



PONTIFICIA  
UNIVERSIDAD  
CATÓLICA  
DE CHILE

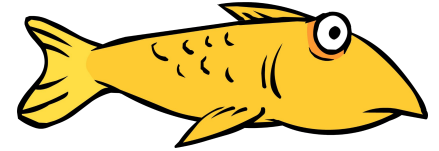
# Reinforcement Learning (Q-Learning)

Daniel Florea - IIC2613 Inteligencia Artificial

# Introducción



# Introducción



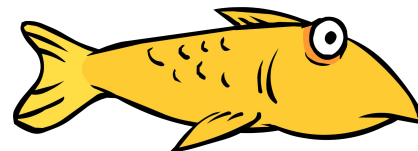
# Introducción



# Introducción



**-5**



**+5**

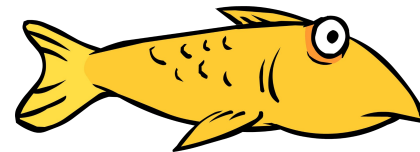
# Introducción



**-5**



	Izquierda	Derecha
QValue	0	0



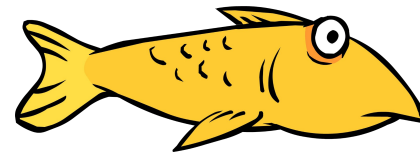
**+5**

# Introducción

	Izquierda	Derecha
QValue	-5	0



**-5**

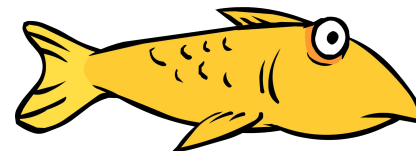


**+5**

# Introducción



**-5**



**+5**

	Izquierda	Derecha
QValue	-5	0

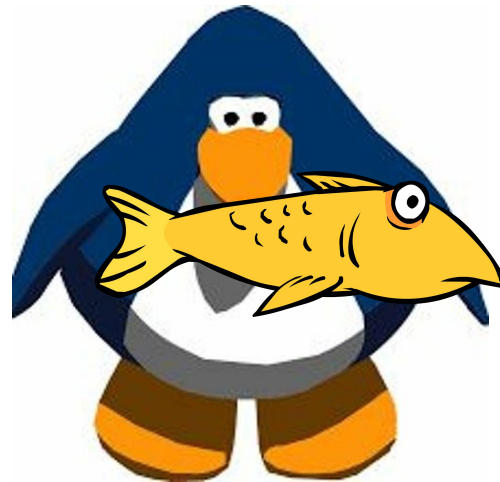


# Introducción



**-5**

	Izquierda	Derecha
QValue	-5	+5



**+5**

# Introducción

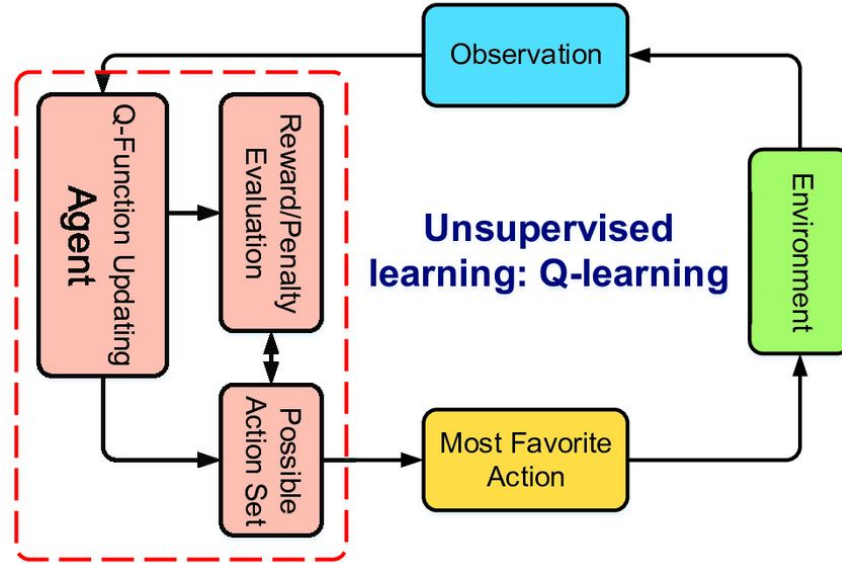


	Izquierda	Derecha
QValue	-5	+5

# Q-Learning

# ¿Qué es Q-Learning?


Un algoritmo de aprendizaje reforzado no supervisado



# Principales componentes

- Una tabla de estados o Q-Table:  $Q$
- Una tasa de exploracion:  $\epsilon$
- Una tasa de aprendizaje (LR):  $\alpha$
- Un conjunto de estados para el entorno:  $X_t = [x_1, x_2, \dots, x_n]$
- Un conjunto de acciones de las que elegir:  $U = [u_1, u_2, \dots, u_k]$

# Pasos para entrenar

- 1 Inicializar Q-Table y al agente
  - 2 Obtener el estado actual del agente
  - 3 Hashear el estado actual del agente para consultar la Q-Table
  - 4 Consultar la Q-Table/ Elegir acción random y ejecutarla
  - 5 Recibir recompensa y feedback del entorno
  - 6 Actualizar la Q-Table según la recompensa de la acción ejecutada
  - 7 Actualizar la tasa de exploración hasta llegar a un mínimo
- 
- A vertical line on the left side of the list connects the bottom of step 7 to the left of step 2, with an arrow pointing right towards step 2, indicating a loop in the training process.

