TD1

Artificial Intelligence - Introduction to Reinforcement Learning
ENSISA 2A

Ali El Hadi ISMAIL FAWAZ

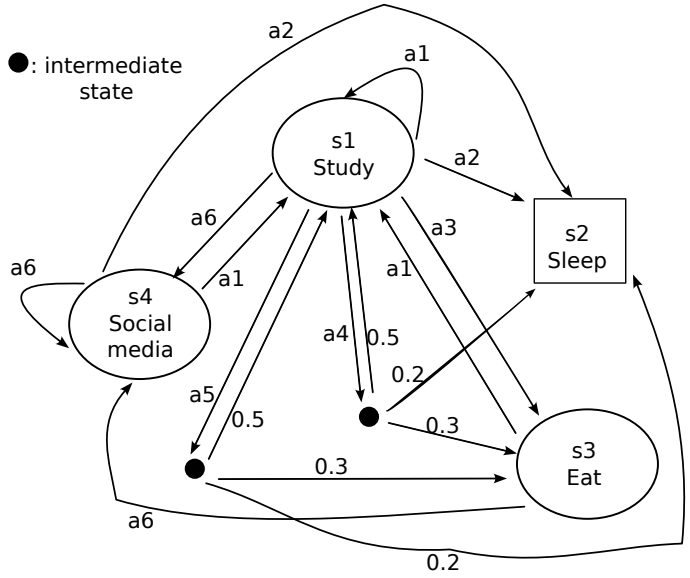
November 3, 2025

Problem I: Covid Lockdown MDP

Consider the state-transition diagram of your Covid lockdown routine that includes 4 states:

- State s_1: Study
- State s_2: Sleep (terminal state)
- State s_3: Eat
- State s_4: Social Media

Here is a figure representing the MDP:



For this same MDP, the following actions are possible to take with their corresponding reward:

- Action a_1: Study, reward = +1
- Action a_2: Sleep, reward = +2
- Action a_3: Eat, reward = +1
- Action a_4: Sport, reward = +1
- Action a_5: Chess, reward = +1
- Action a_6: Sleep, reward = −1

Example:
If you are at state s1, you can do the following:

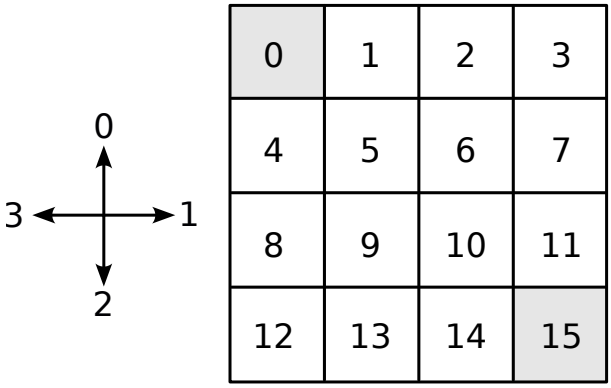
- Take action a2 and get to state s2 with a transition probability $P^{a_2}_{s_1 s_2} = 1$ and get a reward $R^{a_2}_{s1} = +2$
- Take action a_5, get reward $R^{a_5}_{s_1} = +1$ and go to:
  - state s_1 again with transition probability $P^{a_5}_{s_1 s_1} = 0.5$
  - state s_3 with transition probability $P^{a_5}_{s_1 s_3} = 0.3$
  - state s_2 with transition probability $P^{a_5}_{s_1 s_2} = 0.2$
- ...

Questions

1. For each action, define the state transition probability matrices
2. Recall the definition of the state-value function $V^{\pi}(s)$ following a policy π for a discounted MDP
3. Recall the Bellman expectation equation for the state-value function
4. Assuming a uniform random policy $\pi(a|s)$ that gives an equal probability for all actions **that could be taken** from state s. For instance, in the case of state s_1, given that all possible acions can be taken, then $\pi(a_i|s_1) = 1/6\ i \in \{1, 2, 3, 4, 5, 6\}$. Given the initial state-value function $V_1^{\pi}(s) = 2 \forall s$, apply one epoch of the policy evaluation to compute the new state-value function $V_2^{\pi}(s) \forall s$. Assume the discount factor $\gamma = 1$.
5. Apply one epoch of the policy improvement algorithm and in a greedy manner, compute the new policy π_2.
6. Recall the definition of the optimal state-value function $V*(s)$ of an MDP.
7. Recall the Bellman optimality equation for state-value function.
8. Starting with initial value function $V_1(s) = 2 \forall s$, apply one epoch using value iteration to compute $V_2(s) \forall s$, use discount factor $\gamma = 1$.
9. Is your new value function optimal ? Why ?

