Aprendizaje por refuerzo

Extracto de MIT 6S191

Tipos de aprendizaje

Supervised Learning

Data: (x, y)
x is data, y is label

Goal: Learn function to map
$x \rightarrow y$

Apple example:

This thing is an apple.

Unsupervised Learning

Data: x
x is data, no labels!

Goal: Learn underlying structure

Apple example:

This thing is like the other thing.

Reinforcement Learning

Data: state-action pairs

Goal: Maximize future rewards over many time steps

Apple example:

Eat this thing because it will keep you alive.

Aprendizaje por refuerzo

Cómo aprendemos a jugar?

Aprendizaje por refuerzo

Action: a move the agent can make in the environment.

Action space A: the set of possible actions an agent can make in the environment

Observations: of the environment after taking actions.

State: a situation which the agent perceives.

Reward: feedback that measures the success or failure of the agent's action.

Q function

$$R_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \cdots$$

Total reward, R_t, is the discounted sum of all rewards obtained from time t

$$Q(s_t, a_t) = \mathbb{E}[R_t | s_t, a_t]$$

The Q-function captures the **expected total future reward** an agent in state, s, can receive by executing a certain action, a

Política

$$Q(s_t, a_t) = \mathbb{E}[R_t | s_t, a_t]$$

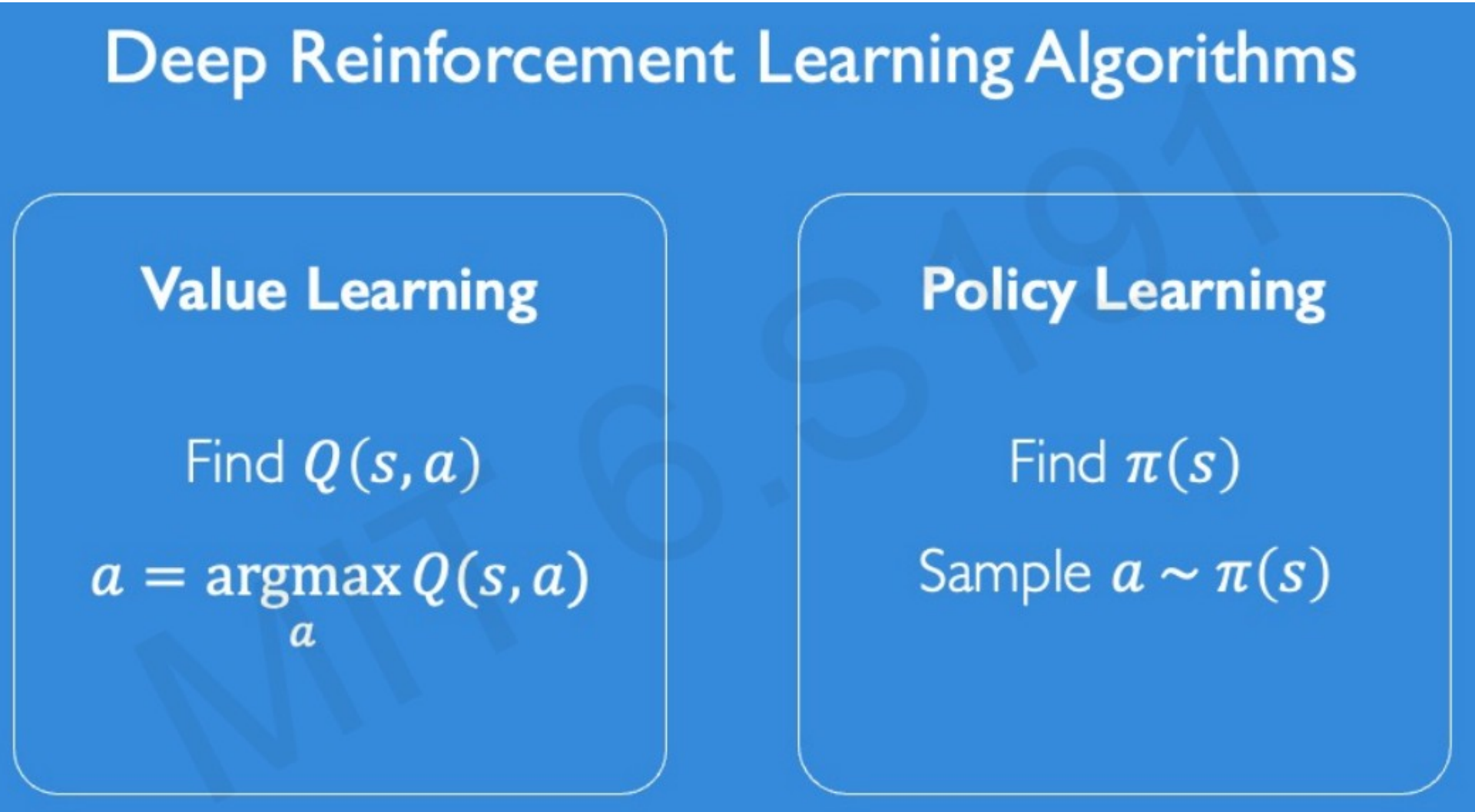
(state, action)

