Aprendizaje por refuerzo con Open AI Gym

Ana Laura Diedrichs @anadiedrichs
Link a esta presentación:
https://tinyurl.com/pydatarl

Hola!

- Ph.D. candidate at UTN FRM

- Software engineer -

- IoT ML consultant

- Teacher assistant

- ana.diedrichs@frm.utn.edu.ar
  Web https://anadiedrichs.github.io/
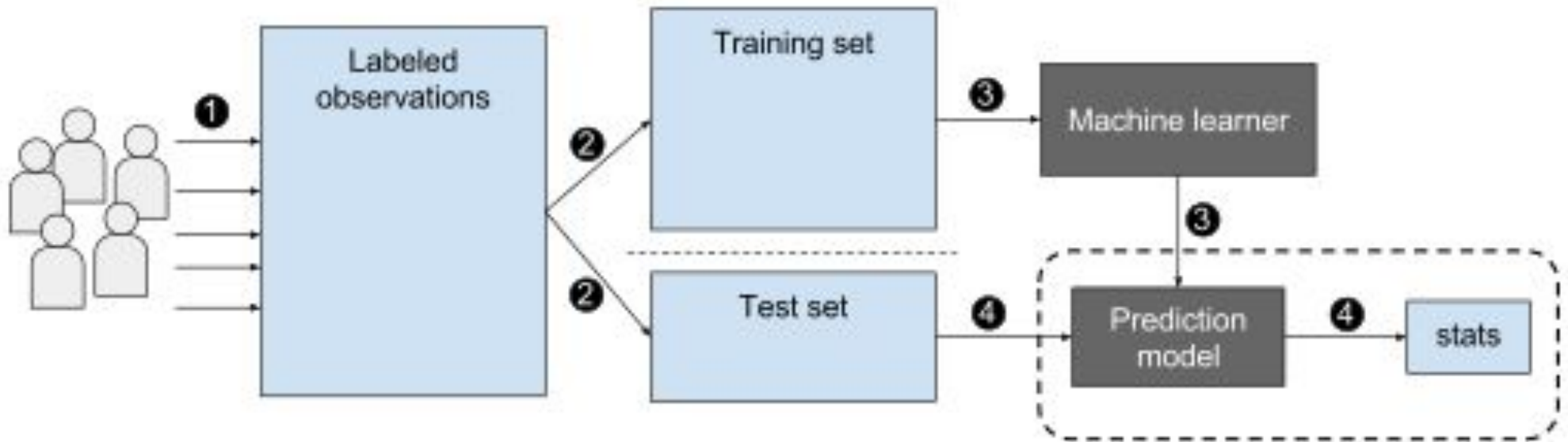 Twitter @anadiedrichs

Motivación

Supervised learning
Aprendizaje supervisado

https://commons.wikimedia.org/wiki/File:Supervised_machine_learning_in_a_nutshell.svg

CLASSICAL MACHINE LEARNING

Data is pre-categorized or numerical

Data is not labeled in any way

SUPERVISED

UNSUPERVISED

Predict a category

Predict a number

Divide by similarity

Identify sequences

CLASSIFICATION

«Divide the socks by color»

CLUSTERING

«Split up similar clothing into stacks»
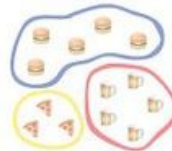
Find hidden dependencies

ASSOCIATION

«Find what clothes I often wear together»
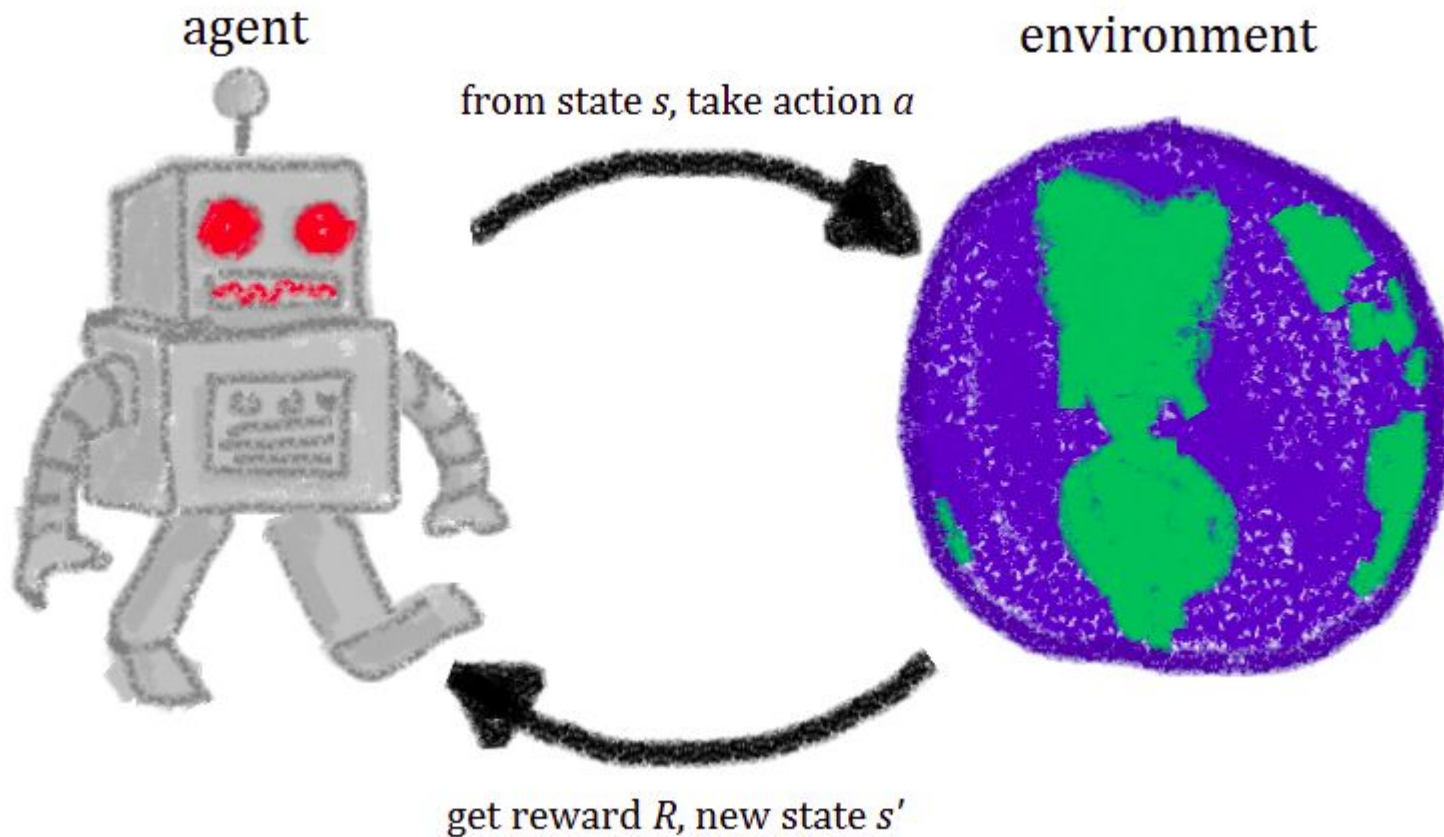
REGRESSION

