

# Aprender a Actuar

2.2 Introducción a Reinforcement Learning

Jorge Vásquez

# Agenda

- Aproximación de RL
- Conceptos Básicos
- Loop cerrado de sensorización y acción
- Ejemplos de loops cerrados
- Limitaciones de RL
- Estimación de Estados
- Comparación con aprendizaje supervisado

# Aproximación de RL

# Aprender a Actuar

- Aprender a actuar es un objetivo por sí solo.
- Los algoritmos de RL están conducidos por un objetivo
- Envuelve todo el problema de Inteligencia Artificial en el mundo real

# ¿Cómo se crean los comportamientos?

- ❖ Psicología
- ❖ Neurociencia

*“Behavior are shaped by reinforcement, instead of free-will” –  
Skinner*

- ❖ Comportamientos recompensados se tienden a repetir
- ❖ Comportamientos castigados se tienen a extinguir

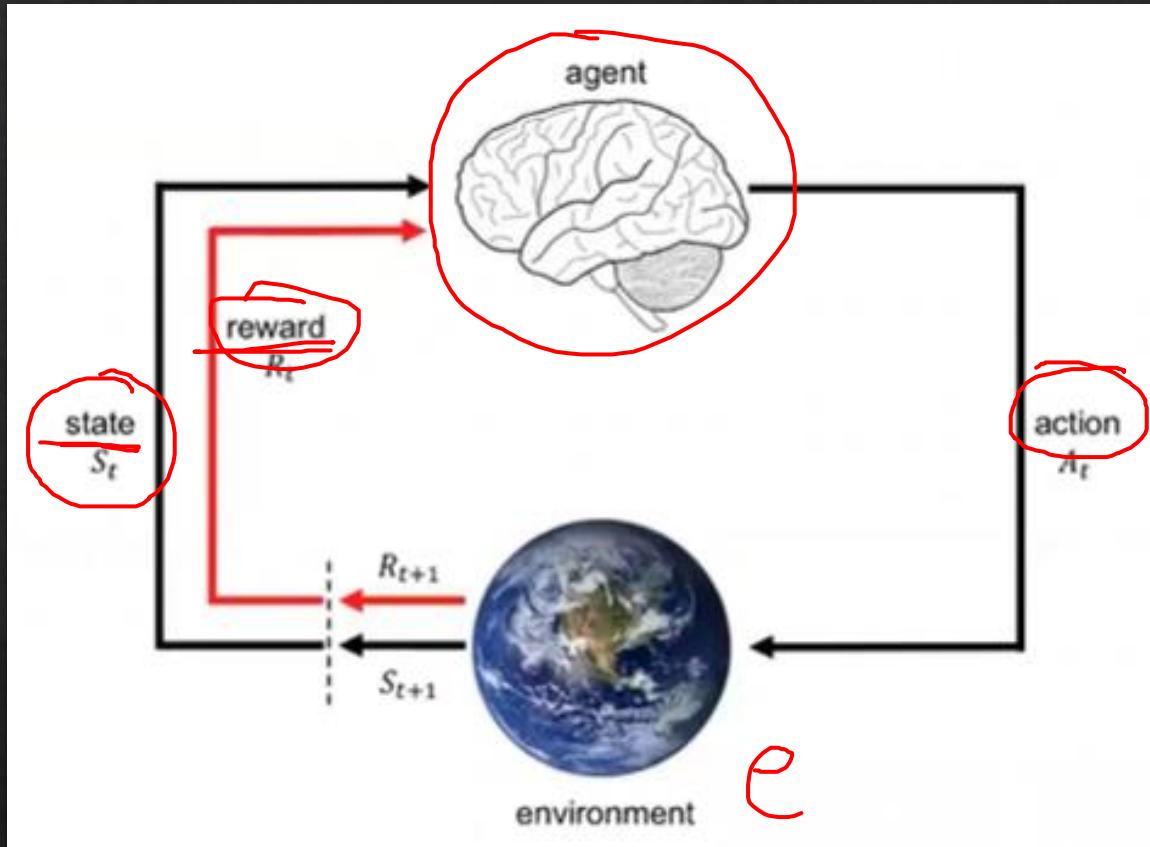
# ¿Cómo se crean los comportamientos?

- Descubrir un comportamiento desde cero
  - A través de prueba y error
  - Tabula rasa y Entorno estático
- Transferir (generalizar) comportamiento para diferentes escenarios
  - Desde un conocimiento previo, lo enriquezco con prueba y error
  - Velocidad de transferencia de comportamiento

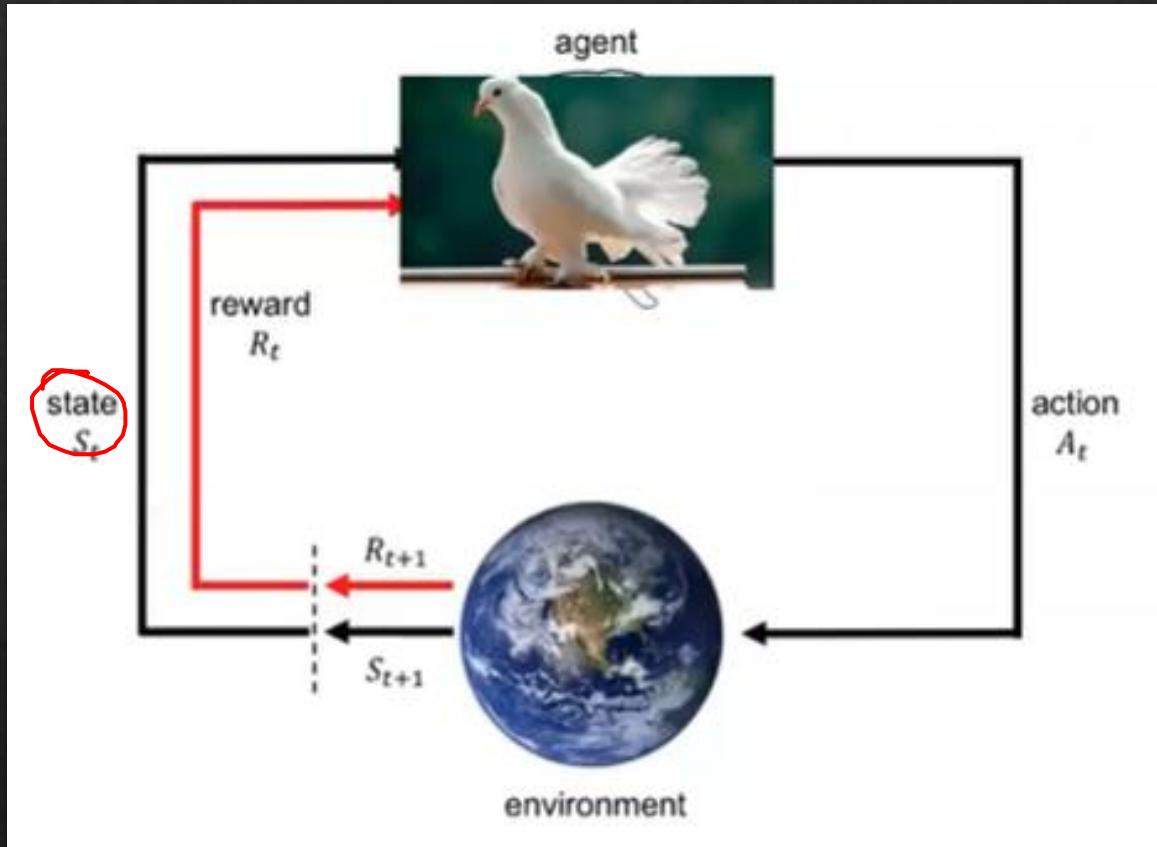
# Aproximación de RL es Prueba y Error



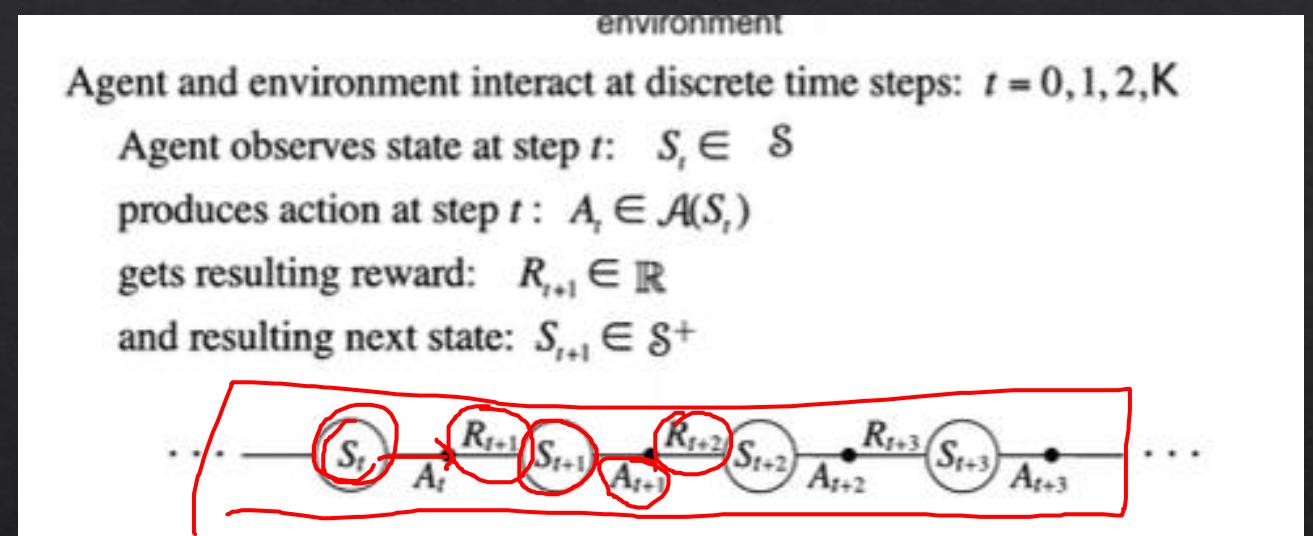
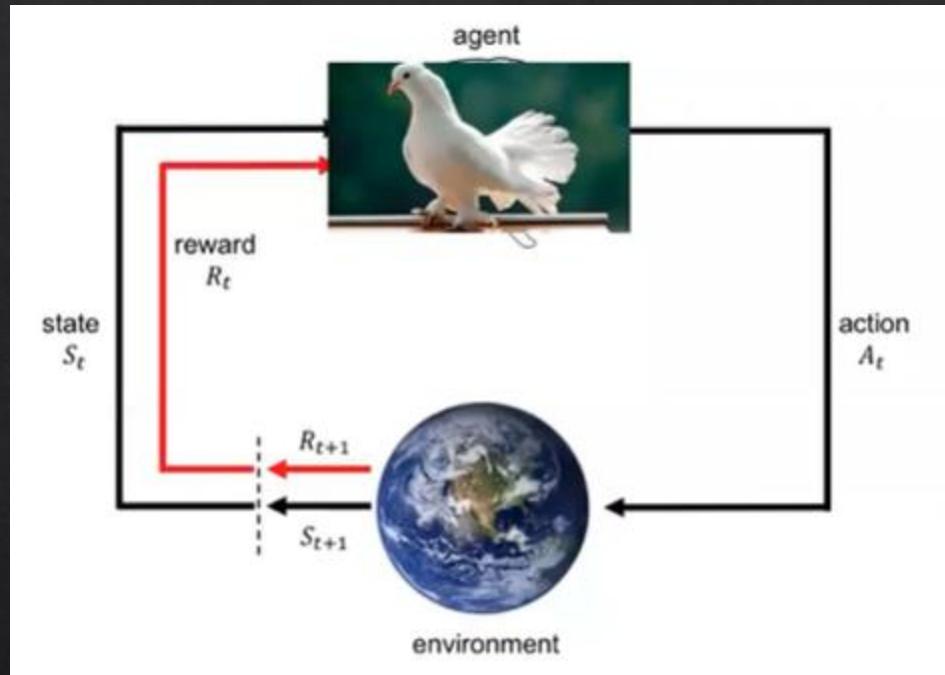
# Aproximación de RL es Prueba y Error



