

딥 스테리딩 발표

강화 학습 기초 (1)

MDP와 벨만 방정식, Part 1

옥찬호

Nexon Korea, Microsoft MVP

utilForever@gmail.com

오늘 다룰 내용

딥스터딩 발표
강화 학습 기초 (1)

- Part 1

- 강화학습이란?

- MDP

- 상태, 행동, 보상함수, 상태 변환 확률, 감가율, 정책

- Part 2

- 가치함수

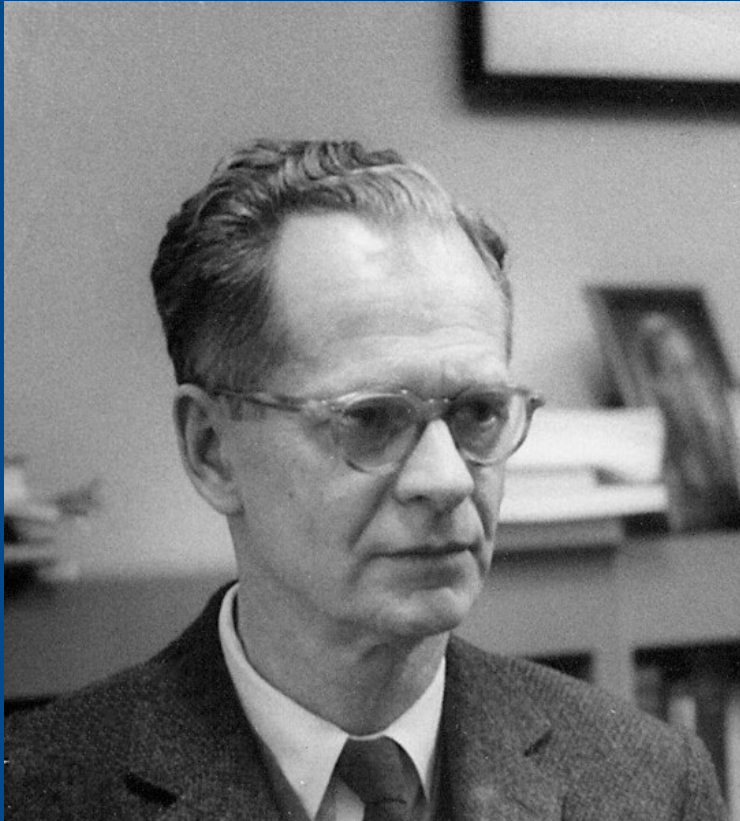
- 가치함수, 큐함수

- 벨만 방정식

- 벨만 기대 방정식, 벨만 최적 방정식

강화학습이란?

딥스터딩 발표
강화 학습 기초 (1)

