Markov Decision Process

Link to the video can be found here
The lecture notes can be found here

Introduction

- All RL problems can be formalised as MDPs
- MDPs formally describe an environment for Reinforcement Learning

Markov Property

> A state S is said to have the *Markov Property* if
>
> - The future is independent of the past given the present
> - If the current state is known, the past can be thrown away

State Transition Matrix

> - $P_{ss'} = P[S_{t+1} = s' \mid S_t = s]$
> - **State transition matrix** defines transition probabilities from all states ***s*** to all successor states ***s'***.
> - Each row defines a different start state and subsequent probabilities for all successor states starting from that state

Markov Process

> A *Markov Process* is a sequence of random states $S_1, S_2 \ldots$ that satisfy the *Markov Property*.

Markov Reward Process

- A *Markov Rewared Process* is a Markov chain with values

> A *Markov Reward Process* is a tuple $\langle S, P, R, \gamma \rangle$
>
> - S is a finite set of states
> - P is a State Transition Probability Matrix
> $P_{ss'} = \mathbb{P}[S_{t+1} = s' \mid S_t = s]$
> - R is a Reward function, $R_s = \mathbb{E}[R_{t+1} \mid S_t = s]$
> - γ is a discount factor, $\gamma \in [0, 1]$

- The reward function just gives us the reward at the current state
- In *Reinforcement Learning* we care about maximizing the cumulative sum of these rewards.

Return

> The *return* G_t is the total discounted reward from time-step t
>
> $$G_t = R_1 + \gamma R_2 + \gamma^2 R_3 + \ldots = \sum_{k=0}^{\infty} \gamma^k R_k$$

> - $\gamma \in [0, 1]$ tells us the present value of future rewards

- We favour immediate rewards compared to later ones

Value Function

> The state *value function* $v(s)$ of an MRP is the expected return starting from state s
>
> $v(s) = \mathbb{E}[G_t \mid S_t = s]$
>
> - You can think of it as averaging over a bunch of possible random outcomes starting from a particular state

Bellman Equation for MRPs

> We can break down the value function into immediate reward R_{t+1} and discounted value of successor state $\gamma v(S_{t+1})$
>
> $v(s) = \mathbb{E}[R_{t+1} + \gamma v(S_{t+1}) \mid S_t = s]$

- Prove for yourself, it's fairly easy, you just have to substitute the definition of each term

> Using State Transition Matrix this can be written as $v = R + \gamma P v$

Markov Decision Process

- A *Markov Decision Process* is a *Markov Reward Process* with decisions.

> A *Markov Decision Process* is a tuple $\langle S, A, P, R, \gamma \rangle$
>
> - S is a finite set of states
> - A is a finite set of actions
> - P is a State Transition Probability Matrix
> $P^a_{ss'} = \mathbb{P}[S_{t+1} = s' \mid S_t = s, A_t = a]$
> - R is a Reward function, $R^a_s = \mathbb{E}[R_{t+1} \mid S_t = s, A_t = a]$
> - γ is a discount factor, $\gamma \in [0, 1]$

- This now provides us our *environment* where depending on one of the finite *actions* we are able to take *decisions*.

Policy

- A policy π is distribution over actions given states. i.e if you are in state s, π defines the probability distribution over all the possible actions a

> $\pi(a \mid s) = \mathbb{P}[A_t = a \mid S_t = s]$

- The policy fully defines the *behaviour* of our agent

Value function for an MDP

1. State-value function

2. Action-value function

> **State-Value function**
>
> It is the expected return starting from a state s and then following a policy π
> $v_\pi(s) = \mathbb{E}_\pi[G_t \mid S_t = s]$

> **Action-Value function**
>
> It is the expected return starting from a state s, taking an action a and then following a policy π
> $q_\pi(s,a) = \mathbb{E}_\pi[G_t \mid S_t = s, A_t = a]$

Bellman Equations

> $v_\pi(s) = \mathbb{E}_\pi[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S_t = s]$
> $q_\pi(s,a) = \mathbb{E}_\pi[R_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1}) \mid S_t = s, A_t = a]$

Optimal Value Function

> Maximum value function over all policies
> $v_*(s) = max(v_\pi(s))$
> $q_*(s,a) = max(q_\pi(s,a))$
>
> - So solving MDP would be same as finding the *optimal value function*