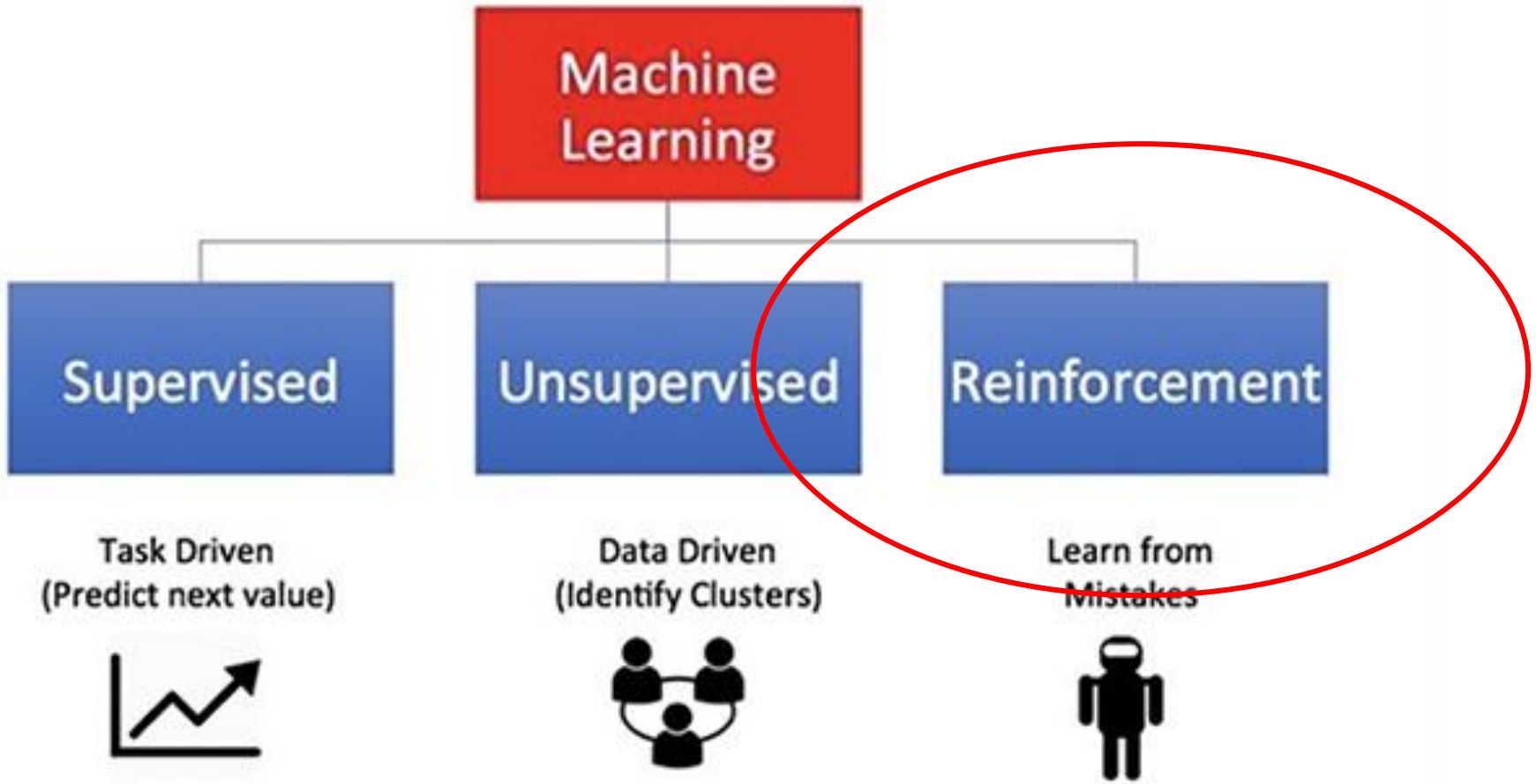
Machine Learning

- Reinforcement Learning -

2022. 5. 6

정 준 수 PhD

Types of Machine Learning



최근에는 self-sunervised learning이라는 영역이 추가되는 분위기지만, 우선은 머신러닝은 위처럼 크게 세 가지로 분류함. 위 분류를 분석 목적 관점과 데이터 형태의 관점으로 구분지어 설명하면,

목적

· Reinforcement Learning: 에이전트의 보상을 최대화하기 위함

· Supervised Learning: 목적변수(Y)를 설명변수 (X)로 예측하기 위함

· Unsupervised Learning: 분석하고자하는 데이터의 숨겨진 구조를 규명하기 위함

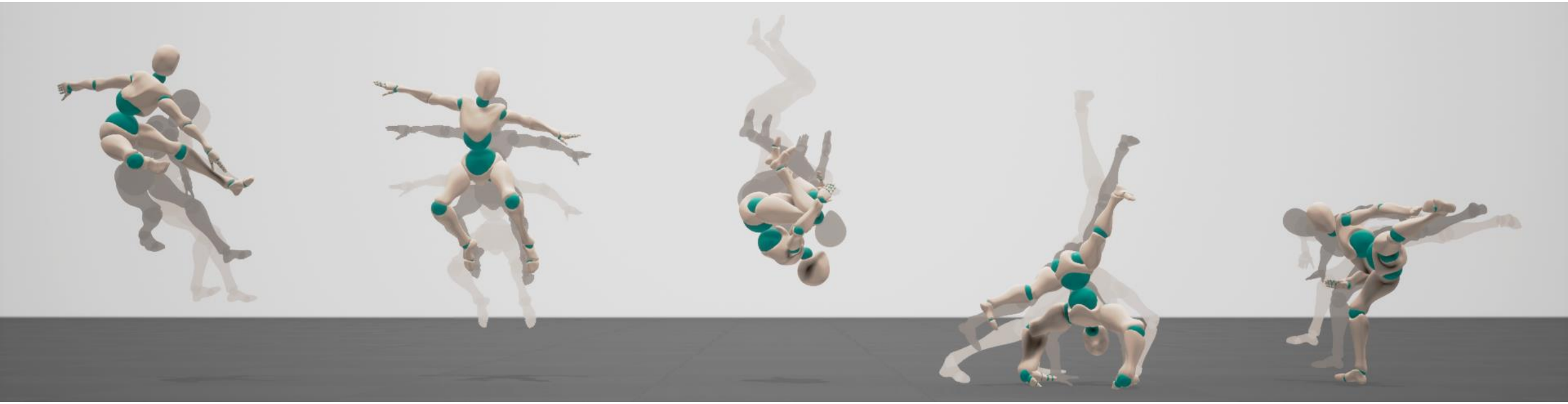
데이터 형태

· Reinforcement Learning: 환경, 에이전트, 보상, 상태만 잘 정의하면 됨
(상황에 따라 데이터가 필요할 수도 있다)

· Supervised Learning: 목적변수 (Y) 설명변수 (X) 형태의 데이터

· Unsupervised Learning: 설명변수 (X) 형태의 데이터

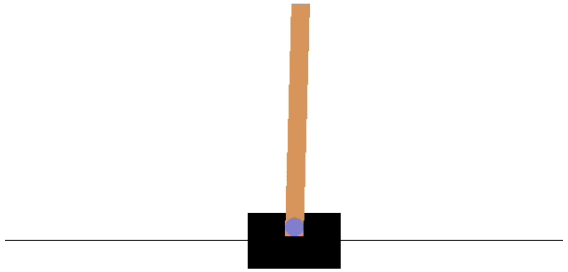
Learning a family of motor skills from a single motion clip

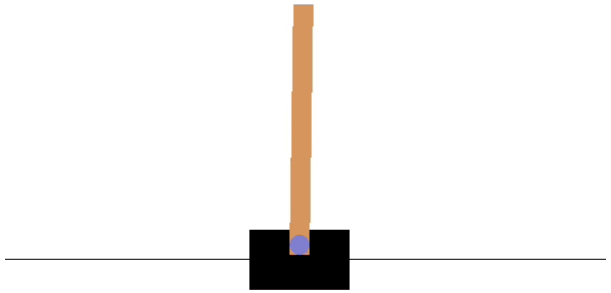


The parameterized motor skills of obstacle jump, jump, backflip, cartwheel and kick motions.

https://mrl.snu.ac.kr/research/ProjectParameterizedMotion/ParameterizedMotion.html

Cart-Pole Reinforcement Learning





Feedback Control of a Cassie Bipedal Robot: Walking, Standing, and Riding a Segway

https://gym.openai.com/envs/BipedalWalker-v2/

https://www.youtube.com/watch?v=UhXly-5tEkc

Behaviorism, also known as behavioral psychology, is a theory of learning which states all behaviors are learned through interaction with the environment through a process called conditioning. Thus, behavior is simply a response to environmental stimuli.

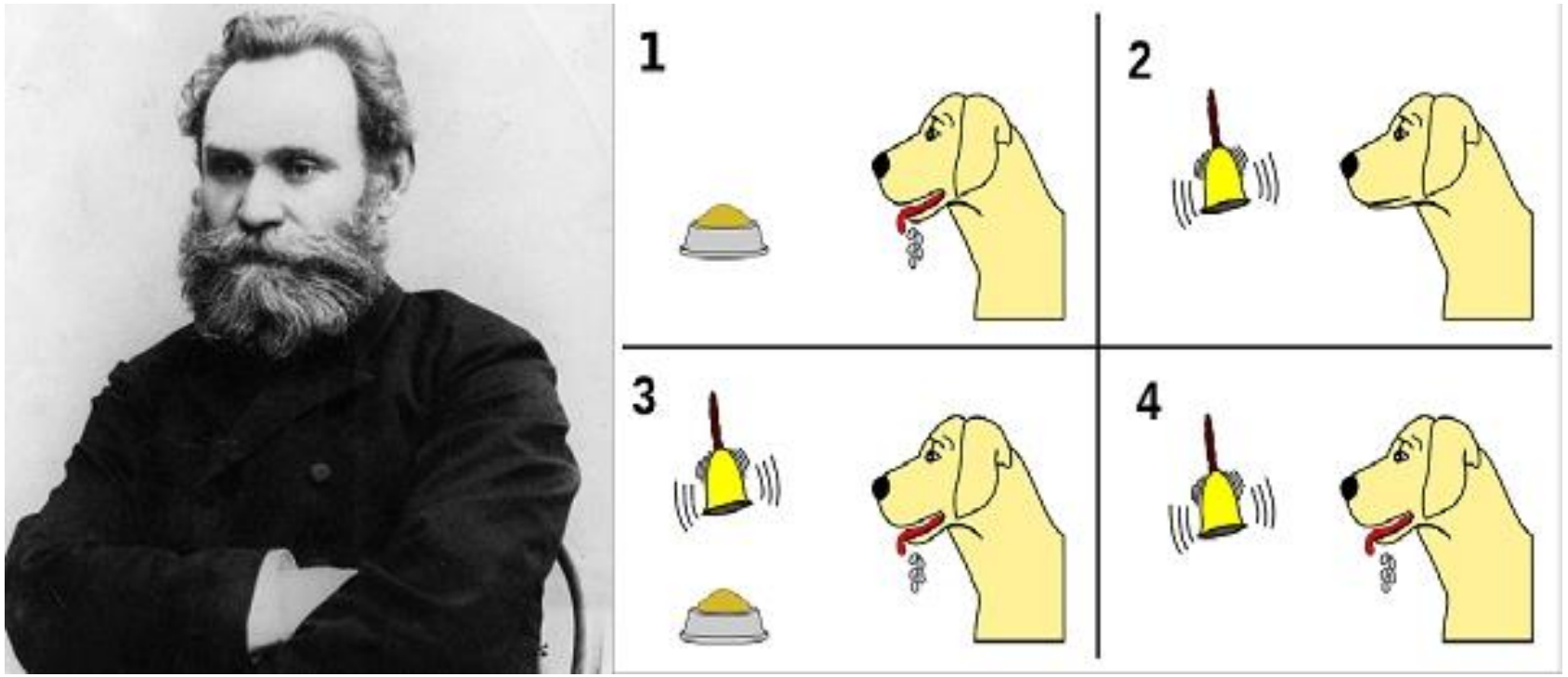
https://www.simplypsychology.org/behaviorism.html

The behaviorist movement began in 1913 when John Watson wrote an article entitled 'Psychology as the behaviorist views it,' which set out a number of underlying assumptions regarding methodology and behavioral analysis.

The History of Behaviorism

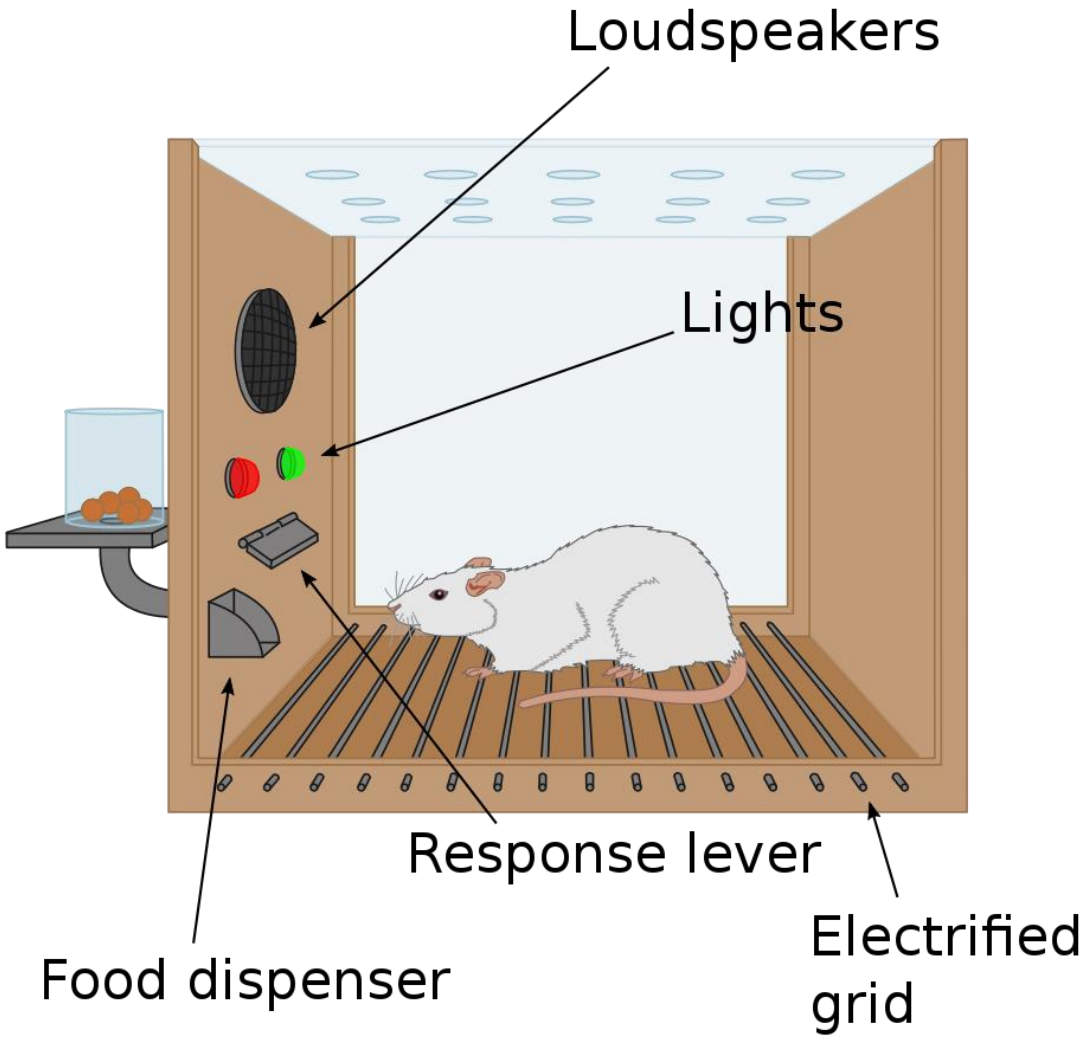
•Pavlov (1897) published the results of an experiment on conditioning after originally studying digestion in dogs.
•Watson (1913) launches the behavioral school of psychology, publishing an article, *Psychology as the behaviorist views it*.
•Watson and Rayner (1920) conditioned an orphan called Albert B (aka Little Albert) to fear a white rat.
•Thorndike (1905) formalized the *Law of Effect*.
•Skinner (1938) wrote *The Behavior of Organisms* and introduced the concepts of operant conditioning and shaping.
•Clark Hull's (1943) *Principles of Behavior* was published.
•B.F. Skinner (1948) published *Walden Two*, in which he described a utopian society founded upon behaviorist principles.
•Journal of the *Experimental Analysis of Behavior* begun in 1958.
•Chomsky (1959) published his criticism of Skinner's behaviorism, "*Review of Verbal Behavior*."
•Bandura (1963) publishes a book called the *Social Leaning Theory and Personality development* which combines both cognitive and behavioral frameworks.
•B.F. Skinner (1971) published his book, *Beyond Freedom and Dignity*, where he argues that free will is an illusion.

Pavlov`s dog



개는 음식을 보면 침을 흘리는 '무조건 반응'을 보이지만 종소리에는 아무런 반응을 하지 않는다. 이를 '중성자극'이라 한다. 뒤이어 중성자극인 종소리를 들려주고 무조건 자극인 음식을 주는 행위를 반복하면 개는 중성자극인 종소리만 듣고도 침을 흘리는 무조건 반응을 일으킨다.

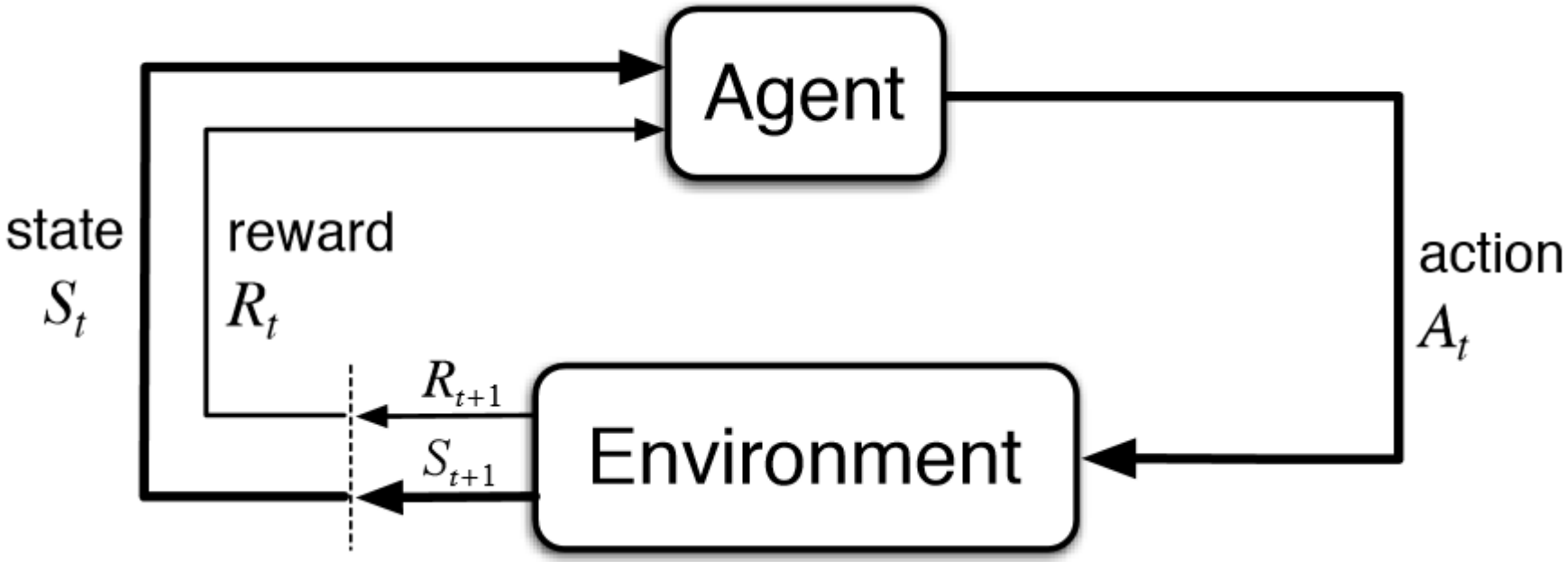
스키너 상자(Skinner box, Operant conditioning chamber라고도 함)



Reinforcement Learning

Key terms that describe the basic elements of an RL problem are:

1.Environment — Physical world in which the agent operates
2.State — Current situation of the agent
3.Reward — Feedback from the environment
4.Policy — Method to map agent's state to actions
5.Value — Future reward that an agent would receive by taking an action in a particular state



Examples

조련사들이 돌고래를 훈련시키는 방법은 다음과 같다. 훈련을 처음 시작한 조련사는 수족관 아래쪽에 링을 설치한 후, 돌고래가 링을 넘어갈 때마다 먹이를 제공한다. 돌고래에게 '링을 넘어가는 행동을 하면 먹이를 먹을 수 있다'는 인식을 심어주는 것이다.
매번 돌고래가 링을 넘어가는 행동을 하는 것은 아니지만, 인내심을 갖고 기다려주면 얼마 지나지 않아 돌고래는 링을 넘어가는 행동에 익숙해 지게 된다.

