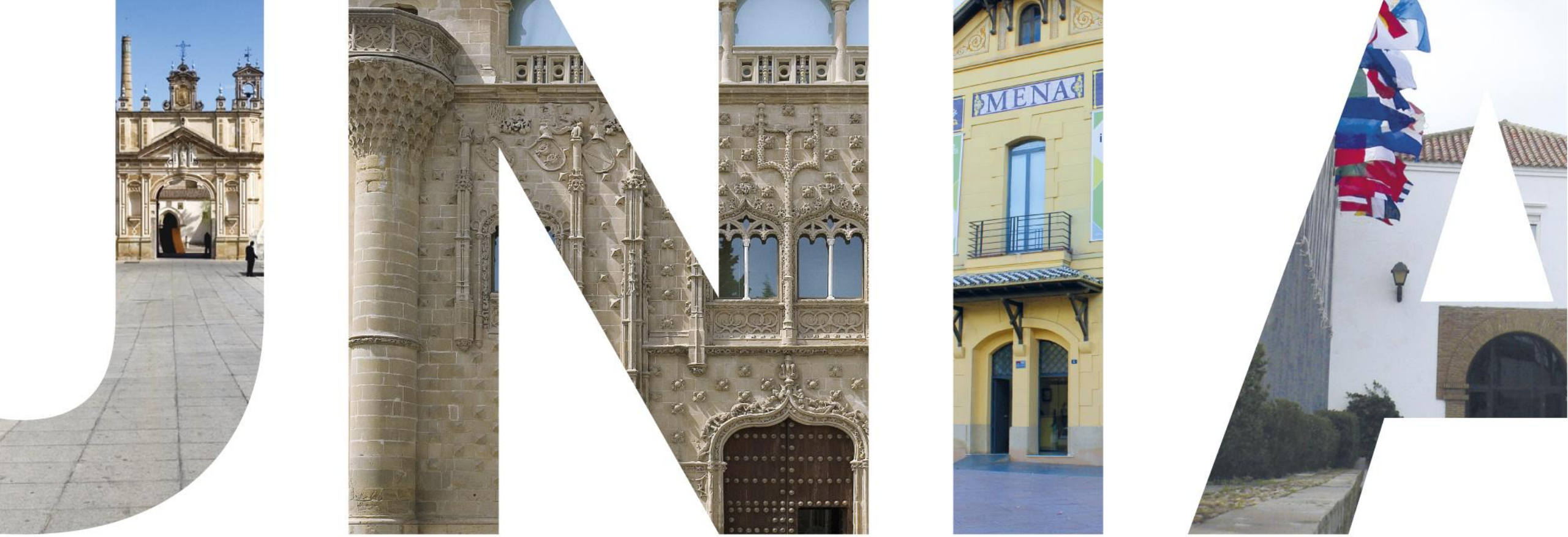


Tu universidad
de postgrado

Your university
for graduate
studies

Tu universidad
para una formación
permanente

Your lifelong
learning university

Tu universidad
para una enseñanza
innovadora

Your innovative
education university

**La universidad
para tu futuro**

**The university
for your future**

unia
Universidad
Internacional
de Andalucía

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

BioSiP
Biomedical Signal Processing, Computational Intelligence and Communications Security

SiPBA

DaSCI
Instituto Andaluz Interuniversitario en Data Science and Computational Intelligence

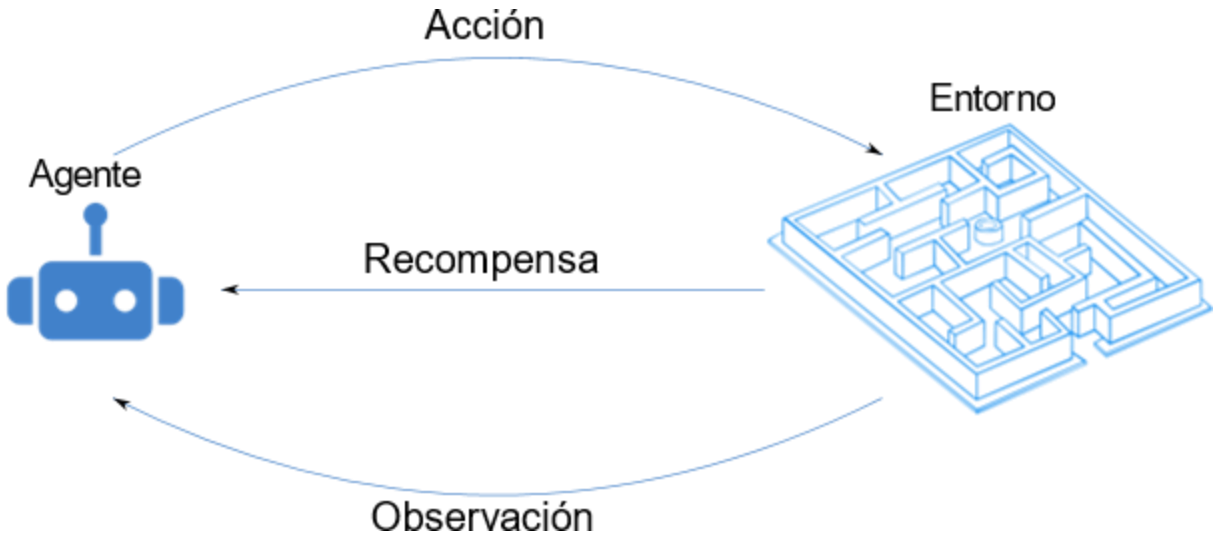
UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD DE GRANADA

Reinforcement Learning

Marco A. Formoso

Introducción

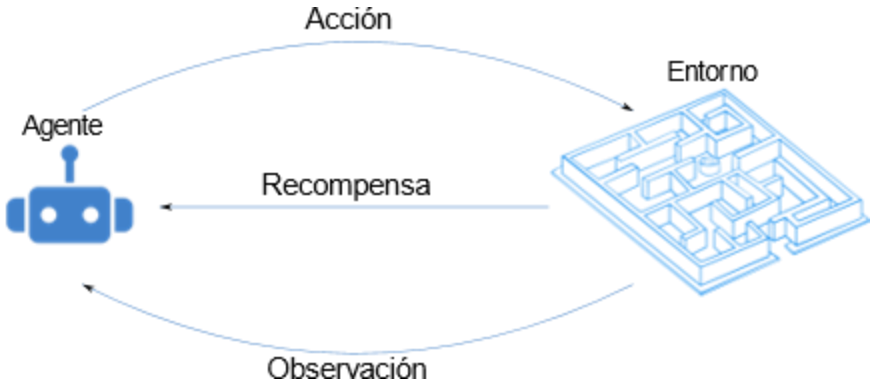


Introducción

Dónde un agente aprende de su propia experiencia.