# 1 Inicializar Q-Table

	$u_1$	$u_2$	$\dots$	$u_k$
$x_1$	0	0	$\dots$	0
$x_2$	0	0	$\dots$	0
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$x_n$	0	0	$\dots$	0

$(n \times k)$

2

## Obtener el estado actual

$$\left[ s_0, s_1, \dots, s_m \right]$$

Descriptores del entorno



3

## Hashear el estado actual

$$\left[ s_0, s_1, \dots, s_m \right] \longrightarrow x_i$$

Descriptores del entorno

Índice de la Tabla

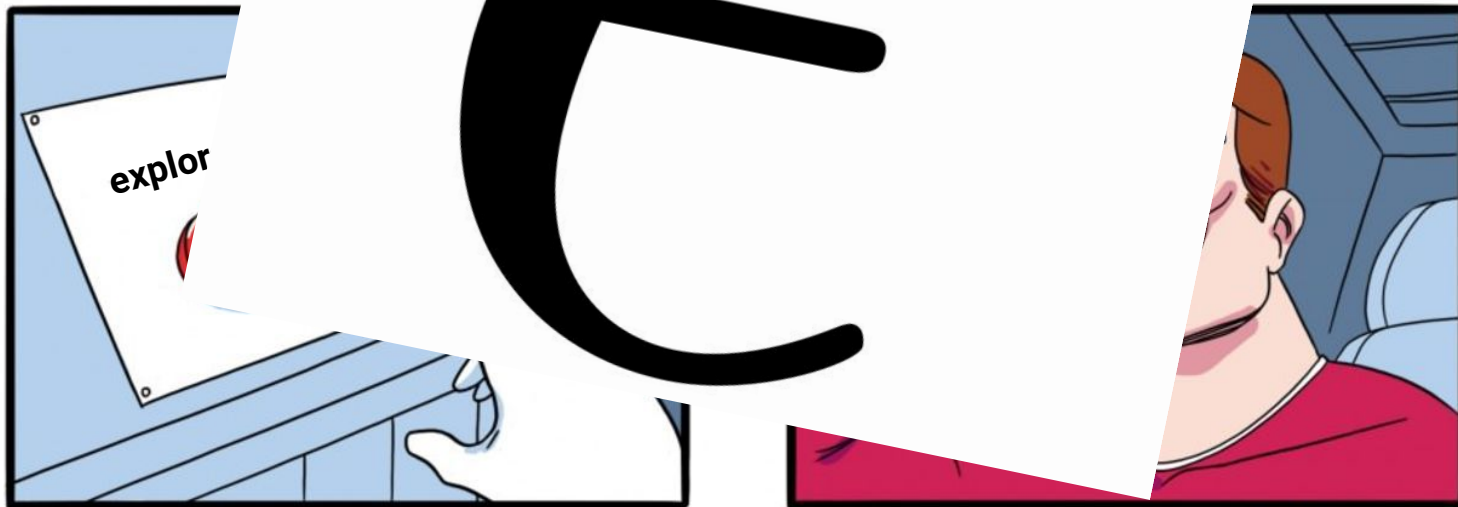
## 4 Explorar v/s Explotar

- **Explorar:** Elegir una acción aleatoria para conocer mejor el comportamiento del entorno
- **Explotar:** Utilizar el conocimiento aprendido para comportarse en el entorno



## 4 Explorar

- **Explorar:** Elegir u
- **Explotar:** Utiliza



## 4 Explorar v/s Explotar

- **Exploration Rate:** Da cuenta de una probabilidad con la que el agente explorará en esta iteración (entre 0 y 1)

Consultar la Tabla

$$probabilidad = 1 - \epsilon$$

Elegir acción random

$$probabilidad = \epsilon$$

4.1

# Consultar la Tabla

$$\operatorname{argmax} Q[x_i] = u_i$$

$x_i$

	$u_1$	$u_2$	$\dots$	$u_k$
$x_1$	0	0	$\dots$	0
$x_2$	0	0	$\dots$	0
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$x_n$	0	0	$\dots$	0

$(n \times k)$

## 4.1 Consultar la Tabla

$$\operatorname{argmax} Q[x_i] = u_i$$

		$u_1$	$u_2$	$\dots$	$u_k$
$x_1$		23	-26	$\dots$	0
$x_i$	$\rightarrow x_2$	-5	103	$\dots$	2
$\dots$		$\dots$	$\dots$	$\dots$	$\dots$
$x_n$		0	0	$\dots$	0

