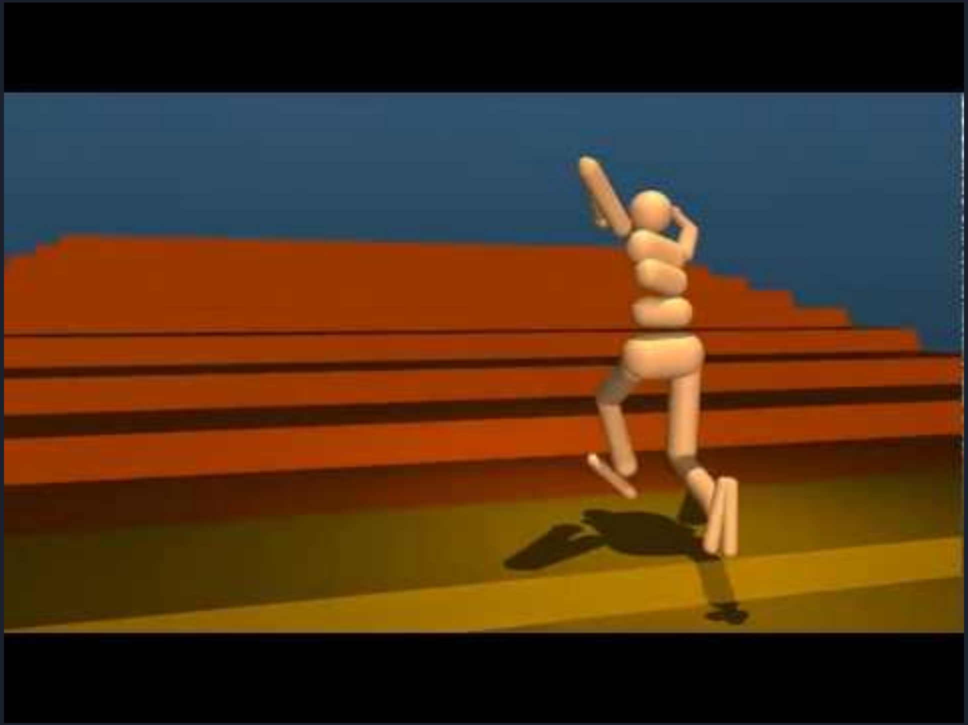
Deep Reinforcement Learning

Reinforcement Learning

Applications

Applications

Applications



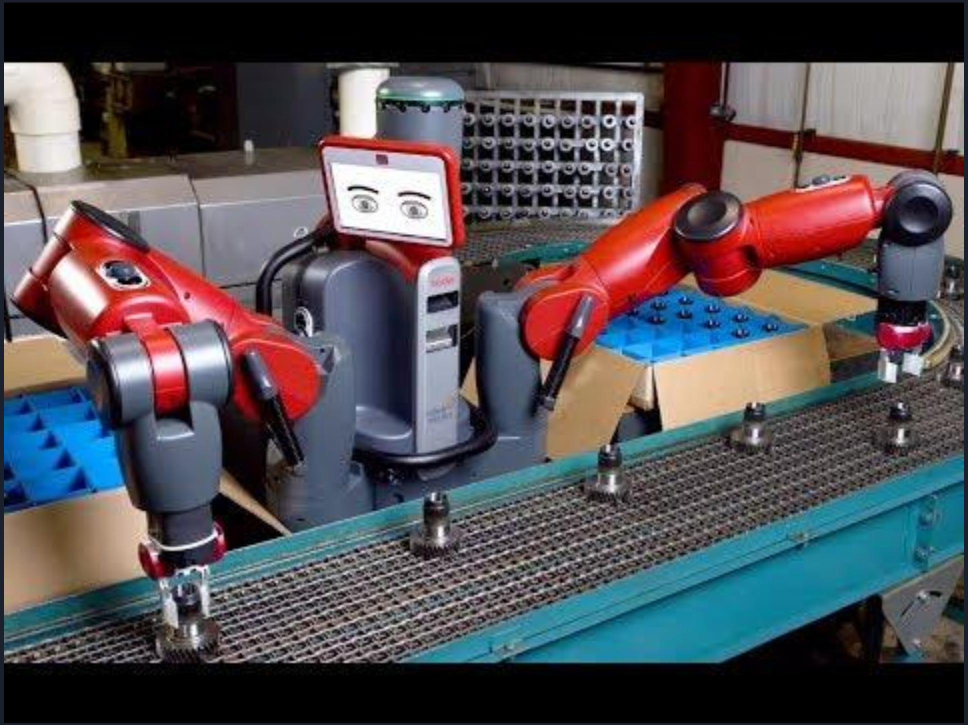
Applications

Applications

Applications

Applications