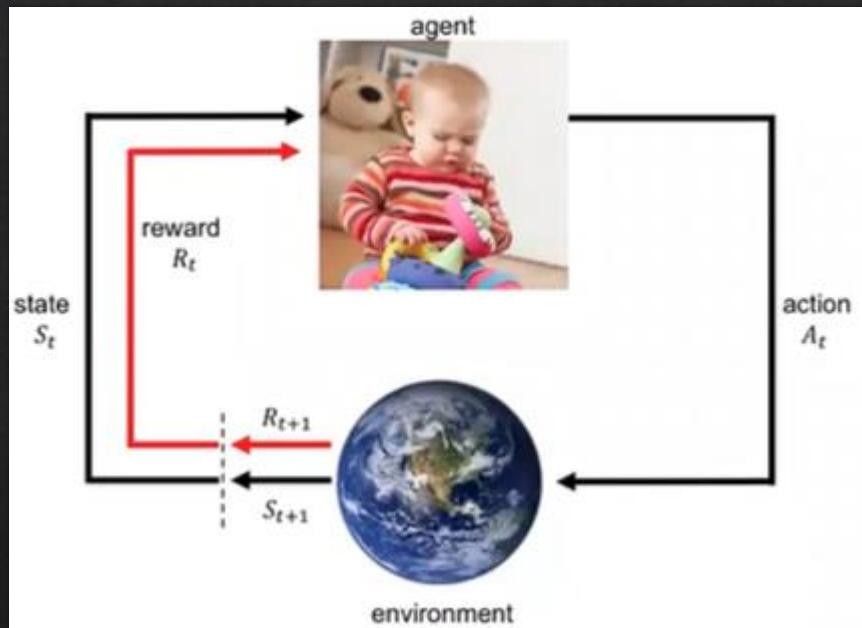
# Aproximación de RL es Prueba y Error



# Aproximación de RL : Prueba y Error



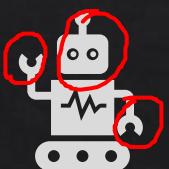
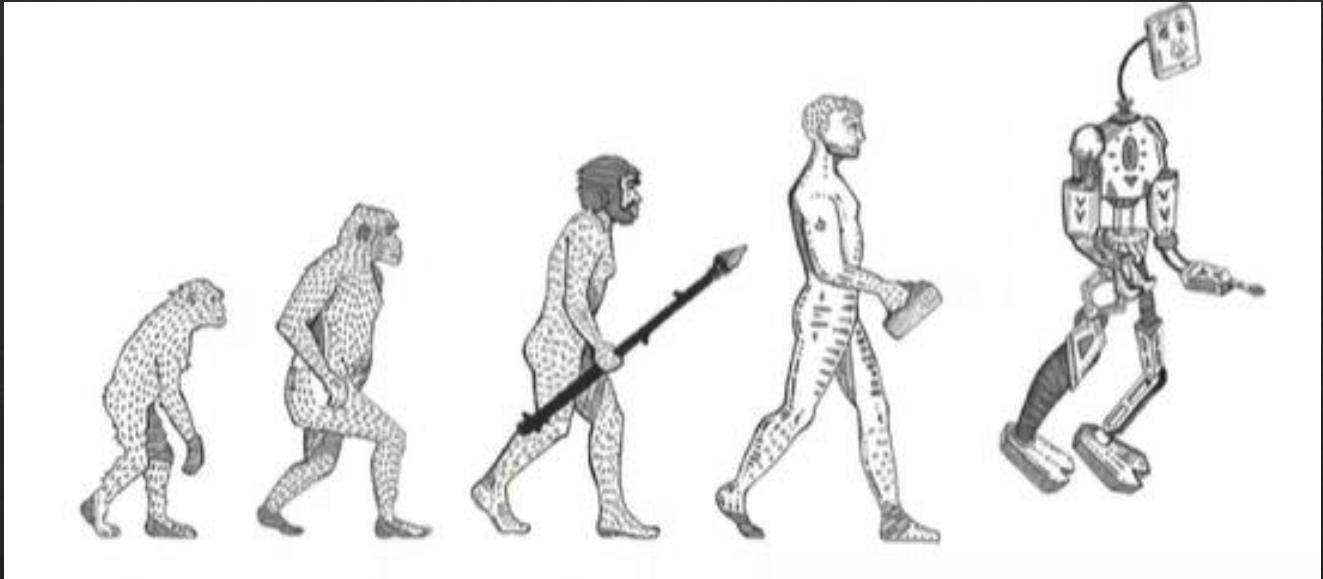
Que pasa cuando las recompensas no es comida,  
sino...



# Conceptos Básicos en RL

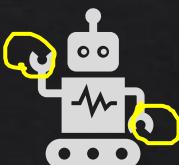
# RL: Conceptos Básicos

- ❖ **Agente:** Es una entidad equipada con:
  - ❖ Sensores, para sentir su entorno
  - ❖ End-effectors, para actuar con su entorno y
  - ❖ Objetivos, que quiere lograr



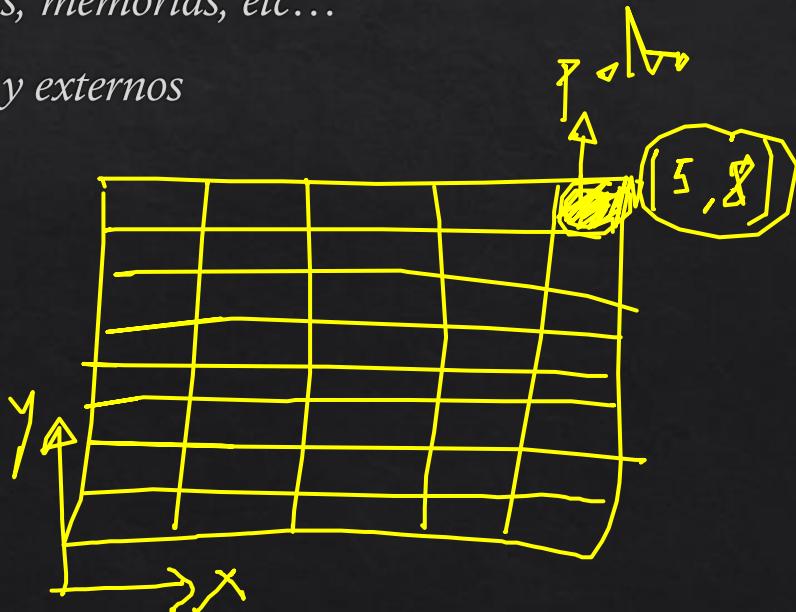
# RL: Conceptos Básicos

- ◊ *Acciones*, son usados por el agente para interactuar con su entorno
  - ◊ Tocar una canción
  - ◊ Bajar las luces
  - ◊ Subir el volumen
  - ◊ Llamar a la abuela
  - ◊ Mostrar comerciales Ads
  - ◊ Sugerir canciones
  - ◊ Seguir derecho, doblar, frenar, etc.
  - ◊ Torques de un robot
  - ◊ Traslación de un gripper
  - ◊ Etc....



# RL: Conceptos Básicos

- ❖ **Estados (States)**, son una representación del mundo actual o del entorno.
  - ❖ Captura cualquier información disponible que tomó el agente al paso “ $t$ ” sobre su entorno.
  - ❖ Incluye observaciones inmediatas y altamente procesadas
  - ❖ Estructuras lógicas que se van construyendo a través del tiempo desde secuencias de sensaciones, memorias, etc...
  - ❖ Son internos y externos



# RL: Conceptos Básicos

- ❖ ***Observaciones***, es una sensación, son los inputs del sensor
  - ❖ Imágenes
  - ❖ Señales táctiles, contacto
  - ❖ Formas de onda
  - ❖ Señales eléctricas
  - ❖ Etc...



