

# Introdução ao Reinforcement Learning

# O que é RL

*Particulares*

- Três grandes domínios do ML:
  - Supervisionado: Aprender por dados fornecidos por um oráculo
  - Não-supervisionado: Encontrar padrões em dados
  - Reforçado: Maximizar um sinal através de ações

# O que é RL

*Antes de começar*

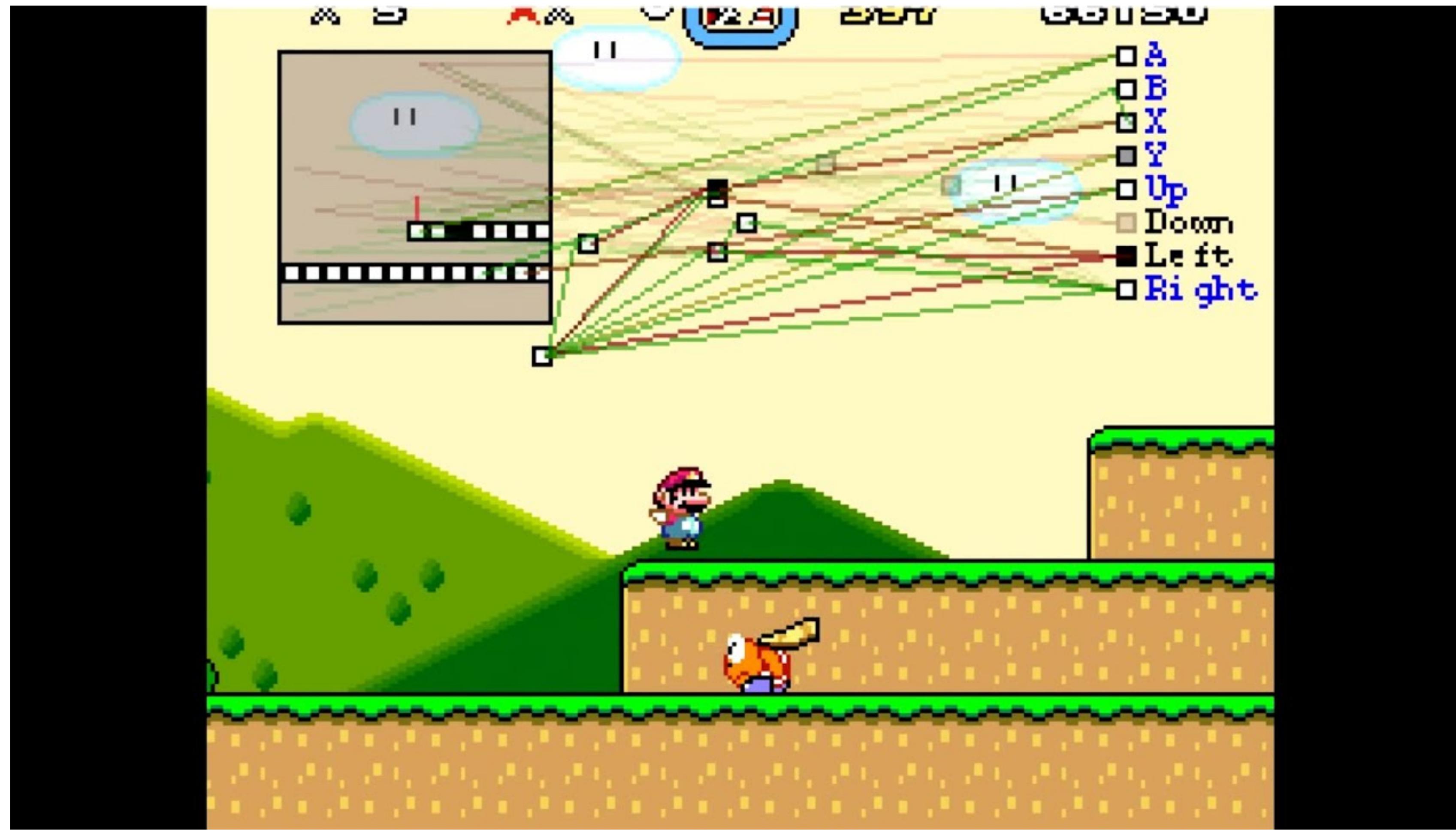
- RL NÃO SE RESUME A MACHINE LEARNING
- Três grandes domínios do ML:
  - Supervisionado
  - Não supervisionado
  - Reforçado
- RL é o *primo distante* dos outros dois. Existem adaptações de técnicas de Aprendizado Supervisionado e Não-supervisionado para RL

# O que é RL

## *Definição*

- “É aprender o que se fazer, como mapear situações a ações, de forma a maximizar um sinal de pagamento numérico. O agente não é dito quais ações tomar, ao invés disso ele deve descobrir quais ações provém o maior pagamento através de tentativa e erro” - Sutton, Barto 2018
- Como um Bebê aprende a andar? Nasce e anda?