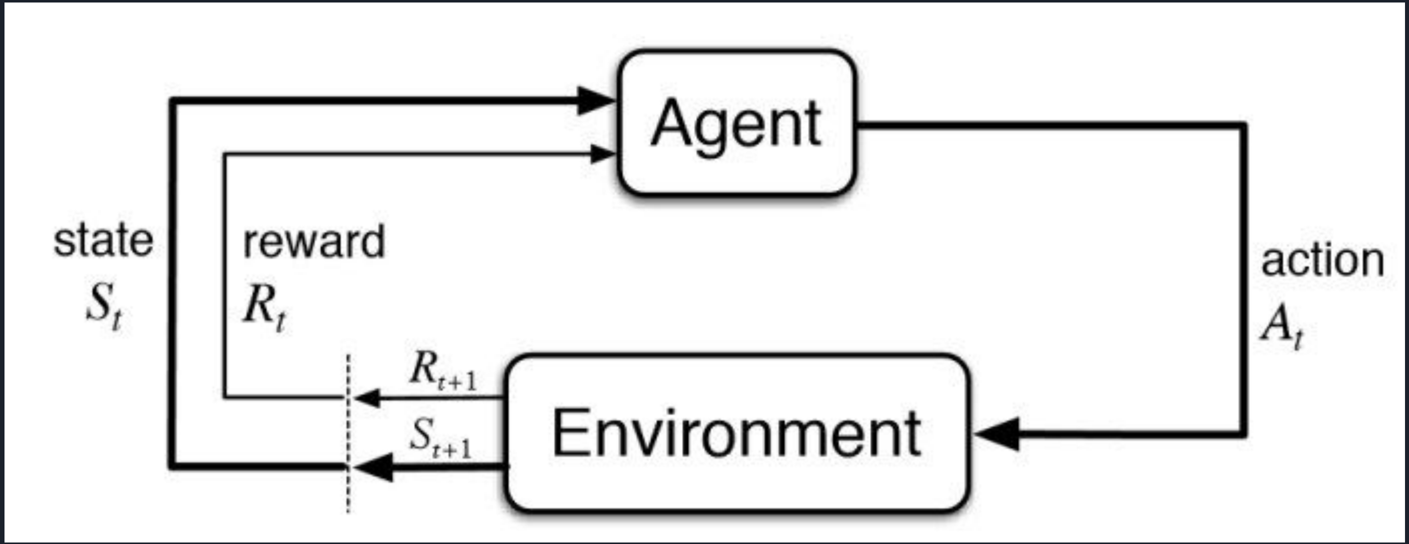
Applications



Problems with Supervised, Unsupervised Learning

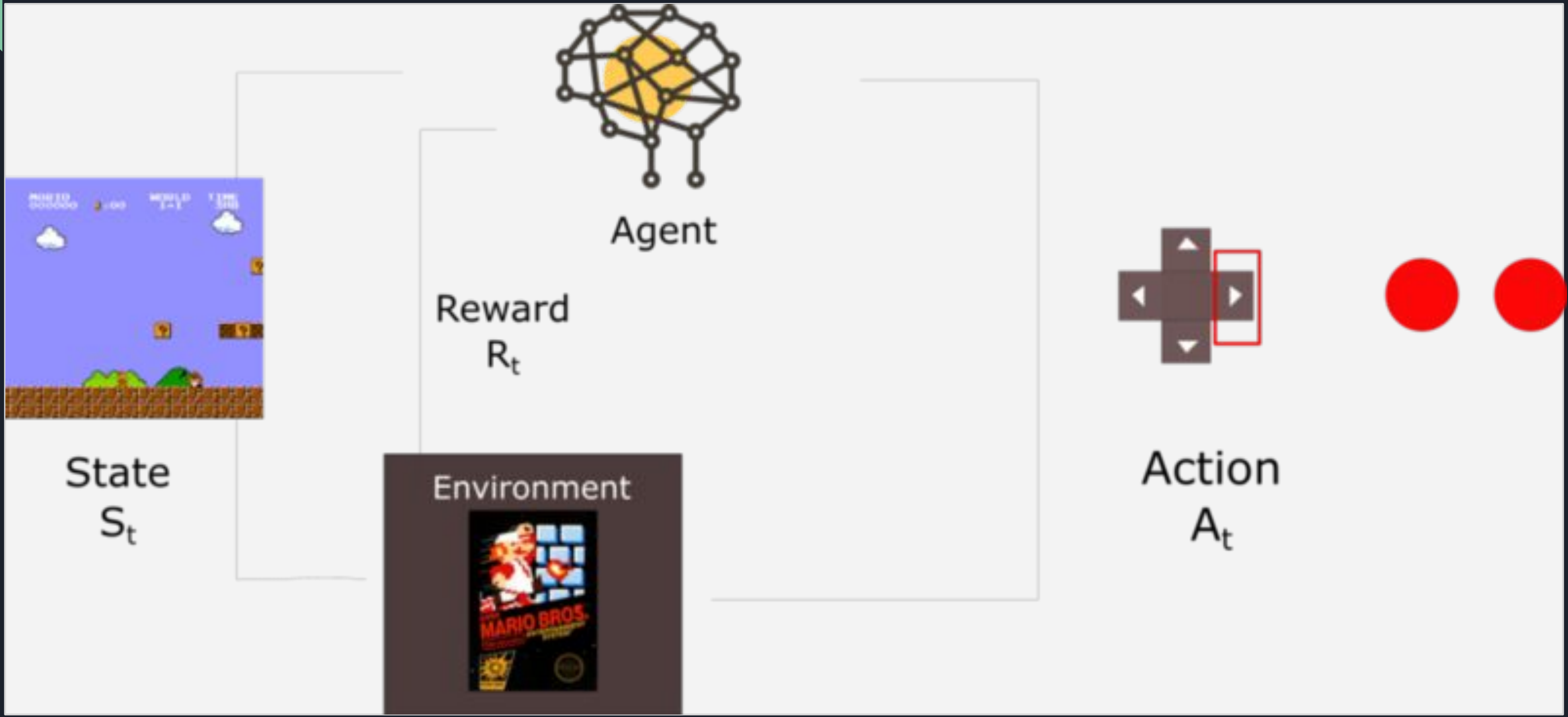
- Training set of labeled examples - data dependant.
- Description of a situation.
- Tries to generalize.
- For interactive tasks, it is impractical to obtain examples of desired behavior of all situations. An agent must be able to learn from its own experience.
- Finding structure in unlabeled data.