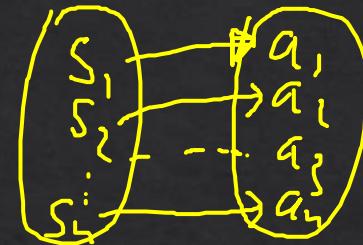
$$\pi: S \rightarrow a$$

# RL: Conceptos Básicos

$$\pi: S \rightarrow a$$

◆ **Política**, una función de mapeo desde los estados a las acciones de los efectores finales (end-effectors).

$$\begin{array}{c} S \\ a \end{array}$$



$$\pi(a | s) = \mathbb{P}[A_t = a | S_t = s]$$

$a \rightarrow, \leftarrow$

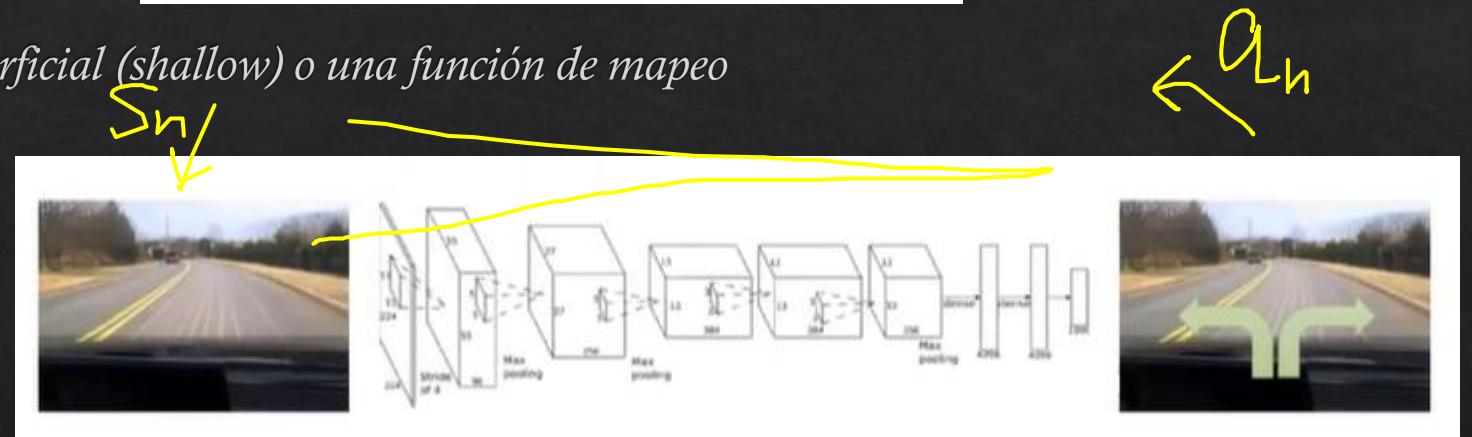
# RL: Conceptos Básicos

$s \rightarrow s'$

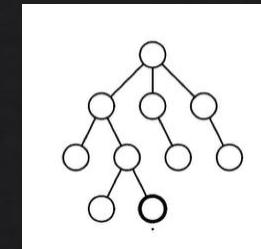
- ❖ **Política**, una función de mapeo desde los estados a las acciones de los efectores finales (end-effectors).

$$\pi(a | s) = \mathbb{P}[A_t = a | S_t = s]$$

- ❖ Puede ser una función de mapeo superficial (shallow) o una función de mapeo profunda (deep)

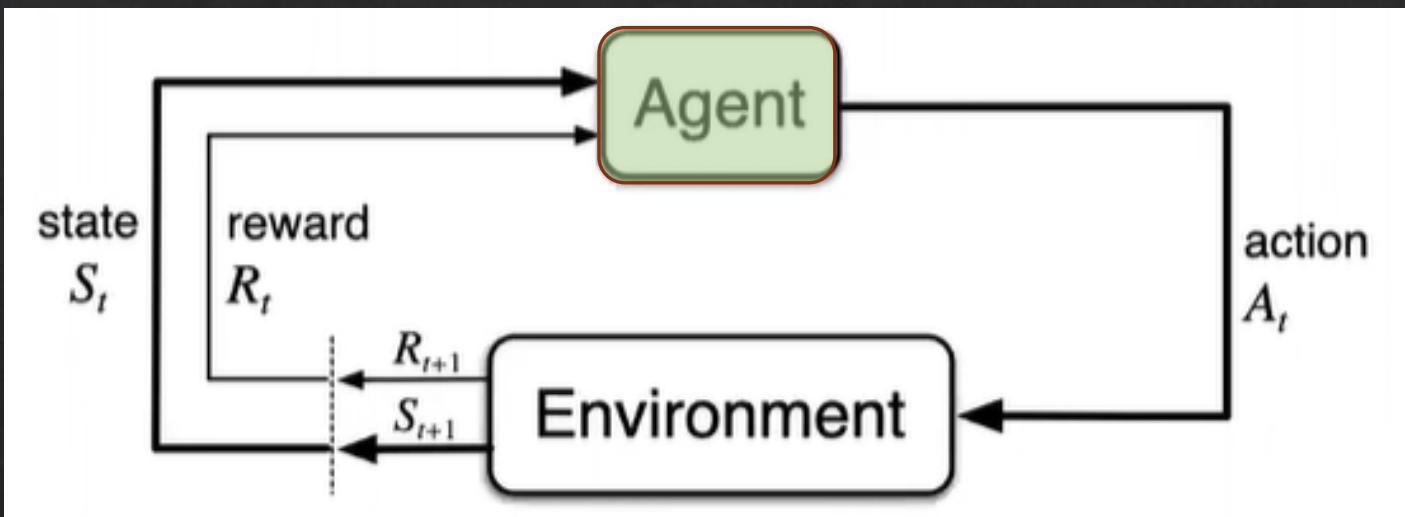


- ❖ O puede ser como un árbol de decisión



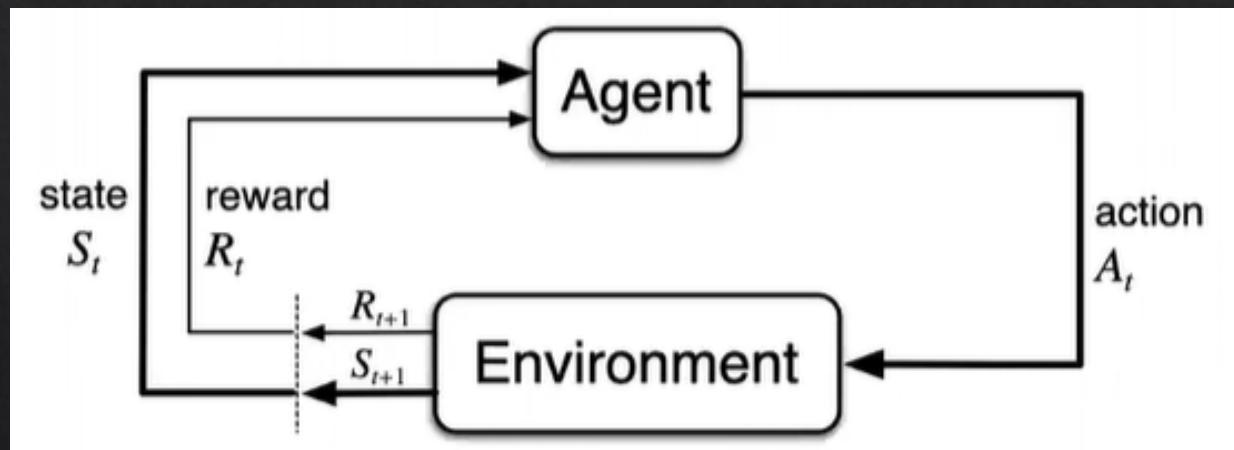
*Loop Cerrado de Sensorización y Acción*

# *Loop Cerrado de Sensorización y Acción*



# *Loop Cerrado de Sensorización y Acción*

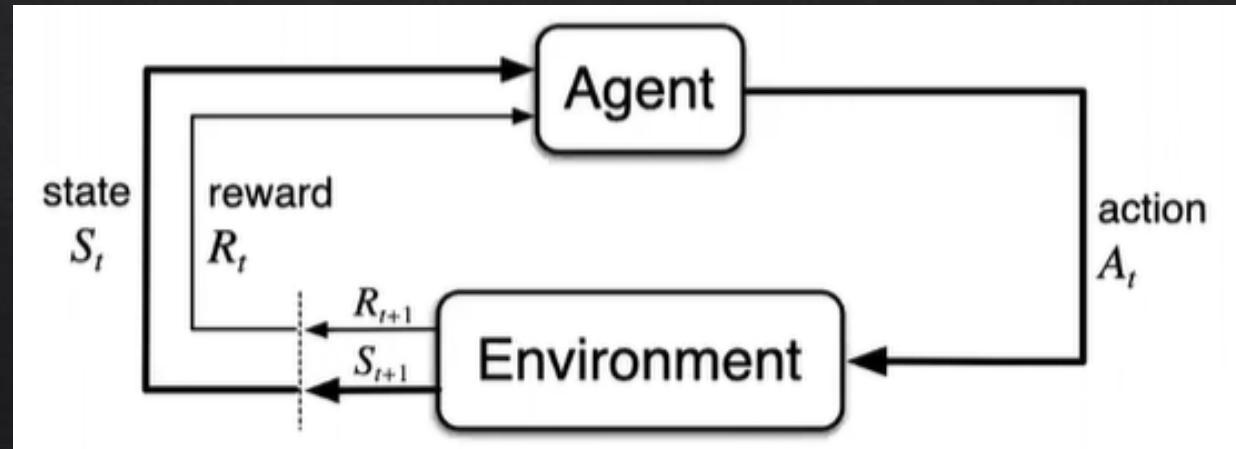
- ❖ Sensorización es siempre imperfecta. Nuestras habilidades motoras son gracias a una sensorización continua con algo grado de actualización (servoing). El loop de sensorización acción es extremadamente rápido.



