

Un vistazo al Aprendizaje por Refuerzo

Luis David Solano Santamaría
luis.solanosantamaria@ucr.ac.cr



¿Quién soy?

Luis David Solano

- Egresado de la Escuela de Ciencias de la Computación e Informática con énfasis en CC.
- Asistente de cursos en esta misma.
- Asistente de machine learning en proyecto de investigación en el CICA.
- Investigador en temas de machine learning aplicado a ambiente y medicina.



Dinámica



GitHub con material



Apuntes colectivos

Contenidos de la sesión

01

Teoría

02

Algoritmos

03

Aplicaciones

04

Recomendaciones

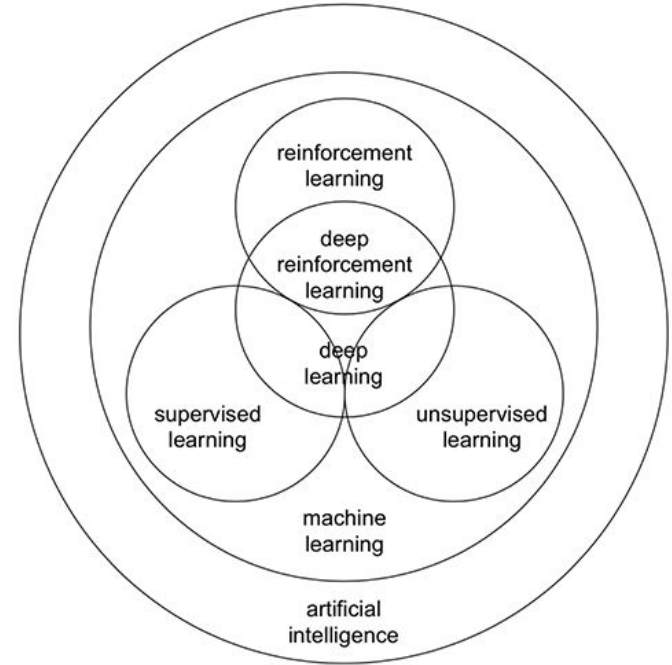
05

Ejercicio práctico

01

Teoría

¿Inteligencia Artificial?



Yuxi Li, Deep Reinforcement Learning, arXiv, 2018

Agente inteligente

Un **agente** es algo que puede ser modelado de una manera que perciba su ambiente y actúa sobre este para cumplir una tarea.

Los **agentes inteligentes** buscan resolver estas tareas de maneras **racionales**. Debe seleccionar la acción que maximice su rendimiento, según la evidencia que ha percibido y el conocimiento interno que posea.

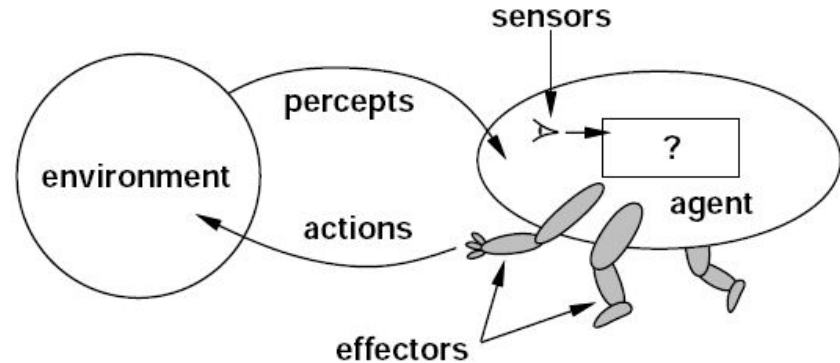


Foto de [Agents in AI](#)

¿Machine Learning?

Supervised Learning

1. Consiste en aprender de un conjunto de **datos de entrenamiento**, con **etiquetas** sobre la naturaleza de estos.
2. La idea es **generalizar** respuestas para que actúe correctamente en situaciones no presentes en el entrenamiento.
3. **Ejemplos:** clasificación de imágenes, detección de spam...

Unsupervised Learning

1. Consiste en aprender de un conjunto de datos **sin etiquetas**, buscando la **estructura** escondida en colecciones de datos.
2. **Ejemplos:** agrupamientos de datos en búsqueda de clases.

¿Qué es el aprendizaje por refuerzo?

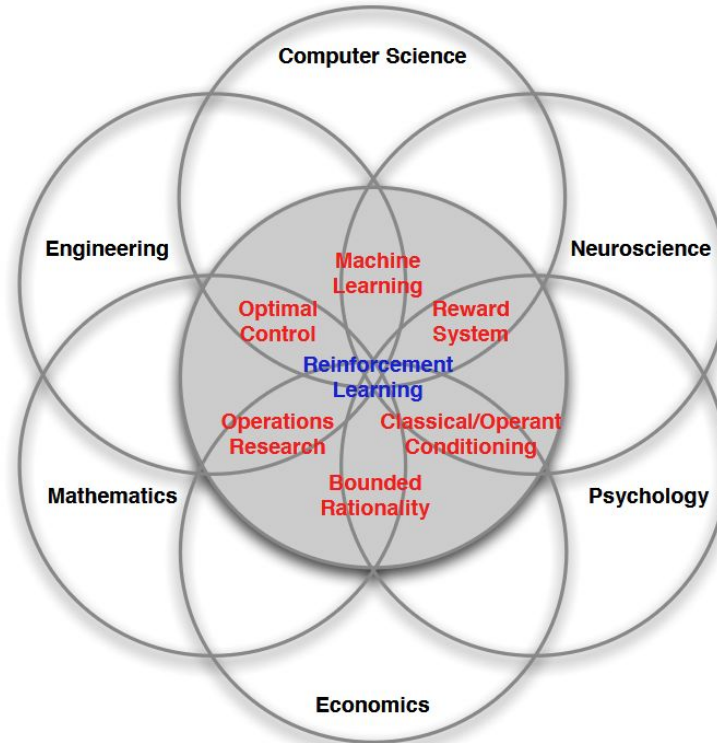
El **aprendizaje por refuerzo**, o **reinforcement learning (RL)** es una rama del aprendizaje automático.

Aquí **un agente** aprende mediante una serie de **refuerzos**, los cuales pueden ser castigos o recompensas, a actuar en un ambiente.

Estos agentes aprenden por medio de **interacción** con el **ambiente**, similar a como las personas o animales aprenden.

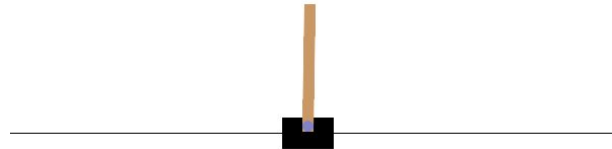
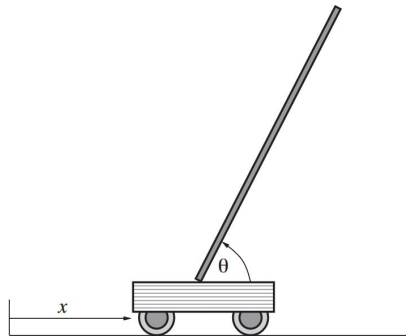


¿Qué es el aprendizaje por refuerzo?



¿Qué es el aprendizaje por refuerzo?

1. En el aprendizaje por refuerzo, el agente aprende a **relacionar situaciones a acciones** con el propósito de **maximizar** una **señal numérica de recompensa**.
2. El agente no sabe cuáles acciones tomar, pero en vez debe descubrir cuáles acciones dan la mejor recompensa al probarla por medio de prueba



GIF de [Gymnasium API](#)

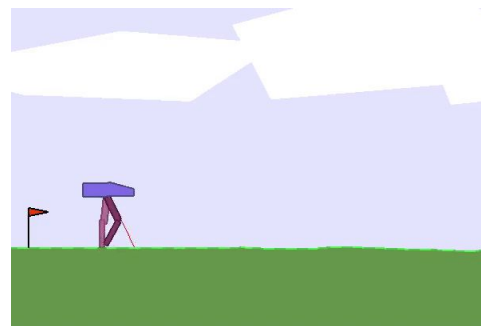
Hablemos sobre la recompensa...

La recompensa R_t es una señal escalar de retroalimentación para el tiempo t .

La recompensa lo es todo para este tipo de aprendizaje. El propósito de los agentes es **maximizar** su recompensa total.

¿Cómo esto nos permite resolver problemas?