“러 해군, 우크라 특수부대 막으려 훈련 받은 돌고래 풀었다”

UNSI 뉴스는 “해양 전문가들은 훈련 받은 돌고래가 탐지 및 무력화(無力化)에 동원될 수 있어, 잠수 요원에 대한 효율적인 방어책으로 간주한다”고 전했다. 인간 잠수요원이 민첩성·속도·시야 확보에서 돌고래·바다표범 등과 상대가 안 되고, 군사용 돌고래가 잠수요원을 포착해 해상에 부표를 띄우면 육상에서 공격할 수 있게 된다는 것이다.

미국과 러시아가 예전부터 해양 포유동물을 군사용으로 훈련해 온 것은 공공연한 사실이다. 미 해군은 샌디에이고 항구에서 1960년대부터 수중 위협에 대비해 돌고래와 바다사자를 훈련해 온 것이 1990년대 비밀 해제된 프로그램을 통해 밝혀졌다. 미국은 9·11 테러 이후 2003년 ‘지속적인 자유(Enduring Freedom)’ 작전에서 바레인에 군사 목적의 훈련을 받은 바다 사자들을 풀어놨다.

https://biz.chosun.com/international/international_general/2022/04/28/NVQ3VIPJ7BEIVE44MEV3SYCAYY/

"군사적 용도로 돌고래 훈련시키는 북한의 '돌고래 부대' 정황이 위성사진에 포착됐다“

미국 해군연구소(USNI)가 운영하는 군사전문 매체 USNI 뉴스는 현지 시간으로 지난 12일 북한 해군기지가 있는 남포항 부근에서 군사적 용도로 돌고래를 훈련하고 있는 정황이 포착됐다고 보도했는데요.

인공위성 사진을 분석한 결과 해군기지가 있는 남포항에 위치한 조선소와 석탄 부두 부근 해상에서 돌고래용 우리가 발견됐다는 것입니다.

https://www.animalplanet.co.kr/contents/?artNo=13795

베르나르 베르베르의 소설

원제는 최후 비밀(L'ultime Secret). 번역자는 이세욱. 2001년에 프랑스에서 출간되었다.

1997년 제작된 체스 컴퓨터 '딥 블루'와 가리 카스파로프의 대결과, 1954년 맥길 대학교의 심리학 교수 제임스 올즈의 실험[1]을 주제 삼아 쓰여진 소설이다.

2002년에 한국에서 베스트셀러가 되었다.

작중에서 끊임없이 언급되는 '최후 비밀'이란 뇌 속에 있는 '쾌감의 중추'를 의미한다. 이를 자극하면 쾌감을 느끼게 되며, 작중 인물들은 이것이 권력의 손아귀에 넘어가면 사람들이 쾌감의 노예가 될 것이라고 걱정한다.[2] 제임스 올즈는 원래 이를 비밀 실험으로 하고자 하였으나 작중에서는 체르니엔코라는 이름의 박사가 정보 관리를 소홀히 했다는 설정으로 나왔다.

사뮈엘 핀처는 이 실험에 착안하여 인간의 한계를 시험하기 위해 자신의 뇌에 전극을 꽂고, 환자이자 동료인 장 루이 마르탱에게 자신이 무언가를 이뤘을때 줄 보상으로 쾌감을 조작할 것을 요구하였다. 그리고 그 **보상**을 얻기위해 노력한 덕분에 사뮈엘 핀처는 세계 최고의 체스기사이자 딥 블루 IV를 이긴 사람이 될 수 있었던 것이다.

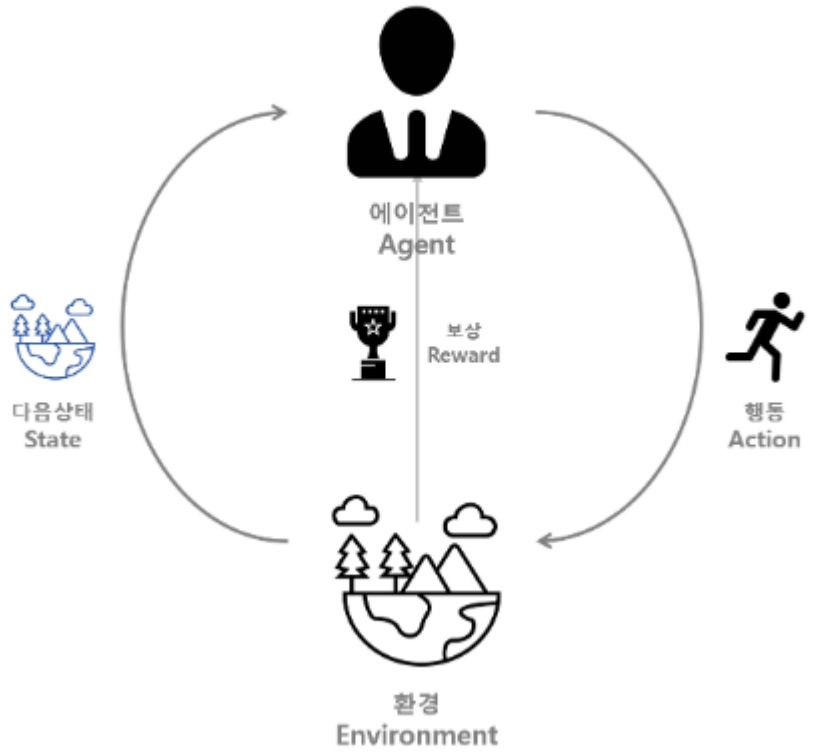
AWS Deep Racer on Re:Invent track

https://www.youtube.com/watch?v=XtTqr3WeQA4

FinRL: Deep Reinforcement Learning for Quantitative Finance

https://github.com/JSJeong-me/FinRL

강화 학습(Reinforcement Learning)의 상호작용

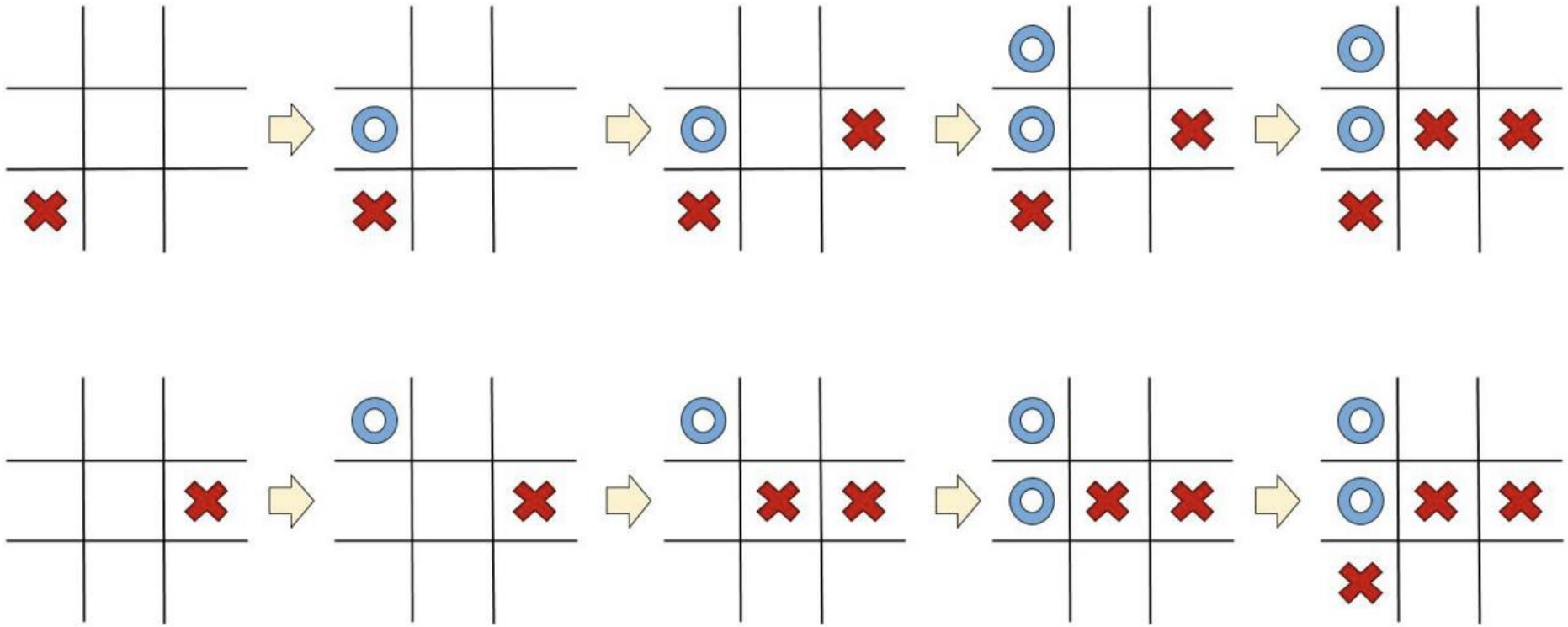


강화학습은 환경과 에이전트가 행동, 보상, 다음 상태로 상호작용하며 이루어 집니다.

강화 학습(Reinforcement Learning)

- Markov Decision Processes (MDP)
- The Bellman equation
- Q-networks
- Policy gradients

Markov State



위쪽과 아래쪽, 두 게임 모두 결과적으로 맨 오른쪽을 보면
같은 상태에 도달했지만, 그 과정은 다릅니다.

Markov Decision Processes(MDPs)

Definition

A Markov Process (or Markov Chain) is a tuple $\langle S, P \rangle$,

- S is a (finite) set of states
- P is a state transition probability matrix

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

random variable, random process, markov process 사이의 관계는 아래와 같은데, markov process는 시간 개념이 들어가 있는 random process의 markov property가 추가된 special case이다. (보다 연산에 있어서 효율적) [출처] https://daljoong2.tistory.com/195

Random variable → random process → Markov process

(시간 개념 X) (시간 개념 O) (시간 개념 O + Markov Property)

$$\mathrm{P}[S_{t+1}|S_t] = \mathrm{P}[S_{t+1}|S_1, \dots, S_t]$$

Bellman Equation

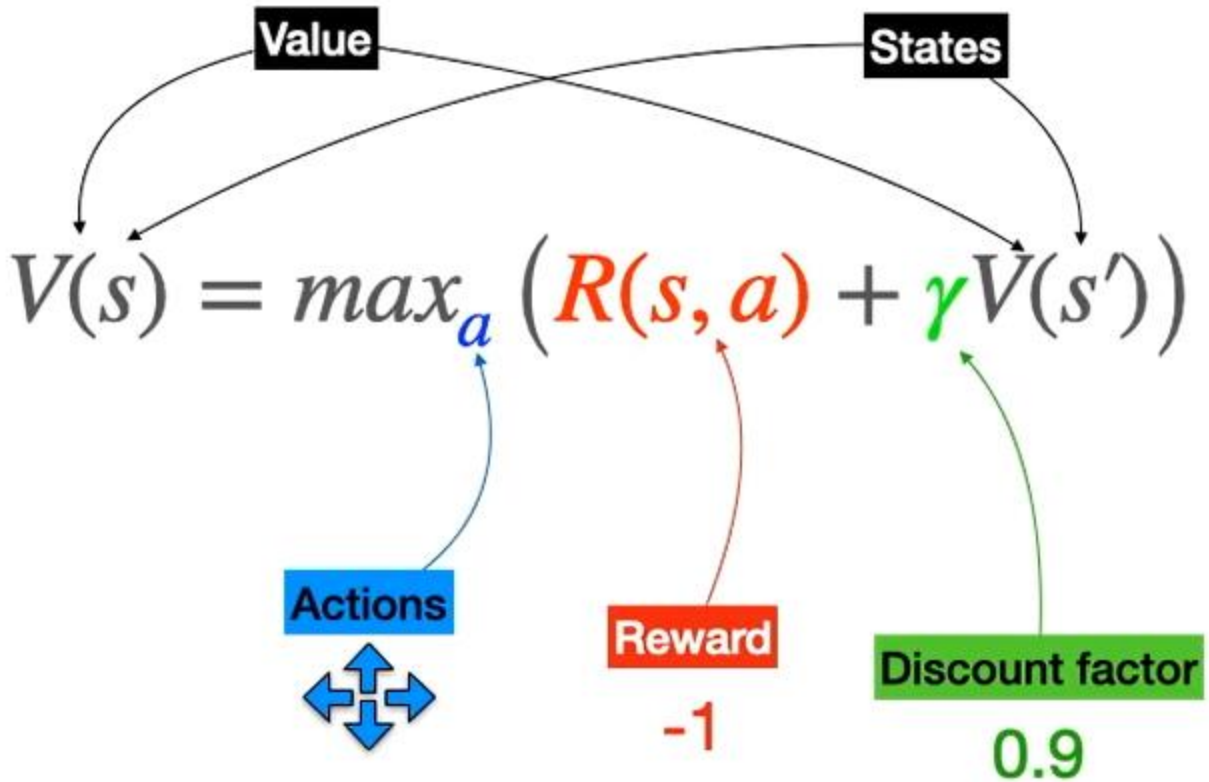
$$v_*(s) = \max_a \mathrm{E}[\ R_{t+1} + \gamma v_*(S_{t+1}) \mid S_t=s,\ A_t=a\ ]$$ **벨만 최적방정식**

$$q_*(s,a) = \mathrm{E}[\ R_{t+1} + \gamma \max_{a'} q_*(S_{t+1},a') \mid S_t=s,\ A_t=a\ ]$$ **큐함수에 대한 벨만 최적 방정식**

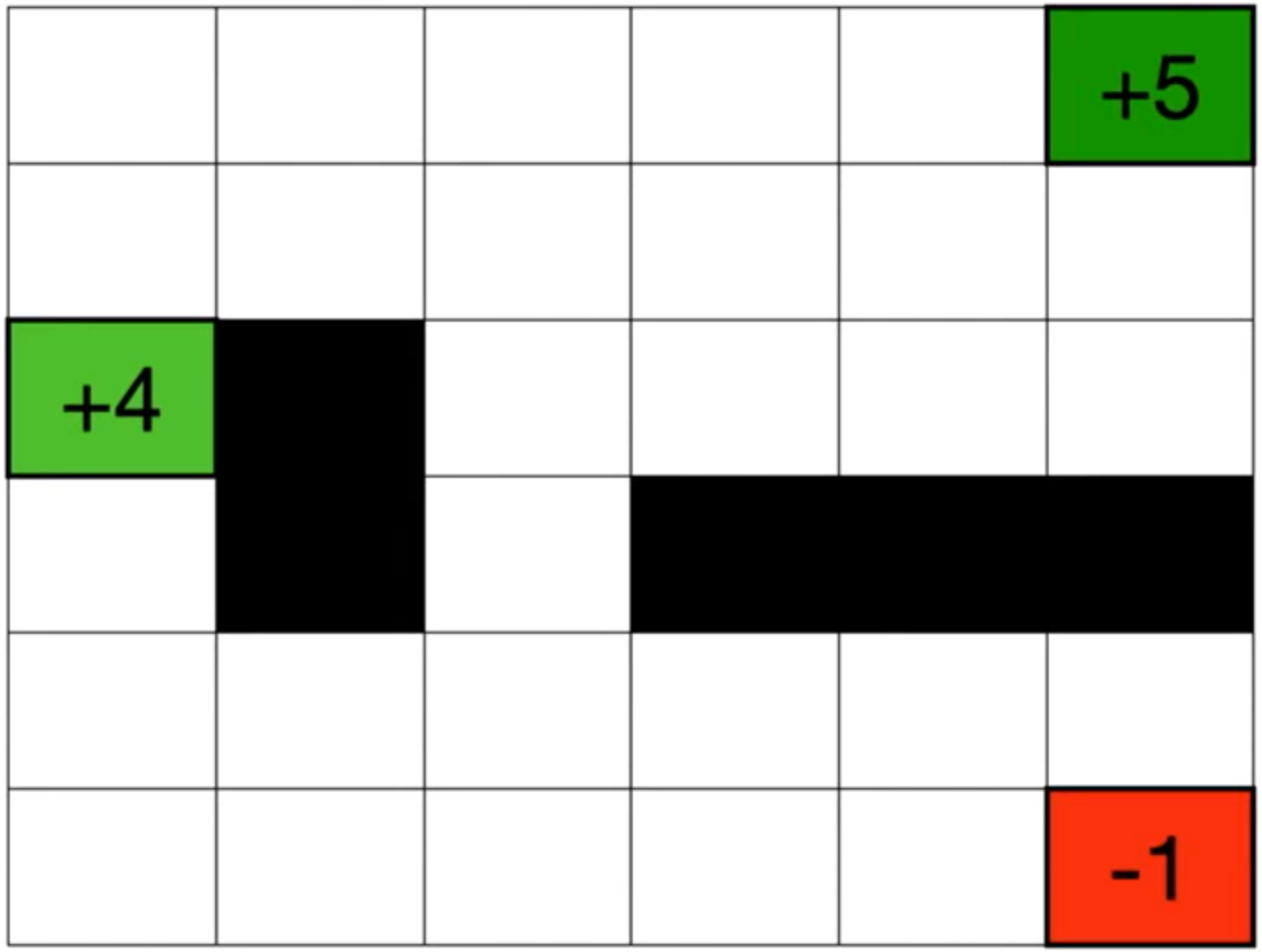
최종적으로 가장 높은 보상을 받을 수 있는 행동들을 연속적
으로 취하여 받게 되는 결과

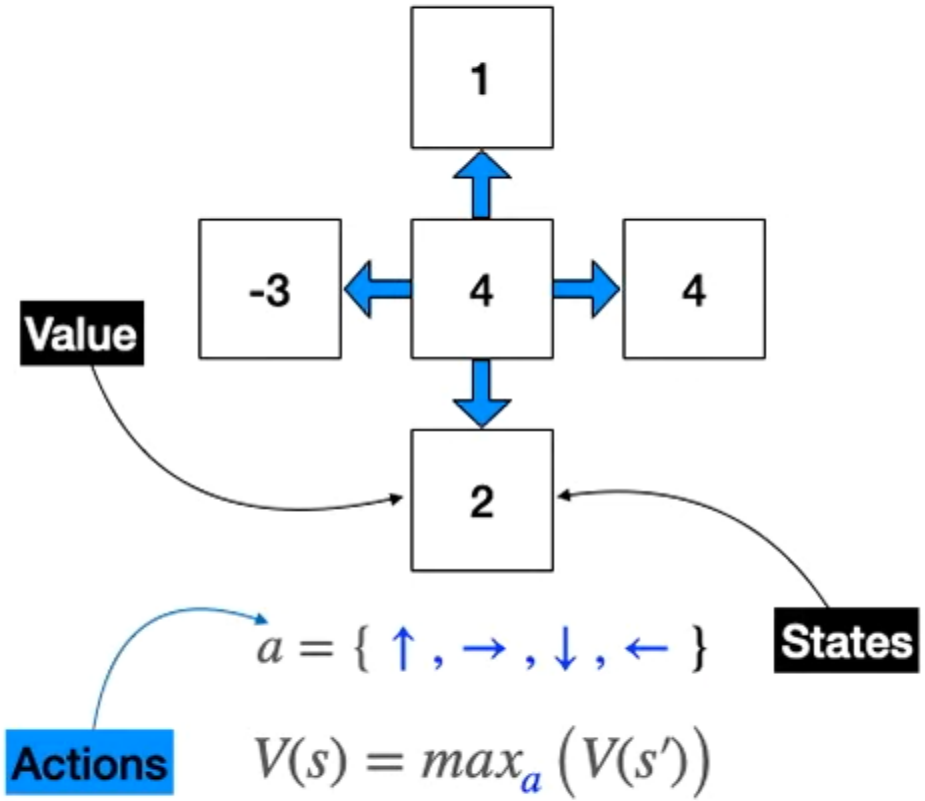
Bellman equation

$$V(s) = max_a \left(R(s,a) + V(s')\right) \qquad V(s) = max_a \left(\gamma V(s')\right)$$



