REINFORCEMENT LEARNING

강화학습특론

-1주차-

Reinforcement Learning Overview

03

Reinforcement Learning: Major Components

학습내용

- Major Components: Policy, Value Function & Model
- Maze Example
- Categorizing Reinforcement Learning Agents
- Learning and Planning
- Exploration and Exploitation
- Prediction and Control

03 Reinforcement Learning: Major Components

Major Components: Policy, Value Function & Model

- An RL agent may include one or more of these components:

Policy	agent's behavior function
Value Function	how good is each state (or state & action)
Model	agent's representation of the environment

Major Components: Policy

- A policy is the agent's behavior
- It is a mapping function from state to action, e.g.

Deterministic policy

Stochastic policy

Major Components: Value Function

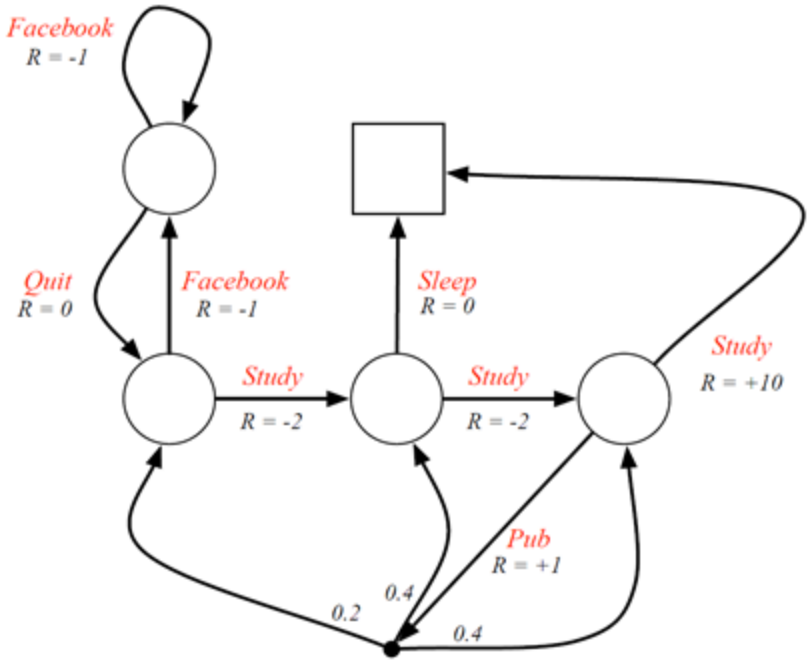
- Value function is a prediction of future cumulated rewards
- Can be used to evaluate the goodness/ badness of states
- And decide how to act by comparing

$$v_\pi(s) = \mathbb{E}_\pi[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots \mid S_t = s]$$

where:

- π represents the policy
- E_π is the expectation for the policy π
- γ is the discount factor

Major Components: Model



- Model predicts what the environment will do next
- predicts the next state
- predicts the next (immediate) reward