Ultimately, the agent needs a **policy $\pi(s)$**, to infer the **best action to take** at its state, s

Strategy: the policy should choose an action that maximizes future reward

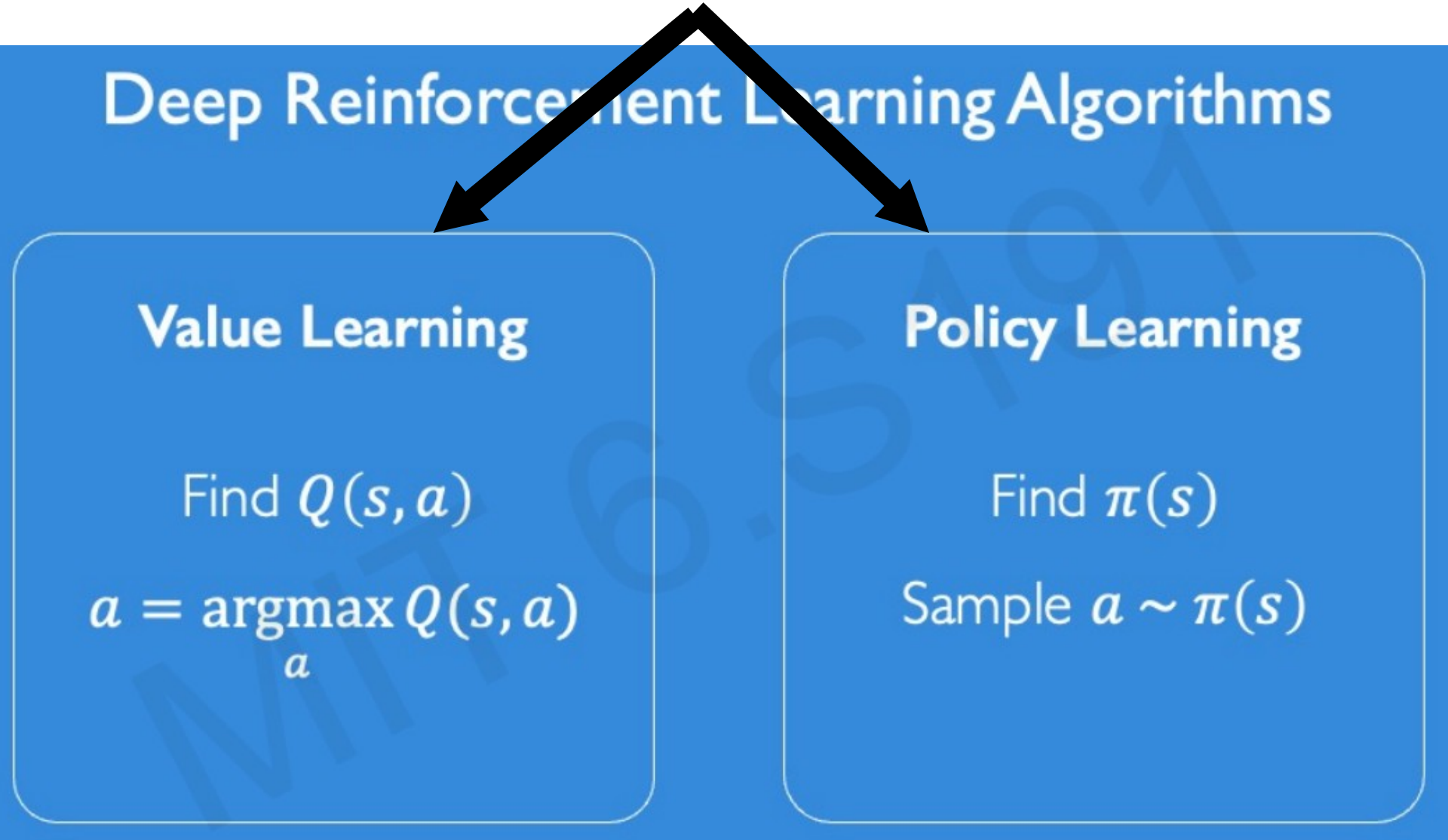
$$\pi^*(s) = \underset{a}{\operatorname{argmax}}\, Q(s, a)$$