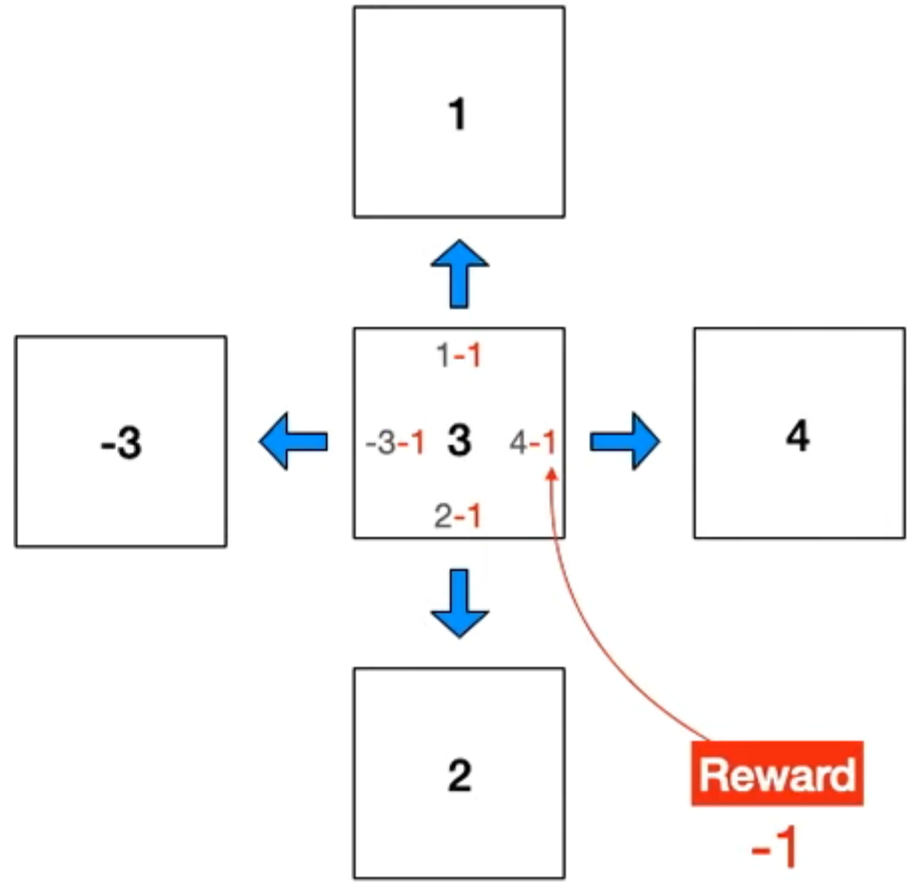
Value and policy





Reward

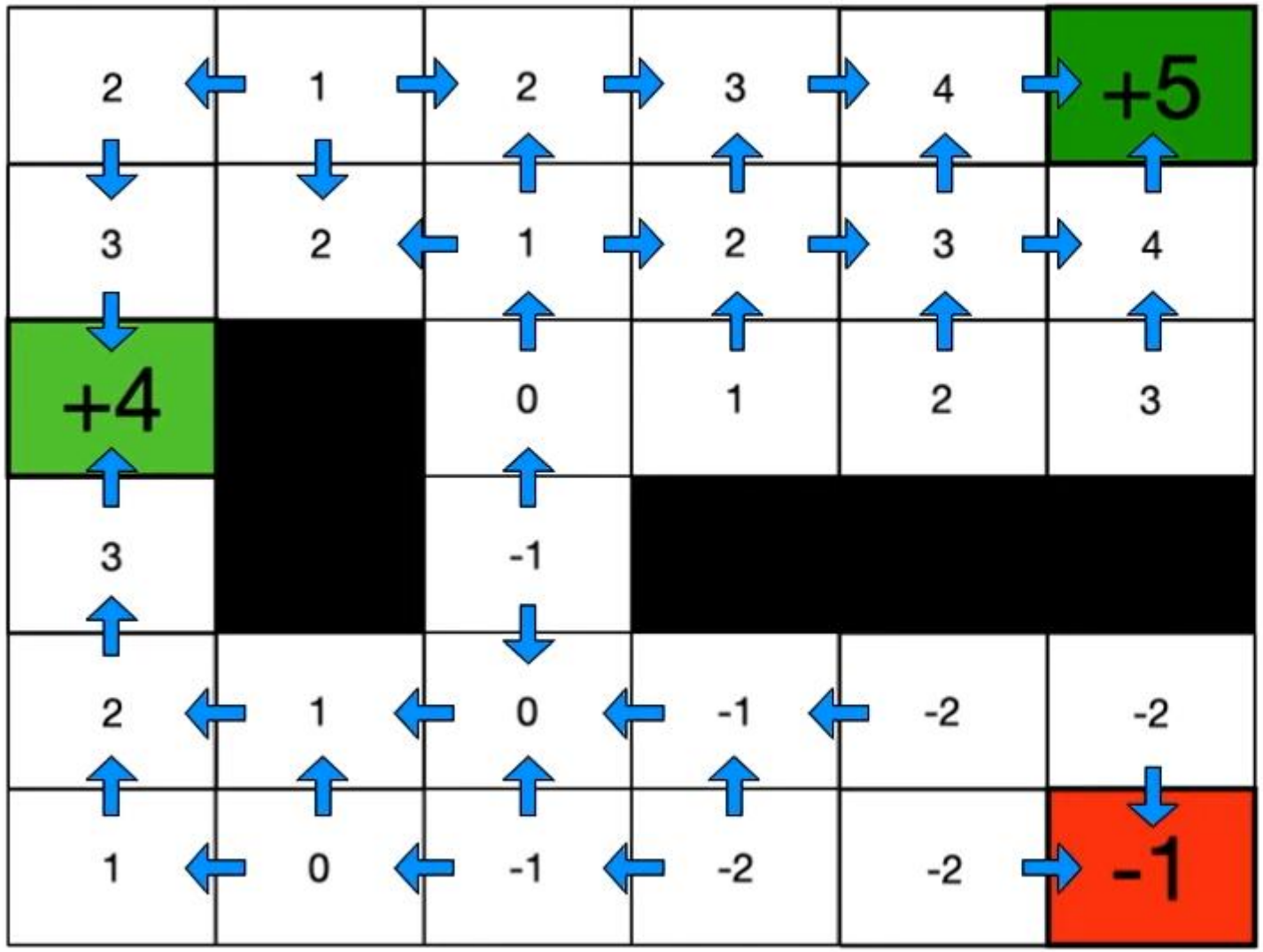
2	1	2	3	4	+5
3	2	1	2	3	4
+4		0	1	2	3
3		-1			
2	1	0	-1	-2	-2
1	0	-1	-2	-2	-1

