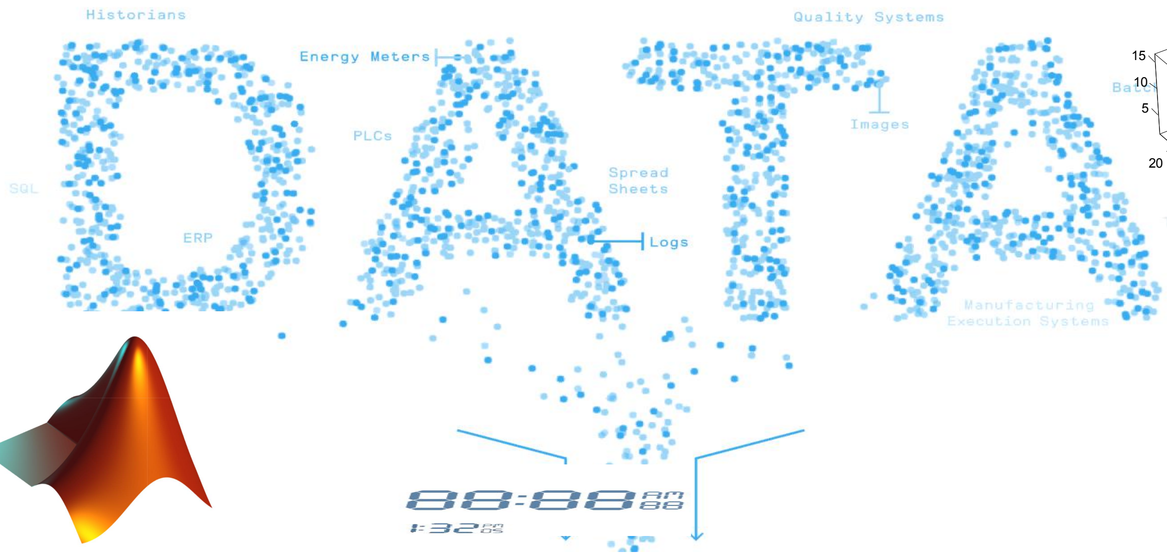
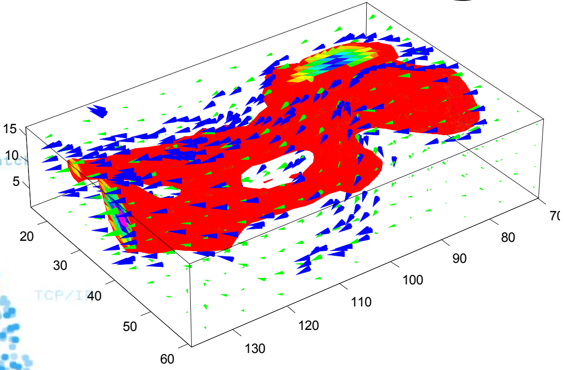
# Introduction to Artificial Intelligence
# - 05-01-03 Reinforcement Learning



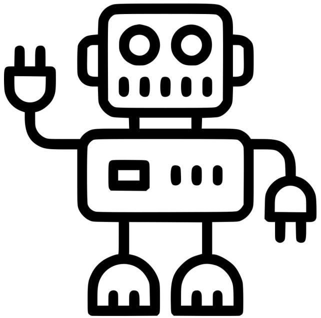


**Dr Leo Chen**
leo.chen@ieee.org

06/Sep/2021

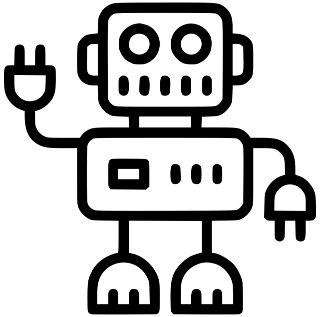
# Module Contents

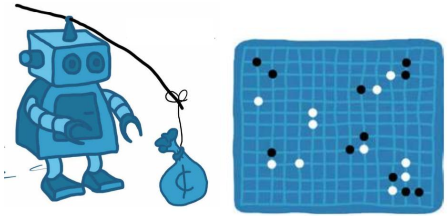
1. Introduction
2. Evolutionary Computation
3. Artificial Neural Network
4. Fuzzy Logic and Fuzzy Systems
5. More AI Subsets
6. AI and Industry 4.0
7. AI Applications
8. Labs
9. Courseworks