Diferente de aprendizaje supervisado y no supervisado. El agente aprende en base a una recompensa solamente interactuando con su entorno.

Ejemplo ajedrez: si lo tomamos como supervisado, difícilmente podremos obtener una función Estado - Acción ya que virtualmente los estados son infinitos.



UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

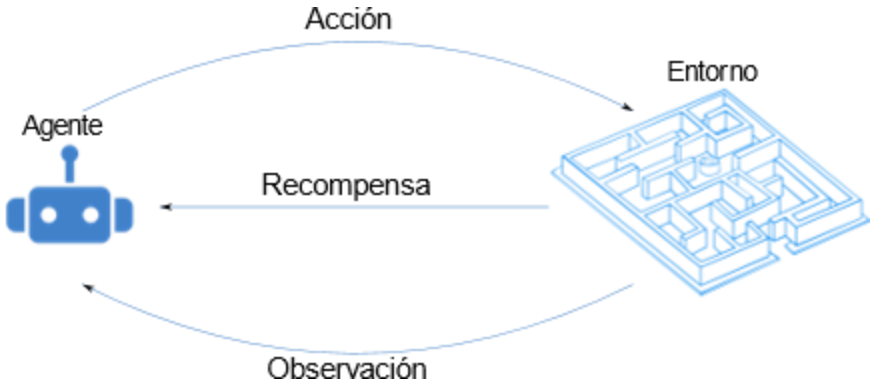
Introducción

AlphaStar:
Creado para jugar a StarCraft II

[Video]

[Blog AlphaStar]

[Blog StarCraft Entorno]

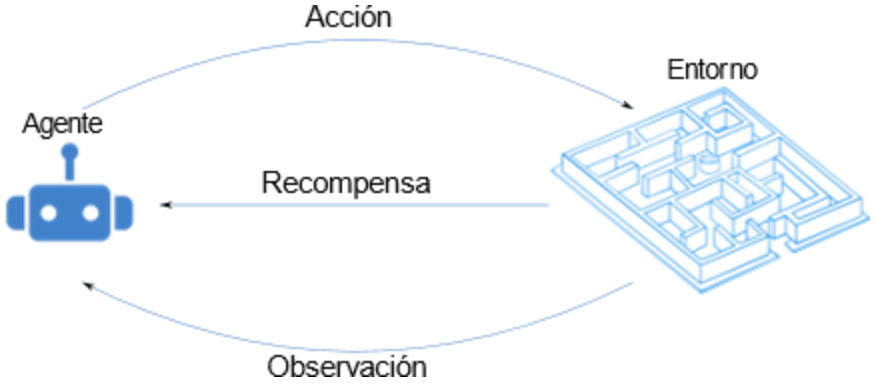


UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Introducción

AlphaGo:
Creado para jugar a Go

Blog



UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

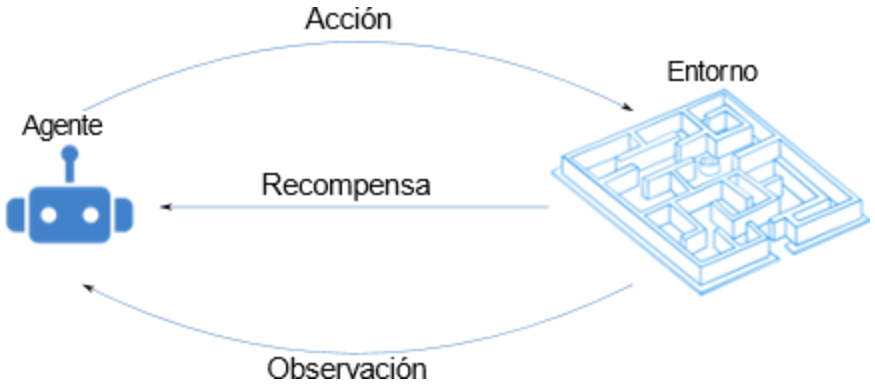
Introducción

Agente:

Entidad que es capaz de percibir su entorno mediante sensores, actuar en ese medio y recibir recompensas por ello.

Entorno:

Problema para el que el agente es solución. Se comunica con el agente mediante las recompensas, observaciones, y acciones.



- Observable vs Parcialmente Observable
- Determinista vs Estocástico
- Episódico vs Secuencial
- Estático vs Dinámico

Introducción

Acción:

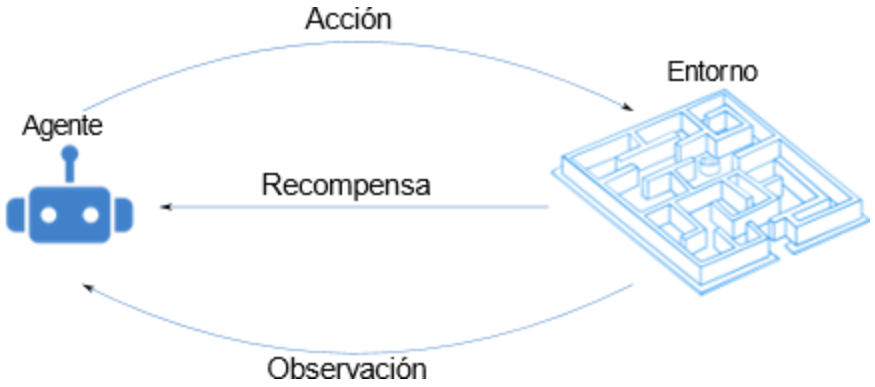
Respuestas o decisiones a las observaciones del agente. Discretas o continuas,

Observación:

Información que el agente obtiene del entorno y capta mediante sus sensores (Estado ≠ Observación). Puede llevar la recompensa ofuscada.

Recompensa:

”Estímulo” que recibe el agente. Positivo o negativo.



UNIVERSIDAD
INTERNACIONAL
DE ANDALUCÍA

Procesos de decisión de Markov

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Cadenas de Markov

Conjunto de estados:

Estados en el que el sistema puede estar.

