
INTRODUCTION TO REINFORCEMENT LEARNING

BY: DIMPLE BOHRA



CONTENTS

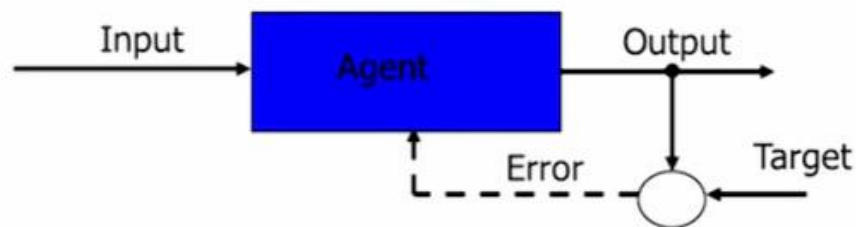
- Reinforcement Learning (RL)
- Elements of RL
- Approaches for solving RL
 1. Value based
 2. Policy based
 3. Model based
- Exploration-Exploitation Dilemma
- Evolutionary Methods
- Immediate RL



What is Reinforcement Learning?

- Learning about stimuli and actions based on rewards and punishments alone.
- No detailed supervision available
- Trial-and-error learning
- Delayed rewards
- Sequence of actions required to obtain reward
- Associative learning required
 - Need to associate actions to states
- Learn about policies not just actions
- Typically in a stochastic world

RL IS NOT SUPERVISED LEARNING



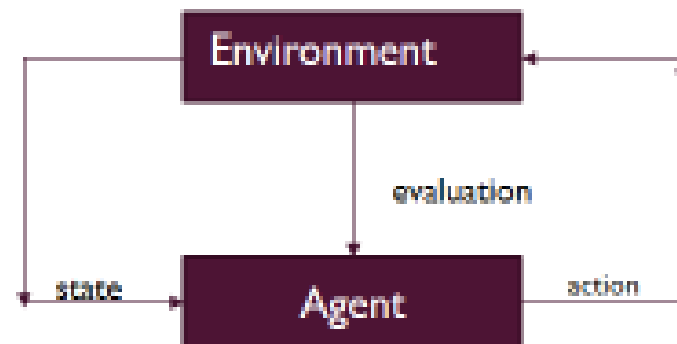
- Very sparse “supervision”
- No target output provided
- No error gradient information available
- Action chooses next state
- Explore to estimate gradient – Trail and error learning

RL IS NOT UNSUPERVISED LEARNING



- Sparse “supervision” available
- Pattern detection not primary goal

RL FRAMEWORK



- Learn from close interaction
- Stochastic environment
- Delayed scalar evaluation
- Maximize a measure of long term performance

