Name: SAHAS MARWAH

Roll No.: 2020237

RL HW-2

1. $p(\text{even}) = 2\,p(\text{odd})$, Gaussian $(i, 1)$
1. $\rightarrow$ 10 ; Lever / arm 10 times

$E[\text{total reward after 10 pulls}] = 10 \times E[q_*(a)]$

and, $E[q_*(a)]$

$= \sum p(a) \cdot q_*(a)$

$= \frac{2}{15}(2+4+6+8+10) + \frac{1}{15}(1+3+7+5+9)$

$= 4 + \frac{5}{3} = \frac{17}{3}$

$$p[a] = \begin{cases} 2/15, & a \in \text{even} \\ 1/15, & a \in \text{odd} \\ & \text{b/w } [1, 10] \\ & \in I. \end{cases}$$

Hence, $10\,E[q_*(a)]$

$\Rightarrow \frac{170}{3} = 56.66$

---

2. $R = 0 \mid p = 1/2$
$R = 1 \mid p \neq 1/2$
$\Big\} \forall\, a \in \{1, 2, 4, 5, 7, 9, 10\}$

$R = 0 \mid p = 3/10$
$R = 0.2 \mid p = 3/10$
$R = 1 \mid p = 4/10$
$\Big\} \forall\, a \in \{3, 6, 8\}$

Now, $q_*^{(1)}(a) = 0(0.5) + 1(0.5) = \underline{0.5}$

and, $q_*^{(2)}(a) = 0(0.3) + 0.2(0.3) + 1(0.4)$

$= \underline{0.46}$

Now, for optimal stochastic policies we will make 6 combinations of arms with $q_*(a) = 0.5$.

1. Arms(1, 2, 4, 5, 7) with 0.2 each
2. Arms(1, 2, 4, 5) with 0.25 each
3. Arms(1, 2, 4) with 1/3 each
4. Arms(1, 2) with 1/2 each
5. Arms(1, 2, 4, 5, 7, 9) with 1/6 each
6. Arms(1) with 0.4 and Arm(2) with 0.6.

---

4. Table for $p(s', r|s, a)$

s	a	s'	r'	$p(s', r|s, a)$
high	search	high	r_{search}	α
high	search	low	r_{search}	$(1-\alpha)$
low	search	high	-3	$(1-\beta)$
low	search	low	r_{search}	β
high	wait	high	r_{wait}	1
low	wait	low	r_{wait}	1
low	recharge	high	0	1

★. This table was derived by MDP table given in Example 3.3. and the transition graph.

★. As rewards have no dependence on prob. distribution we can say that $p(s', r|s, a) = p(s'|s, a)$.

3. $\gamma \in \{0, 1\}$. explore at $t = 1, 3, 5$ (over non-greedy only)
$Q_1(a)$, $a \in \{1, 2, 3\}$ exploit at $t = 2, 4, 6$
Sample Mean: Let $Q_1(1) = 0.2$
$Q_1(2) = 0.3$
$Q_1(3) = 0.5$

At $t = 1$:
Let $A_1 = 1$, $R_1 = 1$
$Q_2(1) = \frac{0.2 + 1}{2} = 0.6$, $Q_2(2) = 0.3$, $Q_2(3) = 0.5$

At $t = 2$:
(Greedy) Let $A_2 = 1$, $R_2 = 0$
$Q_3(1) = \frac{0.6 + 0}{3} = 0.2$, $Q_3(2) = 0.3$, $Q_3(3) = 0.5$

At $t = 3$:
Let $A_3 = 2$, $R_3 = 0$
$Q_4(1) = 0.2$, $Q_4(2) = \frac{0.3 + 0}{2} = 0.15$, $Q_4(3) = 0.5$

At $t = 4$:
(Greedy) Let $A_4 = 3$, $R_4 = 1$
$Q_5(1) = 0.2$, $Q_5(2) = 0.15$, $Q_5(3) = \frac{0.5 + 1}{2} = 0.75$

At $t = 5$:
Let $A_5 = 1$, $R_5 = 1$
$Q_6(1) = \frac{0.2 + 1}{4} = 0.3$, $Q_6(2) = 0.15$, $Q_6(3) = 0.75$

(Greedy) At $t = 6$:
Let $A_6 = 3$, $R_6 = 0$
$Q_7(1) = 0.3$, $Q_7(2) = 0.15$, $Q_7(3) = \frac{0.75 + 0}{3} = 0.25$

Not Req.

