

## RL Lecture 1 (David Silver)

- History: Sequence of observations, actions, and rewards since beginning of episode
- State  $\rightarrow$  information used to determine what happens next
  - State is a function of History
  - $S_t = f(H_t)$
- Three definitions of state
  - Environment state
    - Environment's private representation
    - Usually not visible to the agent
    - Algorithms cannot depend on environment state
    - May contain information irrelevant to our task
  - Agent state is agent's internal representation of environment
    - Agent's own internal representation
    - This is the information used by RL algorithm
    - Can be function of history
  - Third (more mathematical definition)
- Information state (Markov state) contains all useful information from the history
  - "Future is independent of the past given the present"
  - Environment state is Markov
  - History  $H_t$  is Markov

...ion from the history. ... state) contains all useful

**Definition**

A state  $S_t$  is **Markov** if and only if

$$\mathbb{P}[S_{t+1} \mid S_t] = \mathbb{P}[S_{t+1} \mid S_1, \dots, S_t]$$

### Case 1:

Full observability: agent directly observes environment state

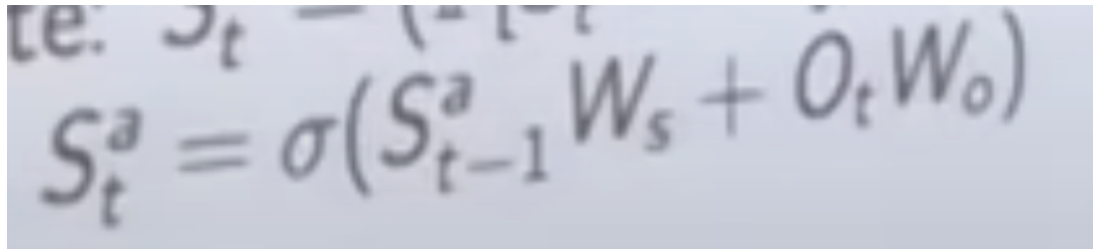
Observation = Agent state = Environment state

This is formally called a Markov Decision Process (MDP)

### Case 2:

Partial Observability: Agent indirectly observes environment

- Robot with camera vision isn't told its actual location
- Trading agent only observes current prices
- Poker playing agent only observes public cards
- Agent state  $\neq$  environment state
- POMDP: Partially observable Markov decision process
- Agent must construct its own state representation
  - Complete history:  $S_t = H_t$
  - Beliefs of environment state:  $S_t = (P[S_t = s_1] \dots P[S_t = s_n])$
  - Recurrent neural network:



A photograph of a handwritten equation on a light blue background. The equation is  $S_t^a = \sigma(S_{t-1}^a W_s + O_t W_o)$ . The text is written in a cursive, handwritten style. Above the main equation, there is some faint, partially visible text that appears to be "te.  $S_t = (s_1, \dots, s_n)$ ".

Inside an RL Agent

- Policy: Agent's behavior function
- Value function: how good is each state and/or action
- Model: agent's representation of environment

Policy is agent's behavior

- Deterministic or stochastic policy

Value function is prediction of future reward given state

$$v_{\pi}(s) = \mathbb{E}_{\pi} [R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \dots \mid S_t = s]$$

Model:

- Predicts what the environment will do next
- Usually split into two parts

Transitions:

- $P$  predicts the next state (i.e. environment dynamics)

Rewards:

- $R$  predicts the next (immediate) reward e.g.
- Note: one-step reward

$$\mathcal{P}_{ss'}^a = \mathbb{P}[S' = s' \mid S = s, A = a]$$
$$\mathcal{R}_s^a = \mathbb{E}[R \mid S = s, A = a]$$

