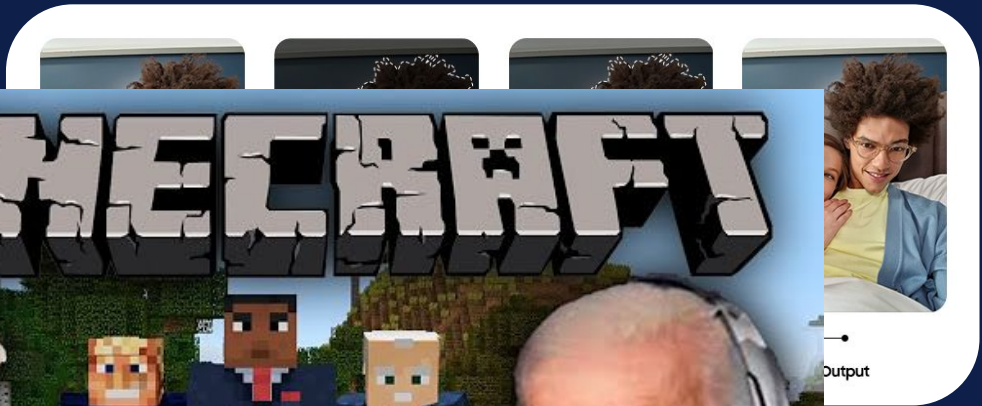


ASTRONOMÍA Y CIENCIA DE DATOS: DE LAS ESTRELLAS A LOS NÚMEROS

Clase 7: Introducción al Aprendizaje Supervisado I:
Redes neuronales simples





ML en astronomía!

Letter | Published: 15 November 1990

Adaptive optics for array telescopes using neural-network techniques

[J. R. P. Angel](#), [P. Wizinowich](#), [M. Lloyd-Hart](#) & [D. Sandler](#)

[Nature](#) **348**, 221–224 (1990) | [Cite this article](#)

942 Accesses | **99** Citations | [Metrics](#)

ML en astronomía!

Morphological classification of galaxies by Artificial Neural Networks

M. C. Storrie-Lombardi,¹ O. Lahav,¹ L. Sodré, Jr^{2,3} and L. J. Storrie-Lombardi¹

¹*Institute of Astronomy, Madingley Road, Cambridge CB3 0HA*

²*Royal Greenwich Observatory, Madingley Road, Cambridge CB3 0EZ*

³*Instituto Astronômico e Geofísico da Universidade de São Paulo, CP9638, 01065, São Paulo, Brazil*

Accepted 1992 September 2. Received 1992 August 24

ML en astronomía!

Así era la red!
Tenía 18 neuronas.
Un total de 252
parámetros
ajustables...

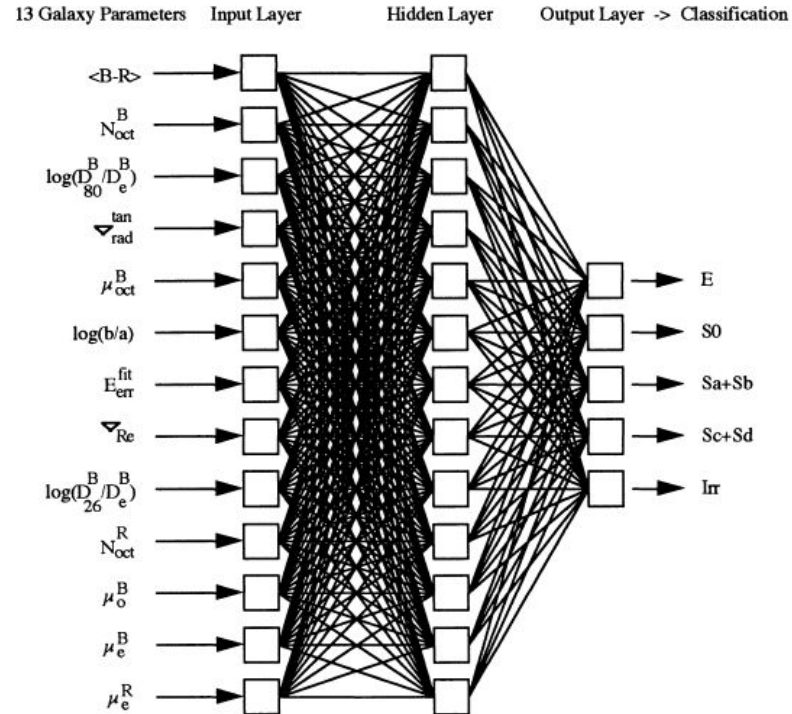


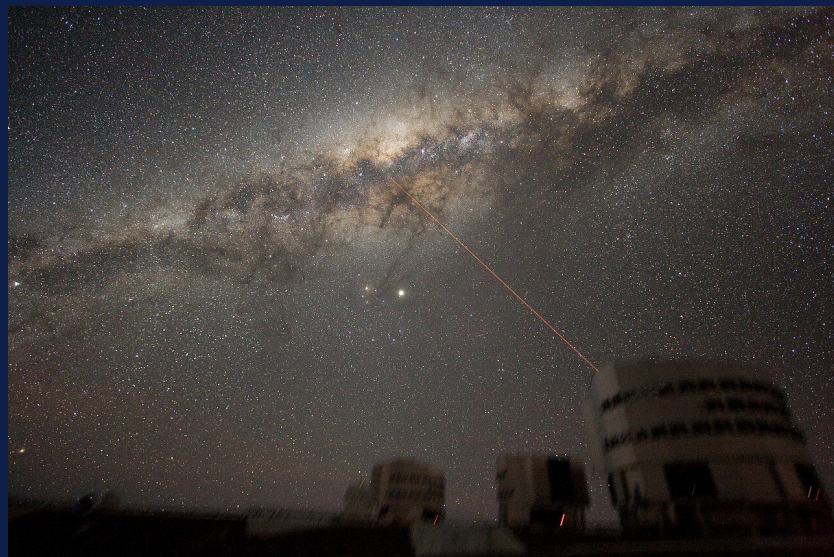
Figure 1. The ANN configuration (13; 13, 5) used in our study, with an input (galaxy parameters) layer of 13 nodes, a hidden layer of 13 nodes, and an output (classification) layer of 5 nodes. All nodes in a given layer are fully connected to all nodes in the next layer. The input parameters are explained in Table 1.