«Divide the ties by length»

DIMENSION REDUCTION (generalization)

«Make the best outfits from the given clothes»

Reinforcement learning (RL)
Aprendizaje por refuerzo

agent

from state *s*, take action *a*

environment

get reward *R*, new state *s'*

https://commons.wikimedia.org/wiki/File:RI_agent.png

Reinforcement learning (RL)
Aprendizaje por refuerzo

https://en.m.wikipedia.org/wiki/File:Learning_to_walk.png

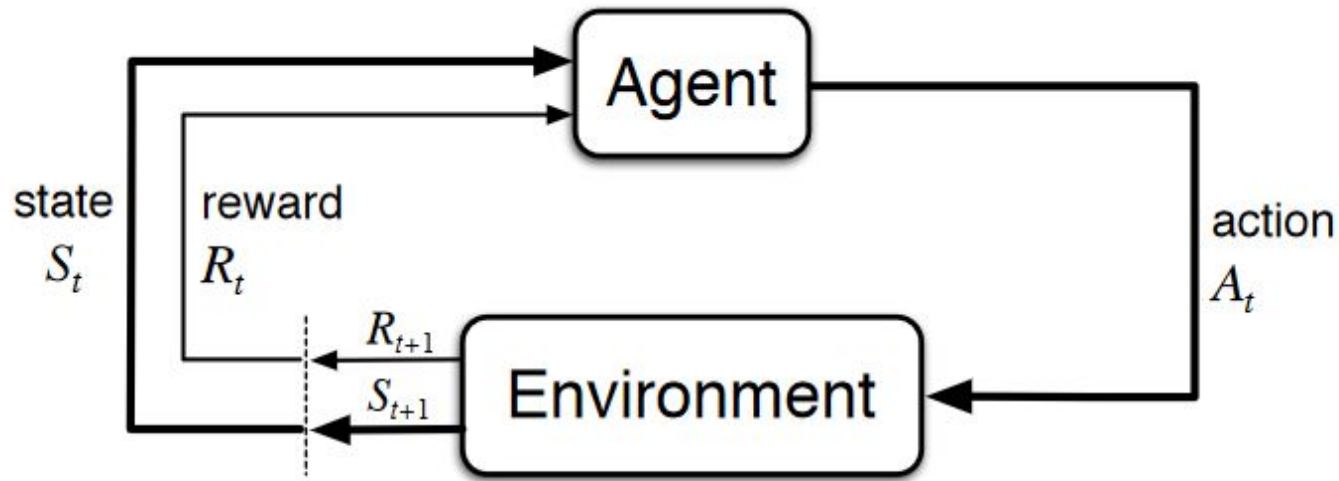# Reinforcement learning (RL)
Aprendizaje por refuerzo

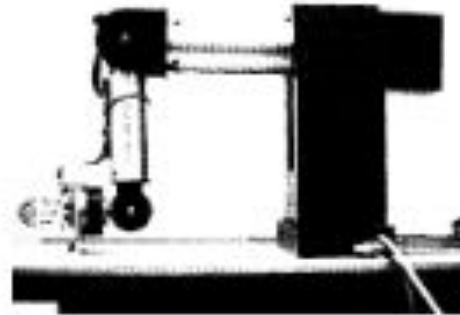Figure 3.1: The agent–environment interaction in a Markov decision process.