6. 3.15

The signs of rewards is not important, only the intervals between them are as the difference of rewards matters.

Generally, if good reward = -1 or 5
and bad reward corresponding = -3, 3
They will have same results.

Now,

Eq. 3.8 $\Rightarrow$ $G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$

$= R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} \text{ ---- } \infty$

Adding a const. c to all rewards:

$\Rightarrow G_t = \sum_{0}^{\infty} \gamma^k [R_{t+k+1} + c]$

$\therefore G_t = \frac{1}{1-\gamma} (R_{t+k+1} + c)$

As every state reward gets an additional const. term, added to it, $v_c = \mathbb{E}\left[\sum_{0}^{\infty} \gamma^k c\right]$

$\therefore v_c = \frac{c}{1-\gamma}$

Hence, it does not affect relative values in any state as $v_c = \frac{c}{1-\gamma}$ interms of c and γ.

3.16

In an episodic task, adding a const. to all rewards does affect the agent because the cumulative reward is dep. on the length of the episode.

let us say if $r = -1$, the lenght of the episode decreases as the agent wants to find the exit quickly. (in maze runner)

But if say $r = +1$ because of some +ve value of c, the lenght of episode increases as agent does not want to find the exit, in maze runner.

* This case will go onto ∞.

$s' = p(s', r | s, a)$

$(s', a') = \pi(a', s') \cdot \sum_{\delta} p(s', r, s, a)$

8. v_* in terms of q_*:

$$v_*(s) = \max_{a \in A(s)} q_{\pi *}(s, a)$$

$$= \max_{a \in A(s)} \sum_{s', r} p(s', r | s, a) [r + \gamma \max_{a \in A(s')} v_*(s', a)]$$

11. RV R_{t+2} dependent on S_t, A_t =?

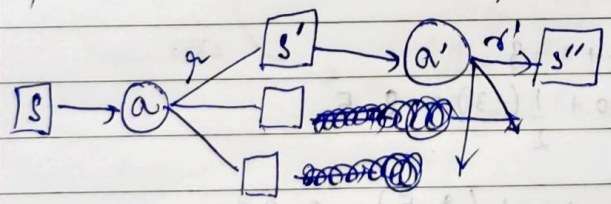
$p[R_{t+2} | s', a']$

$= \sum_{s''} p(R_{t+2}, s'' | s', a')$

- Now, $p[R_{t+2} | s, a) = \sum_{s, a} p(s', a' | s, a) \cdot p(R_{t+2} | s', a')$

$= \sum_{s', a'} \left[\pi(a' | s') \cdot \sum_{r} p(s', r | s, a) \right] \cdot \sum_{s''} \left[p(R_{t+2}, s'' | s', a') \right]$

$= \sum_{s', r} p(s', r | s, a) \cdot \sum_{s'', a'} \pi(a' | s') p(s'', r' | s', a)$



12. $E[R_{t+2} | S_t = s, A_t = a]$

$= \sum_{r'} r' P(r = R_{t+2} | S_t = s, A_t = a)$

(from above)

$= \sum_{r'} r' \sum_{s', r} p(s', r | s, a) \sum_{r, s''} \pi(a' | s') p(s'', r' | s', a)$

↳ MDP's PMF

13. $v_\pi(s) = E[G_t | S_t = s]$

$= E_\pi[R_{t+1} + \gamma G_{t+1} | S_t = s]$

$= E_\pi[R_{t+1} | S_t = s] + \gamma E_\pi[G_{t+1} | S_t = s]$

$= E_\pi[R_{t+1} | S_t = s] + \gamma E_\pi[R_{t+2} + \gamma G_{t+2} | s]$ —— recursively

$= E_\pi[R_{t+1} | s] + \gamma E_\pi[E_\pi[G_{t+1} | S'] | s]$

$= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r | s, a)[r + \gamma E_\pi[G_{t+1} | S_{t+1} = s']]$

$= \sum_a \pi(a|s) \sum_{s',r} p(s', r | s, a)[r + \gamma v_\pi(s')]$ $\forall s \in S$.