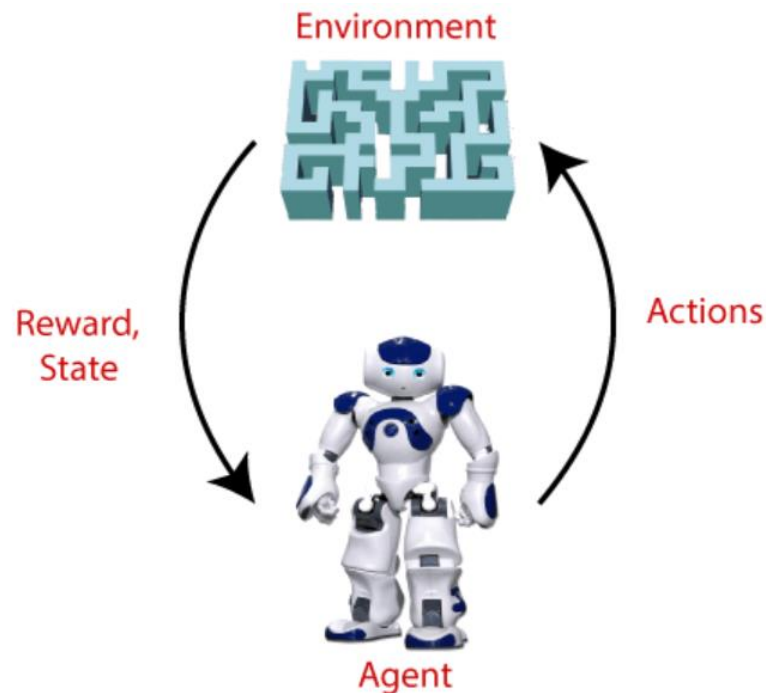
REINFORCEMENT LEARNING: BASIC IDEA

- Learn to take correct actions over time by experience
- Similar to how humans learn: “trial and error”
- Try an action –
 - “see” what happens
 - “remember” what happens
 - Use this to change choice of action next time you are in the same situation
 - “Converge” to learning correct actions
- Focus on long term return, not immediate benefit
 - Some action may seem not beneficial for short term, but it’s long term return will be good.

Example: Movie: *Cast Away* , actor :Tom Hanks

REINFORCEMENT LEARNING

Intelligent agent (computer program) interacts with the environment and learns to act within that.



Counter Strike Example



1. The RL Agent (Player1) collects state S^0 from the environment
2. Based on the state S^0 , the RL agent takes an action A^0 , initially the action is random
3. The environment is now in a new state S^1
4. RL agent now gets a reward R^1 from the environment
5. The RL loop goes on until the RL agent is dead or reaches the destination

TERMS IN RL



Agent: The RL algorithm that learns from trial and error

Environment: The world through which the agent moves



Action (A): All the possible steps that the agent can take

State (S): Current condition returned by the environment



TERMS IN RL



Reward (R): An instant return from the environment to appraise the last action



Policy (π): The approach that the agent uses to determine the next action based on the current state



Value (V): The expected long-term return with discount, as opposed to the short-term reward R

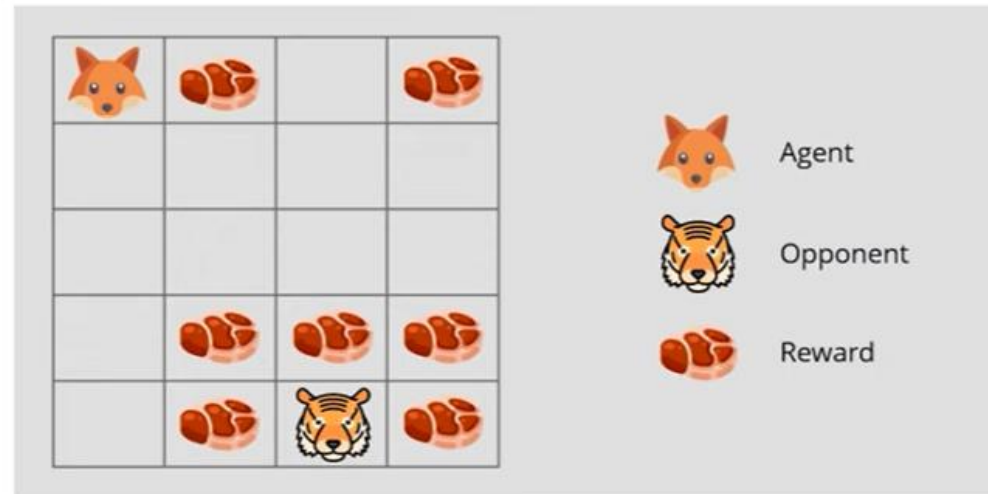


Action-value (Q): This similar to Value, except, it takes an extra parameter, the current action (A)