Matriz de transición:

Nos indica las probabilidades de pasar de un estado a otro. Los procesos de Markov son estacionarios (la matriz de transición no cambia con el tiempo)

Normalmente no es conocida. Solo tenemos episodios de transiciones.

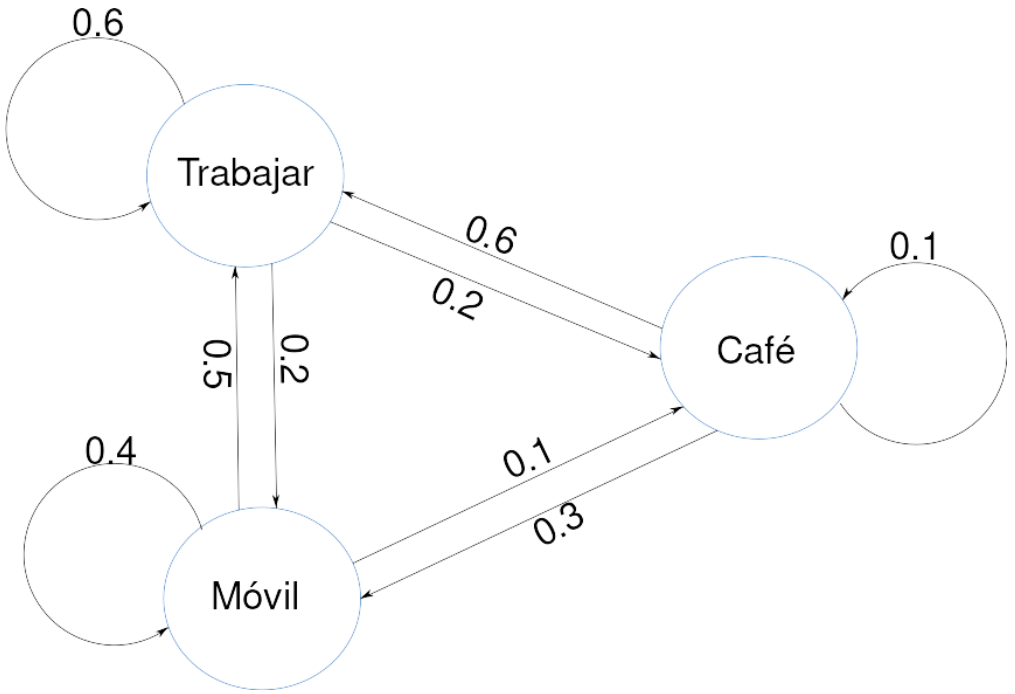
Hipótesis de Markov:

Los valores y decisiones solo son dependientes del estado actual.

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Cadenas de Markov

	Trabajar	Café	Móvil
Trabajar	0,6	0,2	0,2
Café	0,6	0,1	0,3
Móvil	0,5	0,1	0,4



Introducción

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Procesos de decisión de Markov

Recompensas:

Aditivas: $$U_t = R_{t+1} + R_{t+2} + R_{t+3} + \ldots = \sum_{k=0}^{\infty} R_{t+k+1}$$

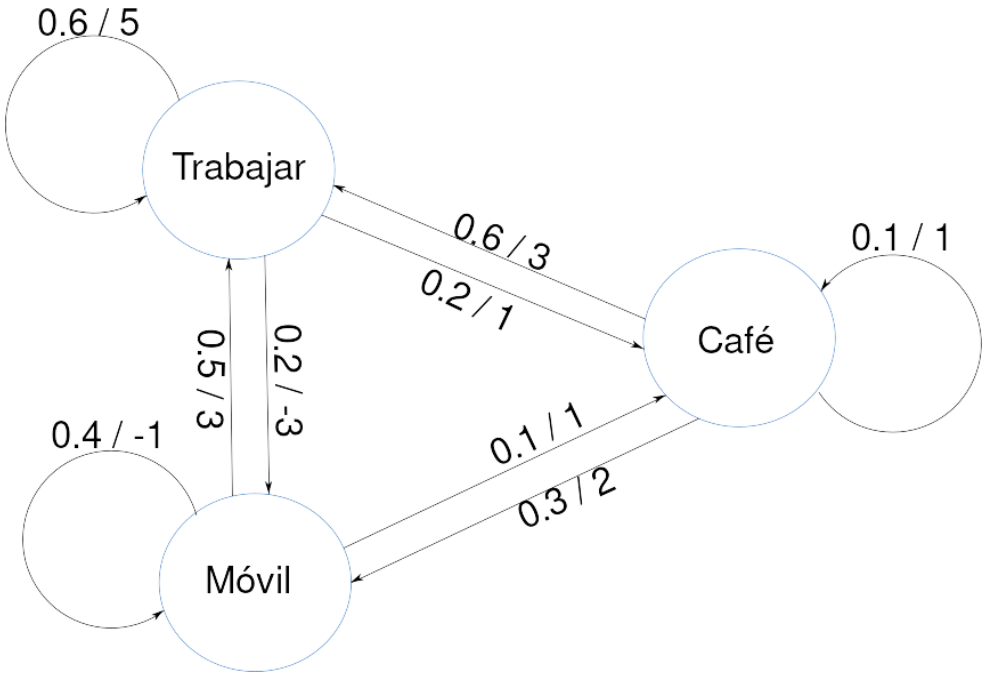
Con descuento: $$U_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \ldots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

Factor descuento: $0 < \gamma < 1$

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Procesos de decisión de Markov

	Trabajar	Café	Móvil
Trabajar	0,6 / 5	0,2 / 1	0,2 / -3
Café	0,6 / 3	0,1 / 1	0,3 / 2
Móvil	0,5 / 3	0,1 / 1	0,4 / -1



UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Procesos de decisión de Markov

Valor de un estado:

Medida que indica la bondad de un determinado estado.

$$V(s) = \mathbb{E}[U|S_t = s]$$

En el ejemplo anterior (si $\gamma = 0$):

V(Trabajar) = 0.6 * 5 + 0.2 * 1 + 0.2 * -3 = 2.6

V(movil)= 0.4 * -1 + 0.5 * 3 + 0.1 * 1 = 1.2

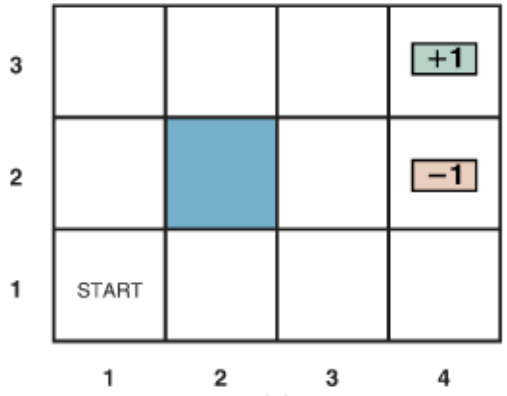
¿$\gamma = 1$?