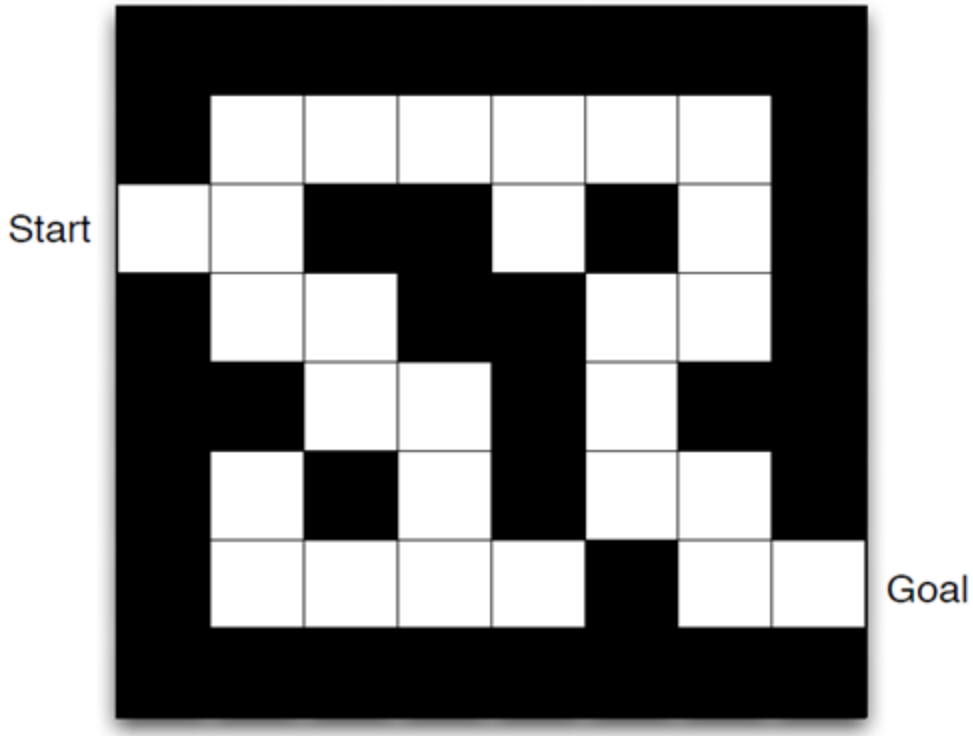
$$\mathcal{P}_{ss'}^{a} = \mathbb{P}[S_{t+1} = s' \mid S_t = s, A_t = a]$$

$$\mathcal{R}_{s}^{a} = \mathbb{E}[R_{t+1} \mid S_t = s, A_t = a]$$

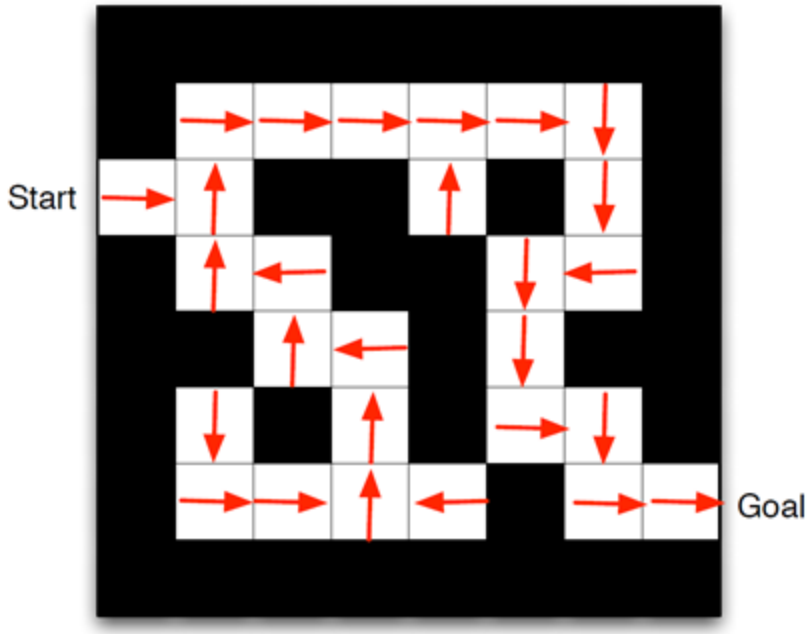
03 Reinforcement Learning: Major Components

Maze Example

- States : Agent's location
- Actions : up, down, left, right
- Rewards : -1 per time-step