# *Loop Cerrado de Sensorización y Acción*

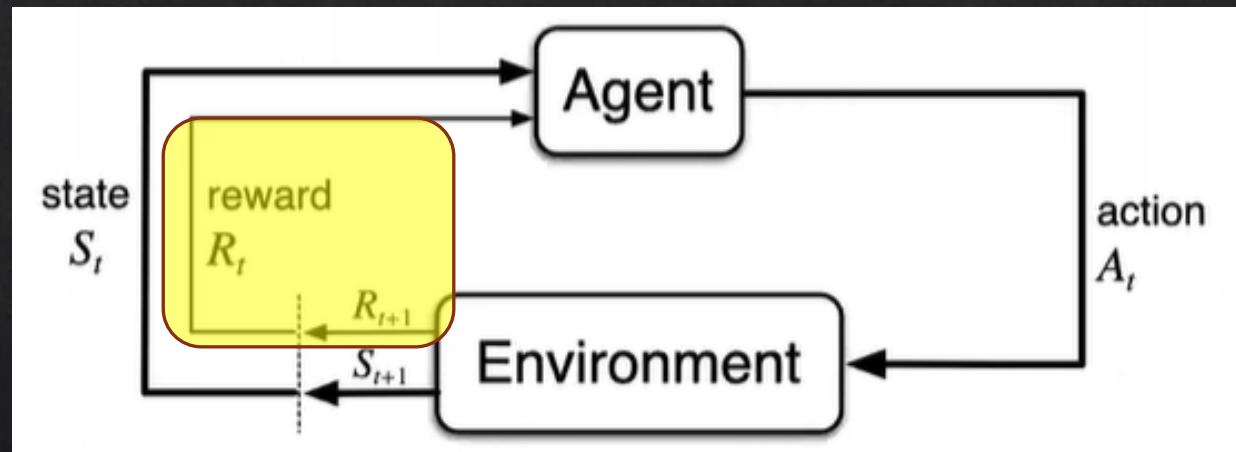
- ❖ *Ejemplo*

- ❖ *Imagina un agente que quiere tomar un objeto y tiene una política que predice como las acciones deberían ser en los próximos 2 segundos.*
- ❖ *Si apagaremos los sensores, se ejecutarían las acciones predeterminadas, fallaría en los próximos segundos*



# Loop Cerrado

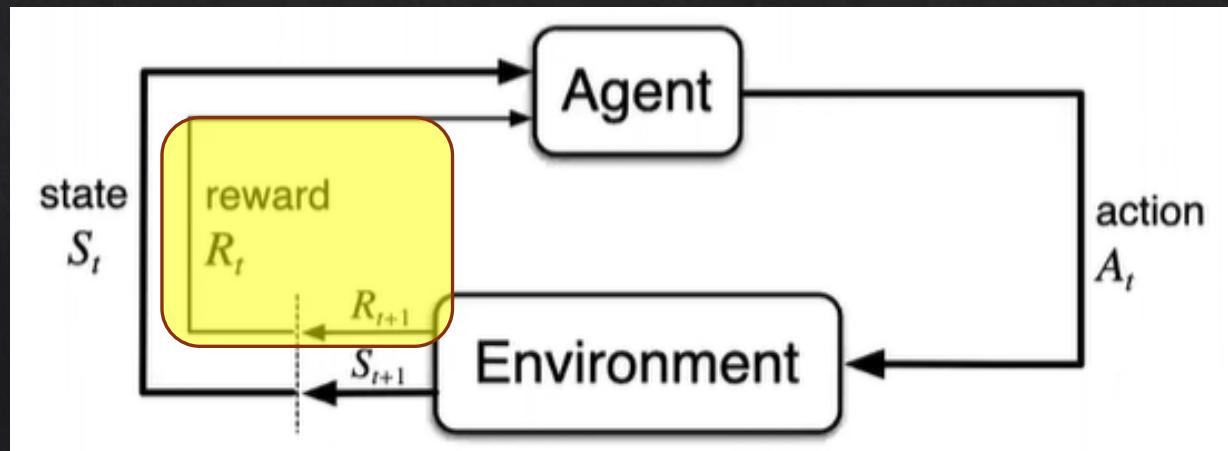
- ❖ **Recompensas (Rewards)**, son valores escalares entregados al agente para indicar si los objetivos han sido logrados.
  - ❖ Ejemplo. 1 si se cumplió el objetivo, 0 si no, -1 por cada espacio de tiempo que no se cumple.



# Loop Cerrado

- ❖ **Recompensas (Rewards)**, son valores escalares entregados por el agente para indicar si los objetivos han sido logrados.
  - ❖ Recompensas especifican QUE debe hacer el agente, no el COMO

0	0	0	0
0	+1	0	0
0	-1	0	0
0	0	0	0

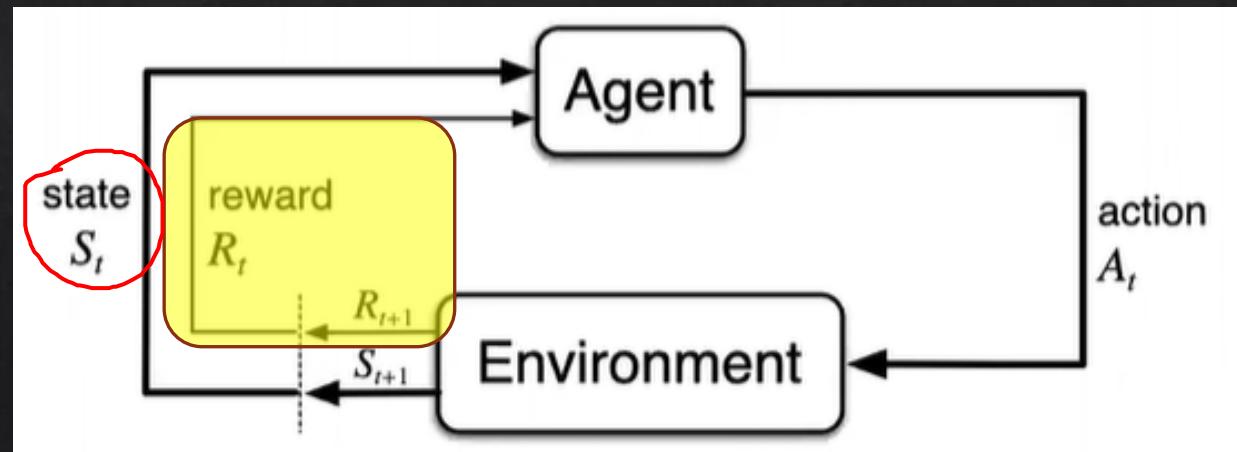


# Loop Cerrado

- ❖ **Recompensas (Rewards)**, son valores escalares entregados por el agente para indicar si los objetivos han sido logrados.
  - ❖ La forma más simple y barata de supervisión puede ser codificada matemáticamente como la maximización de una suma acumulativa de señales escalares recibidas. (rewards)

$$\max \sum_{t=0}^H R_t$$

$R_a + R_b \sqrt{V_a} -$



# Loop Cerrado

- ❖ **RETORNOS**, comportamiento de búsqueda de objetivos de un agente puede ser formalizado como el comportamiento que busca maximizar el valor esperado de la suma acumulativa de retornos.

❖ Potencialmente se descuenta el tiempo

❖ Queremos maximizar retornos:

$$G_t = R_{t+1} + R_{t+2} + \dots + R_T$$

Resta  
Resta  
Resta  
Inmediata

Resta  
Resta  
Resta  
Futura

# Ejemplos del Loop Cerrado

- *Juego de BackGammon*

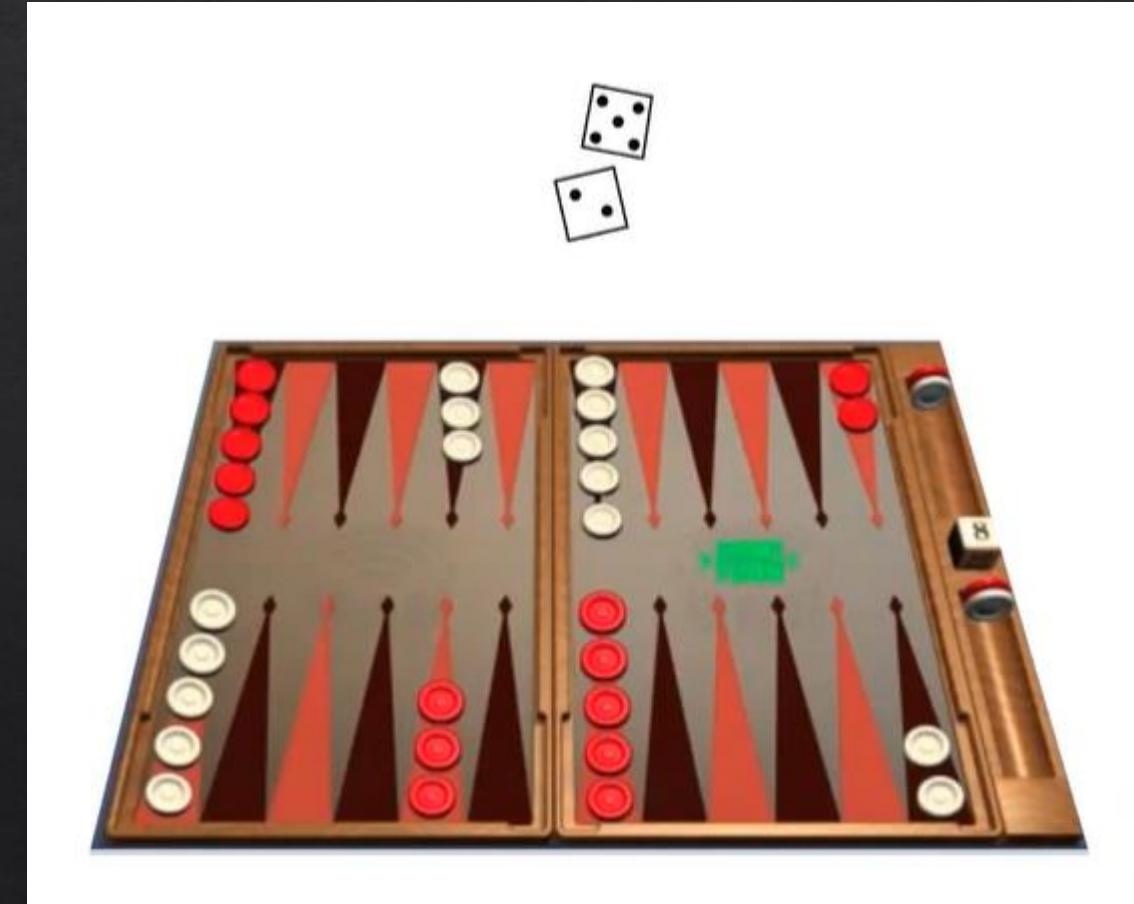
- *Estados: configuraciones del tablero*

