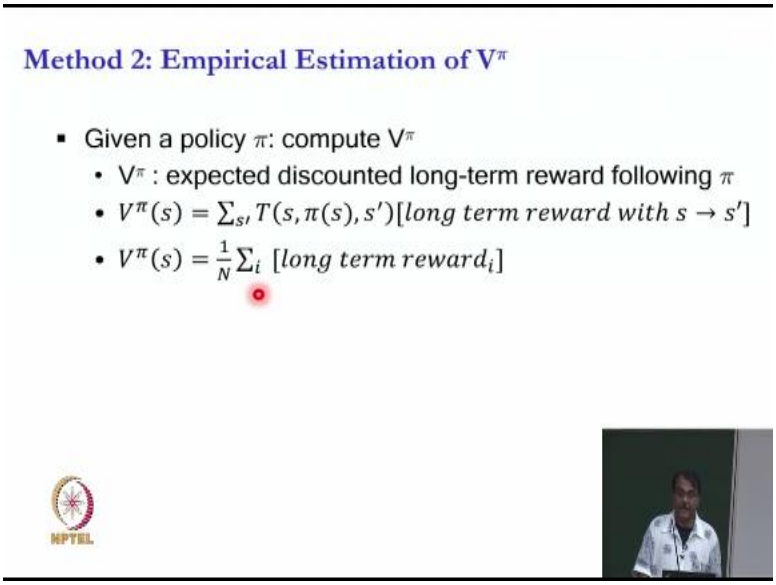
Artificial Intelligence
Prof. Mausam
Department of Computer Science and Engineering
Indian Institute of Technology-Delhi

Lecture - 78
Reinforcement Learning: Model-free Learning for Policy Evaluation (Passive Learning)

Model free learning we will do two methods, okay.

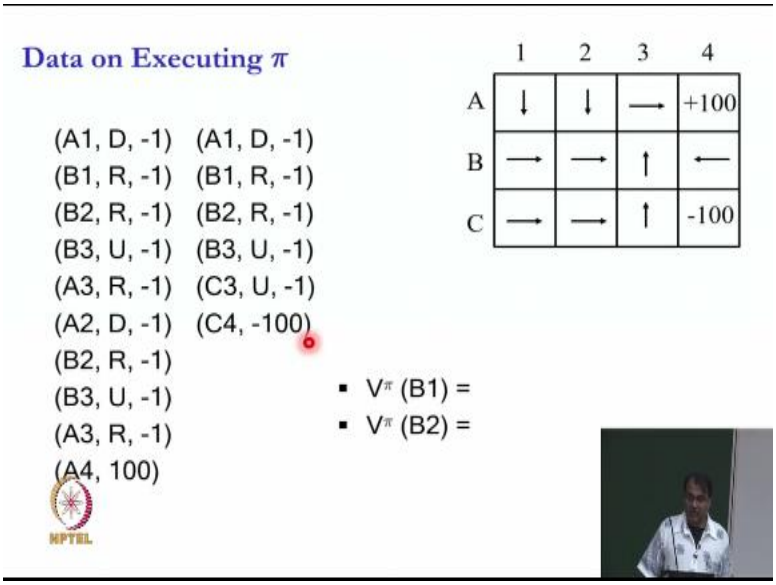
(Refer Slide Time: 00:23)



Now our goal is still the same. Given a policy pi compute V pi. Now V pi can be thought of as expected discounted long-term reward following pi. You can write it down as summation s prime the transition from s to s prime long-term reward with the transition s to s prime, right. Immediate reward plus long-term reward. I have just written it down like this right, r plus the reward starting in s prime.

Now whenever you are given an expectation you can compute that expectation by, we just did you can compute the expectation model free by average. So I can write this down as 1 by N of all the various samples I have, summation i long-term reward i. That is it. This is empirical estimation of the V pi.