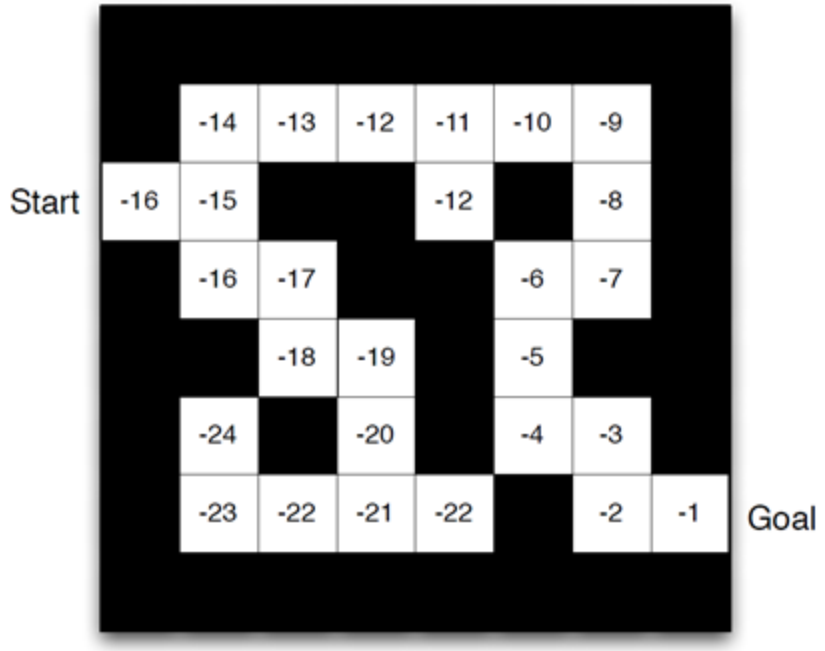


Maze Example: Policy



- Arrows represent the action that needs to be taken for each state
- These actions originate from a single policy
- A different policy might give rise to a different action for the same state

Maze Example: Value Function



- Numbers represent the value of each state
- In this example, the numbers represent the number of steps that would be required from each state to reach the goal
- These numbers provide an idea about the goodness of a state

03 Reinforcement Learning: Major Components

Categorizing RL agents (1/3)

1 Value Based

- No Policy (implicit)
- Value Function

2 Policy Based

- Policy
- No Value Function

Categorizing RL agents (2/3)

3 Actor Critic

- Policy
- Value Function

4 Model-free

- Policy and/or Value Function
- no Model

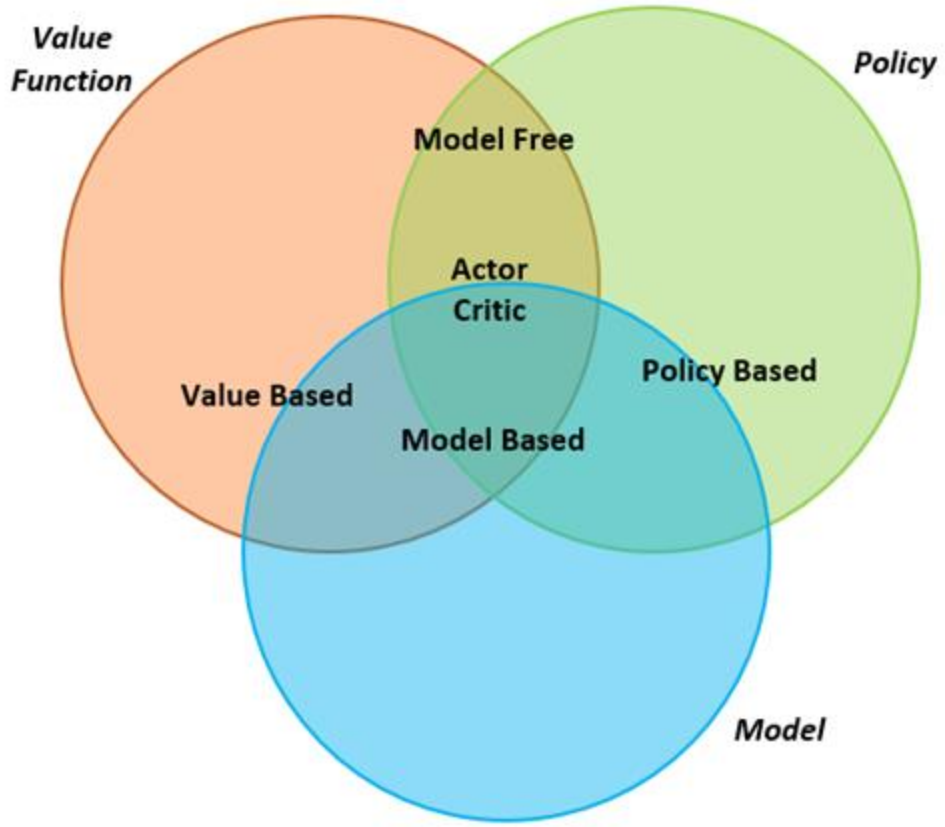
Reinforcement Learning: Major Components