$(n \times k)$

## 4.1 Consultar la Tabla


		$u_1$	$u_2$	$\dots$	$u_k$
$x_i$	$x_1$	23	-26	$\dots$	0
	$x_2$	-5	103	$\dots$	2
	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
	$x_n$	0	0	$\dots$	0

$(n \times k)$

4.2

## Elegir una acción random

$u_i = \text{random element from } U$

$$U = [u_1, u_2, \dots, u_k]$$




5

Recibir recompensa del entorno

$$reward = env.play\_step(u_2)$$

6

## Actualizamos la Q-Table

Learning Rate

Tasa de Descuento

$$new\_q\_value = (1 - \alpha) \cdot \underset{\text{Acción ejecutada}}{\operatorname{argmax}} Q[x_i] + \alpha(\underset{\text{Calidad de la mejor acción en el estado futuro}}{reward} + \gamma \cdot \underset{\text{Calidad de la mejor acción en el estado futuro}}{\operatorname{argmax}} Q[x_{i+1}])$$

Acción ejecutada

Calidad de la  
mejor acción en el  
estado futuro

6

# Actualizamos la Q-Table

$$new\_q\_value = (1 - \alpha) \cdot \argmax Q[x_i] + \alpha(reward + \gamma \cdot \argmax Q[x_{i+1}])$$

	$u_1$	$u_2$	...	$u_k$
$x_1$	23	-26	...	0
$x_2$	-5	New Q-Value	...	2
...	...	...	...	...
$x_n$	0	0	...	0

$(n \times k)$

## 7 Actualizar la tasa de exploración

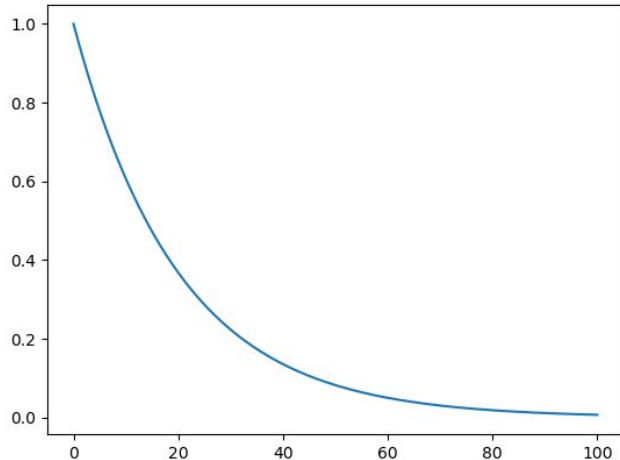
- **Exploration Rate:** Da cuenta de una probabilidad con la que el agente explorará en esta iteración

$$\epsilon = \epsilon_{min} + (\epsilon_{max} - \epsilon_{min}) \cdot e^{-decay\_rate \cdot num\_iteration}$$

## 7 Actualizar la tasa de exploración

- **Exploration Rate:** Da cuenta de una probabilidad con la que el agente explorará en esta iteración

$$\epsilon = \epsilon_{min} + (\epsilon_{max} - \epsilon_{min}) \cdot e^{-decay\_rate \cdot num\_iteration}$$



## 7 Actualizar la tasa de exploración

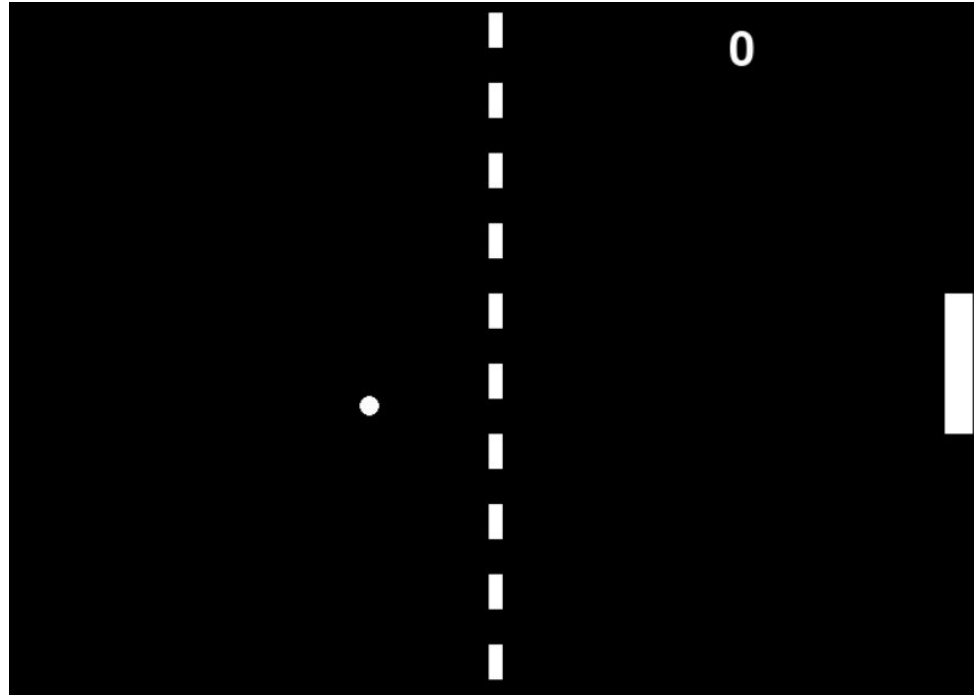
- **Exploration Rate:** Da cuenta de una probabilidad con la que el agente explorará en esta iteración

