Perceptrón

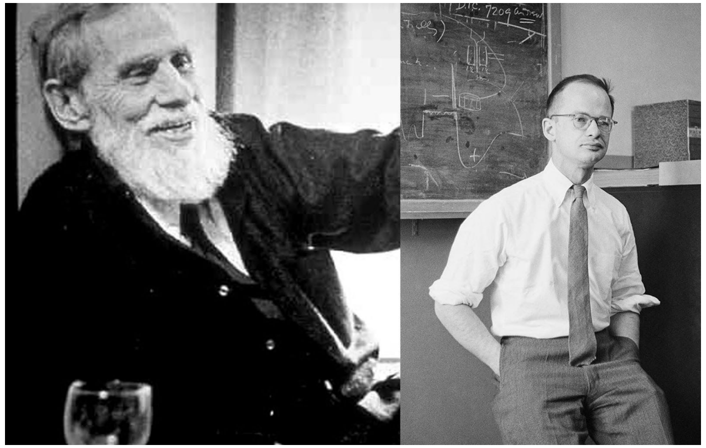
Profesor:
David Campoy Miñarro

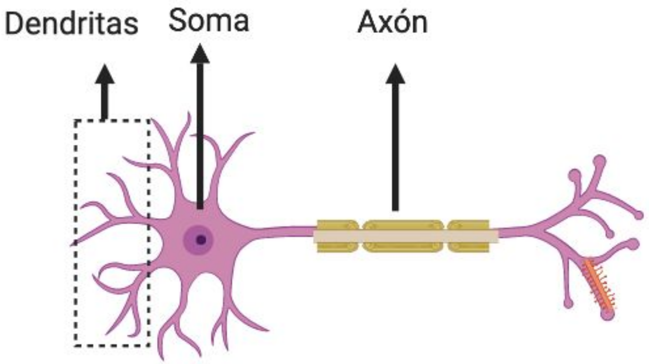
iesperemariaorts

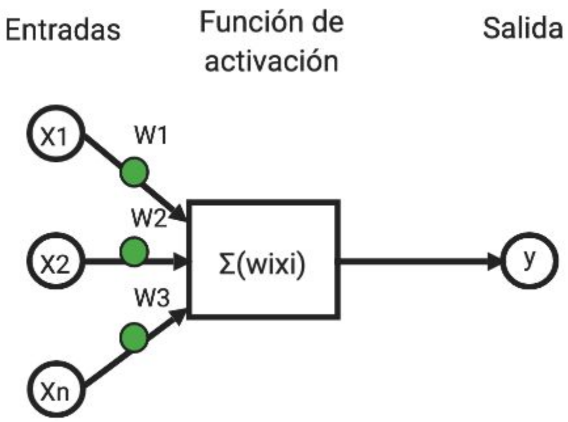
La idea de la neurona artificial

En 1943, McCulloch y Pitts propusieron un modelo de neurona artificial que imitaba aspectos físicos de las neuronas naturales.

Este modelo se basaba en conexiones de entrada, pesos relativos y una entrada adicional permanente llamada "bias" o polarización.





