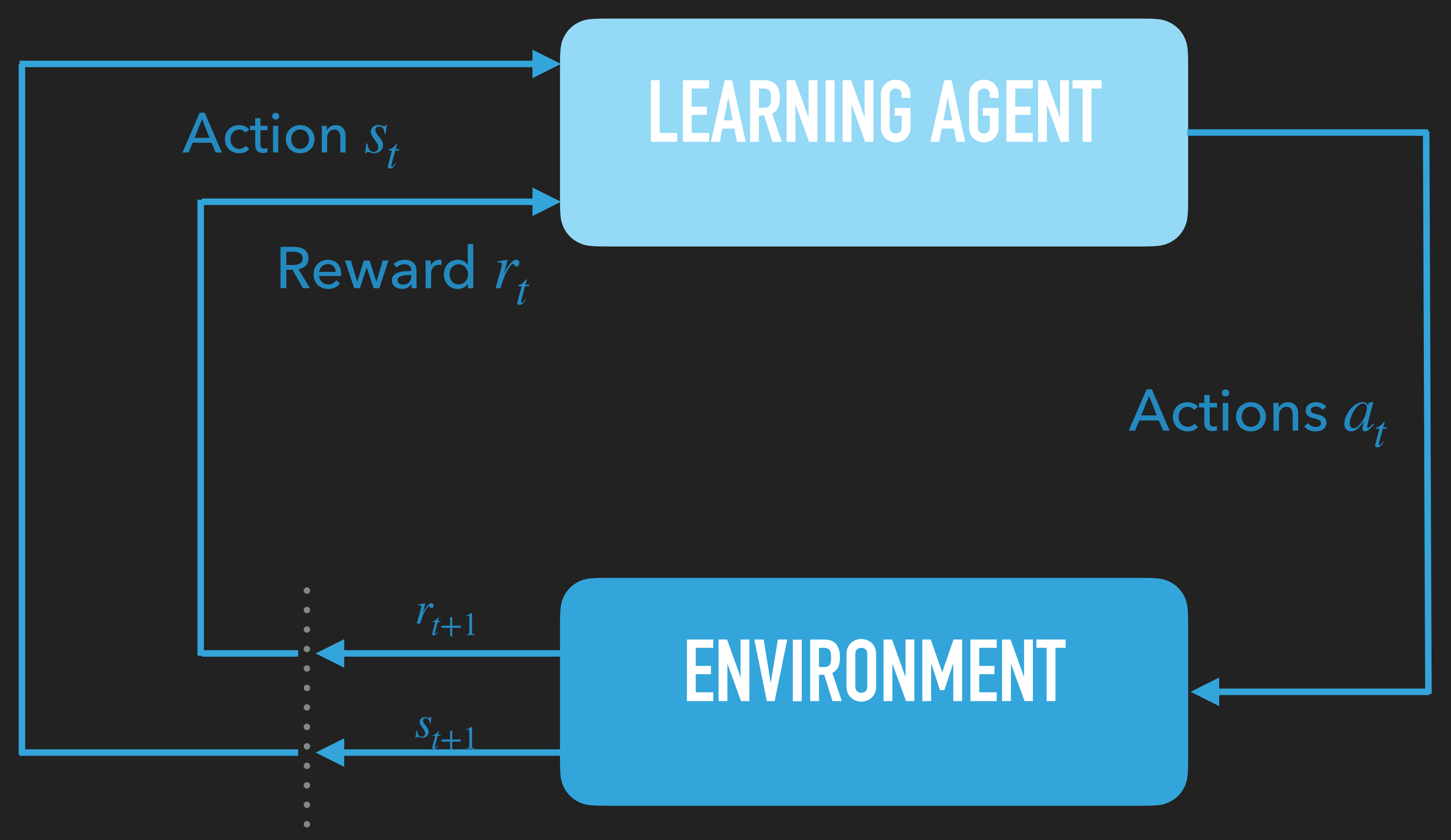
SOPHIA YAZZOURH - STUDENT SEMINAR 06/06/24