Categorizing RL agents (3/3)

5. Model-based
   - Policy and/or Value Function
   - Model

03 Reinforcement Learning: Major Components

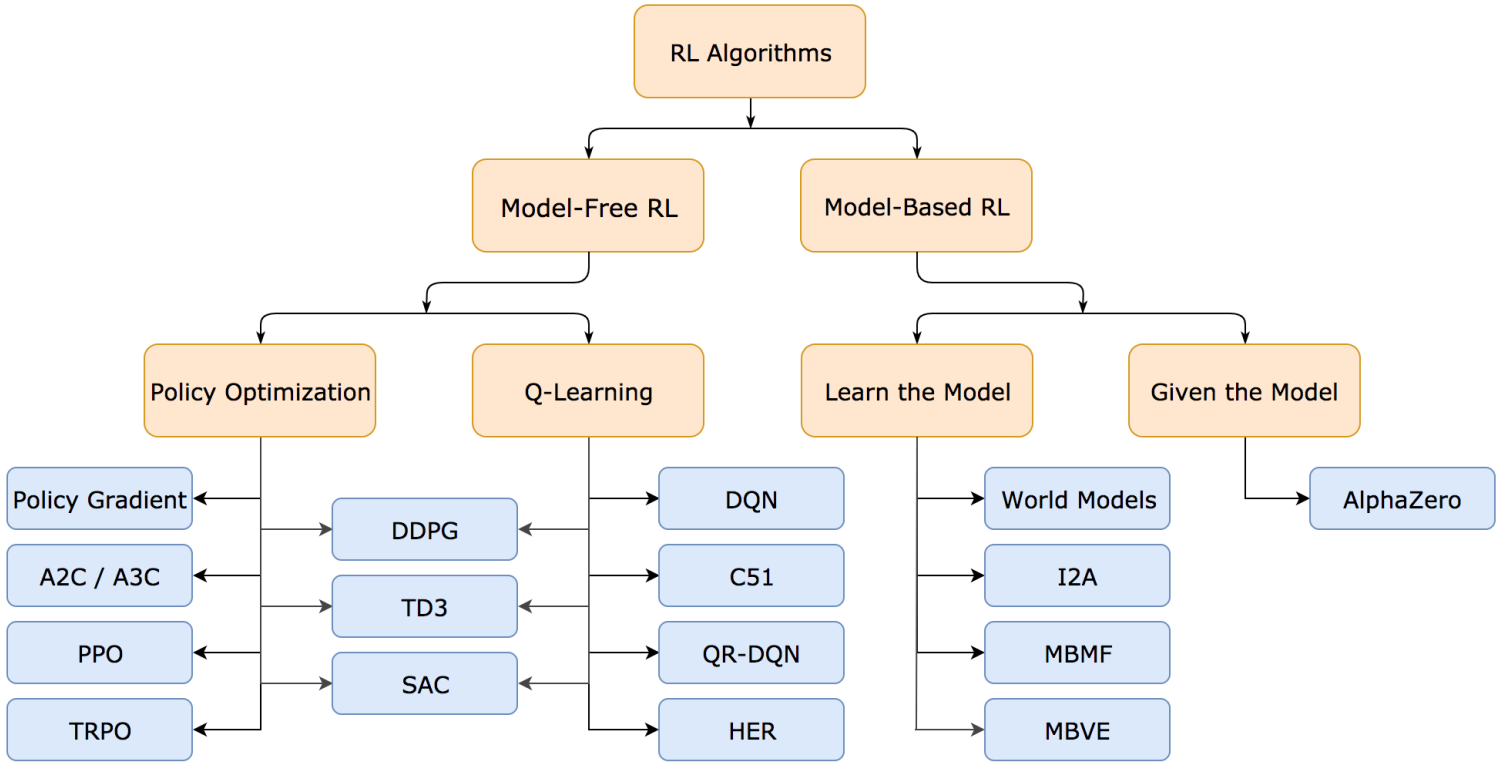
RL Agent Taxonomy



출처 https://www.researchgate.net/figure/Taxonomy-of-RL-Agents-Weng-2018_fig3_363737930