*(~1020)*

- *Acciones: movimientos*

- *Rewards*

- *Ganar: +1*
- *Perder: -1*
- *Else: 0*



# Ejemplos del Loop Cerrado

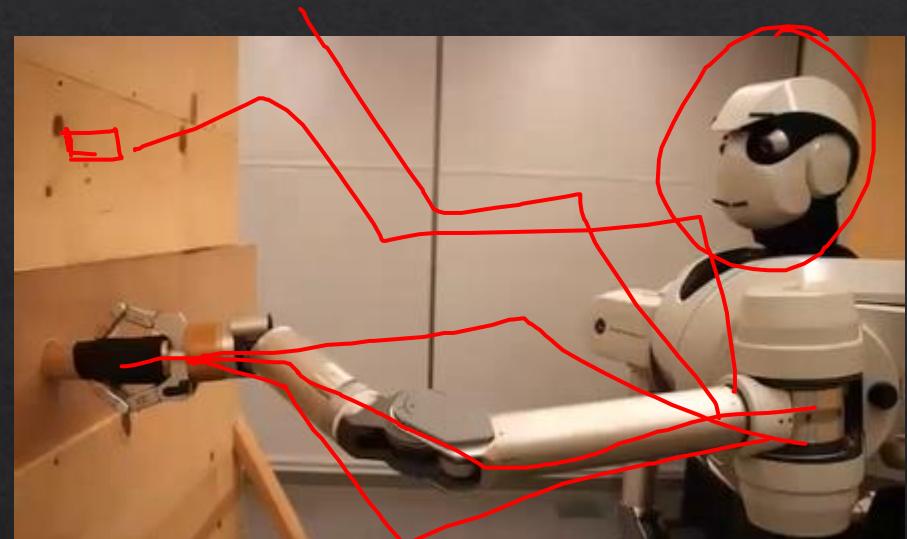
- *Aprender a Manejar*
  - *Estados:* Tráfico, clima, tiempo , hora, ..
  - *Acciones:* Mover el manubrio, frenar, acelerar, etc..
  - *Rewards:*
    - Llegar a un punto a con un tiempo limite  $(:+1)$
    - Chocar:  $-100$
    - Tocar la bocina:  $-1$



# Ejemplos del Loop Cerrado

- *Clavija en el agujero*

- Estados: configuraciones del brazo robótico
- Acciones: Torques en la articulación, por ej.
- Rewards:
  - Penalizar movimientos erráticos
  - Alcanzar una posición objetivo



# Ejemplos del Loop Cerrado

- **Clavija en el agujero**
  - **Estados:** configuraciones del brazo robótico
  - **Acciones:** torques en la articulación por ej.
  - **Rewards:**
    - Penalizar movimientos erráticos
    - Alcanzar una posición objetivo



Generalización

Ubicuidad

# El Mundo Real (Dinámica)

$S_{t-1}$   
 $a_{t-1}$   
 $R^t$   
 $S_t$

- La **Dinámica** codifica los resultados de las acciones de un agente

→ Cómo los estados y recompensas cambian dado las acciones del agente

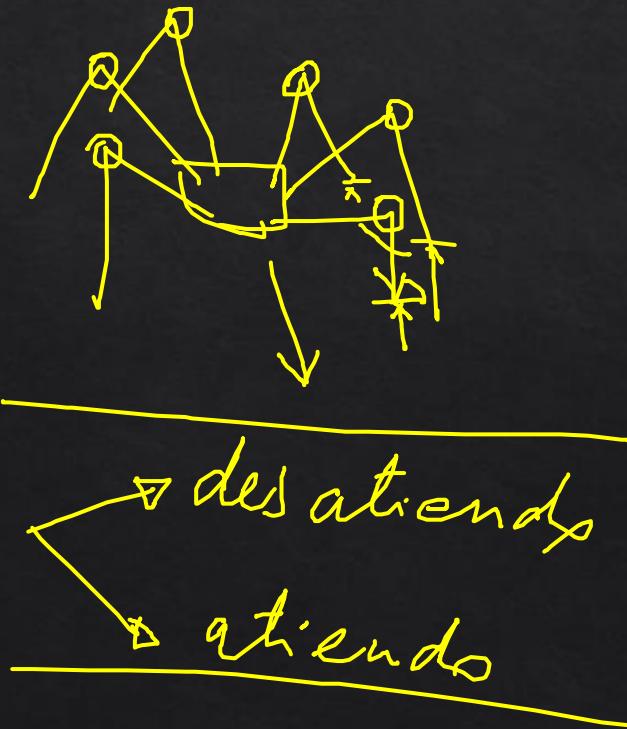
$$p(s', r | s, a) = \mathbb{P}\{S_t = s', R_t = r | S_{t-1} = s, A_{t-1} = a\}$$

- Función de transición o función del siguiente paso

$$T(s' | s, a) = p(s' | s, a) = \mathbb{P}\{S_t = s' | S_{t-1} = s, A_{t-1} = a\} = \sum_{r \in \mathbb{R}} p(s', r | s, a)$$

# El Mundo Real (Dinámica)

- “*La idea de que predecimos las consecuencias de nuestros comandos motores ha surgido como un concepto teórico importante en todos los aspectos del control sensoriomotor.*”



**Prediction Precedes Control in Motor Learning**

J. Randall Flanagan,<sup>1,\*</sup> Philipp Vetter,<sup>2</sup>  
Roland S. Johansson,<sup>3</sup> and Daniel M. Wolpert<sup>2</sup>

Procedures for details). Figure 1 shows, for a single subject, the hand path (top trace) and the grip (middle)

**Predicting the Consequences of Our Own Actions: The Role of Sensorimotor Context Estimation**

Sarah J. Blakemore, Susan J. Goodbody, and Daniel M. Wolpert  
Sobell Department of Neurophysiology, Institute of Neurology, University College London, London WC1N 3BG,

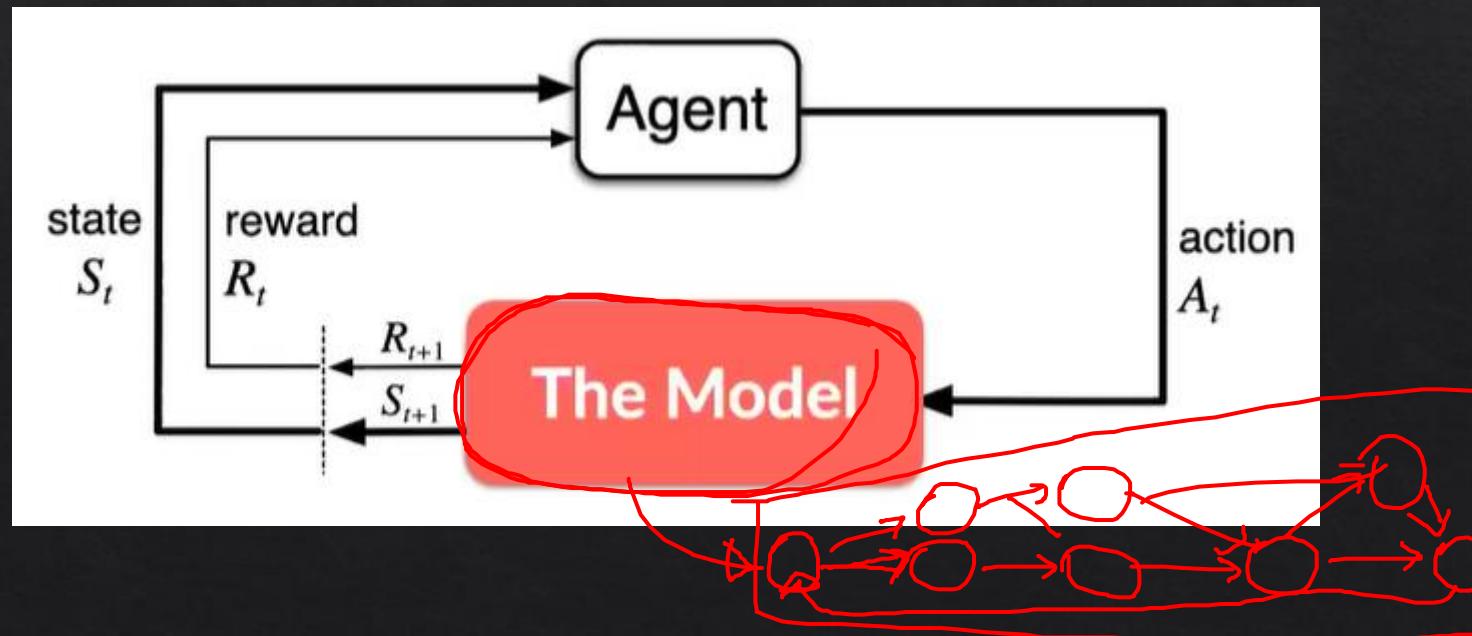
**Predictive coding in the visual cortex:  
a functional interpretation of some extra-classical receptive-field effects**

Rajesh P. N. Rao<sup>1</sup> and Diana H. Ballard<sup>2</sup>

Sergey Levine

# Planificación (Planning)

- Planificar: Desenrollar o consultar un modelo hacia adelante en el tiempo y seleccionar la mejor secuencia de acción que satisfaga un objetivo específico.
- Plan: una secuencia de objetivos.



# **Limitaciones de RL**

# Limitaciones de RL

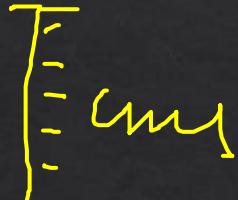
- Los problemas de comportamiento muchas veces modelar usando prueba y error o no vale la pena.
- Los agentes deben tener la posibilidad de tratar o fallar muchas veces.
  - Esto es imposible si el periodo toma mucho tiempo.
  - Esto es imposible cuando la seguridad es un problema, “contigo aprendo” es inaceptable en transporte por ejemplo.

