Ashish Sethi
MT18024

Date: / / /

RL - Assignment - 3

Q1. Ex - 5.4.

Initialize:

$\pi(s) \in A(s)$, for all $s \in S$.
$Q(s,a) \in \mathbb{R}$ for $s \in S$, $a \in A(s)$
returns $(s,a) \leftarrow$ empty list $s \in S$, $a \in A(s)$.

Loop forever (for each episode).

Choose $S_0 \in S$, $A_0 \in A(S_0)$ randomly such that all pairs probability > 0.

Generate an episode from S_0, A_0 following π: $S_0, A_0, R_1, \ldots S_{T-1}, A_{T-1}, R_T$.

$G \leftarrow 0$

Loop for each step of episode $t = T-1, T-2, \ldots 0$

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

Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1, \ldots S_{t-1}, A_{t-1}$

Date / / /

$$mean_t \; count_t = Returns(S_t, A_t)$$

$$\text{New mean} = \frac{mean_t \times Count_t + G_t}{Count_t}$$

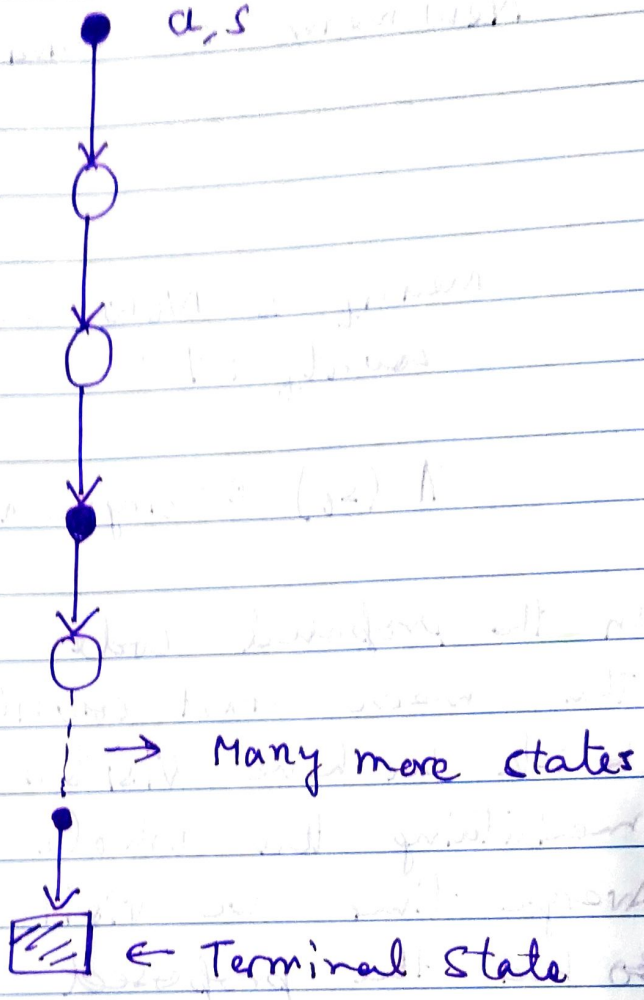
$$Mean_t = \text{New mean}$$
$$count_t + +$$

$$\pi(S_t) \leftarrow \arg\max_a Q(S_t, a)$$

In the proposed code we are just maintaing the mean and count of total no. of terms which we have visited till it some as meintaing the whole list and calculating Avarge time we visit the state again. so but the proposed code requires less memory as compared to the original one.

Date: / / /

Q 2 Exercise 5.3



Q 3 Exercise 5.6

$$q(s,a) = \frac{\sum_{b \in \tau(s,a)} \rho_b : T(t)-1 \; G_t}{\sum_{b \in \tau(s,a)} \rho_b : T(t)-1}$$

Date / / /

Q5 Ex - 6.2

Considering the hint which is mentioned in the question if we have lot of driving experience and suddenly we shift to new building and new parking lot.

In this scenario TD will perform better than the Monte carlo estimation because of the following reasons.

1. TD method exploits the markove property. ie future rewards rely on upon the current state and therefore it generally more efficient to use. TD in markov environment because in our case starting and ending location had change but intermediate states are still the same so TD method will able to converge easily and will faste adapt to the environment as compared to the M.C.

As for as MC methods are concern they are not based on the markov property as it is based on the rewards of the entire learning process. So in our case initially MC method will suffer to converge but later it will converge to optimal value.

Q8 **Exercise 6.12**

Yes Q-learning will behave exactly same as the SARSA if action selection is greedy because in this case Targeted policy will be same as the behaviour policy and in SARSA we also have the same scenario.

Q6 **Exercise-6.3**

The problem is discounted ($\lambda = 1$) and taking $\gamma = 0.1$ for TD(0) update we obtain

$$V(S_t) \leftarrow V(S_t) + 0.1\left(r_{t+1} + V(S_{t+1}) - V(S_t)\right)$$

for transition among states that do not end in one of the terminal states we receive a zero reward. and since intially our value function begins as the constant

if we take step left

$$V(A) \leftarrow V(A) + 0.1\left(0 + 0 - V(A)\right)$$

$$= 0.9\,V(A) = 0.45.$$

which agrees with plotted value of V(A) · for the first iteration

Exercise - 6.4

MC methods are susceptible to wide values of α which we can see the graph itself. In TD if we used values α > 0.5 it will converge more faster than the shown in the graph.

Exercise - 6.5 -

Large value of α imply more V(s) in update in each step. This will make TD(0) algorithm depend more heavily on specific returns received at each step of the specific value.

And smaller values of α learning takes longer to do but is much less sensitive to random step.