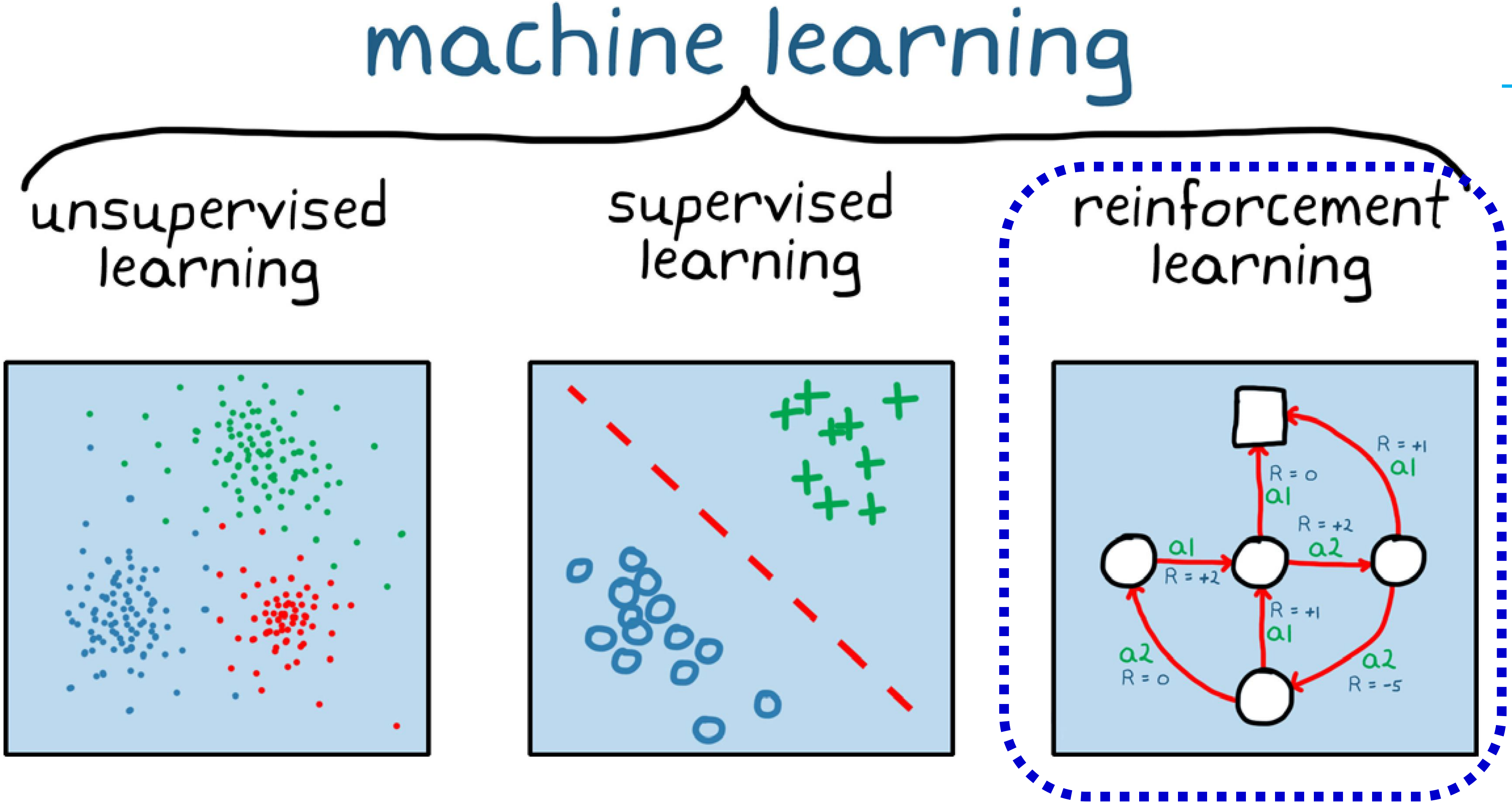


# Chapter Contents



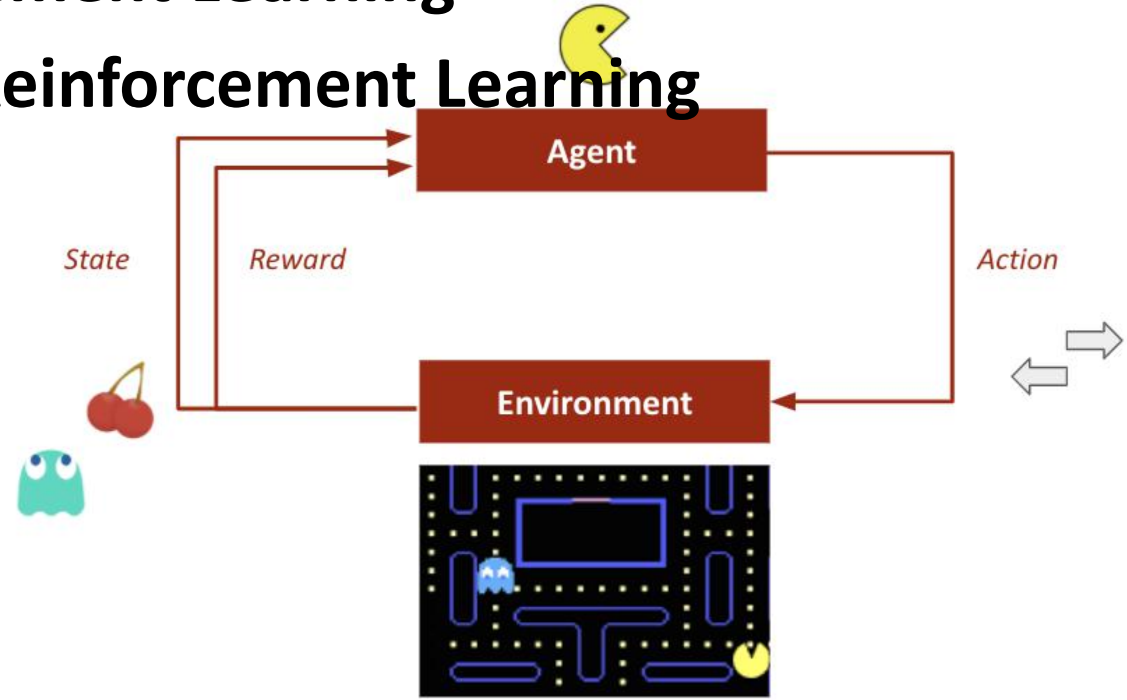




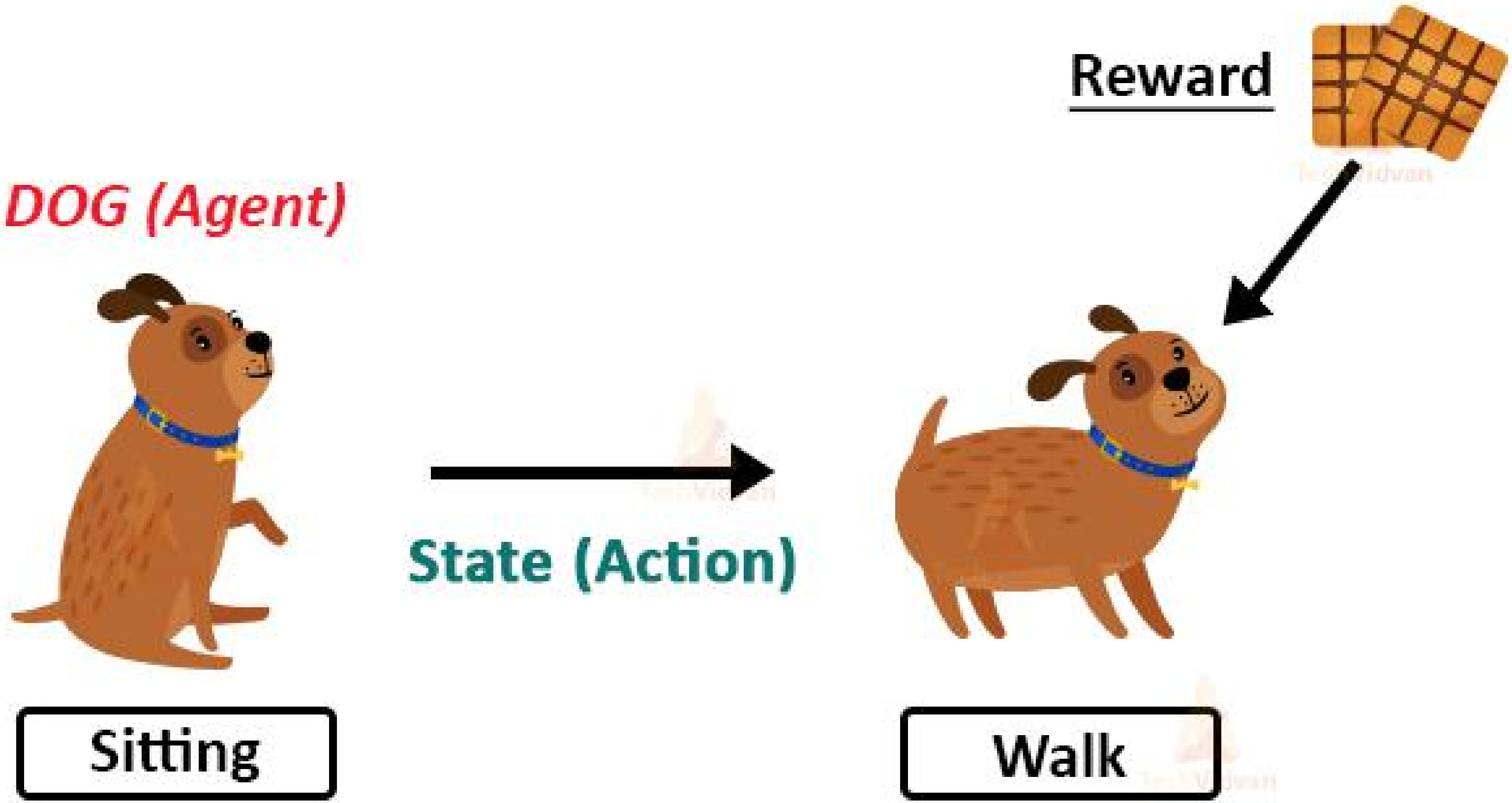

machine learning
unsupervised learning
supervised learning
reinforcement learning
R = +1
a1
R = 0
a1
R = +2
a2
a1
R = +2
R = +1
a1
a2
R = 0
a2
R = -5

# Section Contents

1. What Is Reinforcement Learning
2. Why Use Reinforcement Learning
3. Fundamentals of Reinforcement Learning
4. Applications