REWARD MAXIMIZATION

Reward Maximization

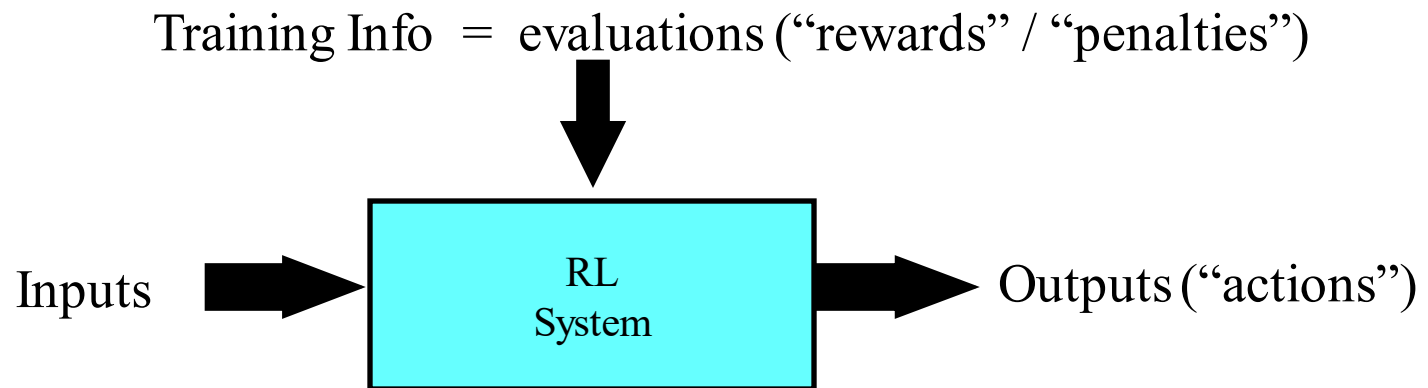
Reward maximization theory states that, *a RL agent must be trained in such a way that, he takes the best action so that the reward is maximum.*



KEY FEATURES OF RL

- Learner is not told which actions to take
- Trial-and-Error search
- Possibility of delayed reward
 - Sacrifice short-term gains for greater long-term gains
- There is need to ***explore*** and ***exploit***
- Considers the whole problem of a goal-directed agent interacting with an uncertain environment

HOW REINFORCEMENT LEARNING WORKS



Objective: get as much reward as possible

- Absolutely no “already existing training data”
- Agent learns ONLY by experience

EXAMPLES OF RL

- A master chess player makes a move. The choice is informed both by planning—anticipating possible replies and counterreplies—and by immediate, intuitive judgments of the desirability of positions and moves.
- A gazelle calf struggles to its feet minutes after being born. Half an hour later it is running at 20 miles per hour.
- A mobile robot decides whether it should enter a new room in search of more trash to collect or start trying to find its way back to its battery recharging station. It makes its decision based on the current charge level of its battery and how quickly and easily it has been able to find the recharger in the past.

ELEMENTS OF RL

- Beyond the agent and the environment, one can identify four main sub elements of a reinforcement learning system:
 1. **A policy** defines the learning agent's way of behaving at a given time.
 2. **A reward signal** defines the goal in a reinforcement learning problem (in immediate sense)
 3. **A value function** is the total amount of reward an agent can expect to accumulate over the future, starting from that state (in long run)
 4. **A model of the environment** mimics the behavior of the environment, or more generally, that allows inferences to be made about how the environment will behave

ELEMENTS OF RL

- **Reward:** A reward is a scalar feedback signal it indicates how well the agent is doing at step t . The agent's sole objective is to maximize the total reward it receives over the long run.
 - The reward signal is the primary basis for altering the policy.
 - $R(s)$ indicates the reward for simply being in the state S .
 - $R(S,a)$ indicates the reward for being in a state S and taking an action a .
 - $R(S, a, S')$ indicates the reward for being in a state S , taking an action a and ending up in a state S' .

POLICY

- It maps the perceived states of the environment to the actions taken on those states.
- A policy is the core element of the RL as it alone can define the behavior of the agent. In some cases, it may be a simple function or a lookup table, whereas, for other cases, it may involve general computation as a search process.
- It could be deterministic or a stochastic policy

POLICY

Deterministic policy

It is a mapping $\pi: \mathcal{S} \rightarrow \mathcal{A}$. For each state $s \in \mathcal{S}$, it yields the action $a \in \mathcal{A}$ that the agent will choose while in state s .

Stochastic policy

If an agent follows policy π at time t , then $\pi(a|s)$ is the probability that $A_t = a$ if $S_t = s$. This means that, at time t , under policy π , the probability of taking action a in state s is $\pi(a|s)$.

Note that, for each state $s \in \mathcal{S}$, π is a probability distribution over $a \in \mathcal{A}(s)$.