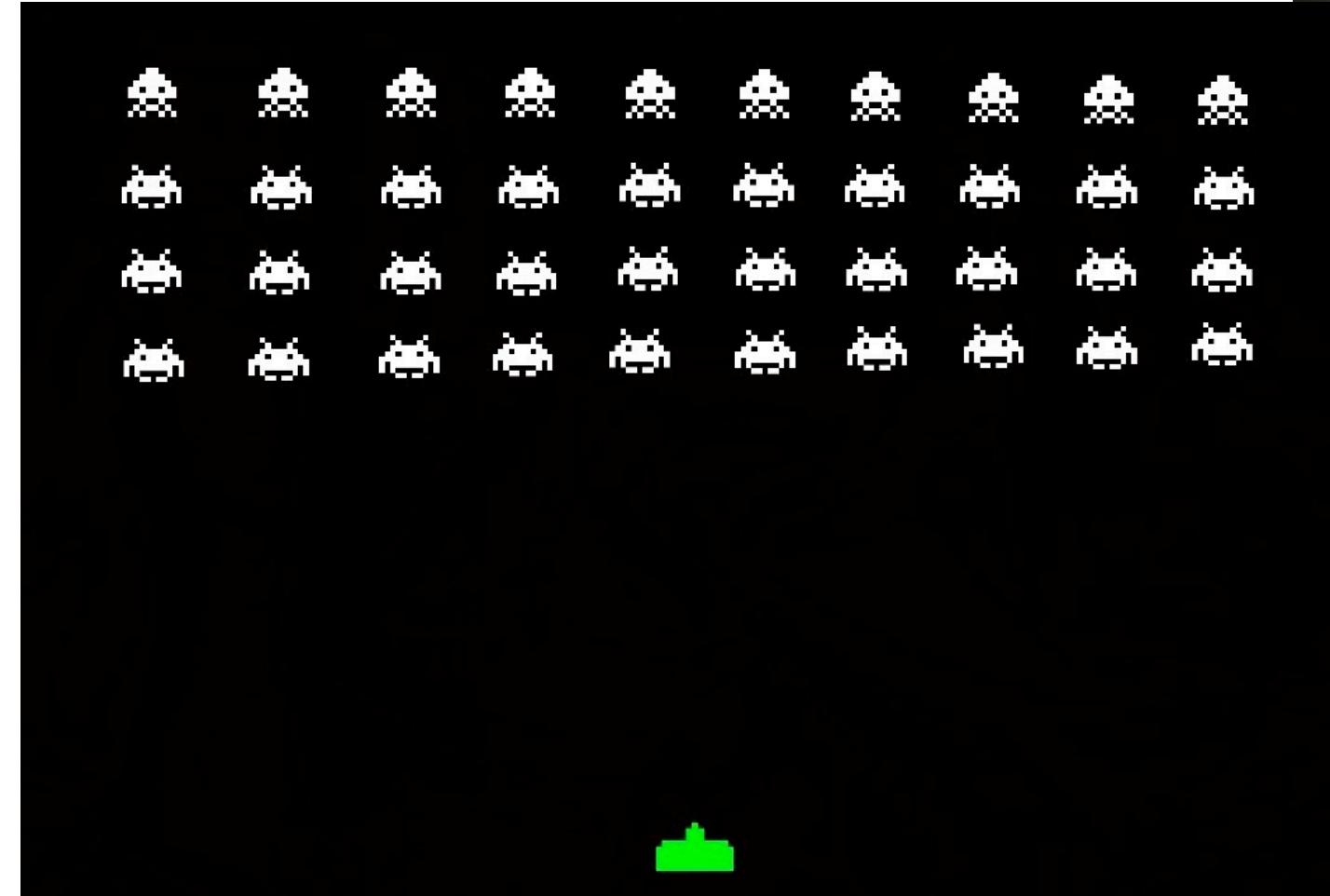
# O que é RL



# O que é RL

*Exemplos*

- Exemplos de RL:



# Elementos de RL

*O que vai estar quase sempre presente em qualquer tarefa de RL*

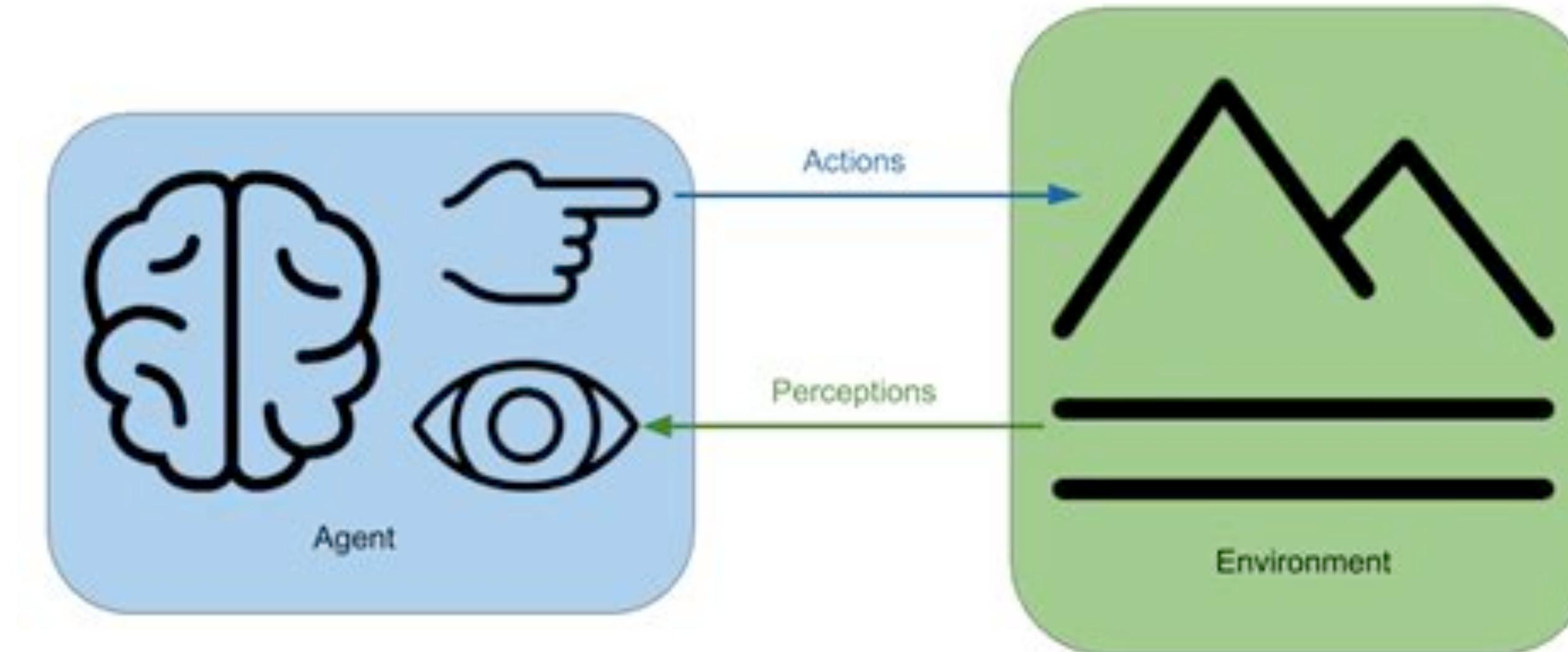
- De acordo com a definição de Sutton e Barto<sup>\*</sup>:
  - Agente
  - Ambiente
  - Pagamento (Reward)
  - Função de Valor (Value Function)
  - Política (Policy)
  - Modelo (Model)

<sup>\*</sup>Difere de Autor para Autor, usaremos a de Sutton e Barto por ser a mais consolidada

# Elementos de RL

*Agente*

- **Agente**: aquele que vai executar a ação e aprende no ambiente
- Relacionados: Estado, Ação, Pagamento



# Elementos de RL

*Agente*

- Estado (State,  $s_i$ ): Estado em que o Agente se encontra no ambiente
- Ação (Action,  $a_i$ ): Ação que o Agente toma no ambiente
- Pagamento (Reward,  $r_i$ ): Pagamento que o agente recebe de acordo com seu estado e ação no ambiente