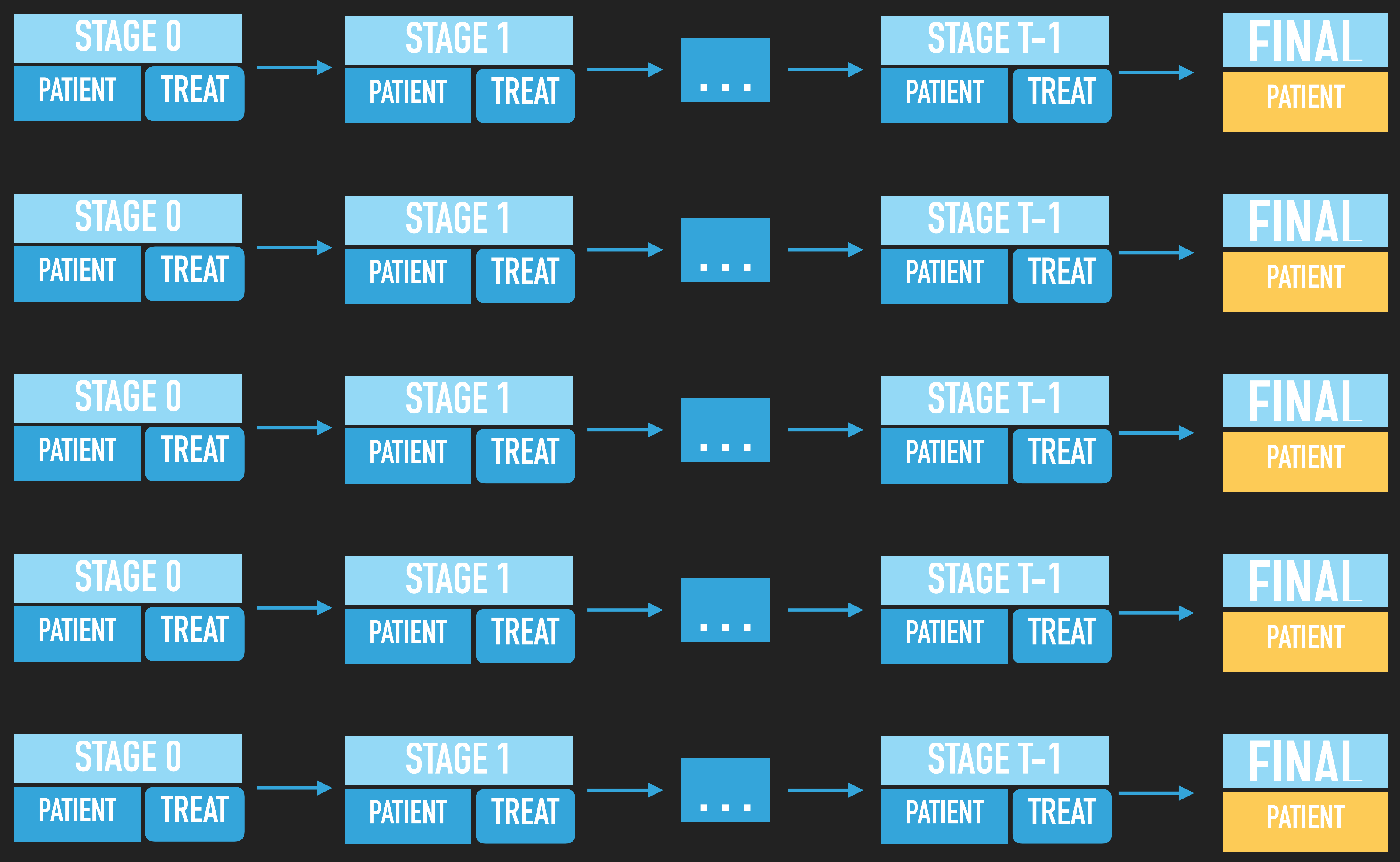
INTRODUCTION TO REINFORCEMENT LEARNING

QUID OF REINFORCEMENT LEARNING?