POLICY: EXAMPLE

- For example, imagine a world where a robot moves across the room and the task is to get to the target point (x, y) , where it gets a reward. Here:
- A room is an *environment*
- Robot's current position is a *state*
- A *policy* is what an agent does to accomplish this task:
 - dumb robots just wander around randomly until they accidentally end up in the right place (policy #1)
 - others may, for some reason, learn to go along the walls most of the route (policy #2)
 - smart robots plan the route in their "head" and go straight to the goal (policy #3)

VALUE FUNCTION

- Value functions are functions of states, or of state-action pairs, that estimate how good it is for an agent to be in a given state, or how good it is for the agent to perform a given action in a given state.
- This notion of *how good* a state or state-action pair is given in terms of expected return.
- The rewards an agent expects to receive are dependent on what actions the agent takes in given states. So, value functions are defined with respect to specific ways of acting.
- Since the way an agent acts is influenced by the policy it's following, then we can see that value functions are defined with respect to policies.

STATE VALUE FUNCTION AND ACTION VALUE FUNCTION


Definition

We call v_π the **state-value function** for policy π .

The value of state s under a policy π is

$$v_\pi(s) = \mathbb{E}_\pi [G_t \mid S_t = s]$$

For each state s ,
it yields the expected return
if the agent starts in state s
and then uses the policy
to choose its actions for all time steps.




Definition

We call q_π the **action-value function** for policy π .

The value of taking action a in state s under a policy π is

$$q_\pi(s, a) = \mathbb{E}_\pi [G_t \mid S_t = s, A_t = a]$$

For each state s and action a
it yields the expected return
if the agent starts in state s
then chooses action a
and then uses the policy
to choose its actions for all time steps.



STATE VALUE FUNCTION

The *state-value function* for policy π , denoted as v_π , tells us how good any given state is for an agent following policy π . In other words, it gives us *the value of a state* under π .

Formally, the value of state s under policy π is the expected return from starting from state s at time t and following policy π thereafter. Mathematically we define $v_\pi(s)$ as

$$v_\pi(s) = E_\pi[G_t \mid S_t = s]$$

ACTION VALUE FUNCTION

Similarly, the *action-value function* for policy π , denoted as q_π , tells us how good it is for the agent to take any given action from a given state while following policy π . In other words, it gives us *the value of an action* under π .

Formally, the value of action a in state s under policy π is the expected return from starting from state s at time t , taking action a , and following policy π thereafter. Mathematically, we define $q_\pi(s, a)$ as

$$q_\pi(s, a) = E_\pi [G_t \mid S_t = s, A_t = a]$$

Conventionally, the action-value function q_π is referred to as the *Q-function*, and the output from the function for any given state-action pair is called a *Q-value*. The letter “ Q ” is used to represent the *quality* of taking a given action in a given state.

MODEL

- In the context of reinforcement learning (RL), the model allows inferences to be made about the environment. For example, the model might predict the resultant next state and next reward, given a state and action.
- The model is used for planning, which means it provides a way to take a course of action by considering all future situations before experiencing those situations.
- Methods for solving reinforcement learning problems that use models and planning are called **model-based methods**, as opposed to simpler **model-free methods** that are explicitly trial-and-error learners.

APPROACHES FOR IMPLEMENTING RL

- There are three ways to implement RL:
 1. Value Based
 2. Policy Based
 3. Model Based

VALUE BASED RL

- The value-based approach is about to find the optimal value function, which is the maximum value at a state under any policy.
- Therefore, the agent expects the long-term return at any state(s) under policy π .