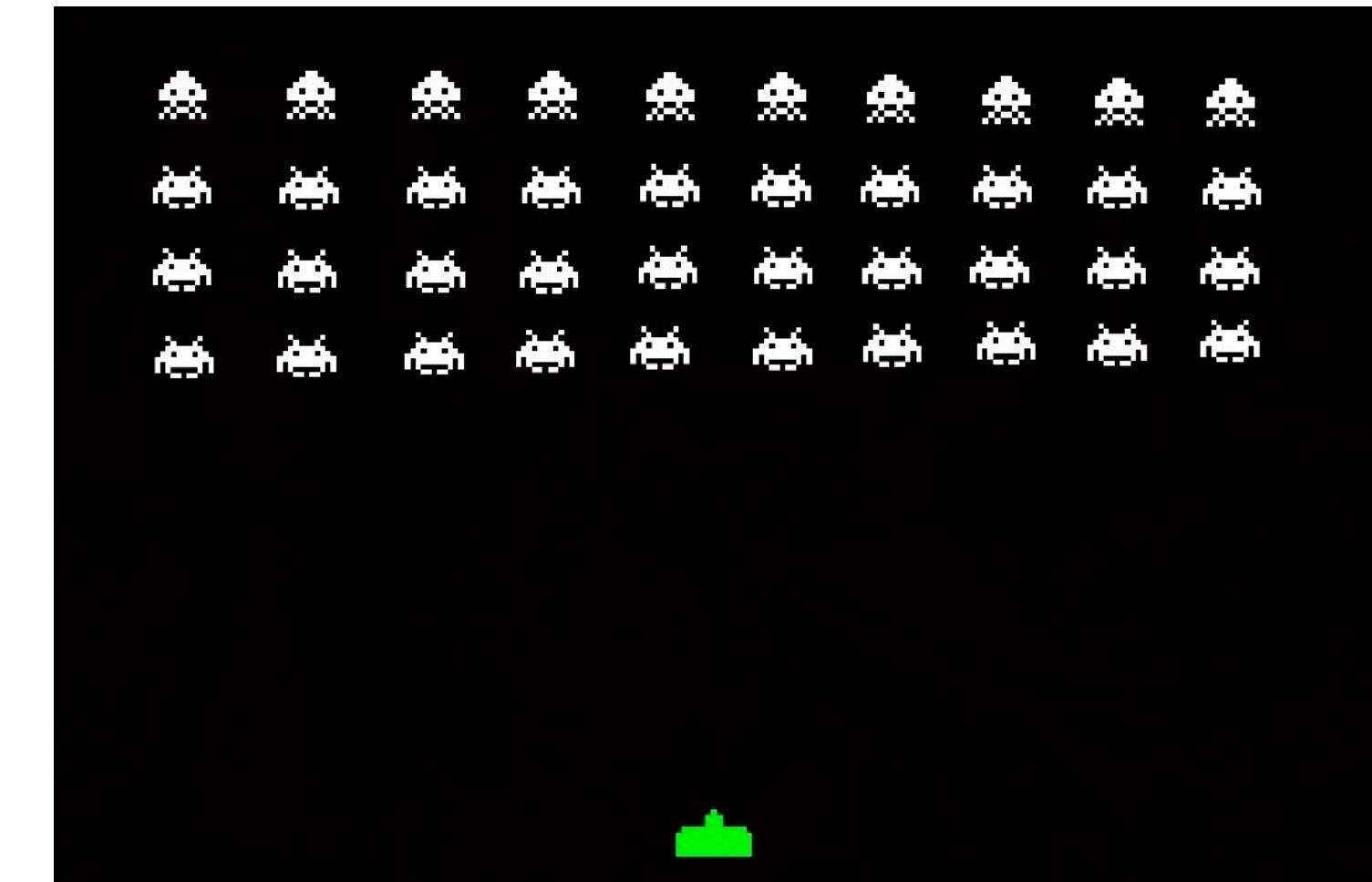
**LEMBRE-SE:** Pagamento é o valor numérico IMEDIATO recebido pelo Agente

# Elementos de RL

## Ambiente

**Ambiente (*Environment*)**: É o meio onde o agente está situado

- O ambiente proverá o pagamento de acordo com a **ação** e/ou o estado do Agente
- Exemplos:



# Tipos de RL

*Tipos de Ambiente*

- Ambiente Finito



- Ambiente Infinito



# Tipos de RL

*Tipos de Ambiente*

- Ambiente Episódico



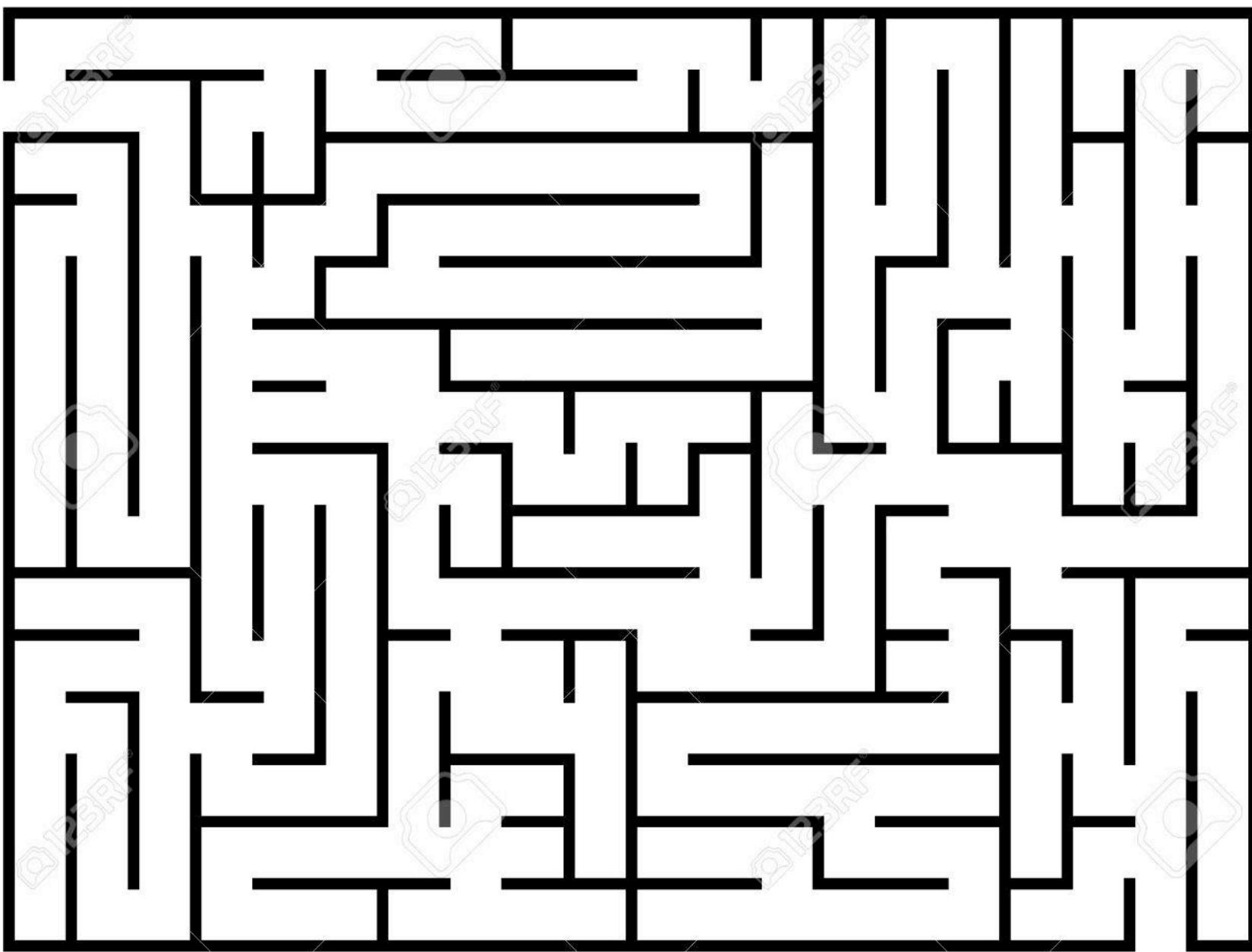
- Ambiente Contínuo



# Tipos de RL

## *Tipos de Ambiente*

- Ambiente com **Informação completa**



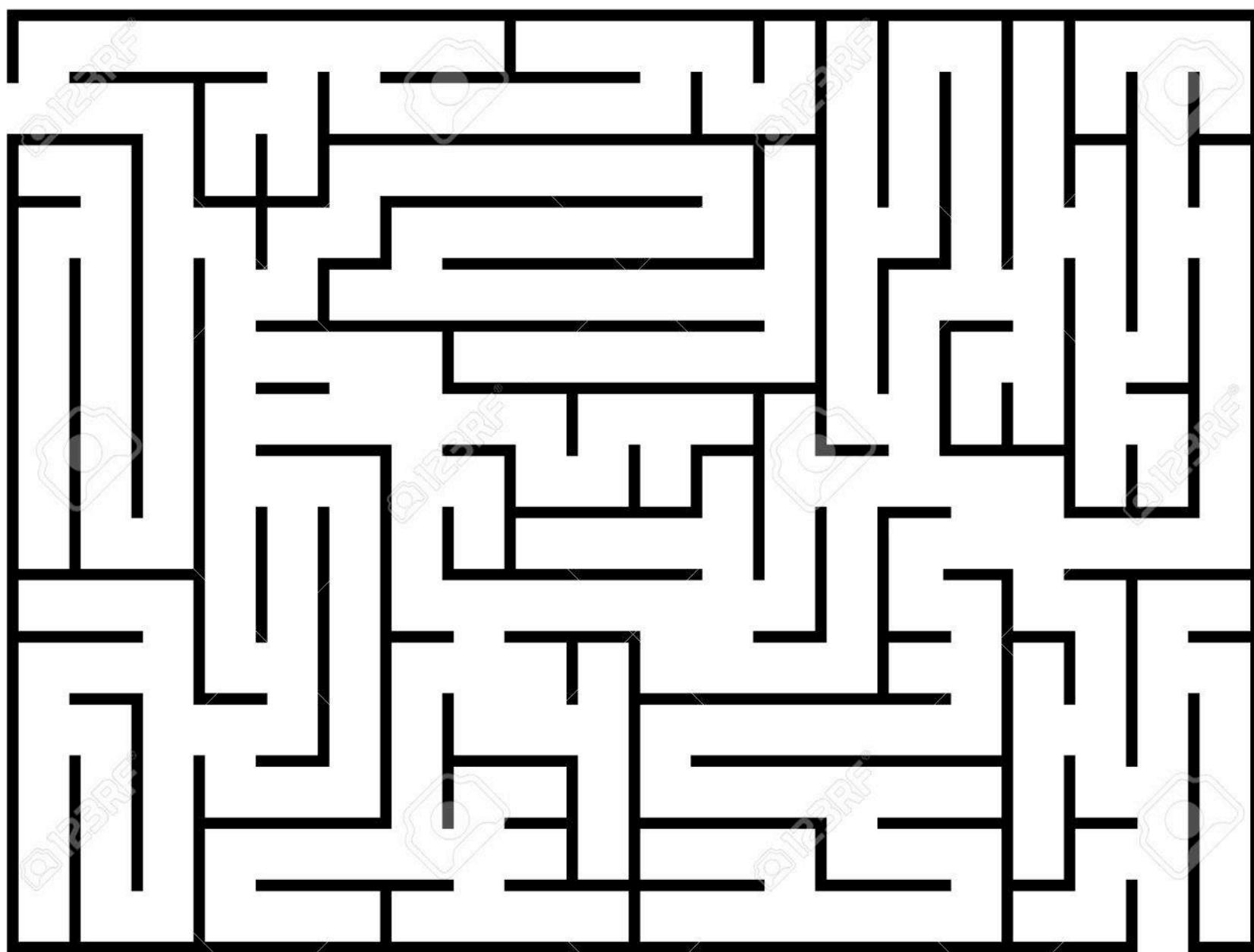
- Ambiente com **informação parcial**



# Tipos de RL

## *Tipos de Ambiente*

- Ambiente com ações discretas



- Ambiente com ações continuas



# Elementos de RL

*O que vai estar quase sempre presente em qualquer tarefa de RL*

## Função de Pagamento (Value Function)

- Agentes recebem pagamento cada vez que tomam uma ação, como saber qual será o pagamento em  $t + 1$ ,  $t + 2$  ou  $t + n$ ?
- O que é melhor? **Muito pagamento imediato e pouco no futuro ou pouco pagamento imediato e muito no futuro?**
- Normalmente, pagamentos são descontados ao longo do tempo a fim de punir jogadas em que o Agente fique “preso”