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Procesos de decisión de Markov

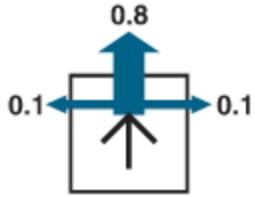
Acciones:

Conjunto finito en cada estado.



Proceso de decisión de Markov:

- Conjunto de estados: **S**
- Conjunto de acciones: **A**
- Modelo de transición: **P(s | s,a)**
- Función de recompensa: **R(s,a,s')**



UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Procesos de decisión de Markov

Política:

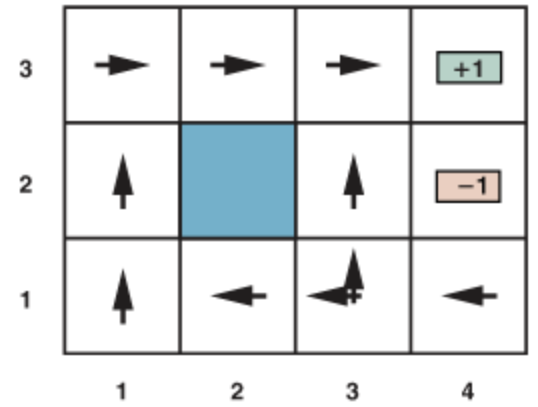
Acciones a realizar por el agente en cada estado. Π

Valor del estado:

$$V^{\pi}(s) = \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t R(S_t, \pi(S_t), S_{t+1})]$$

Política óptima:

$$\pi^* = \arg\max_{\pi} V^{\pi}(s)$$

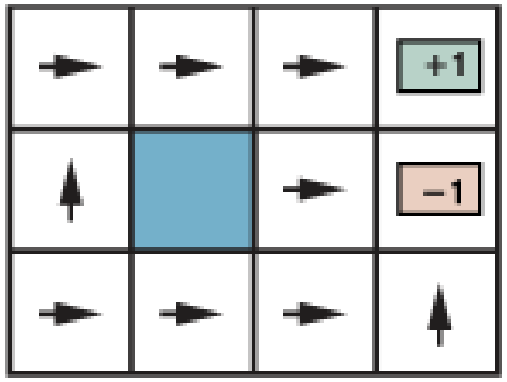


3	0.8516	0.9078	0.9578	+1
2	0.8016		0.7003	−1
1	0.7453	0.6953	0.6514	0.4279
	1	2	3	4

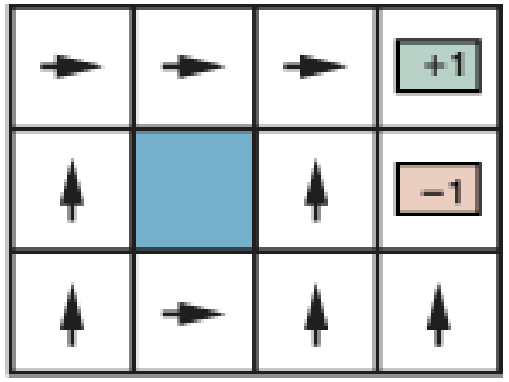
γ=1 and r = −0.04

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

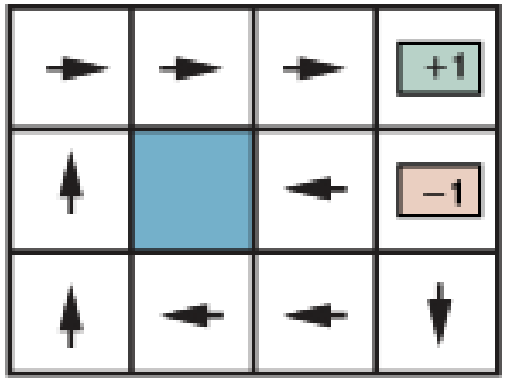
Procesos de decisión de Markov



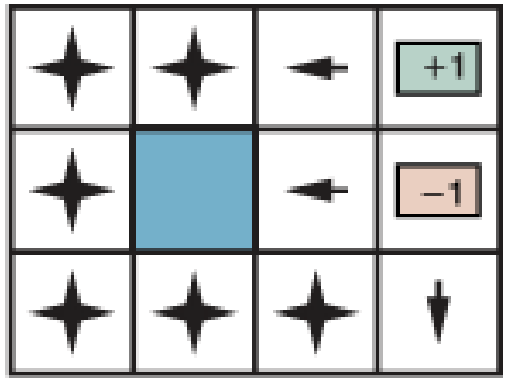
$r < -1.6497$



$-0.7311 < r < -0.4526$



$-0.0274 < r < 0$



$r > 0$

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Procesos de decisión de Markov

$$\pi^*(s) = \arg\max_{a \in A} \sum_{s'} P(s'|s,a) \left[R(s,a,s') + \gamma V(s') \right]$$

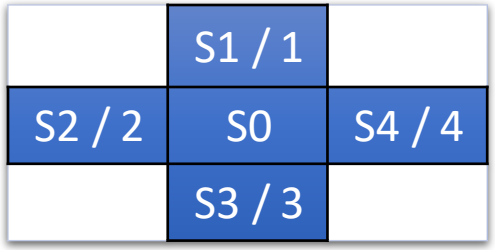
Ecuación de Bellman (Richard Bellman, 1957):

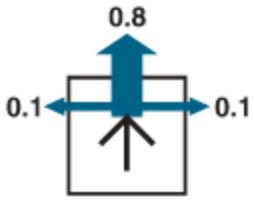
$$V(s) = \max_{a \in A} \sum_{s'} P(s'|s,a) \left[R(s,a,' s) + \gamma V(s') \right]$$

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

$$V(s) = \max_{a \in A} \sum_{s'} P(s'|s,a) \left[R(s,a,'s) + \gamma V(s') \right]$$

Procesos de decisión de Markov





Acción Arriba:

0.8 * 1 + 0.1 * 2 + 0.1 * 4 = 1,4

Acción Abajo:

0.8 * 3 + 0.1 * 2 + 0.1 * 4 = 3

Acción Derecha:

0.8 * 4 + 0.1 * 1 + 0.1 * 3 = 3.6

Acción Izquierda:

0.8 * 2 + 0.1 * 3 + 0.1 * 1 = 2

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Procesos de decisión de Markov

Q-function:

$$V(s) = \max_a Q(s,a) \qquad \pi^*(s) = \arg\max_a Q(s,a)$$

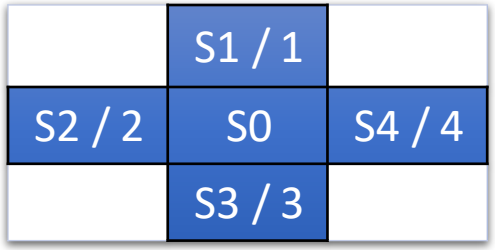
Ecuación de Bellman para Q:

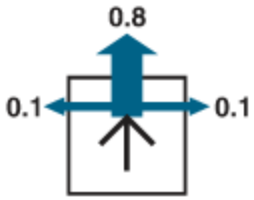
$$Q(s,a) = \sum_{s'} P(s'|s,a)\left[R(s,a,s') + \gamma \max_{a'} Q(s',a')\right]$$

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

$$Q(s,a) = \sum_{s'} P(s'|s,a) \left[R(s,a,s') + \gamma \max_{a'} Q(s',a') \right]$$

Procesos de decisión de Markov





Acción Arriba : **Q(S0, Arriba)**

0.8 * 1 + 0.1 * 2 + 0.1 * 4 = 1,4

Acción Abajo: **Q(S0, Abajo)**

0.8 * 3 + 0.1 * 2 + 0.1 * 4 = 3

Acción Derecha: **Q(S0, Derecha)**

0.8 * 4 + 0.1 * 1 + 0.1 * 3 = 3.6

Acción Izquierda: **Q(S0, Izquierda)**

0.8 * 2 + 0.1 * 3 + 0.1 * 1 = 2

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Tabular Learning

$$Q(s,a) = \sum_{s'} P(s'|s,a) \left[R(s,a,s') + \gamma \max_{a'} Q(s',a') \right]$$

Procesos de decisión de Markov

Acción/Estado	Estado 1	Estado 2	Estado N
Acción 1			
Acción 2			
Acción 3			
Acción N			

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Procesos de decisión de Markov

Algoritmo de Iteración de Valores:

Bellman Update:

$$V_{t+1}(s) \leftarrow \max_{a \in A} \sum_{s'} P(s'|s,a) \left[R(s,a,s') + \gamma V_t(s') \right]$$

1. Inicializar variables a un valor inicial.
2. Realizar Bellman Update para cada estado.
3. Repetir para un número determinado de iteraciones o hasta que los cambios Δ sean mínimos.

Sigue en Colab: IntroGYM_ValueIteration.ipynb

UNIVERSIDAD
INTERNACIONAL
DE ANDALUCÍA

Temporal-difference learning

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

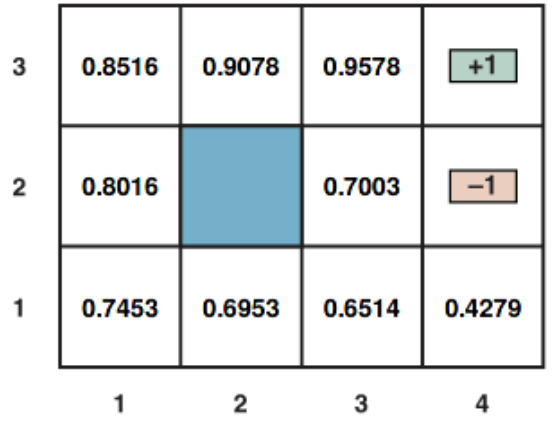
Temporal-difference learning

Consideremos la transición - (1,3) -> (2,3):
Si la transición ocurre la mayoría de veces, el valor esperable de V(1,3) sería entonces:

V(1,3) = -0.04 + V(2,3) = -0.04 + 0.9078 = 0.8678

Error: $\left[R(s,\pi(s),s') + \gamma V^{\pi}(s') - V^{\pi}(s)\right]$

$$V^{\pi}(s) \leftarrow V^{\pi}(s) + \alpha \left[R(s,\pi(s),s') + \gamma V^{\pi}(s') - V^{\pi}(s)\right]$$



UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Temporal-difference learning

QLearning

Bellman Update:

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[R(s,a,s') + \gamma \max_{a'} Q(s',a') - Q(s,a) \right]$$

$$Q(s,a) \leftarrow (1-\alpha) Q(s,a) + \alpha \left[R(s,a,s') + \gamma \max_{a'} Q(s',a') \right]$$

1. Inicializar valores Q(s,a)
2. Obtenemos (s,a,r,s')
3. Bellman update
4. Repetir desde 2 si no se cumple la condición de parada

Sigue en Colab: TD_QLearning.ipynb

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Temporal-difference learning

Problemas:

Aunque ya no tengamos que iterar sobre todos los estados, el número de estados puede seguir siendo muy alto:

1. estados representados por números reales
2. Atari: 160 x 192 pixels, 128 colors - 128^(160*192)

Podemos usar una función no lineal de un estado y una acción y que nos devuelva un valor -> Regresión.

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Deep QLearning

MNIH, Volodymyr, et al. Human-level control through deep reinforcement learning. *nature*, 2015

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Deep QLearning

Usemos una NN para la regresión.

