Reinforcement Learning

Reinforcement Learning (RL) is a type of machine learning where an agent learns to interact with an environment to maximize a reward.

Example:

A dolphin with its trainer who uses rewards (fish)

What Can RL Do?

Reinforcement Learning (RL) has a wide range of applications across various domains.

Game Playing

- **Mastering complex games:** RL has been used to create agents that can defeat human champions in games like Go, Chess, and Dota 2.

Robotics

- **Autonomous navigation:** RL can enable robots to learn to navigate complex environments without explicit programming.

Finance

- **Risk management:** RL can help identify and mitigate risks in financial markets.

Other Applications

- **Self-driving cars:** RL can help self-driving cars learn to navigate roads and make safe driving decisions.

RL Problem Setup

Agent Environment Action Reward → (Area)

Example:

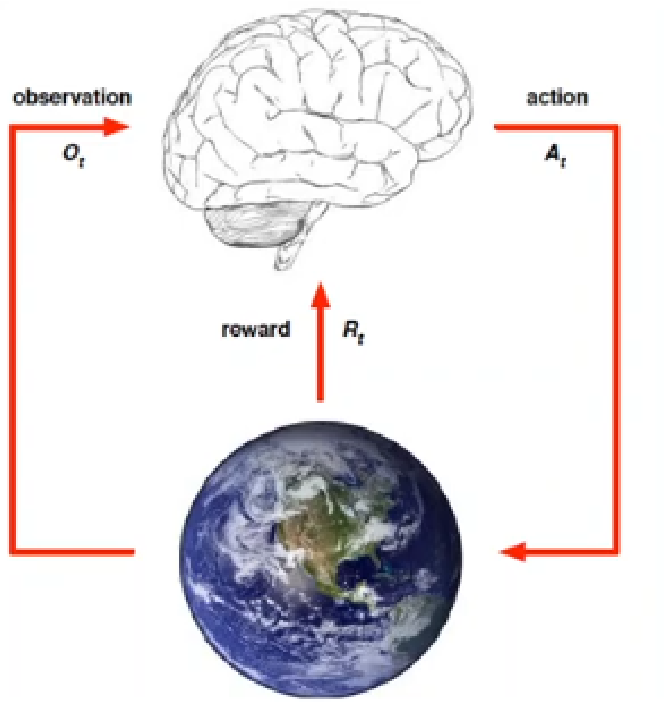
A Robot Learning to Navigate a Maze

Agent: A robot.

Environment: A maze with walls and a goal.

Action Space: Move forward, turn left, turn right.

Reward Function: Positive reward for reaching the goal, negative reward for hitting a wall.



Reward

Rewards are signals that an agent receives from the environment to indicate the success or failure of its actions. The objective of an agent in reinforcement learning is to maximize its cumulative reward over time.

Types

- **Intermittent Rewards:** Rewards that are given infrequently or at irregular intervals.
- **Cumulative Rewards:** Rewards that accumulate over time, reflecting the overall performance of the agent.

Example

Consider a robot that is learning to navigate a maze with a hidden goal. The robot receives an intermittent reward of 100 points when it reaches the goal, and a negative reward of -1 point for each step it takes. The robot's cumulative reward would be the sum of all rewards it receives over the course of its exploration.

State

In reinforcement learning, a state represents the current situation or condition of the agent within its environment.

Environment State

The environment state is the overall condition of the environment at a particular time. It includes information about the objects, positions, and relationships between different elements within the environment.

Agent State

The agent state is the internal representation of the environment state that the agent maintains. It may include the agent's current position, velocity, and other relevant information.

Example

Consider a self-driving car.

- **Environment State:** The environment state might include the car's current position, speed, the positions of other vehicles, road conditions, traffic signals, and weather conditions.
- **Agent State:** The agent state might include the car's internal sensors (e.g., lidar, radar, cameras), its current speed, steering angle, and the status of its various components.

Fully Observable Environments

- **Definition:** In a fully observable environment, the agent has complete access to all relevant information about the environment state at any given time. This means that the agent can perfectly observe the current situation and make decisions based on a complete understanding of the environment.
- **Examples:**
    - Chess: The agent (player) can see the entire board and the positions of all pieces.
    - Tic-Tac-Toe: Both players have a complete view of the game board.

Partially Observable Environments

- **Definition:** In a partially observable environment, the agent does not have access to all relevant information about the environment state. This means that the agent must make decisions based on limited or incomplete information.

- **Examples:**
  - Poker: Players only have partial information about their opponents' hands.
  - Autonomous driving: The car's sensors may not be able to detect all potential hazards, such as hidden objects or unexpected changes in the road.

RL Agent Ingredients

Policy

It essentially maps states to actions, guiding the agent's behavior.

Types of Policies

1. **Deterministic Policies:**
   - **Definition:** A deterministic policy maps each state to a single, specific action. In other words, given a particular state, the policy always chooses the same action.
   - **Example:** A chess-playing agent might always choose to capture a pawn if one is available.
2. **Stochastic Policies:**
   - **Definition:** A stochastic policy maps each state to a probability distribution over actions. This means that for a given state, the agent can choose different actions with varying probabilities.
   - **Example:** A robot navigating a maze might choose to turn left or right with equal probability at an intersection.

Value

Value in reinforcement learning refers to the expected future reward that an agent can obtain from a given state by following a particular policy.

Types of Value Functions

1. **State-Value Function (V):**
    - **Definition:** The state-value function estimates the expected future reward that the agent can obtain from a given state by following a particular policy.
    - **Notation:** V(s) represents the value of state s under policy π.
    - **Calculation:** The value of a state is typically calculated recursively, considering the expected rewards and the values of subsequent states.
2. **Action-Value Function (Q):**
    - **Definition:** The action-value function estimates the expected future reward that the agent can obtain from a given state by taking a particular action and then following a particular policy.
    - **Notation:** Q(s, a) represents the value of taking action a in state s under policy π.
    - **Calculation:** The action-value function is also calculated recursively, considering the expected rewards and the values of subsequent states.

Model

Environment Model is a representation of the environment that the agent interacts with. It predicts the next state and reward given the current state and action. This model can be used to plan ahead and make more informed decisions.

Types of Environment Models

1. **State Transition Model:**
    - **Definition:** A state transition model predicts the next state given the current state and action.
    - **Example:** In a grid world, the state transition model might predict the next grid cell based on the current cell and the action taken (e.g., move up, down, left, or right).
2. **Reward Model:**

- **Definition:** A reward model predicts the reward that the agent will receive given the current state and action.
- **Example:** In a game of chess, the reward model might assign a positive reward for capturing a piece and a negative reward for losing a piece.

RL Problem

Reinforcement learning (RL) problems can be broadly categorized into two main types: prediction and control.

Prediction

- **Goal:** To estimate the expected future reward from a given state or state-action pair.
- **Tasks:**
  - **Policy Evaluation:** Given a fixed policy, estimate the value function for each state or state-action pair.
  - **Prediction:** Predict the future outcome of a specific action or sequence of actions.
- **Example:** In a chess game, a prediction task could be to estimate the probability of winning from a given board position under a particular playing strategy.

Control

- **Goal:** To find an optimal policy that maximizes the expected cumulative reward over time.
- **Tasks:**
  - **Policy Optimization:** Find the best policy to follow in a given environment.
  - **Decision Making:** Make optimal decisions in real-time based on the current state and the learned policy.
- **Example:** In a robot navigation task, a control problem would be to find the best path for the robot to reach its goal while avoiding obstacles.