Source: Reinforcement Learning:An Introduction

RL applications
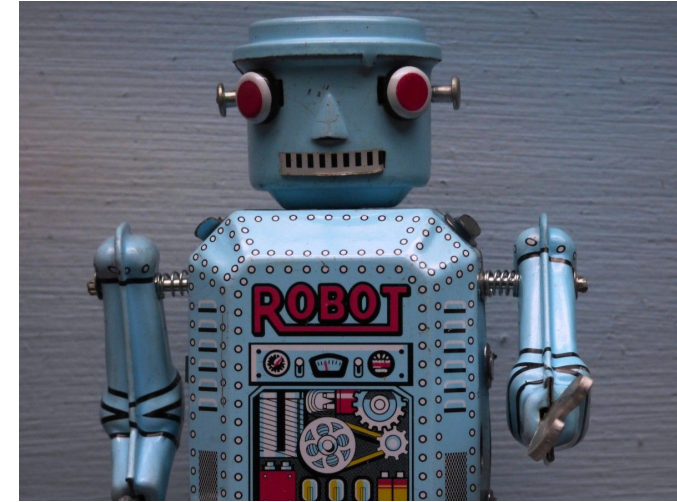# Aplicaciones

(a) OBELIX robot

(b) Zebra Zero robot

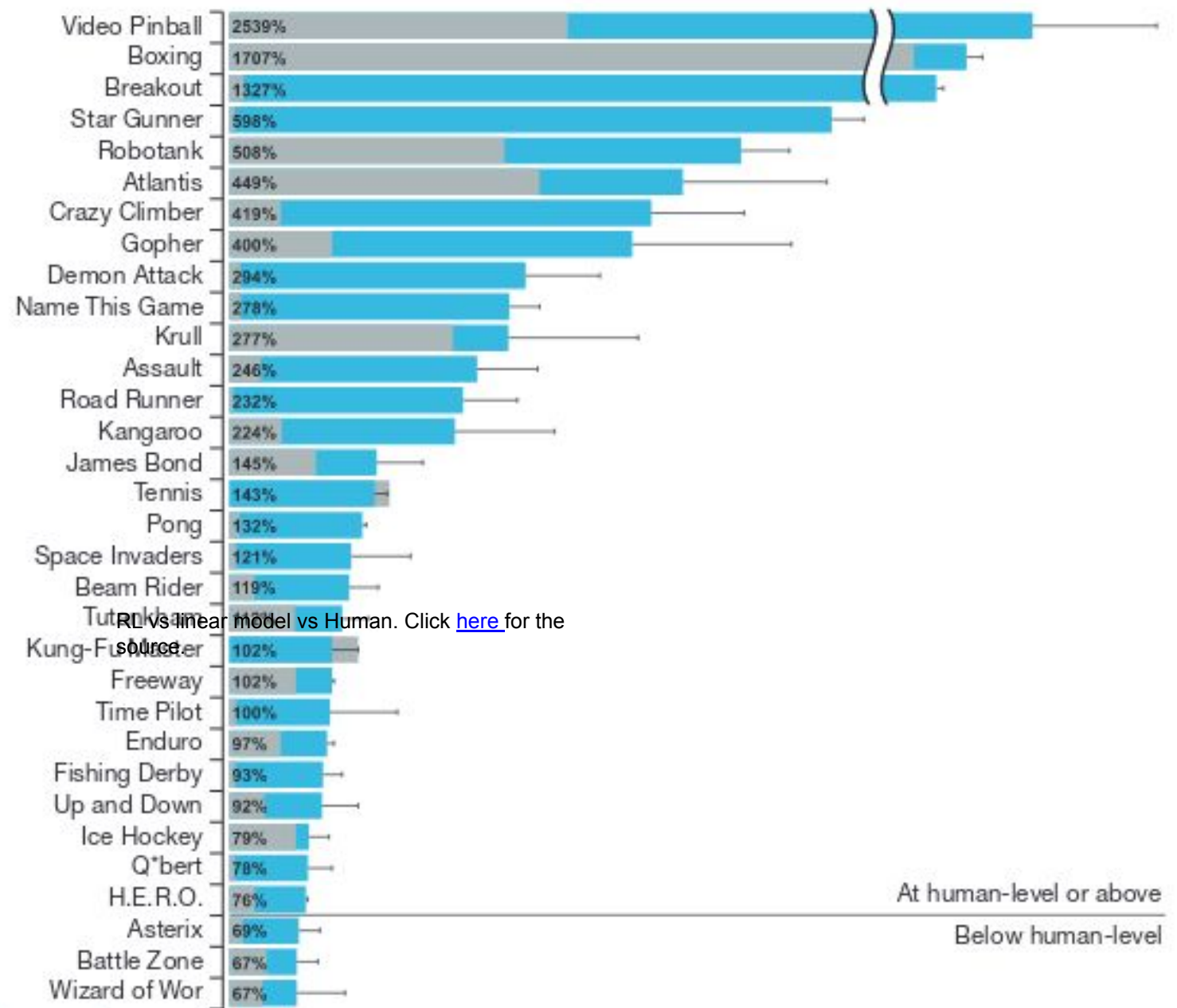(c) Autonomous helicopter

(d) Sarcos humanoid DB

J. Kober, J. A. D. Bagnell, J. Peters. Reinforcement Learning in Robotics: A survey. Int. J. Robot. Res. Jul. 2013.
https://www.ias.informatik.tu-darmstadt.de/uploads/Publications/Kober_IJRR_2013.pdf

RL applications
Aplicaciones

Video Pinball	2539%
Boxing	1707%
Breakout	1327%
Star Gunner	598%
Robotank	508%
Atlantis	449%
Crazy Climber	419%
Gopher	400%
Demon Attack	294%
Name This Game	278%
Krull	277%
Assault	246%
Road Runner	232%
Kangaroo	224%
James Bond	145%
Tennis	143%
Pong	132%
Space Invaders	121%
Beam Rider	119%
Tutankham	
Kung-Fu Master	102%
Freeway	102%
Time Pilot	100%
Enduro	97%
Fishing Derby	93%
Up and Down	92%
Ice Hockey	79%
Q*bert	78%
H.E.R.O.	76%
Asterix	69%
Battle Zone	67%
Wizard of Wor	67%

