CLASSTIME | Page No.
Date / /

RL - A3

Abhishek Maiti
2016005

Q1. To make the update function efficient, we need a step update function

$$\theta_n \triangleq (S_t, A_t) = \frac{1}{n}\sum_{i=1}^{n} G_i$$

$$= \frac{1}{n}\left(G_n + \sum_{i=1}^{n-1} G_i\right)$$

$$= \frac{1}{n}\left(G_n + (n-1)\cdot\frac{1}{(n-1)}\sum_{i=1}^{n-1} G_i\right)$$

$$= \frac{1}{n}\cdot\left(G_n + (n-1)\,\theta_{n-1}\right)$$

$$= \frac{G_n}{n} + \theta_{n-1} - \frac{\theta_{n-1}}{n}$$

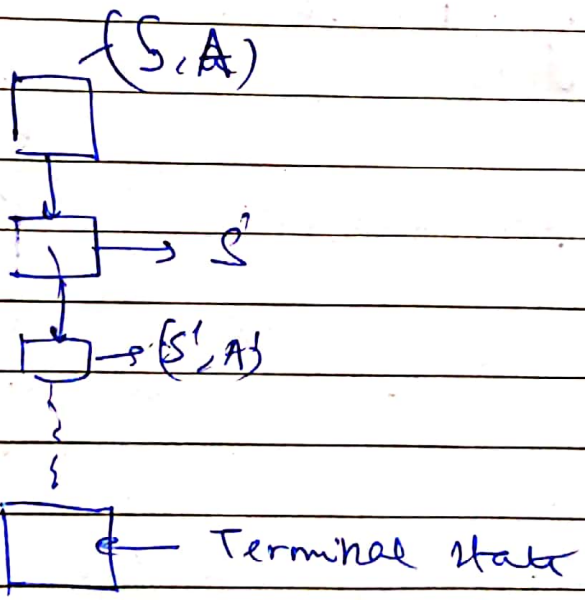
$$= \theta_{n-1}(S_t, A_t) + \frac{1}{n}\left(G_n - \theta_{n-1}(S_t, A_t)\right)$$

Now we only need to maintain the count of each time considering (S_t, A_t)

We don't need to keep track of the previous updates.

Pseudo code done at the end

Q2 Ans: Back up diagram for $q_\pi(s, a)$, since



Monte-carlo only follows one particular episode, there are no other branches.

Q3 Ans:

Let

Suppose all the ~~first~~ time steps in which state 's' is visited and action 'a' is taken be denoted by $f(s,a)$.

Then $Q(s,a) = \frac{\sum_{t \in f(s,a)} \ell_{t+1:T-1} G_t}{\sum_{t \in f(s,a)} \ell_{t+1:T-1}}$

where G_t are the returns and $\ell_{t+1:T-1}$ is the importance sampling parameter.

~~Starting from the first~~

CLASSTIME | Page No.
Date / /

Q5 Ans:

When we coming from a new building, but the part after highway remains same, then we just need to update the portion before reaching the highway as rest of the estimates remain same.

Q6 Ans: Even if the action is greedy, Q-learning is different from Sarsa because in Q-learning, we change the value (update it) and then select the action in the next time step but in Sarsa, we ~~first~~ choose the action based on old values and then update it.

Q6: Ex. 6.3

Given that $\alpha = 0.1$, and $\gamma = 1$

We know that:

$$V(S_t) = V(S_t) + \alpha\left[R_{t+1} + V(S_{t+1}) - V(S_t)\right]$$

We know that all the transitions give 0 reward apart from when going to transition state.

Also, we initialised our value function with 0.5.

When we ended up in the left terminal state we got reward 0. But we were supposed to get

0.5

So the update looks like.

$$V(A) := V(A) + 0.1[0 + 0 - 0.5]$$

$$:= 0.9\, V(A)$$

$$:= 0.9 \times 0.5$$

$$:= 0.45$$

So the new value of the left terminal state is 0.45 and decreased by 0.05.

For all the other states

$$\alpha[R_{t+1} + V(S_{t+1}) - V(S)$$

is 0.

hence no updates.

CLASSTIME | Page No.
Date / /

Exercise 6.5

'α' can be seen as the jump parameter or the step-size, if we overshoot the optimal estimation, it tries to come back hence the observed ups and down in the graph.

Exercise 6.4

Yes, if the α's are bigger we will get results for TD(0) method which are worse performing than ~~MC~~ monte carlo. This is because, 'α' represents the jump size, and if the jump size is large it is harder to optimize.

Q3 Pseudo code

Input : a policy π to be evaluated

Initialize:

$V(s) \in \mathbb{R}$, arbitrarily, for all $s \in S$

Returns(s) $\leftarrow$ an empty list for all $s \in S$

Loop Forever (for each episode):

Generate an episode π : $S_0, A_0 \ldots$

$G \leftarrow 0$

Loop for each step of episode, $t = T-1, \ldots, 1$

$G \leftarrow \gamma G + R_{t+1}$

Unless S_t appears in $S_0, S_1 \ldots S_{t-1}$:

$$V(S_t) \leftarrow V(S_t) + \frac{1}{n}\left(G - V(S_t)\right)$$