Relativo a chat gpt-4...



Como 80 billones de neuronas,
con 1.76×10^{12} parámetros

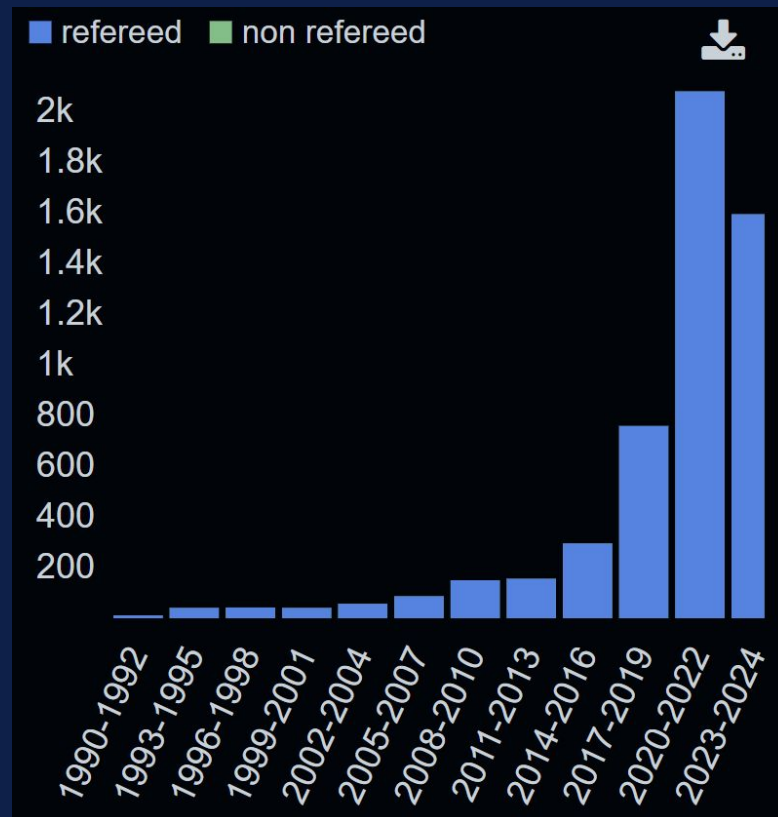
Más parámetros que estrellas
en la Vía Láctea!



ML en astronomía

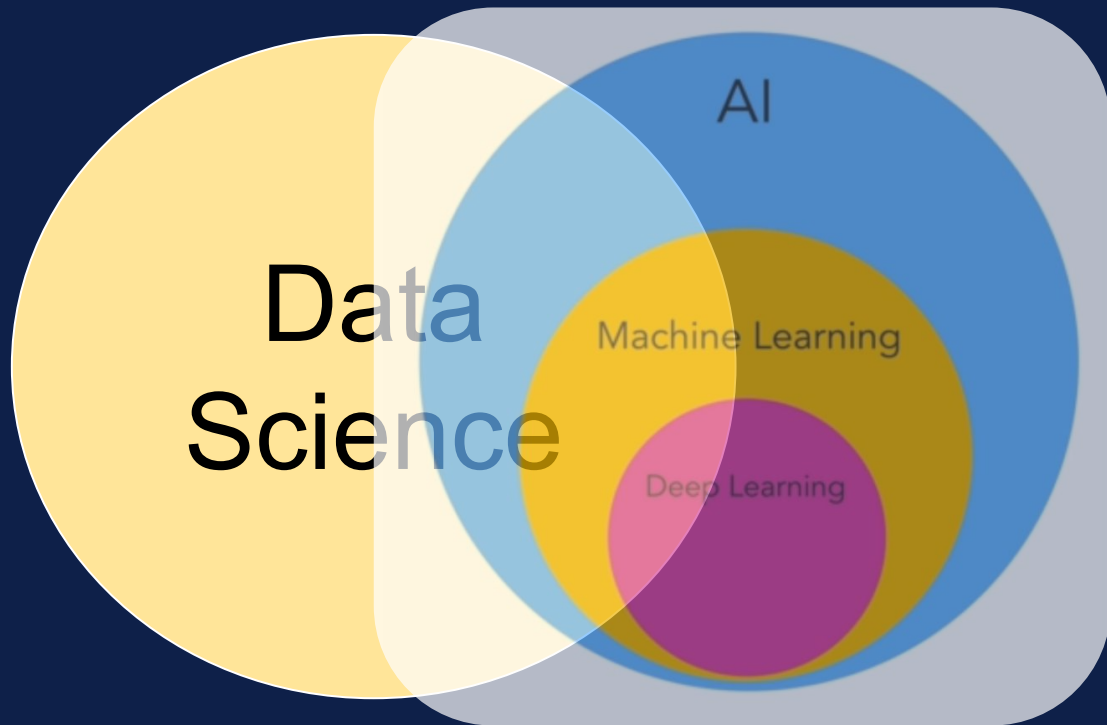
Número de artículos científicos que fueron publicados, utilizando alguna técnica de machine learning.

Algo así como el 10% de todos los artículos que se producen hoy!



¿Qué es el machine learning?

- IA: imitar funciones cognitivas de los seres humanos.
- Machine learning: hacer que una máquina aprenda en base a la “experiencia”.
- Data science: extraer información sobre los datos.