# Reinforcement Learning in ML

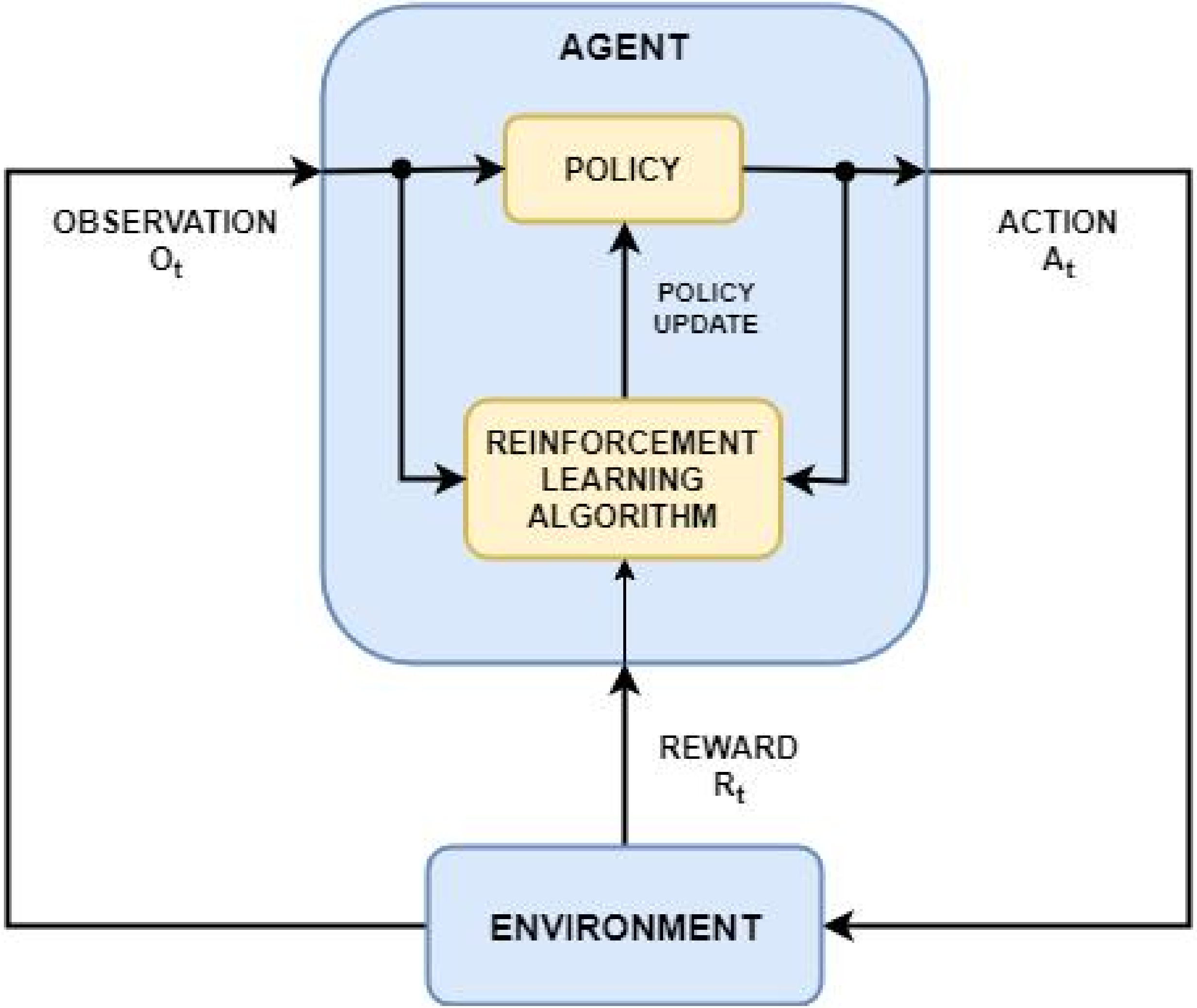


# 1 What Is Reinforcement Learning

- Reinforcement learning is a **goal-directed** computational approach where a computer learns to **perform** a task by **interacting** with **an unknown dynamic environment**.
- This learning approach **enables** a computer to **make a series of decisions** to maximize the cumulative reward for the task **without** human intervention and **without** being explicitly programmed to achieve the task.

# 1 What Is Reinforcement Learning

- The **goal** of reinforcement learning is to **train** an **agent** to complete a task within **an unknown environment**.
- The **agent** receives **observations** and a **reward** from the environment and sends actions to the environment.
- The **reward is a measure** of how successful an action is with respect to completing the task goal.


AGENT
POLICY
OBSERVATION
$O_t$
POLICY
UPDATE
ACTION
$A_t$
REINFORCEMENT
LEARNING
ALGORITHM
REWARD
$R_t$
ENVIRONMENT

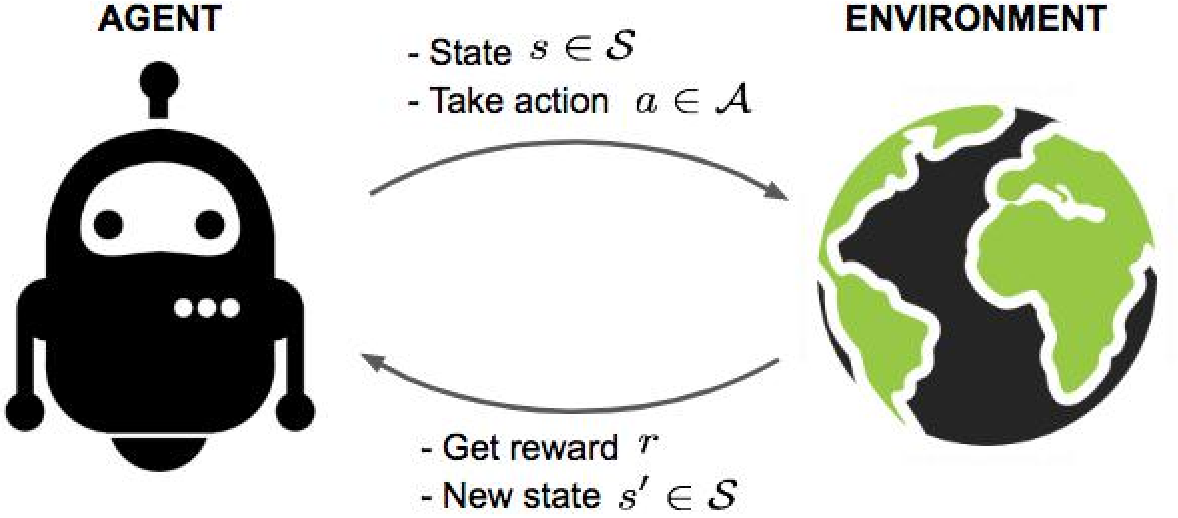
# The agent

## Two components:

- *a policy*
- *a learning algorithm*

# Policy

- The policy is a **mapping** that **selects actions** based on the **observations** from the **environment**.
- Typically, the policy is a **function approximator** with **tunable parameters**, such as a *deep neural network*.



# Learning algorithm

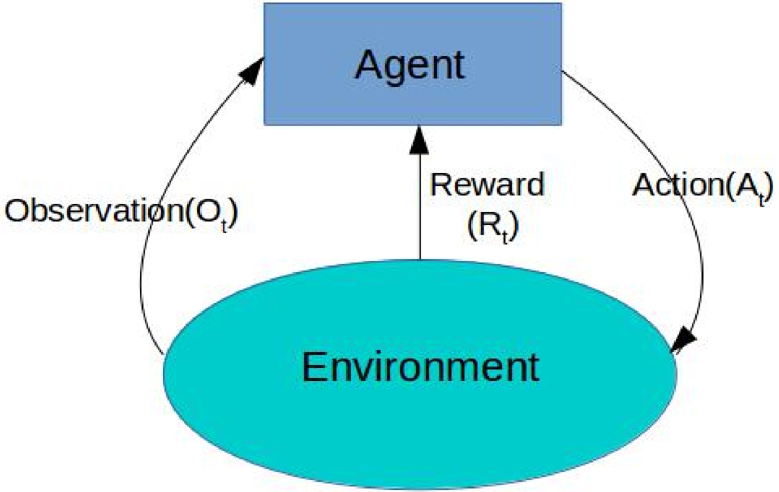
- The learning algorithm **continuously updates** the **policy parameters** based on the actions, observations, and reward.
- The goal of the learning algorithm is to **find an optimal policy** that **maximises** the cumulative **reward** received during the task.

# 2 Why Use Reinforcement Learning

- It helps you to find which **situation needs an action**
- Helps you to **discover** which **action** yields the **highest reward** over the **longer period**.
- Reinforcement Learning also provides the **learning** agent with a **reward** function.
- It also allows it to figure out the best method for **obtaining large rewards**.