Aprendizaje por refuerzo

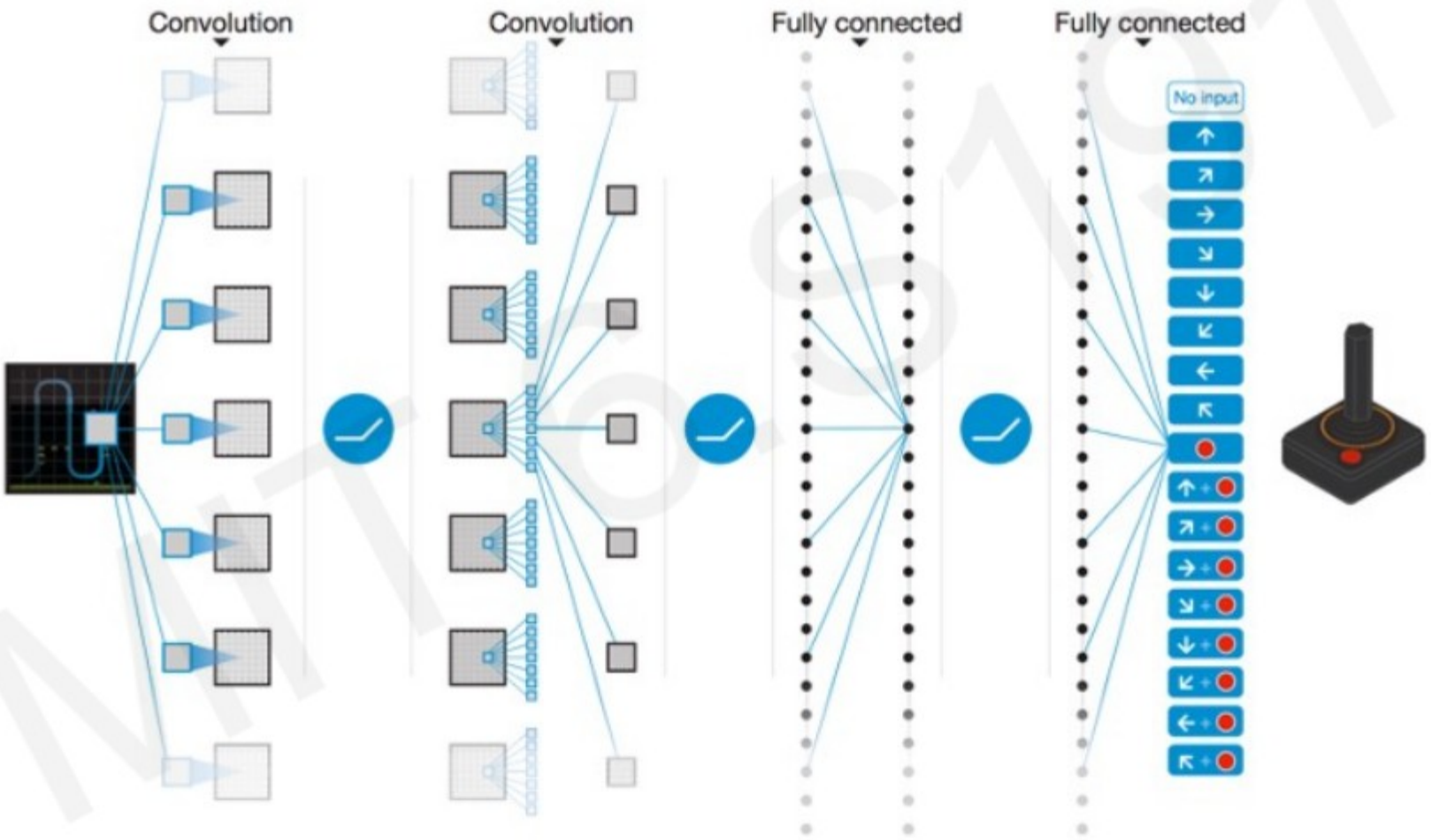


Aprendizaje por refuerzo