At human-level or above

Below human-level
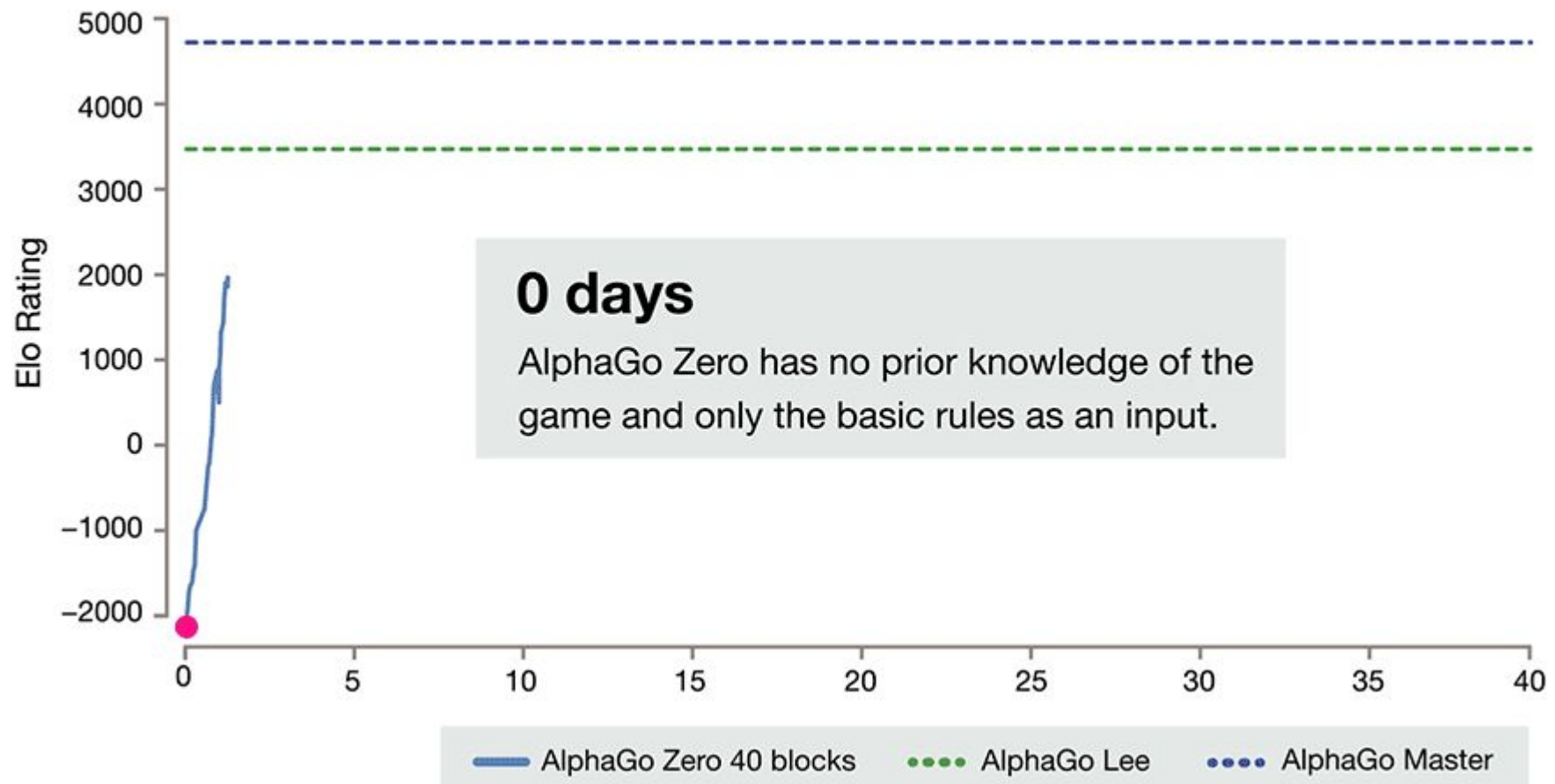
RL vs linear model vs Human. Click here for the source

RL vs linear model vs Human. Click here for the source
https://web.stanford.edu/class/psych209/Readings/MnihEtAlHassibis15NatureControlDeepRL.pdf
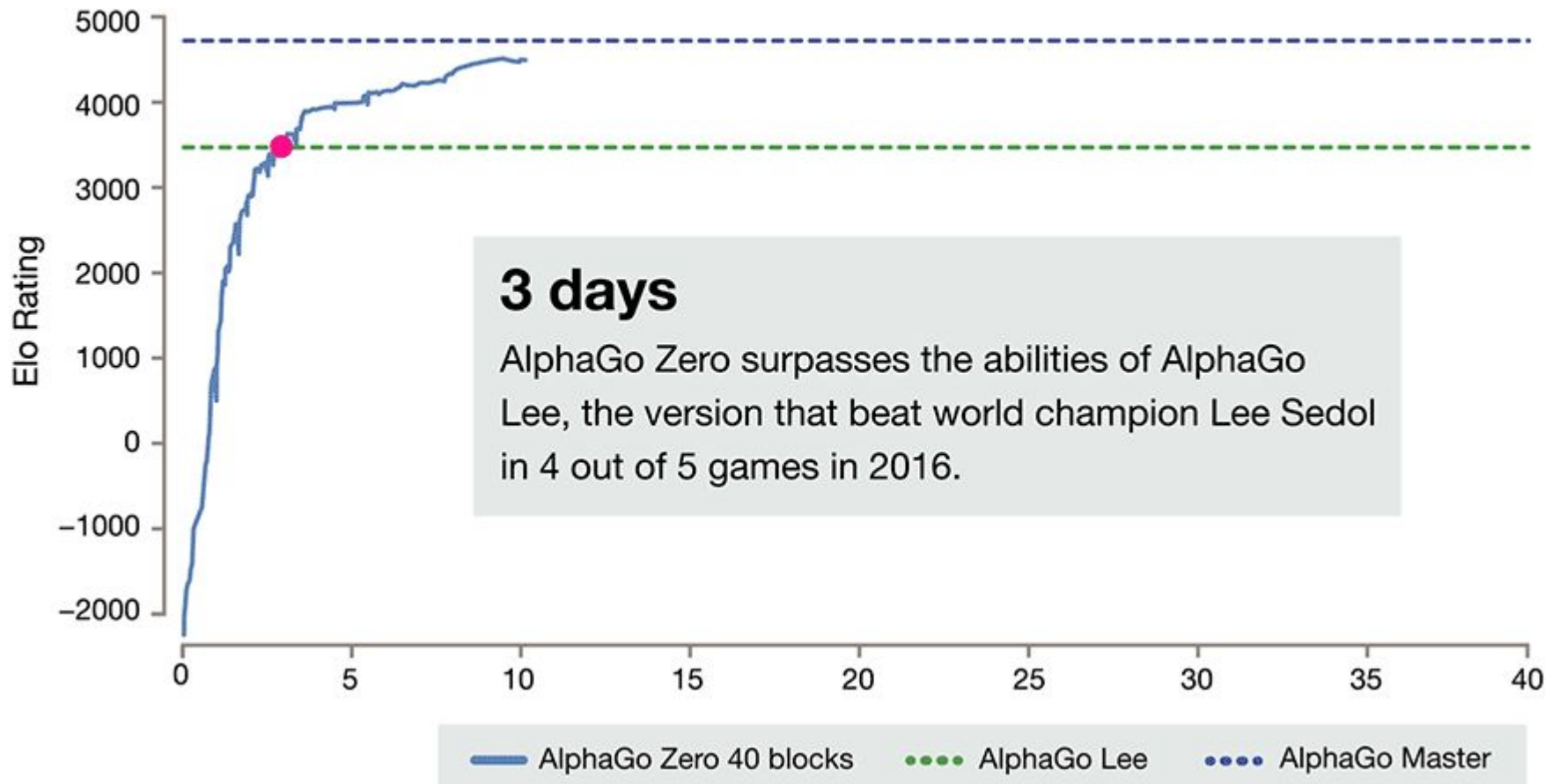
RL applications
Alpha Go zero

https://deepmind.com/blog/article/alphago-zero-starting-scratch

RL applications
Aplicaciones

3 days

AlphaGo Zero surpasses the abilities of AlphaGo Lee, the version that beat world champion Lee Sedol in 4 out of 5 games in 2016.

https://deepmind.com/blog/article/alphago-zero-starting-scratch

RL applications
Alpha Go zero

21 days

AlphaGo Zero reaches the level of AlphaGo Master, the version that defeated 60 top professionals online and world champion Ke Jie in 3 out of 3 games in 2017.