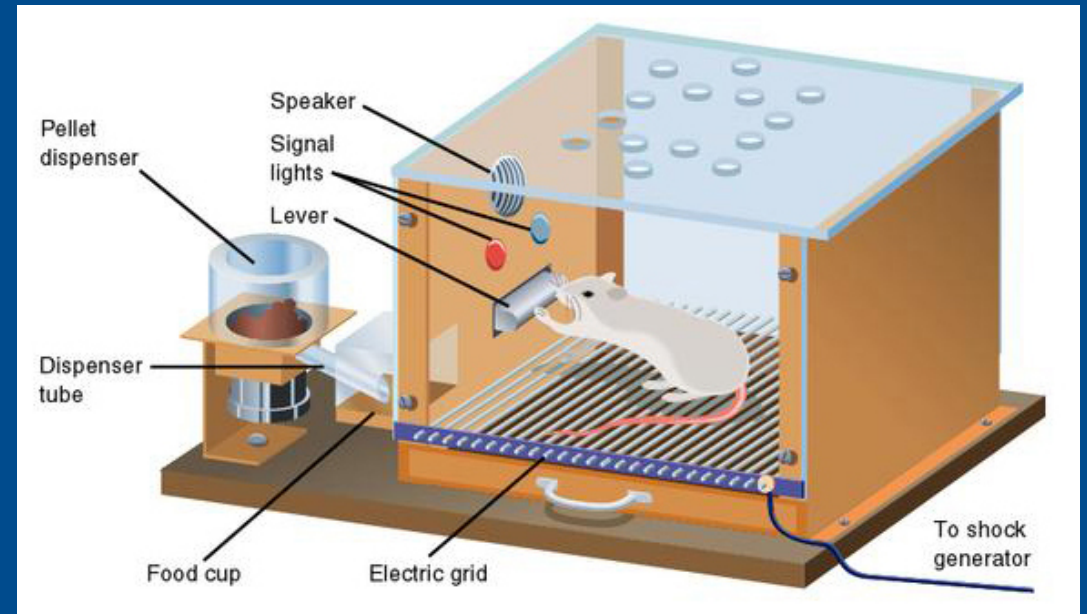


B. F. Skinner (1904~1990)

스키너의 강화 연구

딥스터딩 발표
강화 학습 기초 (1)

1. 굶긴 쥐를 상자에 넣는다.
2. 쥐는 돌아다니다가 우연히 상자 안에 있는 지렛대를 누르게 된다.
3. 지렛대를 누르자 먹이가 나온다.
4. 지렛대를 누르는 행동과 먹이와의 상관관계를 모르는 쥐는 다시 돌아다닌다.
5. 그러다가 우연히 쥐가 다시 지렛대를 누르면 쥐는 이제 먹이와 지렛대 사이의 관계를 알게 되고 점점 지렛대를 자주 누르게 된다.
6. 이 과정을 반복하면서 쥐는 지렛대를 누르면 먹이를 먹을 수 있다는 것을 학습한다.



우리 주변에서의 강화

딥스터딩 발표
강화 학습 기초 (1)

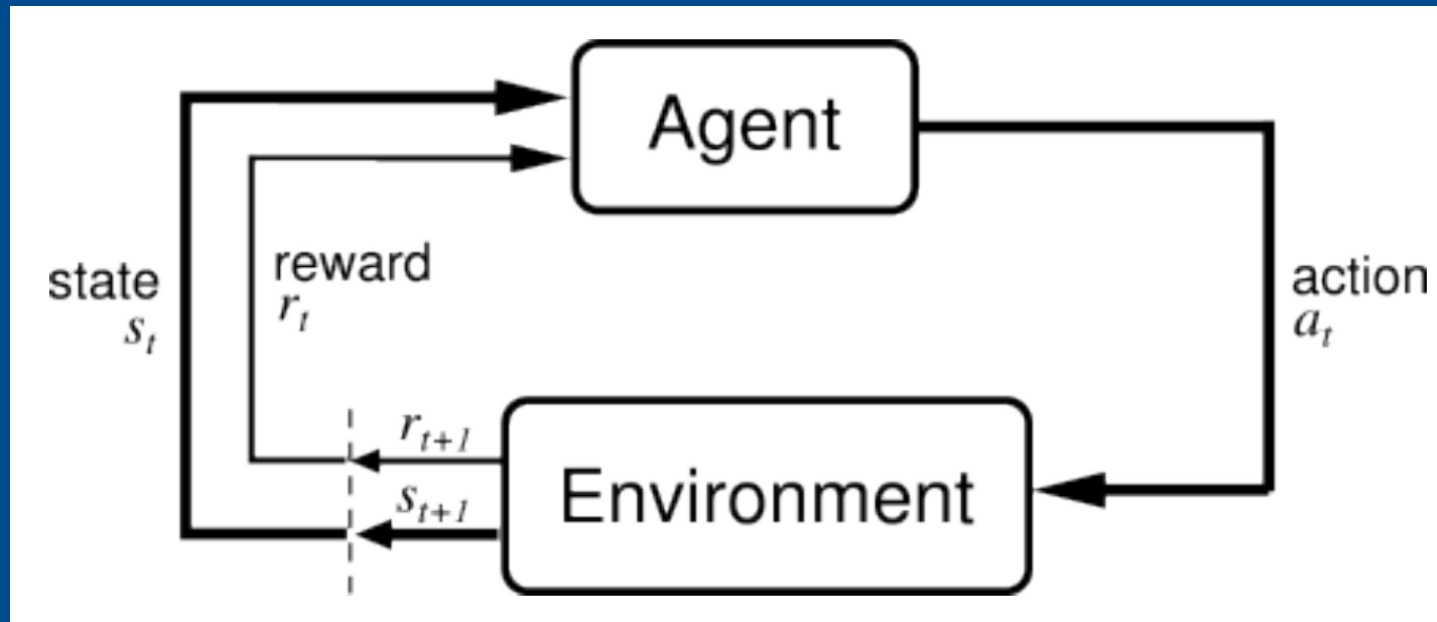
아이가 첫걸음을 떼는 과정도 일종의 강화라고 할 수 있다.