# Limitaciones de RL

- *¿Qué otras formas de supervisión tenemos los humanos para aprender a actuar en el mundo real?*

# Formas de supervisión para aprender comportamientos

- *Aprender desde las recompensas*



- *Aprender desde las demostraciones*

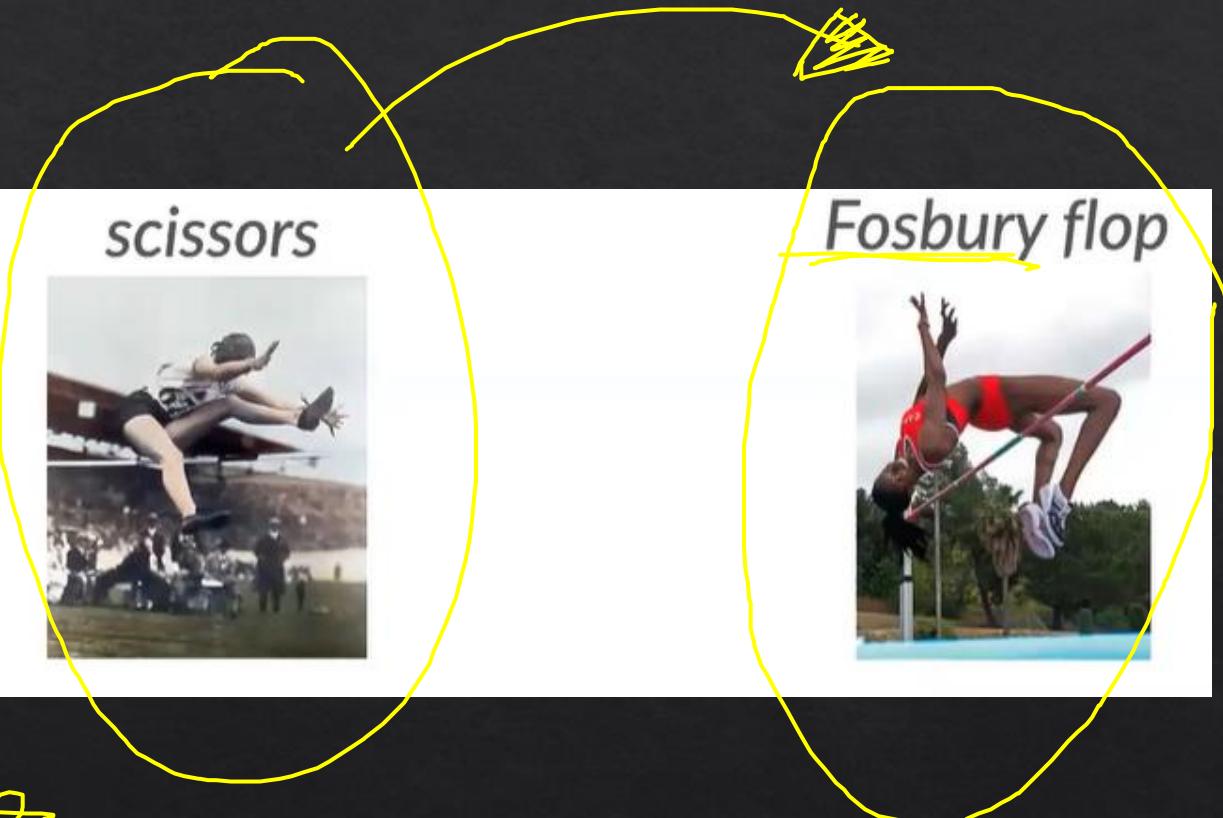
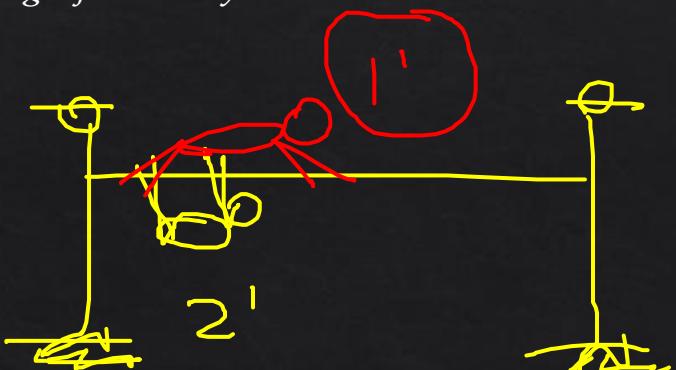


- *Aprender desde especificaciones de un comportamiento óptimo.*

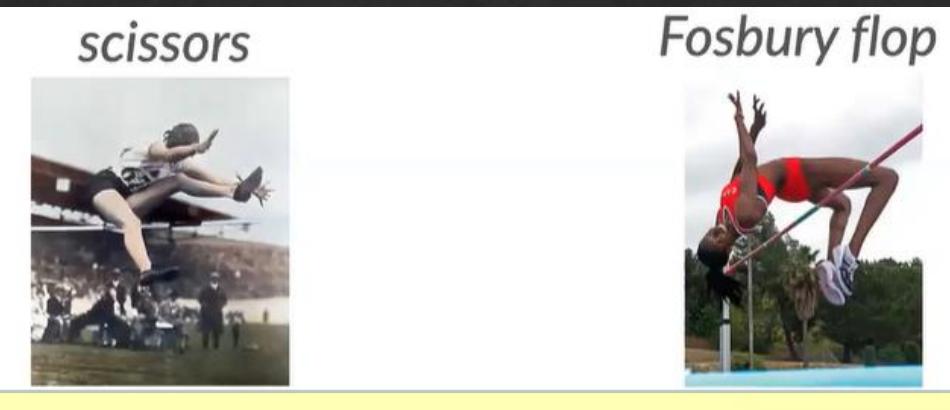


# Comportamiento: Salto Alto

- Aprendiendo por *recompensas*
  - Reward: Salta lo más alto posible (llevó años a los atletas para encontrar el comportamiento para lograr este objetivo).
- Aprendizaje por *demonstraciones*
  - Fue mucho más fácil aprender cuando alguien les enseñó la trayectoria ideal del salto.
- Aprendizaje por *especificaciones de comportamiento óptimo*
  - Para novatos, es mucho más fácil replicar un comportamiento si una guía adicional es entregada en lenguaje natural. Donde va el pie, como va la espalda, te lo entrega en un lenguaje natural y táctil. Mundo real



# Comportamiento: Salto Alto



- *Aprendiendo por recompensas*
  - Reward: salta lo mas alto posible, llevó años a los atletas para encontrar el comportamiento para lograr este objetivo
- *Aprendizaje por demostraciones*
  - Fue mucho mas fácil aprender cuando alguien les enseñó la trayectoria ideal del salto
- *Aprendizaje por especificaciones de comportamiento óptimo*
  - Un profesor que te diga donde va el pie, como va la espalda, te lo entrega en un lenguaje natural y táctil.
  - Mundo real

Feature Learning

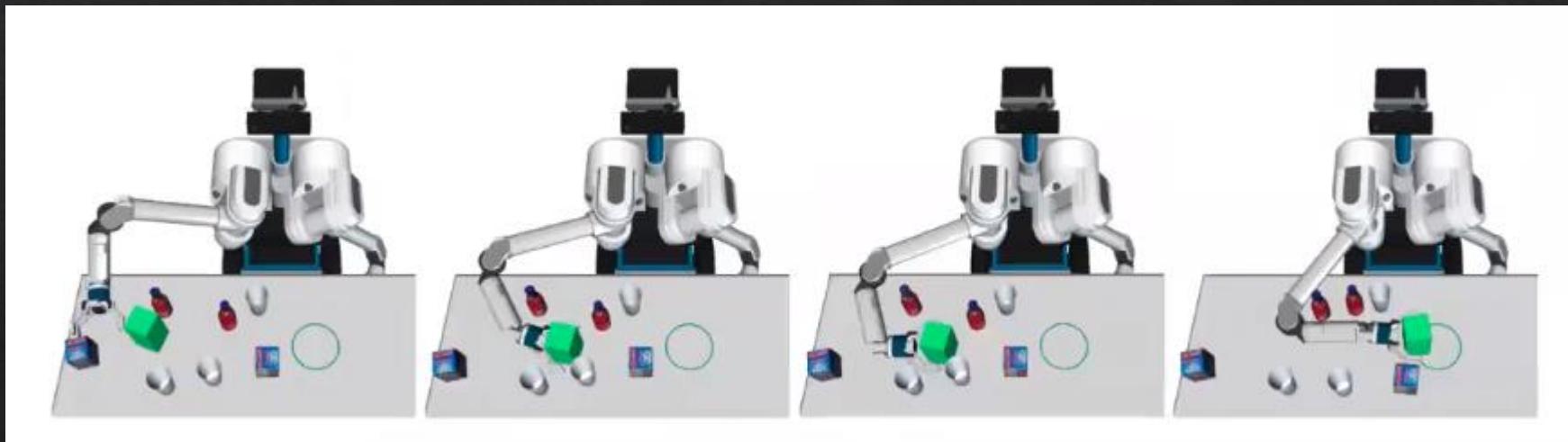
**Representation Learning, como le ayuda a RL**

