Assignment 3

Answer 1.

Pseudo code

Initialise

$\pi(s) \in A(s)$, for all $s \in S$

$Q(s,a) \in R$ for all $s \in S, a \in A(s)$ } arbitrarily

$N(s) \leftarrow 0 \quad \forall S$

Loop infinite (for each episode)

choose $S_0 \in S$, $A_0 \in A(S_0)$ randomly such that all pair have probability >0

Generate an episode from S_0, A_0 following policy 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 , $t = T-1, T-2 \ldots 0$.

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

$\gamma \leftarrow$ discount factor.

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

$N(S_t) \leftarrow N(S_t) + 1$

$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{1}{N(S_t)} [G - Q(S_t, A_t)]$

$\pi(s_t) \leftarrow \underset{a}{\text{argmax}}\ q(s_t, a)$

$Q(s_t, A_t) \leftarrow \text{average}(\text{Return}(s_t, A_t))$

Above step computes avg. Return for (s_t, A_t).
We can do the above using the incremental mean. For example consider $\mu_k \leftarrow$ avg; $k \leftarrow$ no. of samples.
$x_1, x_2 \ldots x_k$ are samples.

$$\mu_k = \frac{1}{k} \sum_{j=1}^{k} x_j$$

$$= \frac{1}{k} \left[x_k + \sum_{j=1}^{k-1} x_j \right] \quad -(1)$$

$$\sum_{j=1}^{k-1} x_j = \underbrace{\frac{1}{k-1} \sum_{j=1}^{k-1} x_j}_{\mu_{k-1}} (k-1)$$

eq (1) can be written as.

$$\mu_k = \frac{1}{k} \left[x_k + (\mu_{k-1}) * k-1 \right]$$

$$\mu_k = \mu_{k-1} + \frac{1}{k} \left[x_k - \mu_{k-1} \right]$$

↳ equivalent to.

$$Q(s_t, A_t) \leftarrow q(s_t, A_t) + \frac{1}{N(s_t)} \left[G - q(s_t, A_t) \right]$$

Answer 1.

Backup diagram can be shown as,

$$q(s,a) = q(s,a) + \alpha[R + \gamma Q(s', a') - q(s,a)]$$

$\gamma \leftarrow$ discounting factor.

Ans 2.

$$E[\rho_{t:T-1} G_t \mid S_t = s, A_t = a] = q_\pi(s,a)$$

$$\rho_{t:T-1} = \prod_{k=t}^{T-1} \frac{\pi(A_k \mid S_k)}{b(A_k \mid S_k)}$$

$$q(s,a) = \frac{\sum_{t \in T(s)} \rho_{t:T-1} G_t}{|T(s)|}$$

In case of weighted importance sampling

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

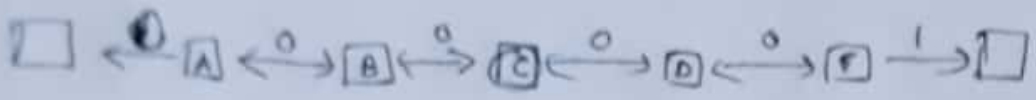
Ans 5.

TD learning will be much better than MC learning in such case.

Since only the initial initial route is changed and other states encountered in new building will be same such as existing highway, secondary road. These the value function estimates for the states of new building will be very close to the old building. So if we guess the initial root then function convergence will be faster in case of TD estimate.

Ans 6.

example 6.3.



$\lambda = 1, \quad \alpha = 0.1$

If we take TD(0) update.

$$V(S_t) = V(S_t) + 0.1\,(R_{t+1} + V(S_{t+1}) - V(S_t))$$

Since initial fn is const.

first update is $V(S_t) = V(S_t)$.

$$V(A) = V(A) + 0.1\,(0 + 0 - V(A))$$

$V(S_{t+1}) = 0$

$R_{t+1} = 0.$

$V(A) = 0.5$ (initial value)

$V(A) = 0.45.$

Thus in first episode state value is decreased by 0.05.

exercise 6.4.

TD performs better for wide range of values of α.

exercise 6.5

$\alpha \uparrow$ (higher) → more value $V(s)$ update for each step.

Temporal diff. depends on return received on each step.

Thus the going down and up for RMS error may be due to randomness in the reward.

learning is takes a lot of time for smaller values of alpha.

Ans 8,

Q learning is off-policy because target and behaviour policy are different.

Q learning and sarsa (SARSA) → which is on-policy, becomes same if action selection is greedy.

We continually estimate q_π and at the same time change behaviour policy π towards greediness wrt q_π.

Thus since Q-learning becomes on-policy as SARSA they will make same action-selection and step up dates.