1. 아이는 걷는 것을 배운 적이 없다.
2. 아이는 스스로 이것저것 시도해 보다가 우연히 걷게 된다.
3. 자신이 하는 행동과 걷게 된다는 보상 사이의 상관관계를 모르는 아이는 다시 넘어진다.
4. 시간이 지남에 따라 그 관계를 학습해서 잘 걷게 된다.



EARLY BABY DEVELOPMENT

- 에이전트는 사전 지식이 없는 상태에서 학습함
- 에이전트는 자신이 놓인 환경에서 자신의 상태를 인식한 후 행동
- 환경은 에이전트에게 보상을 주고 다음 상태를 알려줌
- 에이전트는 보상을 통해 어떤 행동이 좋은 행동인지 간접적으로 알게 됨

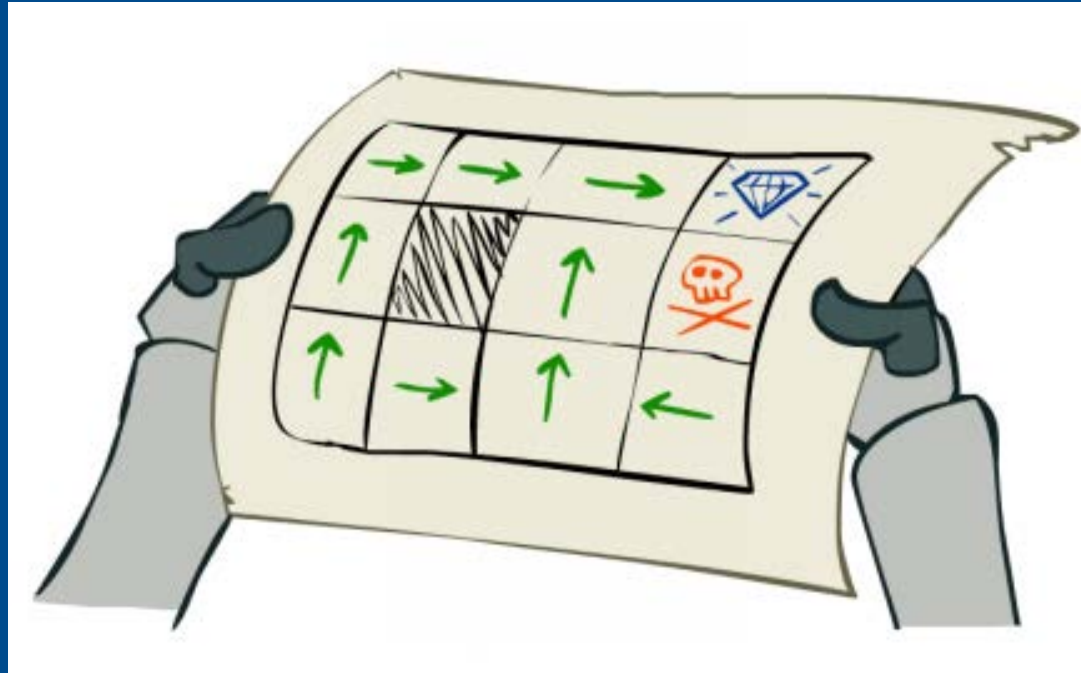


강화학습 문제

딥스터딩 발표
강화 학습 기초 (1)

결정을 순차적으로 내려야 하는 문제에 강화학습을 적용한다.