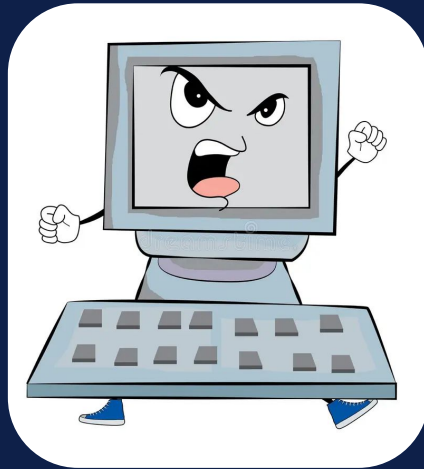


Tipos de aprendizaje

Supervisado



Se busca resolver un problema en específico.



No supervisado

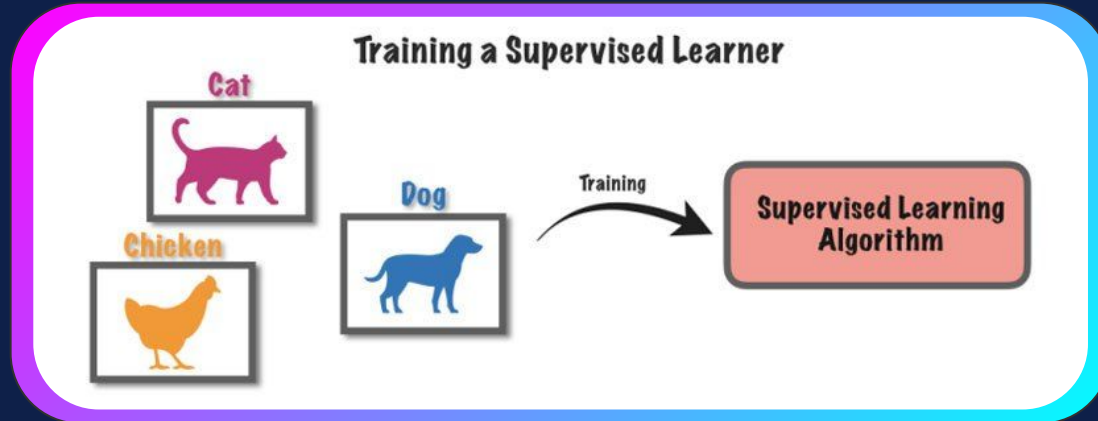


Se busca explorar un dataset.

Tipos de aprendizaje

Supervisado

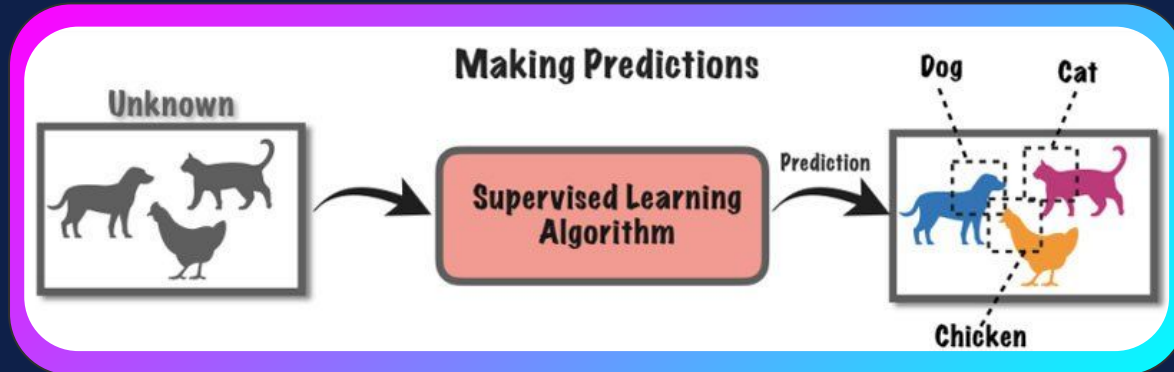
- Se conoce de antemano el resultado esperado en los datos de entrenamiento.
- La máquina “aprende” un patrón en función de los dichos resultados esperados.



Tipos de aprendizaje

No supervisado

- La máquina solo ve los datos, no tiene un “resultado esperado”.
- Busca correlaciones, grupos o categorías para los datos.
- Trata problemas de “Dimensionality Curse”.



Aprendizaje supervisado

Datos

$$\{ inputs = \vec{x} \quad resultado = \hat{y} \}$$

Clasificación

Regresión

\hat{y}

Representa una clase.
Ej.: 0 = perro, 1 = gato

\hat{y}

Es un número real!
Ej.: redshift de una
galaxia.

Aprendizaje supervisado

Clasificación múltiple

\hat{y}

Cada entero es una clase distinta.
Ej.: 0 = perro, 1 = gato, león = 2, lagarto = 3,...

\hat{y}

Es un vector de 0s y 1s, cada vector representa una clase!

“One-hot encoding”

Human-Readable

Pet
Cat
Dog
Turtle
Fish
Cat

Machine-Readable

Cat	Dog	Turtle	Fish
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1
1	0	0	0

Redes neuronales



De vuelta al colegio...

¿Qué es una neurona?

qué es una neurona (un par de oraciones)



Una neurona es una célula nerviosa especializada en transmitir información a través de impulsos eléctricos y químicos. Forma parte del sistema nervioso y está involucrada en funciones como la percepción, el pensamiento y el control del movimiento.



No olviden ser amables..