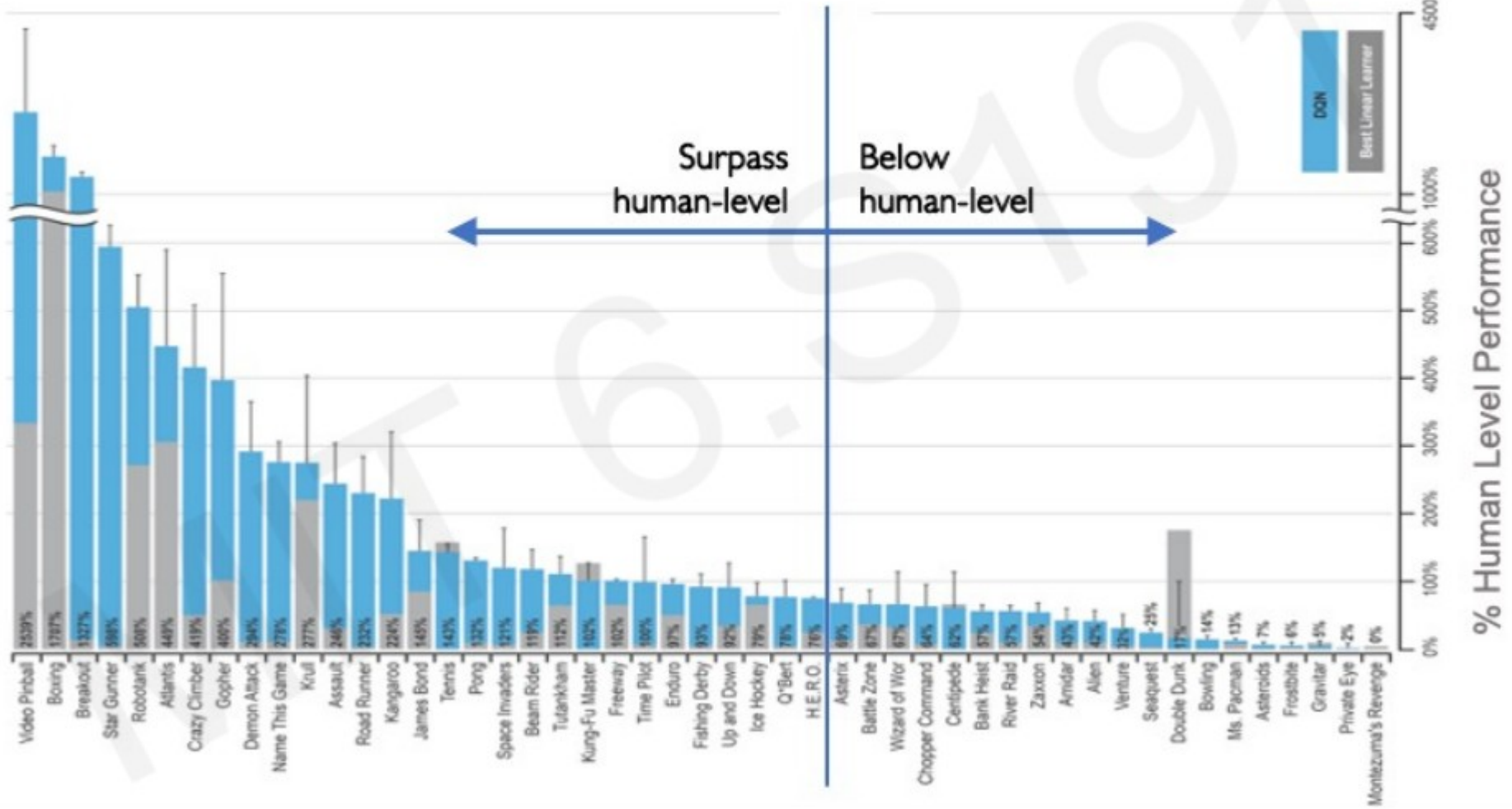
Podemos usar Deep Learning para aprender las dos!



