AI 3000

MID-TERM EXAM

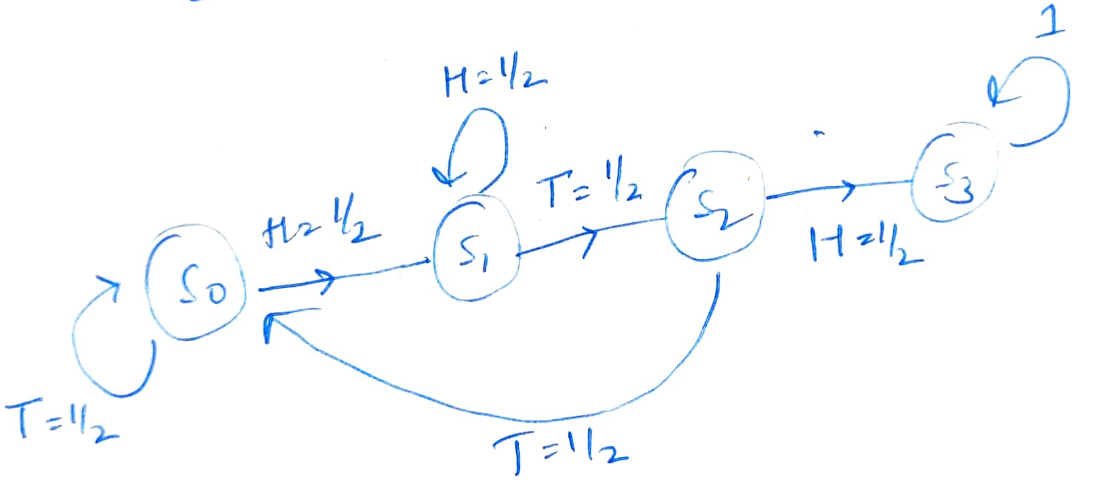
EE18BTECH11026

K. Surya Prakash

Q1) State-Transition diagram

S_0: start state
S_3: End state

$S = \{S_0, S_1, S_2, S_3\}$



$$P = \begin{bmatrix} 1/2 & 1/2 & 0 & 0 \\ 0 & 1/2 & 1/2 & 0 \\ 1/2 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Let Reward:

$$R(S) = \begin{cases} -1 & ; s \in \{S_0, S_1, S_2\} \\ 0 & ; s \in \{S_3\} \end{cases}$$

Let discount factor : $(\gamma = 1)$

$\therefore$ This is a MRP (S, P, R, γ)

---

* Finding Expected no. of tosses through Bellman Eqn's

$$V(S) = R(S) + \gamma \sum_{s' \in S} P_{ss'} V(s') \rightarrow (1)$$

Eq(1) for all states:

$$V(S_0) = -1 + \frac{1}{2}(V(S_0) + V(S_1))$$
$$V(S_1) = -1 + \frac{1}{2}(V(S_1) + V(S_2))$$
$$V(S_2) = -1 + \frac{1}{2}(V(S_0) + V(S_3))$$
$$V(S_3) = 0$$

Solving:

$$V(S_2) = -1 + \frac{1}{2}(V(S_0))$$
$$V(S_1) = -1 + \frac{1}{2}\left(\frac{3}{2} V(S_1) - 1\right)$$
$$\boxed{V(S_1) = -6}$$

$$\Rightarrow V(S_0) = -1 + \frac{1}{2}(V(S_0) - 6)$$

$$\Rightarrow \boxed{V(S_0) = -8}$$

* Since we choose $R(S) = -1 \;\forall\; S \in \{S_0, .. S_2\}$

It represents steps counted whenever we move from one state to other.

$\therefore$ $V(S_0)$ should determine Expected no. of trials to reach the end state (S_4).