이 문제를 풀기 위해서는 문제를 수학적으로 정의해야 한다.



수학적으로 정의된 문제는 다음과 같은 구성 요소를 가진다.

1. 상태 (State)
현재 에이전트의 정보 (정적인 요소 + 동적인 요소)
2. 행동 (Action)
에이전트가 어떠한 상태에서 취할 수 있는 행동
3. 보상 (Reward)
에이전트가 학습할 수 있는 유일한 정보, 자신이 했던 행동을 평가할 수 있는 지표
강화학습의 목표는 시간에 따라 얻는 보상의 합을 최대로 하는 정책을 찾는 것
4. 정책 (Policy)
순차적 행동 결정 문제에서 구해야 할 답
모든 상태에 대해 에이전트가 어떤 행동을 해야 하는지 정해놓은 것

강화 학습은 순차적으로 행동을 계속 결정해야 하는 문제를 푸는 것

→ 이 문제를 수학적으로 표현한 것이 MDP(Markov Decision Process)

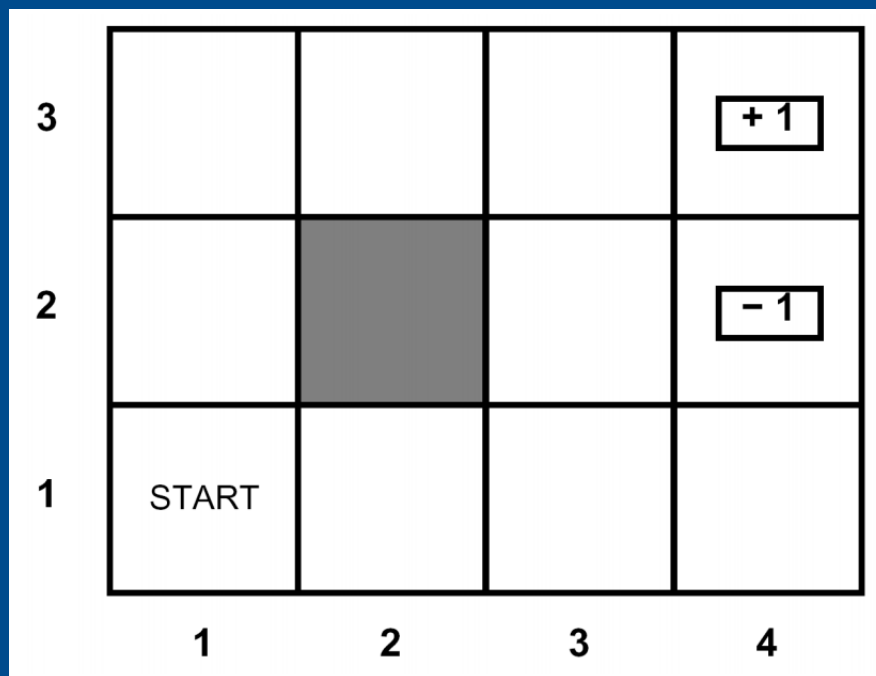
- MDP의 구성 요소

- 상태
- 행동
- 보상 함수
- 상태 변환 확률 (State Transition Probability)
- 감가율 (Discount Factor)

딥스터딩 발표 강화 학습 기초 (1)