# When Not to Use Reinforcement Learning?

- When you have **enough data** to solve the problem with a **supervised** learning method
- Reinforcement Learning is **computing-heavy** and **time-consuming**. in particular when the action space is **large**.



# 2 Why Use Reinforcement Learning

- Reinforcement learning involves an **agent** **learning** the **optimal *behavior*** through ***repeated*** **trial-and-error interactions** with the environment without human involvement.
- As an example, consider the task of parking a vehicle using an **automated driving** system.

# Automated driving system

The goal of this task is for the **vehicle computer (agent)** to park the vehicle in the **correct position and orientation**, to do so the **controller** uses:

- readings from cameras, accelerometers, gyroscopes, a GPS receiver, and lidar (**observations**)
- generating steering, braking, and acceleration commands (**actions**)

# Automated driving system

- The action commands are sent to the **actuators** that **control the vehicle**.
- The resulting **observations depend on** the actuators, sensors, vehicle dynamics, road surface, wind, and many other less-important factors.
- All these **factors**, that is, everything that is **not** the agent, make up the environment in reinforcement learning.

# Automated driving system

- To **learn** how to <u>generate the correct actions</u> from the observations, the computer repeatedly tries to park the vehicle using a **trial-and-error** process.
- To **guide** the learning process, you provide a **signal** that is one when the car **successfully** reaches the **desired** position and orientation and zero otherwise (**reward**).
- **During each trial**, the computer selects actions using a mapping (**policy**) initialised with some default values.

# Automated driving system

- After each trial, the computer updates the mapping to **maximise** the reward (**learning algorithm**).
- This process continues **until** the computer learns an optimal mapping that **successfully parks** the car.

# Reinforcement Learning Workflow

- Training an agent using reinforcement learning is an **iterative process**.
- Decisions and results in **later stages** can require you to return to an **earlier stage** in the learning workflow.

# For example

if the training process does **not** converge to an optimal policy within a **reasonable** amount of time, you

might have to **update** any of **the following** before retraining the agent:

- Training settings
- Learning algorithm configuration
- Policy representation
- Reward signal definition
- Action and observation signals
- Environment dynamics

# 3 Fundamentals of Reinforcement Learning

Three **methods** for reinforcement learning are

**A** Value-based

**B** Policy-based

**C** Model based learning

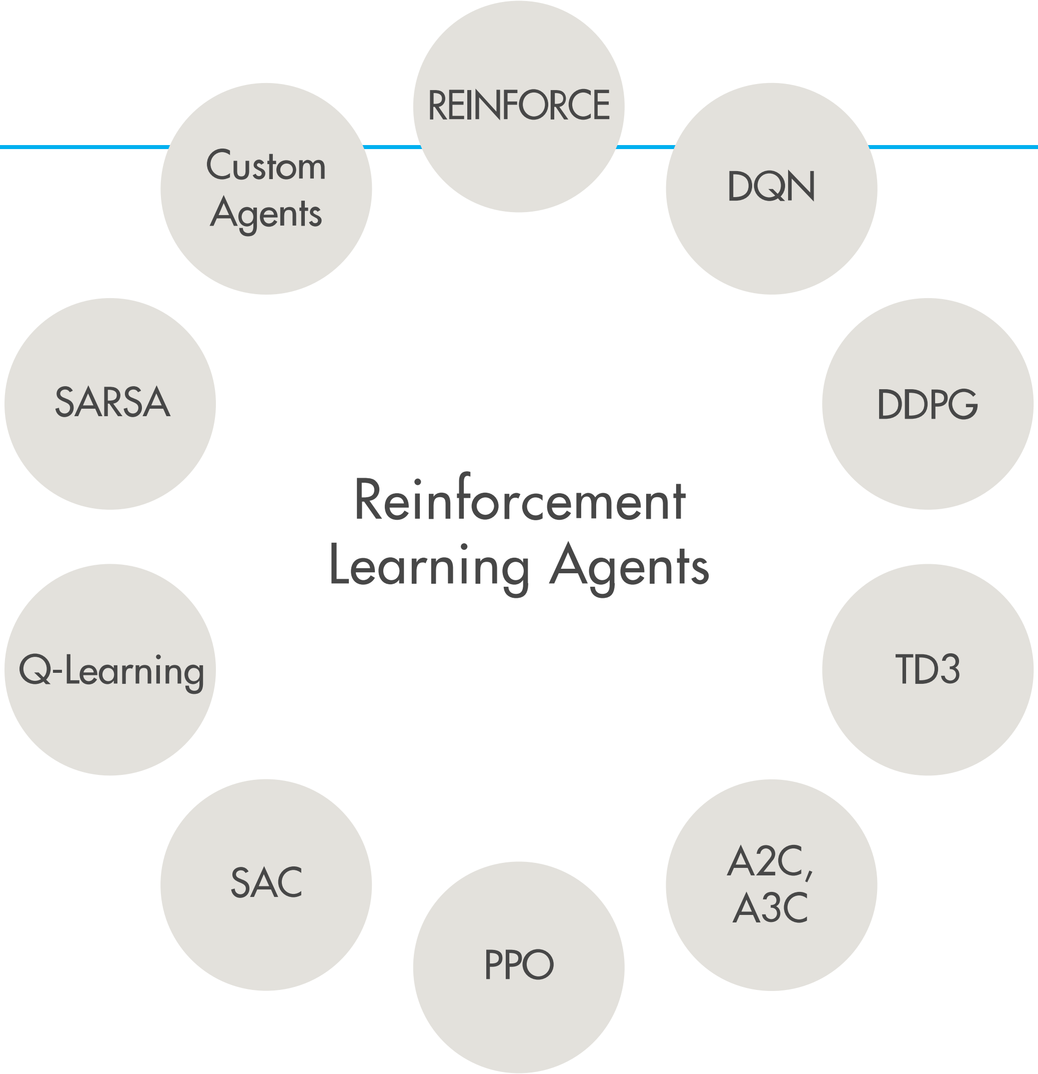
Two **types** of reinforcement learning are

**I** Positive

**II** Negative

# 3 Fundamentals of Reinforcement Learning

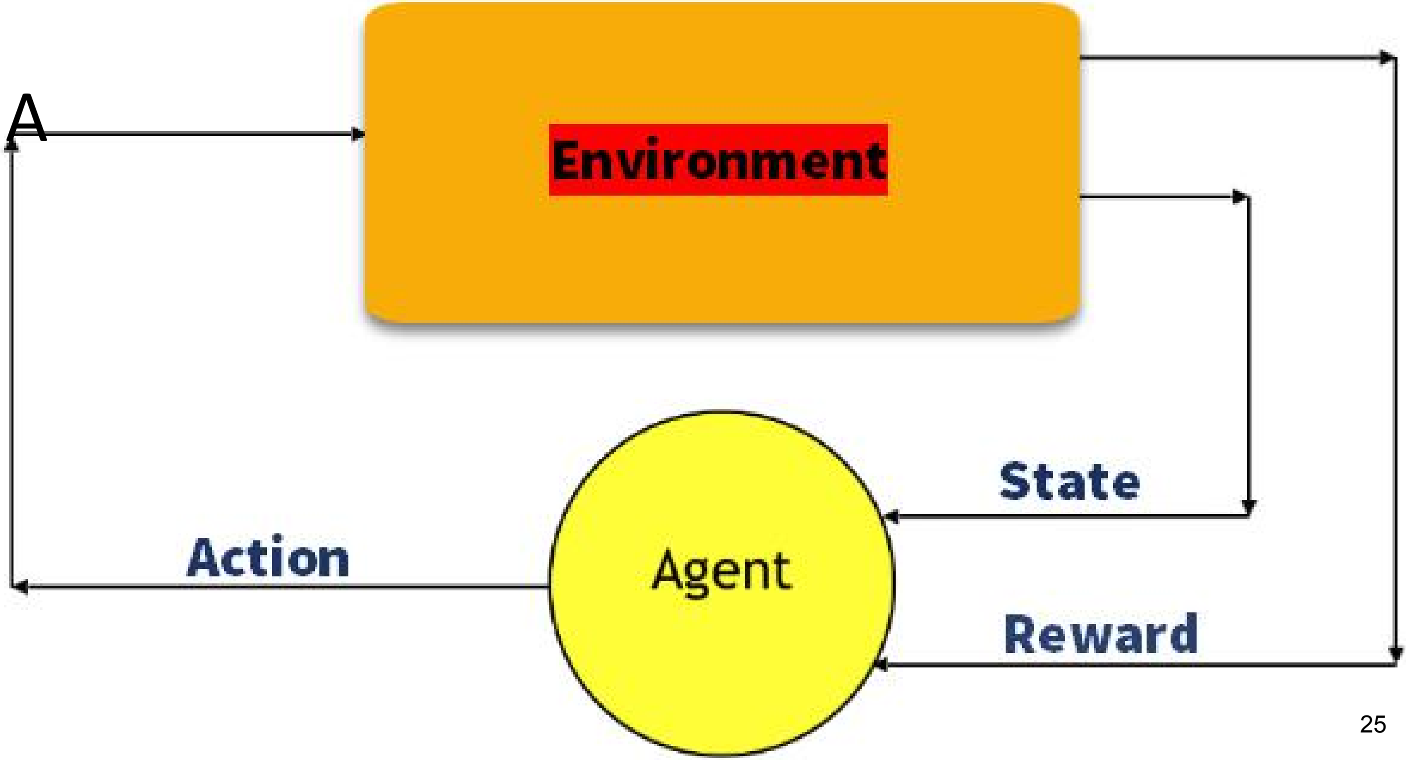
- Q-learning [A]
- Deep Q network [A + C]
- Markov Decision Process (MDP) = State-Action-Reward-State-Action (SARSA) [C]
- Deep Deterministic Policy Gradient (DDPG) [A+B]