---

---

# Estimación de Estados – Dos extremos

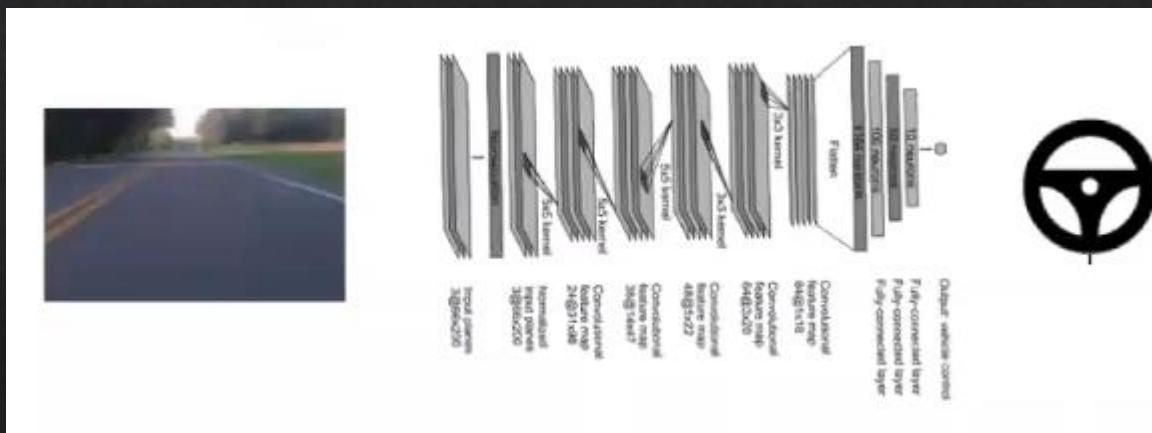
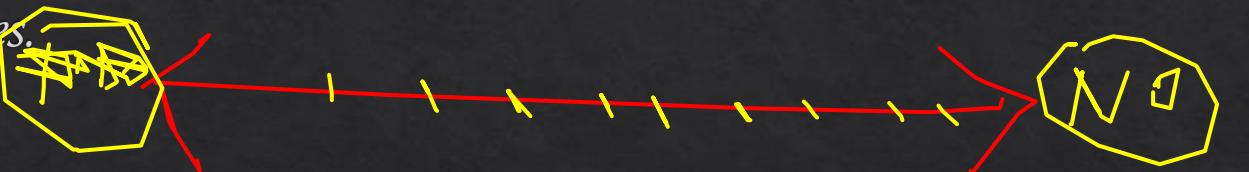
- ◊ Asumiendo que sabemos todo sobre el mundo real (ubicación de objetos, formas en 3D, propiedades físicas) y la dinámica /Usar planificadores en búsqueda de secuencias de acciones para lograr un objetivo deseado.



Rearrangement Planning vis Heuristic Search, Sidd

# Estimación de Estados – Dos extremos

- ❖ Asumiendo **NO** sabemos nada del mundo alrededor. Aprender sobre píxeles de mapas directamente a acciones mientras optimizamos para la tarea final. Ej.: no chocar y obedecer señalética, o imitar demostraciones.



End-to-End Learning for Self Driving Cars, NVIDIA

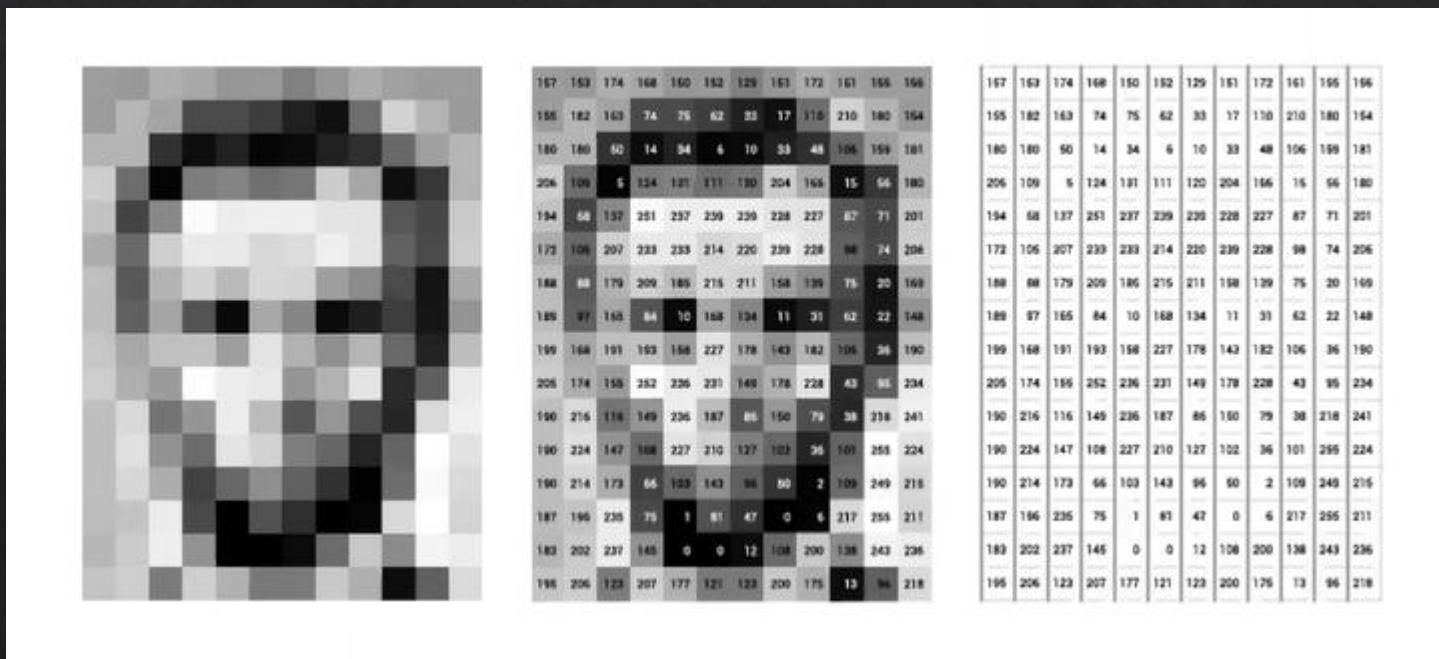
# Estimación de Estados – Dos extremos

- ❖ *Se necesita mucho conocimiento para transferir observaciones a estados.*



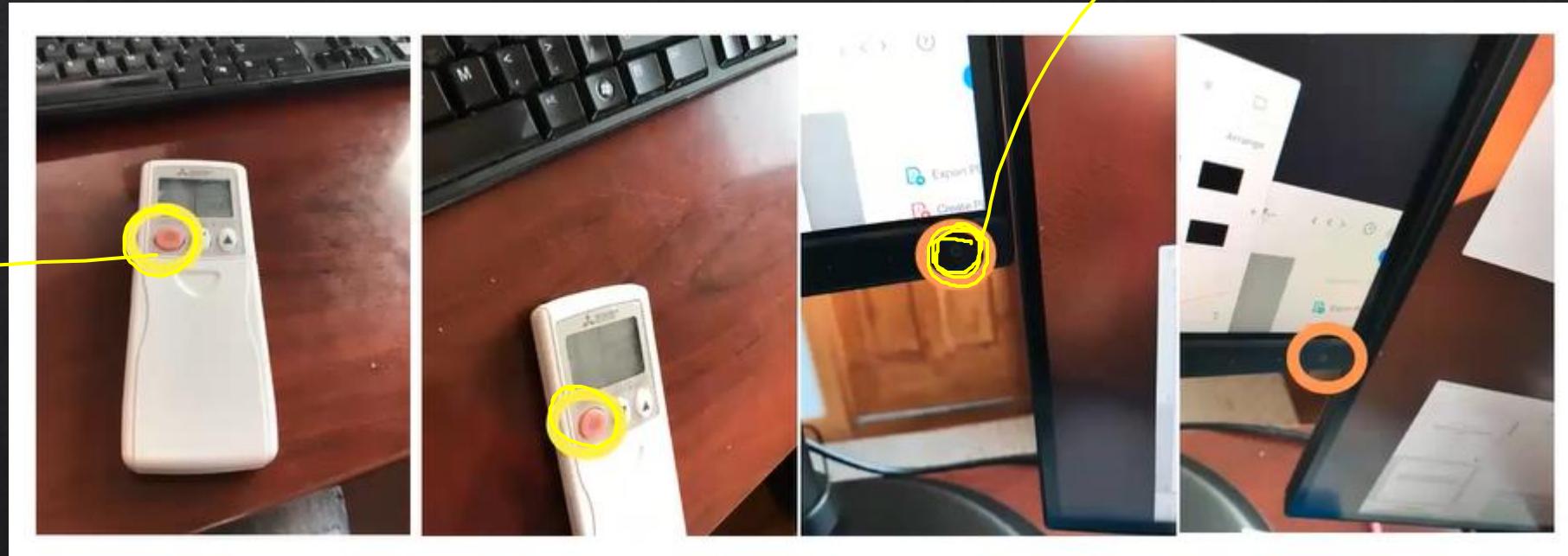
# Cómo ayuda *Representation Learning* para aprender a

- ◆ Aprendizaje por representación:
  - ◆ Mapear observaciones sin procesar para en atributos y estructuras, para que sea mas fácil de inferir desde el mapeo a acciones o desde el mapeo a etiquetas semánticas.



# Cómo ayuda *Representation Learning* para aprender a actuar

- ❖ Aprendizaje por representación:
  - ❖ Mapear observaciones sin procesar para en atributos y estructuras, para que sea mas fácil de inferir desde el mapeo a acciones o desde el mapeo a etiquetas semánticas.



FOV

# Cómo ayuda *Representation Learning* para aprender a actuar

- 
- ◊ Aprendizaje por representación:
    - ◊ Tener representaciones pre entrenadas con tareas auxiliares esta directamente relacionado con decrecer el numero de interacciones con el entorno.
    - ◊ Tareas auxiliares como detectar objetos, clasificar imágenes, etiquetar pixeles, etc..