에이전트가 관찰 가능한 상태의 집합 : S

- 그리드 월드에서 상태의 개수는 유한
- 그리드 월드에 상태가 5개 있을 경우, 수식으로 표현하면
 $S = \{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5)\}$
- 그리드 월드에서 상태는 격자 상의 각 위치(좌표)
- 에이전트는 시간에 따라 상태 집합 안에 있는 상태를 탐험한다.
이 때 시간을 t , 시간 t 일 때의 상태를 S_t 라고 표현한다.
- 예를 들어, 시간이 t 일 때 상태가 $(1, 3)$ 이라면 $S_t = (1, 3)$



에이전트가 관찰 가능한 상태의 집합 : S

- 어떤 t 에서의 상태 S_t 는 정해진 것이 아니다.
- 때에 따라서 $t = 1$ 일 때 $S_t = (1, 3)$ 일 수도 있고 $S_t = (4, 2)$ 일 수도 있다.

“상태 = 확률 변수(Random Variable)”



$$S_t = s$$

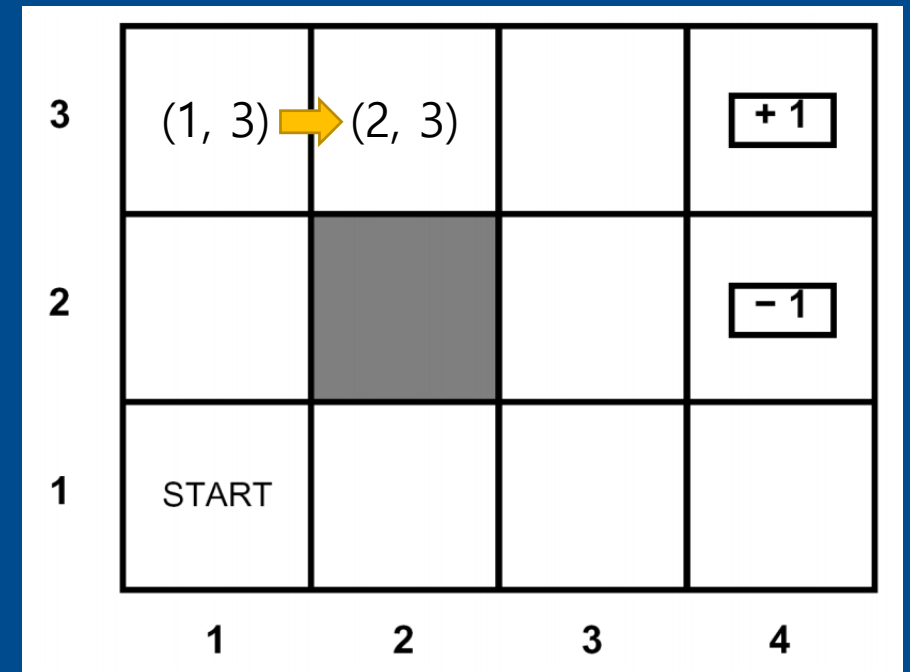
“시간 t 에서의 상태 S_t 가 어떤 상태 s 다.”

에이전트가 상태 S_t 에서 할 수 있는 가능한 행동의 집합 : A

- 보통 에이전트가 할 수 있는 행동은 모든 상태에서 같다.

$$A_t = a$$

- “시간 t 에 에이전트가 특정한 행동 a 를 했다.”
- t 라는 시간에 에이전트가 어떤 행동을 할 지는 정해져 있지 않으므로 A_t 처럼 대문자로 표현한다.
- 그리드 월드에서 에이전트가 할 수 있는 행동은 $A = \{\text{up, down, left, right}\}$
- 만약 시간 t 에서 상태가 $(1, 3)$ 이고 $A_t = \text{right}$ 라면 다음 시간의 상태는 $(2, 3)$ 이 된다.