Reward hypothesis establece que todas las metas pueden describirse mediante la **maximización de la recompensa** acumulativa esperada.



Ejemplos de reward hypothesis

Hacer maniobras en un helicóptero

- Recompensa + por seguir la trayectoria deseada.
- Recompensa - por estrellarse.

Derrotar al campeón mundial de ajedrez

- Recompensa + / - por ganar / perder un juego.

Controlar una estación de energía

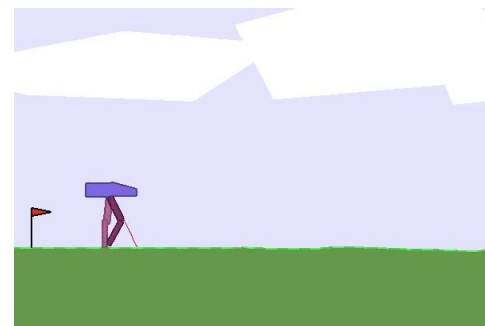
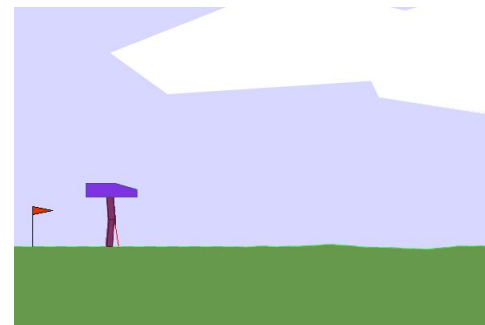
- Recompensa + por producir energía.
- Recompensa - por exceder regulaciones de seguridad.

Hacer un robot humanoide caminar.

- Recompensa + por movimiento hacia el frente.
- Recompensa - por caerse.

Jugar juegos de Atari

- Recompensa + / - por incrementar / perder puntos.



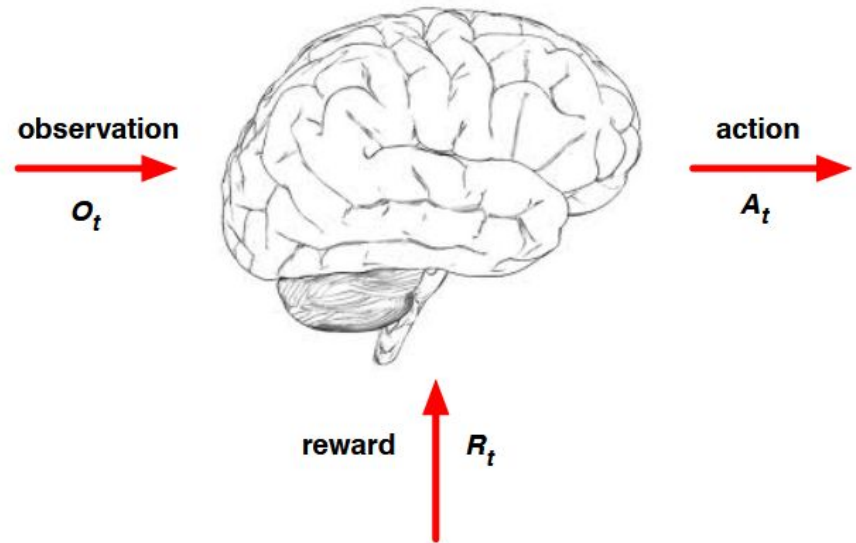
Ejemplos de reward hypothesis



Agente en aprendizaje por refuerzo

El aprendizaje por refuerzo trabaja con **agentes racionales** para completar una tarea en un ambiente desconocido.

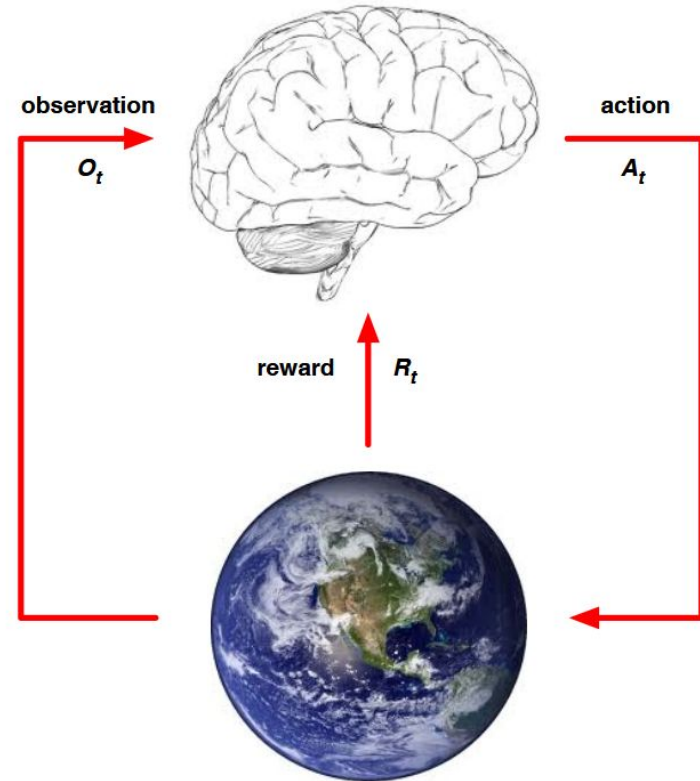
En intervalos de tiempo, al agente se le da información del **ambiente** y una **recompensa**. Este responde con una **acción**.



Agente en aprendizaje por refuerzo

El aprendizaje por refuerzo trabaja con **agentes racionales** para completar una tarea en un ambiente desconocido.

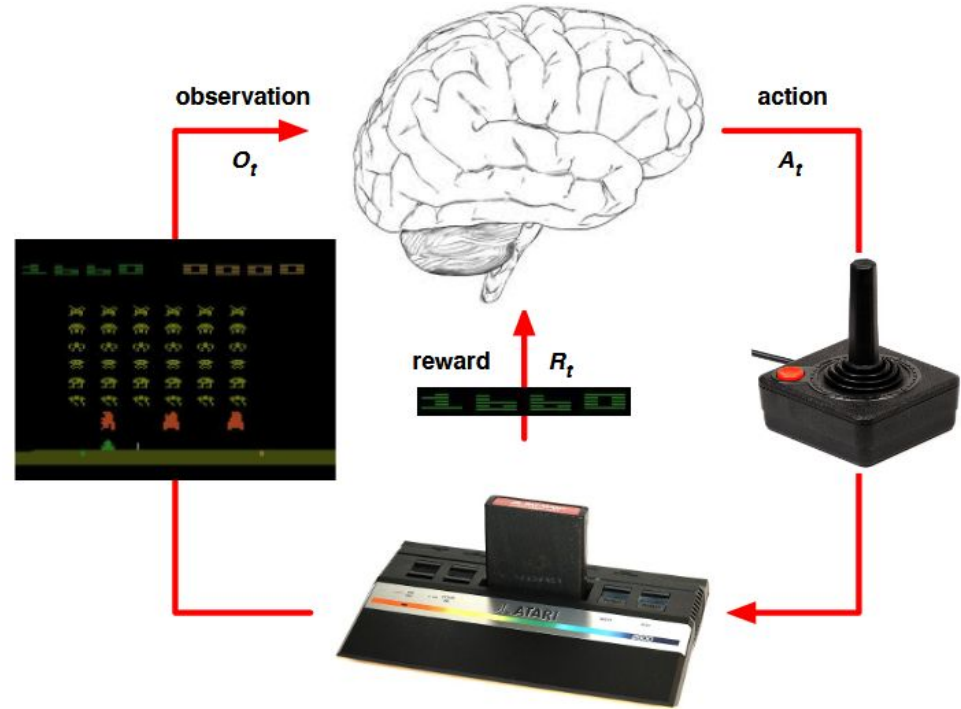
En intervalos de tiempo, al agente se le da información del **ambiente** y una **recompensa**. Este responde con una **acción**.



Agente en aprendizaje por refuerzo

El aprendizaje por refuerzo trabaja con **agentes racionales** para completar una tarea en un ambiente desconocido.

En intervalos de tiempo, al agente se le da información del **ambiente** y una **recompensa**. Este responde con una **acción**.



Consideremos una analogía

Un maestro de ajedrez va a realizar
un movimiento.

¿Esta información en qué se basa?



¿Qué es la historia?