$\therefore$ Averge no. of trails = $\underline{\underline{8}}$

— X —

Q1 - DONE

Question 2

Q2.a) $R(s,a) = R_1(s,a) + R_2(s,a)$

$$\pi_1^*(s) = \arg\max_{a'} Q_1^{\pi_1^*}(s,a')$$

$$\pi_2^*(s) = \arg\max_{a'} Q_2^{\pi_2^*}(s,a')$$

$$Q^{\pi_1^*}(s,a) = R_1(s,a) + \gamma \sum P_{ss'}^{a} V^{\pi_1^*}(s')$$

$$Q^{\pi_2^*}(s,a) = R_2(s,a) + \gamma \sum_{s'} P_{ss'}^{a} V^{\pi_2^*}(s')$$

For π^* : Reward: $R(s,a) = R_1(s,a), R_2(s,a)$

$$\pi^* = \arg\max_a Q^*(s,a)$$

$$Q^*(s,a) = (R_1(s,a) + R_2(s,a)) + \gamma \sum_{s'} P^a_{ss'} (V^{\pi^*}(s'))$$

$\therefore$ It is not possible to formulate π^* simply.

Alternate Explanation:

Since π depends on V

$$V^{\pi_1^*} = (I - \gamma P^{\pi_1})^{-1} R_1$$

$$V^{\pi_2^*} = (I - \gamma P^{\pi_2})^{-1} R_2$$

$$V^{\pi^*} = (I - \gamma P^{\pi})^{-1} (R_1 + R_2)$$

Cannot simply formulate V^{π^*} as $V^{\pi'}$ & $V^{\pi_2^*}$

due to P^{π}.

P.2b) $M = \langle S, A, P, R, \gamma \rangle$ P2b

$f, g : S \times A \to R$; $\mathcal{L} \to$ Bellman optimality operator

~~fIS~~ $V_f(s) = \max_a f(s,a)$

Given;

$$(\mathcal{L}f)(s,a) = R(s,a) + \gamma P(s,a) V_f(s')$$

Prove that:

$$\|\mathcal{L}f - \mathcal{L}g\|_\infty \le \gamma \|f - g\|_\infty$$

$\Rightarrow \|\mathcal{L}f - \mathcal{L}g\|_\infty =$

$$\max_s \max_a (|\mathcal{L}f(s,a) - \mathcal{L}g(s,a)|)$$

$$= \max_s \max_a (|(R(s,a) + \gamma P(s,a) V_f(s)) - (R(s,a) + \gamma P(s,a) V_g(s))|)$$

$$= \max_s \max_a (|\gamma P(s,a) \underbrace{(V_f(s') - V_g(s')}|)$$

$$= \| \gamma P (V_f - V_g \|_\infty$$

$$\leq \gamma \|P\|_\infty \| V_f - V_g \|_\infty \quad \Big\} \begin{array}{l} P \text{ is probability} \\ \text{matrix} \\ \text{all elements} \leq 1 \end{array}$$

$$\leq \gamma \| V_f - V_g \|_\infty \quad \rightarrow (1)$$

we know that $V_f(s) = \max_a f(\cdot, a)$

$$\| V_f - V_g \|_\infty = \max_a \max_s | V_f(s) - V_g(s) |$$

$$= \max_a \max_s \left(\max_a (f(s,a)) - \max_a (g(s,a)) \right)$$

$$\leq \max_a \max_s \left(\max_a (f(s,a) - g(s,a)) \right)$$

↳ IMP !

$$\leq \max_a \max_s (f(s,a) - g(s,a))$$

$$\leq \| f - g \|_\infty$$

$\therefore \|V_f - V_g\|_\infty \leq \|f-g\|_\infty$

continuing from (1)

$\leq \gamma \|V_f - V_g\|_\infty$

$\leq \gamma \|f-g\|_\infty$

$$\therefore \boxed{\|L_f - L_g\|_\infty \leq \gamma \|f-g\|_\infty}$$

* Q-2 DONE *

Q4) Problem Formulation:

To make it simple:

State of an ingridient depends on:

1) Quanty (< thresh , > threshold)

2) Days old (0, 1, 2, ≥3)

Each State is defined by quantity of the ingridiet (if less than a certain threshold (T) or not)

ii) day older (~~0, 1, 2, ≥3~~). (≤ 3, >3)
binary.