03 Reinforcement Learning: Major Components

RL Algorithms



출처 https://spinningup.openai.com/en/latest/

Learning and Planning

- Two fundamental problems in sequential decision making

Planning
- A model of the environment is known - The agent performs computations with its model (without any external interaction) - The agent improves its policy

03 Reinforcement Learning: Major Components

Learning and Planning

- Two fundamental problems in sequential decision making

Reinforcement Learning
- The environment is initially unknown - The agent interacts with the environment - The agent improves its policy

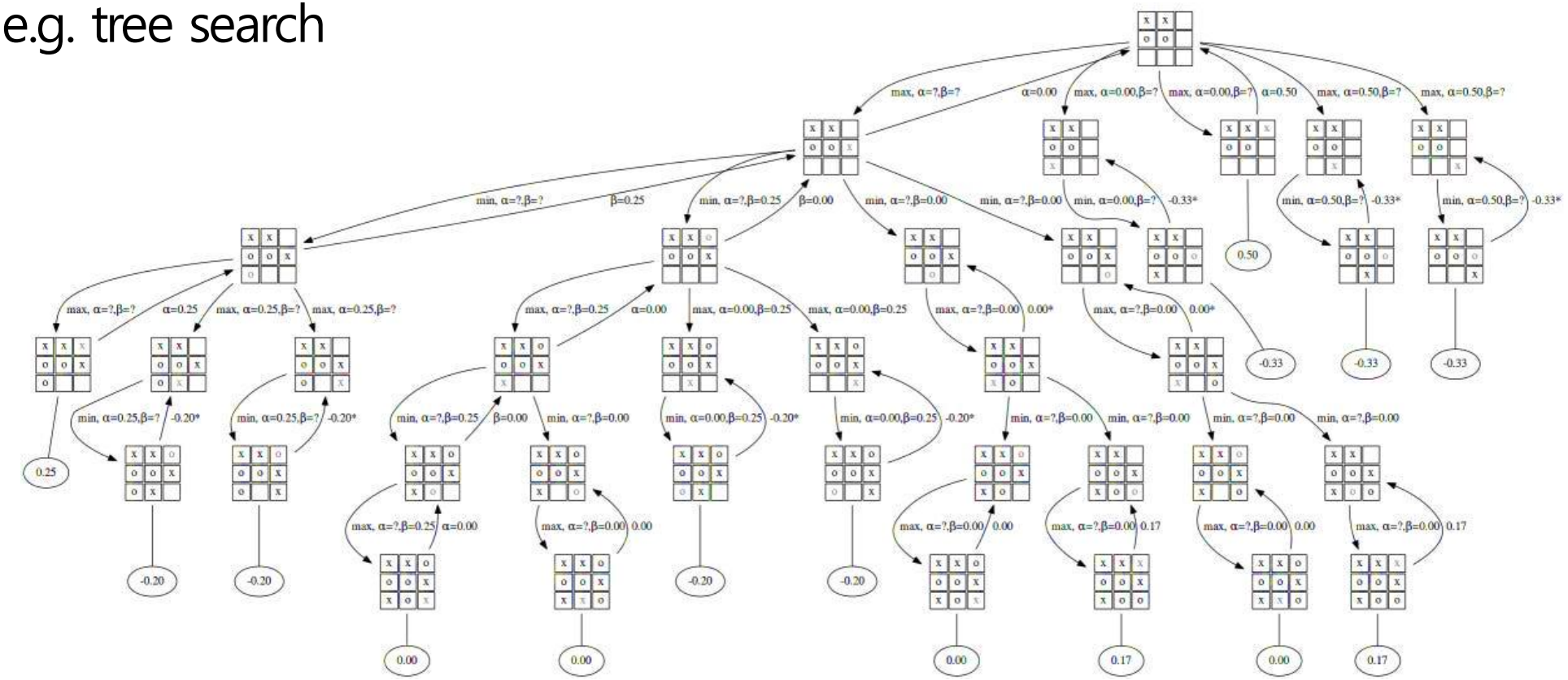
03 Reinforcement Learning: Major Components

Example: Planning

- Rules of the game are known
- Can query simulator which has a(perfect) model
- If I take action from state :
  - what would the next state be?
  - what would the score be?