# Elementos de RL

*O que vai estar quase sempre presente em qualquer tarefa de RL*

## Política (Policy) $\pi$

- Define a forma de um Agente a agir em um certo tempo e em determinado estado
- Define o “comportamento” de um agente
- Deterministica ou estocástica

# Elementos de RL

*O que vai estar quase sempre presente em qualquer tarefa de RL*

## Política (Policy) $\pi$

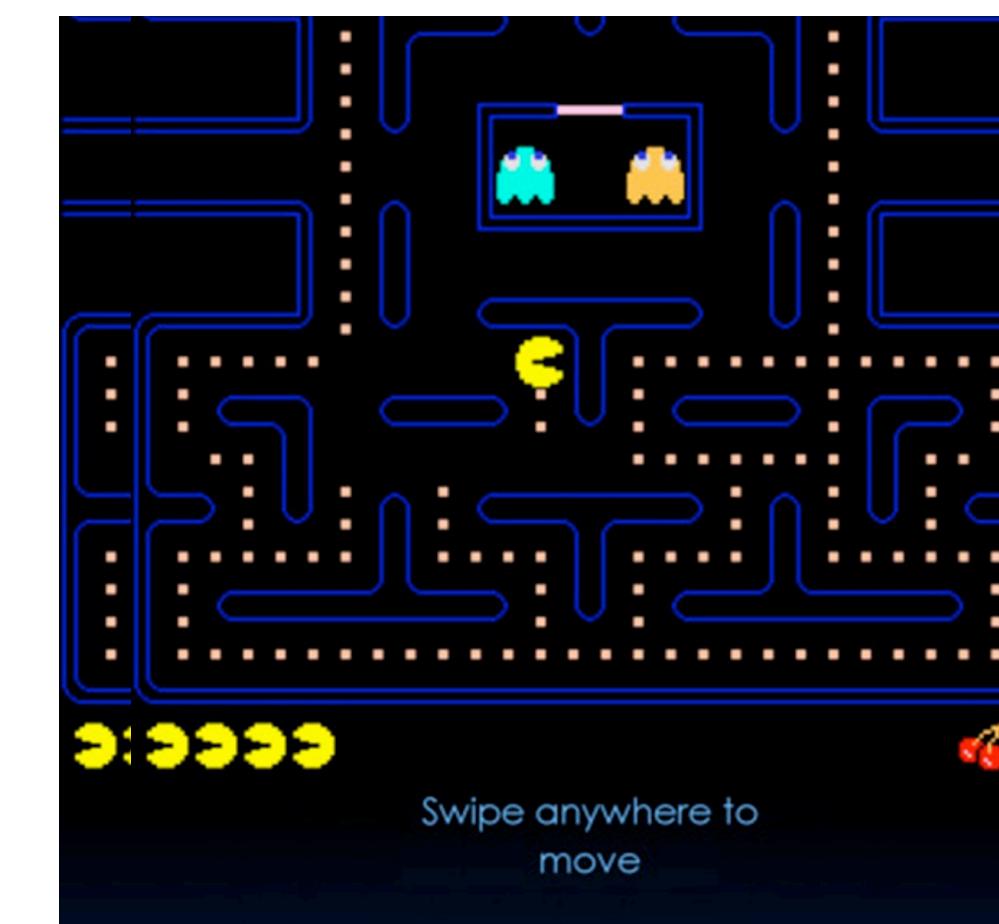
- Função  $\pi : S \rightarrow A$
- Escrito:
  - $\pi(s)$  para determinística
  - $\pi(s | a)$  para estocástica

# Elementos de RL

*O que vai estar quase sempre presente em qualquer tarefa de RL*

## Modelo (Model)

- Um simulacro de como o ambiente vai se comportar
- Podemos chamar de “planejamento”
- *Model-based* vs. *Model-free*: planejamento vs. tentativa e erro



# Ambientes Discretos

*Um início*

- Para começarmos, vamos ver os algoritmos mais clássicos para ambientes episódicos, com ações discretas e políticas determinísticas
- Ambientes contínuos usam variações dos algoritmos discretos
- A mesma coisa para métodos de RL que usam Deep Learning, versões de algoritmos clássicos que adicionam redes neurais

# Ambientes Discretos

*Tipos de Algoritmos*

- *Markov Decision Process (MDP)*
- Programação Dinâmica
- Métodos de Monte-Carlo
- SARSA
- Q-Learning

# A partir de Agora

Todos os nossos algoritmos terão o seguinte modelo:

