Lecture 1: Introduction to Reinforcement Learning

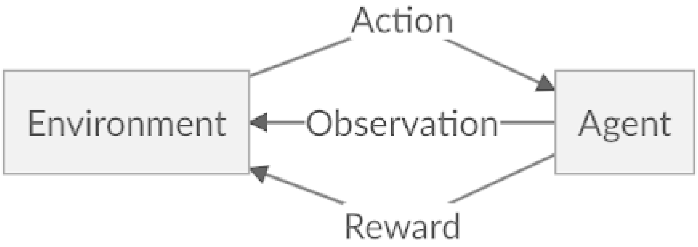
What is reinforcement learning?

Not supervised learning

- There's no supervisor, only a received **reward signal** (that can be delayed).
- Possibly no pre-existing dataset; agent's actions influence environment/data dynamically (*read*: sequential, non-I.I.D.).

Concepts

- **Reward** R_t: Scalar feedback signal at time t. The agent's goal is to maximize cumulative reward.
- **History**: Sequence of **actions** A_t, **observations** O_t, and **rewards** R_t up to time t. At each step t:



But, history can be noisy. It might not contain useful information or be relevant.

States

States are information used by the agent to determine the next action.

$$S_t = f(H_t)$$

Real quick, a state is **Markov** if and only if

$$P[S_{t+1}|S_t] = P[S_{t+1}|S_1, \ldots, S_t]$$

That is, *the future is independent of the history, given the present*. Technically, the entire history is a Markov state, albeit not a very useful one.

The **environment state** S_t^e is a private representation of data used by the environment to make decisions. It's usually not visible to the agent, and is Markov.

The **agent state** S_t^a is a representation of data used by the agent to pick the next action. This is used by RL algorithms.

Environments

In a **fully observable environment**, the agent observes the complete environment. This is also known as a **Markov Decision Process** (MDP).

$$O_t = S_t^a = S_t^e$$

In a **partially observable environment**, the agent indirectly observes the environment and creates its own S_t^a. This is also known as a **Partially Observable Markov Decision Process** (POMDP).

Reinforcement learning agents

A **policy** is an agent's behavior function, which maps a *state* to an *action*.

- Deterministic: $a = \pi_s$
- Stochastic: $\pi(a|s) = P[A_t = a|S_t = s]$

A **value function** is a prediction of future reward. How good is each state and/or action?

A **model** is an agent's representation of an environment. There are two models:

- **Transition** models predict the next state given current reward.
- **Reward** models predict the next reward given current state.

Types of agents

- **Value-based** agents have a value function and implicit policy.
- **Policy-based** agents don't have a value function.
- **Actor-critic** agents have both a value function and policy.

Types of RL problems

- **Model-free** learning don't explicitly understand an environment (no model). They directly use a value function or policy.
- **Model-based** learning builds up a model initially.

Reinforcement learning subproblems

- **Learning**: The environment/model is unknown. The agent interacts with an environment and improves its policy.
- **Planning**: The environment/model is fully known. The agent performs computations (no external interaction).
- **Explore/exploit**: I really like Chapter 2 of Algorithms to Live By for the intuition behind this.

Reference

- Video lecture
- Slides