- The nature of learning is basically interacting with the environment
- Use the environment response to our actions to take the next decision
- How to map situations to actions?
- Solve decision-making problems

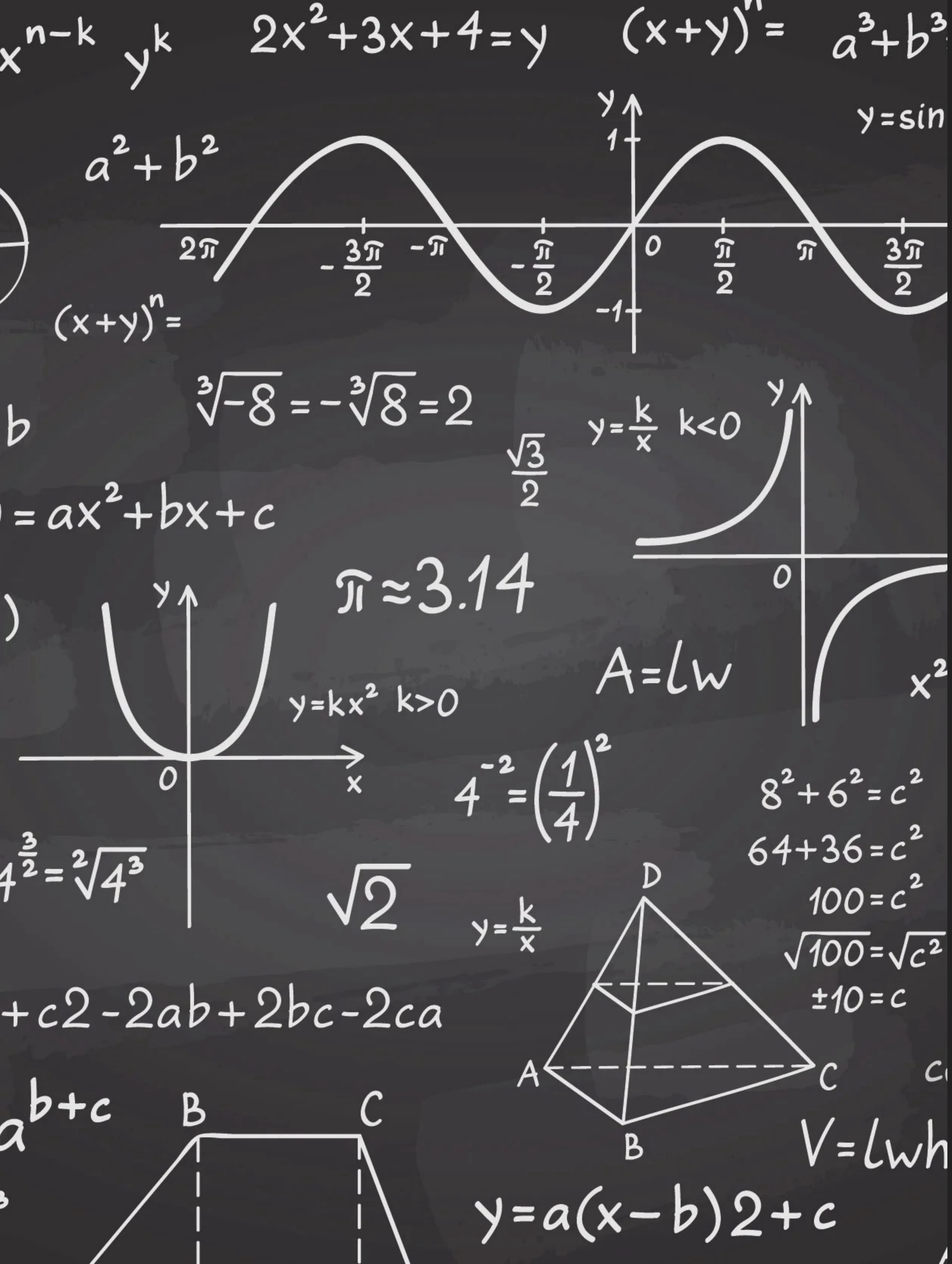
ONLINE VS OFFLINE





PLAN

- I. Mathematical framework
- II. Algorithms
- III. Properties
- IV. Researches issues



INTRODUCTION TO RL

MATHEMATICAL FRAMEWORK

DECISION PROCESS $(\mathbb{S}, \mathbb{A}, \{\mathbb{A}(s) \mid s \in \mathbb{S}\}, \nu)$ ON $\mathbb{T}$

- A family of random variables $\{S_t, t \in \mathbb{T}\}$ in $\mathbb{S}$ called space of states
- A family of random variable $\{A_t, t \in \mathbb{T}\}$ in $\mathbb{A}$ called space of actions
- A set $\{\mathbb{A}(s) \mid s \in \mathbb{S}\}$of non empty measurable subsets of . $\mathbb{A}(s)$ is the set of realizable actions when the system is in the state $s \in \mathbb{S}$. We will ask to be a measurable subset of $\mathbb{S} \times \mathbb{A}$.
- An initial probability law ν on $\mathbb{S}$.

