# Reinforcement Learning

## Tutorial on OpenAI Gym environment

Gaurav Mahajan

University of California, San Diego

# Goals

- Implement state of art algorithms
- Achieve deeper understanding on current issues
- Identify canonical models and complexity

# Preliminaries: RL Paradigm

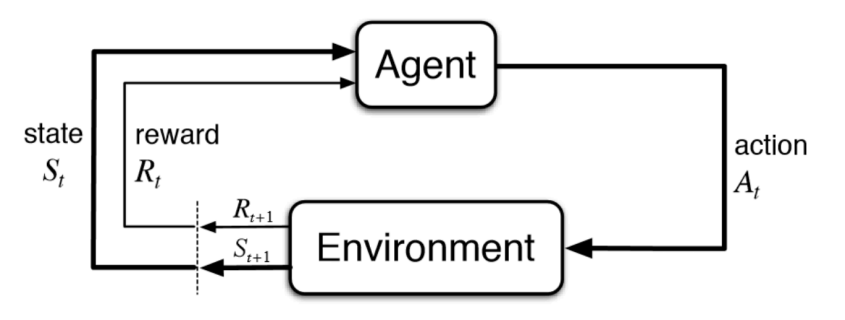


**Figure 1:** At each time t, agent observes state of environment $s_t$ and chooses to perform action $a_t$. As a result, environment changes state from $s_t$ to $s_{t+1}$ and agent receives reward $r_{t+1}$. The goal of agent is to maximize $E(\sum_{i=0}^{\infty} r_i)$. Sometimes future rewards are discounted by a factor $\gamma < 1$

# Open AI Gym

**Algorithm 1** Semantics in Open AI gym to evaluate a policy

$s_0 \leftarrow env.reset()$
$t \leftarrow 0$
**while** True **do**
  $a_t \leftarrow \pi(s_t)$
  $s_{t+1}, r_t, done \leftarrow env.step(a_t)$
  **if** done **then**
    break
  **end if**
  $t \leftarrow t + 1$
**end while**
**return** $\sum r_t$

# Value Bellman Equation

Given real-valued function $V\colon S \to \mathbb{R}$ such that for $\forall s \in S$ satisfies the equation

$$V^*(s) = \max_{a \in A} \sum_{s'} P(s'|s,a) \left[ R(s,a,s') + \gamma V^*(s') \right] \tag{1}$$

immediately leads to an optimal policy

$$\pi^*(s) = arg \max_{a} \sum_{s'} P(s'|s,a) \left[ R(s,a,s') + \gamma V^*(s') \right] \tag{2}$$

# Q-Value Bellman Equation

Given real-valued function $Q\colon S \times A \to \mathbb{R}$ such that for $\forall s, a \in S \times A$ satisfies the equation

$$Q^*(s,a) = \sum_{s'} P(s'|s,a) \left[ R(s,a,s') + \gamma \max_{a' \in A} Q^*(s',a') \right] \tag{3}$$

immediately leads to an optimal policy

$$\pi^*(s) = arg \max_a Q^*(s,a) \tag{4}$$

# Code examples

- PyTorch RL examples
- Baselines in TensorFlow
- Baselines in PyTorch

Questions?