La **historia** es la secuencia de observaciones, recompensas y acciones.

$$H_t = O_1, R_1, A_1, \dots, A_{t-1}, O_t, R_t$$

Entonces, representa todo lo que ha ocurrido en lo que el **agente ha interactuado**.

Queremos que nuestro agente seleccione una **acción según lo que ha ocurrido**.

¿Cuánto ocupamos de la historia?

¡Un estado!

Estado de Markov / Estado de Información

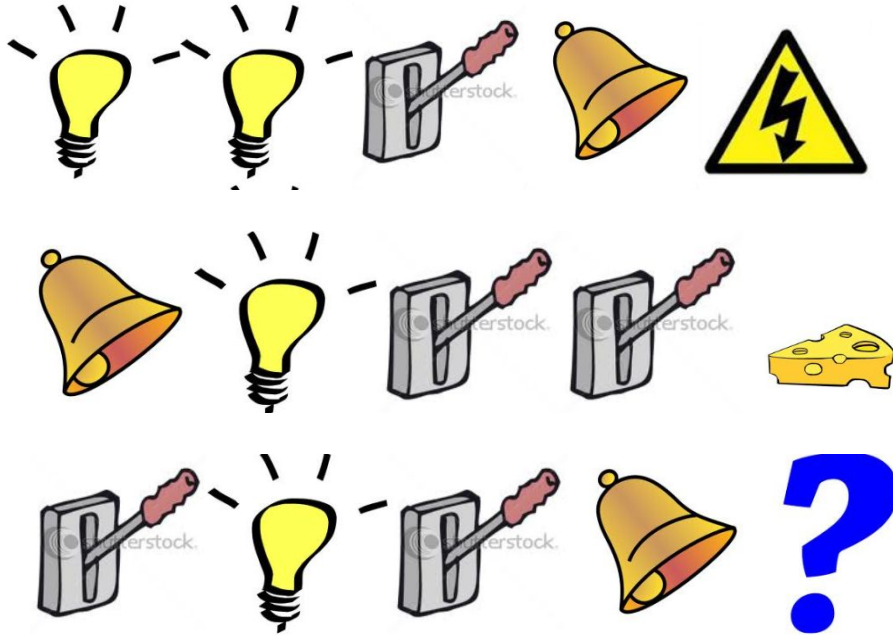
Un **estado de Markov** contiene toda la **información útil de la historia** para tomar una decisión.

Un estado S_t se considera de Markov si y sólo si

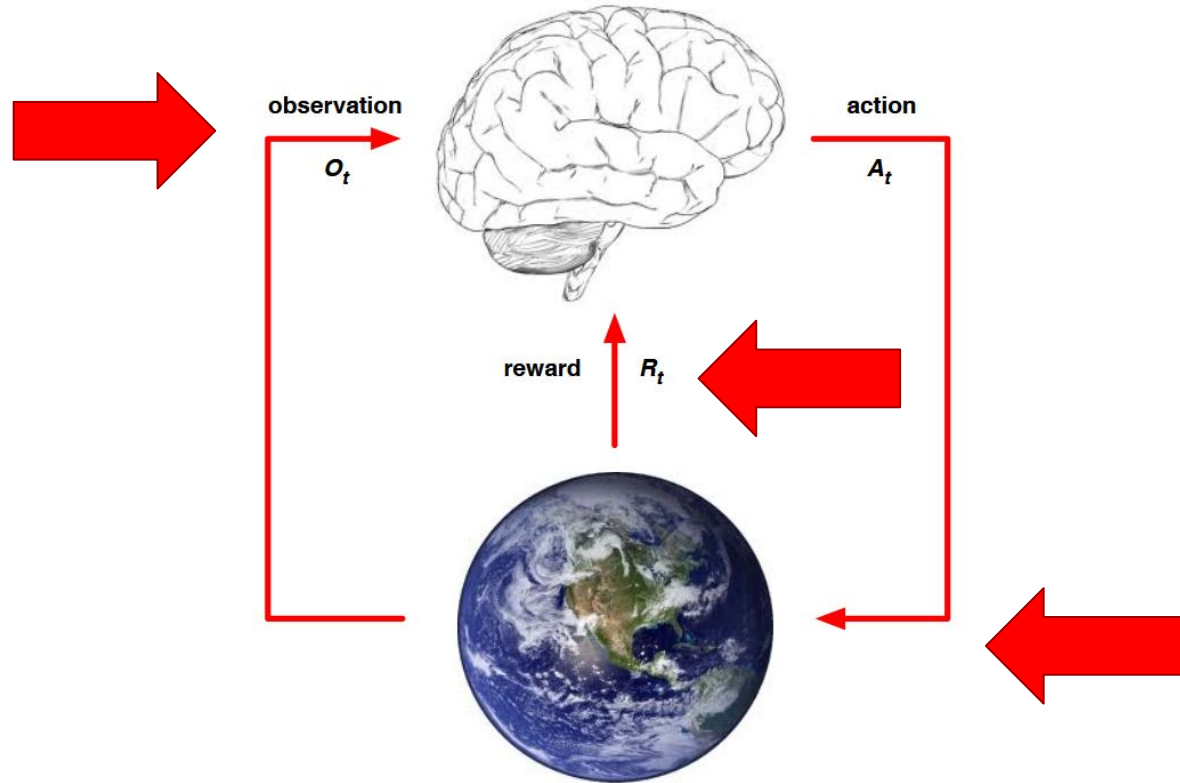
$$P[S_{t+1} \mid S_t] = P[S_{t+1} \mid S_1, S_2, \dots, S_t]$$

Podemos descartar todos los estados anteriores y conservar sólo el estado actual para obtener la misma caracterización del futuro.

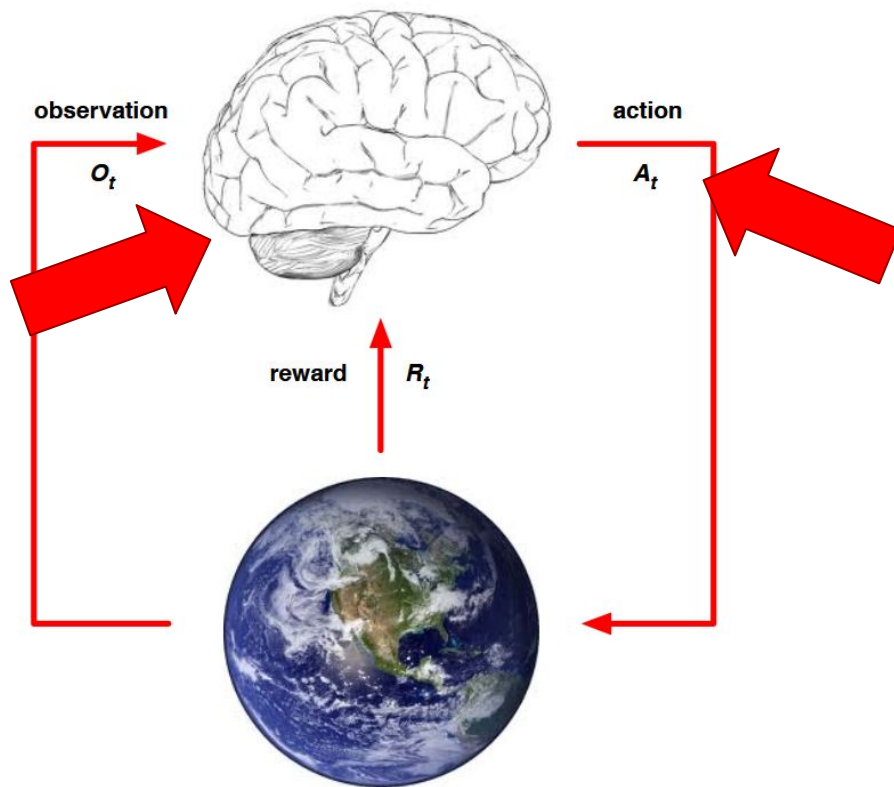
Veamos un ejemplo de la importancia del estado



Hemos hablado mucho de...



Nos falta...



¿Espacio de acción?

Discreto

Tenemos un **conjunto finito** de acciones disponibles para realizar sobre el ambiente.



Continuo

Tenemos un número de acciones **infinita** disponible para nosotros.



Dentro de un agente...