$$\epsilon = \epsilon_{min} + (\epsilon_{max} - \epsilon_{min}) \cdot e^{-decay\_rate \cdot num\_iteration}$$

Y repetir hasta que  $\epsilon = \epsilon_{min}$

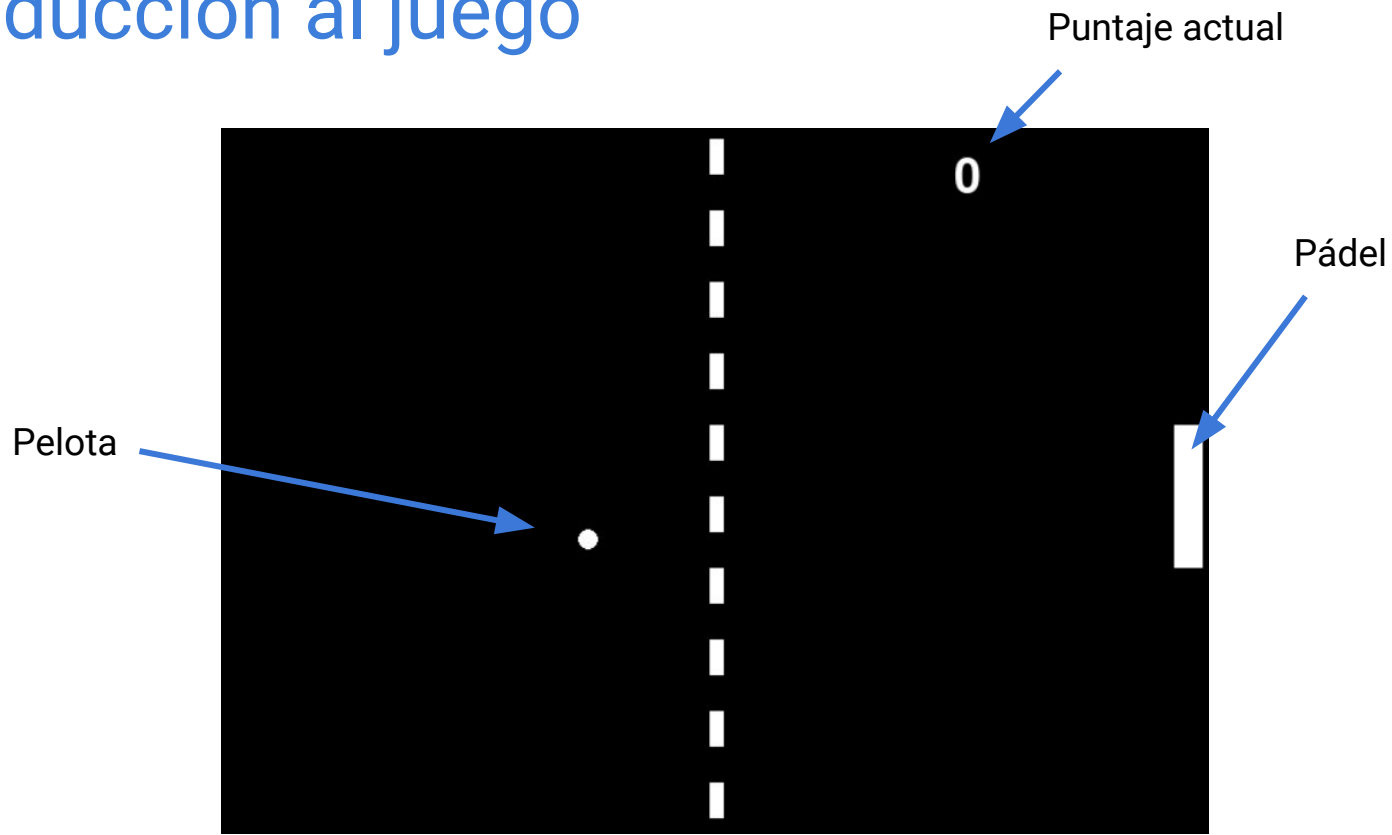
# Implementación en Código

# PongAI





# Introducción al juego



# Introducción al juego



PongAI



Modela el juego y recompensas (no editar)