Legend: AlphaGo Zero 40 blocks — AlphaGo Lee ····· AlphaGo Master ·····

https://deepmind.com/blog/article/alphago-zero-starting-scratch

RL applications
Alpha Go zero

40 days

AlphaGo Zero surpasses all other versions of AlphaGo and, arguably, becomes the best Go player in the world. It does this entirely from self-play, with no human intervention and using no historical data.

AlphaGo Zero 40 blocks ••••• AlphaGo Lee ••••• AlphaGo Master

https://deepmind.com/blog/article/alphago-zero-starting-scratch

RL applications
Aplicaciones

Z. Zhou, X. Li, and R. N. Zare. Optimizing Chemical Reactions with Deep Reinforcement Learning. ACSCentral Science3, 2017.

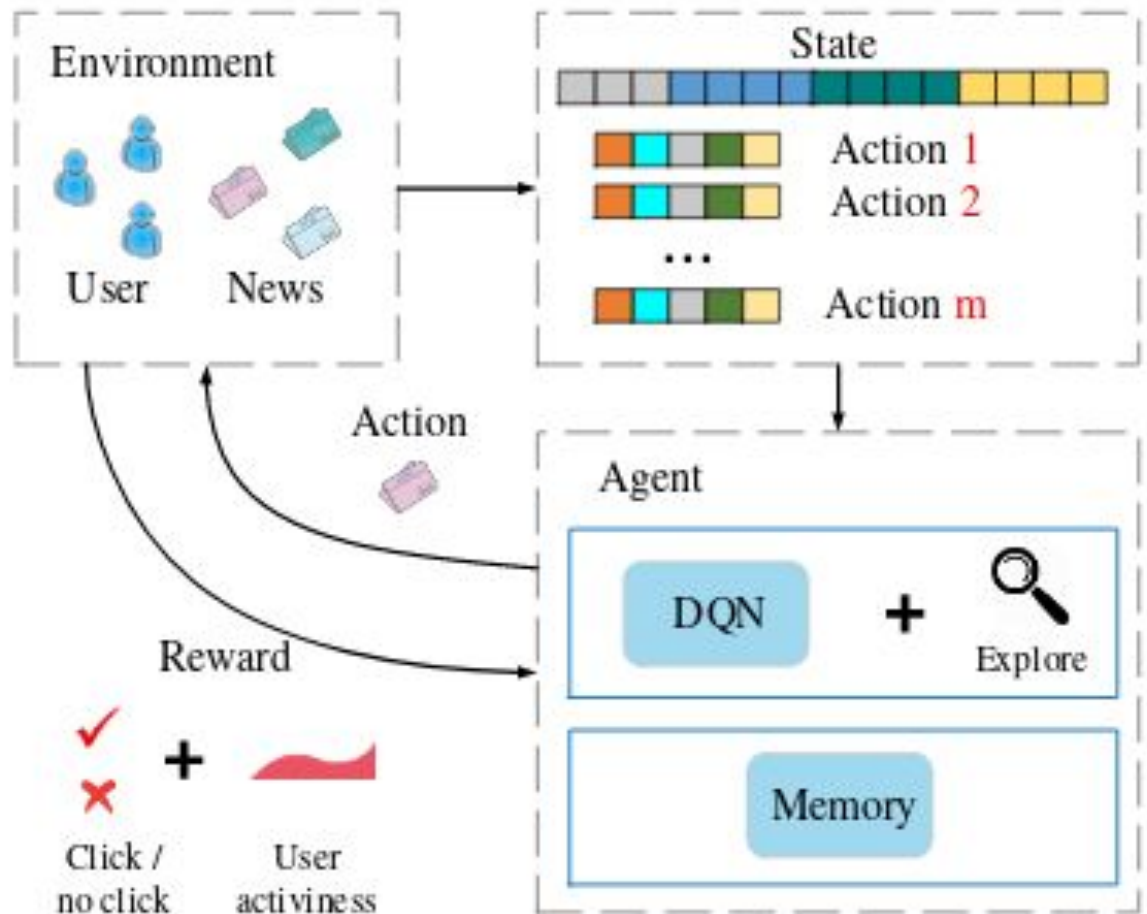https://github.com/lightingghost/chemopt

RL applications
Aplicaciones

Figure 2: Deep Reinforcement Recommendation System

- G. Zheng, F. Zhang, Z. Zheng, Y. Xiang, Ni. J. Yuan, X. Xie, and Z. Li. DRN: A Deep Reinforcement Learning Frameworkfor News Recommendation. 2018.

RL applications
Aplicaciones

- Polydoros, Athanasios S., and Lazaros Nalpantidis. "Survey of model-based reinforcement learning: Applications on robotics." Journal of Intelligent & Robotic Systems 86.2 (2017): 153-173.

- H.Mao, Alizadeh, M. Alizadeh, Menache, I.Menache, and S.Kandula. Resource Management With deep Reinforcement Learning. In ACM Workshop on Hot Topics in Networks, 2016.

- X. Bu, J. Rao, C. Z. Xu. A reinforcement learning approach to online web systems auto-configuration. In Distributed Computing Systems, 2009. ICDCS'09.29th IEEE International Conference on. IEEE , 2019.

- J. Jin, C.Song, H. Li, K. Gai, J.Wang amd W. Zhang. Real-Time Bidding with Multi-Agent Reinforcement Learning in Display Advertising. arXiv preprint arXiv:1802.09756, 2018.