POLICY BASED RL

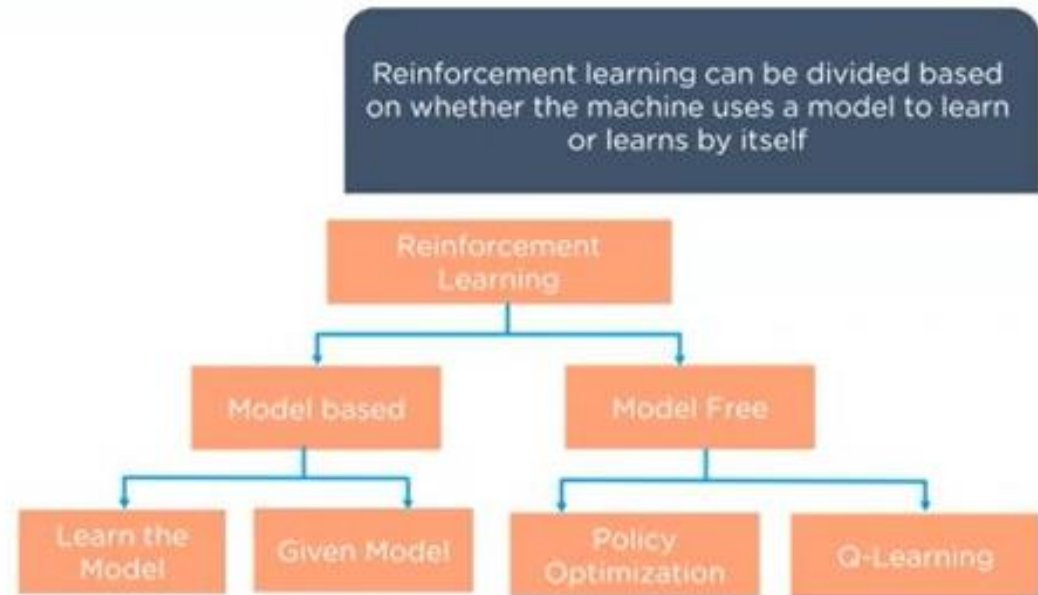
- Policy-based approach is to find the optimal policy for the maximum future rewards without using the value function.
- In this approach, the agent tries to apply such a policy that the action performed in each step helps to maximize the future reward.

The policy-based approach has mainly two types of policy:

- **Deterministic:** The same action is produced by the policy (π) at any state.
- **Stochastic:** In this policy, probability determines the produced action.

MODEL BASED RL

- In the model-based approach, a virtual model is created for the environment, and the agent explores that environment to learn it.



MODEL BASED VS MODEL FREE

- In the *model-based* approach, a system uses a predictive model of the world to ask questions of the form “what will happen if I do x ?” to choose the best x_1 .
- In the alternative *model-free* approach, the modeling step is bypassed altogether in favor of learning a control policy directly.
- According to Sutton & Barto, the distinction between model-free and model-based reinforcement learning algorithms is analogous to habitual and goal-directed control of learned behavioral patterns.

EXPLOITATION AND EXPLORATION IN RL

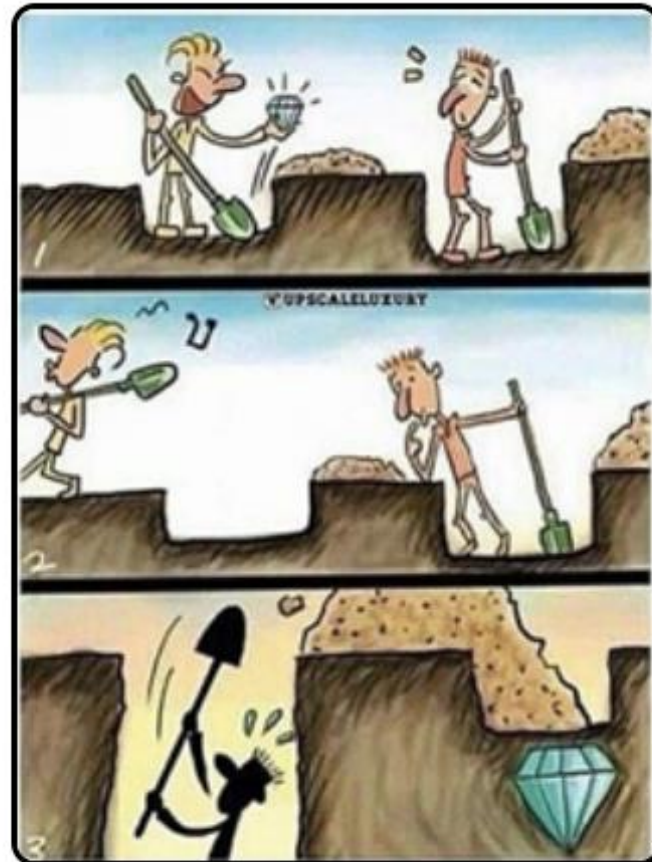
- Exploitation: keep on doing what you were doing
- Exploration: explore what is new
- Exploitation is defined as a greedy approach in which agents try to get more rewards by using estimated value but not the actual value. So, in this technique, *agents make the best decision based on current information.*
- Unlike exploitation, in exploration techniques, agents primarily focus on improving their knowledge about each action instead of getting more rewards so that they can get long-term benefits. So, in this technique, *agents work on gathering more information to make the best overall decision.*