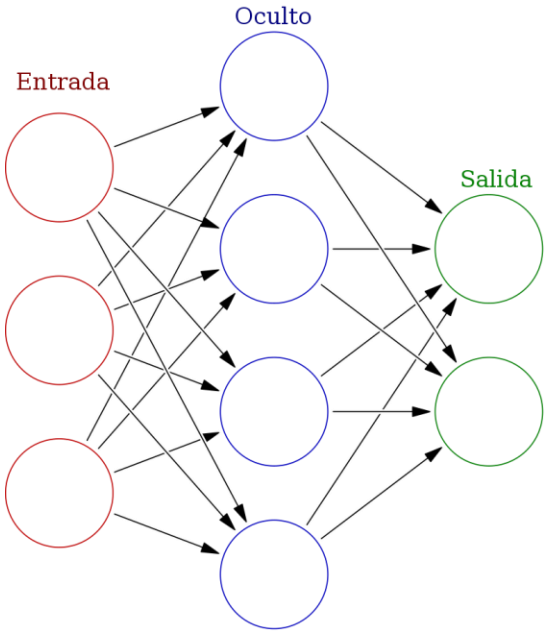
Algoritmo:

1. Iniciamos Q(s,a)
2. Obtenemos la transición (s,a,r,s')
3. Calculamos la función de pérdida:

$$\mathscr{L} = (Q(s,a) - r)^2$$

$$\mathscr{L} = (Q(s,a) - (r + \gamma \underset{a'}{max} Q(s',a'))^2$$

4. Actualizamos usando SGD
5. Repetimos



UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Deep QLearning

Algoritmo básico problemas:

Exploration vs Exploitation
Con cierta probabilidad ϵ seleccionamos una acción aleatoria. $0 \leq \epsilon \leq 1$
EPSILON-GREEDY

SGD requiere que los datos sean independientes y uniformemente distribuidos:

1. Muestras pertenecientes al mismo episodio
2. La política va cambiando según entrenemos

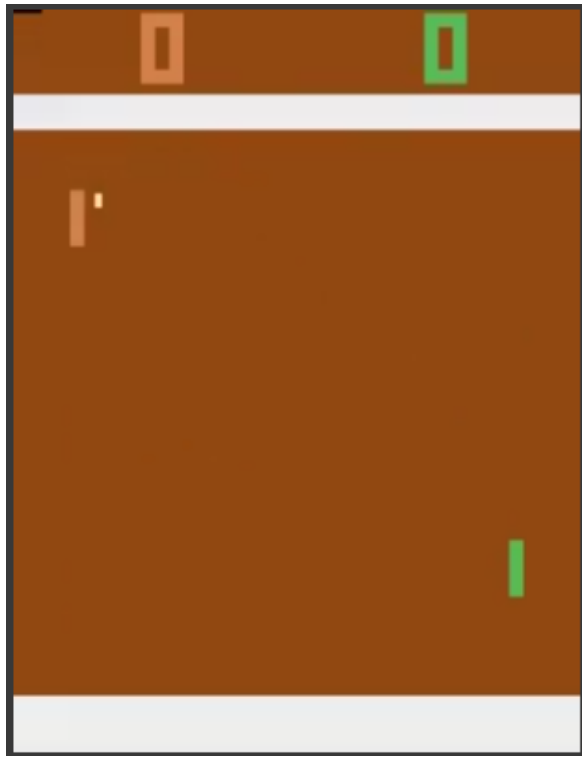
REPLAY BUFFER

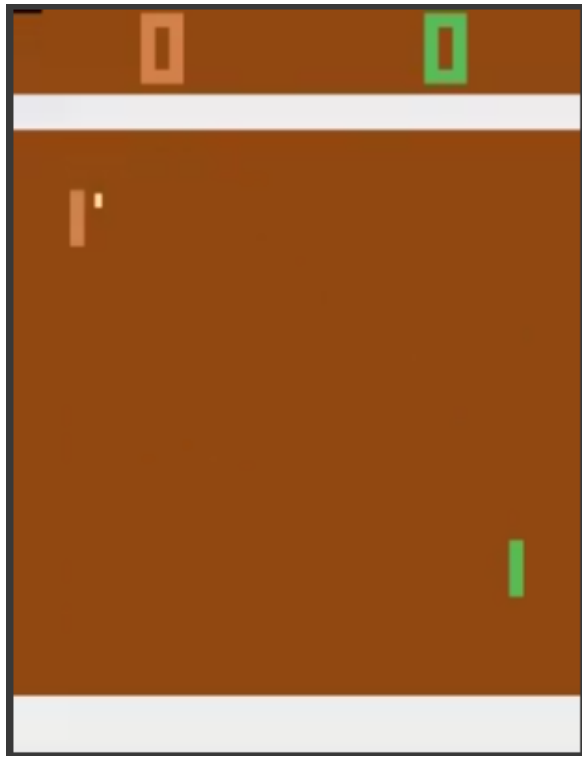
Estados consecutivos Q(s,a) - Q(s',a') casi iguales. Al actualizar el valor de Q(s,a) indirectamente cambiamos el de Q(s',a').
USAR DOS NN

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Algoritmo básico problemas:
Propiedad de Markov

Deep QLearning





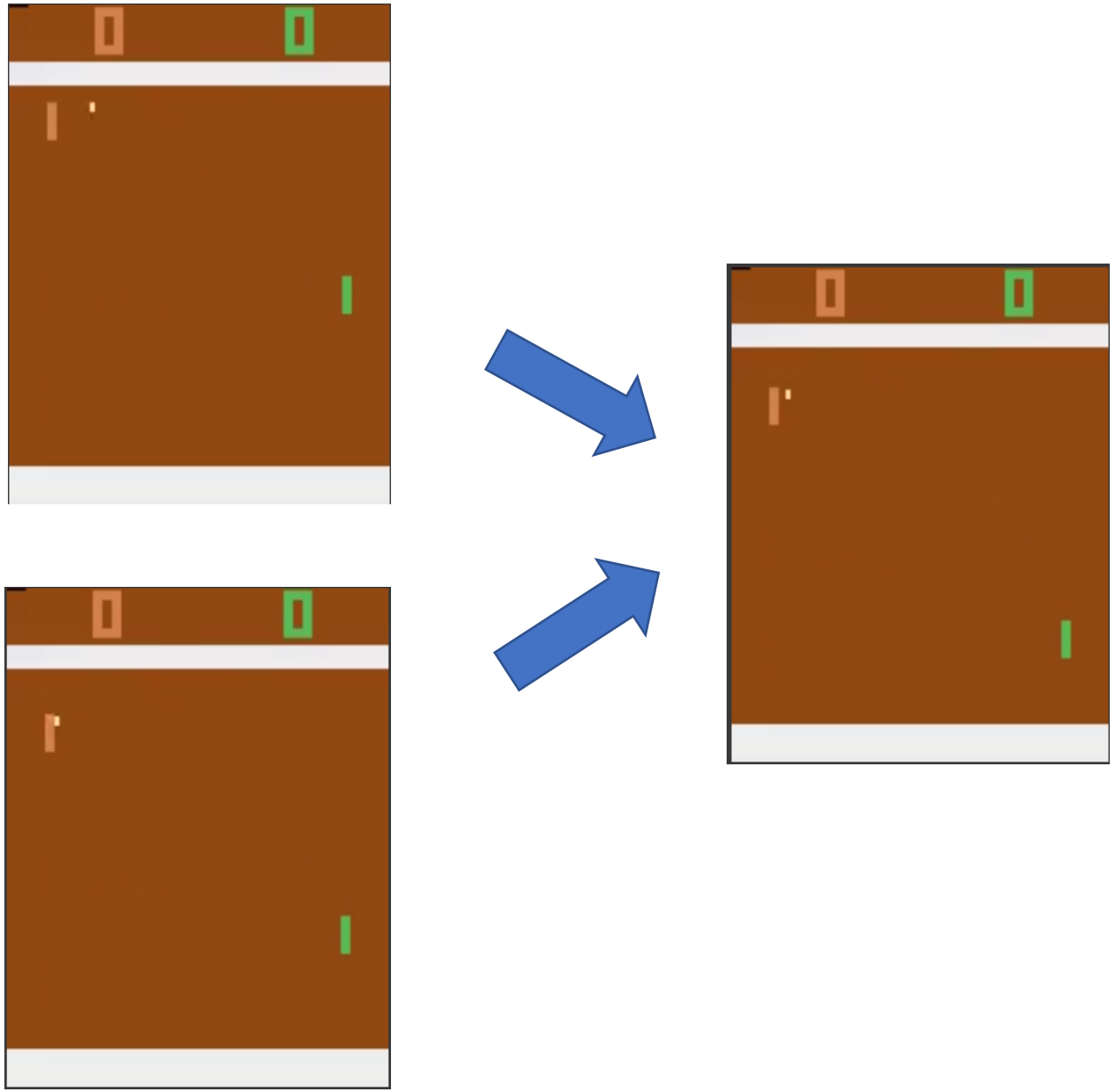
UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Deep QLearning