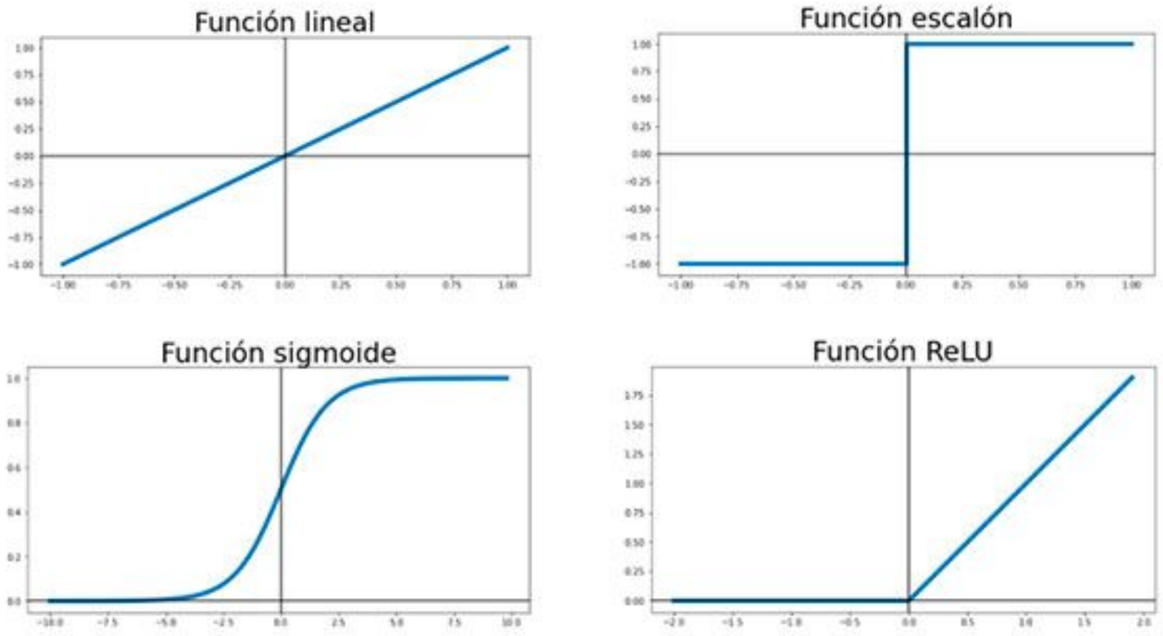


https://www.eldiario.es/hojaderouter/ciencia/walter-pitts-mcculloch-pioneros-cibernetica-inteligencia-artificial_1_4320200.html

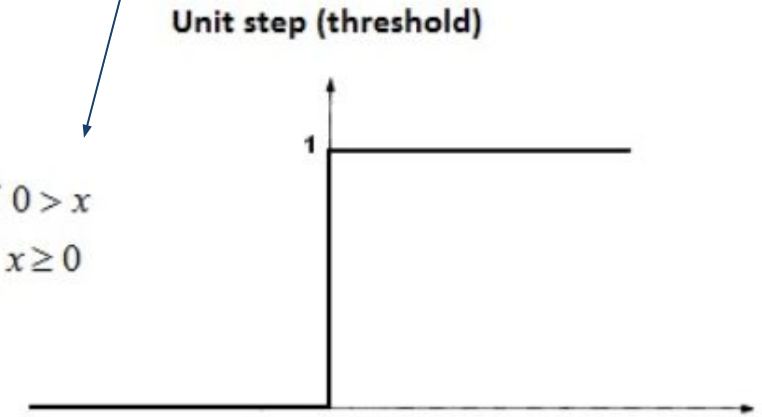
Función de la neurona artificial

$$y = \theta(\sum_j w_{ij}x_j + \mu_i)$$

Donde x_j es la entrada, w_{ij} son los pesos relativos, y θ es la función escalón.



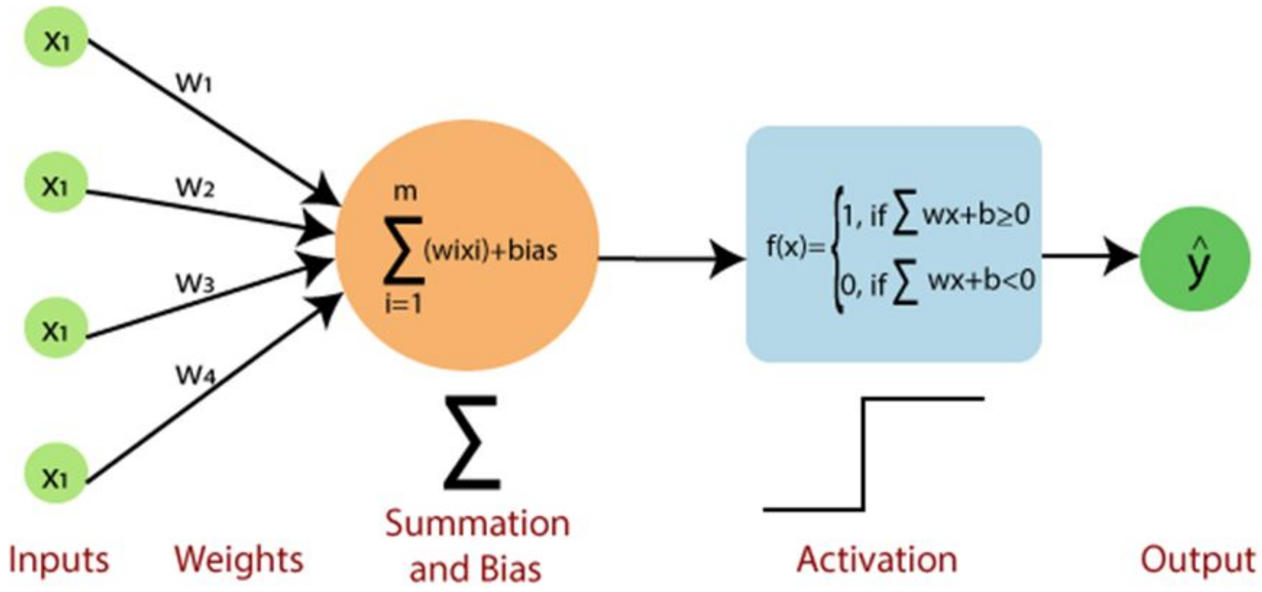
$$f(x) = \begin{cases} 0 \text{ if } 0 > x \\ 1 \text{ if } x \geq 0 \end{cases}$$