01

Política

02

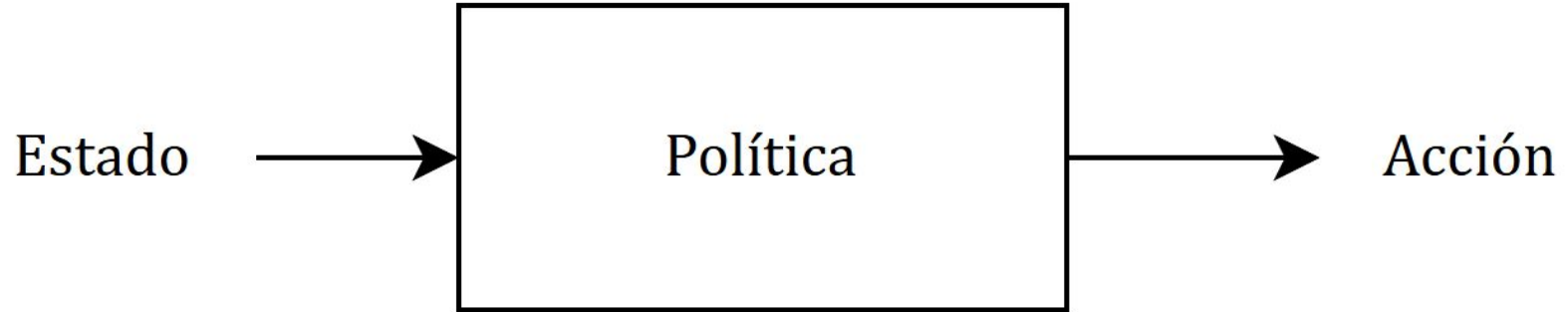
Función de valor

03

Modelo del
ambiente

¿Política?

La **política** se refiere a la **función** que define el **comportamiento del agente**.



¿Política?

$$\pi : S \rightarrow A$$

La política π es una función que mapea cada estado $s \in S$ a una acción $a \in A$

$$\pi(s) = a$$

Si es determinista cada estado s siempre retorna la misma acción a

$$\pi(a \mid s) = P[A_t = a \mid S_t = s]$$

Si es estocástica, π retorna una distribución de probabilidad

¿Función de valor?

Se utiliza para **predecir** la recompensa futura.

Con esta predicción evaluamos en realidad qué tan bueno o malo es un estado.

De esta manera tenemos una **política implícita**, no tenemos una política real.

Como sabemos lo bueno del estado podemos **inferir** la acción que queremos tomar.

¿Función de valor?

State Value Function $V(s)$

$$V^{\pi}(s) = \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t R_t \mid S_0 = s, \pi \right]$$

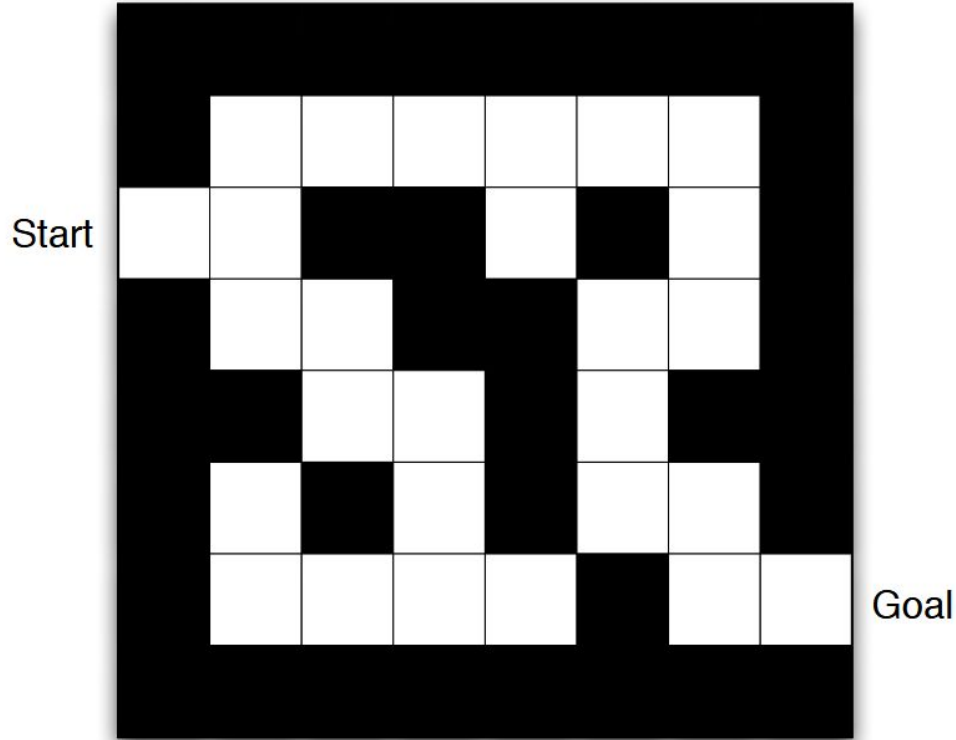
Representa la recompensa en el futuro de un estado, qué tan bueno es un estado.

Action Value Function $Q(s,a)$

$$Q^{\pi}(s, a) = \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t R_t \mid S_0 = s, A_0 = a, \pi \right]$$

Representa la recompensa tomando una acción para un estado, qué tan bueno es seguir esta acción.

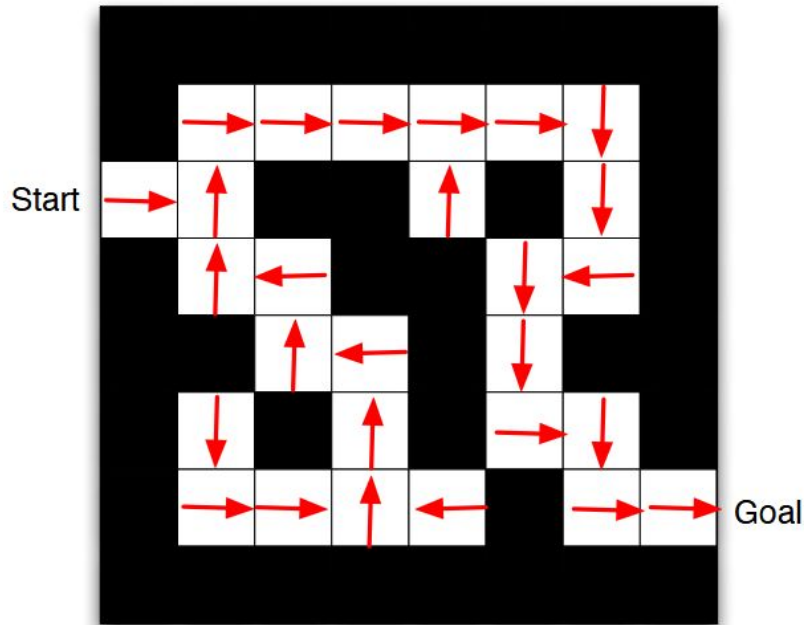
Ejemplo de un laberinto



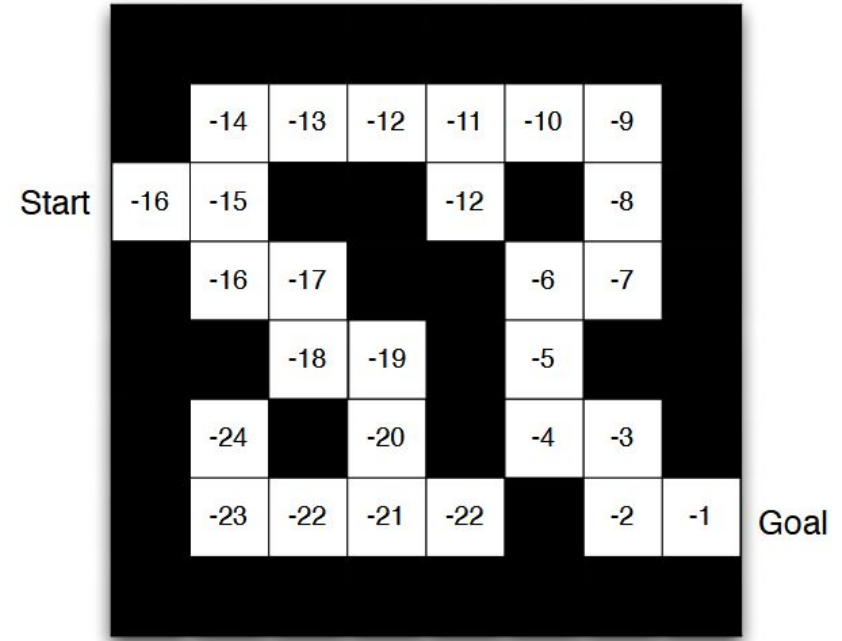
1. **Recompensa:** -1 por cada tiempo t
2. **Acciones:** Arriba, Abajo, Izq, Der
3. **Estados:** Cuadro actual

Ejemplo de un laberinto

Política $\pi(s)$



Función de valor $v(s)$



Categorías de RL

Value based

Posee la política de manera implícita, por medio del **value function**

Policy based

Posee una **política**, por lo que no necesita un value function.