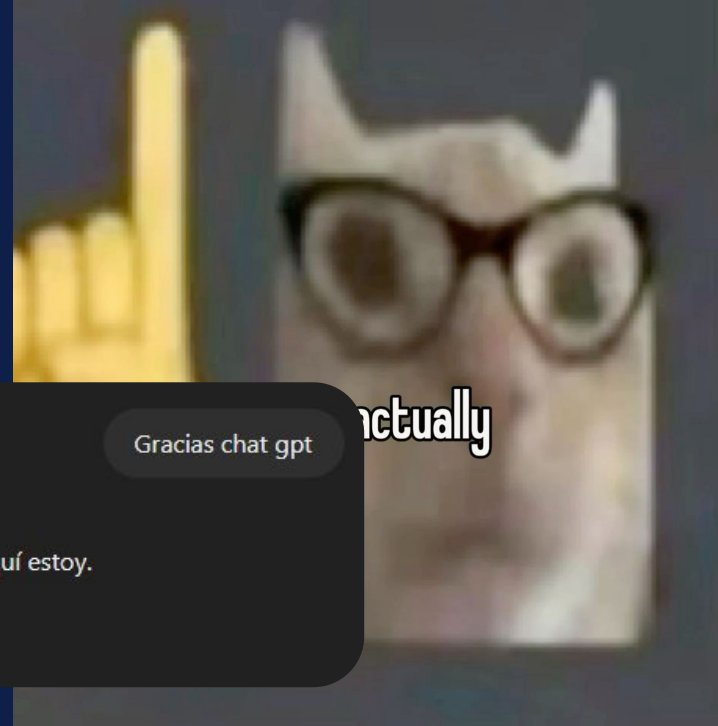


¡De nada! Si tienes más preguntas o necesitas ayuda con algo más, aquí estoy.



Gracias chat gpt

actually



De vuelta al colegio...

NEURON

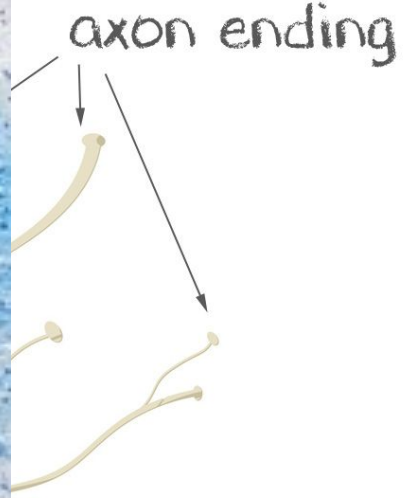
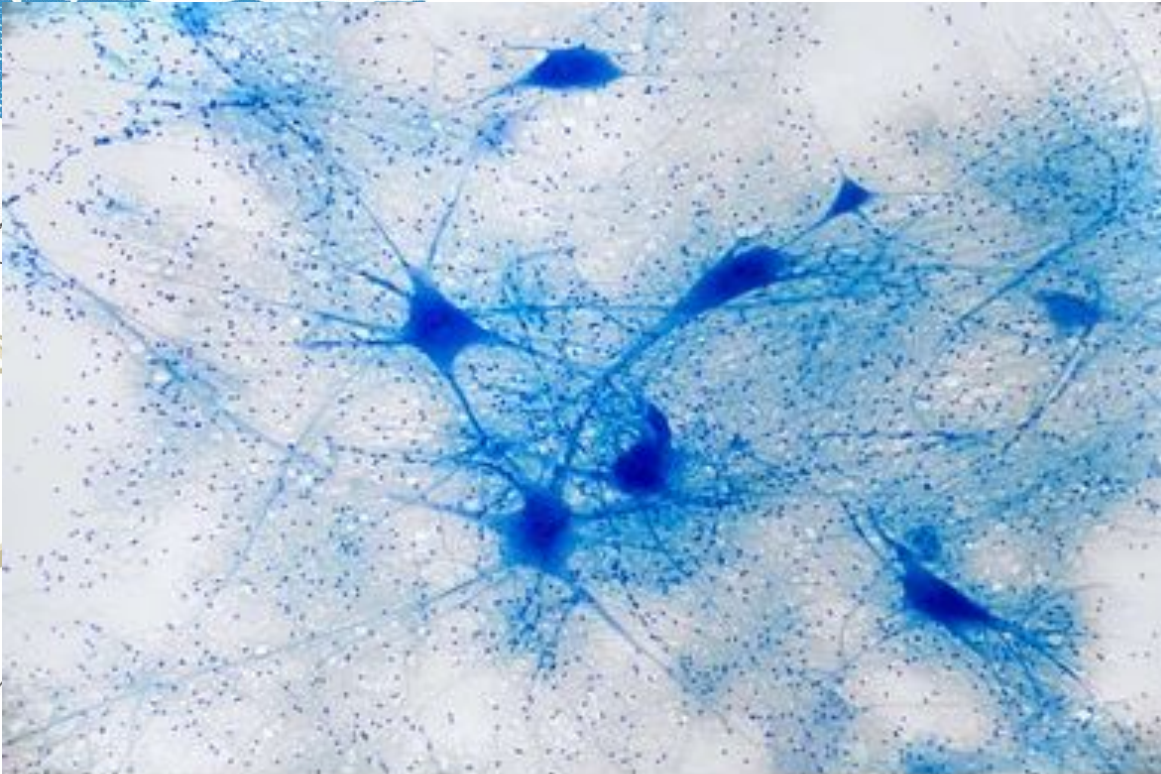
dendrite