TRAJECTORY / HISTORY

- An admissible trajectory h_n at time n is a vector containing the states visited by the system and the actions taken is described by $(s_0, a_0, \ldots, s_{n-1}, a_{n-1}, s_n)$

MARKOV DECISION PROCESS

- The point of main importance to deal with decision process is $\mathbb{P}_\nu[S_{n+1} = s_{n+1} \,|\, H_n = h_n, A_n = a_n]$
- Demands significant computational resources because of the increasing length of the vector h_n as n increases
- The Markov Assumption : $\mathbb{P}_\nu[S_{n+1} = s_{n+1} \,|\, H_n = h_n, A_n = a_n] = \mathbb{P}_\nu[S_{n+1} = s_{n+1} \,|\, S_n = s_n, A_n = a_n]$

POLICY

- A policy is a sequence $\pi = (\pi_n)_{n\in\mathbb{N}}$ of conditional distributions from $\mathbb{A}$ given $\mathbb{H}_n$ defined, for any $\mathscr{A} \in \mathscr{B}(\mathbb{A})$ and all $h_n \in \mathbb{H}_n$, by :

$$\pi_n(\mathscr{A}, h_n) = \mathbb{P}[A_n \in \mathscr{A} \,|\, H_n = h_n]$$

- Plan that establishes a sequence of actions
- Tailored to align with a specified objective.

REWARD

- Reward is defined as a family of bounded $\mathbb{R}$-valued random variables $\{R_n, n \in \mathbb{N}\}$. For a sake of simplicity, let us denote for a given $n \in \mathbb{N}$, for all $h_n \in \mathbb{H}_n$, all $a_n \in \mathbb{A}$and all $s_{n+1} \in \mathbb{S}$:
$\mathscr{R}_{n+1}(h_n, a_n, s_{n+1}) = \mathbb{E}_\nu^\pi[R_{n+1} \,|\, H_n = h_n, A_n = a_n, S_{n+1} = s_{n+1}]$
- Given $\gamma \in [0,1]$ a discount parameter, the stage n long term discounted reward function is defined for all $n \in \mathbb{N}$, by: $G_n = \sum_{j=n+1}^{\infty} \gamma^{j-n-1} R_j$

VALUE FUNCTIONS AND OPTIMALITY (MDP)

- Given $(S, A, \{\mathbb{A}(s) \mid s \in \mathbb{S}\}, \nu)$ a decision process on $\mathbb{T}$, $\{R_n, n \in \mathbb{N}\}$a family of rewards, π a policy and $\gamma \in [0,1]$ a discount parameter.
  - The V-function for a state s_n is given by: $V_n^\pi(s_n) = \mathbb{E}_\nu^\pi[G_n \mid S_n = s_n]$
  - The Q-function for a history s_n, taking a_n is given by: $Q_n^\pi(s_n, a_n) = \mathbb{E}_\nu^\pi[G_n \mid S_n = s_n, A_n = a_n]$
- Optimal policy : $V_n^*(s_n) = \max_\pi \; V_n^\pi(s_n)$ and $Q_n^*(s_n, a_n) = \max_\pi \; Q_n^\pi(s_n, a_n)$