# **Reinforcement Learning vs Supervised Learning**

# Aprendizaje Reforzado vs Aprendizaje Supervisado

- ❖ *Reinforcement Learning* es una forma de aprendizaje activo
  - ❖ *El agente tiene la posibilidad de **coleccionar su propia data** actuando en el entorno, consultándole a humanos, entre otras.*
  - ❖ *La data **cambia** a través del tiempo,*
  - ❖ *Para consultar el entorno efectivamente, el agente necesita mantener trackeada la **incertidumbre**, que sabe, que no sabe, y que debe explorar en el siguiente paso.*
- ❖ *Supervised Learning* es una forma de **aprendizaje pasivo**
  - ❖ La data no depende del agente, se entrega por los que etiquetan
  - ❖ La data es estática durante el entrenamiento

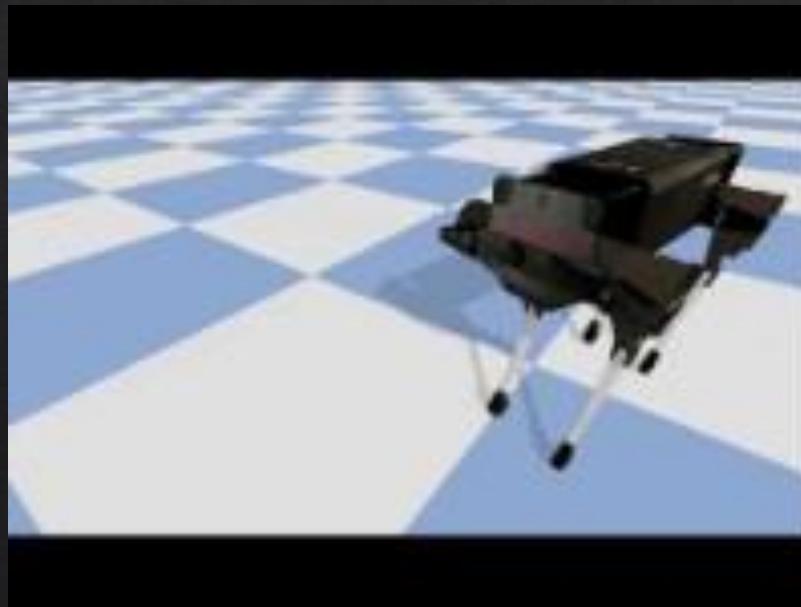
# Aprendizaje Reforzado vs Aprendizaje Supervisado

- ❖ En Aprendizaje Reforzado frecuentemente NO se puede usar optimización basada en gradientes
  - ❖ Ej.: *Cuando el agente no sabe ni del mundo del mundo para desenrollar ni una función de refuerzo para maximizar.*
- ❖ En Aprendizaje Supervisado, usualmente se usan optimización basada en gradiente
  - ❖ Ej.: *Consideramos una forma paramétrica de nuestro regresor o clasificador y la optimizamos mediante Descenso de Gradientes Estocásticos (SGD).*

# Aprendizaje Reforzado vs Aprendizaje Supervisado

- ❖ En *Reinforcement Learning* se consume mucho tiempo.
  - ❖ Las acciones toman tiempo para llevar a realizar en el mundo real.
  - ❖ El objetivo es que el agente minimice la cantidad de interacciones con el entorno mientras es exitoso en su tarea.
  - ❖ Podemos usar experiencia simulada para la transferencia en el mundo real. *Sim2Real*

E



# Aprendizaje Reforzado vs Aprendizaje Supervisado

- ❖ En *Reinforcement Learning* se consume mucho tiempo.
  - ❖ Podemos tener robots 24/7
  - ❖ Podemos tener muchos robots



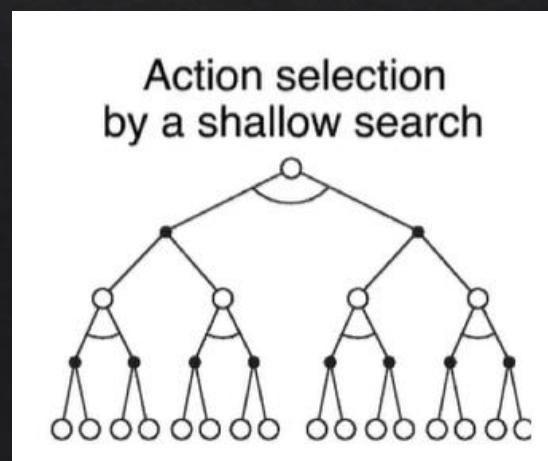
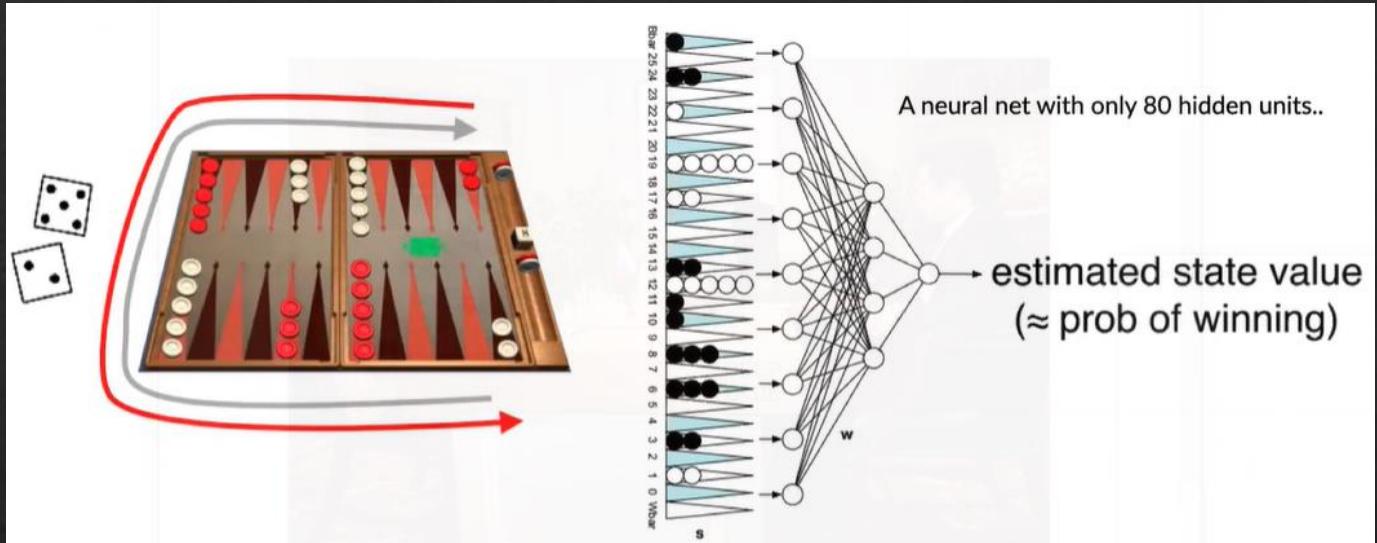
# Deep Blue

- ❖ SL vs RL
  - ❖ Deep blue usaba arboles de decisiones
  - ❖ Ahora si hay programas con RL



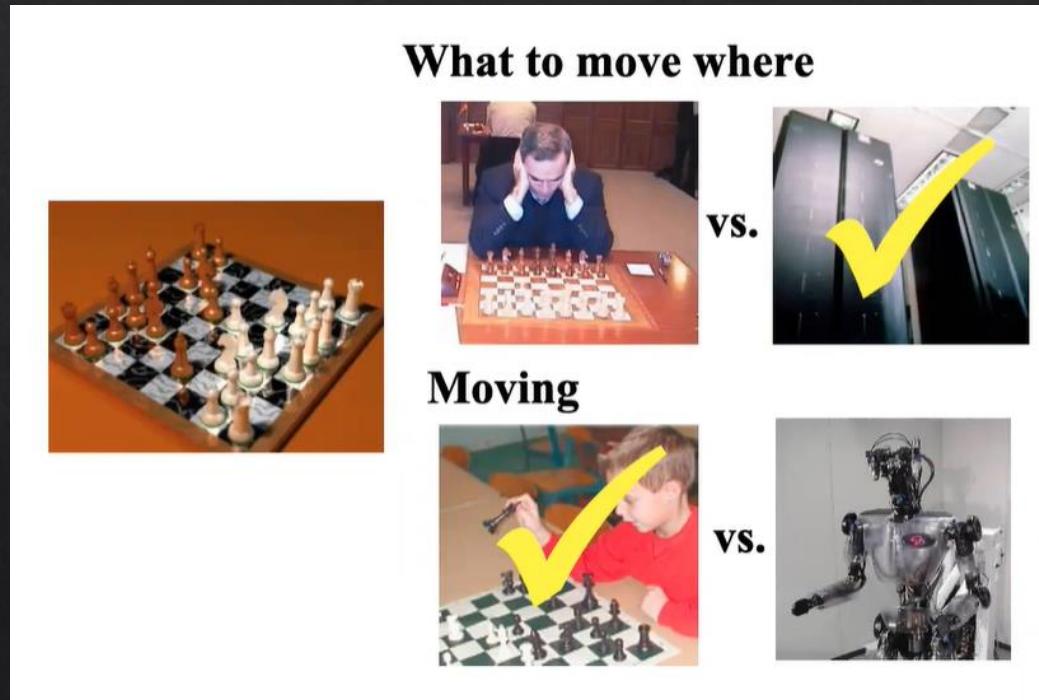
# BackGammon

- ❖ Neuro Gammon (1989)
  - ❖ Imitation Learning
  - ❖ Usaba supervised Learning
- ❖ TD Gammon (1992)
  - ❖ Red Neuronal se entrenaba a ella misma para ser una función de evaluación, jugando contra si misma.
  - ❖ Jugaba como un jugador top.



# Paradigma de la IA

- ❖ AlphaGo, ocupaba solo RL
- ❖ La dificultad del control de movimiento



# RL en el mundo real , ejemplo de AlphaGo

- ❖ Saber el entorno, en AlphaGo las entidades y dinámica es conocida vs entorno desconocido
- ❖ Necesidad de tener comportamientos capaces de ser transferidos en diferentes entornos
- ❖ Acciones discretas vs continuas
- ❖ Un objetivo vs muchos objetivos
- ❖ Recompensas son entregadas automáticamente vs recompensas deben ser descubiertas
- ❖ Las interacciones toman tiempo, explorar en forma inteligente



# Explicación Evolucionaria

- ❖ *IA es capaz de ganarle ajedrez al campeón del mundo peor no es capaz de mover un objeto como un niño de 2 años*
  - ❖ *Debemos esperar la dificultad de ingeniería inversa en cada habilidad humana como lo que demoramos en evolucionar como animales*
  - ❖ *Las habilidades humanas mas antiguas con inconscientes*
  - ❖ *Las habilidades que parecen inconscientes son muy difíciles de hacer en ingeniería inversa, pero las habilidades que requieren un esfuerzo puede que sean mas fáciles de ingenierizar.*

- Hans Moravec



# Aprender desde los niños más chicos

- ❖ Ser multimodal
- ❖ Ser incremental
- ❖ Ser físico
- ❖ Explorar
- ❖ Se social
- ❖ Aprende un idioma