Categorías de RL

Model based

Puede ser cualquiera de los anteriores y **posee un modelo del ambiente** para predecir lo que va a ocurrir.

Model free

Puede ser cualquiera de los anteriores y **no posee un modelo del ambiente** para predecir lo que va a ocurrir, aprende **meramente por experiencias**.

Dilemas del Reinforcement Learning

| Explotación | Exploración |
|--|--|
| <ul style="list-style-type: none">— Agente intenta maximizar con la información que tiene.— Para maximizar la recompensa, el agente debe preferir las acciones que ya conoce que son productivas haciendo resultados. | <ul style="list-style-type: none">— Agente intenta explorar nuevas opciones.— Para maximizar recompensa, debe probar nuevas a ver si existe alguna mejor. |

02

Algoritmos

Q-Learning

¿A qué les suena la Q en el nombre?

$$Q^{\pi}(s, a) = \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t R_t \mid S_0 = s, A_0 = a, \pi \right]$$

Queremos **aprender** la función Q

Tenemos un algoritmo **value based** y **model free**

La función Q en acción

Q Table

Acciones

Estados

| | | | | | | |
|--|--|--|--|--|--|--|
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

El Q-Learning utiliza dos políticas

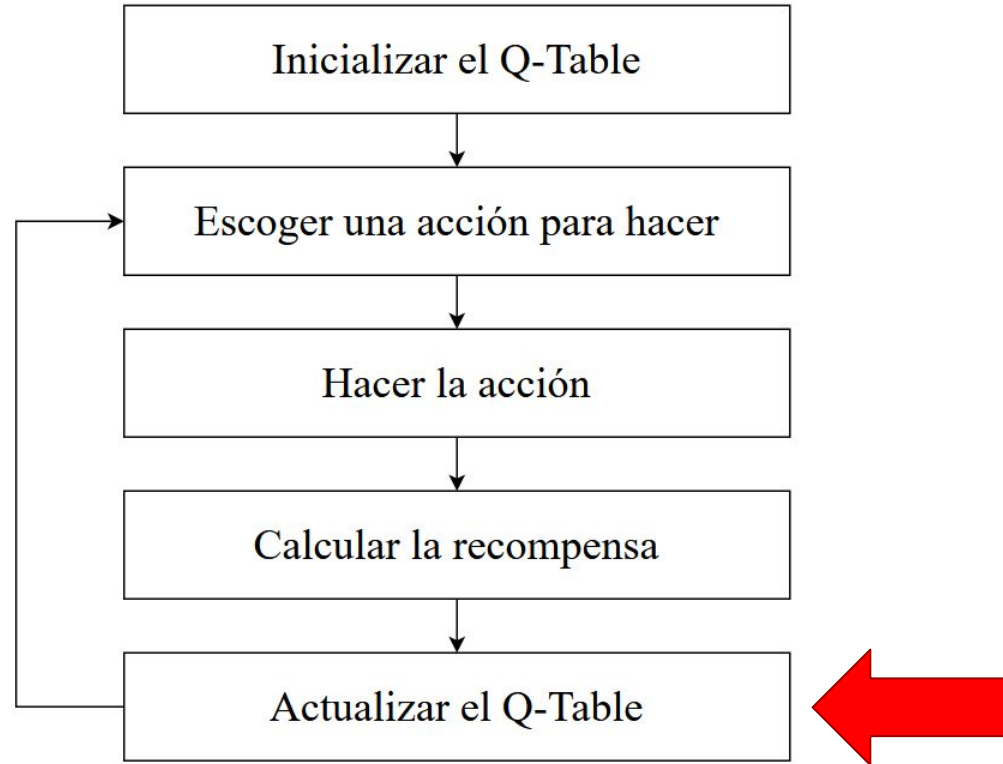
Behavior Policy

Política para **explorar el ambiente.**

Target Policy

1. Utilizada para la toma de decisiones.
2. Guardada implícitamente en el **Q Table.**

Diagrama del aprendizaje del algoritmo



Q-Learning

¿Cuál es un **gran problema**?

¡Hay que **guardar la tabla en memoria!**

Entre más crece la complejidad de nuestro problema también lo hace la tabla.

Deep Q-Learning

Playing Atari with Deep Reinforcement Learning

Volodymyr Mnih Koray Kavukcuoglu David Silver Alex Graves Ioannis Antonoglou

Daan Wierstra Martin Riedmiller

DeepMind Technologies

`{vlad,koray,david,alex.graves,ioannis,daan,martin.riedmiller} @ deepmind.com`

Deep Q-Learning

Playing Atari with Deep Reinforcement Learning

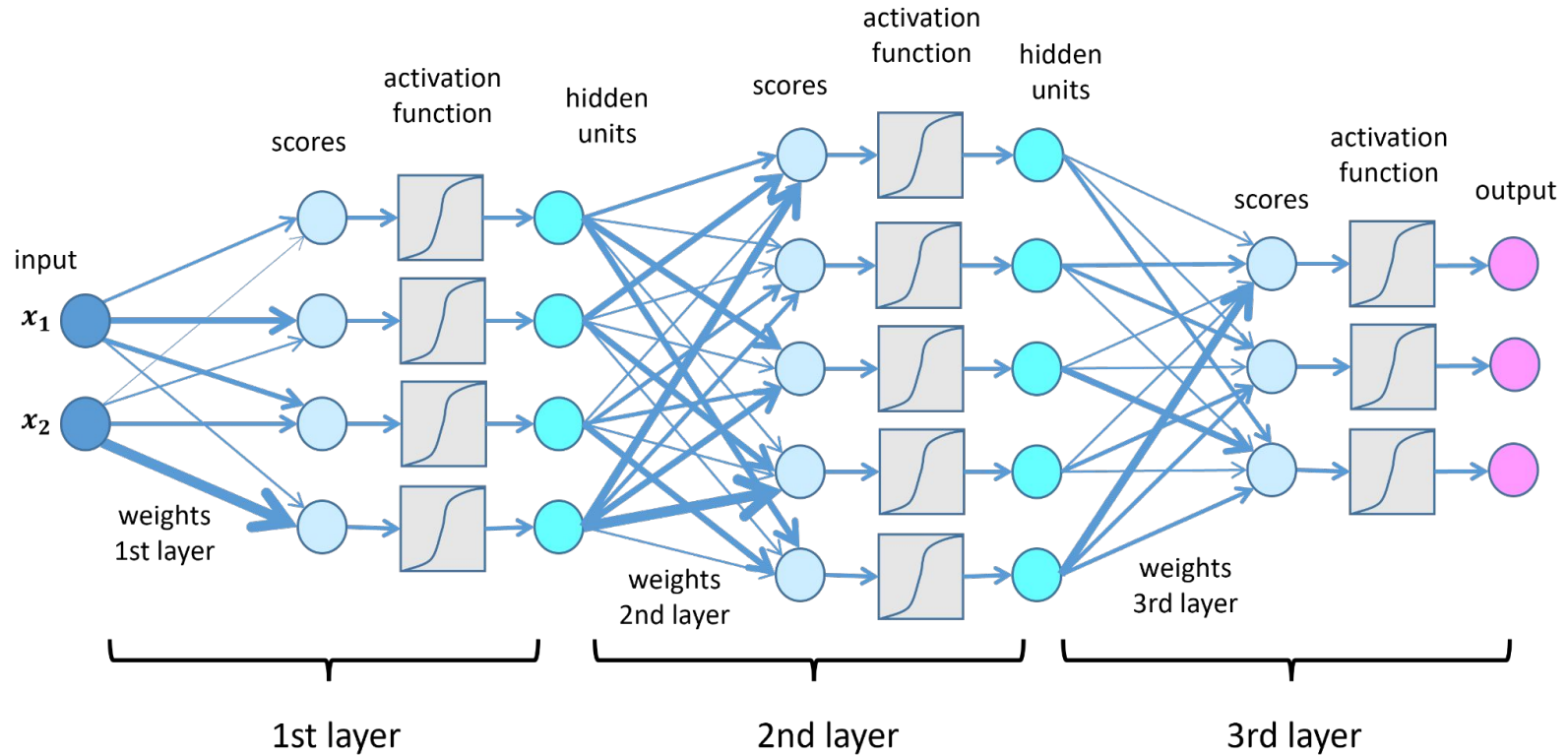
Volodymyr Mnih Koray Kavukcuoglu David Silver Alex Graves Ioannis Antonoglou