에이전트가 학습할 수 있는 유일한 정보

- 보상 함수 (Reward Function)

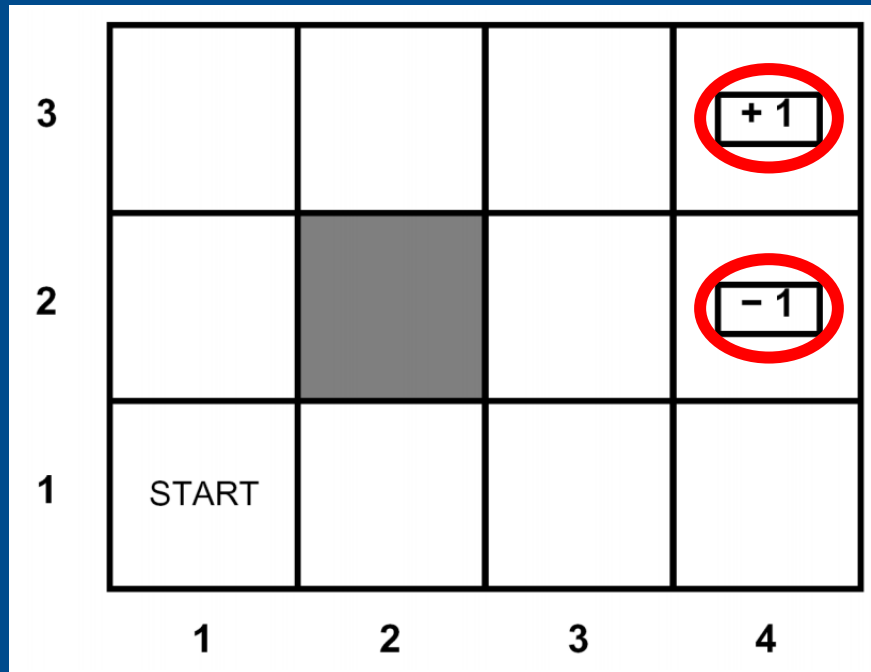
$$R_s^a = E[R_{t+1} | S_t = s, A_t = a]$$

- 시간 t 일 때 상태가 $S_t = s$ 이고 그 상태에서 행동이 $A_t = a$ 를 했을 경우 받을 보상에 대한 기댓값(Expectation) E
- 에이전트가 어떤 상태에서 행동한 시간 : t
보상을 받는 시간 : $t + 1$
- 이유 : 에이전트가 보상을 알고 있는게 아니라 환경이 알려주기 때문
에이전트가 상태 s 에서 행동 a 를 하면 환경은 에이전트가 가게 되는 다음 상태 s' 와 에이전트가 받을 보상을 에이전트에게 알려준다. 이 시점이 $t + 1$ 이다.

보상 함수

딥스터딩 발표
강화 학습 기초 (1)

에이전트가 학습할 수 있는 유일한 정보



에이전트가 어떤 상태에서 어떤 행동을 취하면 상태가 변한다.

하지만 어떤 이유로 인해 다음 상태로 변하지 못할 수도 있다.

→ 상태의 변화에는 확률적인 요인이 들어간다.

이를 수치적으로 표현한 것이 상태 변환 확률!

$$P_{ss'}^a = P[S_{t+1} = s' | S_t = s, A_t = a]$$

에이전트는 항상 현재 시점에서 판단을 내리기 때문에
현재에 가까운 보상일수록 더 큰 가치를 갖는다.

보상의 크기가 100일 때, 에이전트가 현재 시각에 보상을 받을 때는
100의 크기 그대로 받아들이지만 현재로부터 일정 시간이 지나서
보상을 받으면 크기가 100이라고 생각하지 않는다.

에이전트는 그 보상을 얼마나 시간이 지나서 받는지를 고려해
감가시켜 현재의 가치로 따진다.



A 은행

“당첨금 1억 원을 지금 당장 드리겠습니다.”

VS

B 은행

“지금 당장 받으면 막쓰다가 탕진할 가능성이 크니,
10년 후에 당첨금 1억 원을 드리겠습니다.”

A 은행

“당첨금 1억 원을 지금 당장 드리겠습니다.”

