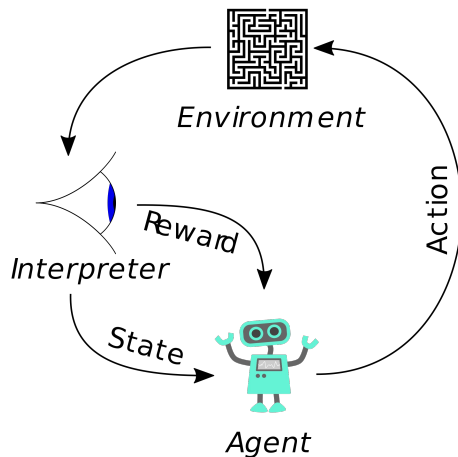


Chapter 1: Introduction

Ryan Lee

Reinforcement Learning

- Learning by interacting with the environment
- Goal: maximize a numerical reward signal by choosing correct actions
 - **Trial and error:** learner is not told the best action
 - **Delayed rewards:** actions can affect all future rewards



vs. Supervised and Unsupervised Learning

- No external supervisor / teacher
 - No training set with labeled examples (answers)
 - Need to interact with environment in uncharted territories
- Different goals
 - Supervised Learning: Generalize existing data to minimize test set error
 - Reinforcement Learning: Maximize reward through interactions
 - Unsupervised Learning: Find hidden structure

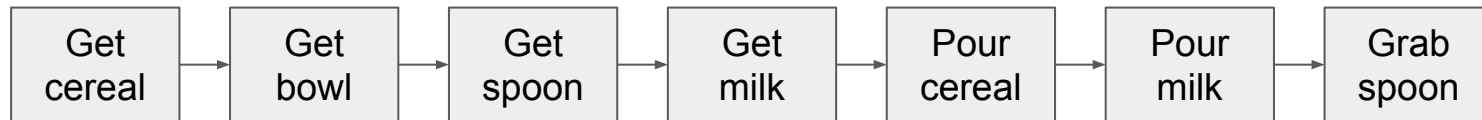
→ Reinforcement Learning is a new paradigm of Machine Learning

Characteristics of Reinforcement Learning

- Interactions between agent and environment
- Uncertainty about the environment
 - Effects of actions cannot be fully predicted
 - Monitor environment and react appropriately
- Defined goal
 - Judge progress through rewards
- Present affects the future
 - Effect can be delayed
- Experience improves performance

Example: Preparing Breakfast

- Complex sequence of interactions to achieve goal



- Need to observe and react to the uncertainty of the environment
 - Grab different bowl if current bowl is dirty
 - Stop pouring if the bowl is about to overflow
- Actions have delayed consequences
 - Failing to get spoon does not matter until you start eating
- Experience improves performance

Exploration vs Exploitation

- **Exploration:** Try different actions
- **Exploitation:** Choose best known action
- Need both to obtain high reward

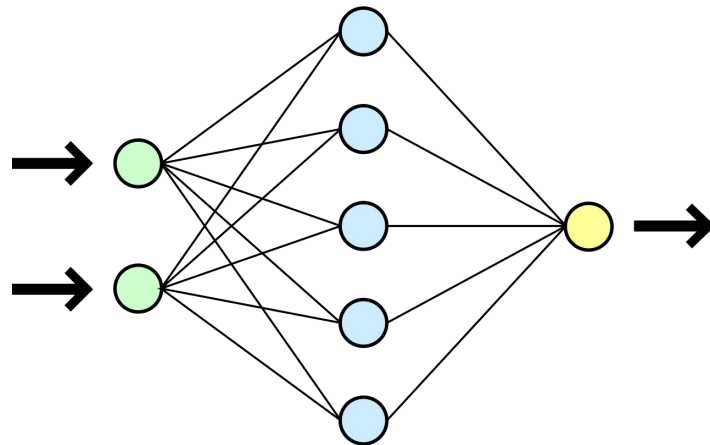
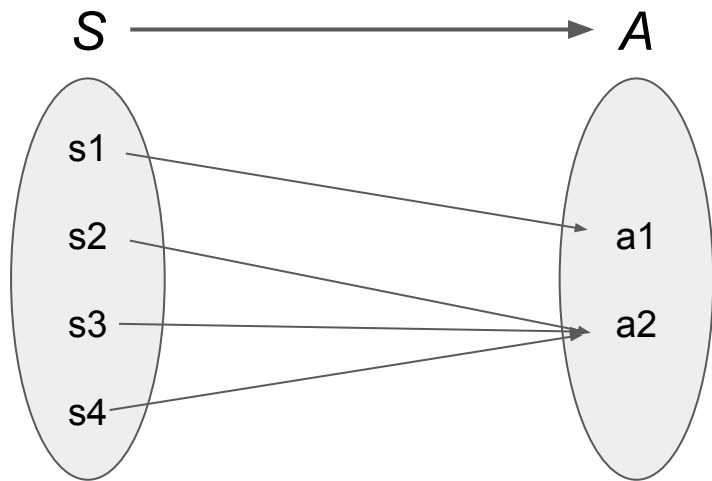


Elements of Reinforcement Learning

- **Policy** defines the agent's behavior
- **Reward Signal** defines the goal of the problem
- **Value Function** indicates the long-term desirability of state
- **Model** of the environment mimics behavior of environment

