

# Matemáticas especiales: Proyecto final

Por Simón Patiño Idarrága





01 { ..

¿Cómo regular la  
velocidad de un motor  
DC?



# ¿Qué es un encoder?



Figura 1: motor con encoder integrado

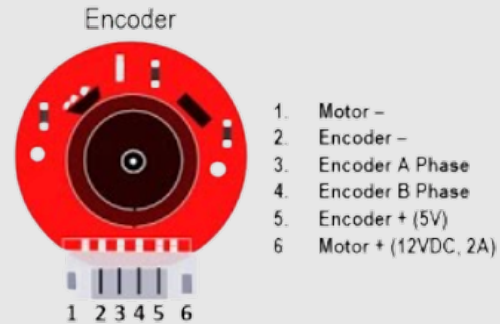


Figura 2: esquema del encoder



# ¿Puede un microcontrolador comunicarse con el motor?

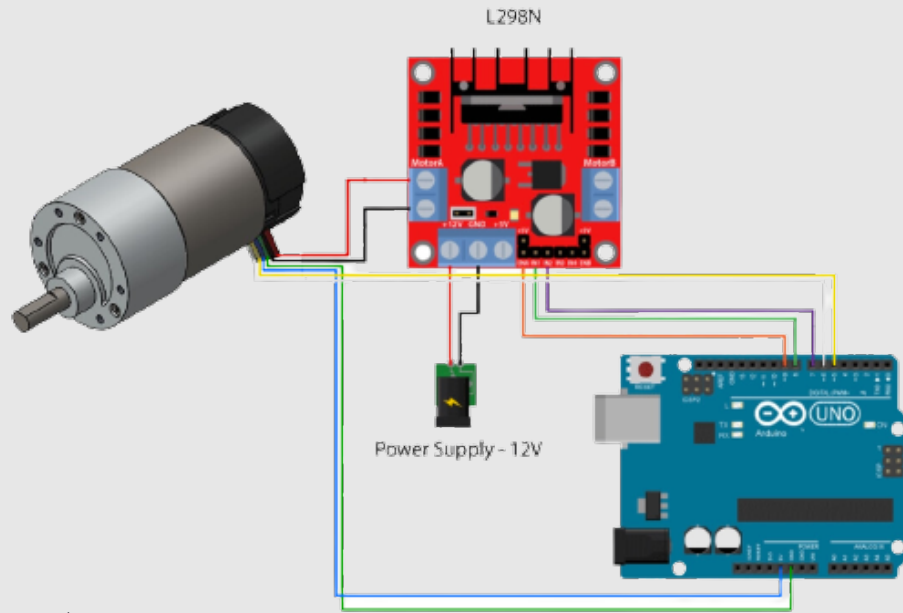