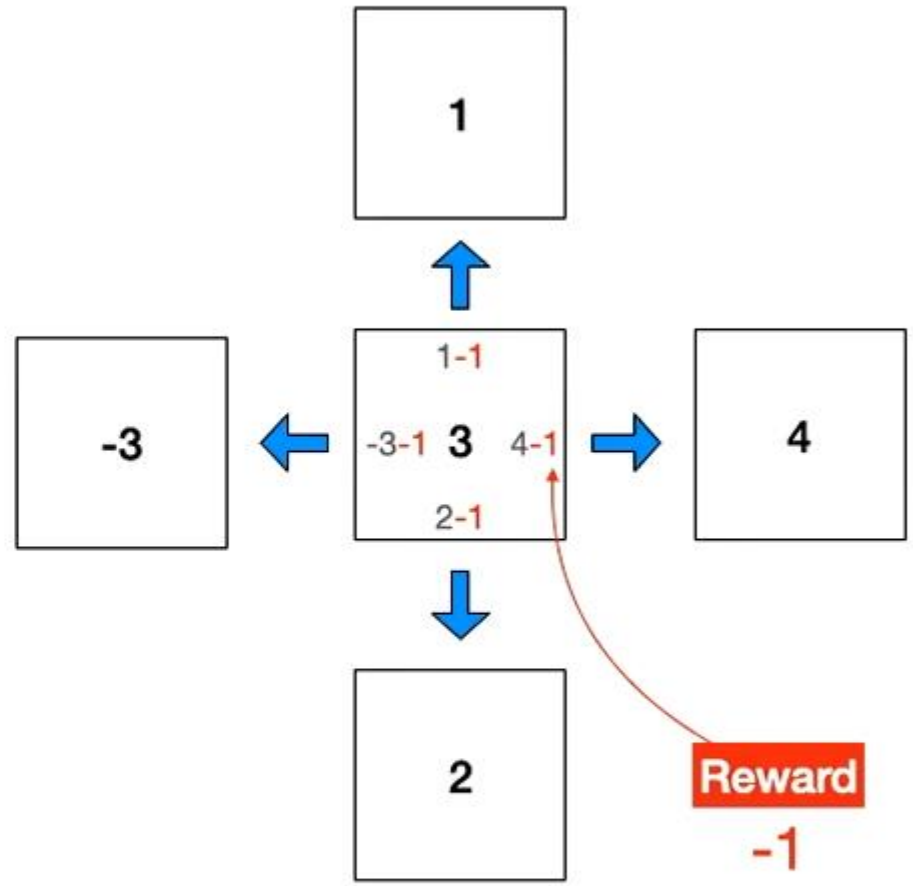


$$V(s) = max_a \left(R(s,a) + V(s') \right)$$

$$a = \{ \uparrow, \rightarrow, \downarrow, \leftarrow \}$$

Reward

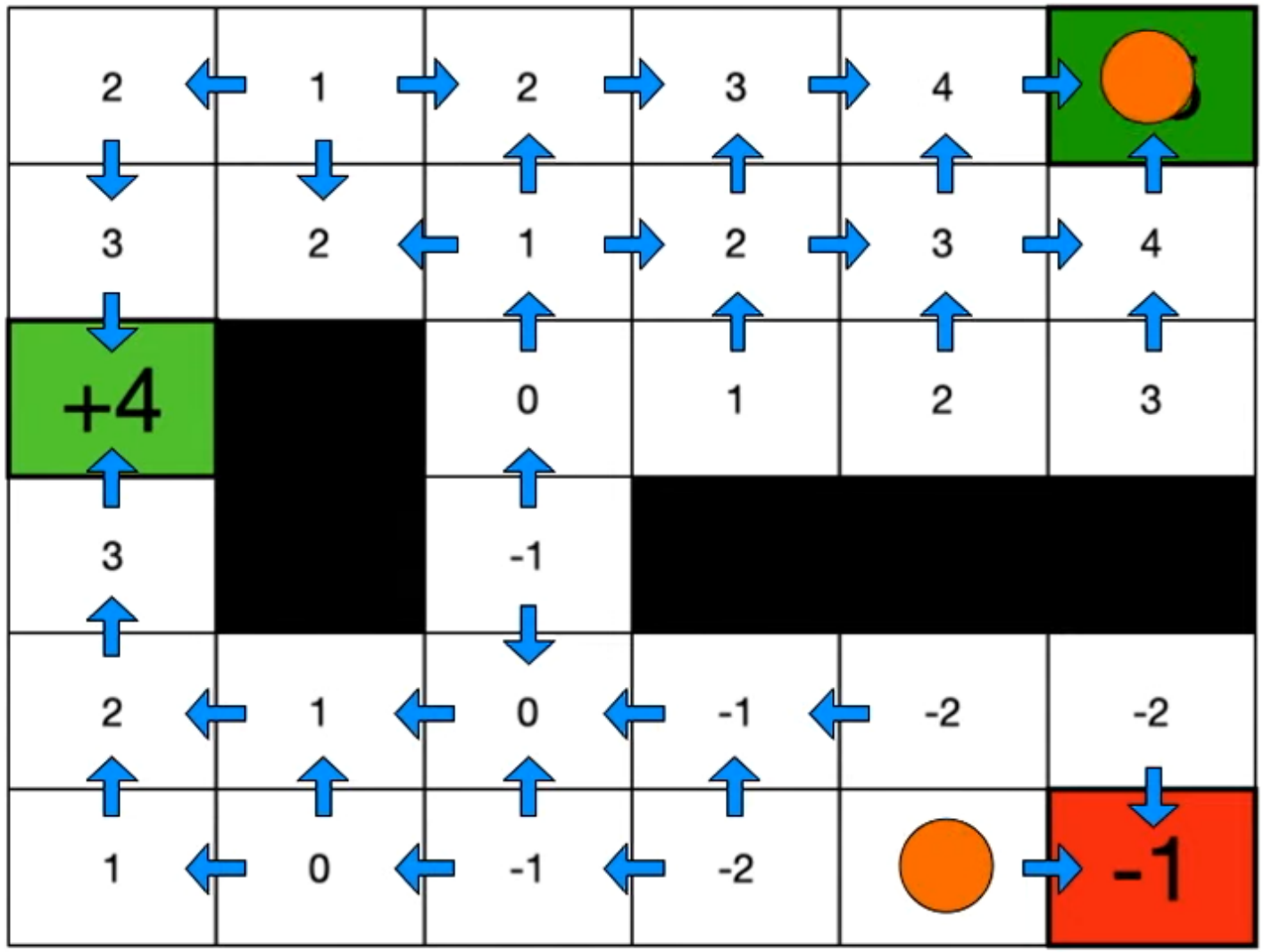




$$V(s) = max_a \left(R(s,a) + V(s') \right)$$

$$a = \{ \uparrow, \rightarrow, \downarrow, \leftarrow \}$$

Reward



Discount factor

Today

1

Tomorrow

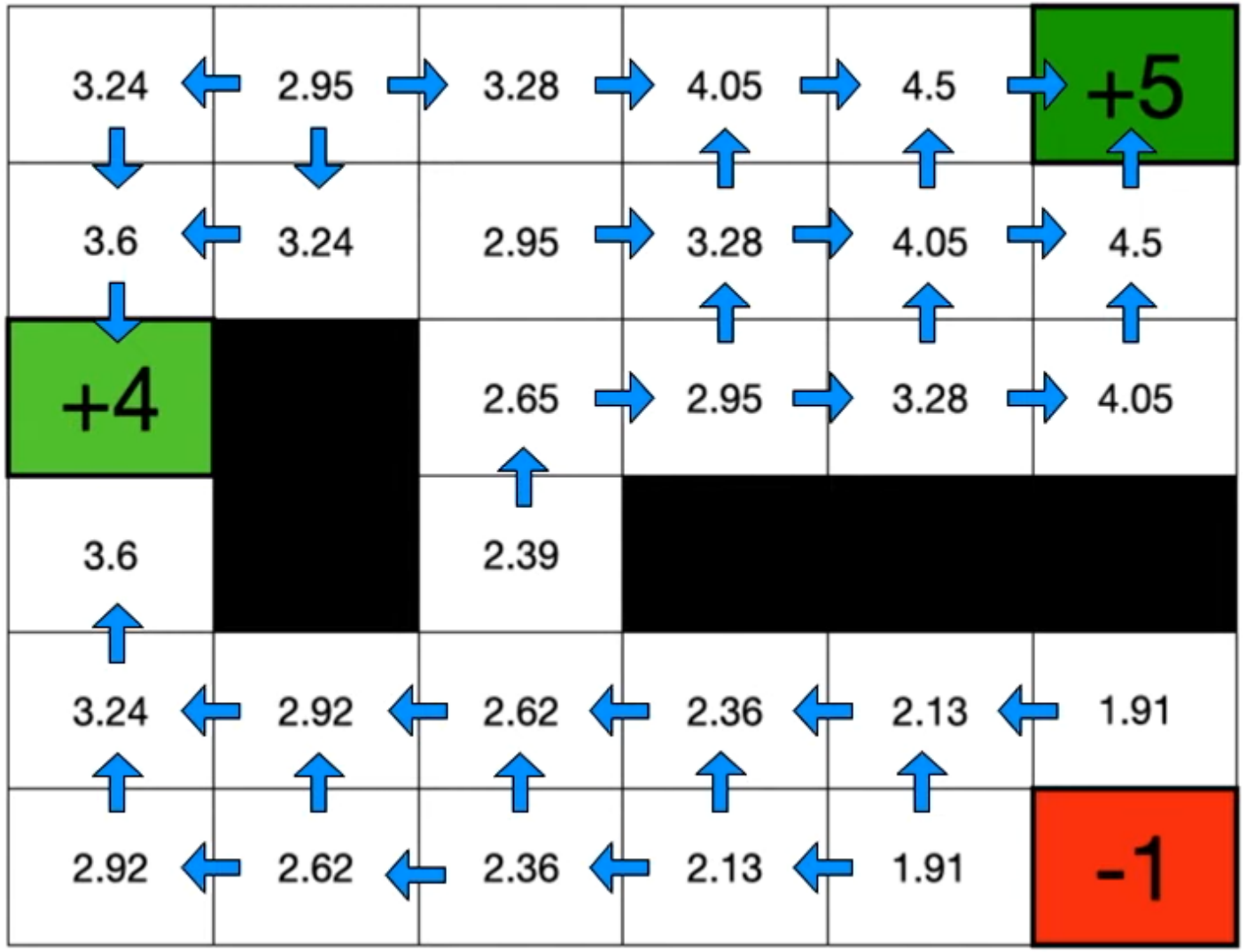
$1 \cdot 0.9 = 0.9$

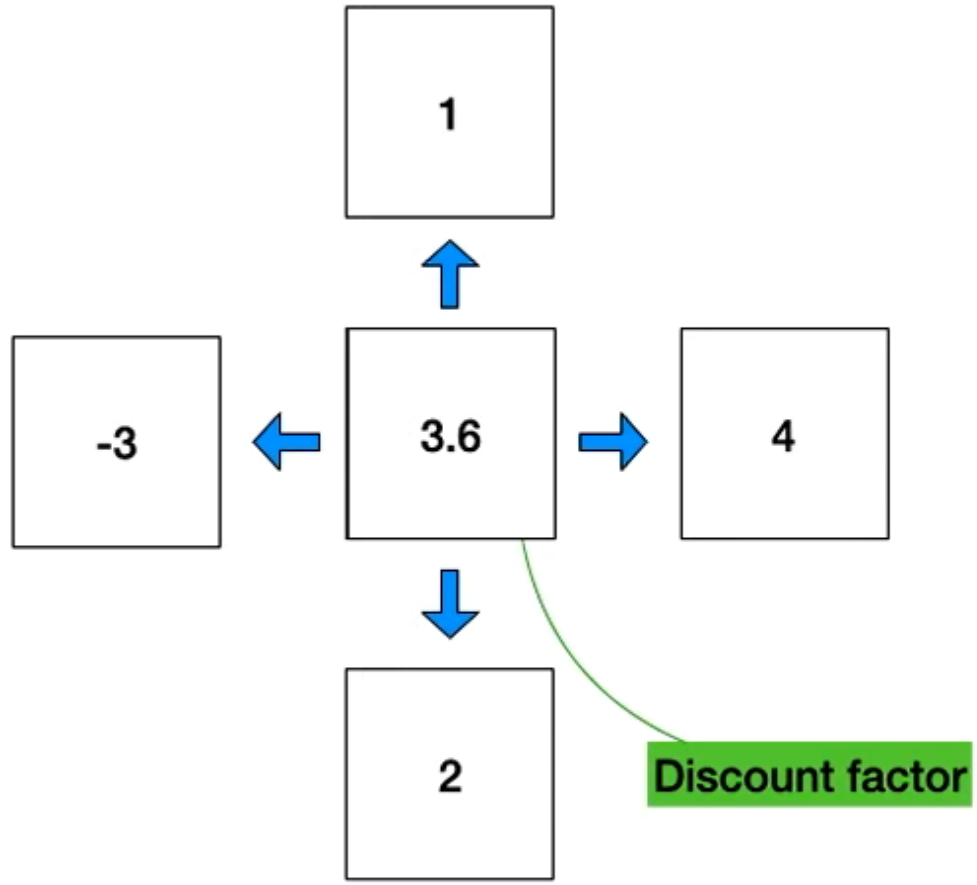
Day after

$0.9 \cdot 0.9 = 0.81$

Discount factor

Discount factor



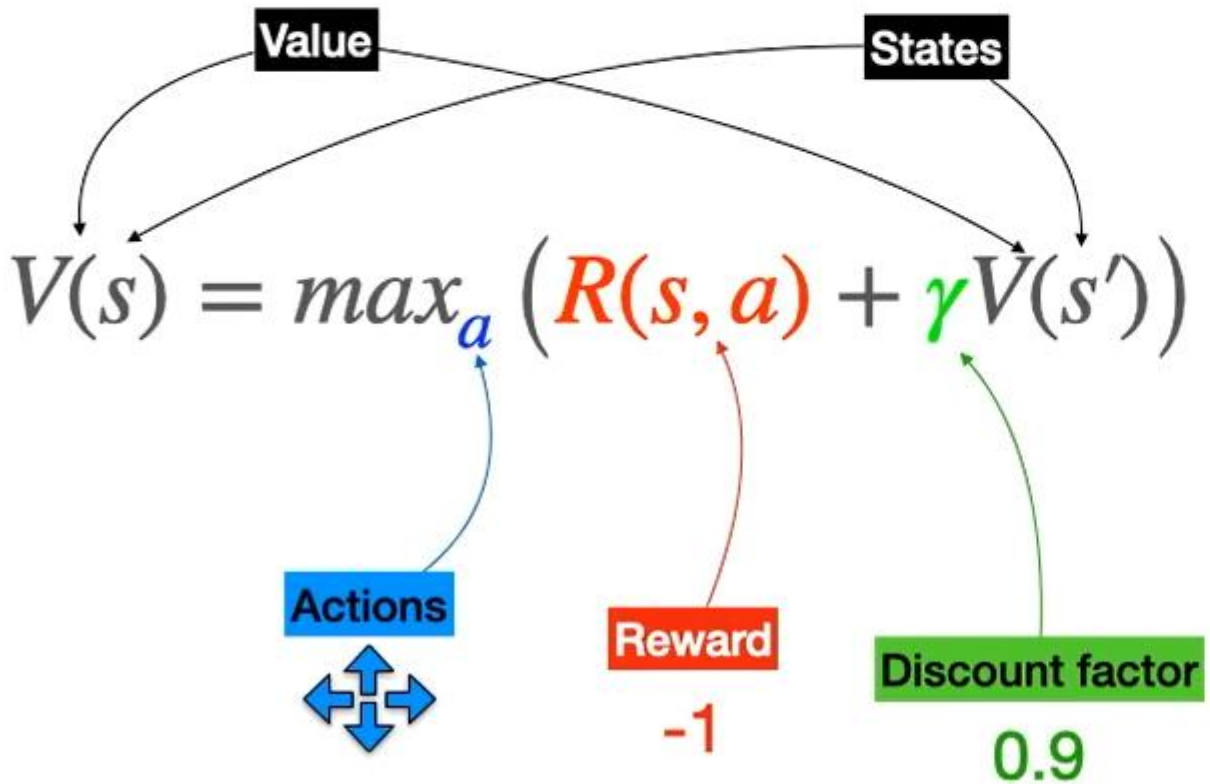


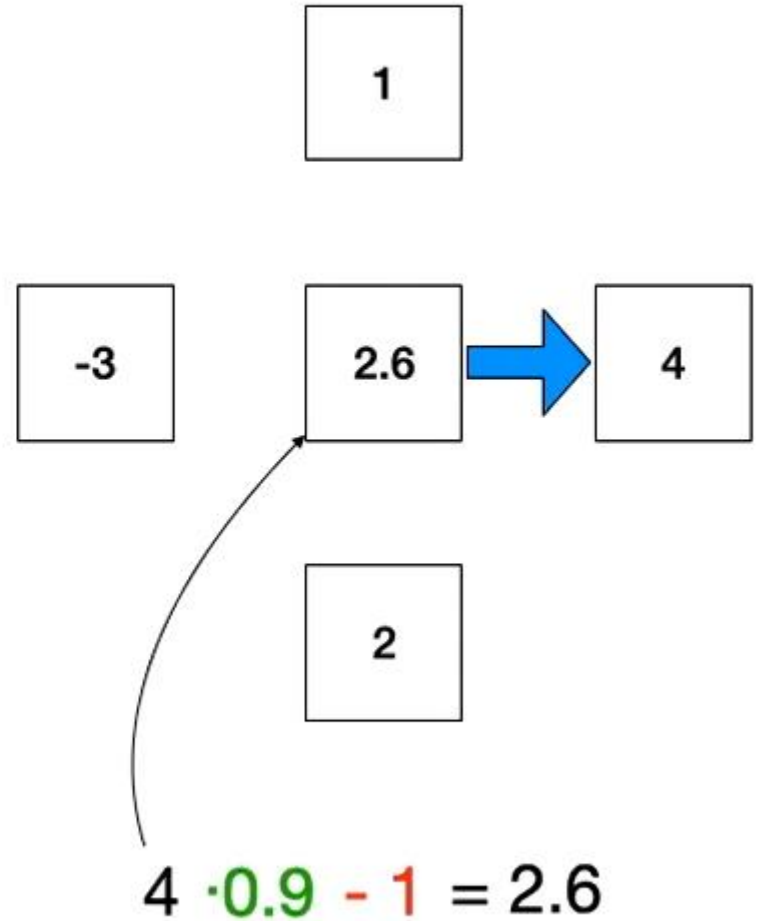
$$V(s) = max_a \left(\gamma V(s')\right)$$

$$a = \{ \uparrow, \rightarrow, \downarrow, \leftarrow \}$$

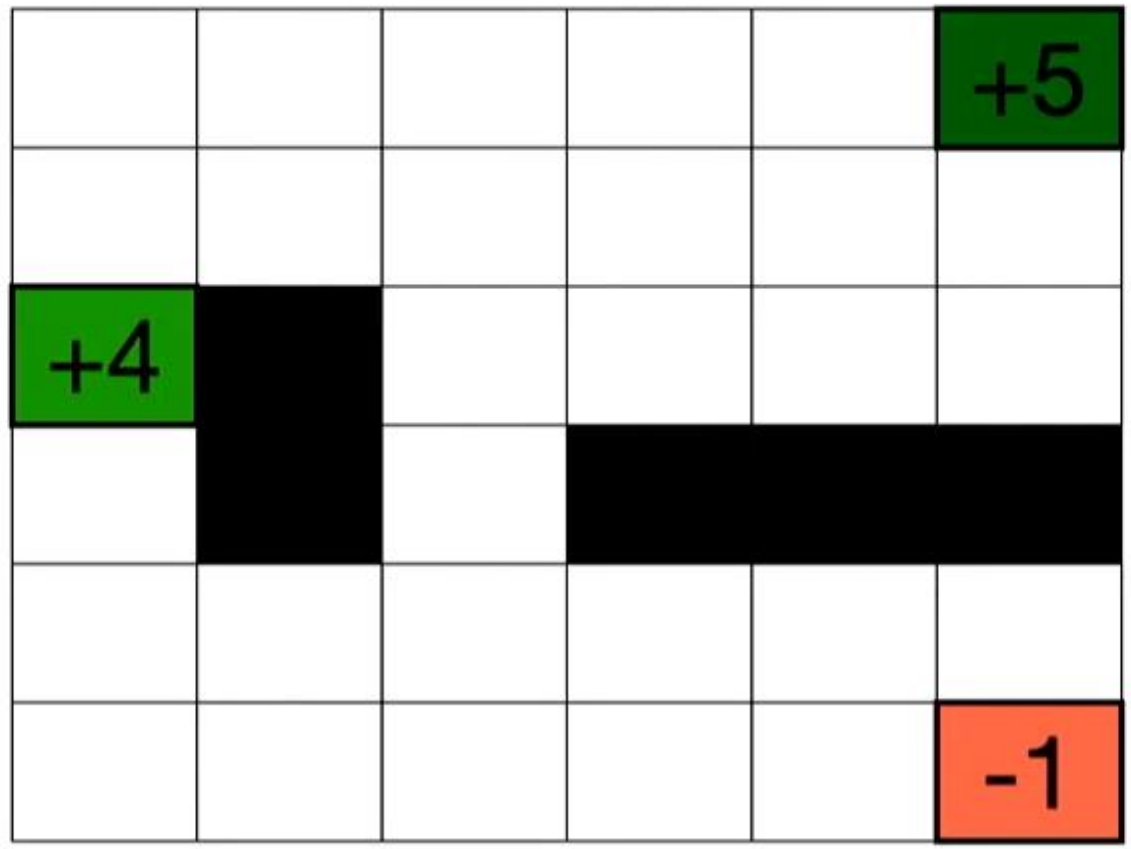
Bellman equation

$$V(s) = max_a \left(R(s,a) + V(s')\right) \qquad V(s) = max_a \left(\gamma V(s')\right)$$



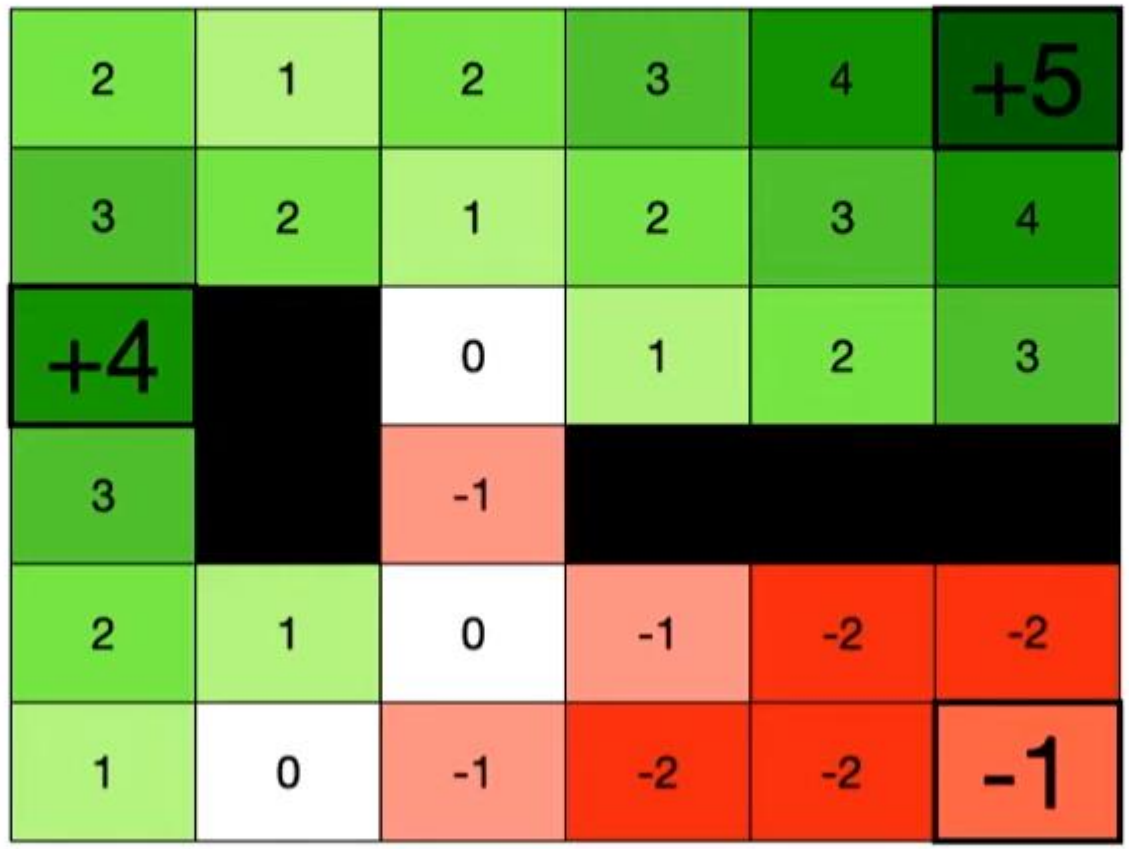


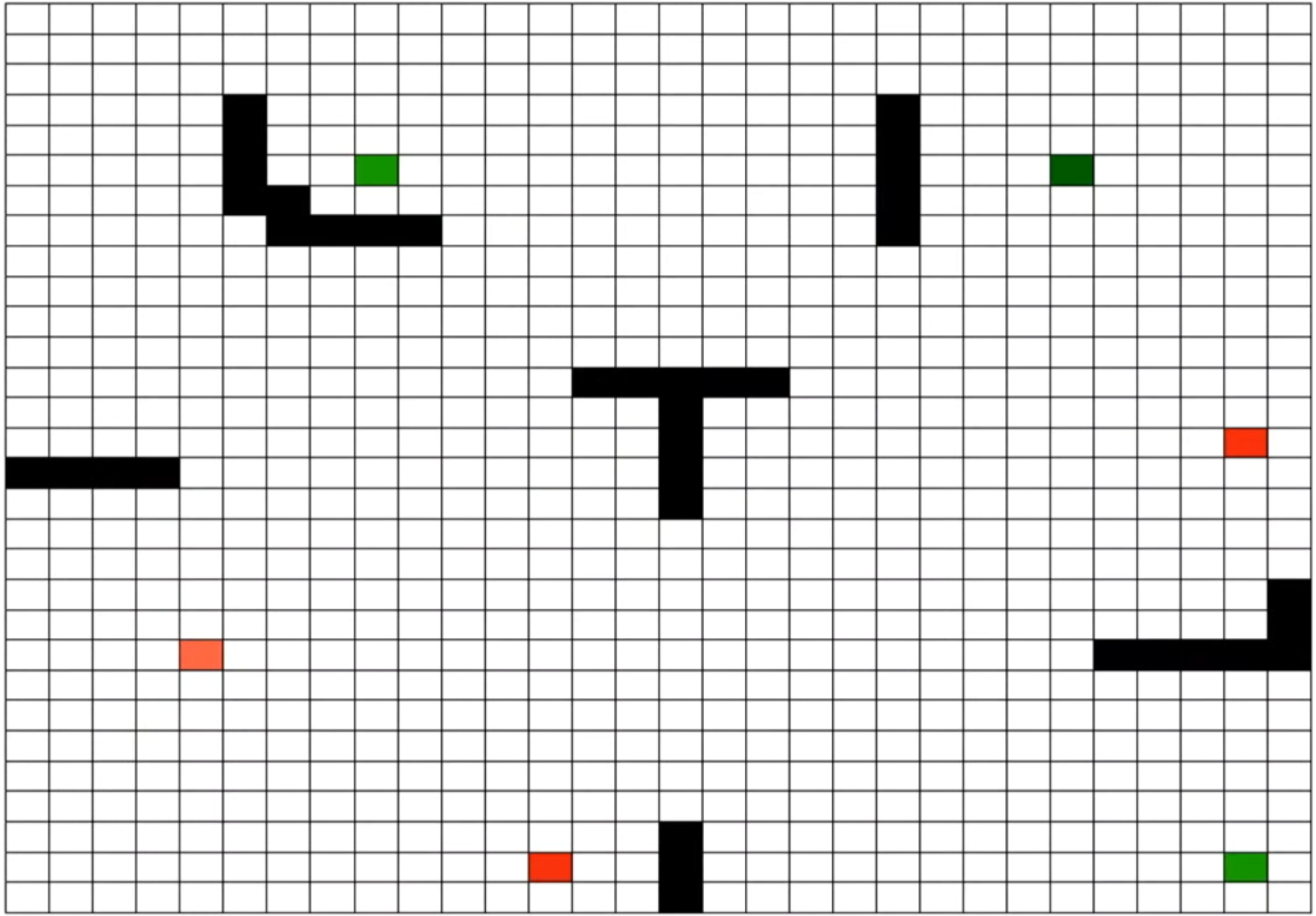
How to solve the bellman equation?



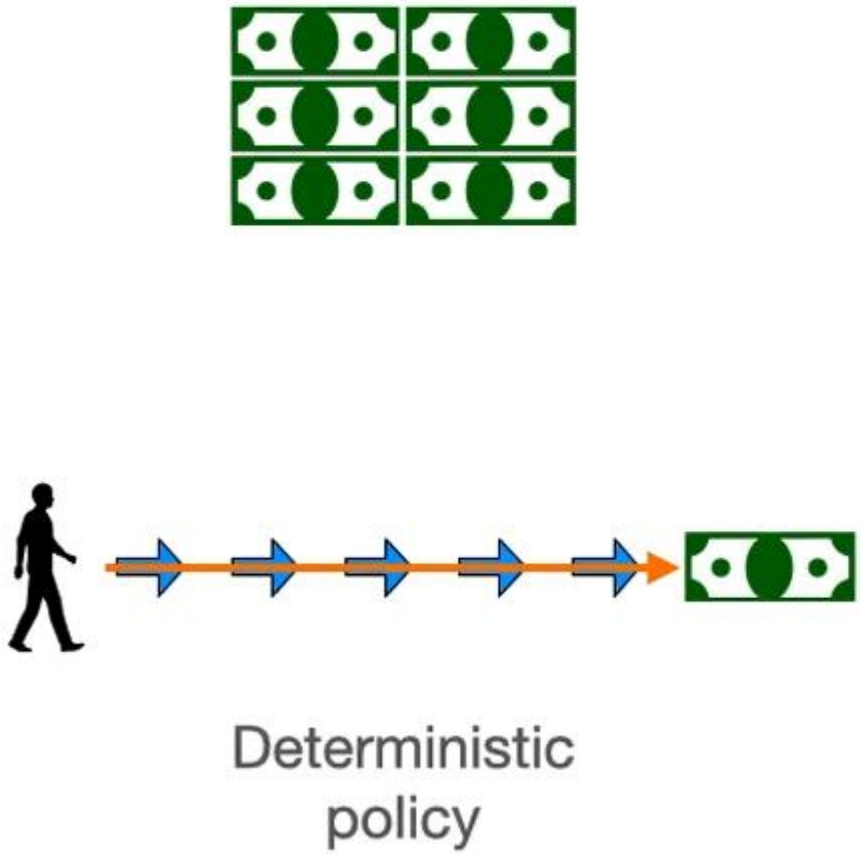
How to solve the bellman equation?

How to solve the bellman equation?