(Refer Slide Time: 01:26)

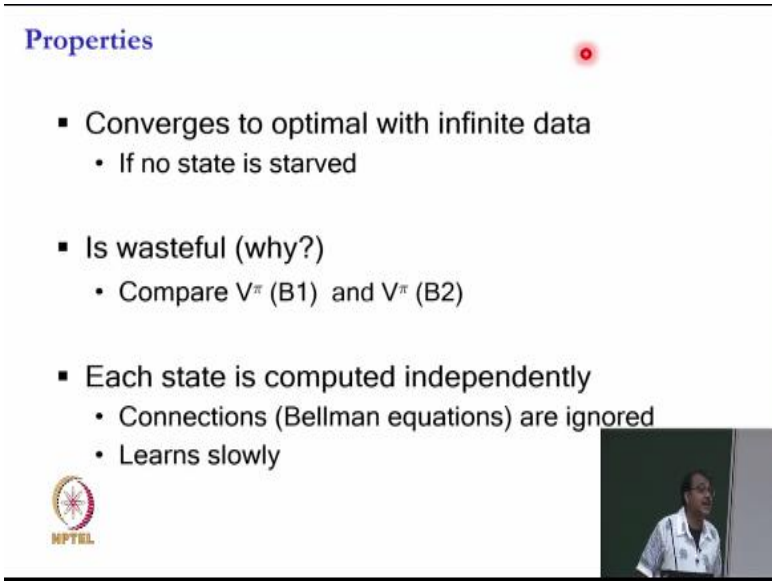


So again, let us look at this example. We are given the same example. Let us say I ask you what is the estimated V pi of B1, discount factor is 1. What is the estimated V pi of B1? So what would you do? Okay, I will give you 30 seconds to compute. What is V Pi of B1? And similarly what is V pi of B2? So what is the denominator of V pi of B1? Two, how many times do I reach B1, twice?

The first time when I reach B1, what is the total reward I get, 92 right? -1, -1, -1, -1, -1, -1, -1, -1. So that is -8 plus 100, 92. The second time I reach B1 what is the reward I get? – 104. So my average my V pi of B1 is -6. However, what about V pi of B2? How many times do I reach B2? Three times. So my denominator would be 3 and what will be the numerators.

In one case I get 93, in one case I get 97 **97** and in the third case I get -103. That gives me a V pi of B2 to be 29.

(Refer Slide Time: 03:07)



Now. This also is correct. This also converges to optimal with infinite data as long as no state is starved. By the way this convergence will always require things to not be starved because if I do not even get to somewhere then I would not know what is the value right? So I have to make sure that every state is visited infinitely often and so on so forth. As long as no state is starved, I will always converge to optimal with infinite data.

But it is a wasteful way of doing this. And the reason it is wasteful, I mean it is easy and everything but it is wasteful because I am directly estimating it. I am sort of ignoring the fact that these states are interlinked with each other. If you think about it V pi of B1 was -6. Whereas V pi of B2 was 29. Does that make sense?

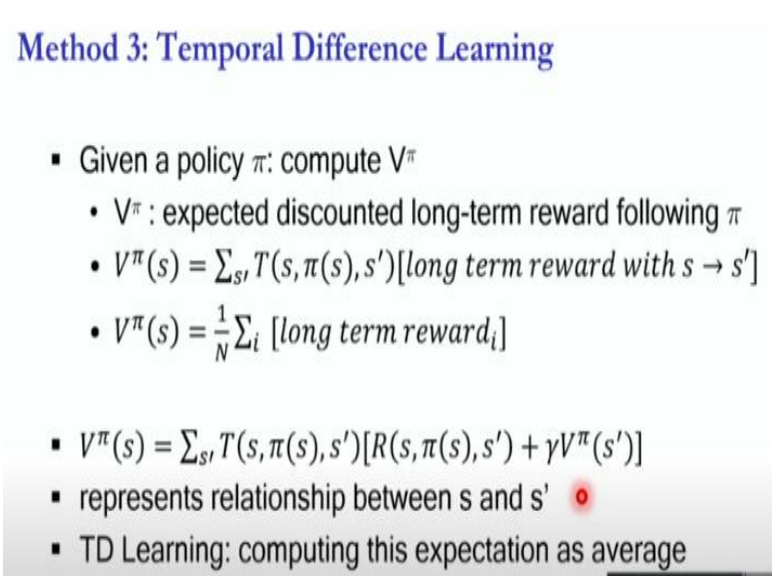
It is given the fact that I am going to take action right in B1 and given the fact that you know every time I take action right in B1, I get to B2 at least twice I have done it and twice that has happened and that action only costs -1. What do you expect? You expect V pi of B1 and V pi of B2 to be approximately within one of each other. This is the internal consistency that you would want from your algorithm.