EXPLORATION EXPLOITATION DILEMMA

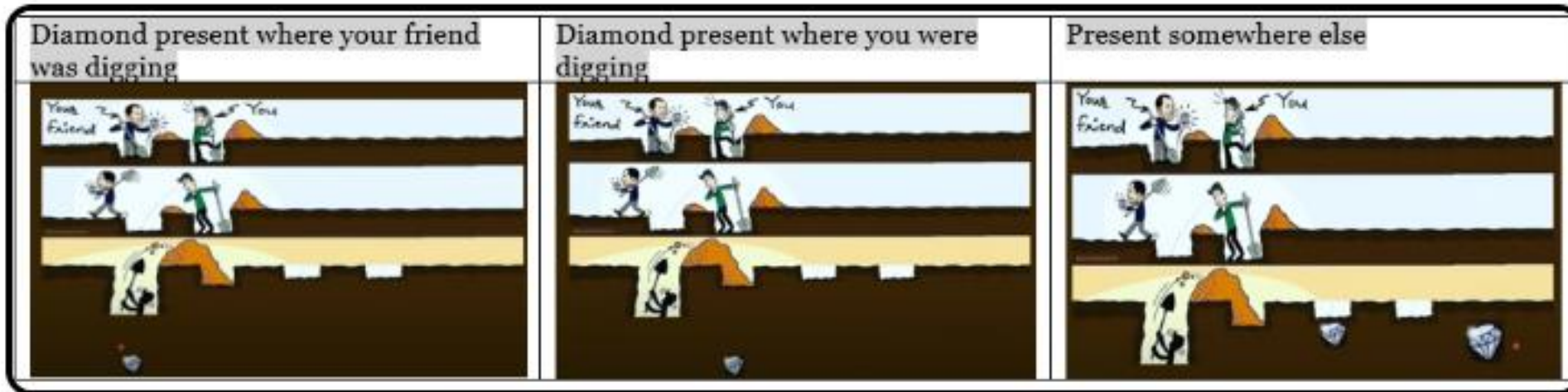
- The dilemma is between choosing what you know and getting something close to what you expect ('exploitation') and choosing something you aren't sure about and possibly learning more ('exploration').
- Reinforcement learning agent will be in a dilemma on whether to exploit the partial knowledge to receive some rewards or it should explore unknown actions which could result in many rewards.

EXAMPLE

Let's say your friend and you are digging in the hope that both will get a diamond out of it.