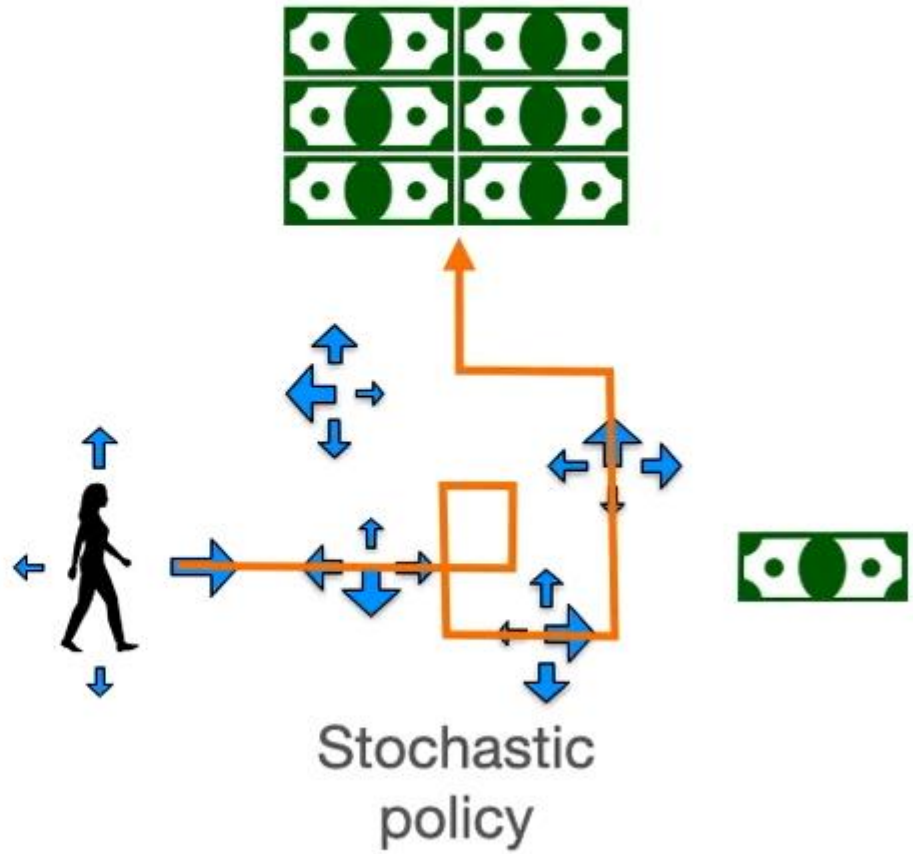




Deterministic vs stochastic

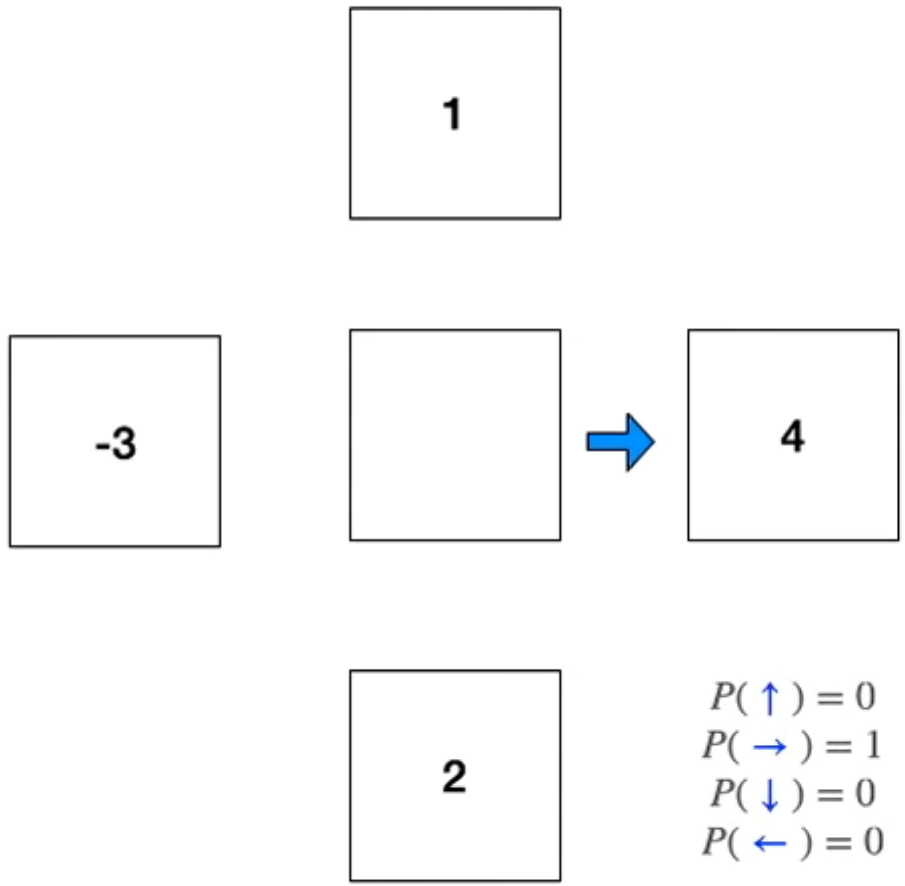


Deterministic policy



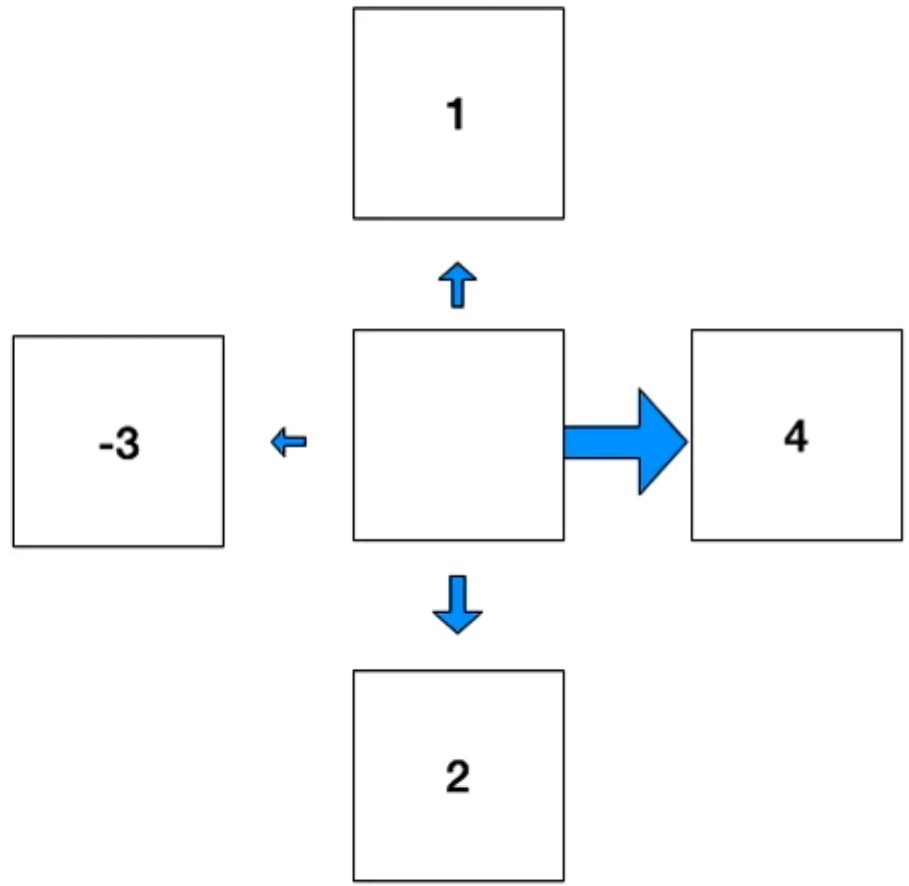
Stochastic policy

Deterministic vs stochastic



$P(\uparrow) = 0$
$P(\rightarrow) = 1$
$P(\downarrow) = 0$
$P(\leftarrow) = 0$

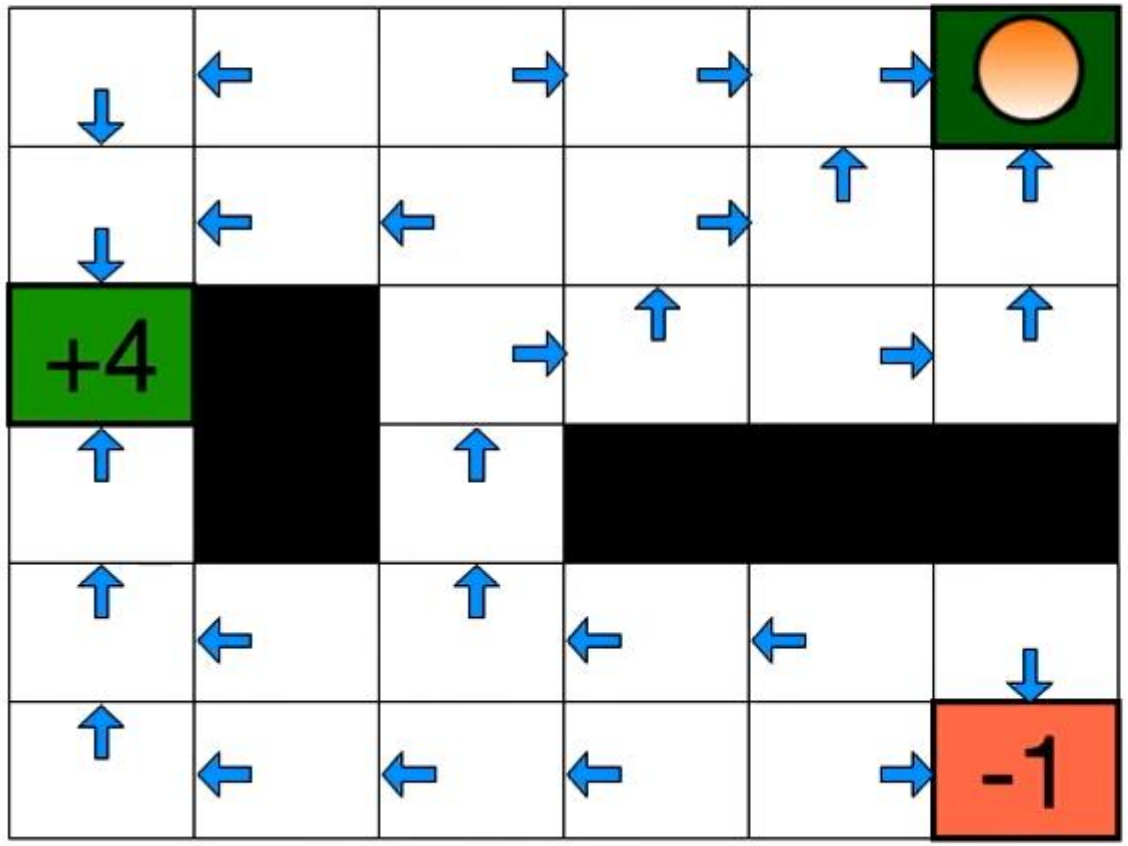
Deterministic



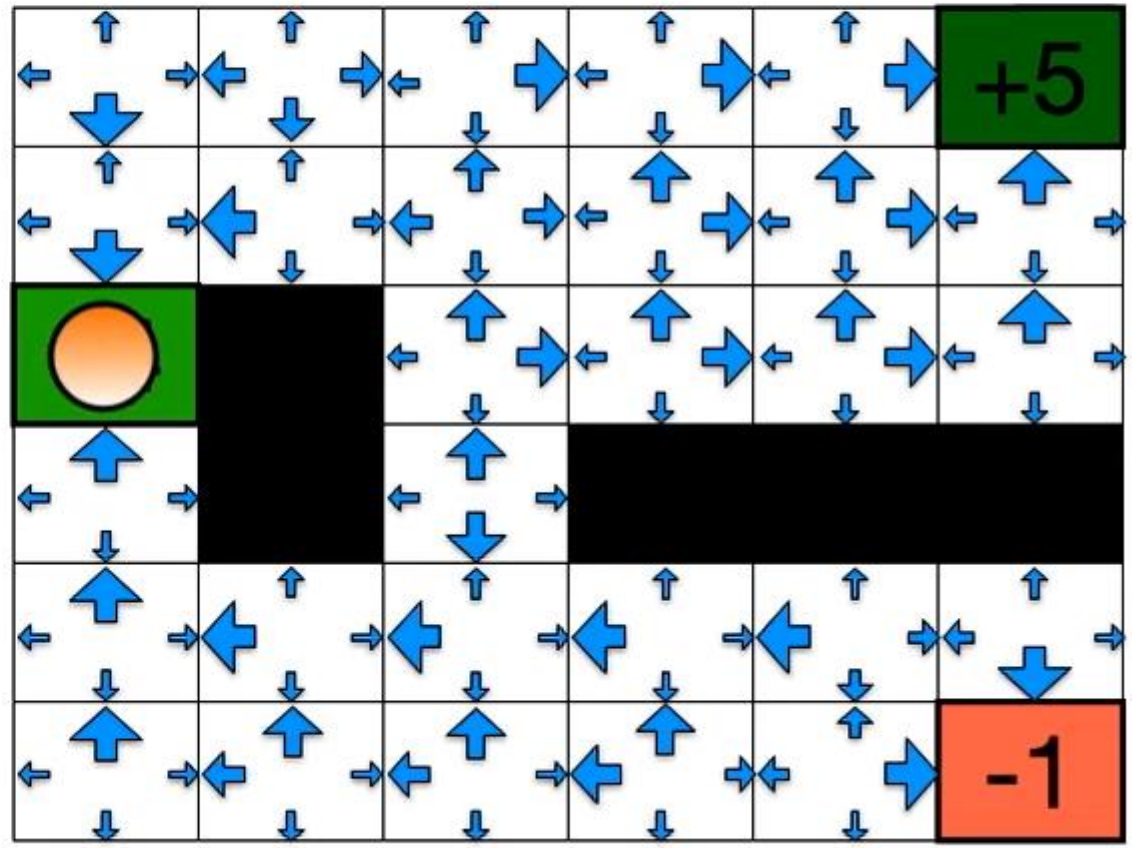
Stochastic

Deterministic vs stochastic

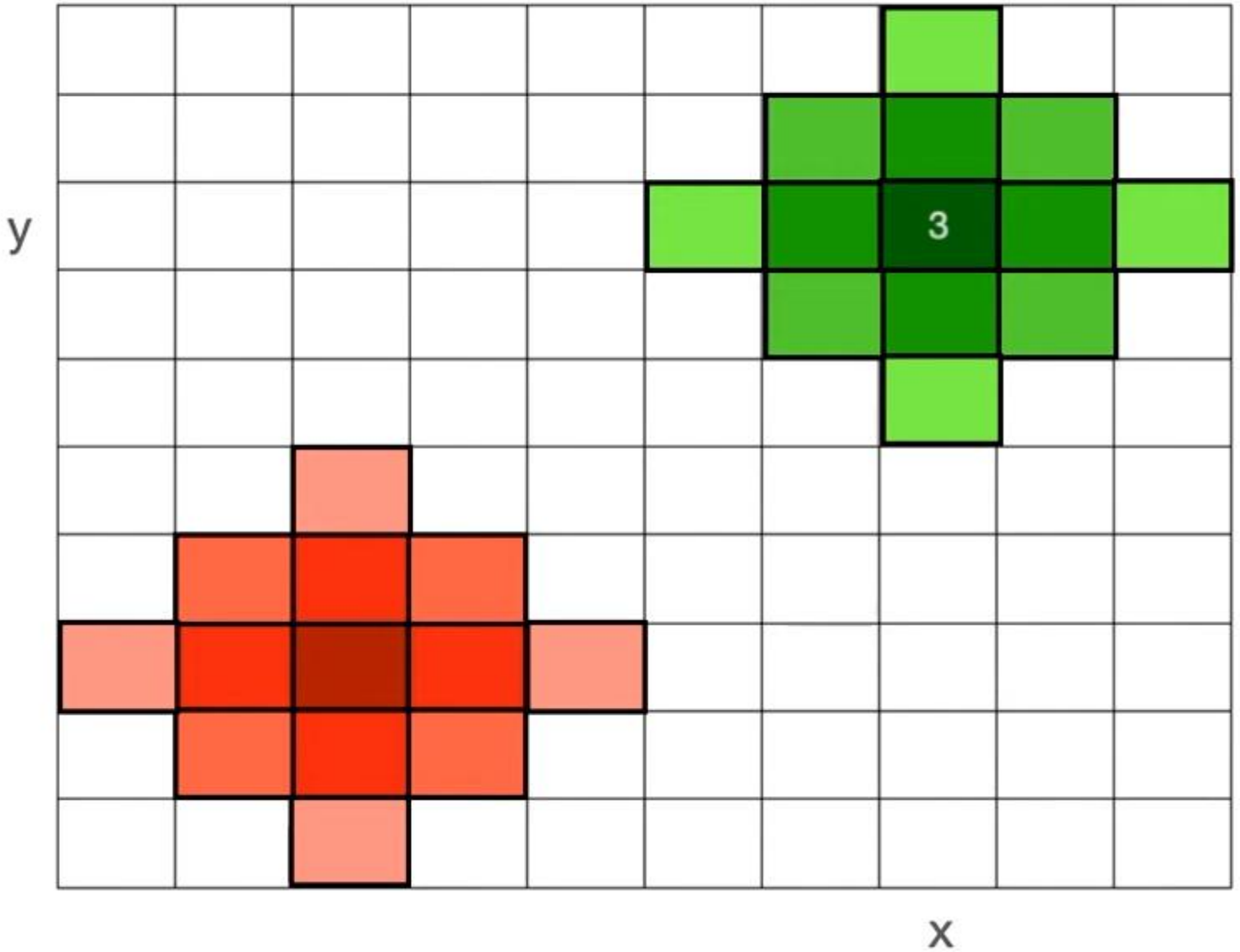
Deterministic

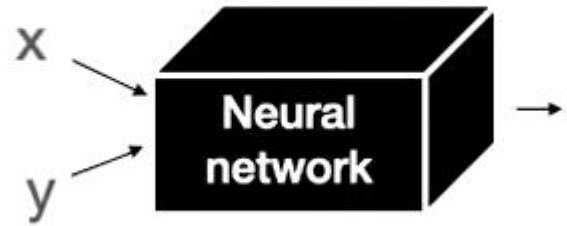


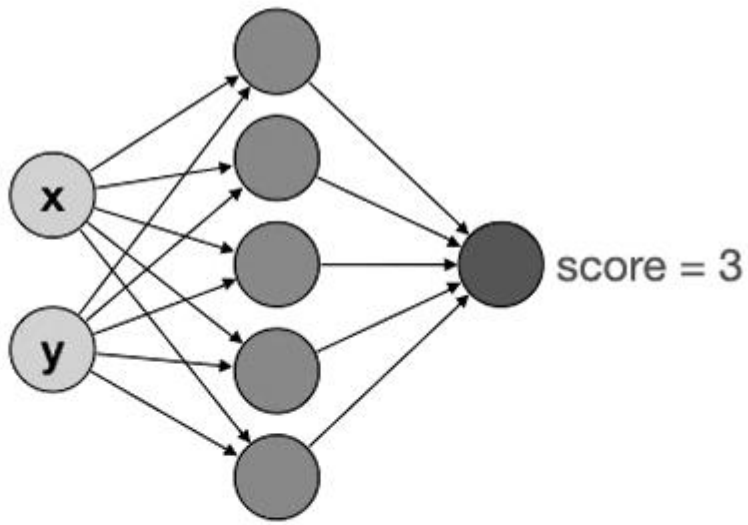
Stochastic



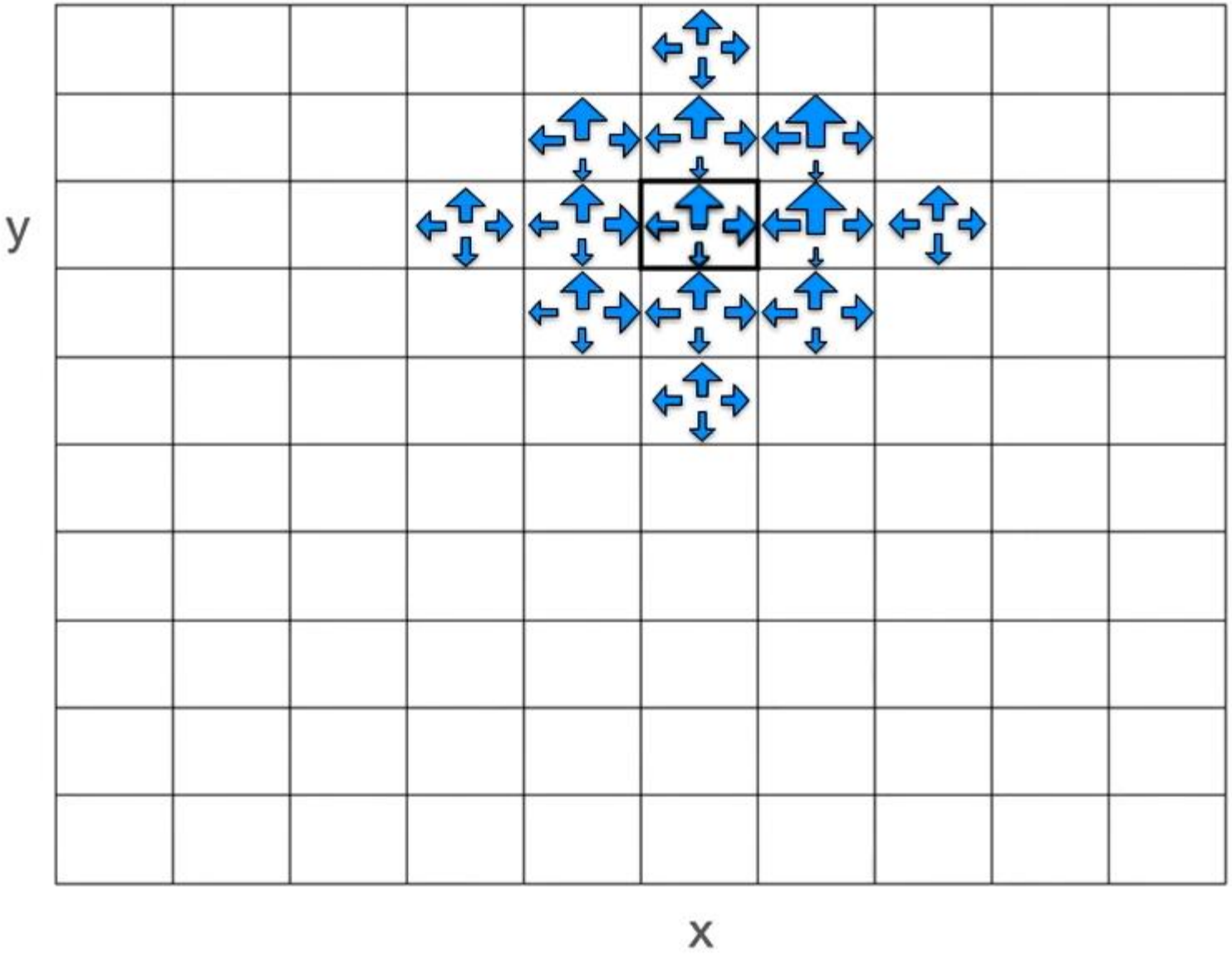
Value network