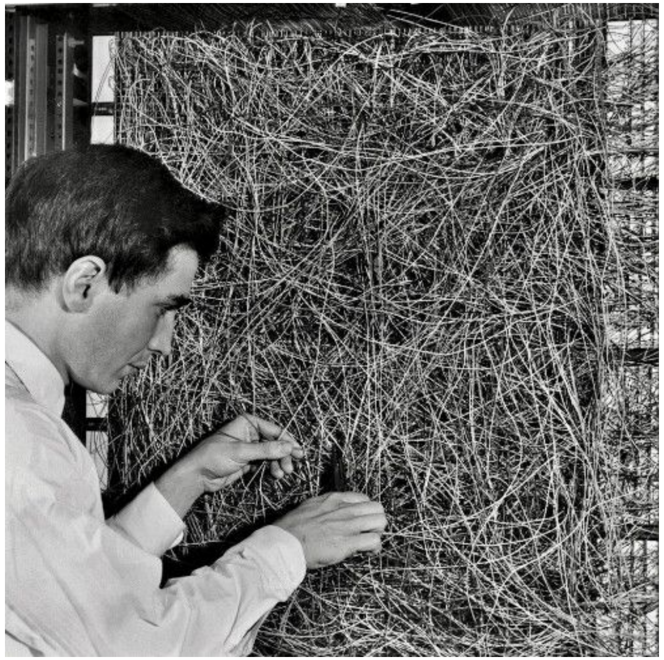


Historia del Perceptrón

Rosenblatt (1957) introdujo el perceptrón, una red neuronal de una sola capa con capacidad de aprendizaje y reconocimiento de patrones.

Limitaciones iniciales: incapacidad para clasificar correctamente si las clases no eran linealmente separables.