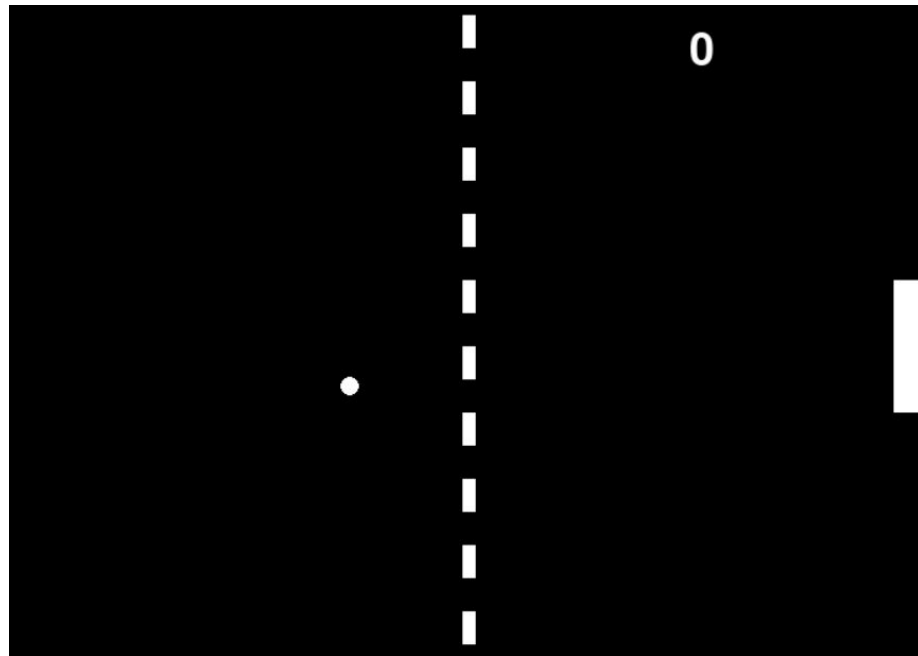
QAgent



Modela el agente y su entrenamiento (editar)

# Estados de juego

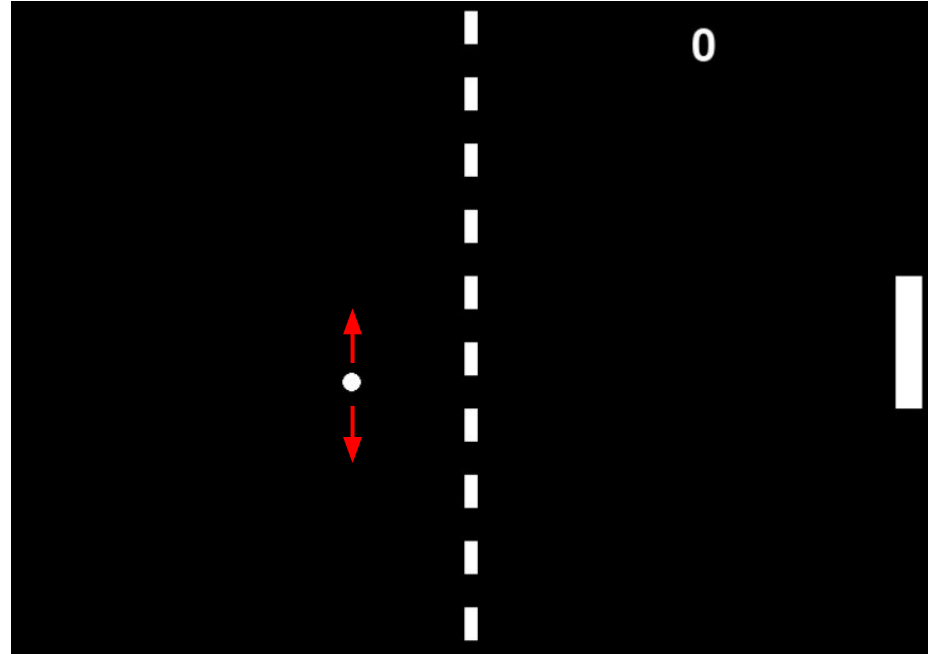
1. Velocidad en el eje Y
2. Proximidad al padel
3. Altura del balon respecto al extremo superior del padel
4. Altura del balon respecto al extremo inferior del padel
5. Numero de rebotes



# Estados de juego

1. Velocidad en el eje Y
2. Proximidad al padel
3. Altura del balon respecto al extremo superior del padel
4. Altura del balon respecto al extremo inferior del padel
5. Numero de rebotes