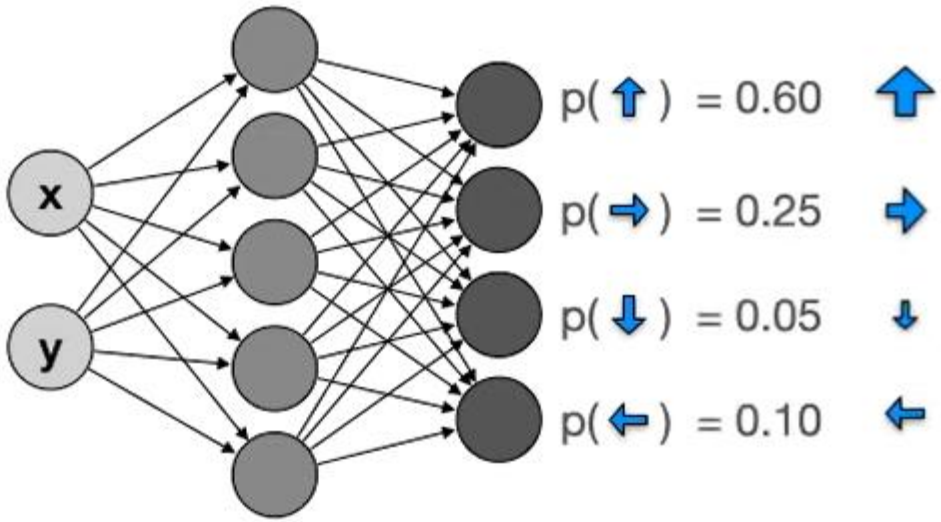






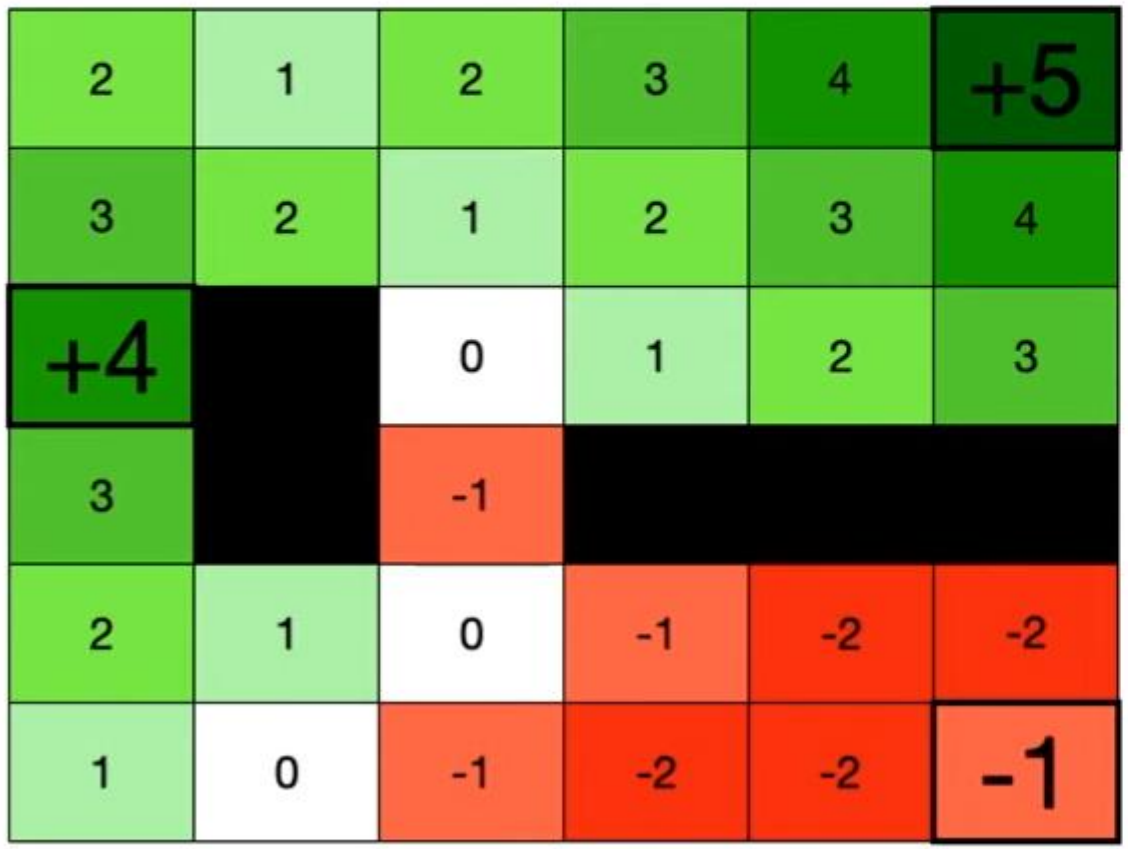
Policy network



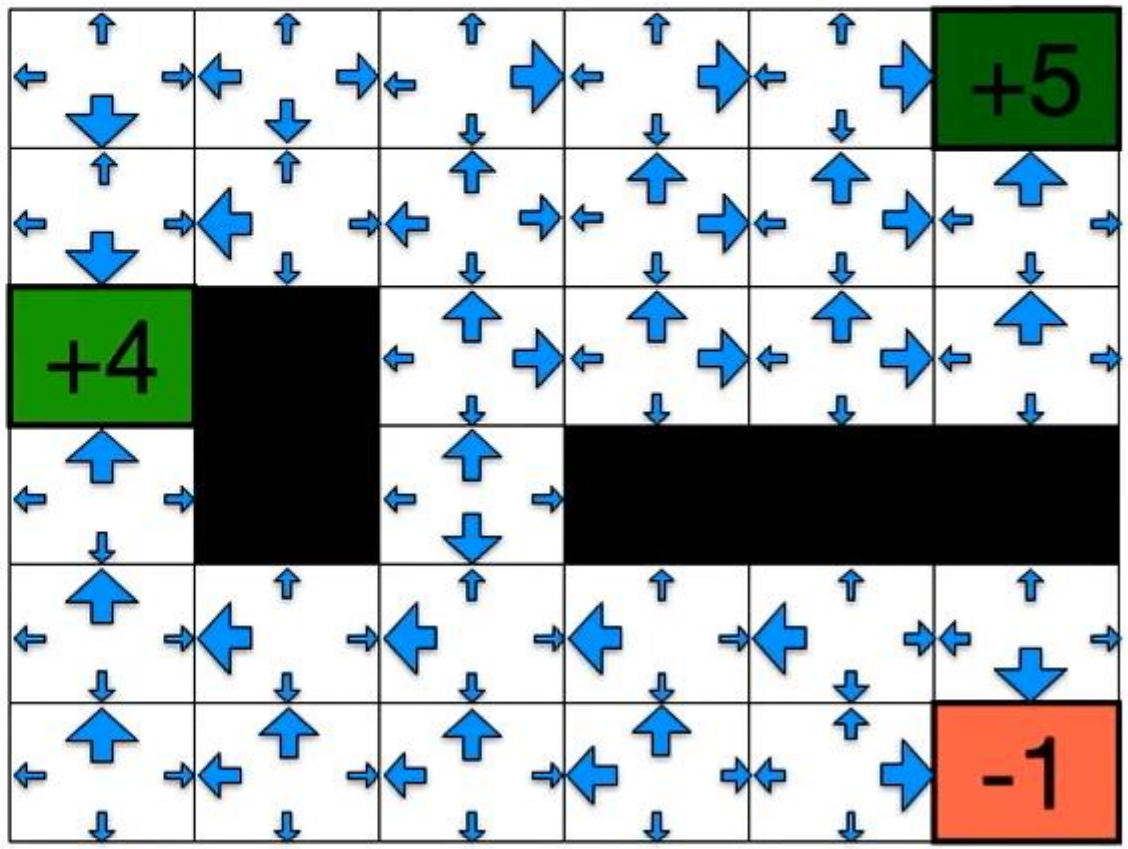


Two neural networks

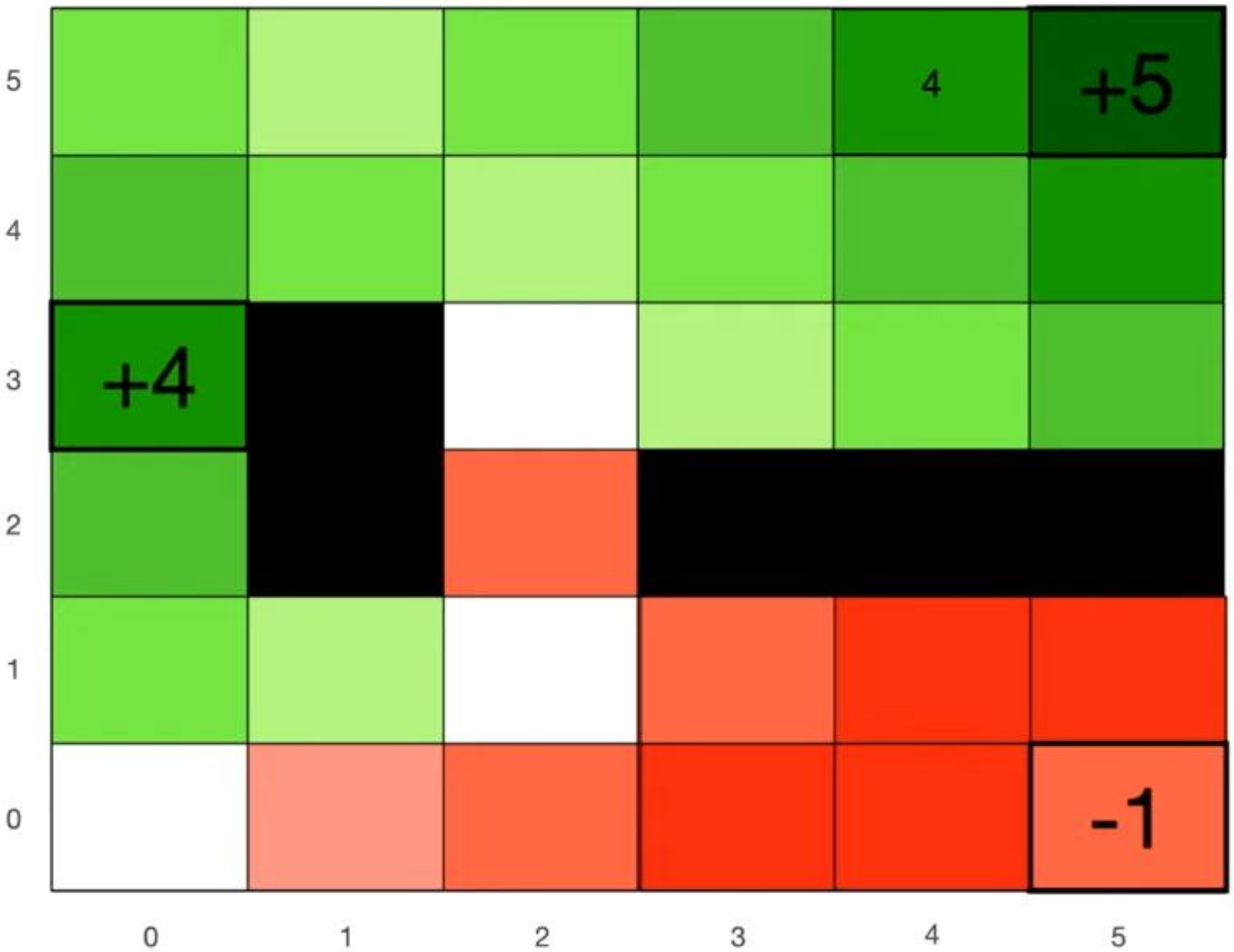
Value neural network

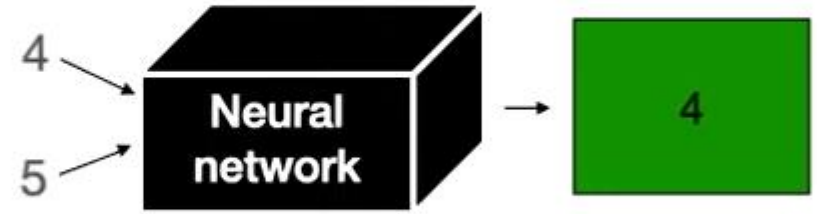


Policy neural network



Value neural networks





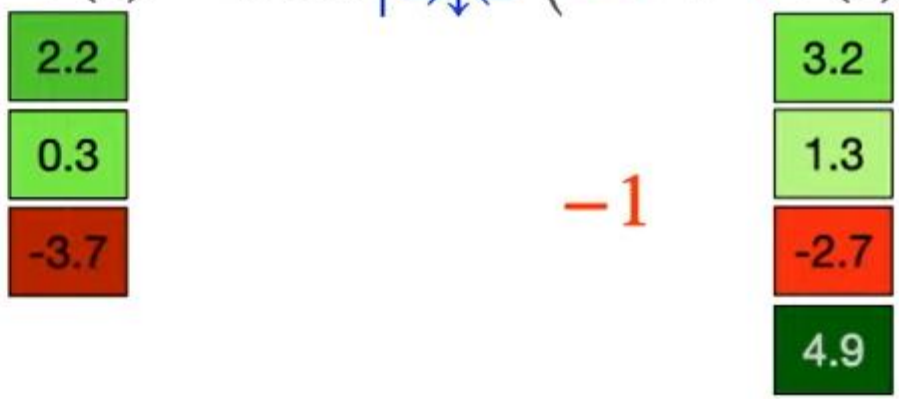
$$V(s) = max_a \left(R(s,a) + \gamma V(s) \right)$$

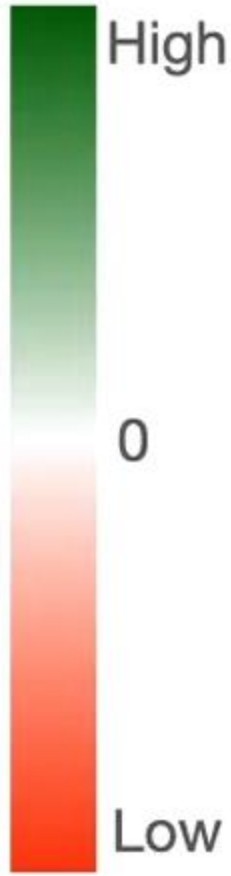
Value neural networks

features label

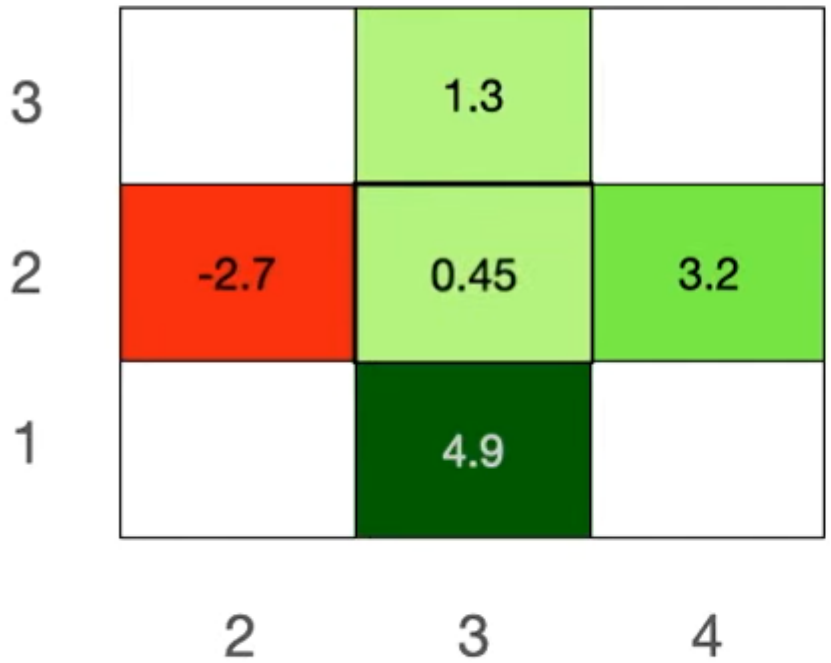
$$V(s) = max_a \left(R(s,a) + \gamma V(s) \right)$$

$$V(s) = max_{\uparrow \rightarrow \downarrow \leftarrow} \left(-1 + 1V(s) \right)$$





