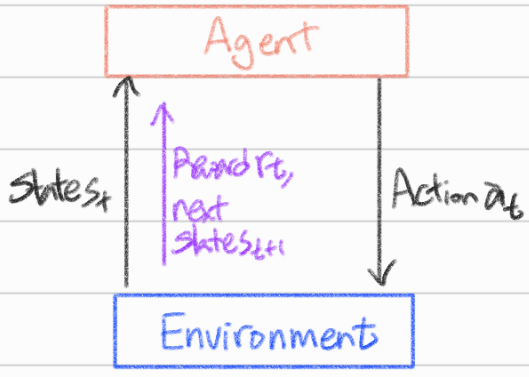
Deep Reinforcement Learning 강화학습

Problems involving an agent interacting with an environment, which provides nemeric reward signals

Goal: Learn how to take actions in order to maximize reward



Cart-Pole prob. Robot Locomotion Go(바둑)

Markov Decision Process

- At time step $t=0$, env samples initial state $s_0 \sim p(s_0)$
- Then, for $t=0$ until done:
  - Agent selects action a_t
  - Env samples reward $r_t \sim R(.|s_t, a_t)$
  - Env samples next state $s_{t+1} \sim P(.|s_t, a_t)$
  - Agent receives reward r_t and next state s_{t+1}

A policy π is a function from S to A that specifies what action to take in each state

Objective: find policy π^* that maximizes cumulative discounted reward: $\sum_{t \geq 0} \gamma^t r_t$

Q-network : Experience Replay

ex. 벽돌깨기 게임 Atari Breakout

Policy Gradients

Q-function can be very complicated, so directly find out Policy Gradients

Actor-Critic Algorithm

Reinforce in action : Recurrent Attention Model

objective : Image Classification

State - glimpse

Action - where to look

∴ Computational efficiency