Daan Wierstra Martin Riedmiller

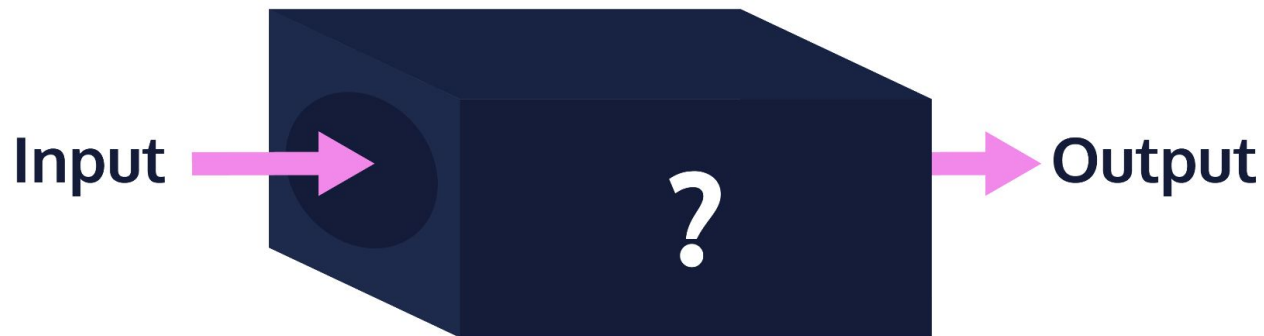
DeepMind Technologies

`{vlad,koray,david,alex.graves,ioannis,daan,martin.riedmiller} @ deepmind.com`

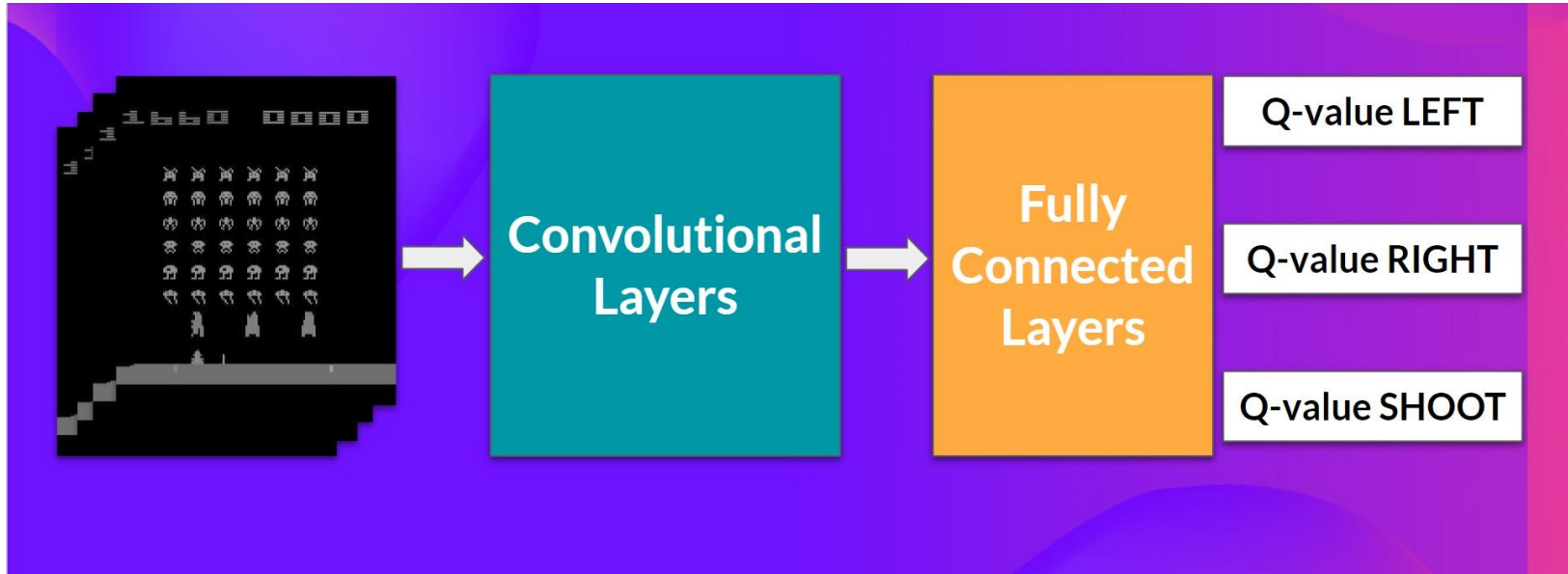
¿Deep?



¿Deep?



¿Deep?



¡Exploren otros algoritmos!