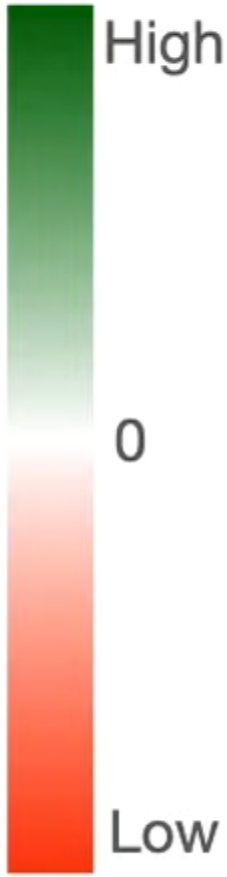
Value neural networks



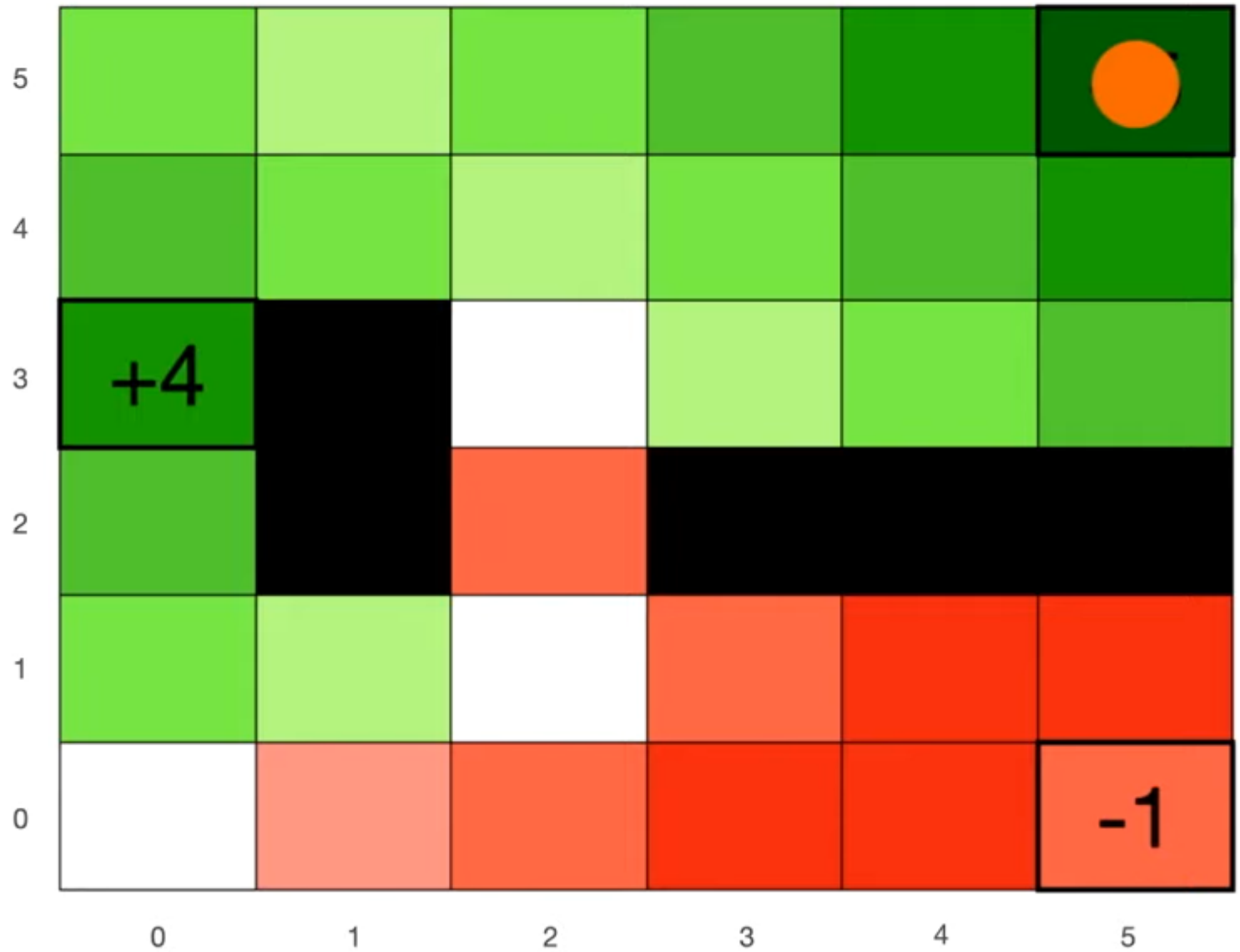
Error function

$$\left(V(s) - max_a \left(R(s,a) + \gamma V(s)\right)\right)^2$$

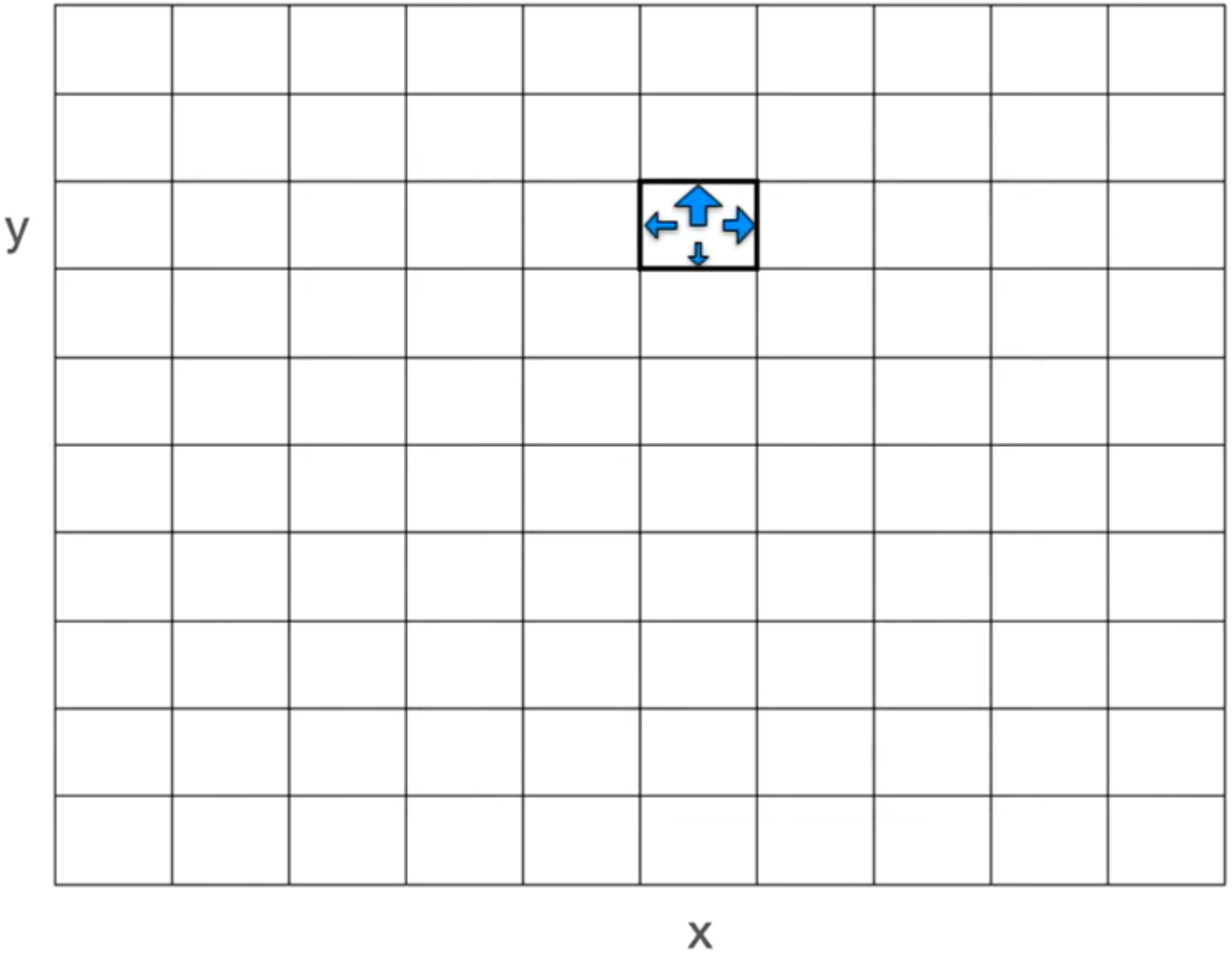
$$(0.2 - \max\{2.2, 0.3, -3.7, 3.9\})^2$$

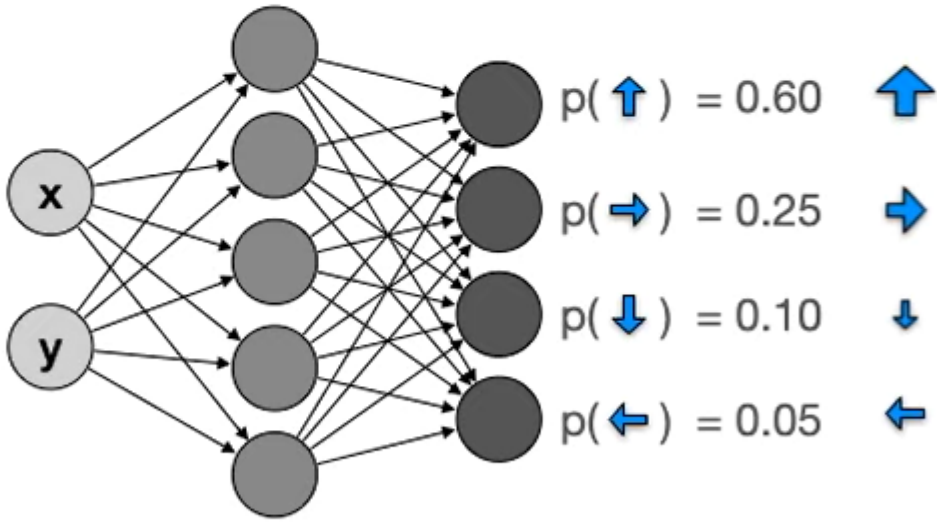


Value neural networks

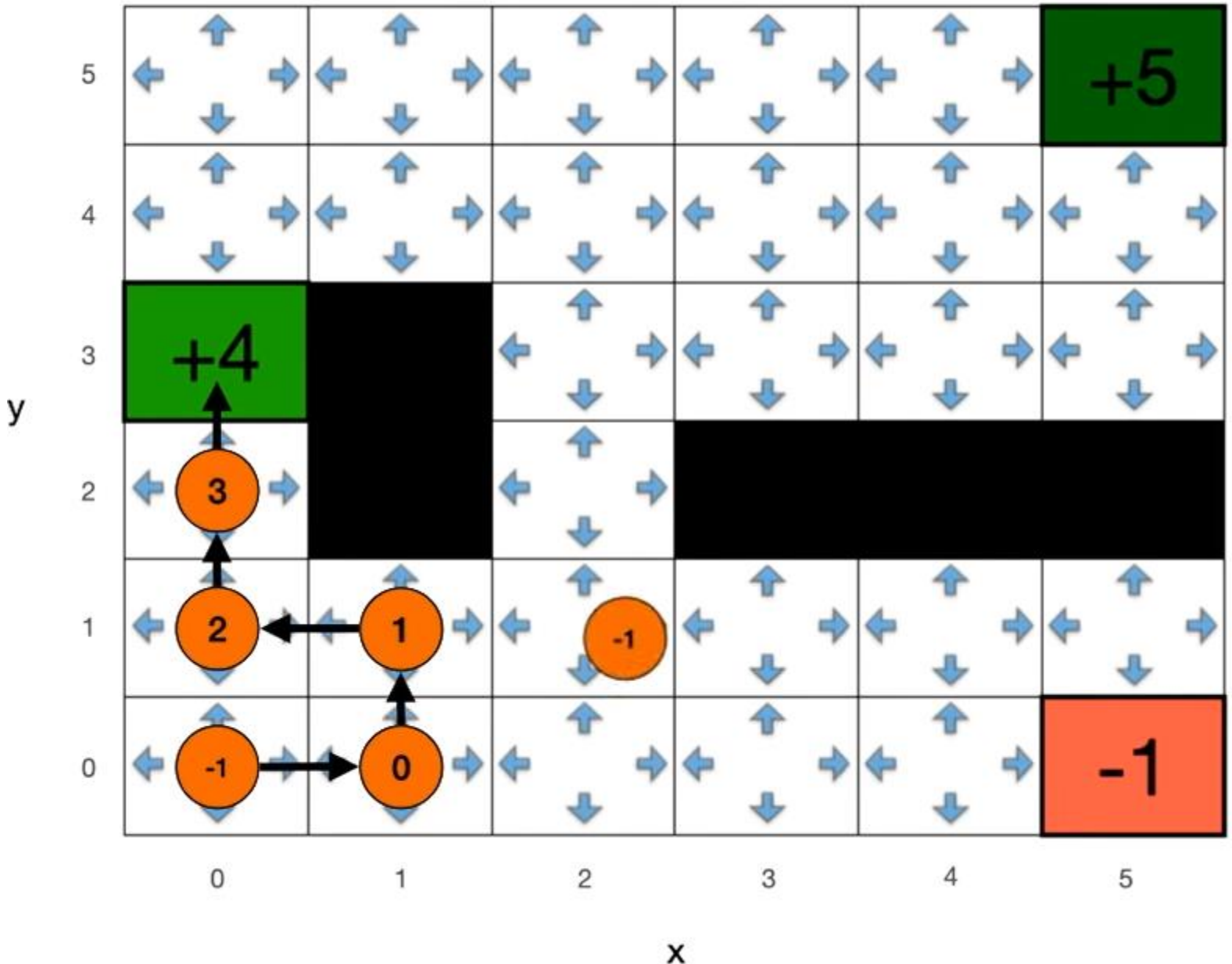


Policy neural network

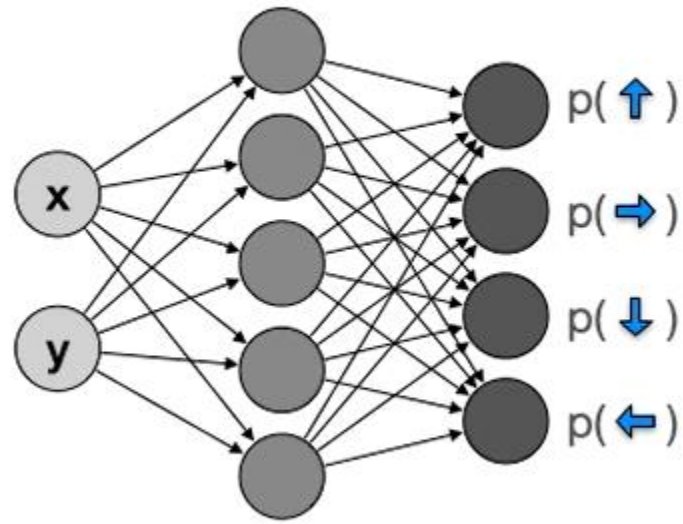




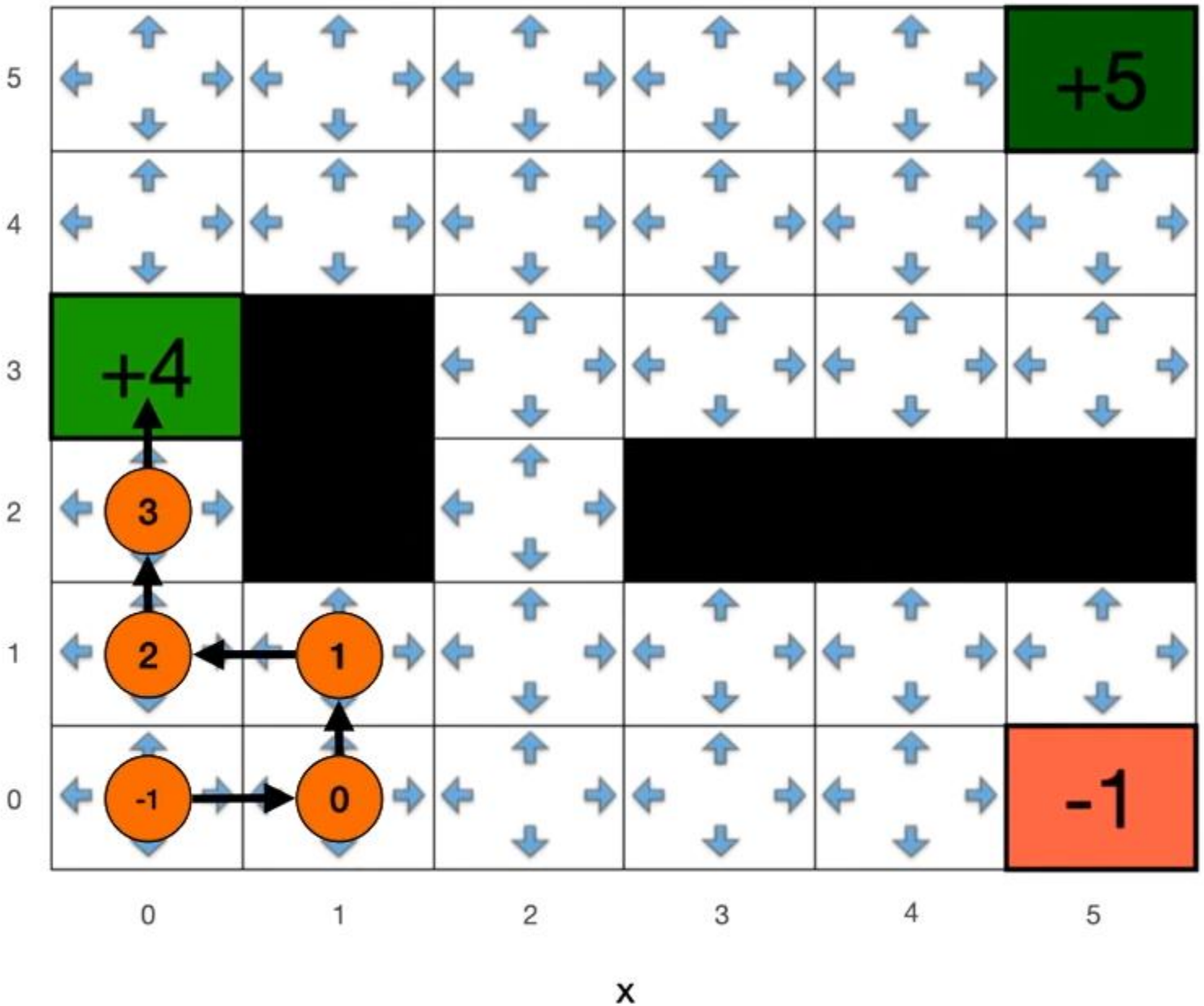
How to train it?



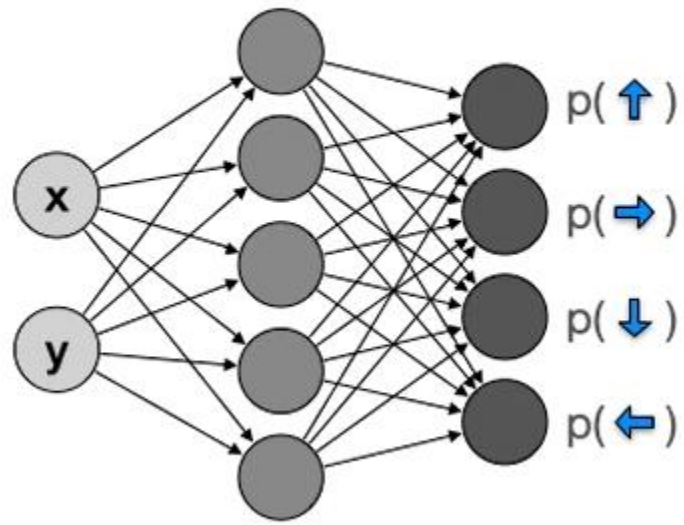
Gain	x	y	Direction
3	0	2	↑
2	0	1	↑
1	1	1	←
0	1	0	↑
	0	0	→



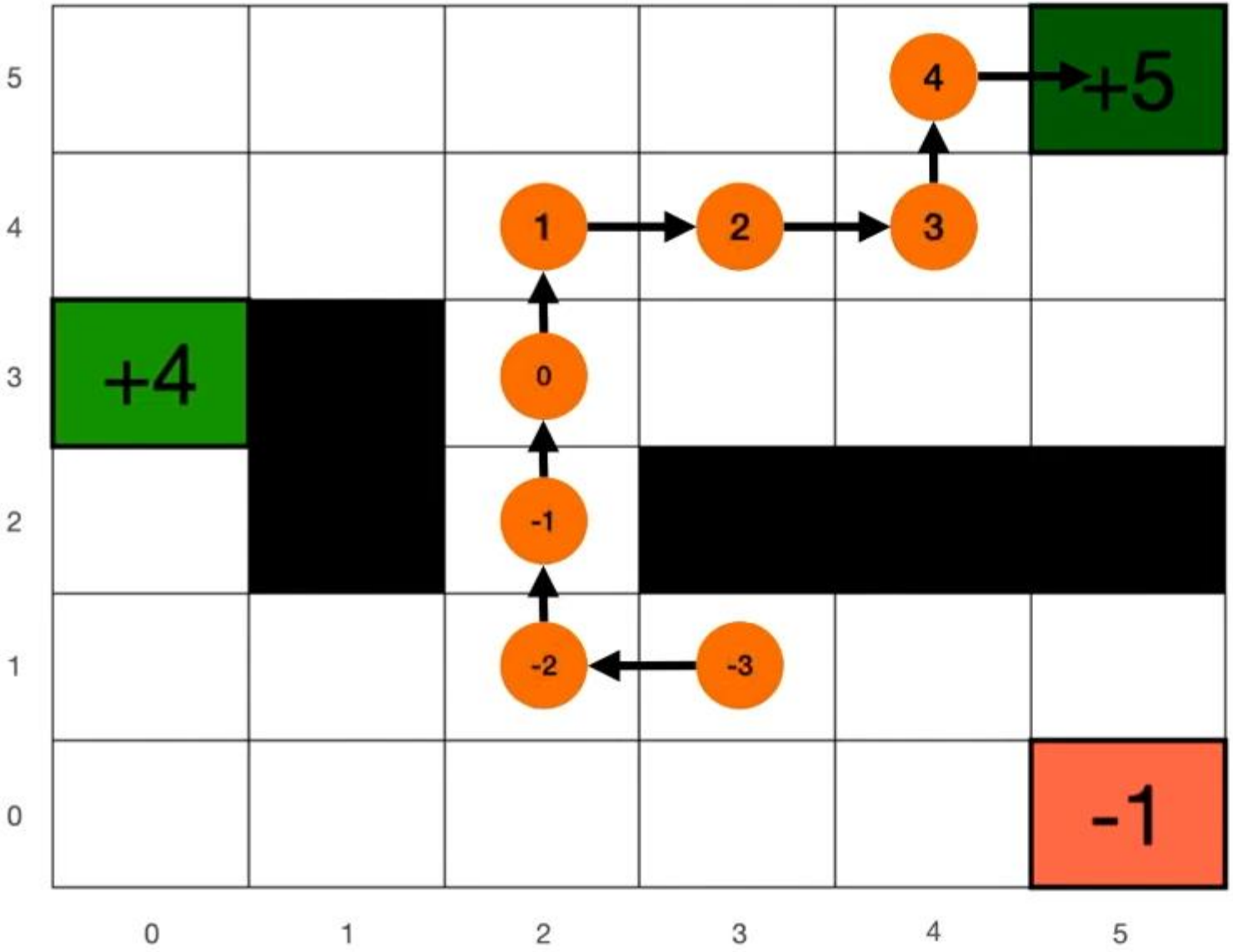
How to train it?