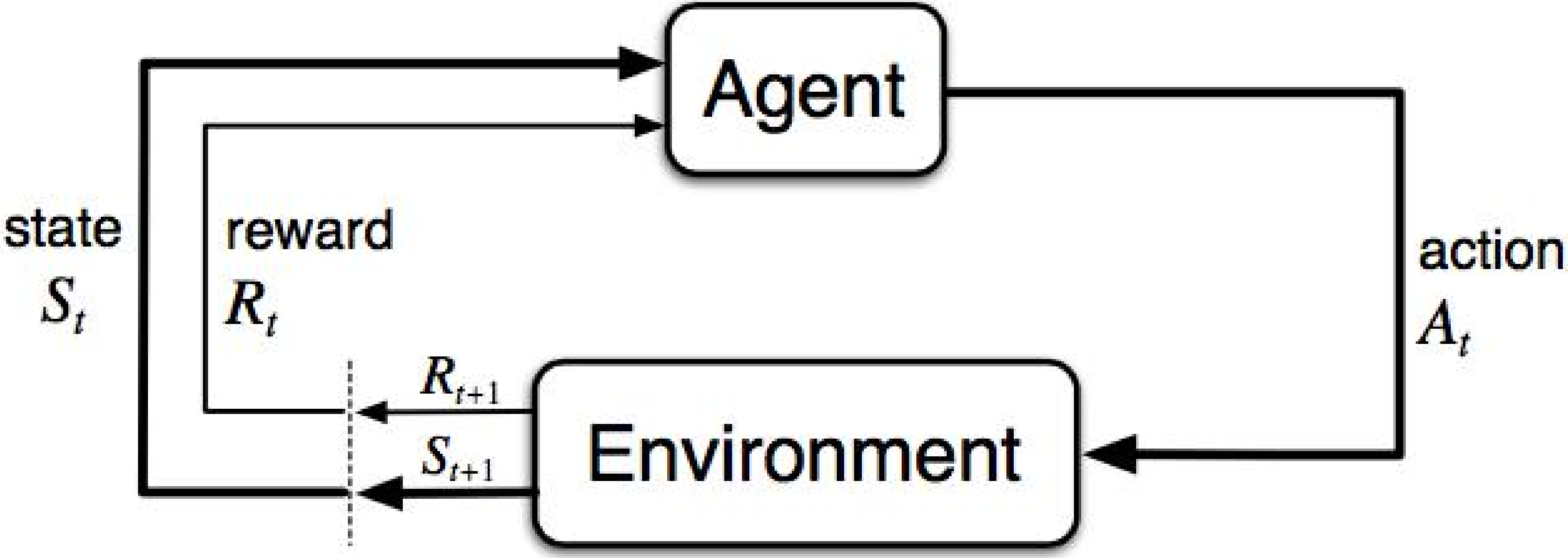

Reinforcement Learning Agents
REINFORCE
DQN
DDPG
TD3
A2C, A3C
PPO
SAC
Q-Learning
SARSA
Custom Agents

# Markov Decision Process

The following parameters are used to get a solution:

- Set of actions- A
- Set of states -S
- Reward- R
- Policy- n
- Value- V




Agent
state
$S_t$
reward
$R_t$
action
$A_t$
$R_{t+1}$
$S_{t+1}$
Environment

# Q-Learning

- Q learning is a **value-based** method of supplying information to inform which action an agent should take.
- Let’s understand this method by the following example:

There are five rooms in a building which are connected by doors.

Each room is numbered 0 to 4

The outside of the building can be one big outside area (5)

Doors number 1 and 4 lead into the building from room 5

# Q-Learning

Next, you need to **associate** a **reward** value to each door:

- Doors which lead **directly to the goal** have a reward of **100**
- Doors which is **not directly connected** to the target room gives **zero** reward
- As doors are two-way, and two arrows are assigned for each room
- Every arrow in the above image contains an instant reward value

# Q-Learning

an agent traverse from room number 2 to 5

Initial state = state 2

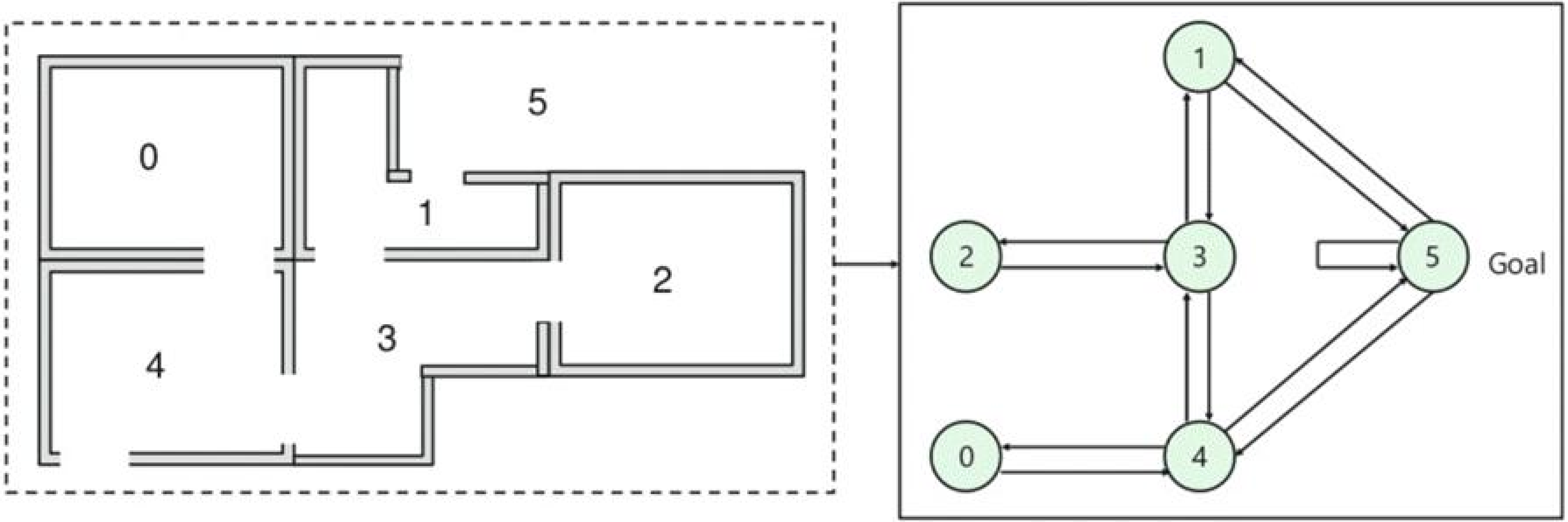
State 2-> state 3

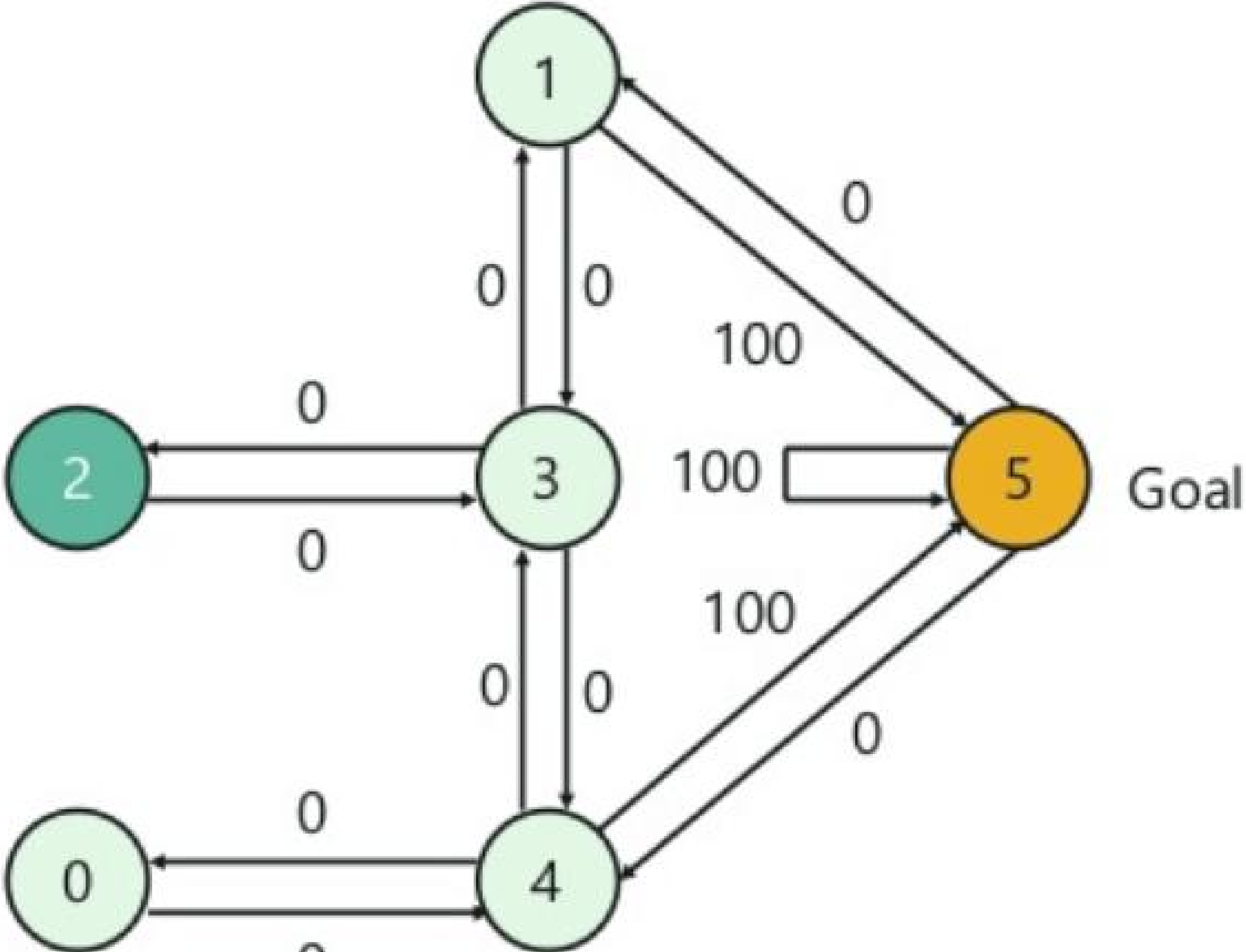
State 3 -> state (2,1,4)

State 4-> state (0,5,3)

State 1-> state (5,3)

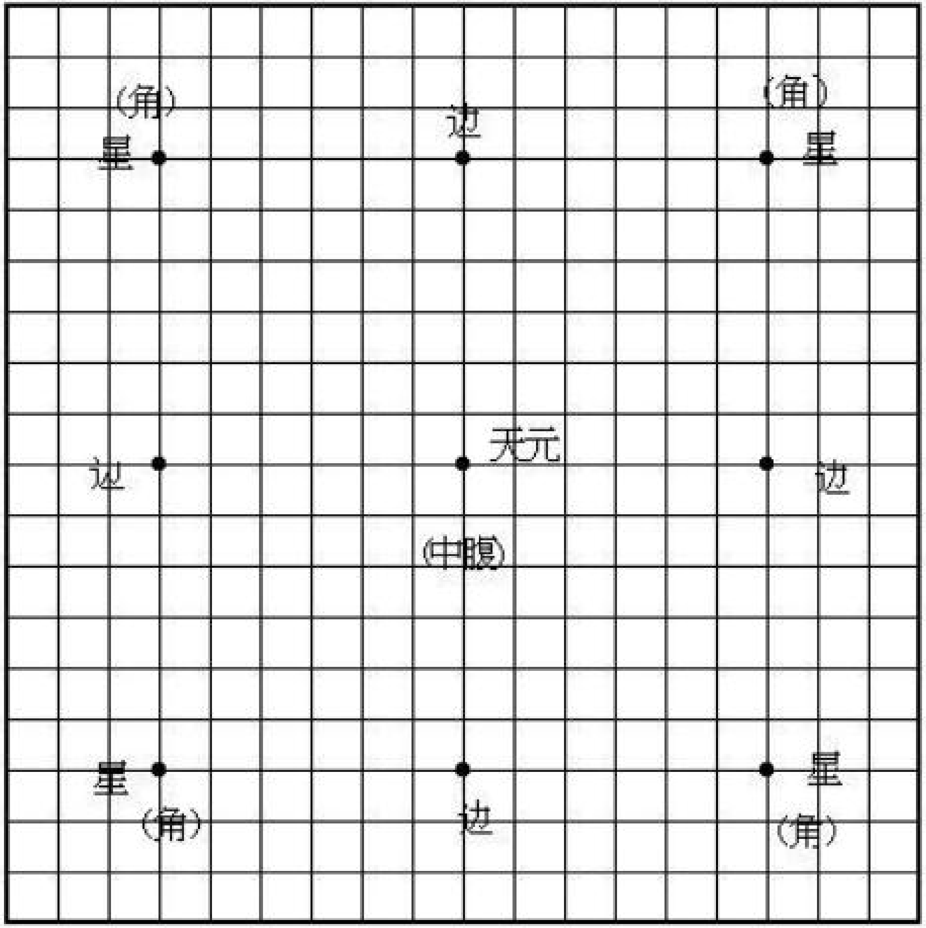
State 0-> state 4

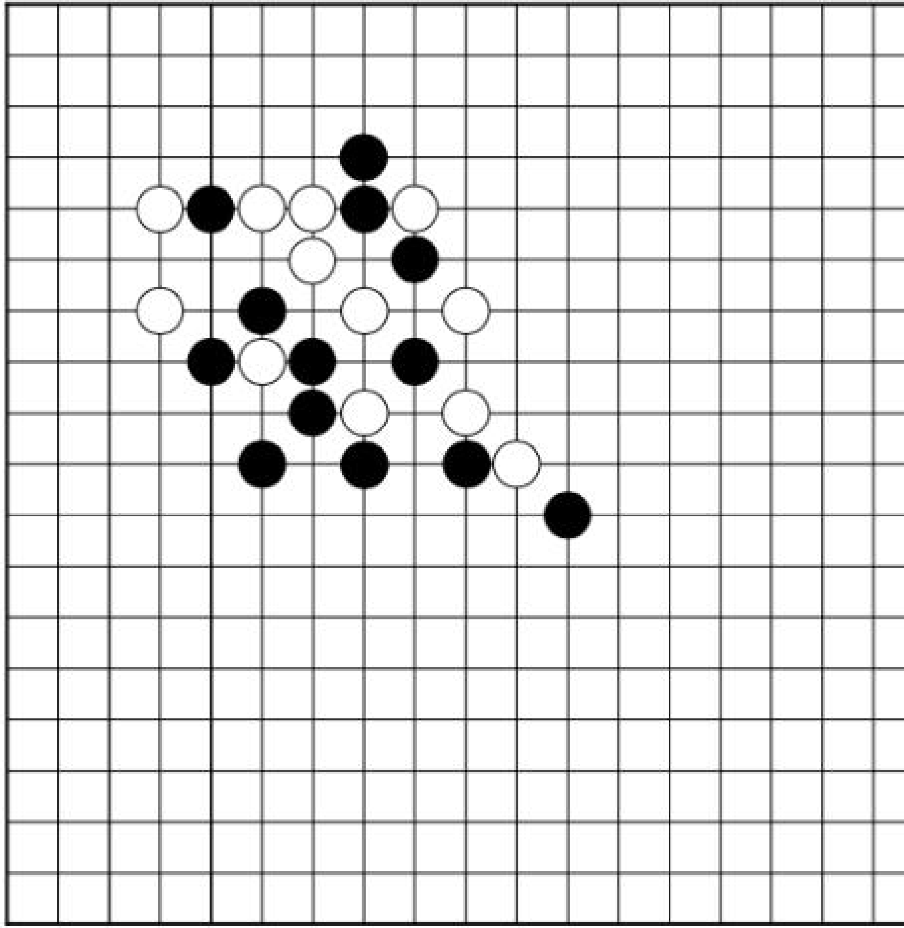

5
0
1
2
3
4
1
2
3
5
Goal
0
4


1
0
0
0
100
0
2
3
100
5
Goal
0
100
0
0
0
0
0
4

# 4 Applications

- Autonomous driving
- Robotics for industrial automation.
- Business strategy planning
- Personalized Recommendations
- Training systems
- Aircraft control and robot motion control
- Bidding and Advertising


(角)
(角)
边
星
星
边
天元
边
(中腹)
星
星
(角)
边
(角)



## Some important terms

- Agent: It is an assumed entity which performs actions in an environment to gain some reward.
- Environment (e): A scenario that an agent has to face.
- Reward (R): An immediate return given to an agent when he or she performs specific action or task.
- State (s): State refers to the current situation returned by the environment.
- Policy (π): It is a strategy which applies by the agent to decide the next action based on the current state.

## Some important terms

- Value (V): It is expected long-term return with discount, as compared to the short-term reward.
- Value Function: It specifies the value of a state that is the total amount of reward. It is an agent which should be expected beginning from that state.
- Model of the environment: This mimics the behavior of the environment. It helps you to make inferences to be made and also determine how the environment will behave.

# Some important terms

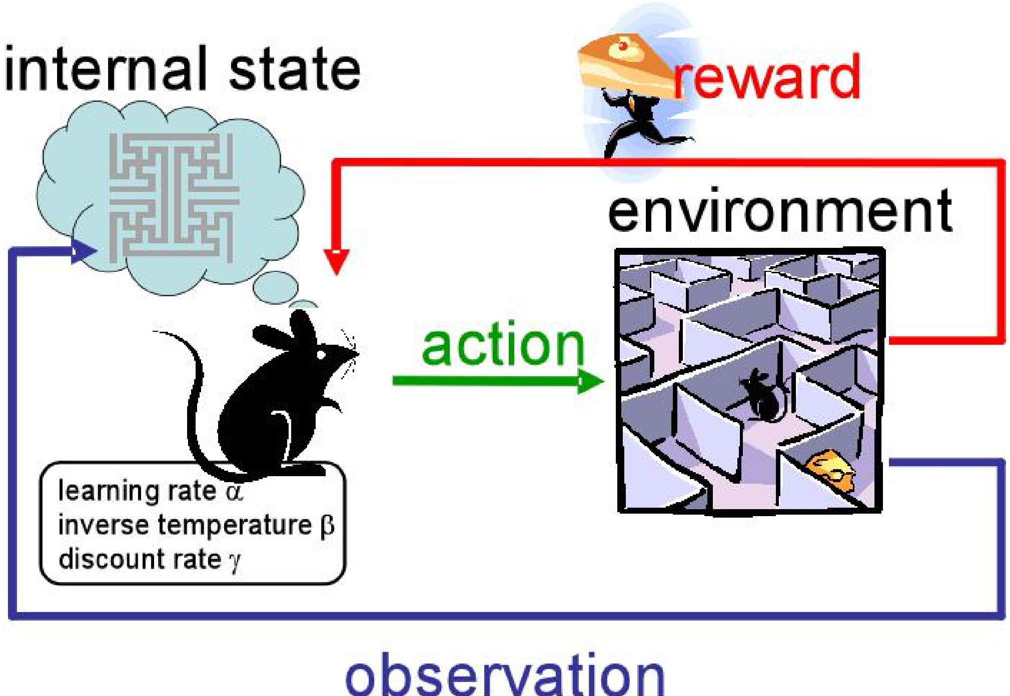
- Model based methods: It is a method for solving reinforcement learning problems which use model-based methods.
- Q value or action value (Q): Q value is quite similar to value. The only difference between the two is that it takes an additional parameter as a current action.

# Reinforcement Learning vs. Supervised Learning

| Parameters | Reinforcement Learning | Supervised Learning |
|---|---|---|
| Decision style | reinforcement learning helps you to take your decisions sequentially. | In this method, a decision is made on the input given at the beginning. |
| Works on | Works on interacting with the environment. | Works on examples or given sample data. |
| Dependency on decision | In RL method learning decision is dependent. Therefore, you should give labels to all the dependent decisions. | Supervised learning the decisions which are independent of each other, so labels are given for every decision. |
| Best suited | Supports and work better in AI, where human interaction is prevalent. | It is mostly operated with an interactive software system or applications. |
| Example | Chess game | Object recognition |

# Videos

- 01 Reinforcement Learning - Crash Course AI (11:27)
- 02 Reinforcement Learning for Engineers
  - ✓ Part 1 (14:10)
  - ✓ Part 2 (13:30)
  - ✓ Part 3 (17:50)
  - ✓ Part 4 (14:50)
  - ✓ Part 5 (16:07)



# Tutorial

## Reinforcement Learning Onramp

- https://matlabacademy.mathworks.com/details/reinforcement-learning-onramp/reinforcementlearning
- https://matlabacademy.mathworks.com/

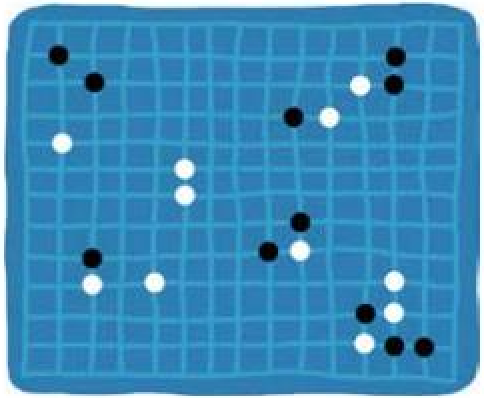
# Reference

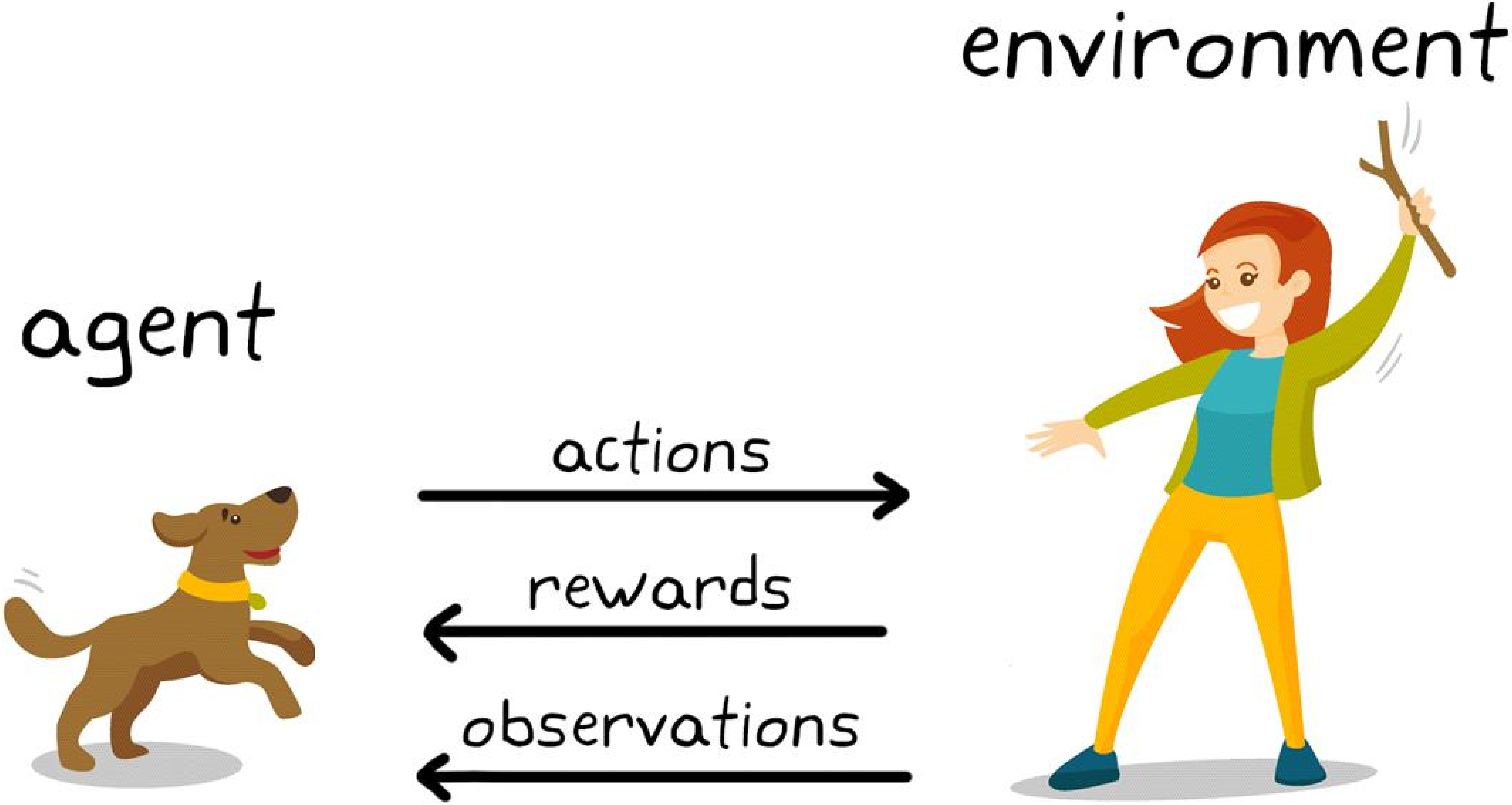
[1] Reinforcement Learning with MATLAB and Simulink

https://www.mathworks.com/campaigns/offers/reinforcement-learning-with-matlab-ebook.html

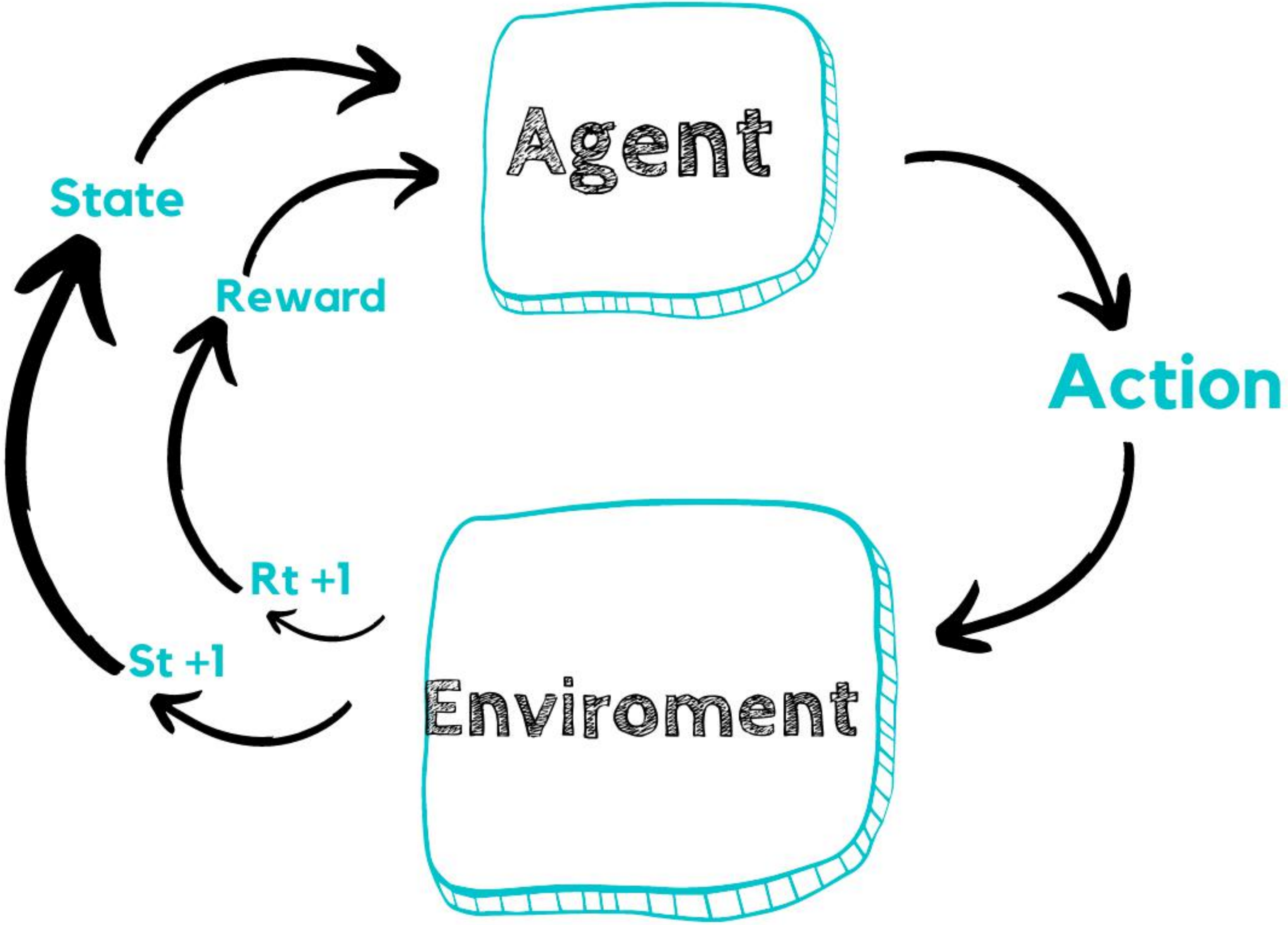
[2]What Is Reinforcement Learning?

https://www.mathworks.com/discovery/reinforceme





environment
agent
actions
rewards
observations



Agent
State
Reward
Action
Rt +1
St +1
Enviroment

# Introduction to Artificial Intelligence
# - 05-01-03 Reinforcement Learning

**Dr Leo Chen**
leo.chen@ieee.org
06/Sep/2021

**Thanks and Questions**