Reinforcement Learning

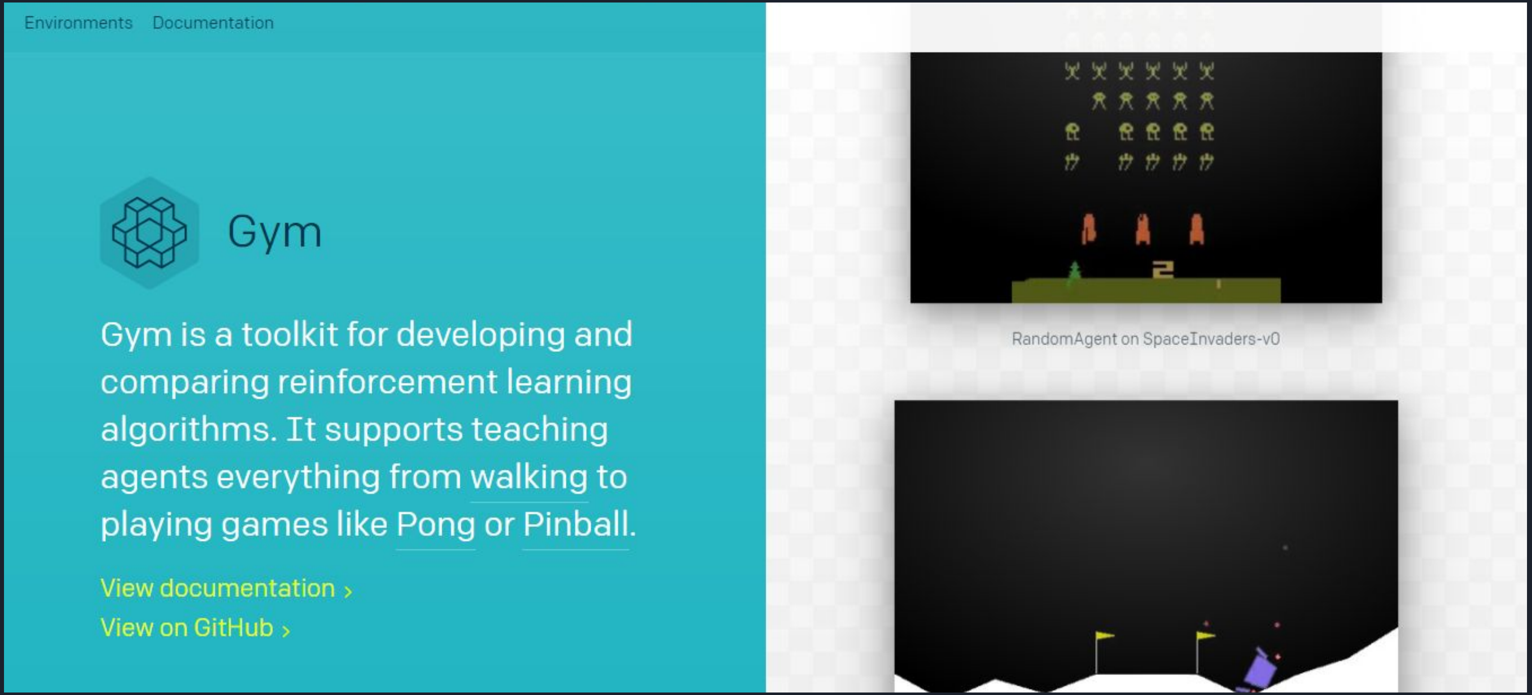


A policy can be defined agent’s way of behaving at a given time: finding an optimal policy is the key.

Reinforcement Learning



Environments



Environment tasks

- A task is an instance of a Reinforcement Learning problem.
- Episodic tasks:
  - Have a starting point and an ending point (a terminal state).
  - This creates an episode: a list of States, Actions, Rewards, and New States.
- Continuous tasks:
  - These are tasks that continue forever (no terminal state).
  - The agent chooses the best actions and while interacting with the environment.

Environments

The reward hypothesis

All goals can be described as the outcome of maximizing a cumulative reward.

To have the best behavior, we need to maximize the expected cumulative reward.

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \, where \gamma \in [0, 1)$$

$$R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} \ldots$$

RL Learning

Monte Carlo Approach

Collecting the rewards at the end of the episode and then calculating the maximum expected future reward

Temporal Difference

Estimate the rewards at each step

Exploration/Exploitation trade off

- Exploration is finding more information about the environment.
- Exploitation is exploiting known information to maximize the reward.
- If we only focus on reward, our agent may never reach the expected objective.
    - Instead, it will only exploit the nearest source of rewards.
- If our agent does a little bit of exploration, it can find a bigger reward, and reach the objective.

RL Types

- Value based - Q Learning
  - The goal is to optimize the value function V(s).
  - The value function is a function that tells us the maximum expected future reward the agent will get at each state.

- Policy based - Policy Gradients
  - In policy-based RL, we want to directly optimize the policy function π(s) without using a value function.
  - The policy is what defines the agent behavior at a given time.