nucleus

cell body

myelin sheath

axon ending



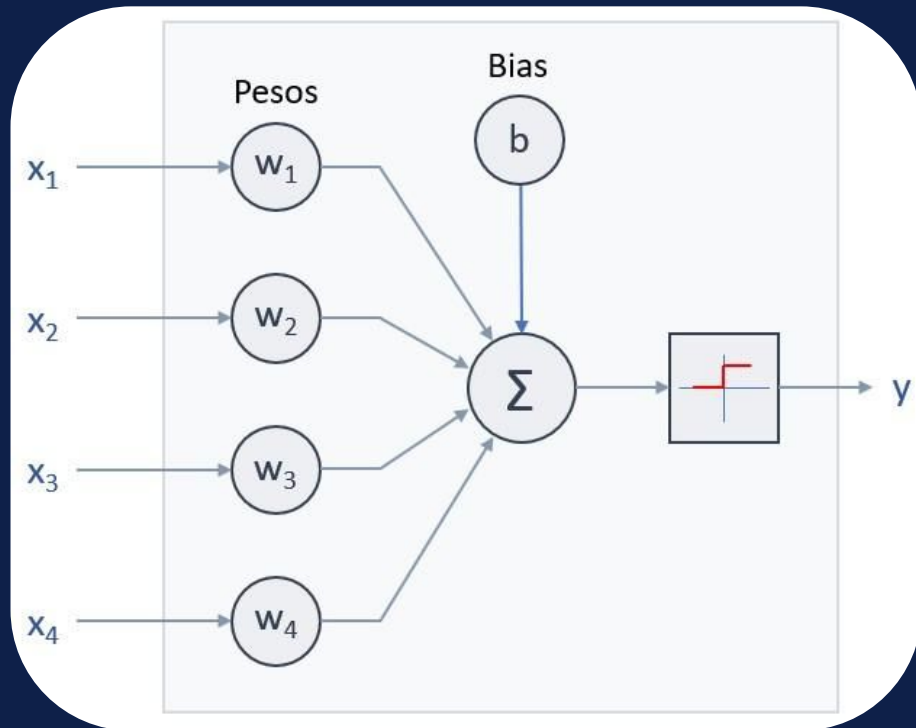
Ahora sí, ¿qué es una neurona?

La definiremos en función de lo que hace matemáticamente.

Tiene 2 partes fundamentales:

- Un combinador lineal.
- Una función no-lineal de activación.

$$y = f(\vec{w} \cdot \vec{x} + b)$$



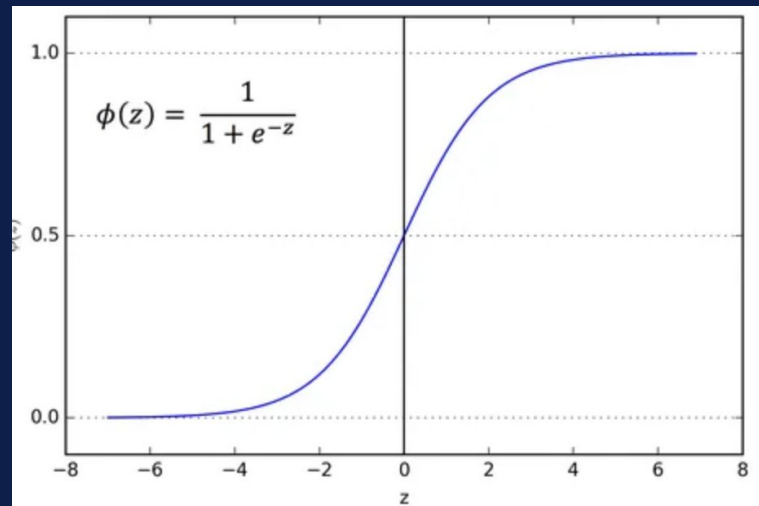
Ahora sí, ¿qué es una neurona?

$$y = f(\vec{w} \cdot \vec{x} + b)$$

¿Qué es f ?

Función logística

$$f(x) = \frac{1}{1 + e^{-x}}$$



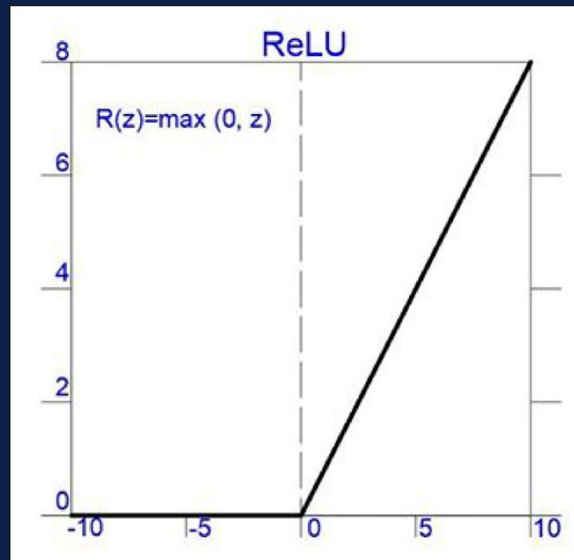
Ahora sí, ¿qué es una neurona?

$$y = f(\vec{w} \cdot \vec{x} + b)$$

¿Qué es f ?

Función RELU

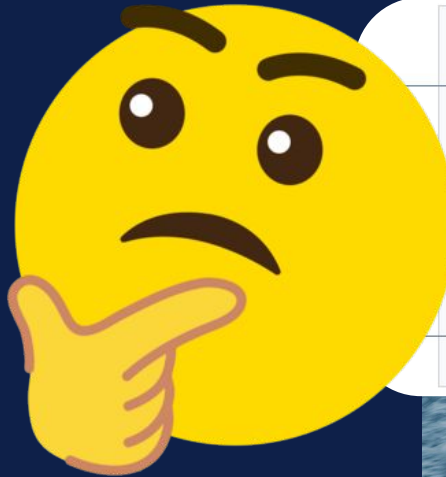
$$f(x) = \max(0, x)$$



Cómo entrenar a tu neurona

- Si una neurona recibe N entradas, entonces tiene $N+1$ pesos.
- Es un problema de ajuste de parámetros no lineal!

¿Qué falta?



Funciones de costo

Suelen depender de del problema a resolver.

Ejemplos típicos

1 Error cuadrático (regresión!) $E(y) = (y - \hat{y})^2$

2 Entropía cruzada (clasificación binaria)

$$E(y) = \hat{y} \ln(y) + (1 - \hat{y}) \ln(1 - y)$$

¿Cómo actualizar los pesos?