↳ Bellman Equation for $v_\pi(s)$.

---

14. $R_1 = 2$; $R_2 = -1$, $R_3 = 10$, $R_4 = -3$, $\gamma = 0.5$

Discounted reward for each timestep:

$G_0 = R_1 + \gamma G_1$

$G_1 = R_2 + \gamma G_2$

$G_2 = R_3 + \gamma G_3$

$G_3 = R_4 + \gamma G_4$ ~~0~~ $\therefore G_3 = R_4 = -3$

$\Rightarrow G_2 = 10 + \frac{1}{2}(-3) = 8.5$

$\Rightarrow G_1 = -1 + \frac{1}{2}(8.5) = 3.25$

$\Rightarrow G_0 = 2 + \frac{1}{2}(3.250) = 3.625$

New $G_t = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} \text{ —— } \infty$

$= c + \gamma c + \gamma^2 c \text{ —— } \infty$

$= c\left(\frac{1}{1-\gamma}\right) \equiv \frac{c}{1-\gamma}$

15. Given: $v_*(s) \quad \forall s \in S$

We know, $\pi_*(s) = \max_a q_*(s,a)$

$$\pi_*(s) = \max_a \sum_{s', r} p(s', r | s, a)[r + \gamma v_*(s')]$$

So, in this case we take the best action that returns the highest value to maintain optimality by it erating over all ~~states~~ actions that can be chosen over a state, greedily.

---

17. Given: $\pi'(s) \geq \pi(s)$

Prove: $v_{\pi'}(s) \geq v_\pi(s)$

We know, $v_\pi(s) \leq q_\pi(s, \pi'(s))$

From here

$$= \mathbb{E}[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S, \pi'(s)]$$

$$= \mathbb{E}_{\pi'}[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S]$$

$$\leq \mathbb{E}_{\pi'}[R_{t+1} + \gamma q_\pi(S_{t+1}, \pi'(S_{t+1})) \mid S]$$

$$= \mathbb{E}_{\pi'}[R_{t+2} + \gamma \mathbb{E}[R_{t+2} + \gamma v_\pi(S_{t+2})] \mid S_{t+1}] \mid S_t = S, A_{t+1} = \pi'(S_{t+1})]$$

$$= \mathbb{E}_{\pi'}[R_{t+1} + \gamma R_{t+2} \cdots \gamma^2 v_\pi(S_{t+k}) \mid S]$$

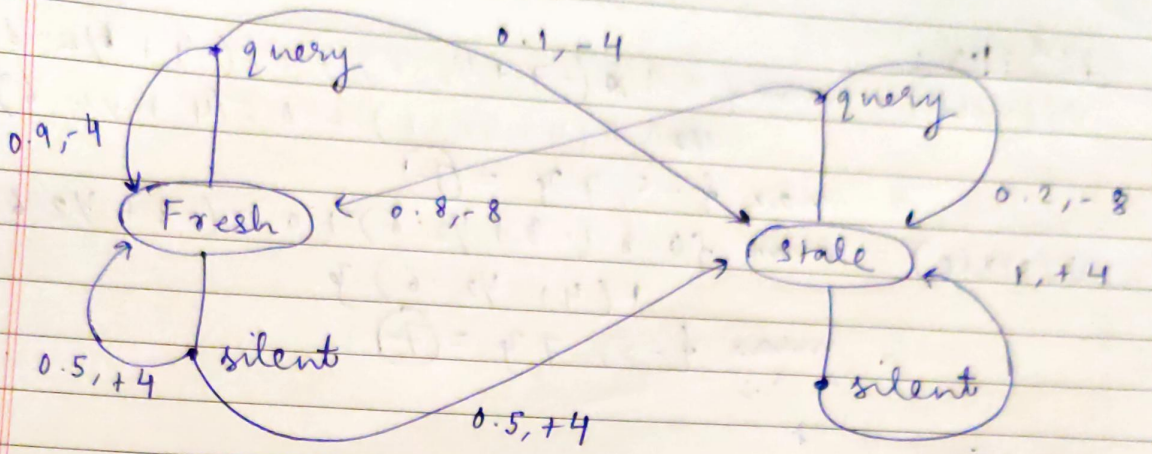
$$= \mathbb{E}_{\pi'}[R_{t+1} + \gamma R_{t+2} \cdots \gamma^5 R_{t+6} \mid S]$$