Example: Planning

- Plan ahead to find optimal policy
  - e.g. tree search

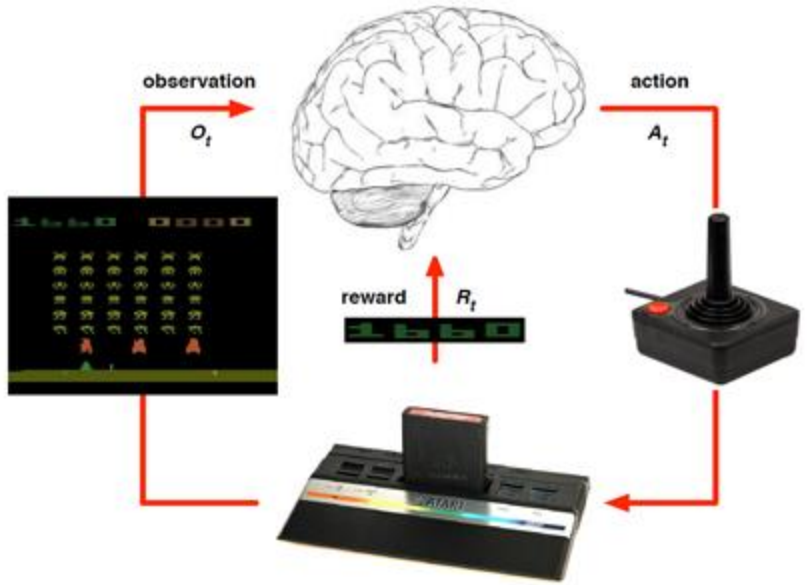


출처 https://medium.com/@jdseo/archived-post-monte-carlo-tree-search-555fc4d0d55

03 Reinforcement Learning: Major Components

Example: Reinforcement Learning

- Rules of the game are unknown
- Learn directly from interactive game-play
- Pick actions and see pixels and scores



출처 https://ralasun.github.io/reinforcement%20learning/2020/07/11/introRL(1)/

Reinforcement Learning: Major Components

Exploration and Exploitation (1/2)

- Reinforcement learning is like trial-and-error learning
- The agent should discover a good policy from its experiences of the environment
- Without losing too much reward along the way

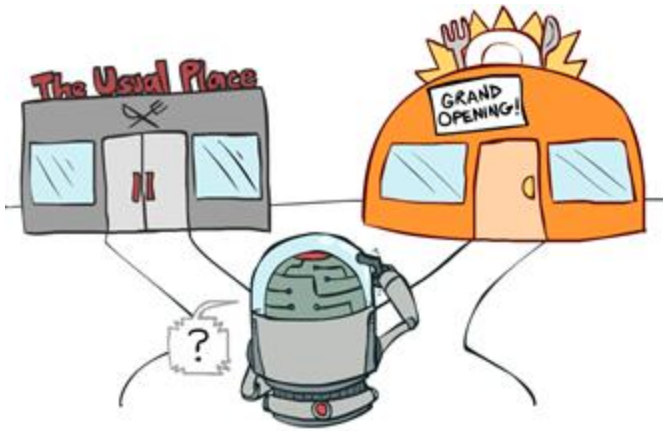
Exploration and Exploitation (2/2)

- Exploration finds more information about the environment
- Exploitation exploits known information to maximize reward
- It is usually important to explore as well as exploit