## RL Taxonomy

### Value based

- Value Function
- No Policy

### Policy based

- Explicit Policy function
- No value function

### Actor Critic (hybrid approach)

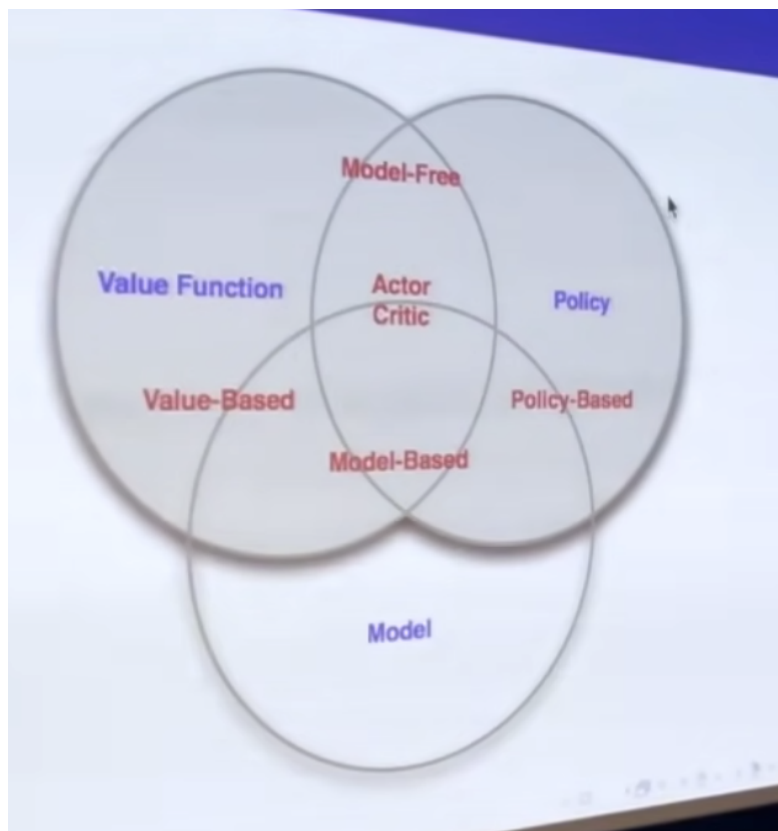
- Policy
- Value function

### Model Free

- Policy and/or value function
- No explicit model

### Model based

- Build dynamics model first
- Also use policy and/or value function



## Reinforcement Learning

- Environmentally is initially unknown
- Agent interacts with the environment
- Agent improves its policy

## Planning

- A model of the environment is known
- Agent performs computations with its model (without any external interaction)

## Exploration vs Exploitation

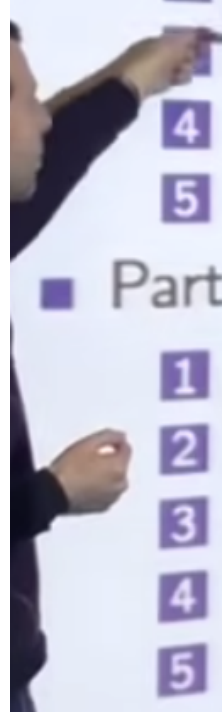
- RL is essentially trial-and-error learning
- Exploration finds more information about environment
- Exploitation exploits information to maximize reward

## Prediction vs Control

- Prediction: evaluate the future
  - given a policy
  - What is the expected reward given a certain policy?
- Control: optimize the future
  - Find the best policy
  - What is the best policy to maximize total expected reward?

Often times, we will need to solve the control problem IN ORDER TO solve the prediction problem

Rest of the course:

- 
- A person's arm and hand are visible on the left side of the slide, pointing towards the third item in the first list.
- Part I: Elementary Reinforcement Learning
    - 1 Introduction to RL
    - 2 Markov Decision Processes
    - 3 Planning by Dynamic Programming
    - 4 Model-Free Prediction
    - 5 Model-Free Control
  - Part II: Reinforcement Learning in Practice
    - 1 Value Function Approximation
    - 2 Policy Gradient Methods
    - 3 Integrating Learning and Planning
    - 4 Exploration and Exploitation
    - 5 Case study - RL in games