Reproducibility issues
Problemas de reproducibilidad

- How to compare?

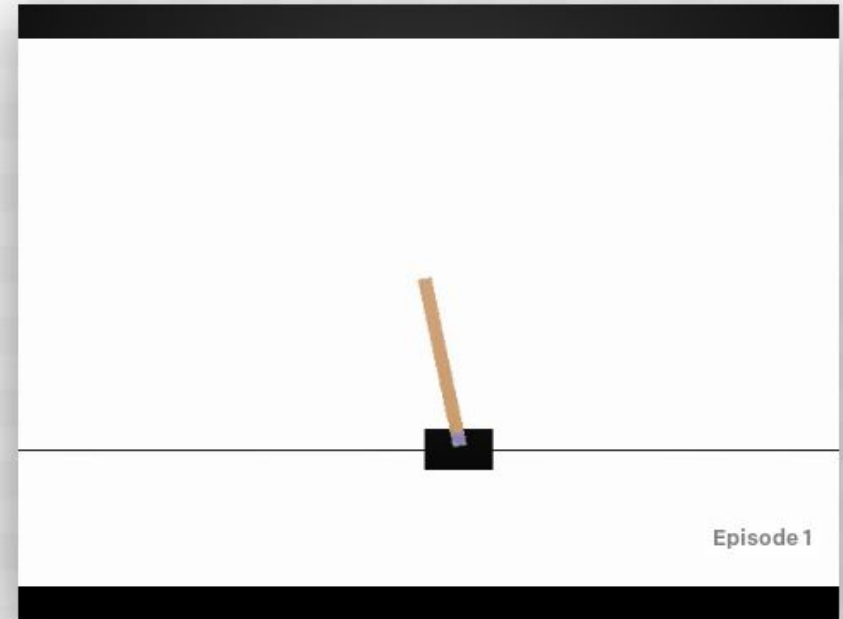- ¿Cómo comparamos? Criterios

Open AI Gym

Gym is a toolkit for developing and comparing reinforcement learning algorithms. It supports teaching agents everything from walking to playing games like Pong or Pinball.

View documentation ›
View on GitHub ›

RandomAgent on CartPole-v1

Episode 1

https://openai.com

https://gym.openai.com/

Box2D
Continuous control tasks in the Box2D simulator.

BipedalWalker-v2
Train a bipedal robot to walk.

BipedalWalkerHardcore-v2
Train a bipedal robot to walk over rough terrain.

CarRacing-v0
Race a car around a track.

LunarLander-v2
Navigate a lander to its landing pad.

LunarLanderContinuous-v2
Navigate a lander to its landing pad.

https://gym.openai.com/

Robotics

Simulated goal-based tasks for the Fetch and ShadowHand robots.

FetchPickAndPlace-v1
Lift a block into the air.

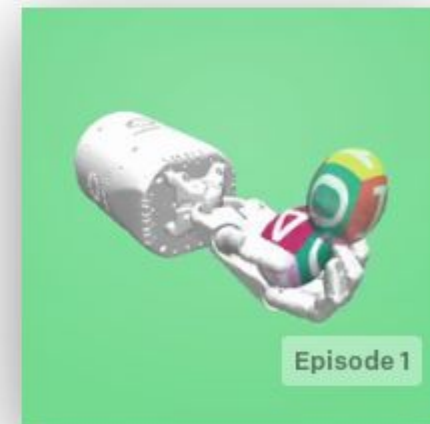**FetchPush-v1**
Push a block to a goal position.

FetchReach-v1
Move Fetch to a goal position.

FetchSlide-v1
Slide a puck to a goal position.

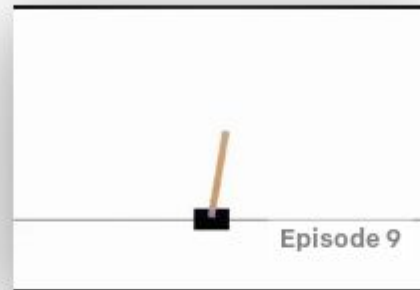**HandManipulateBlock-v0**
Orient a block using a robot hand.

HandManipulateEgg-v0
Orient an egg using a robot hand.

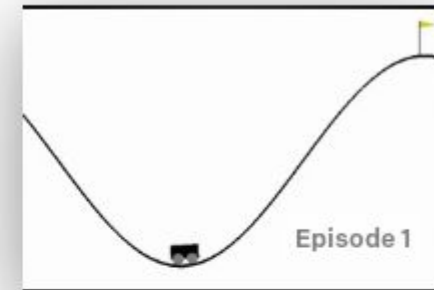https://gym.openai.com/

Classic control
Control theory problems from the classic RL literature.
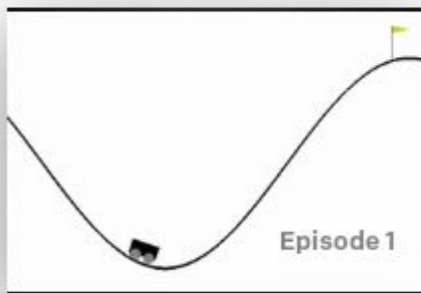
Acrobot-v1
Swing up a two-link robot.

CartPole-v1
Balance a pole on a cart.

MountainCar-v0
Drive up a big hill.

MountainCarContinuous-v0
Drive up a big hill with continuous control.

Pendulum-v0
Swing up a pendulum.