Figura 3: esquemático de conexiones

# ¿Puede Python comunicarse con el motor?

Python:  
comunicación  
serial

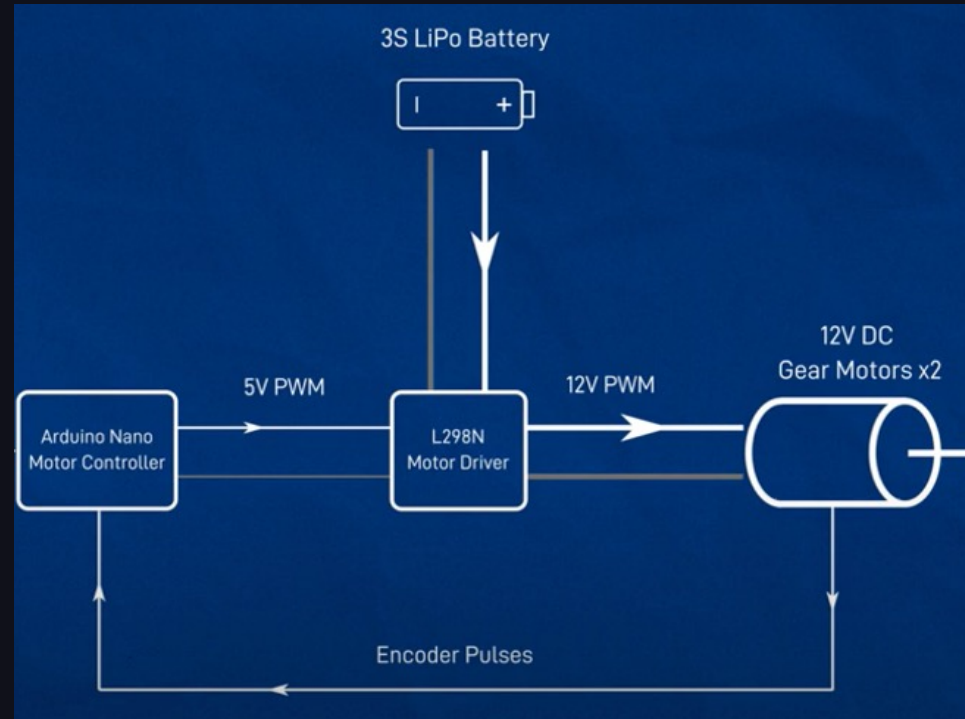


Figura 4: esquema del proceso de comunicación



02{

¿Qué algoritmo me lo permite?



# Control PID:

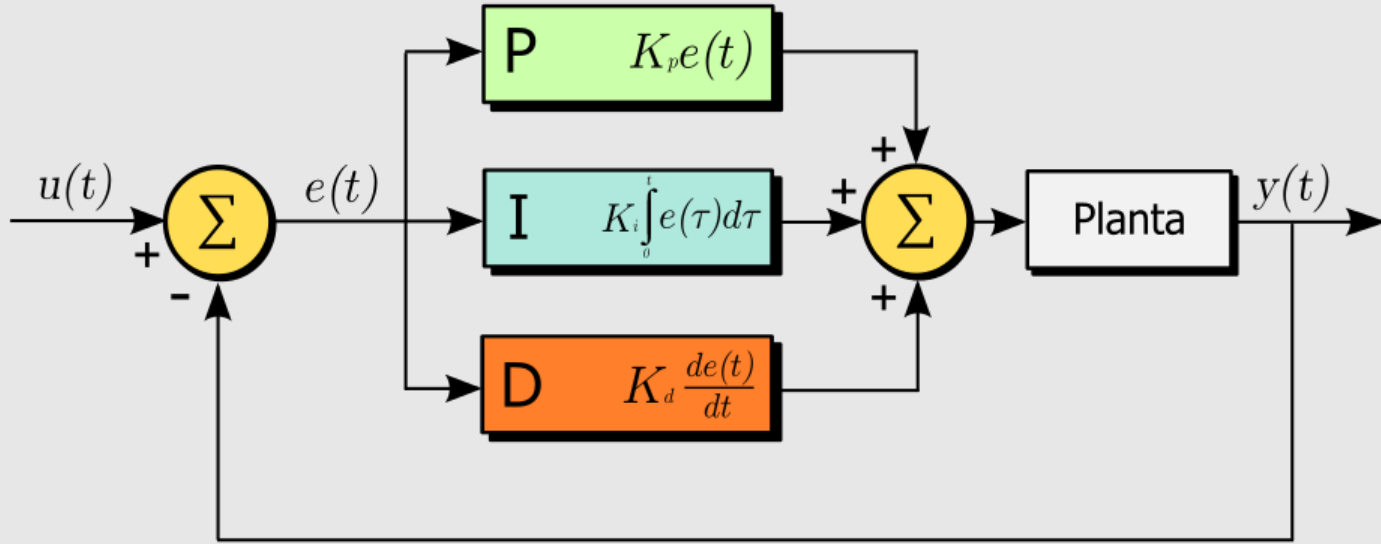


Figura 5: ciclo de un proceso PID



Ya sabemos como funciona. . .