Problem II: Grid World

For this exercise, you will implement the two model-based dynamic programming algorithms: value iteration and policy iteration, to solve the Grid World problem. Consider having and nxn grid, example in Figure below when $n = 4$.



The position of an agent can be from squares 1 to 14 and the grey squares are the terminal state. The goal of the agent is to find the most optimal way to get to one of the two terminal states.

The actions that can be taken by the agent are:

- UP== 0
- RIGHT== 1
- DOWN== 2
- LEFT== 3

If one action throws the agent outside the grid, then its considered to leave the agent in his current position.
A reward of -1 is given to each step the agent takes until reaching a terminal state.
For this problem, consider the following parameters:

- Use γ discount factors of $1, 0.9$ and 0.8
- The stopping criterion is defined as $\max_s(|V_{new}(s) - V_{old}(s)) \leq \theta$
- Use $\theta = 10^{-4}$

Questions

1. Implement the value iteration algorithm to solve such problem. Return the optimal policy and corresponding value function.

2. Implement the policy evaluation and policy improvement algorithms. Return the optimal policy and corresponding value function.

Problem I:

1)

$$P = [P_{ss'}] = \begin{bmatrix} P_{s_1s_1} & P_{s_1s_2} & P_{s_1s_3} & P_{s_1s_4} \\ P_{s_2s_1} & P_{s_2s_2} & P_{s_2s_3} & P_{s_2s_4} \\ P_{s_3s_1} & P_{s_3s_2} & P_{s_3s_3} & P_{s_3s_4} \\ P_{s_4s_1} & P_{s_4s_2} & P_{s_4s_3} & P_{s_4s_4} \end{bmatrix}$$

s_1: study
s_2: sleep
s_3: eat
s_4: social media

1er 2ème

$$R_s^a = R^a \quad \forall s$$

a_1:

	s_1	s_2	s_3	s_4	
s_1	1	0	0	0	✓
s_2	0	1	0	0	✓
s_3	1	0	0	0	✓
s_4	1	0	0	0	✓

a_4:

	s_1	s_2	s_3	s_4	
s_1	0,5	0,2	0,3	0	✓
s_2	0	1	0	0	✓
s_3	0	0	1	0	✓
s_4	0	0	0	1	✓

a_2:

	s_1	s_2	s_3	s_4	
s_1	0	1	0	0	✓
s_2	0	1	0	0	✓
s_3	0	0	1	0	✓
s_4	0	1	0	0	✓

a_5:

	s_1	s_2	s_3	s_4	
s_1	0,5	0,2	0,3	0	✓
s_2	0	1	0	0	✓
s_3	0	0	1	0	✓
s_4	0	0	0	1	✓

a_3:

	s_1	s_2	s_3	s_4	
s_1	0	0	1	0	✓
s_2	0	1	0	0	✓
s_3	0	0	1	0	✓
s_4	0	0	0	1	✓

a_6:

	s_1	s_2	s_3	s_4	
s_1	0	0	0	1	✓
s_2	0	1	0	0	✓
s_3	0	0	0	1	✓
s_4	0	0	0	1	✓

2) State-value function : $V_\pi(s) = E_\pi[G_t \mid s_t = s]$

$G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots$

$\gamma \rightarrow$ discount factor

3) Bellman expectation equation for $V_\pi(s)$:

$V_\pi(s) = E_\pi[r_{t+1} + \gamma V_\pi(s_{t+1}) \mid s_t = s]$: variables

$= \sum_a \pi(a|s)\left[R^a + \gamma \sum_{s'} P^a_{ss'} V_\pi(s')\right]$

policy — action → a — state — reward — tous les s possible — probabilité de transition

Espérance → moyenne pondérée

4) $\pi(a|s)$ équiprobable

$s_1 \rightarrow 1/6 \quad s_2 \rightarrow 1 \quad s_3 \rightarrow 1/2 \quad s_4 \rightarrow 1/3$

$\boxed{V_1^\pi(s) = 2 \quad \forall s}$

- $V_2^\pi(s_1) = \sum_a \pi(a|s_1)\left[R^a + \gamma \sum_{s'} P^a_{ss'} V^\pi(s')\right]$ ($\sum_{s'} P^a_{ss'} = 1$)

$= \pi(a_1|s_1)\left[R^{a_1} + V_1^\pi(s_1)\right] + \pi(a_2|s_1)\left[R^{a_2} + V_1^\pi(s_2)\right] + \pi(a_3|s_1)\left[R^{a_3} + V_1^\pi(s_3)\right] + \pi(a_4|s_1)\left[R^{a_4} + P^{a_4}_{s_1 s_1} V_1^\pi(s_1) + P^{a_4}_{s_1 s_2} V_1^\pi(s_2) + P^{a_4}_{s_1 s_3} V_1^\pi(s_3)\right] +$

($1/6$; reward ; $P^{a_1}_{s_1 s_1} = 1$; $P^{a_2}_{s_1 s_2} = 1$; initialisation pour 1ère itération ; 4 actions)

$\pi(a_5|s_1)\,[R^{a_5} + P^{a_5}_{s_1 s_1} V_1^\pi(s_1) + P^{a_5}_{s_1 s_2} V_1^\pi(s_2) + P^{a_5}_{s_1 s_3} V_1^\pi(s_3)] +$
$\pi(a_6|s_1)\,[R^{a_6} + V_1^\pi(s_4)]$
$= \frac{17}{6}$

- $V_2(s_2) = V_1(s_2) = 2$ ← car on ne peut pas en sortir

- $V_2(s_3) = \pi(a_1|s_3)\,[R^{a_1} + V_1(s_1)] +$
$\pi(a_6|s_3)\,[R^{a_6} V_1(s_4)]$
$= 2$

- $V_2(s_4) = 8/3$ ← même calcul

5) argmax : [1, 5, −1, 10] max = 10
i = 0, 1, 2, 3 argmax = 3

- $R^{a_1} + V_2^\pi(s_1) = 1 + 17/6 = 23/6 \simeq 3{,}83$
$R^{a_2} + V_2^\pi(s_2) = 2 + 2 = 4$
$R^{a_3} + V_2^\pi(s_3) = 1 + 2 = 3$
$R^{a_4} + P^{a_4}_{s_1 s_1} V_2^\pi(s_1) + P^{a_4}_{s_1 s_2} V_2^\pi(s_2) + P^{a_4}_{s_1 s_3} V_2^\pi(s_3) = 1 + 0{,}5 \cdot \frac{17}{6} + 0{,}2 \cdot 2 + 0{,}3 \cdot 2 = 41/12 \simeq 3{,}42$

$R^{a_5} + P^{a_5}_{s_1 s_1} V_2^\pi(s_1) + P^{a_5}_{s_1 s_2} V_2^\pi(s_2) + P^{a_5}_{s_1 s_3} V_2^\pi(s_3) = 1 + 0{,}5 \cdot \frac{17}{6} + 0{,}2 \cdot 2 + 0{,}3 \cdot 2 = 41/12 \simeq 3{,}42$

$R^{a_6} + V_2^\pi(s_4) = -1 + 8/3 = 5/3 \simeq 1{,}67$

↳ max = 4 ⇒ argmax (2)

$\pi_2(s_1) = a_2$

$R^{a_1} + V_2(s_1) = 1 + 17/6 = 23/6 \simeq 3,83$

$R^{a_2} + V_2(s_4) = -1 + 8/3 = 5/3 \simeq 1,67$

↳ $\max = 3,83 \Rightarrow \text{argmax}(1)$

$\pi_2(s_3) = a_1$

$\pi_2(s_4) = a_2$ ← même chose

6) $V^*(s) = \max_\pi V^\pi(s)$

↑ ↑ maximise une policy pour $V^\pi(s)$

State-value function optimale

7) $V^*(s) = \max_a \left[R^a + \gamma \sum_{s'} P^a_{ss'} V^\pi(s') \right]$

8)
$1 + 2 = 3$
$2 + 2 = 4$
$1 + 2 = 3$
$1 + 2 = 3$
$1 + 2 = 3$
$-1 + 2 = 1$

$\max = 4 = V_2^*(s_1)$

$V_2^*(s_2) = \emptyset$
$V_2^*(s_3) = 3$
$V_2^*(s_4) = 4$

⇒ Si on refait une autre itération et que les valeurs restent les mêmes ⇒ optimalité

g) $V_3^*(o_1) = 5 \neq V_3^*(o_1) = 4$ ⇒ pas encore optimal