Algoritmo básico problemas:
Propiedad de Markov

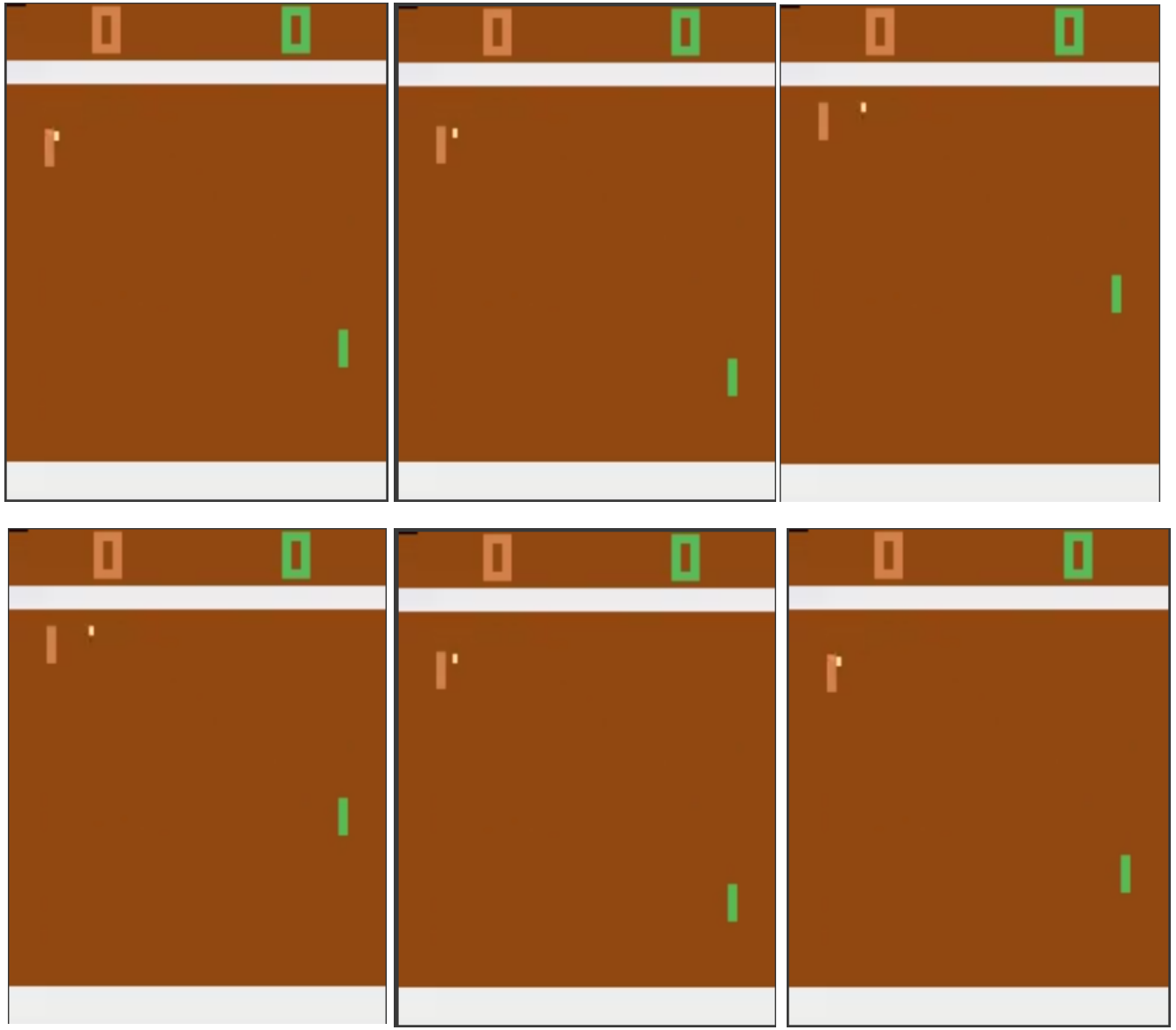
UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Deep QLearning



UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Deep QLearning



CONCATENAMOS VARIOS ESTADOS EN UNO

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Deep QLearning

Usemos una NN para la regresión.