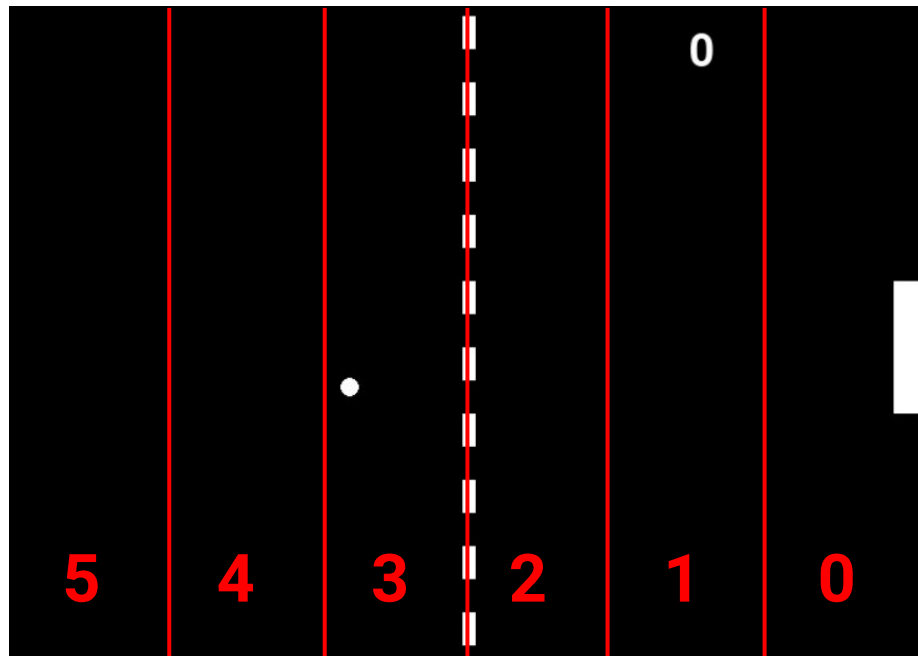
Entre -5 y 5



# Estados de juego

1. Velocidad en el eje Y
2. Proximidad al padel
3. Altura del balon respecto al extremo superior del padel
4. Altura del balon respecto al extremo inferior del padel
5. Numero de rebotes