EXPLORATION EXPLOITATION DILEMMA

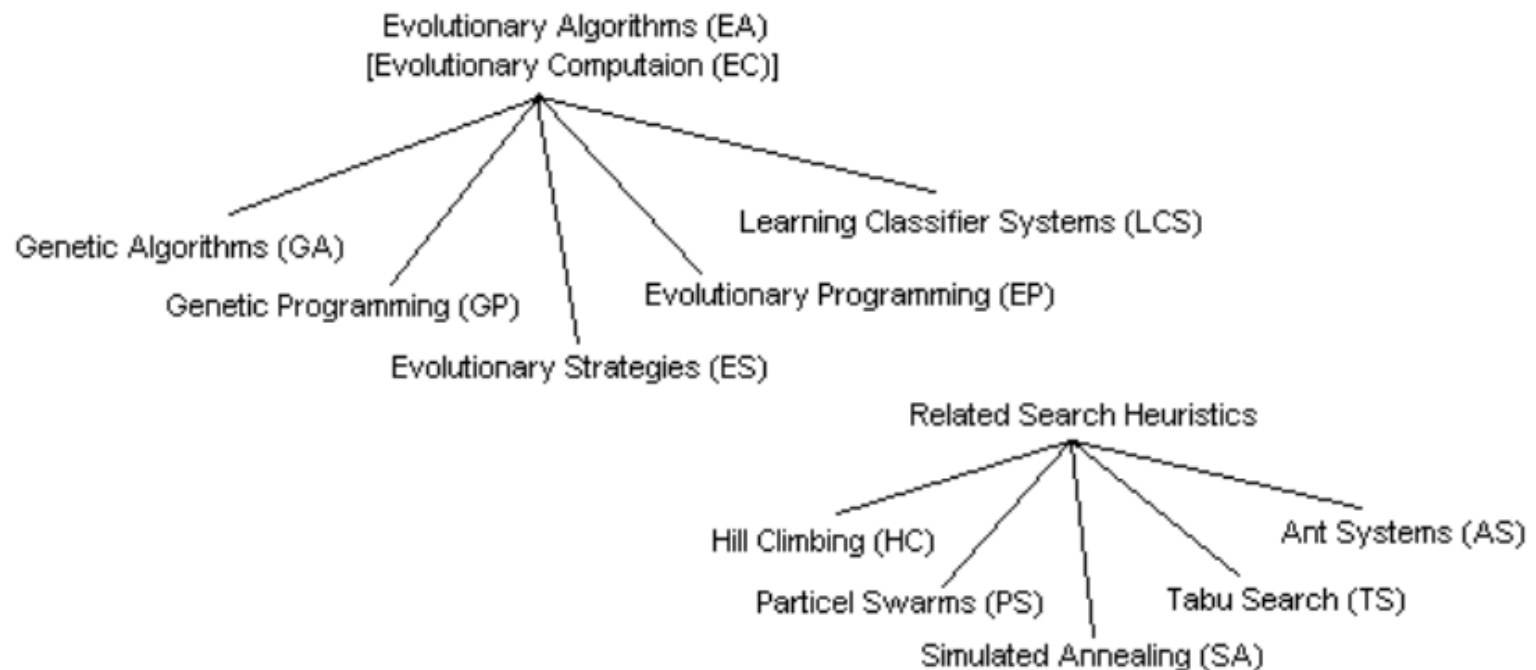


- we cannot choose both explore and exploit simultaneously.
- In order to overcome the Exploration-Exploitation Dilemma, we use the **Epsilon Greedy Policy**.

EVOLUTIONARY METHODS

- An evolutionary algorithm is considered a component of evolutionary computation in artificial intelligence.
- An evolutionary algorithm functions through the selection process in which the least fit members of the population set are eliminated, whereas the fit members are allowed to survive and continue until better solutions are determined
- Evolutionary algorithms are a heuristic-based approach to solving problems that cannot be easily solved in polynomial time, such as classically NP-Hard problems, and anything else that would take far too long to exhaustively process.

CLASSIFICATION OF EA METHODS



EVOLUTIONARY METHODS VS RL

- Reinforcement learning uses the concept of one agent, and the agent learns by interacting with the environment in different ways. In evolutionary algorithms, they usually start with many "agents" and only the "strong ones survive"
- Reinforcement learning agent(s) learns both positive and negative actions, but evolutionary algorithms only learns the optimal, and the negative or suboptimal solution information are discarded and lost.