$$= \mathbb{E}_{\pi'}[G_t \mid S_t = S]$$

$$= v_{\pi'}(s)$$

Hence, $\boxed{v_\pi(s) \leq v_{\pi'}(s)}$

16. (a)



(b) Value Iteration

(c) $\gamma = 1/2$, $v(\text{fresh}) = 0$, $v(\text{stale}) = 0$, initially.

$$v_{k+1}(s) = \max_a \sum_{s',r} p(s', r \mid s, a)(r + \gamma v_k(s'))$$

1st iter:

$$v(\text{fresh}) = \max\{0.9(-4 + 1/2 \cdot 0) + 0.1(-4 + 1/2 \cdot 0),\ 0.5(4 + 1/2 \cdot 0) + 0.5(4 + 1/2 \cdot 0)\}$$
$$= \max\{-4.0, 4.0\} = (4)$$

$$v(\text{stale}) = \max\{0.8(-8 + 1/2 \cdot 0) + 0.2(-8 + 1/2 \cdot 0),\ 1(4 + 1/2 \cdot 0)\}$$
$$= \max\{-8, 4\} = (4)$$

2nd iter:

$$v(\text{fresh}) = \max\{0.9(-4 + 0.5(4)) + 0.1(-4 + 1/2 \cdot 4),\ 0.5(4 + 1/2 \cdot 4) + 0.5(4 + 0.5(4))\}$$
$$= \max\{-2, 6\} = (6)$$

$$v(\text{stale}) = \max\{0.8(-8 + 1/2 \cdot 4) + 0.2(-8 + 1/2 \cdot 4),\ 1(4 + 1/2 \cdot 4)\}$$
$$= \max\{-6, 6\} = (6)$$

3rd iter:

$v(\text{fresh}) = \max\{0.9(-4 + 1/2 \cdot 6) + 0.1(-4 + 1/2 \cdot 6),$
$0.5(4 + 1/2 \cdot 6) + 0.5(4 + 1/2 \cdot 6)\}$
$= \max\{-5, 7\} = 7$

$v(\text{stale}) = \max\{0.8(-8 + 1/2 \cdot 6) + 0.2(-8 + 1/2 \cdot 6),$
$1(4 + 1/2 \cdot 6)\}$
$= \max\{-5, 7\} = 7$

4th iter:

$v(\text{fresh}) = \max\{0.9(-4 + 1/2 \cdot 7) + 0.1(-4 + 1/2\ 7),$
$0.5(4 + 1/2 \cdot 7) + 0.5(4 + 1/2 \cdot 7)\}$
$= \max\{-1/2, 15/2\} = 7.5$

$v(\text{stale}) = \max\{0.8(-8 + 1/2 \cdot 7)) + 0.2(-8 + 1/2\ 7)$
$1(4 + 1/2\ 7)\}$
$= \max\{-4.5, 7.5\} = 7.5$

Policy Iteration

$$v_{k+1}(s) = \sum_a \pi_k(a|s) \sum_{s',r} p(s',r|s,a)\cdot(r + \gamma v_k(s'))$$

$$\pi_{k+1}(s) = \max_a \sum_{s',r} p(s',r|s,a)\cdot(r + \gamma v_{\pi k}(s'))$$

After we do this, we will go onto policy Improvement