≠

B 은행

“지금 당장 받으면 막쓰다가 탕진할 가능성이 크니,
10년 후에 당첨금 1억 원을 드리겠습니다.”

A 은행

“당첨금 1억 원을 지금 당장 드리겠습니다.”

≡

B 은행

“지금 당장 받으면 막쓰다가 탕진할 가능성이 크니,
10년 후에 당첨금 1억 원에 이자까지 드리겠습니다.”

우리는 이자를 통해 나중에 받을 보상에
추가적인 보상을 더해 현재의 보상과 같게 만든다.

→ 반대로 말하면 같은 보상이면 나중에 받을수록 가치가 줄어든다.
이를 수학적으로 표현한 개념이 “감가율(Discount Factor)”

감가율 : 시간에 따라서 감가하는 비율

$$\gamma \in [0, 1]$$

현재의 시간 t 로부터 시간 k 가 지난 후에 받는 보상이 R_{t+k} 라면

현재 그 보상의 가치는 $\gamma^{k-1}R_{t+k}$ 와 같다.

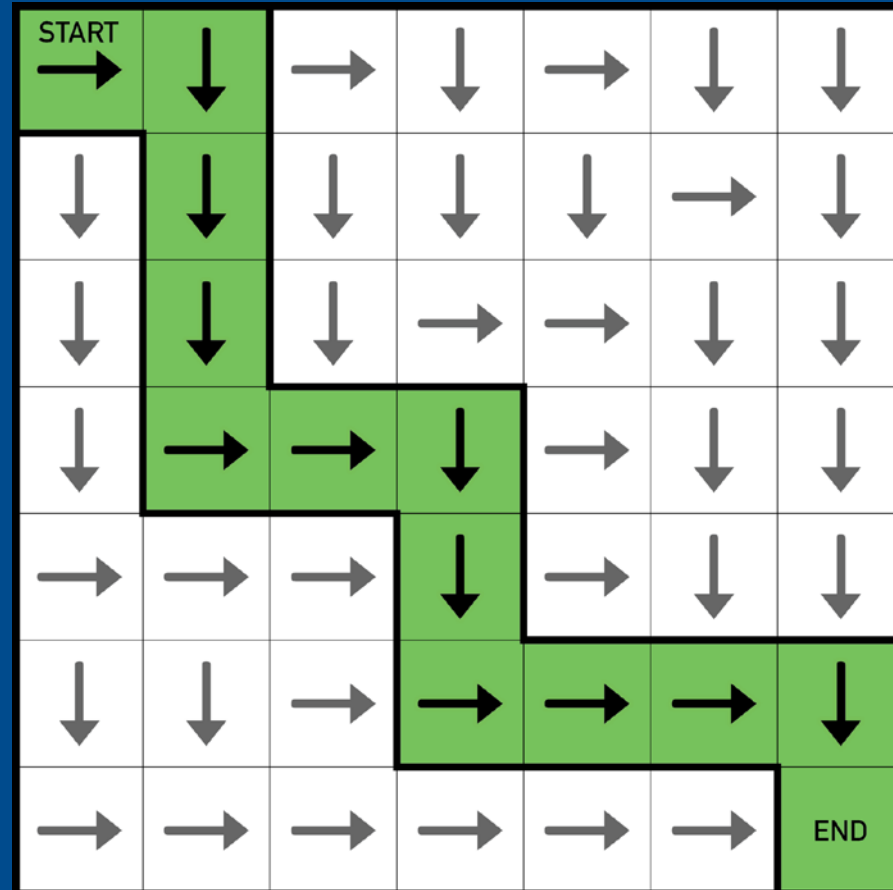
즉, 더 먼 미래에 받을 수록 에이전트가 받는 보상의 크기는 줄어든다.