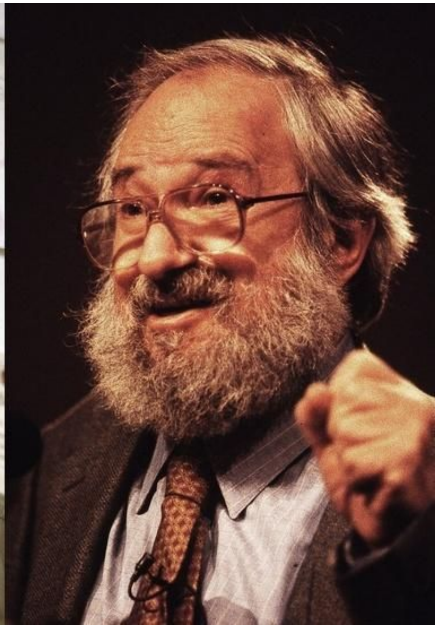




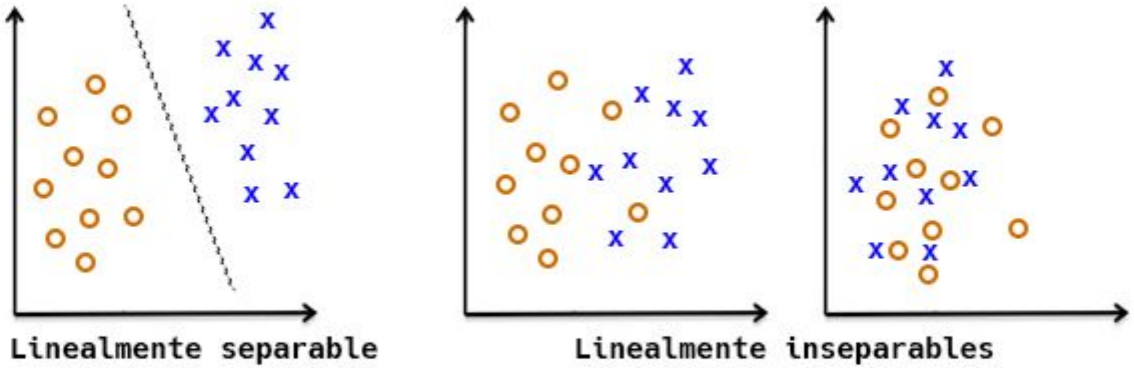
Limitaciones y análisis

Minsky y Papert (1969) mostraron limitaciones en la separabilidad lineal y su convergencia.

Esto desalentó su uso para la mayoría de las tareas de reconocimiento de patrones durante casi dos décadas.



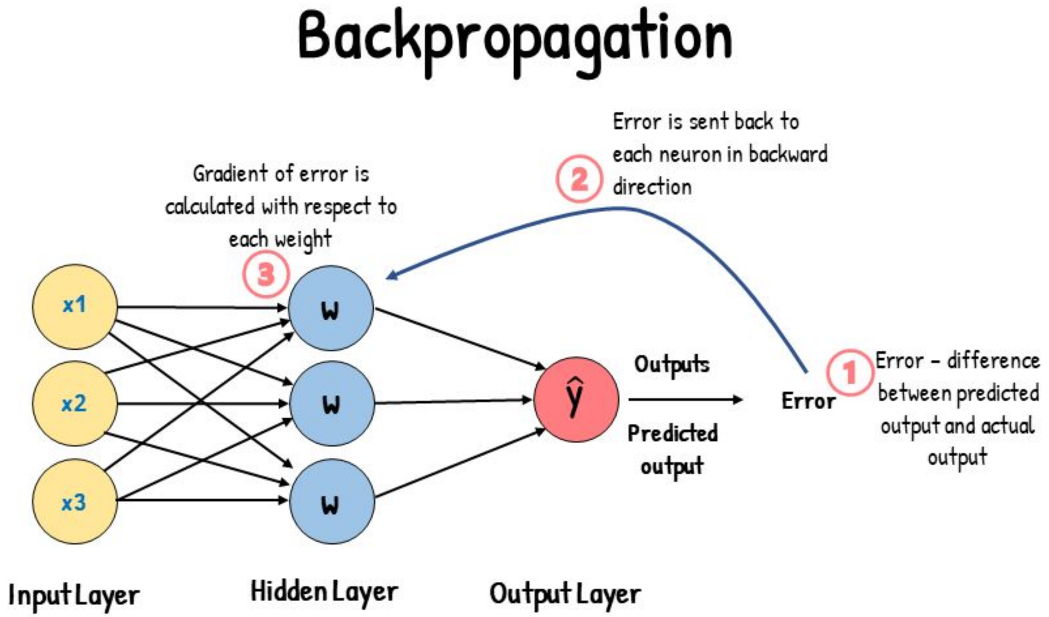
Cuando se dice que las clases no son linealmente separables, significa que no es posible trazar una sola línea recta, plano o hiperplano que pueda separar claramente las clases sin errores.



La solución: Backpropagation

Surgimiento de la regla de aprendizaje para redes neuronales multicapa, superando las limitaciones del perceptrón.