INTRODUCTION TO RL

ALGORITHMS

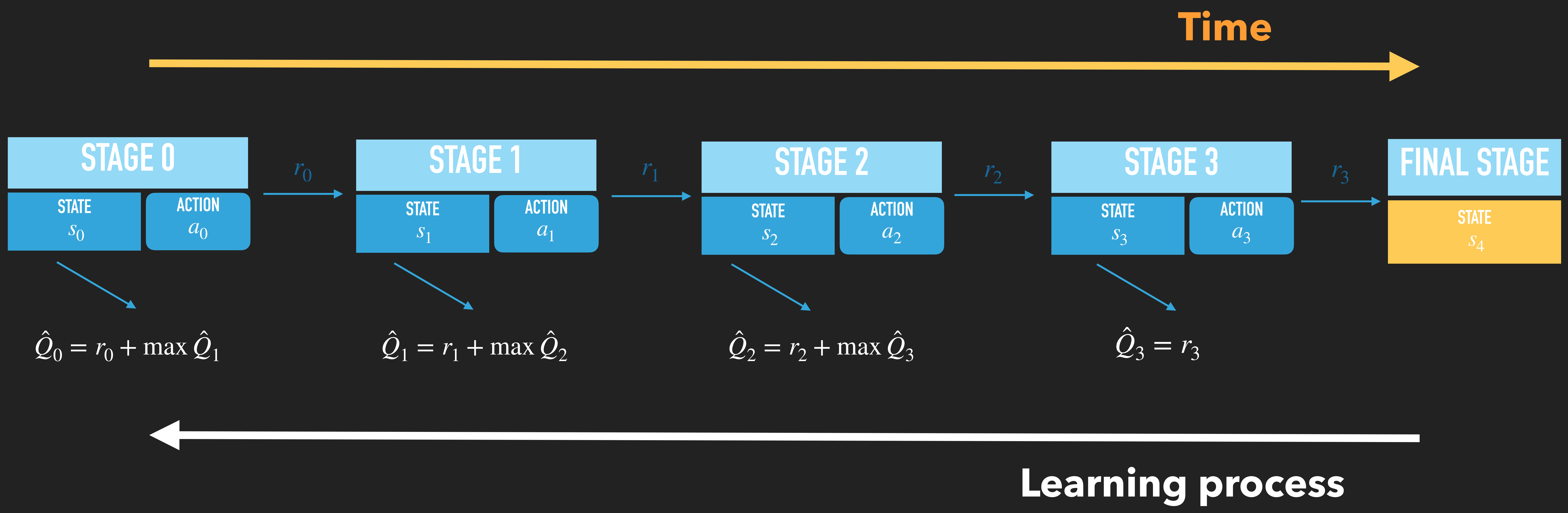
Q-LEARNING (ONLINE ENVIRONMENT)

- **Initialization** : Arbitrary
- **Exploration/Exploitation** :

$$\pi_\epsilon(s) = \begin{cases} \text{random action from } \mathbb{A}(s) & \text{with probability } \epsilon \\ \arg\max_{a \in \mathbb{A}(s)} Q(s,a) & \text{with probability } 1-\epsilon \end{cases}$$

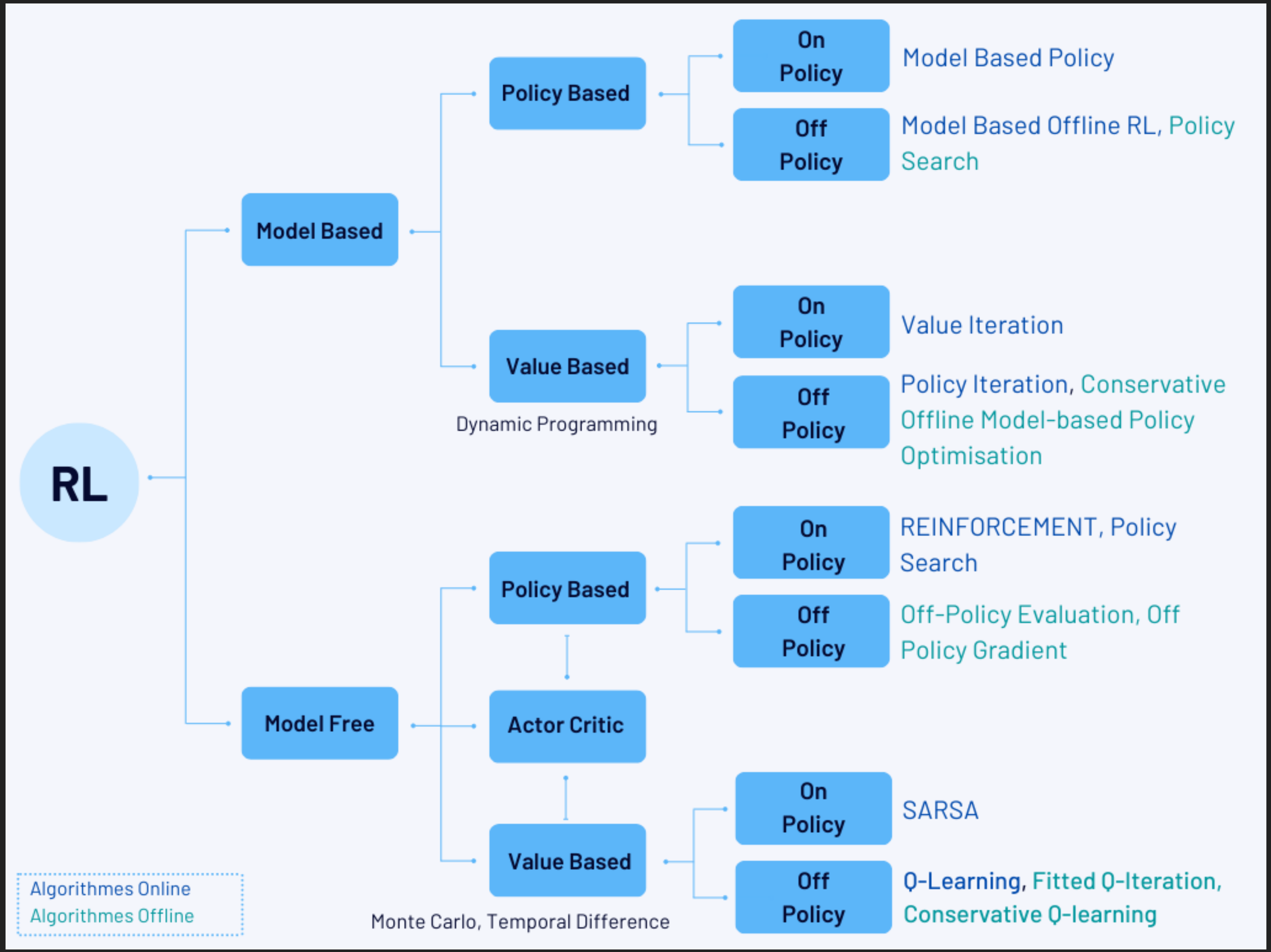
- **Update Q-values** : $Q(s,a) \leftarrow Q(s,a) + \alpha \left(r + \gamma \max_{a'} Q(s',a') - Q(s,a) \right)$
- **Optimal Policy** : $Q_n^*(s_n, a_n) = \max_{\pi} \; Q_n^\pi(s_n, a_n)$