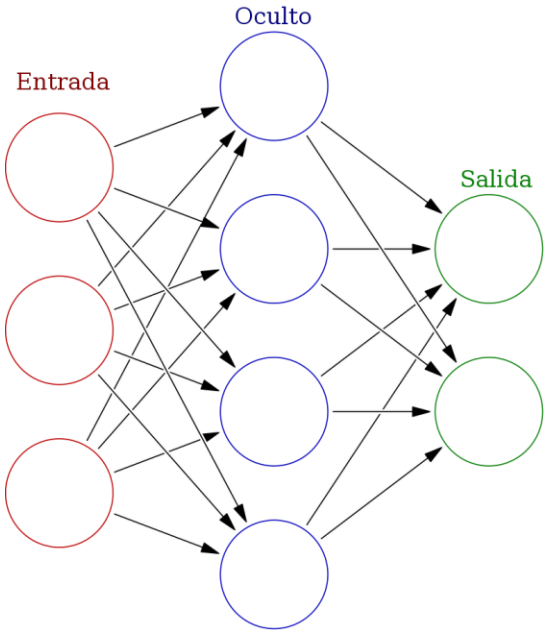
Algoritmo:

1. Iniciamos las redes Q(s,a) y Q(s',a'). ϵ <- 1. Buffer vacío.
2. Con probabilidad ϵ elegimos acción aleatoria, si no *argmaxQ(s,a)*
3. Obtenemos la transición (s,a,r,s') y la guardamos en el buffer.
4. Extraemos del buffer un minibatch
5. Calculamos la función de pérdida:

$$\mathscr{L} = (Q(s,a) - r)^2$$

$$\mathscr{L} = (Q(s,a) - (r + \gamma \underset{a'}{max} Q(s',a'))^2$$

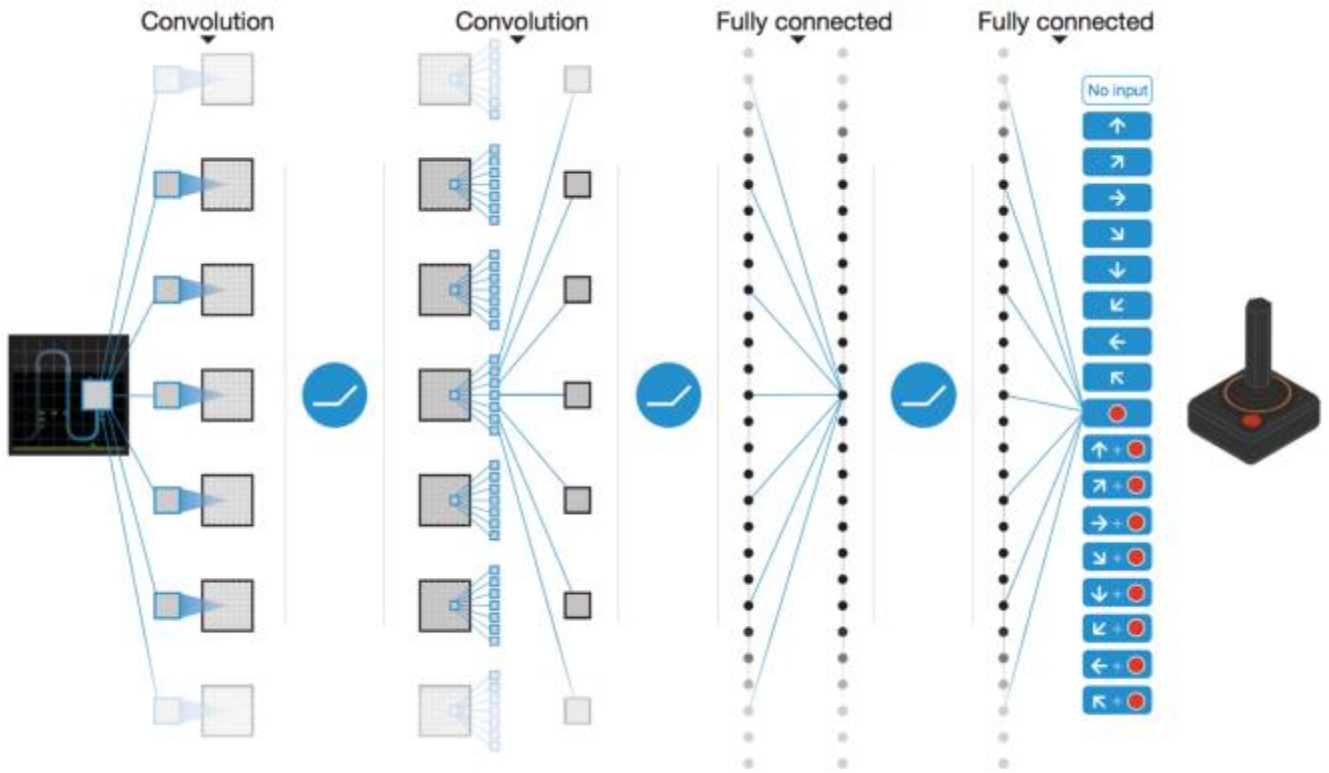
6. Actualizamos la red Q(s,a) usando SGD
7. Cada N pasos copiamos los pesos de *Q(s,a)* a *Q(s',a')*
8. Repetimos



De en:User:Cburnett - File:Colored neural network uk.svg, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=78778791

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Deep QLearning



UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Deep QLearning

Mejoras:

- **N-steps DQN**: Desenrollar la ecuación de Bellman. *Sutton, 1988, Learning To Predict by The Methods of Temporal Differences*
- **Double DQN**: DQN sobreestima los valores.Usar las acciones dadas por la red primaria pero tomando los valores de la secundaria. *Van Hasselt et al., 2015, Deep Reinforcement Learning with Double Q-Learning*
- **Noisy networks**: Exploration vs Exploitation. Añadimos ruido a las capas FC de la red. *Fortunato et al., 2017, Noisy Networks for Exploration*
- **Prioritized replay buffer:** Entrenar en data que te "sorprenda" más. Proporcional al loss. *Tom Schaul et al., 2015, Prioritized Experience Replay*
- **Dueling DQN:** Separar la ventaja de una acción y el valor del estado al final de la red. Q(s,a) = V(s) + A(s,a). *Ziyu et al., 2015, Dueling Network Architectures for Deep Reinforcement Learning*
- **Categorical DQN:** QValues distribution. *Bellemare et al., 2017, A Distributional Perspective on Reinforcement Learning*

UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA

Referencias usadas:

Artificial Intelligence: A Modern Approach, 4th Global ed., Russell, Norvig. Capítulos 16, 23

Deep Reinforcement Learning Hands On, Maxim Lapan, Capítulos 1-6