But we do not have that. Right, but discount factor is 1. So discounting is not really adding that much problem. Again, I am not saying that this will exactly be the case. But it is highly unlikely that the correct answer is V pi of B1 -6 and V pi of B2 29. A difference of 35 is unusual given what we know about the domain right now. We do not know everything of course.

And the reason that is happening is because we got to know more about B2. And therefore we you know computed its expectations slightly differently, but from the point of view of B1, we did not pass on that information. We are computing each V pi's completely sort of independently and that is not very good. We need to think about their connections, even though we are not given the transition function and sort of use Bellman equations.

Because if we do not do this we might learn very slowly. So eventually this is going to converge with infinite data if no state is starved but the sample complexity the number of samples it will take will be very high. So let us do better. And let us try to do better and the algorithm that allows us to do better is called the temporal difference learning algorithm TD learning.

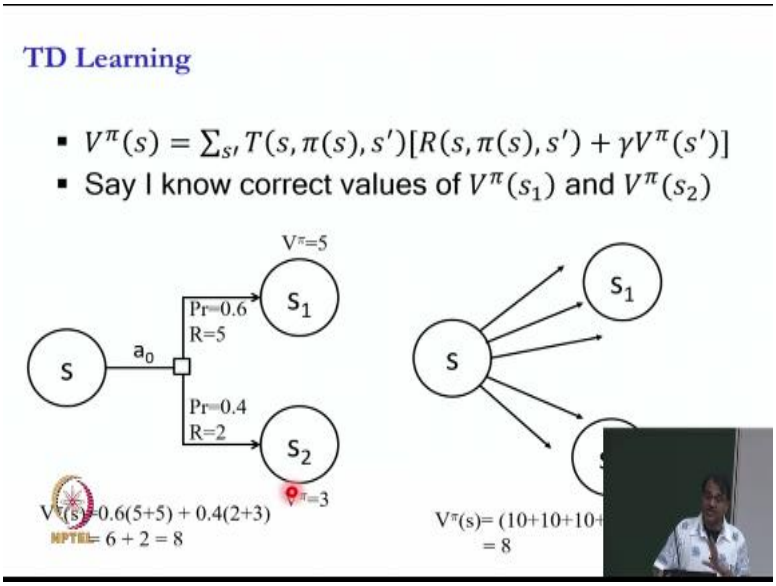
(Refer Slide Time: 06:05)



This is a model-free learning approach which somehow utilizes the Bellman equations, okay. And so let us see if you can follow this. So this is where we were. We were interested in 1 by N summation i long-term reward i but instead of doing this we can alternatively because this does not allow us to look at s and s prime in the same equation. So let us instead use the other equation. V pi of s is expectation of R the immediate reward plus gamma times v pi of s prime.

Now this equation represents the relationship between s and s prime. And again as with every algorithm in this set of lectures expectation is nothing but average over the samples.

(Refer Slide Time: 06:56)

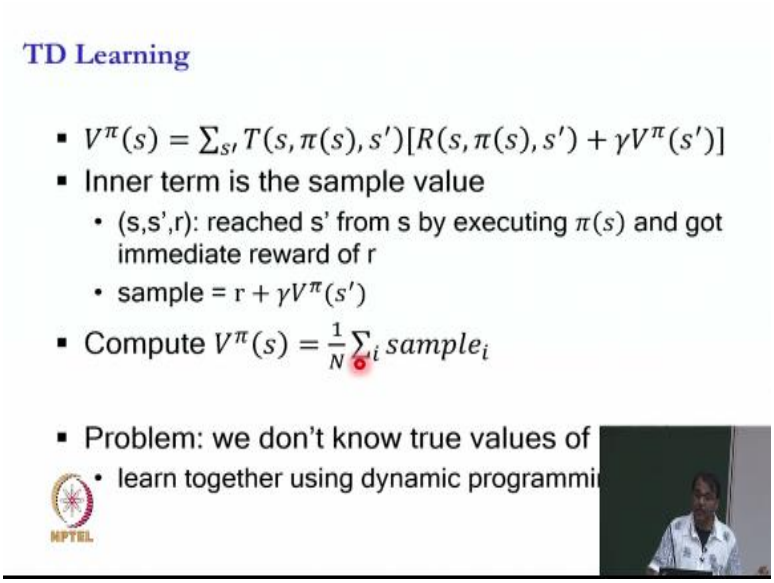


So let us say I am in this world and this is what was given to me so I can estimate V pi to be 0.6. Suppose V pi of S1 and S2 are correct. So let us say 5 and 3 are the correct values. So I can say that V pi of s is nothing but 0.6 times 5 plus 5 plus 0.4 times 2 plus 3. That gives us 6 + 2, 8. But this is not how we are going to compute. We will do it model-free. So we will take samples.

