Parina Lad.

Ex 4: Monte-Carlo Methods.

Q.1)

a) Incremental first-visit MC policy evaluation (estimating value f^n of a policy in MDP)

→ First-visit MC prediction, for estimating $V \approx v_\pi$

Input: policy π to be evaluated.

Initialize:

V(s) ∈ R, arbitrarily, for all s ∈ S

N(s) ← 0 ∀ s ∈ S

Loop forever (for each episode):

Generate an episode following: S0, A0, R1, S1, A1, R2 .... ST-1, AT-1, RT

G ← 0

Loop for each step of episode, t = T-1, T-2 ... 0:

G ← γG + R_{t+1}

Unless St appears in S0, S1 ... St-1:

Append G to Returns(St).

V(St) = incremental_avg(Returns(St), G).

```
def incremental_avg(Returns(St), G):
    N(St) += 1
    V(St) = V(St) + (G - V(St)) / N(St)
    return V(St).
```

this f^n → returns avg of for each state, V(St).

- updates it after each new return is observed.

N(St) = keeps track of number of times state St has been visited.

b) To alter the pseudocode for Monte Carlo Es to maintain just the mean & a count (for each state-action pair) & update them incrementally,

Monte Carlo ES (Exploring starts), for estimating $\pi \approx \pi_*$

Initialize:

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

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

$N(s,a) \leftarrow 0 \quad \forall\ s \in S,\ a \in A(s)$.

Loop forever (for each episode):

choose $S_0 \in S$, $A_0 \in A(S_0)$ randomly such that all pairs have 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 $S_0, A_0, S_1, A_1 \ldots S_{t-1}, A_{t-1}$:

Append G to $Returns(S_t, A_t)$

$N(S_t, A_t) \mathrel{+}= 1$

$Q(S_t, A_t) = Q(S_t, A_t) + (G - Q(S_t, A_t)) / N(S_t, A_t)$

$\pi(S_t) \leftarrow \underset{a}{\operatorname{argmax}}\ Q(S_t, a)$

Q.2) first-visit vs. every-visit

a) We know (in general) →

every-visit MC - expected to produce more accurate estimates of state values than first-visit MC.

↓

it considers all returns following visit to a state.

but first-visit MC only considers - return following the first visit to a state.

if we consider Black Jack problem,

There is constantly changing state in each episode

- The state only appears once, even if every-visit MC is used.
- Hence, it gets same result as using first-visit MC.

b) • First-visit estimator: $\tau = 10, \gamma = 1$

- only considers the return following first-visit to a state.
- return following the 1st visit to the non-terminal state is 10.

$\therefore V(s) = 10$

$G_t = R_{t+1} + \gamma G_{t+1}$

$G_0 = R_1 + G = 10 \quad G_1 = R_2 + G_2 = 9.$

$\vdots \; G_{10} = 0.$

• Every-visit estimator:

- considers all returns following visits to a state.
- Here, only state that is visited is the nonterminal state.

$$\therefore V(s) = \frac{(1 + 2 + 3 + \dots 10)}{10} = \frac{n(n+1)}{2} \cdot \frac{1}{10}$$

$$= \frac{10 \times 11}{2} \times \frac{1}{10} = 5.5$$

→ first-visit estimator produces a higher estimate as → it only considers return following first visit to the state. ie highest possible return.

→ But every-visit produces a lower estimate as it considers all returns following visits to the state, including returns from where episodes ends early

Q. 5) OFF-policy Method –

a) Suppose we have sequence of returns $G_1, G_2, \dots G_{n-1}$ all starting in the same state & each with a corresponding random weight W_i

$$V_n = \frac{\sum_{k=1}^{n-1} W_k G_k}{\sum_{k=1}^{n-1} W_k}, \quad n \geq 2 \qquad \dots \; 5.7.$$

To derive → $V_{n+1} = V_n + \frac{W_n}{C_n} [G_n - V_n], \quad n \geq 1$

derivation:

$$\rightarrow \quad V_{n+1} = \frac{\sum_{k=1}^{n} w_k G_k}{\sum_{k=1}^{n} w_k}, \quad n \geq 1$$

$$= \frac{\sum_{k=1}^{n-1} w_k G_k}{\sum_{k=1}^{n} w_k} + \frac{w_n G_n}{\sum_{k=1}^{n} w_k}$$

$$= \frac{\sum_{k=1}^{n-1} w_k}{\sum_{k=1}^{n-1} w_k} \times \frac{\sum_{k=1}^{n-1} w_k G_k}{\sum_{k=1}^{n} w_k} + \frac{w_n G_n}{\sum_{k=1}^{n} w_k}.$$

$$= V_n - \frac{V_n \times w_n}{\sum_{k=1}^{n} w_k} + \frac{w_n G_n}{\sum_{k=1}^{n} w_k}$$

$$= V_n + (G_n - V_n) \times \frac{w_n}{\sum_{k=1}^{n} w_k} \rightarrow c_n$$

$$= V_n + \frac{w_n}{c_n} \times (G_n - V_n) \quad \ldots \quad (5.8)$$

b) Off-policy MC works by correcting bias in unweighted MC algorithm due to difference between behavior policy b, target policy π.

- can be done by weighting the unweighted returns by the importance sampling ratio, $\pi(A_t|s_t) / b(A_t|s_t)$

- But in the boxed algorithm for off-policy MC control, W update only involves $\frac{1}{b(A_t|s_t)}$

- it's only used to update the weights C, → used to normalize the importance sampling ratio.

- $C_{n+1} = C_n + \alpha_n \cdot \frac{1}{b(A_n|s_n)}$ ← shows how W update is used to normalize the importance sampling ratio.

↓

weights C → sum to 1.

- W update in boxed algorithm for off-policy MC control is only used to normalize the importance sampling ratios.
- As the importance sampling ratios have been normalized, they can be used to weight the unweighted returns
- it will produce weighted returns., this can be used to update the value estimates using MC update rule.