Descenso gradiente!

Supongamos que tenemos N datos de entrenamiento.

$$E(y) = (y - \hat{y})^2 \quad \longrightarrow \quad E_{tot} = \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

$$w_{j,1} = w_{j,0} - \mu \frac{\partial E_{tot}}{\partial w_j}(\vec{w}_0)$$

¿Cómo actualizar los pesos?

Descenso gradiente!

Repetimos m
iteraciones...

$$w_{j,1} = w_{j,0} - \mu \frac{\partial E_{tot}}{\partial w_j}(\vec{w}_0)$$

$$w_{j,2} = w_{j,1} - \mu \frac{\partial E_{tot}}{\partial w_j}(\vec{w}_1)$$

Más genérico:

$$w_{j,m} = w_{j,m-1} - \mu \frac{\partial E_{tot}}{\partial w_j}(\vec{w}_{m-1})$$

¿Cómo actualizar los pesos?

PROBLEMA!

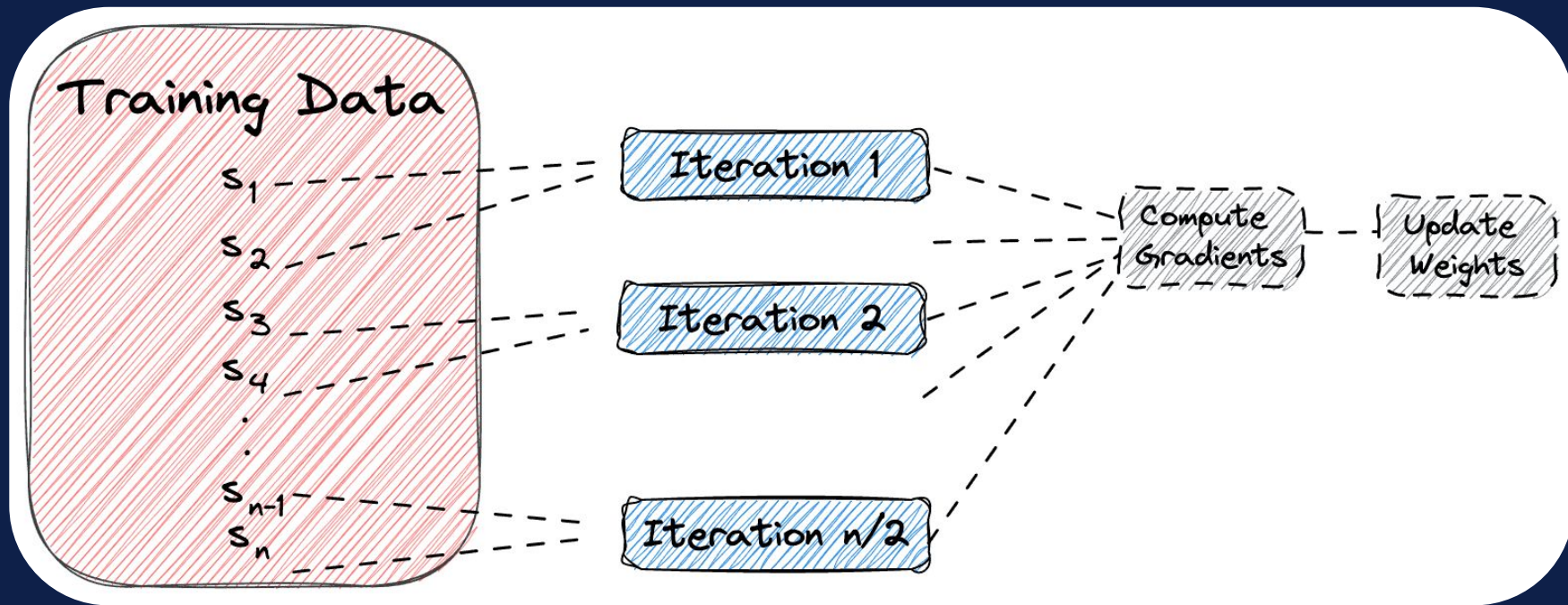
Evaluar el error total
puede ser muy costoso
computacionalmente!

$$E_{tot} = \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

Solución!

Evaluar partes pequeñas del
dataset en cada iteración :)

El entrenar de a “pedazos”



Cada sub-sample se le conoce como “mini-batch”.

El entrenar de a “pedazos”

Cada sub-sample se le conoce como “mini-batch”.

- Para cada iteración se obtiene un gradiente y por ende nuevos pesos.
- Una vez se pasa por todas las iteraciones, se actualizan los pesos tomando el promedio de las variaciones.