SARSA

Proximal Policy
Optimization

Actor-Critic

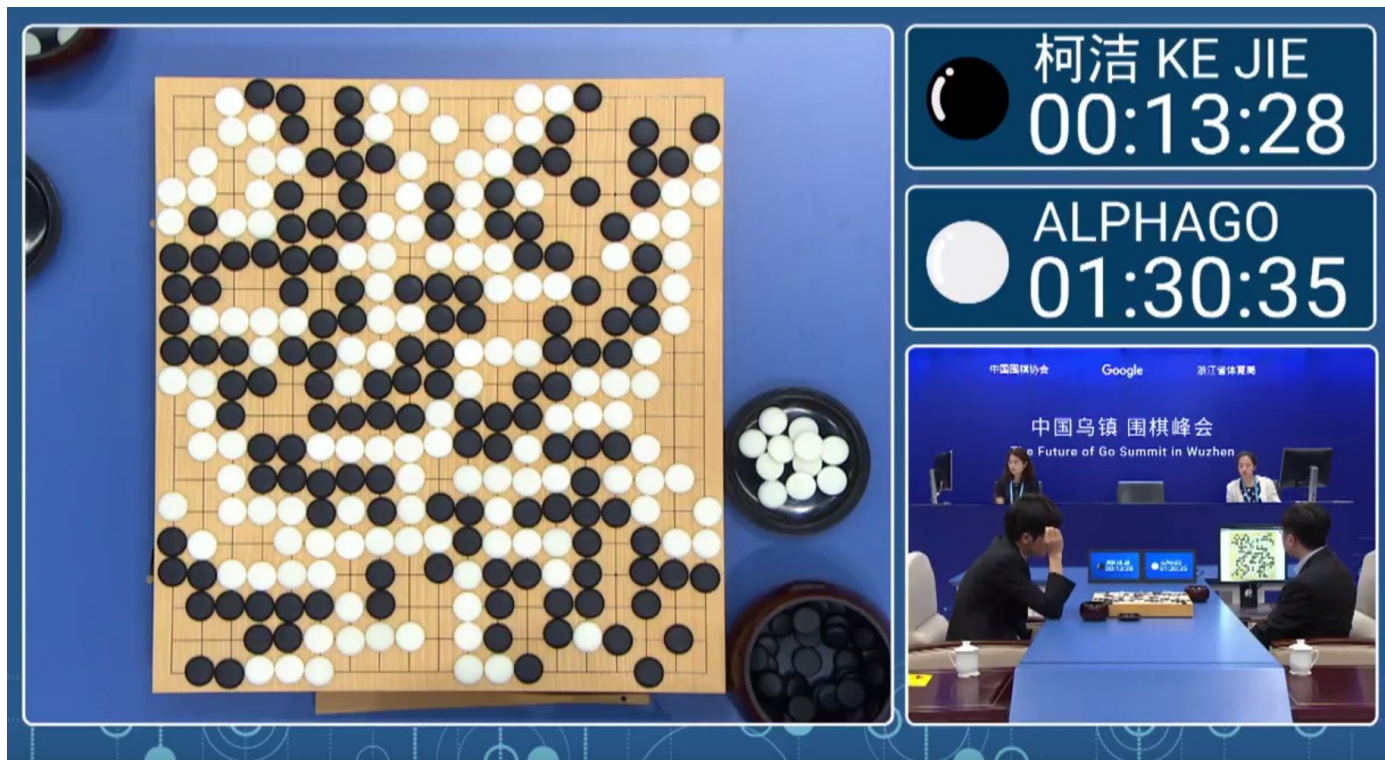
REINFORCE

Policy Gradient

03

Aplicaciones

Google DeepMind - AlphaGo



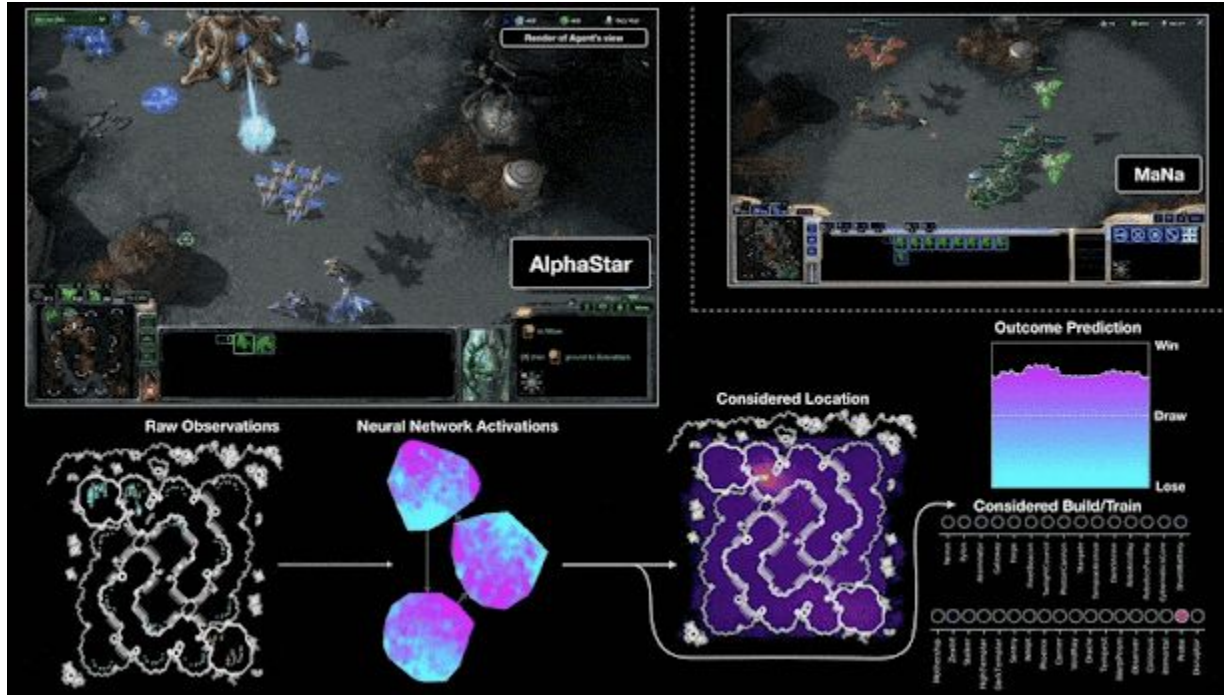
“I thought AlphaGo was based on probability calculation and that it was **merely a machine**. But when I saw this move, I changed my mind. **Surely, AlphaGo is creative.**”

—Lee Sedol



Foto de [The New Yorker](#)

Google DeepMind - AlphaStar



DeepSeek

DeepSeek's Latest Breakthrough Is Redefining AI Race

DeepSeek's R1 Is Not a Sputnik Moment, But a New Chapter
in the AI Race

Microsoft, Meta CEOs defend hefty AI
spending after DeepSeek stuns tech world

Nvidia Stock May Fall As DeepSeek's 'Amazing' AI
Model Disrupts OpenAI

DeepSeek

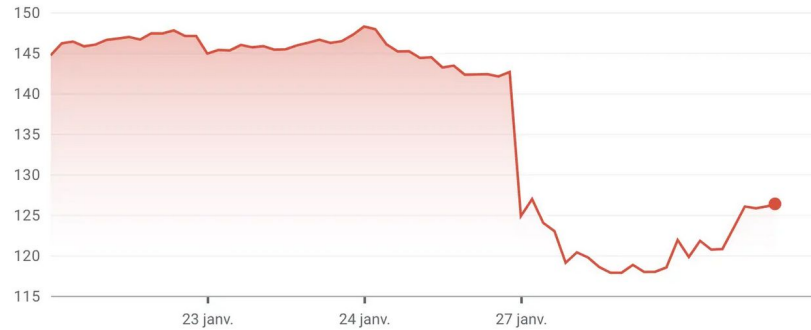
ACCUEIL > NVDA · NASDAQ

Nvidia

126,35 \$ ↓ 12,67 % -18,33 5 j

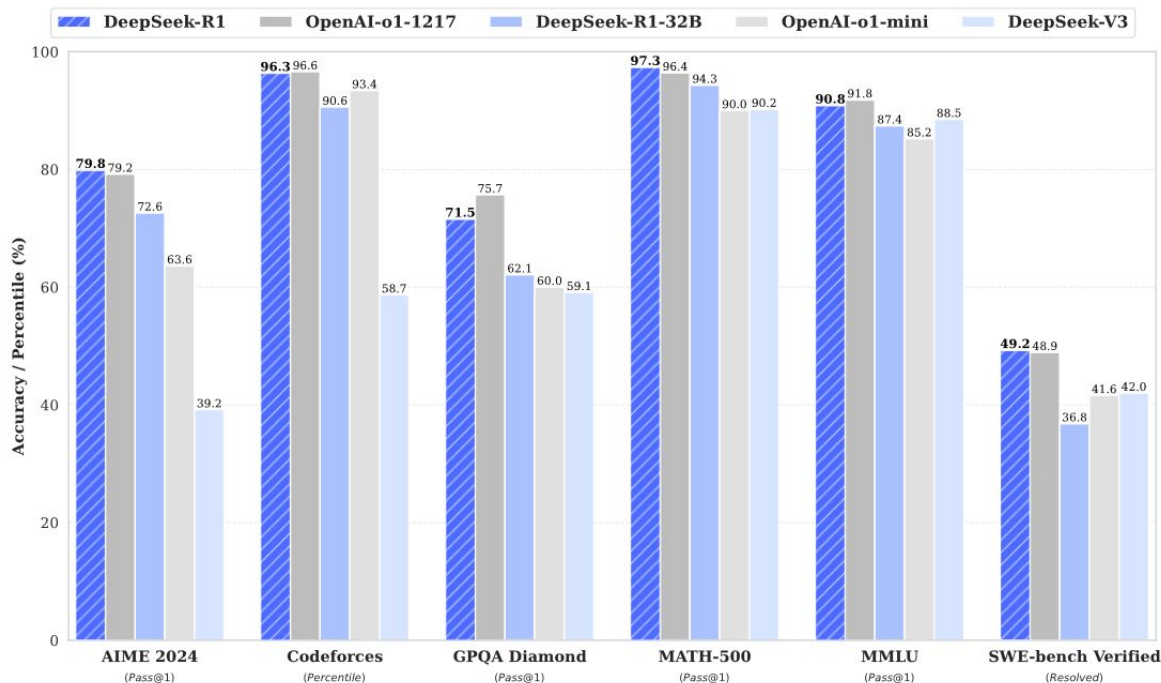
28 janv., 13:51:16 UTC-5 · USD · NASDAQ · Clause de non-responsabilité