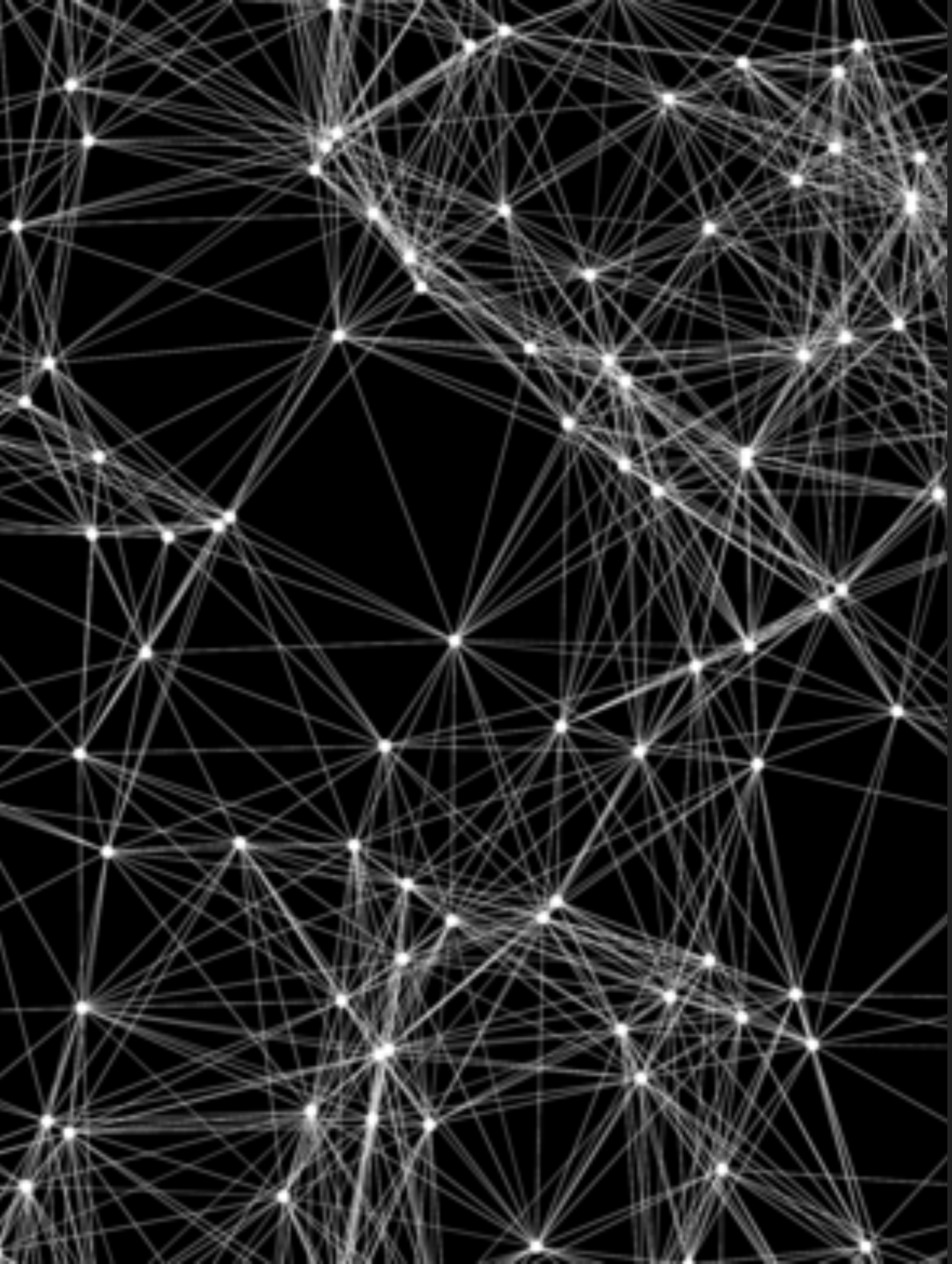
BACKWARD Q-LEARNING (OFFLINE ENVIRONMENT)



- Estimation of Q-values using regression algorithms in a backward manner at each step
- Determination of an optimal strategy at each step

ALGORITHMS (NON-EXHAUSTIVE LIST)





INTRODUCTION TO RL

PROPERTIES

MODEL-BASED VS. MODEL-FREE

$$\mathbb{P}_\nu[S_{n+1} = s_{n+1} \,|\, H_n = h_n, A_n = a_n]$$

- A procedure is considered "model-based" when it relies on knowledge of all transition probabilities from a model

- A model-free method is able to bypass this model and is based on partial information of the associations between states and actions