Action space:

The Owner can have 3 actions

1) Refill: ~~Bg~~ Buy and add ingridients by not discarding alredy non expired ingridients

2) Replace: Discard the existing lot & replace with a new one

3) Idle: Do nothing.

(b) Reward:- $S: S_q, S_d$

~~R(S, a)~~ = $S_q = 1$: if > Thrs
$S_q = 0$; else

$S_d = 0$; if < days
$= 1$; if > days

$R(S, a) \Rightarrow R(S, \text{Refill}) =$

Preferred rewards

q , d — Best action (High rewads

$<T$; <3 → Refill > Idle > Replace

$<T$; >3 → Replace > ~~Idl~~ Refill > Idle

$>T$; >3 → ~~Idle~~ Replace >> Idle > Refill

$>T$; <3 → Idle > Refill > Replace

(c) Since my state space is binary, will be using non-discounted.

But if no. of days taken (0, 1, 2, 3 ..) discounting would give better results.

→ If $d \to$ closer to 3 ; reward for replacing increases.

(d) Since this is an real time problem and accurate formulation of state parameters like (P) are changing. We need to rely on Reinforcement Learning. We can gain more knowledge over time and can make the model better.

e. Since state and action space is finite and less in this case. It will not be harder to traverse through the entire trajectory. ~~We can~~

$\Rightarrow$ MC methods will be used.

$\because$ This is because the length of the episode for each ingredient is finite and less.

(f). Yes. Since the data is based on Learning is ever changing.

→ An acute function approximator might lower the variance. We can model the real world phenomena like: likeability of combination of bread & filling using a function.

Q-4 Done

Q5) MISCELLANEOUS

a) No. MDP formalism requires knowledge of <S, P, R, γ A>.

→ Some times |S| can be huge and will be difficult to keep track. Eg. Atari games

→ P is also hard to formulate for larger environments.

→ This is where we use model-free methods.

---

b) For MDP:

Discount factor } $\gamma \in [0,1]$

→ Modeling MDP requires choosing actions to maximise total discounted reward

$\mathbb{E}(G_t | S_t = s)$:

$$G_t = \sum_{k=0}^{\infty} (\gamma^k r_{t+k+1})$$

If episodes are long. To avoid G_t from exploding we have to have $\gamma \in [0,1)$.

(c) By ordering of policies   Yes

if $\pi > \pi' \Rightarrow V^{\pi}(s) \geq V^{\pi'}(s)$

$\Rightarrow$ In Policy Iteration:

we converge to the best optimal policy π^*

$\Rightarrow$ ie; $\pi^* > \pi_i$ $\forall$ possible policies

$$\pi_*(s) = \arg\max_{a \in A} Q_*(s,a)$$

$$V_*(s) = \max_{a \in A} Q_*(s,a) \quad \forall s \in S$$

$\therefore$ $V^{\pi_*}(s) \geq V^{\pi'}(s)$; π' all other policies,

$$\underbrace{V^*(s) \geq V^{\pi'}(s)}$$

The Optimal policy achieved through policy iteration attains the Optimal value function

$\rightarrow$ In Value Iteration:

$$V_{K+1}(s) \leftarrow \max_a \left(\sum P^a_{ss'} \left(R^a_{ss'} + \gamma V_K(s') \right) \right)$$

$\Rightarrow$ V_K will converge to V_*

$$\boxed{V^*(s) \geq V^{\pi}(s)}$$

(d) In DP-setup, we already modelled the randomness of the system using 'p' which we lack in model-free setting.

→ The ε-greedy in the latter tries to inculcate the randomness. ~~by~~

→ This ensures continual exploration which is already present in DP,

(e)

Advantages of MC over DP

(i) ~~the~~ Compatible with Non-Markovian domain problems

(ii) Since it is a model-free method, does not require ~~the~~ environment parameters like 'p'.

(f) Evaluating value func. in MRP

~~V(S)~~

$$V \leftarrow R + \gamma P V$$

$V \to$ vector of size $|S|$

$R \to$ " " $|S|$

$P \to$ matrix : of shape $|S| \times |S|$

Each update consists of $O(|S|^2)$ operations.