Iteración

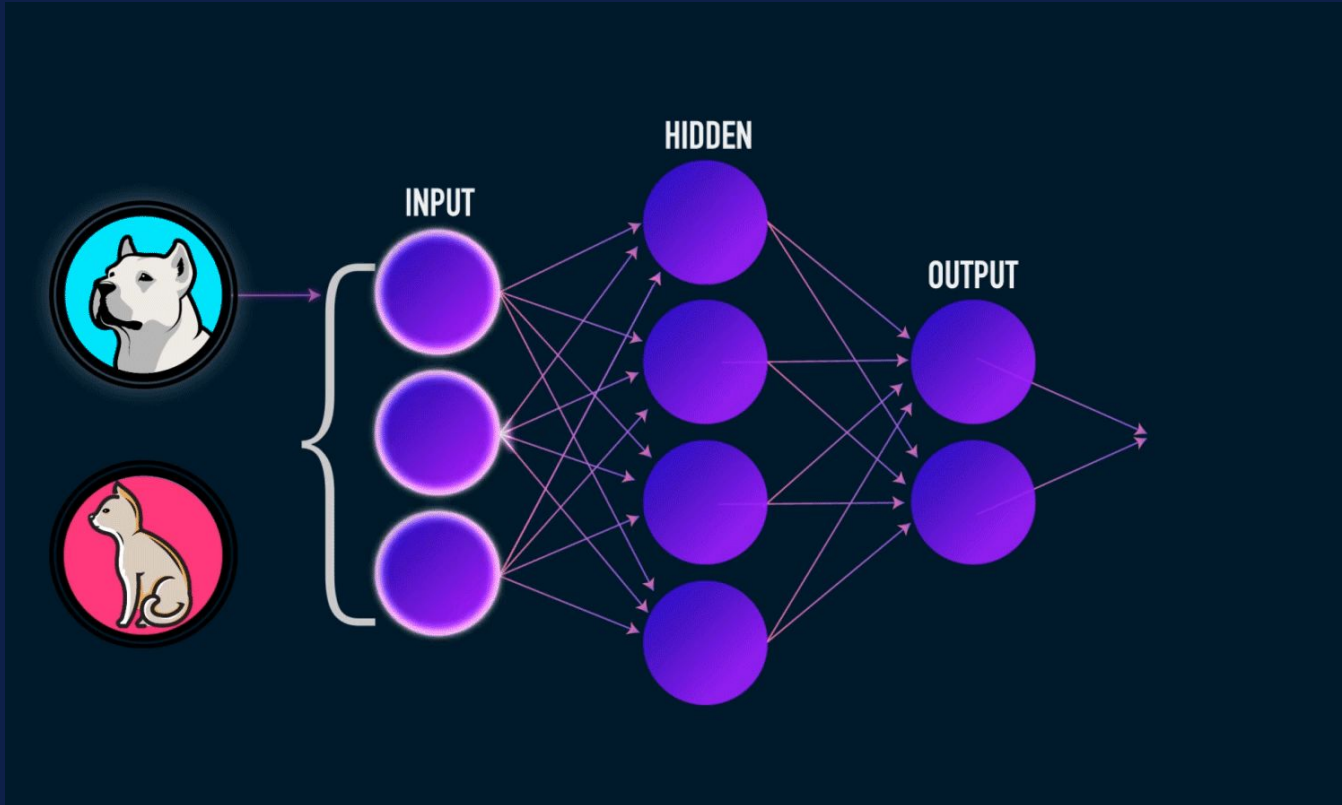
vs

Época

Cuando pasamos
por un
mini-batch.

Pasamos por todos los
batches y actualizamos
los pesos.

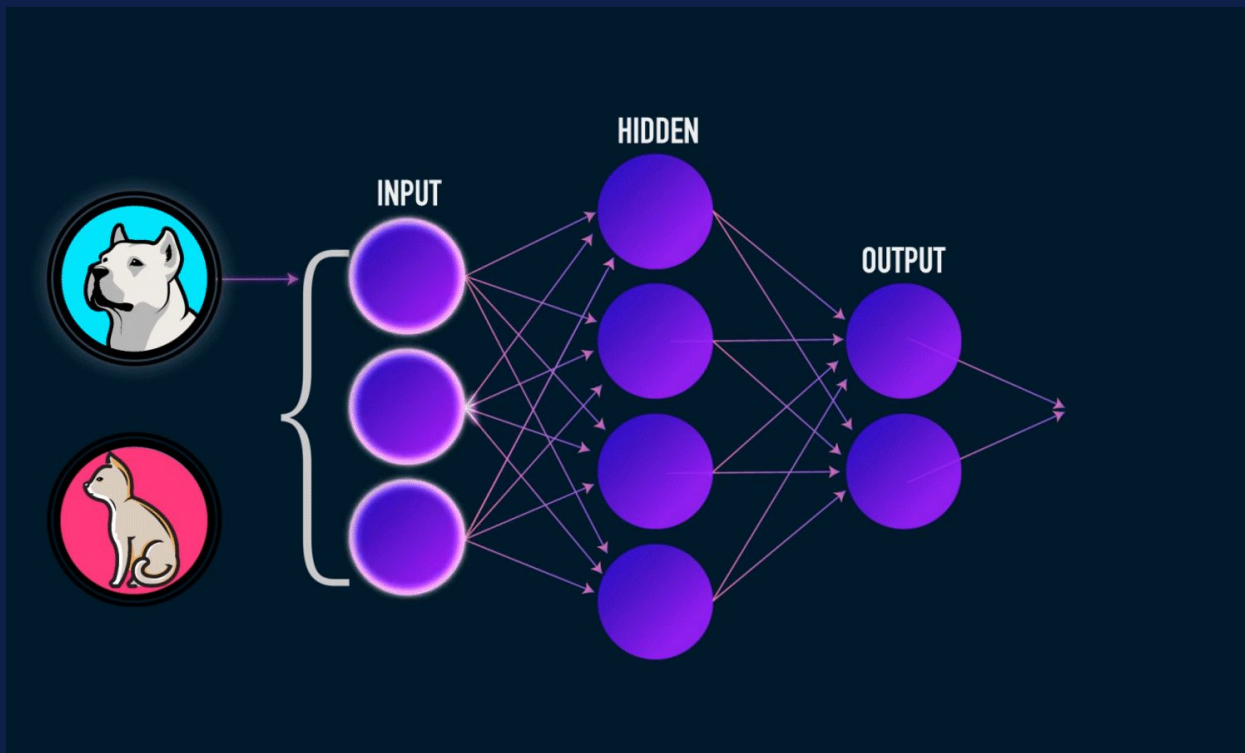
Ahora creemos una red!



Ahora creemos una red!

¿Cuántos
parámetros tiene
esta red?

R: 26 parámetros
entrenables :)



Un poco de historia...

1943

Nace el
concepto

1995

Caen las RNs,
víctimas de los
SVM!

1970

IA winter ;(

2009

Deep learning!

1986

Backpropagation :)!

Evolucionar los pesos
intermedios es difícil...