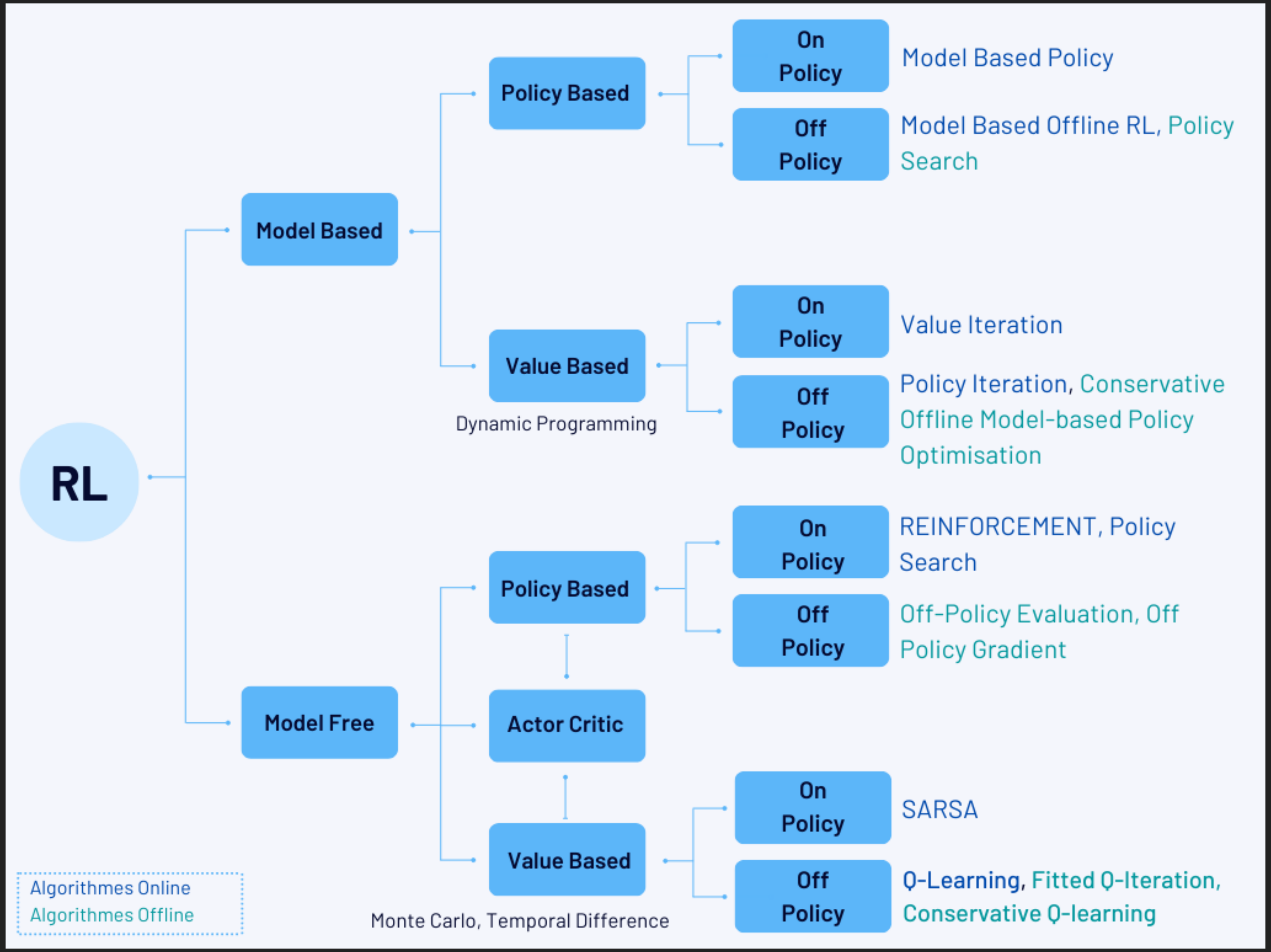
POLICY-BASED VS. VALUE-BASED

- **Direct Computation** :
  Optimization problem directly solve
- **Policy Parametrization** :
  $\theta \in \Theta, \;\; \pi_\theta(s, a)$
- **Objective function** :
  $\theta_{t+1} = \theta_t + \nabla \mathbb{E}^\pi[G_t \,|\, \theta]$

- **Intermediate Element** :
  State-value and Action-value
- **Value functions** :
  $V_n^\pi(s_n) = \mathbb{E}_\nu^\pi[G_n \,|\, S_n = s_n]$
  $Q_n^\pi(s_n, a_n) = \mathbb{E}_\nu^\pi[G_n \,|\, S_n = s_n, A_n = a_n]$
- **Optimality** :
  $V_n^*(s_n) = \max_\pi \;\; V_n^\pi(s_n)$
  $Q_n^*(s_n, a_n) = \max_\pi \;\; Q_n^\pi(s_n, a_n)$

ON-POLICY VS. OFF-POLICY

ALGORITHMS (NON-EXHAUSTIVE LIST)



INTRODUCTION TO RL

RESEARCH ISSUES

« HOT » TOPICS IN REINFORCEMENT LEARNING

- Sample Efficiency for online setting
- Deep Reinforcement Learning
- Multi Agent Reinforcement Learning
- Explicable Reinforcement Learning
- …

REINFORCEMENT LEARNING FOR HEALTHCARE APPLICATIONS

- Reward formulation
- Integration of Prior Knowledge
- Learning from small data
- Futur in Vivo Studies
- …