Entre 0 y 5



# Estados de juego

1. Velocidad en el eje Y

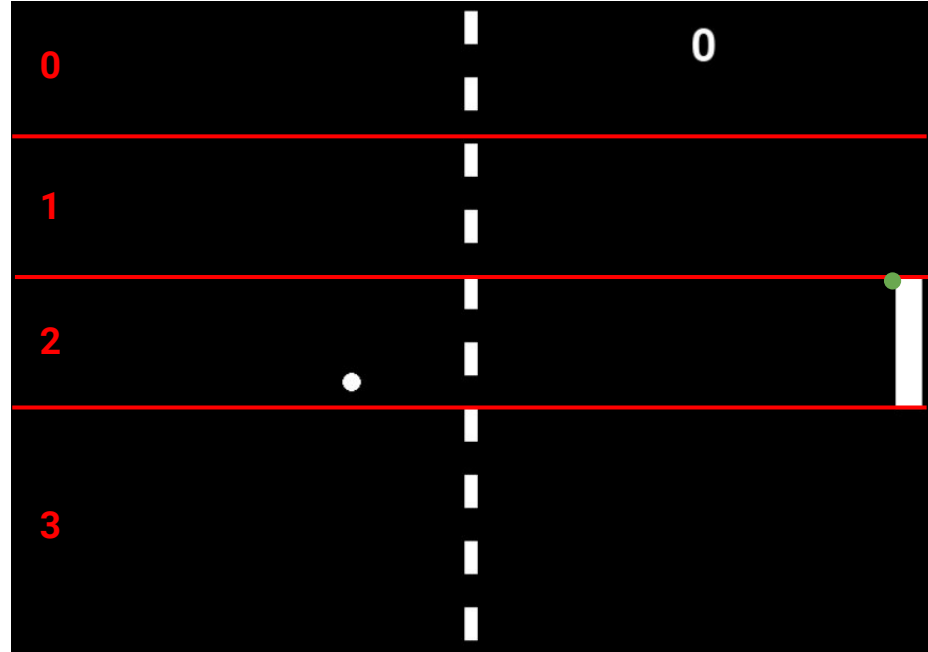
2. Proximidad al padel

3. Altura del balon respecto al extremo superior del padel

4. Altura del balon respecto al extremo inferior del padel

5. Numero de rebotes

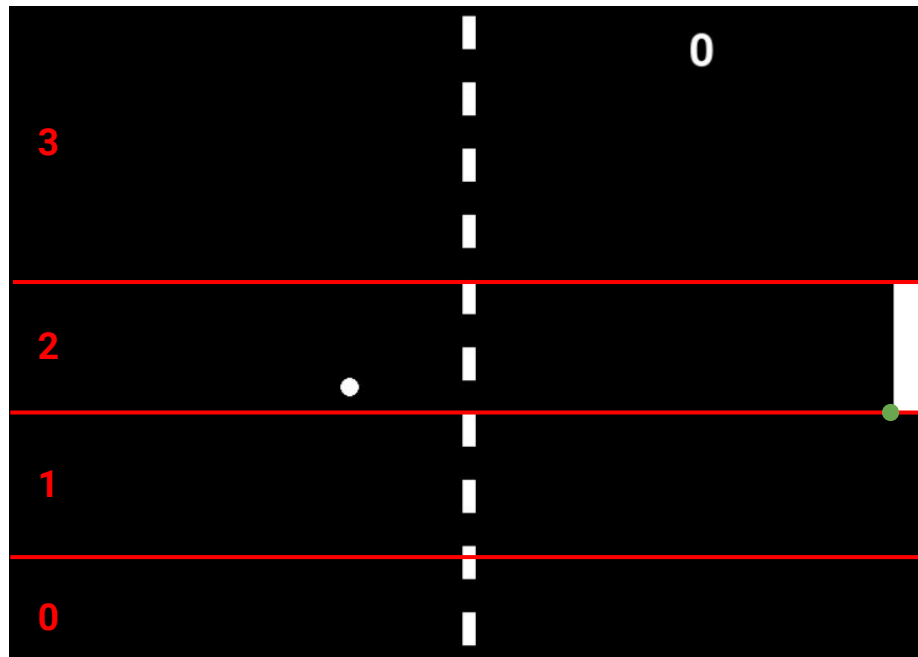
Entre 0 y 3



# Estados de juego

1. Velocidad en el eje Y
2. Proximidad al padel
3. Altura del balon respecto al extremo superior del padel
4. Altura del balon respecto al extremo inferior del padel
5. Numero de rebotes

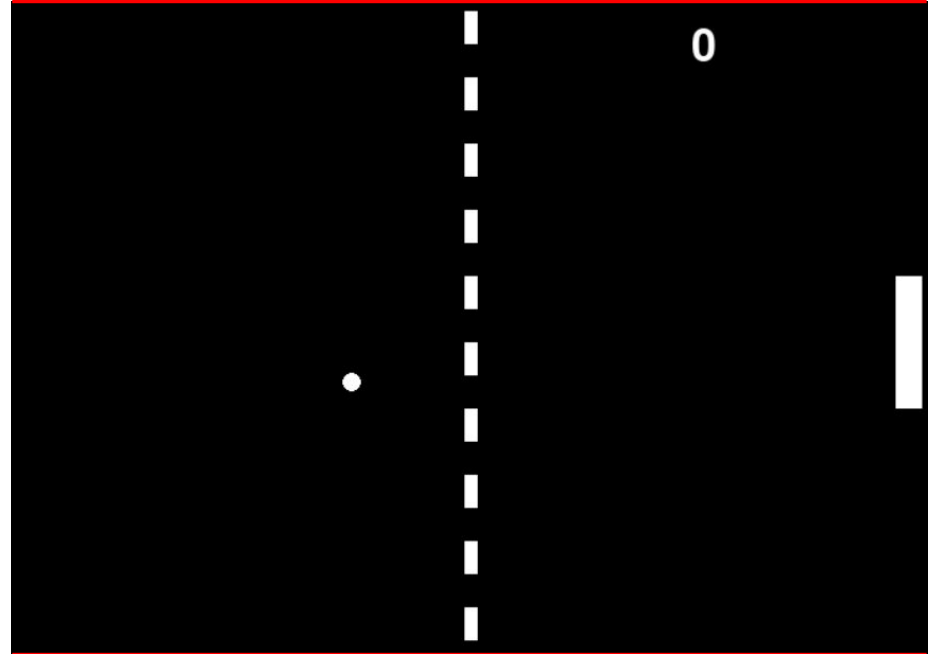
Entre 0 y 3



# Estados de juego

1. Velocidad en el eje Y
2. Proximidad al padel
3. Altura del balon respecto al extremo superior del padel
4. Altura del balon respecto al extremo inferior del padel
5. Numero de rebotes