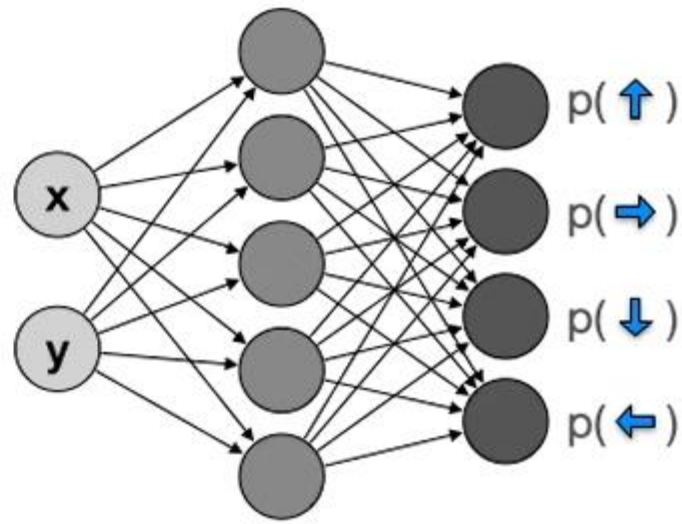


Gain	x	y	Direction	Change
3	0	2	↑	increase
2	0	1	↑	increase
1	1	1	←	increase
0	1	0	↑	stay
-1	0	0	→	decrease

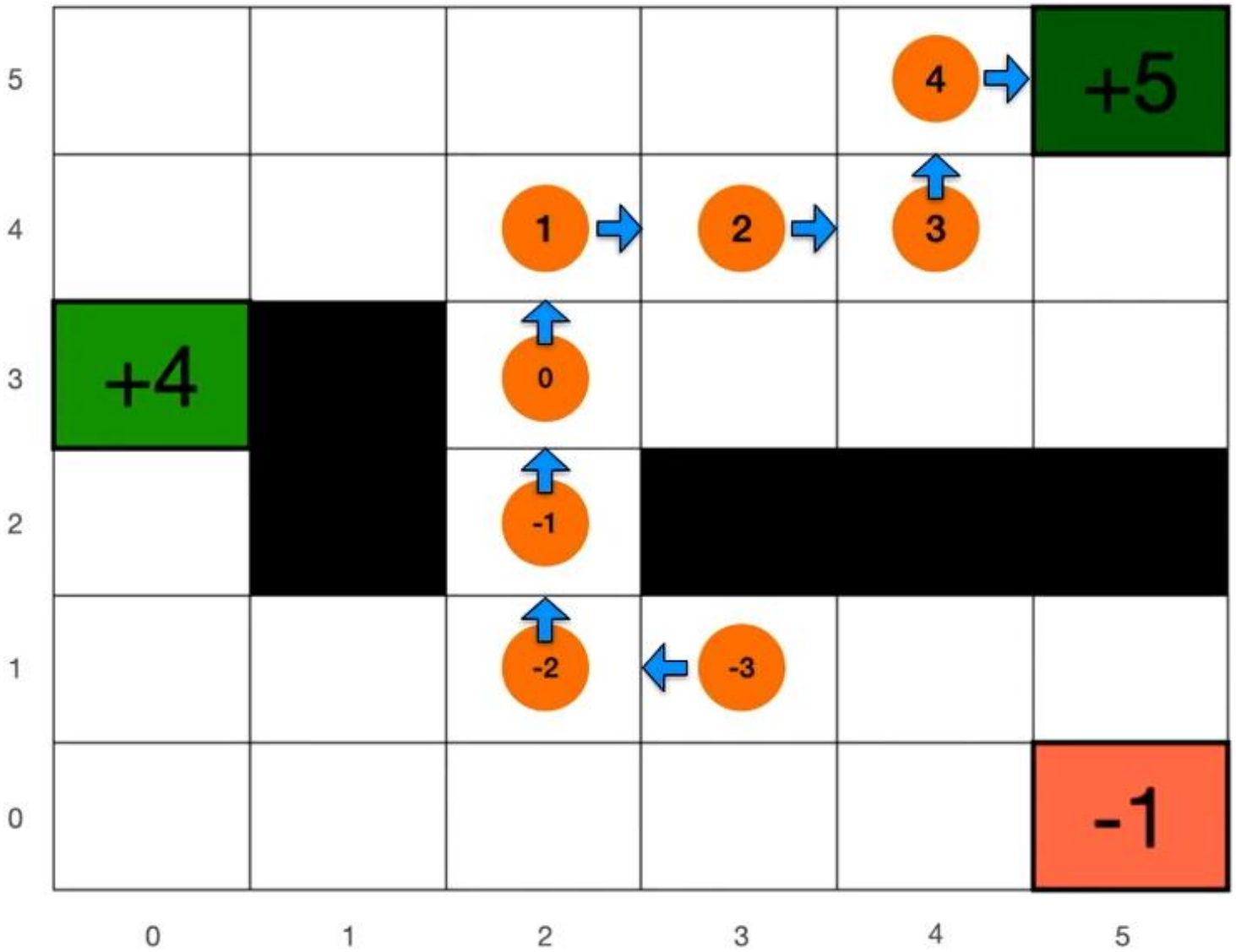


How to train it?

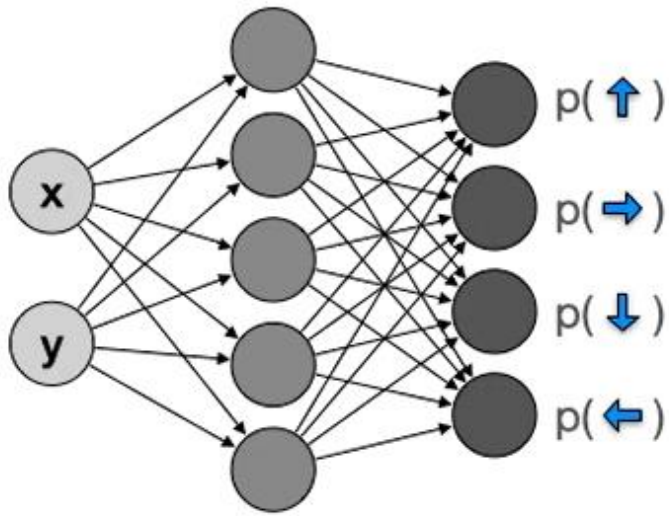




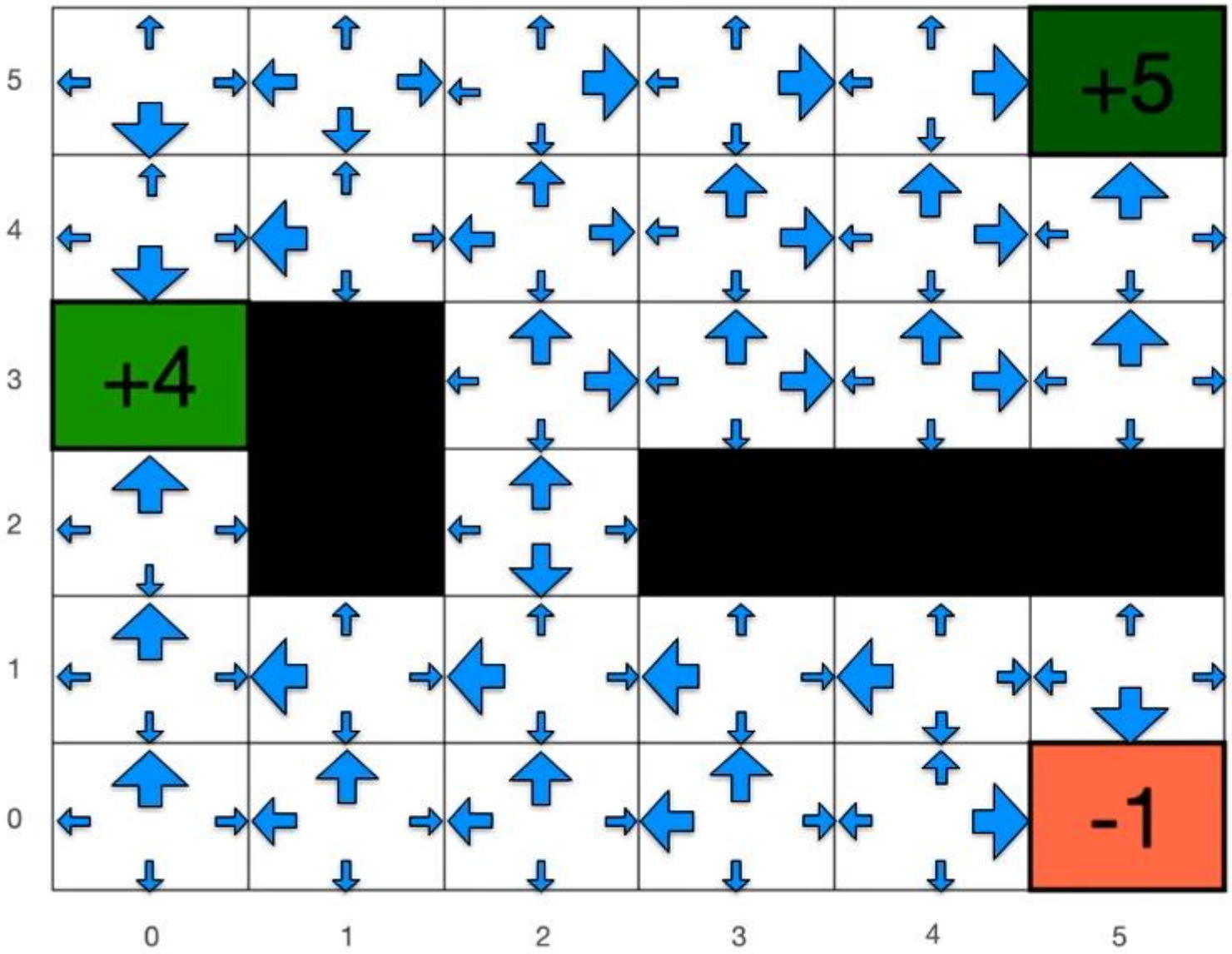
How to train it?

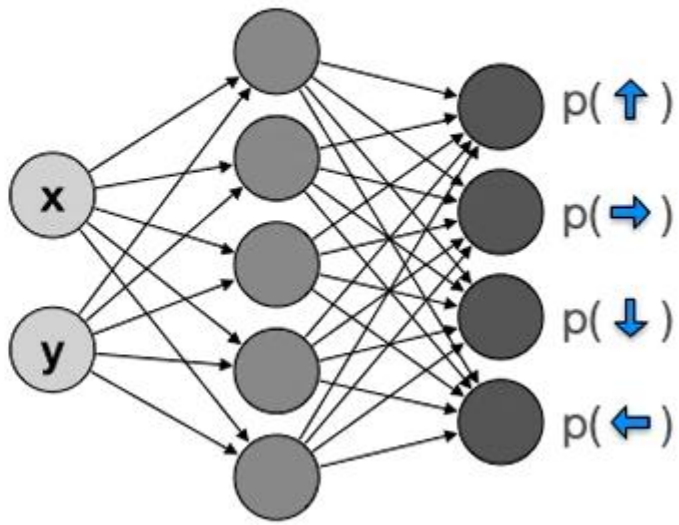


Gain	x	y	action	change
4	4	5	→	increase
3	4	4	↑	increase
2	3	4	→	increase
1	2	4	→	increase
0	2	3	↑	stay
-1	2	2	↑	decrease
-2	2	1	↑	decrease
-3	3	1	←	decrease



After many iterations…



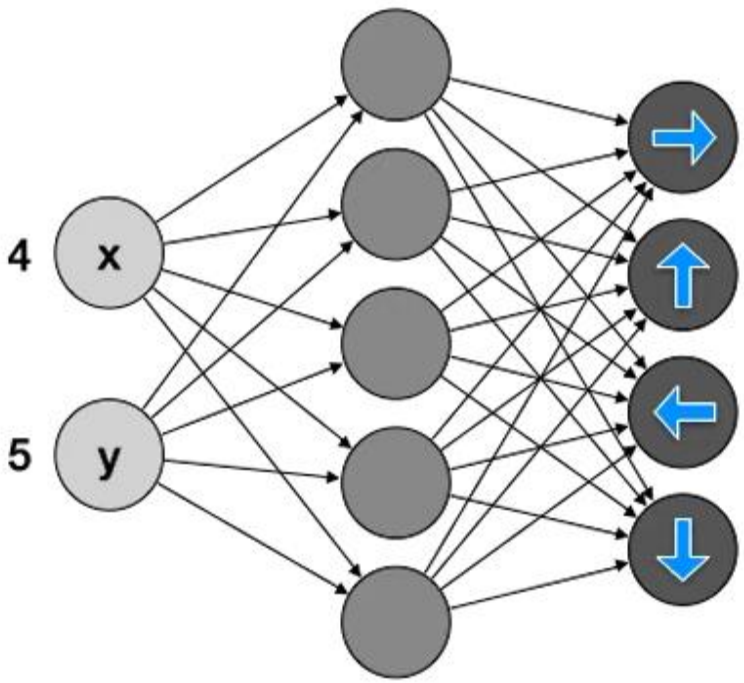


Loss function in the policy neural network

Coordinates
4, 5

Direction

Gain
-3



What I got (probability)	What I wanted (label)	Task	Gradient ascent
$p(\uparrow) = 0.2$	$p(\uparrow) = 1$	Increase the 0.2	Increase -3 $\ln(0.2)$

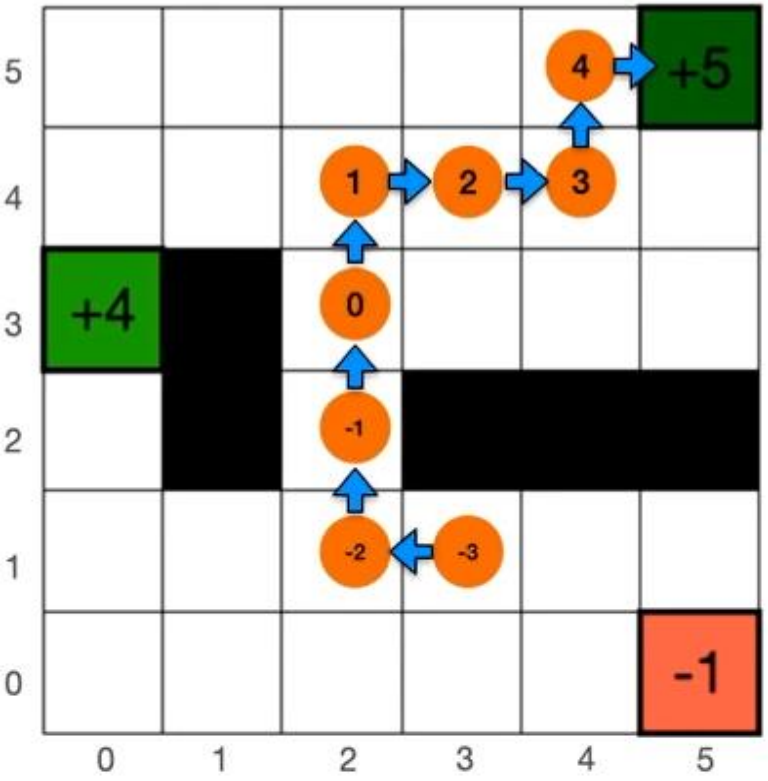
Gradient ascent step

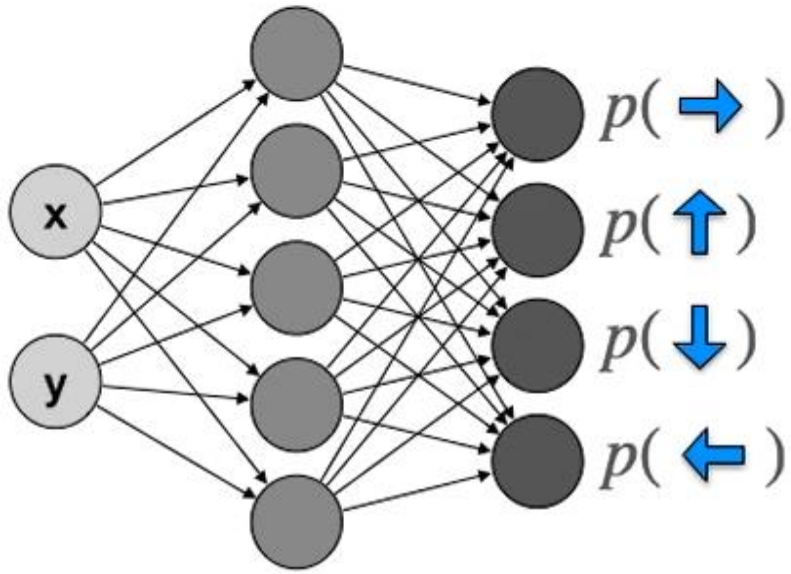
$w \to w +$ (learning rate)(derivative)

Gradient ascent step

$w \to w +$ -3 (learning rate)(derivative)

Training the policy network





Gain	x	y	action	Probability	Increase log loss	change
4	4	5	→	$p(\rightarrow) = 0.3$	$4 \ln(p(\rightarrow))$	increase
3	4	4	↑	$p(\uparrow) = 0.9$	$3 \ln(p(\uparrow))$	increase
2	3	4	→	$p(\rightarrow) = 0.1$	$2 \ln(p(\rightarrow))$	increase
1	2	4	→	$p(\rightarrow) = 0.2$	$1 \ln(p(\rightarrow))$	increase
0	2	3	↑	$p(\uparrow) = 0.5$	$0 \ln(p(\uparrow))$	stay
-1	2	2	↑	$p(\uparrow) = 0.4$	$-1 \ln(p(\uparrow))$	decrease
-2	2	1	↑	$p(\uparrow) = 0.3$	$-2 \ln(p(\uparrow))$	decrease
-3	3	1	←	$p(\leftarrow) = 0.7$	$-3 \ln(p(\leftarrow))$	decrease

Policy gradients