1 j 5 j 1 m 6 m YTD 1 a 5 a MAX







DeepSeek-R1

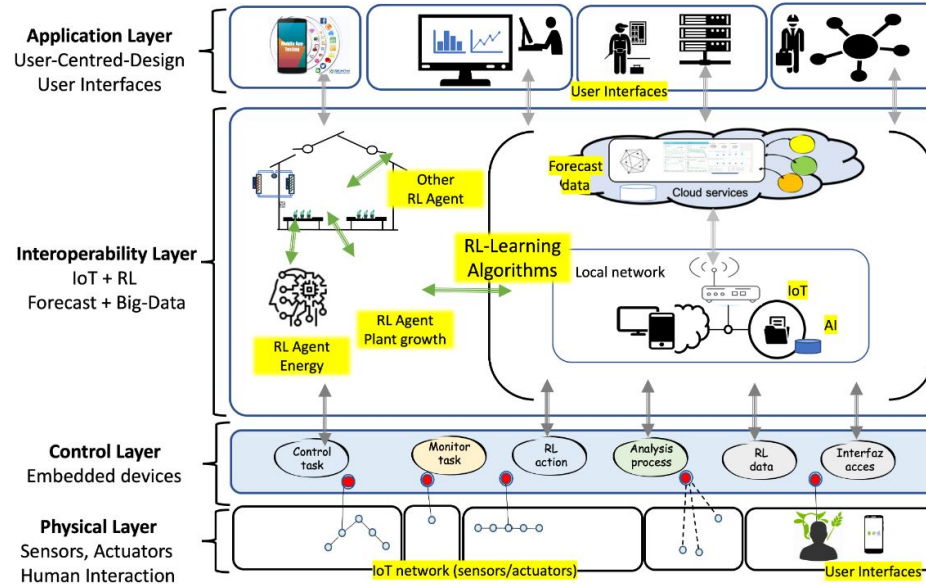
DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning



Article

Enhancing Greenhouse Efficiency: Integrating IoT and Reinforcement Learning for Optimized Climate Control

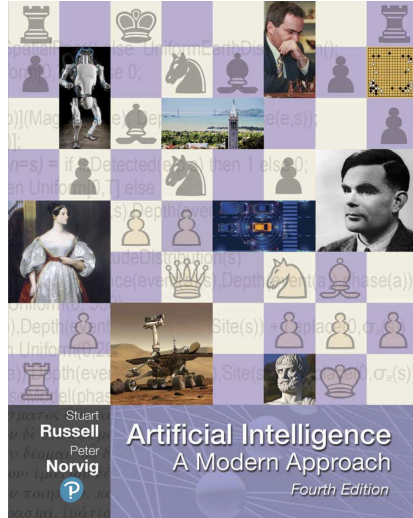
Manuel Platero-Horcajadas ¹, Sofia Pardo-Pina ², José-María Cámara-Zapata ², José-Antonio Brenes-Carranza ³
and Francisco-Javier Ferrández-Pastor ^{1,*}



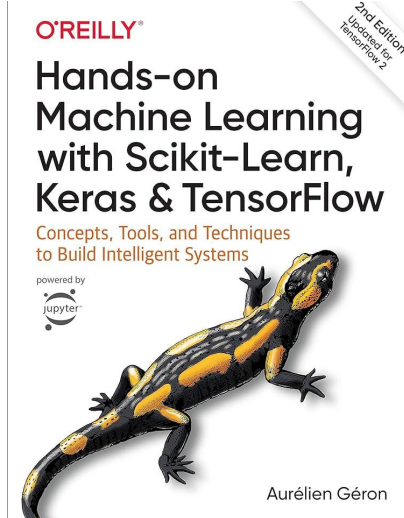
04

Recomendaciones

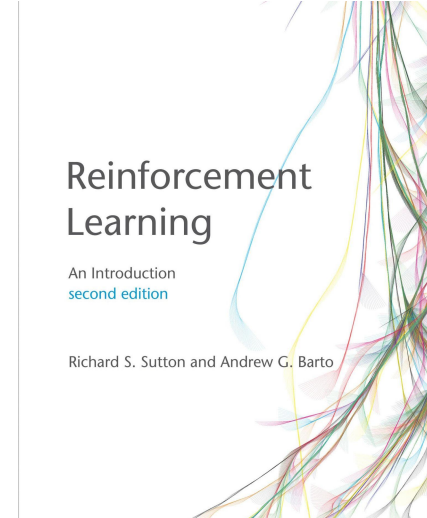
Libros



Artificial Intelligence
A Modern Approach



Hands-on Machine
Learning with
Scikit-Learn, Keras &
TensorFlow



Reinforcement Learning
An Introduction

Cursos

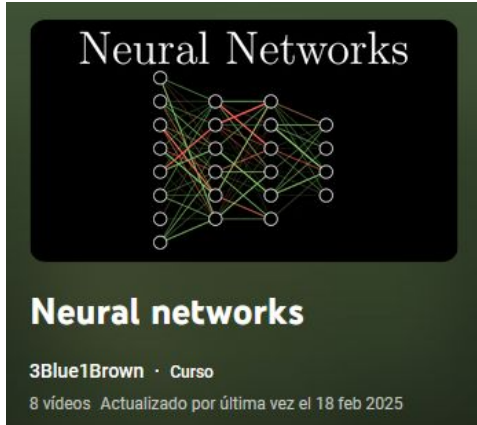


Google DeepMind
Reinforcement Learning Course

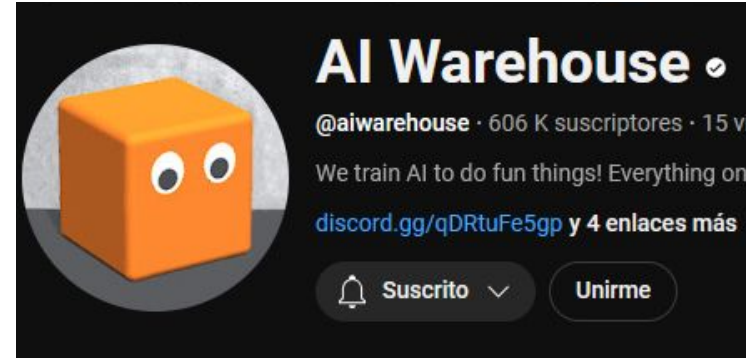


The Hugging Face
Deep Reinforcement Learning
Course

Entretenimiento



Neural networks
3Blue1Brown



AI Warehouse

Bibliografía

- DeepSeek-Ai, Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., Zhang, X., Yu, X., Wu, Y., Wu, Z. F., Gou, Z., Shao, Z., Li, Z., Gao, Z., . . . Zhang, Z. (2025). DeepSeek-R1: Incentivizing reasoning capability in LLMs via Reinforcement Learning. *arXiv (Cornell University)*. <https://doi.org/10.48550/arxiv.2501.12948>
- Google DeepMind. (n.d.). *AlphaGo*. <https://deepmind.google/research/breakthroughs/alphago/>
- Google DeepMind, & Silver, D. (2015, May 13). *RL Course by David Silver - Lecture 1: Introduction to Reinforcement Learning* [Video]. YouTube. <https://www.youtube.com/watch?v=2pWv7GOvuf0>
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing Atari with Deep Reinforcement Learning. *arXiv (Cornell University)*. <https://doi.org/10.48550/arxiv.1312.5602>
- Platero-Horcajadas, M., Pardo-Pina, S., Cámara-Zapata, J., Brenes-Carranza, J., & Ferrández-Pastor, F. (2024). Enhancing greenhouse efficiency: Integrating IoT and reinforcement learning for optimized climate control. *Sensors*, 24(24), 8109. <https://doi.org/10.3390/s24248109>

Bibliografía

Russel, S., & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach* (4th ed.). Prentice Hall. <http://aima.cs.berkeley.edu/>

Simonini, T., Sanseviero, O., & Paul, S. (2023). *The Hugging Face Deep Reinforcement Learning Class*. GitHub.
<https://github.com/huggingface/deep-rl-class>

Sutton, R., & Barto, A. (2020). *Reinforcement Learning: An Introduction* (2nd ed.). The MIT Press.
<http://www.incompleteideas.net/book/RLbook2020.pdf>

The Farama Foundation & OpenAI. (n.d.). *Gymnasium Documentation*. Gymnasium. <https://gymnasium.farama.org/index.html>

Vinyals, O., Babuschkin, I., Chung, J., Mathieu, M., Jaderberg, M., Czarnecki, W., Dudzik, A., Huang, A., Georgiev, P., Powell, R., Ewalds, T., Horgan, D., Kroiss, M., Danihelka, I., Agapiou, J., Oh, J., Dalibard, V., Choi, D., Sifre, L., . . . Silver, D. (2019). *AlphaStar: Mastering the real-time strategy game StarCraft II*. Google DeepMind.
<https://deepmind.google/discover/blog/alphastar-mastering-the-real-time-strategy-game-starcraft-ii/>

¡Muchas gracias!



¿Preguntas?

luis.solanosantamaria@ucr.ac.cr

Credits: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon**, and infographics & images by **Freepik**

Please keep this slide for attribution



Centro de Investigaciones en Tecnologías
de la Información y Comunicación