03 Reinforcement Learning: Major Components

Examples: Exploration & Exploitation

- Restaurant Selection



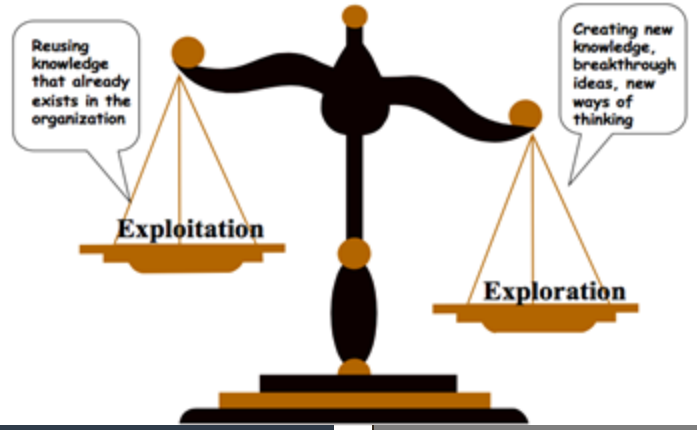
Exploitation	Exploration
Go to your favorite restaurant	Try a new restaurant

출처 https://codetorial.net/articles/cartpole/epsilon_greedy_policy.html

03 Reinforcement Learning: Major Components

Examples: Exploration & Exploitation

- Online Banner Advertisements



Exploitation	Exploration
Show the most successful advert	Show a different advert

출처 https://www.linkedin.com/pulse/organizational-ambidexterity-knowledge-management-anish-prasad-1d

Examples: Exploration & Exploitation

- Oil Drilling

Exploitation	Drill at the best known location
Exploration	Drill at a new location

- Game Playing

Exploitation	Play the move you believe is best
Exploration	Play an experimental move

03 Reinforcement Learning: Major Components

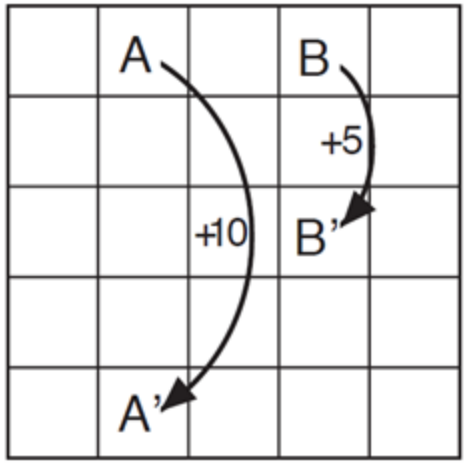
Prediction and Control

- Prediction: evaluate the future
  - Given a policy

- Control: optimize the future
  - Find the best policy

Gridworld Example: Prediction

- What is the value function for the uniform random policy ?



(a)

Actions

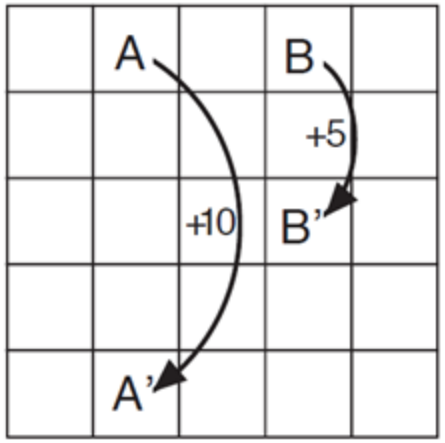
3.3	8.8	4.4	5.3	1.5
1.5	3.0	2.3	1.9	0.5
0.1	0.7	0.7	0.4	-0.4
-1.0	-0.4	-0.4	-0.6	-1.2
-1.9	-1.3	-1.2	-1.4	-2.0

(b)

출처 https://information.koreainfoguide.com/entry/1-Introduction-to-Reinforcement-Learning

Gridworld Example: Control

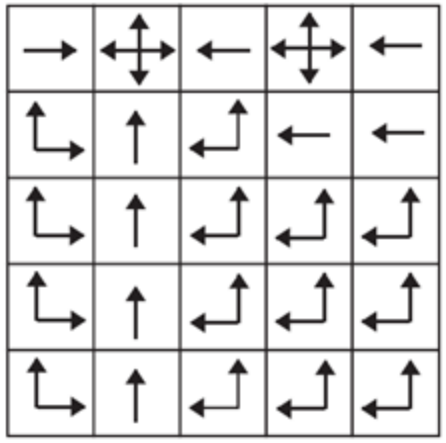
- What is the optimal value function over all possible policies?
- What is the optimal policy?



a) gridworld

22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

b) v_*



c) π_*