# ¿Y si usamos *machine learning*?

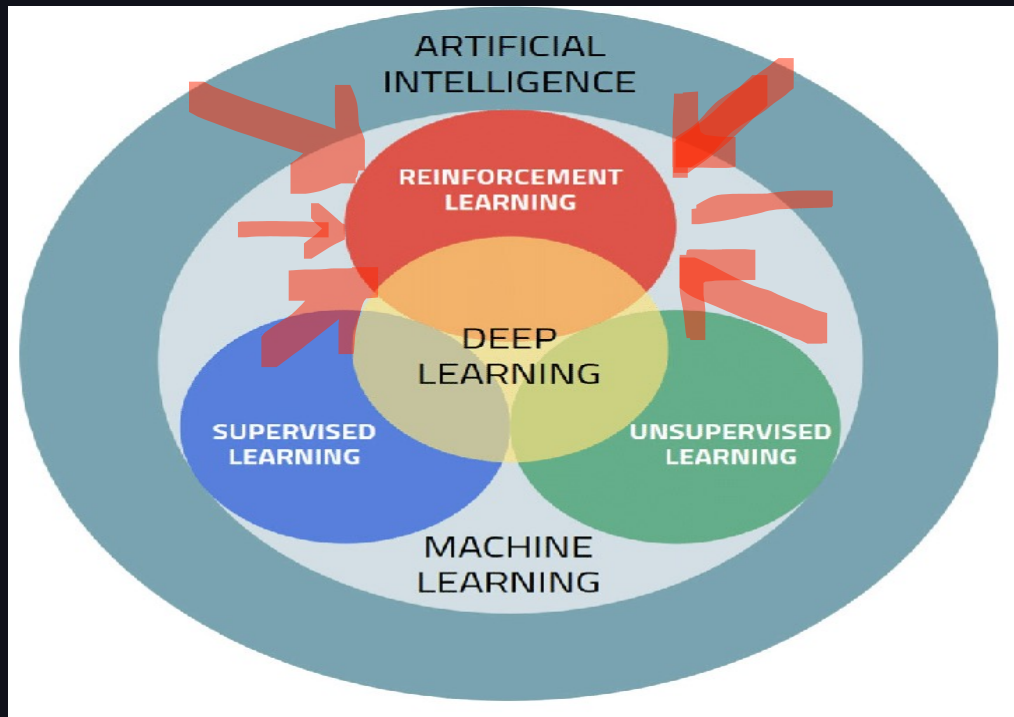


Figura 6: categorías de inteligencia artificial



# Reinforcement learning(RL)

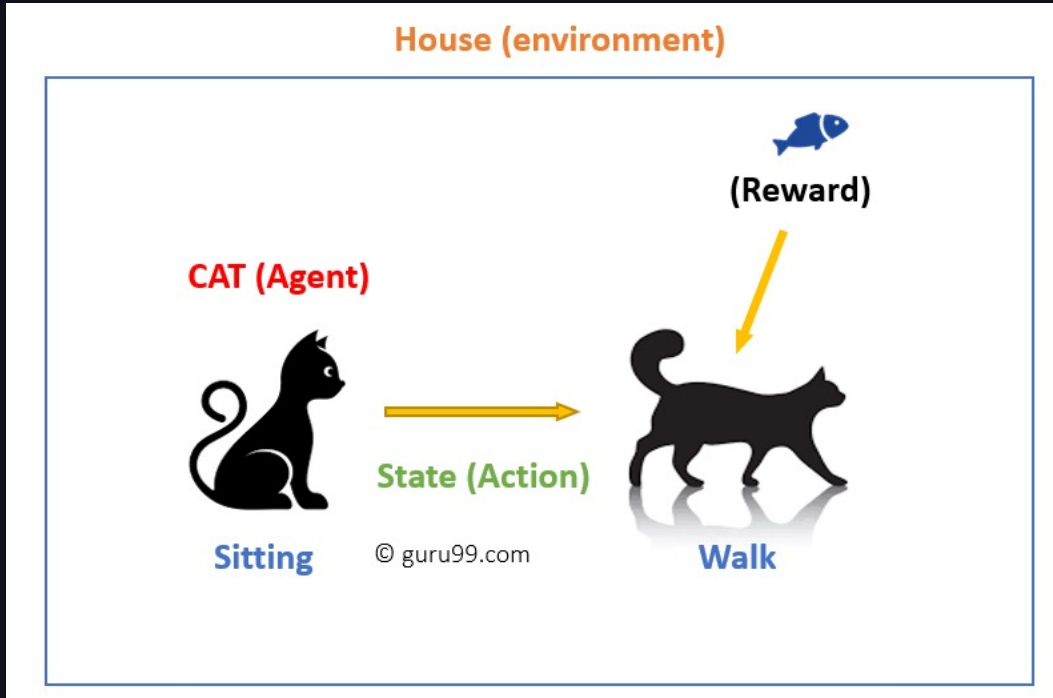


Figura 7: ejemplo de aprendizaje supervisado

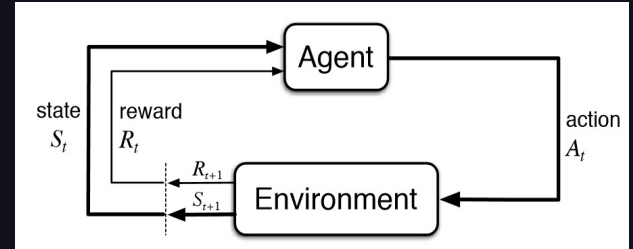


Figura 8: ciclo de un proceso RL



# Q-Learning

El objetivo del *Q-learning* es aprender una serie de normas que le diga a un agente qué acción tomar bajo qué circunstancias (estados). Funciona siempre y cuando se cumpla la propiedad de **Markov**.

La propiedad de Markov nos muestra que el futuro es independiente del pasado, dado el presente, lo cual se expresa en la siguiente formula:

$$\mathbb{P}[S_{t+1}|S_t] = \mathbb{P}[S_{t+1}|S_1, \dots, S_t]$$

La cual significa que el estado actual (representado por  $S_t$ ) contiene toda la información relevante de los estados pasados ( $S_1, \dots, S_t$ ), por lo tanto ya no nos serviría tener mayor información de los estados pasados.