모든 상태에서 에이전트가 할 행동

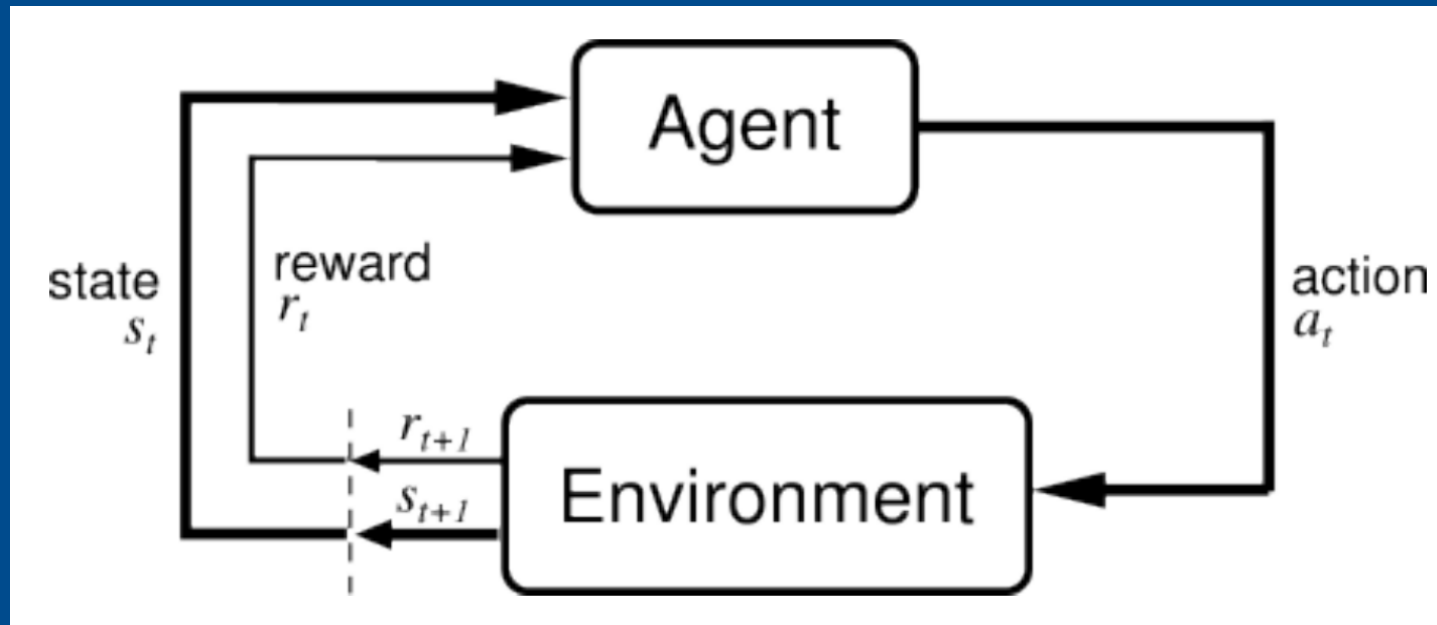
- 상태가 입력으로 들어오면 행동을 출력으로 내보내는 일종의 함수
- 하나의 행동만을 나타낼 수도 있고, 확률적으로 $a_1 = 10\%$, $a_2 = 90\%$ 로 나타낼 수도 있다.

$$\pi(a|s) = P[A_t = a | S_t = s]$$

- 시간 t 에 에이전트가 $S_t = s$ 에 있을 때 가능한 행동 중에서 $A_t = a$ 를 할 확률
- 강화 학습 문제를 통해 알고 싶은 것은 정책이 아닌 “최적 정책”



지금까지 MDP를 통해 순차적 행동 결정 문제를 정의했다.



에이전트는 위 과정을 반복하면서 앞으로 받을 것이라 예상했던 보상이 틀렸다는 것을 알게 된다.
이 과정에서 에이전트는 실제로 받은 보상을 토대로 자신의 정보와 정책을 바꿔나간다.
이 과정을 반복하면 가장 많은 보상을 받게 하는 정책을 학습할 수 있다.

감사합니다

<http://github.com/utilForever>

질문 환영합니다!