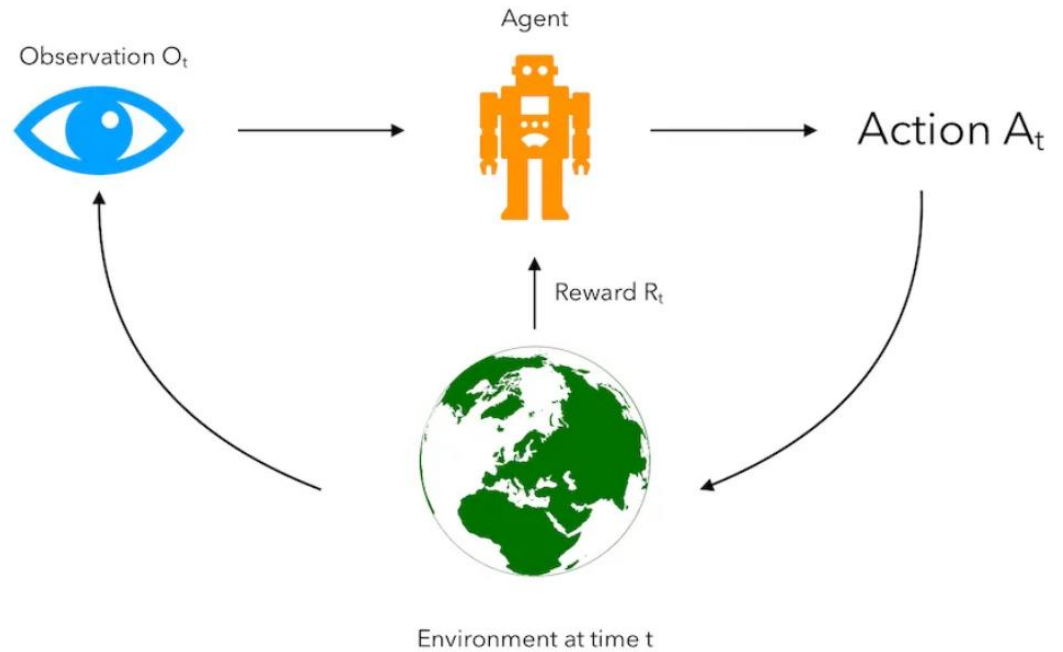
EXAMPLE

Building an algorithm to regulate the temperature in the room:

- The room is 15 °C, and you want it to be 23 °C.
- Using Reinforcement learning, the agent will try a bunch of different actions to increase and decrease the temperature. Eventually, it learns that increasing the temperature yields a good reward. But it also learns that reducing the temperature will yield a bad reward.
- For evolutionary algorithms, it initiates with a bunch of random agents that all have a preprogrammed set of actions it is going to do. Then the agents that has the "increase temperature" action survives and moves onto the next generation. Eventually, only agents that increase the temperature survive and are deemed the best solution. However, the algorithm does not know what happens if you decrease the temperature.

IMMEDIATE RL

■ immediate reinforcement learning problems are the one in which the outcome is immediate. So as soon as I take an action, I get a reward immediately.



EXAMPLE OF IMMEDIATE RL

1. Drug Testing is best example
2. Web site design (placement of BUY button)

QUIZ

1. What is the difference between a state-value function and an action-value function for an agent following a given policy?
- ☐ The state-value function tells us the policy, whereas the action-value function tells us the action and state.
 - ☒ The state-value function tells us how good any given state is for the agent, whereas the action-value function tells us how good it is for the agent to take any action from a given state.
 - ☐ The state-value function tells us the state as well as the action, whereas the action-value function tells us only the action for any state.
 - ☐ Both functions are the same.

QUIZ

2. Value functions are defined with respect to _____.
- ☐ the expected return
 - ☐ the policy
 - ☐ specific ways of acting
 - ☒ the expected return, specific ways of acting, and the policy

QUIZ

3. The agent has to explore and obtain a reward for what it knows already, but it has to exploit it to make better action selections in the future.

I. True

✓ II. False

4. Unlike other machine learnings, the Reinforcement learning has to face off two trade-offs. What are they?

✓ I. Exploitation

✓ II. Exploration

III. Environment

IV. Entertainment

QUIZ

5. A policy defines each step, and the environment sends a single positive or negative number to the reinforcement learning agent as a reward.

I. True

✓ II. False

6. Few Elements of Reinforcement Learning

✓ I. Environment

✓ II. Agent

✓ III. Value function

✓ IV. Reward

✓ V. Policy

QUIZ

7. Reinforcement learning is a computational approach to understand and automate the goal-directed learning and decision-making.

✓ I. True

II. False

8. Which of the following is true about reinforcement learning?

I. The agent gets rewards or penalty according to the action

II. It's an online learning

III. The target of an agent is to maximize the rewards

✓ IV. All of the above

QUIZ

9. How many types of feedback does reinforcement provide?

I. 1

✓ II. 2

III. 3

IV. 4

10. What is an agent in reinforcement learning?

I. Agent is the situation in which rewards are being exchanged

II. Agent is the simple value in reinforcement learning.

✓ III. An agent is an entity that explores the environment.



THANK YOU