...a menos que usemos
CVV :)

Ya tengo mi red,
¿puedo entrenarla?



A background image of a modern gym with rows of dumbbells in the foreground and exercise machines in the background. A semi-transparent blue and purple gradient box is overlaid in the center, containing white text.

Las reglas de oro del
entrenamiento:
separación de los datos

Separación de los datos

Entrenamiento

Aquí la red aprende.
Con estos datos se actualizan los pesos.

Validación

Se evalúa el desempeño de la red en cada época.
Sirve para escoger modelo óptimo.

Testeo

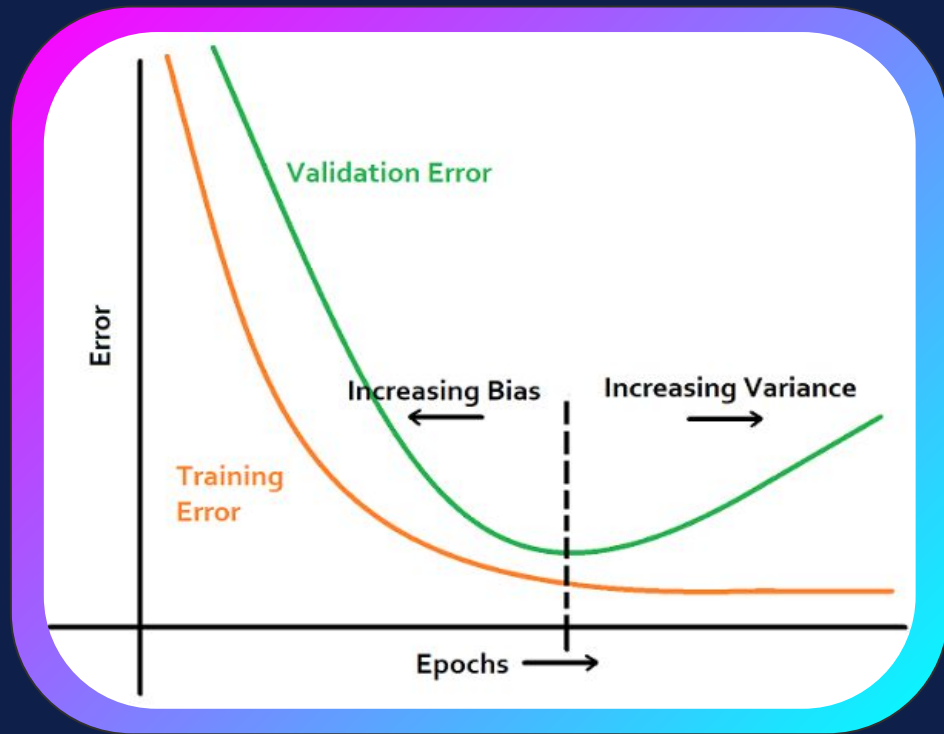
Ejemplos nunca vistos por la red!
Evalúa el desempeño final del modelo.

A photograph of a modern gym interior. In the foreground, there is a rack of dumbbells with black handles and silver weights. The background shows other gym equipment, including treadmills and exercise machines, and a few people working out. The lighting is bright and even.

Las reglas de oro del
entrenamiento: evitar
el overfitting

Evitar el overfitting

Early
stopping! :)

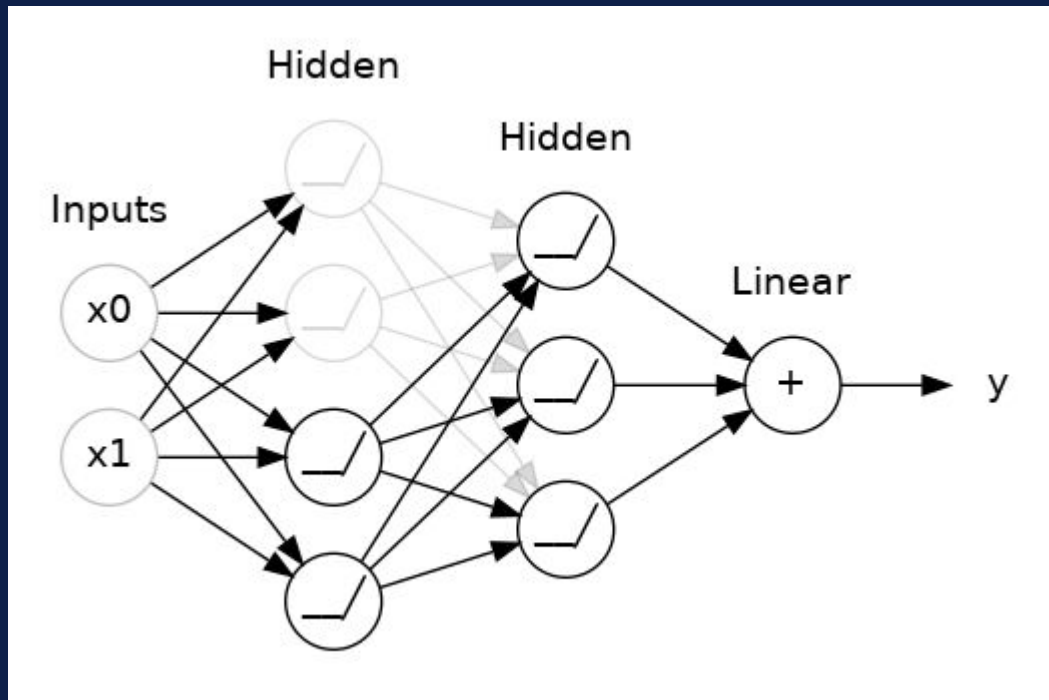


Evitar el overfitting

Dropout!

En cada iteración se apagan algunas neuronas.

Evita que las neuronas “arreglen” errores inducidos por sus “pares”.



Entonces, ¿ahora si
puedo entrenarla?

Sonic
dice...