**Figura 9:** explicación de la propiedad de Markov por Miguel Silvia, científico de datos.



# Q-Table

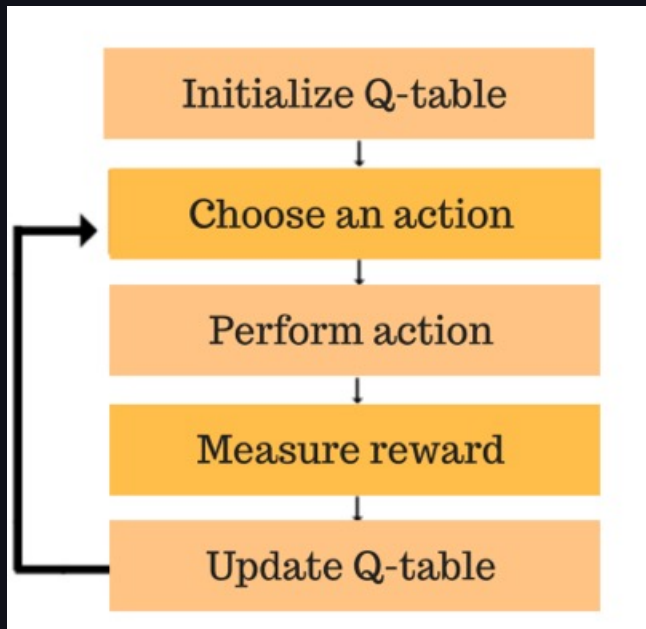


Figura 10: proceso para entrenar mi algoritmo de Q-Learning

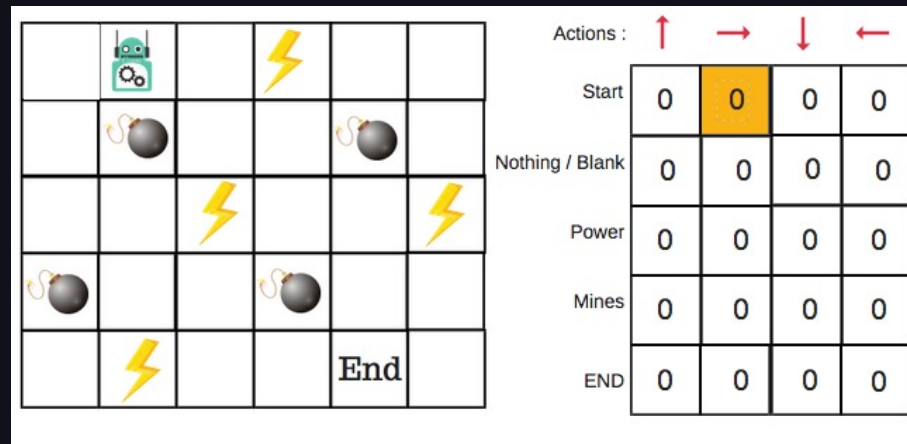


Figura 11: ejemplo práctico

# Q-Table

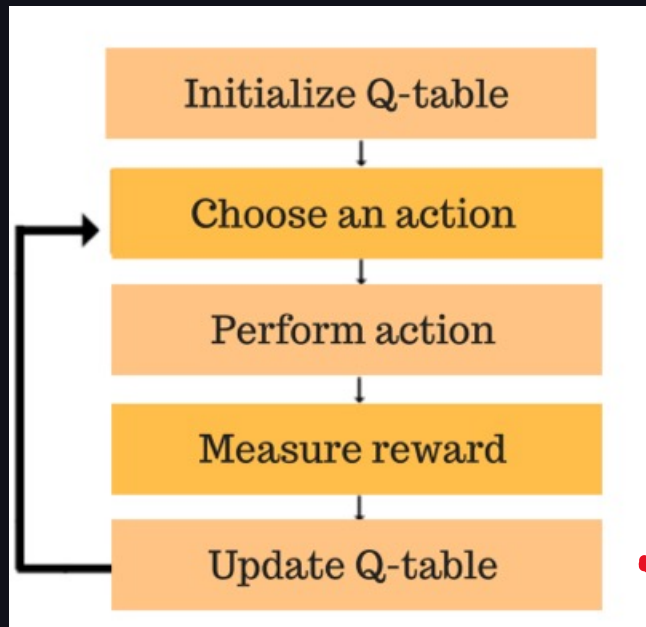


Figura 10: proceso para entrenar mi algoritmo de Q-Learning

New Q(Fila, columna)

$$\text{New } Q(s,a) = Q(s,a) + \alpha [R(s,a) + \gamma \max_{a'} Q'(s',a') - Q(s,a)]$$

- New Q Value for that state and the action
- Learning Rate
- Reward for taking that action at that state
- Current Q Values
- Maximum expected future reward given the new state (s') and all possible actions at that new state.
- Discount Rate

Figura 12: algoritmo para actualizar la Q-Table (ecuación de Bellman)

# Nuestro problema

{

	Aumentar la señal	Disminuir la señal	No hacer nada
0	0	0	0
...	0	0	0
255	0	0	0

}

Figura 13: Q-Table propuesta para regular la velocidad del motor



# Conclusiones

- \* Aún quedan muchos procesos por *optimizar* del proyecto, tanto a nivel de hardware como de software .
- \* Aunque los primeros *papers* que hablan de *Reinforcement learning* fueron publicados alrededor de 1980, *no ha sido muy popular* su aplicación al control de sistemas cerrados.
- \* El gran desarrollo y creciente *popularidad* de Python, ha traído consigo la facilidad de retomar el estudio muchas técnicas de *machine learning*

# Conclusiones



VS