And when we take samples the first time we make it S1. The second time we make it S1. The third time we make it S2. The fourth time we make it S1. The fifth time we make it S2. Over time you will find that you will get S1 to S2 ratio 3:2 because its probabilities are 0.6 to 0.4. And we know that V pi of S1 is 5 V pi of S2 is 3 and the cost that we pay when we go to S1 is 5 and the cost we pay when we go to S2 is 2.

So we can alternatively figure out that three times we got a reward of sorry not cost three times we got the reward of 10 and 2 times we got the reward of 5. So V pi estimated as an average would be 10 + 10 + 10 + 5 + 5 divided by 5, which is also going to be 8. And what are the equations that I am using?

(Refer Slide Time: 08:13)



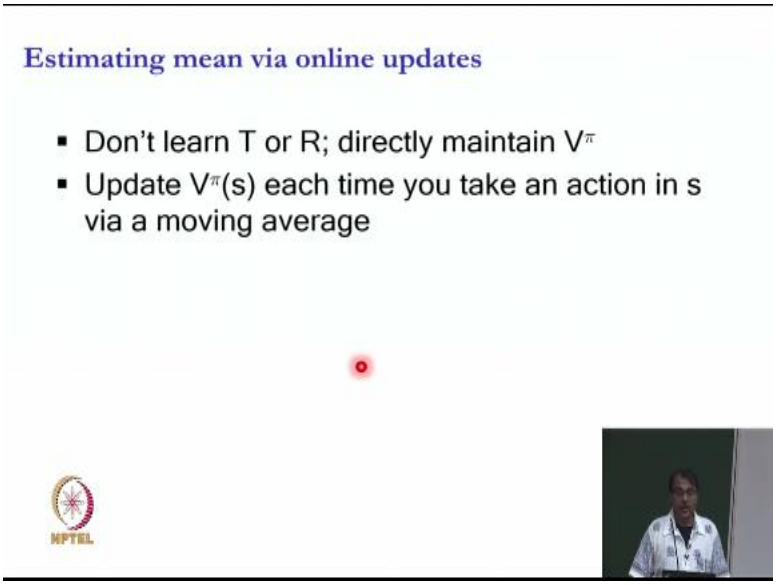
Well what I am saying is that I am going to take this particular expectation and compute it as an average. So for that I will do average of this term in the inner term. The inner term is my sample value. So let us say I was in state s. I took the action pi s. I reached s prime got an immediate reward of r. Then my sample value is going to be r plus gamma times v pi of s prime.

And my V pi of s is nothing but 1 by N summation i sample i. So again, what is the story? The story is that we have this particular equation that we want to estimate by sampling. We recognize that this equation is a expectation. We say great, we know how to do with expectations. We start sampling with this distribution the T distribution and we just take average of the inner term. What is the average of the inner term?

The immediate reward that we get plus the gamma times V pi of s prime the s prime that we reach and that is called sample i. This is the ith time we sample. This is what my sample i is and I take an average of those and that gives me V pi of s. Now of course, there is a problem. We do not know true values of V pi s prime. That is okay. We are just going to do dynamic programming.

So we will just maintain current version of V pi s and then we will be in some state s and we will take the policy. We will get to s prime. Then we will compute the sample value and change the value of that particular s and nothing else.

(Refer Slide Time: 09:57)



So this is module free because we are not learning T or R notice. We do not care for what is the transition function. We do not care for what is the reward function. We are directly estimating V pi. Every time I am in s and I take an action I update V pi of s using the old value of V pi of s prime. This algorithm is called the temporal difference learning. We will talk about it more in the next class starting from this point.

Okay, so we will have the V pi equation. We would have computed average of samples and we will understand why this is correct. And what are the implications of this method? And then how do we solve the full-blown control problem the active version where we do not even know which action to take. We have to figure out which action to take and also compute the optimal policy at the same time. Let us stop here. Thanks.