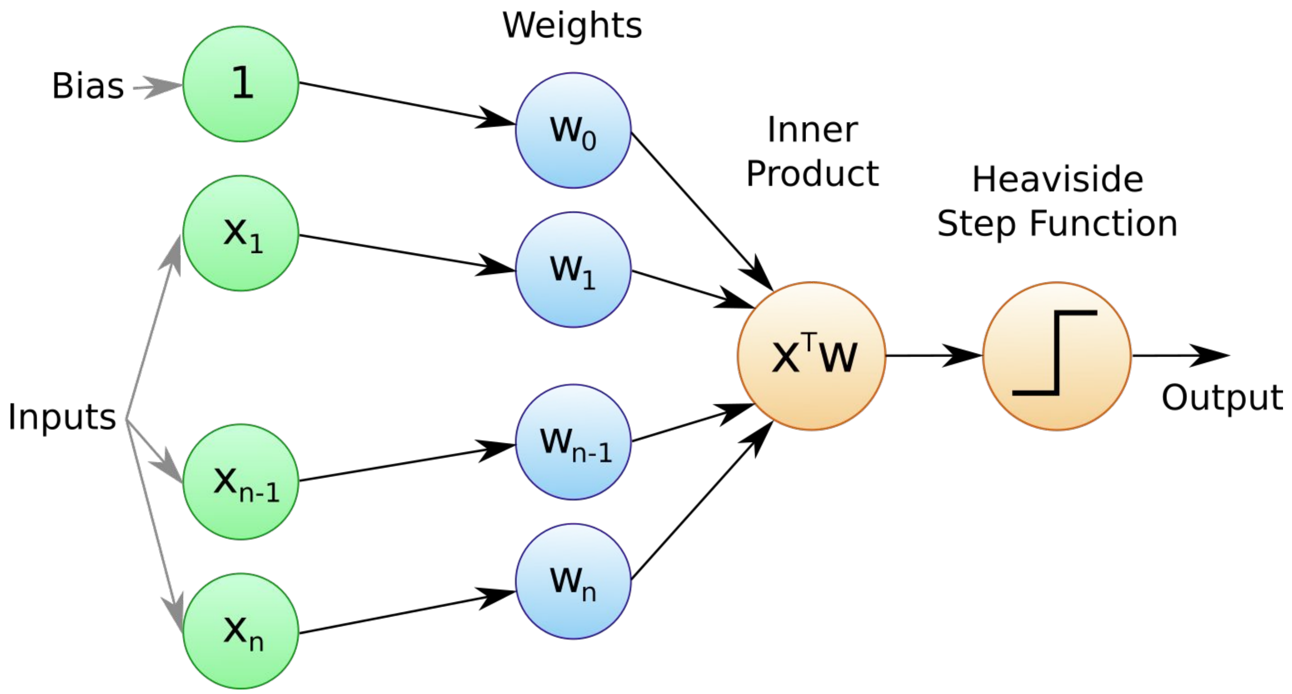
Método de **descenso por gradiente** para minimizar errores y ajustar pesos en función del gradiente de la función de error.



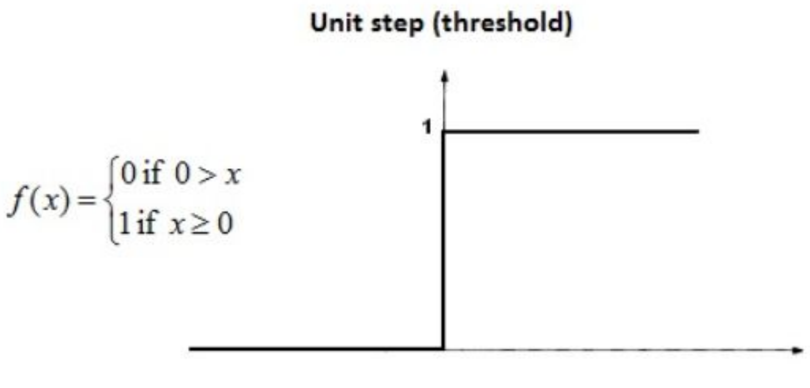
Modelo del Perceptrón

Modelo de aprendizaje supervisado ajustando pesos y polarización para minimizar errores.

Actualización de pesos a medida que se presentan ejemplos de entrenamiento.



$$y_p = f(\sum_i w_i \cdot x_i + b)$$



Épocas en el entrenamiento

Presentación múltiple de datos de entrenamiento a la red para minimizar errores y ajustar pesos.

Primera época

Entrada				Pesos iniciales			Salida					Error	Corrección	Pesos finales		
Valores de sensor			Salida deseada				Sensor			Suma	Red					
x_0	x_1	x_2	z	w_0	w_1	w_2	c_0	c_1	c_2	s	n	e	d	w_0	w_1	w_2
							$x_0 * w_0$	$x_1 * w_1$	$x_2 * w_2$	$c_0 + c_1 + c_2$	if $s > t$ then 1, else 0	$z - n$	$r * e$	$\Delta(x_0 * d)$	$\Delta(x_1 * d)$	$\Delta(x_2 * d)$
1	0	0	1	0	0	0	0	0	0	0	0	1	+0.1	0.1	0	0
1	0	1	1	0.1	0	0	0.1	0	0	0.1	0	1	+0.1	0.2	0	0.1
1	1	0	1	0.2	0	0.1	0.2	0	0	0.2	0	1	+0.1	0.3	0.1	0.1
1	1	1	0	0.3	0.1	0.1	0.3	0.1	0.1	0.5	0	0	0	0.3	0.1	0.1
1	0	0	1	0.3	0.1	0.1	0.3	0	0	0.3	0	1	+0.1	0.4	0.1	0.1

Con cada época verás como el error disminuye poco a poco, o llega un momento es que es cero.