Entre 0 y 2

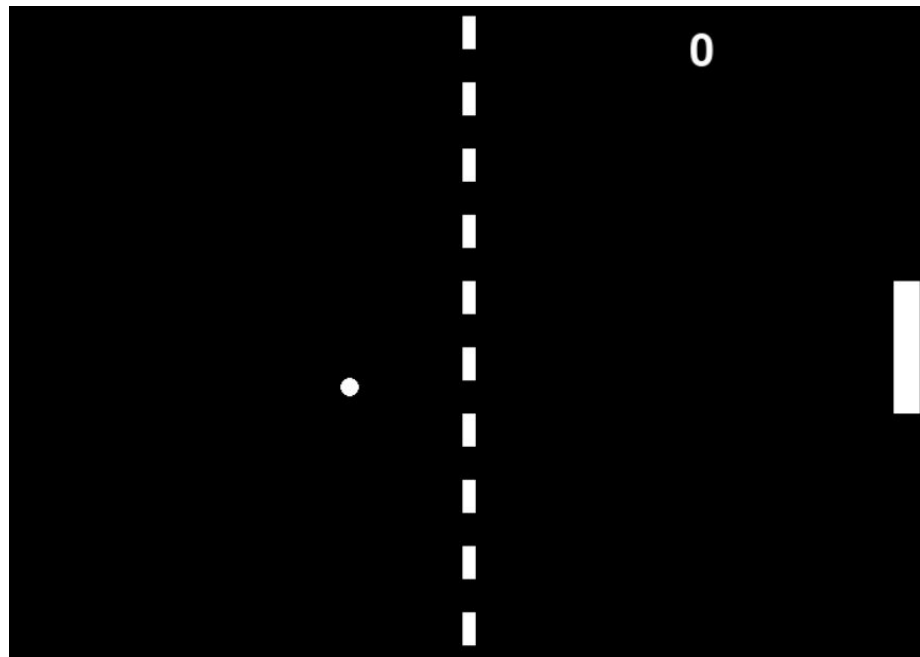




# Estados de juego

1. Velocidad en el eje Y
2. Proximidad al padel
3. Altura del balon respecto al extremo superior del padel
4. Altura del balon respecto al extremo inferior del padel
5. Numero de rebotes

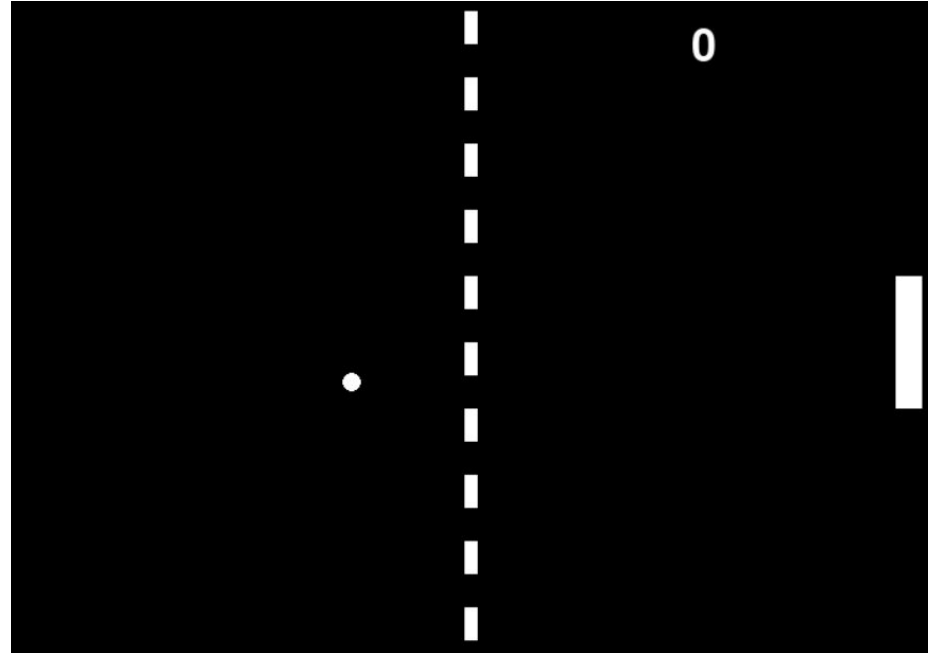
$$11 \cdot 6 \cdot 4 \cdot 4 \cdot 3 = 3168 \text{ filas}$$



# Acciones del Agente

- 0. Arriba
- 1. Mantenerse quieto
- 2. Abajo

```
game.play_step(u)
```



# Q-Table

	0	1	2
0	0	0	0
1	0	0	0
...	...	...	...
3168	0	0	0

# Q-Table

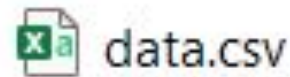
	0	1	2
0	0	0	0
1	0	0	0
...	...	...	...
3168	0	0	0



data.csv

# Q-Table

- El archivo no contiene ni header ni índices (3168 x 7)
- La  $i$ -ésima fila del csv tiene los valores de la  $i$ -ésima fila de la Q-Table (partiendo de 0)
- Las primeras 5 columnas representan el estado actual del entorno (en el orden en que aparecen en enunciado)



						u_0	u_1	u_2
0	-5	0	0	0	0	-87	22	-16
1	-5	0	0	0	1	51	-15	-90
2	-5	0	0	0	2	-62	61	33
3	-5	0	0	1	0	53	-57	-91
4	-5	0	0	1	1	14	28	-64
5	-5	0	0	1	2	-27	-95	58
6	-5	0	0	2	0	100	32	23
7	-5	0	0	2	1	-46	-6	-39
8	-5	0	0	2	2	72	8	-34
9	-5	0	0	3	0	67	6	96
10	-5	0	0	3	1	0	6	26

(...)



PONTIFICIA  
UNIVERSIDAD  
CATÓLICA  
DE CHILE

# Reinforcement Learning

Daniel Florea - IIC2613 Inteligencia Artificial