(g) Yes. It is possible.

There might be no action that can jump from $S_1 \to S_3$.

only possibilities are $S_1 \to S_2 : a_2$

$S_1 \to S_1 : a_1$

→ $\because$ a_2 pushes $S_1 \to S_2$: Reward = 2

a_1 : $S_1 \to S_1$: Reward -1

$\therefore$ If followed a greedy policy

$\pi(S_1) = a_1$

But since, we follow ε-greedy

* 1st action:

  $S_1, a_1, 1, S_1$ } Random

* 2nd action

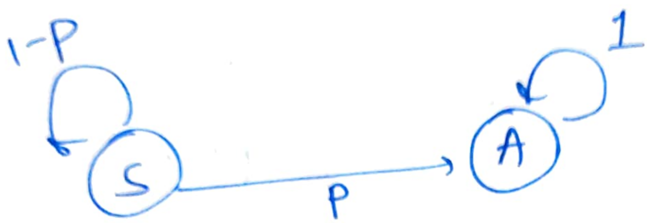
  $S_1, a_2, 2, S_2$ } greedy

$\because Q(S_1, a_2) > Q(S_1, a_1)$.

Problem 3

(Q3) Monte Carlo Methods:

$S = \{S, A\}$; $\gamma = 1$, $p \in (0,1)$



a) Trajectory's generic form.

$S \to S \to S \to \cdots \to \underset{\sim}{A}$

first appearance of 'A'

$S^N A$; where $A = \{1, 2, \ldots \infty\}$

(b) V(S) using first visit MC:

The trajectories can be:

$\{S^N A\}$

Let prob. of getting trajectory:

$$P(S^N A) = p(1-p)^{N-1}$$

~~Rew~~

~~S-A~~

b) First visit MC:

Trajectory	Reward	Prob. of occuring
S-A	1	P
S-S-A	2	P(1-P)
S 3 A	3	P(1-P)^2
⋮ $S^N A$	N	$P(1-P)^N$

First visit MC:

average reward:

$V(S) = 1(p) + 2p(1-p) + 3p(1-p)^2 + \dots N p(1-p)^{N-1}$

$= p(1 + 2(1-p) + 3(1-p)^2 + \dots N(1-p)^{N-1})$

- Upon solving

$$V(S) = \frac{1-(N+1)(1-p)^N + N(1-p)^{N+1}}{p}$$

(c) Every visit MC:

$S - A = 1 \quad p$

$S - S - A = 1+2 \quad p(1-p)$

$S - S - S - A = 1+2+3 \quad p(1-p)^2$

$$\therefore V(S) = \sum_{n=1}^{N} \frac{n(n+1)}{2} (1-p)^{n-1} p.$$

(d) True Estimate

$V(S) = 1 + p(V(A)) + (1-p)(V(S))$

$\Rightarrow V(A) = 0 + 1(V(A)) \Rightarrow \boxed{V(A) = 0}$

$\Rightarrow V(S) = (1-p)(V(S)) + 1$

$\Rightarrow p(V(S)) = 1$

$\Rightarrow \boxed{V(S) = 1/p}$

(e) Yes every visit MC is biased

$\text{Bias}(\hat{V}(s)) = E(\hat{V}(s)) - V(s)$

$=$

$\hat{V}(s) \Rightarrow$ for $\underline{n \text{ trajectories}}$.

$$\left.\begin{aligned} S-A &= 1 \\ S-S-A &= 1+2 \\ S-S-S-A &= 1+2+3 \end{aligned}\right\} \quad V(s) = \sum_{n=1}^{N} \frac{n(n+1)}{2} (1-p)^{n-1} p$$

$$= E\left(\sum_{n=1}^{N} \frac{n(n+1)}{2} (1-p)^{n-1} p\right) - \frac{1}{p}$$

$\neq 0$

$\therefore$ Bias exists

(f) MC convergence is based on the "$\underline{\text{Law of large numbers}}$"

Each element in sequence should be Identical & independent distributed random variables. This leads to point wise convergence.

— Q3 Done —