# Recursos

Estas diapositivas tienen un fin únicamente educativo y todas las imágenes utilizadas están libres de derechos de Copyright.

## Bibliografía:

- Baeldung. (2023, 1 de marzo). Epsilon-Greedy Q-Learning in C#. Recuperado de <https://www.baeldung.com/cs/epsilon-greedy-q-learning>
- FreeCodeCamp. (2018, 3 de septiembre). An Introduction to Q-Learning: Reinforcement Learning. Recuperado de <https://www.freecodecamp.org/news/an-introduction-to-q-learning-reinforcement-learning-14ac0b4493cc/>
- Towards Data Science. (2019, 14 de noviembre). A Beginner's Guide to Q-Learning. Recuperado de <https://towardsdatascience.com/a-beginners-guide-to-q-learning-c3e2a30a653c#:~:text=Q%2Dlearning%20Simple%20Example&text=Let%27s%20say%20an%20agent%20has,boundary%20covered%20by%20the%20obstacles.>
- Towards Data Science. (2019, marzo 18). Simple Reinforcement Learning: Q-learning. Recuperado de <https://towardsdatascience.com/simple-reinforcement-learning-q-learning-fcddc4b6fe56>
- SmartLab AI. (2019, 18 de febrero). Reinforcement Learning Algorithms: An Intuitive Overview. Recuperado de <https://smartlabai.medium.com/reinforcement-learning-algorithms-an-intuitive-overview-904e2dff5bbc>
- Silvia, M. (2019, 30 de mayo). Aprendizaje por Refuerzo: Procesos de Decisión de Markov - Parte 1. Aprendizaje por Refuerzo. Recuperado de <https://medium.com/aprendizaje-por-refuerzo-introducción-al-mundo-del/aprendizaje-por-refuerzo-procesos-de-decisión-de-markov-parte-1-8a0aed1e6c59>