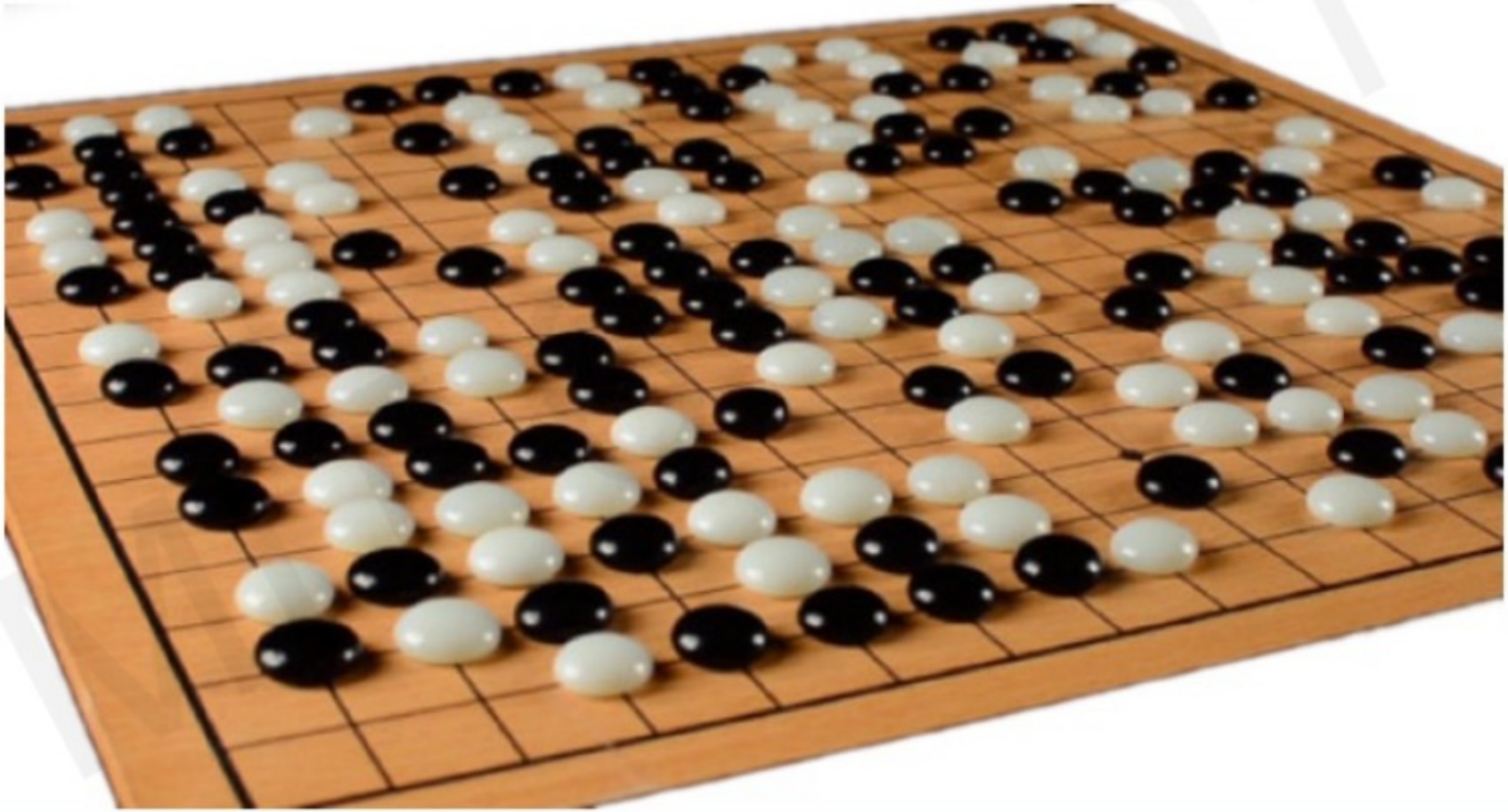
DQN Atari Results



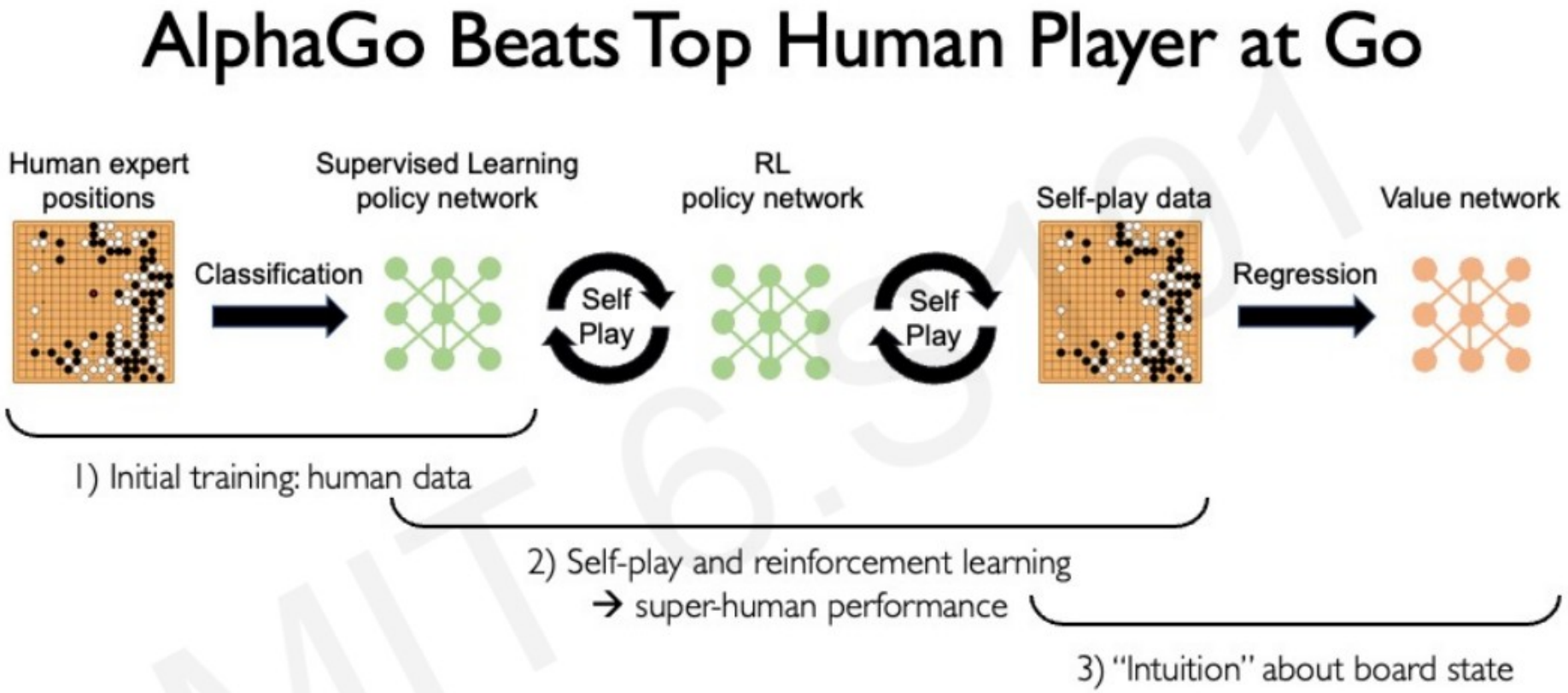
DQN Atari Results



Reinforcement Learning and the Game of Go



Aprendizaje por refuerzo



Aprendizaje por refuerzo