Episode 1
Episode 9
Episode 1

150

100

90   Episode 1
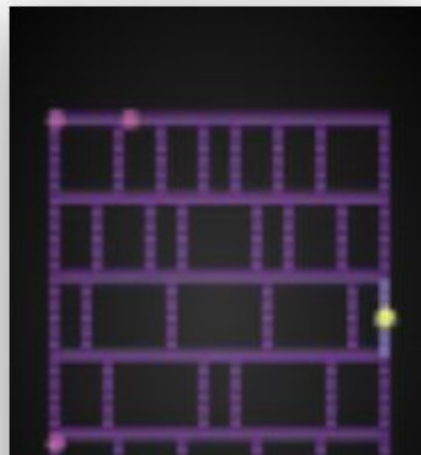
Episode 1   Episode 1

AirRaid-ram-v0
Maximize score in the game
AirRaid, with RAM as input

AirRaid-v0
Maximize score in the game
AirRaid, with screen images
as input

Alien-ram-v0
Maximize score in the game
Alien, with RAM as input

https://gym.openai.com/

Open AI Gym

Open source interface to reinforcement learning tasks.
The gym library provides an easy-to-use suite of reinforcement learning tasks.

```python
import gym
env = gym.make("CartPole-v1")
observation = env.reset()
for _ in range(1000):
  env.render()
  action = env.action_space.sample() # your agent here (this takes random actions)
  observation, reward, done, info = env.step(action)

  if done:
    observation = env.reset()
env.close()
```

https://gym.openai.com/

Demo: pacman aleatorio

- Source:
https://star-ai.github.io/Rendering-OpenAi-Gym-in-Colaboratory/

- https://colab.research.google.com/drive/19cNaxTcwQ-UjAHmjoTDQVtbSSROGG616

Demo: taxi environment

- Taxi - Q-learning
- [https://colab.research.google.com/drive/1mNn9Ij5u e0W66gjn4_fnK6BlPLKHg8wt](https://colab.research.google.com/drive/1mNn9Ij5ue0W66gjn4_fnK6BlPLKHg8wt)
- Frozen lake iteración de valores