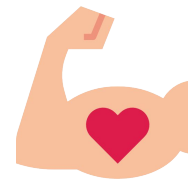
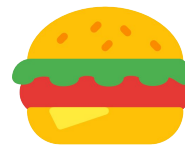
Policy

- Mapping from *observation* to *action*
- Defines the agent's behavior
- Can be stochastic



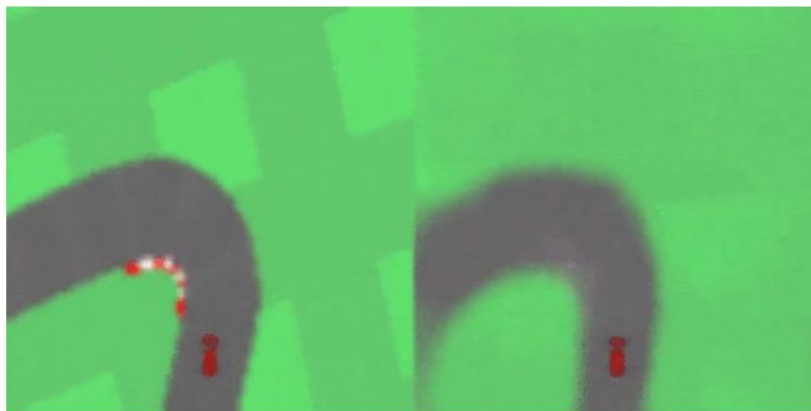
Reward Signal vs. Value Function

- Reward
 - Immediate reward of action
 - Defines good/bad events for the agent
 - Given by the environment
- Value Function
 - Sum of future rewards from a state
 - Long-term desirability of states
 - Difficult to estimate
 - **Primary basis of choosing action**



Model

- Mimics the behavior of environment
- Allow *planning* a future course of actions
- Not necessary for all RL methods
 - *Model-based* methods use the model for planning
 - *Model-free* methods only use trial-and-error

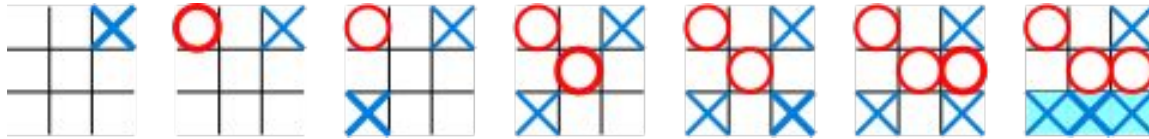


Actual observations from the environment.

What gets encoded into z_t .

Example: Tic Tac Toe

- Assume **imperfect opponent**
- Agent needs to find and exploit imperfections



Tic Tac Toe with Reinforcement Learning

- Initialize value functions to 0.5 (except terminal states)
- Learn by playing games
 - Move *greedily* most times, but *explore* sometimes
- Incrementally update value functions by playing games
- Decrease learning rate over time to converge

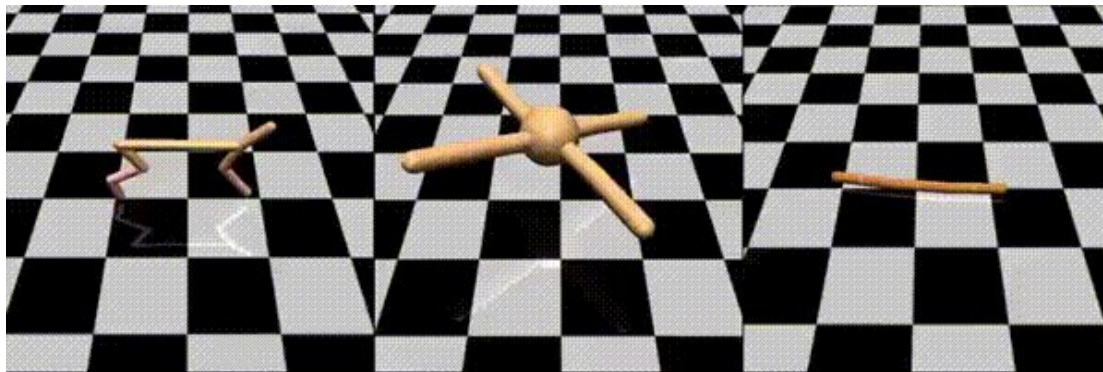
$$V(s) \leftarrow V(s) + \alpha[V(s') - V(s)]$$

Tic Tac Toe with other algorithms

- Minimax algorithm
 - Assumes best play for opponent → Cannot exploit opponent
- Classic optimization
 - Require complete specification of opponent → Impractical
 - ex. Dynamic Programming
- Evolutionary methods
 - Finds optimal algorithm
 - Ignores useful structure of RL problems
 - Works best when good policy can be found easily

Reinforcement Learning beyond Tic-Tac-Toe

- Can be applied to:
 - more complex games (ex. Backgammon)
 - problems without enemies (“games against nature”)
 - problems with partially observable environments
 - non-episodic problems
 - continuous-time problems



Thank you!

Original content from

- [Reinforcement Learning: An Introduction by Sutton and Barto](#)

You can find more content in

- [github.com/seungjaeryanlee](#)
- [www.endtoend.ai](#)