Regla de aprendizaje del perceptrón

Error en la salida y su expresión matemática.
La regla de aprendizaje original de Rosenblatt y otras formas de ajustar pesos.

Enfoques para minimizar el error del perceptrón, incluyendo el **descenso por gradiente**.

Uso actual del descenso por gradiente en las herramientas de Deep Learning.

$$Error_{Global} = \frac{1}{2P}\sum_{k=1}^{P}\sum_{i=1}^{N}(y_i^{(k)} - d_i^{(k)})^2$$

$$\Delta w_{ij} = k\frac{\partial Error_{Global}}{\partial w_{ij}}$$

Actividad a entregar

Deberás resolver el funcionamiento paso a paso de un perceptrón que aprenderá la función NAND.

Entrada				Pesos iniciales			Salida					Error	Corrección	Pesos finales		
Valores de sensor			Salida deseada				Sensor			Suma	Red					
x_0	x_1	x_2	z	w_0	w_1	w_2	c_0	c_1	c_2	s	n	e	d	w_0	w_1	w_2
							$x_0 * w_0$	$x_1 * w_1$	$x_2 * w_2$	$c_0 + c_1 + c_2$	if $s > t$ then 1, else 0	$z - n$	$r * e$	$\Delta(x_0 * d)$	$\Delta(x_1 * d)$	$\Delta(x_2 * d)$
1	0	0	1	0	0	0	0	0	0	0	0	1	+0.1	0.1	0	0
1	0	1	1	0.1	0	0	0.1	0	0	0.1	0	1	+0.1	0.2	0	0.1
1	1	0	1	0.2	0	0.1	0.2	0	0	0.2	0	1	+0.1	0.3	0.1	0.1
1	1	1	0	0.3	0.1	0.1	0.3	0.1	0.1	0.5	0	0	0	0.3	0.1	0.1
1	0	0	1	0.3	0.1	0.1	0.3	0	0	0.3	0	1	+0.1	0.4	0.1	0.1

A	B	Out
0	0	1
0	1	1
1	0	1
1	1	0

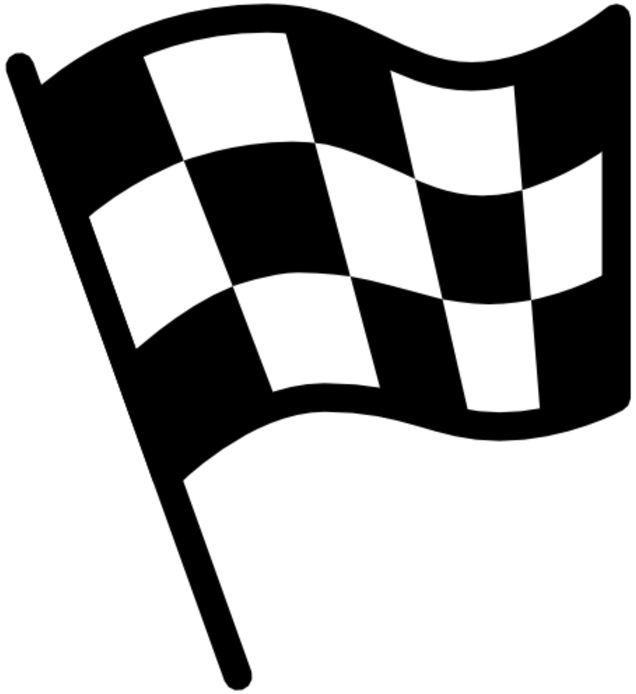
¿Cuántas épocas has necesitado?

Profesor: David Campoy Miñarro

¿Qué has aprendido?

- Aprendizaje supervisado: Perceptrón
- Perceptrón
- Historia de la neurona virtual
- Funciones de activación
- Backpropagation
- Descenso por gradiente

y comprender cómo funciona una red neuronal por dentro.



“Visualizo una época en la que (los humanos) seremos a los robots lo que los perros son para nosotros”

Claude Shannon, matemático, ingeniero eléctrico y criptógrafo