  https://colab.research.google.com/drive/12LVBVI_C 4TrNDV2bMgW-fnuWPeoCrure

Q-learning

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

Initialize Q table
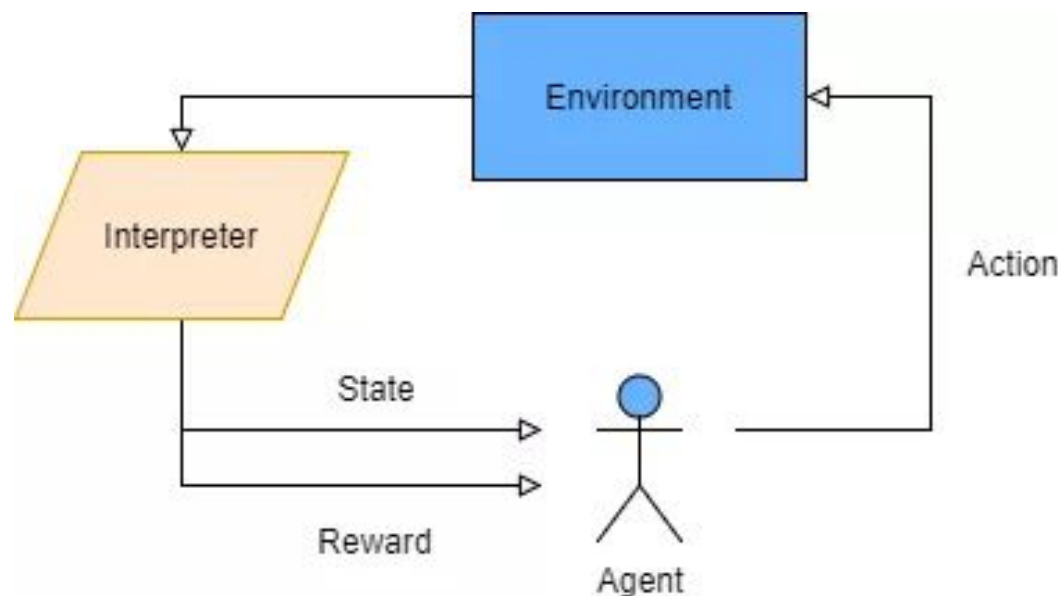
↓

Choose an action a

↓

Perform action

↓

Measure reward
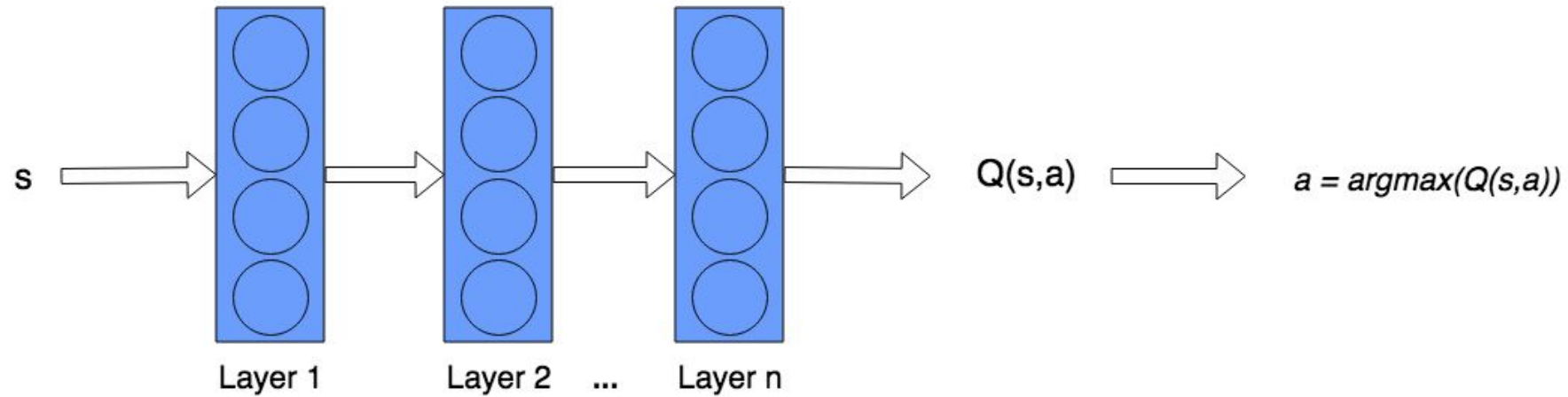
↓

Update Q

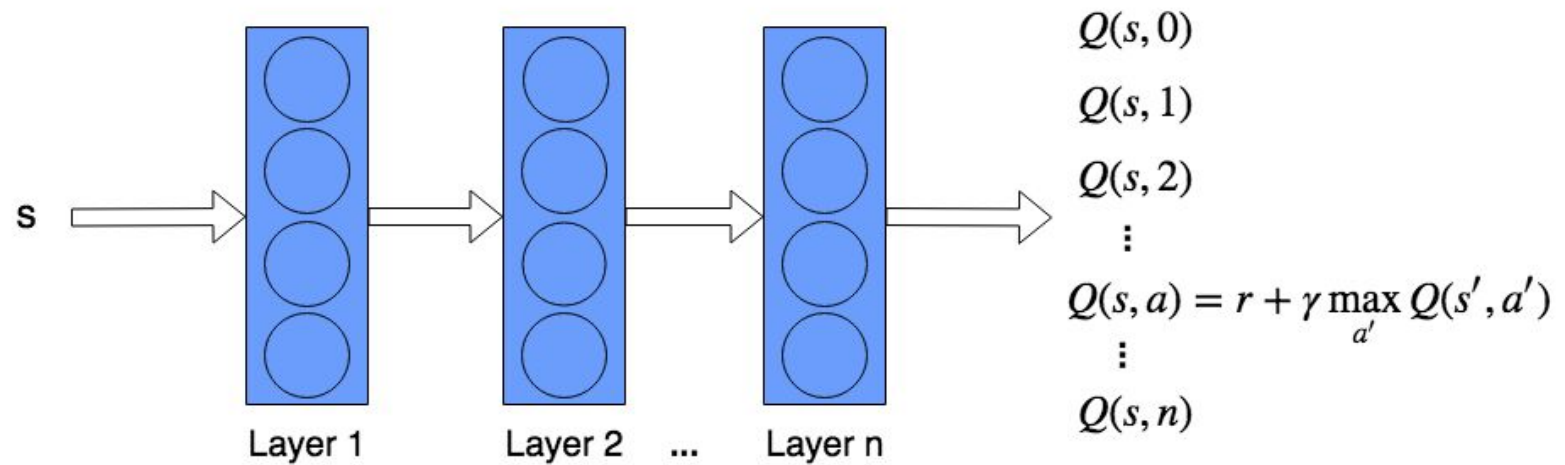At the end of the training

🎉 Good Q*table 🎉

Environment

Interpreter

Action

State

Reward

Agent

Deep RL

Action selection step

s → Layer 1 → Layer 2 ... Layer n → Q(s,a) → $a = argmax(Q(s,a))$

Training step

s → Layer 1 → Layer 2 ... Layer n →

$Q(s,0)$

$Q(s,1)$

$Q(s,2)$

$\vdots$

$Q(s,a) = r + \gamma \max_{a'} Q(s',a')$

$\vdots$

$Q(s,n)$

¿Aplicar RL en mi proyecto?
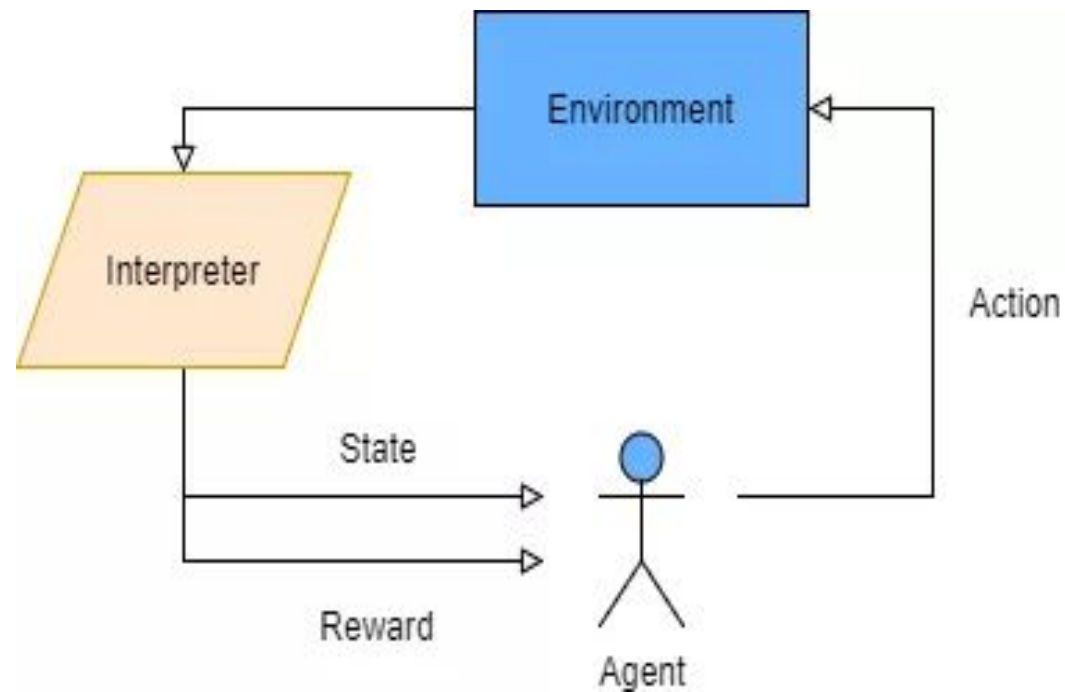
- May I use RL in my project?

  Understand, study

  your

  problem!

¿Aplicar RL en mi proyecto?

RL requirements

- Que debo tener en cuenta antes de aplicar RL

  – MDP definition: definir acciones, estados, recompensa, etc

  – Simulated environment: simular el entorno del agente.

Recursos para aprender

- Implementation of Reinforcement Learning Algorithms.https://github.com/dennybritz/reinforcement-learning

- A curated list of resources dedicated to reinforcement learning.https://github.com/aikorea/awesome-rl

- Experimentos con RL: SLM lab https://kengz.gitbooks.io/slm-lab/content/

- Ejemplo en robótica con Arduino https://www.michaelang.com/project/sisyphus

- https://medium.com/@robertmoni_66330/reinforcement-learning-algorithms-an-intuitive-overview-of-existing-algorithms-c2095902867a

- Notebook RL Qlearning ejemplo curso https://colab.research.google.com/drive/1fI5bZtO4xhJNR34Zayyrpqq2vkeGug6y

¡Muchas gracias PyData!

- Give me feedback :-) [en este formulario](#)
- ¿Preguntas?

- Ana Laura Diedrichs @anadiedrichs

- Ana.diedrichs@frm.utn.edu.ar