

King of Dumbs

el rey de los tontos

Nivel: ♣ Novato

¿Regresión logística o red neuronal? Es la misma neurona

Una regresión logística es una red neuronal de una sola neurona. Qué añade el deep learning, cuándo compensa y cuándo basta lo clásico, en tres niveles.

Español Publicado 4 de octubre de 2026



Escanea para abrir la lección interactiva

<https://kingofdumbs.com/es/lecciones/regresion-logistica-o-red-neuronal>

Qué vas a aprender

- Regresión logística
- Redes neuronales
- Función sigmoide
- Gradiente que se desvanece
- Elegir entre machine learning clásico y deep learning

Símbolos que aparecen

E_a subíndice: la etiqueta de abajo (♠) · Σ sumatorio: súmalos todos (♠) · x_i índice i: el elemento número i (♠) · Π productorio: multiplícalos todos (♠) · $\approx$ aproximadamente igual (♠) · $\sim$ del orden de / se reparte como (♠) · $\leq \geq$ menor o igual / mayor o igual (♠) · $|x|$ valor absoluto: sin signo (♠) · 10^{-3} exponente negativo: dividir (♠) · $h^{(1)}$ superíndice entre paréntesis: no es potencia (♠) · e el número e ($\approx 2,718$) (♠) · $\log$ logaritmo: ¿cuántas veces multiplico? (♦♠) · σ sigma: depende de la lección (♠) · λ lambda: un número que estira (♠) · δ delta minúscula: un error pequeño (♠) · $\|x\|$ norma: la longitud de la flecha (♠) · x^T traspuesta: tumbar la columna (♠) · $\in$ pertenece a (♠) · $\mathbb{R} \mathbb{Z}$ conjuntos de números (♠) · dE/dt derivada: el ritmo de cambio (♠) · ∂ derivada parcial: cambiar una sola cosa (♠) · ∇ gradiente: hacia dónde sube más (♠)

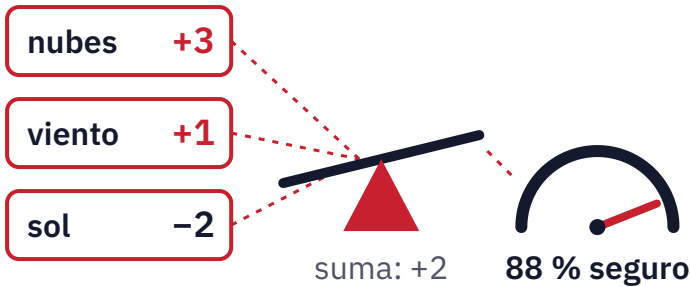
La primera vez que vi una red neuronal dibujada, con sus circulitos y sus flechas, pensé que era otra cosa que la estadística de toda la vida. Luego me tocó programar una regresión logística a mano y me hice la pregunta tonta de esta lección: **¿una regresión logística es una red neuronal?**

La respuesta corta es que sí. Una regresión logística es una red neuronal de **una sola neurona**. No se parecen ni son primas: son el mismo modelo con dos nombres. Si es la misma neurona, la pregunta interesante cambia: ¿qué añaden las redes grandes, y cuándo merece la pena pagarlas?

Elige tu carta: ♣ si empiezas de cero, ♦ si sabes qué es una función o una media, ♠ si quieres la notación. Puedes cambiarla en cualquier momento.

Una neurona, dos nombres

¿Lloverá mañana?



¿Seguro y fallas? Bronca.

Imagina que quieres adivinar si mañana lloverá. Miras pistas: hay nubes, hace viento, por la tarde sale el sol. No todas cuentan igual: las nubes pesan mucho, el viento un poco y el sol resta. Sumas lo que pesa cada pista, y esa suma mueve una aguja que va de “seguro que no” a “seguro que sí”.

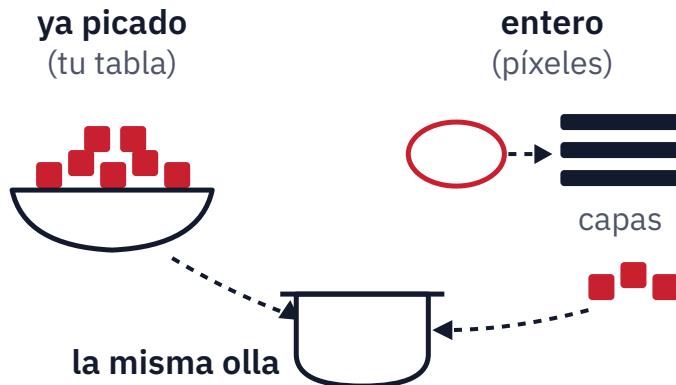
Esa balanza es una **regresión logística**, un método de estadística de toda la vida para responder preguntas de sí o no. Y también es una **neurona**, la pieza con la que se construyen las redes neuronales. Es la misma balanza con dos nombres.

Falta lo más importante: aprender. Al principio, la balanza no sabe cuánto pesa cada pista. Le enseñas muchos días pasados y, cada vez que falla, le echas una bronca, más gorda cuanto más segura estaba. Para evitar broncas, la balanza va ajustando los pesos poco a poco. Y con esta balanza pasa algo tranquilizador: empieces por donde empieces, siempre acabas en los mismos pesos.

El Rey de los Tontos dice:

Yo me pasé una semana creyendo que tenía que elegir entre estadística y deep learning. Resulta que llevaba años haciendo deep learning de una neurona sin saberlo.

Quién fabrica las features

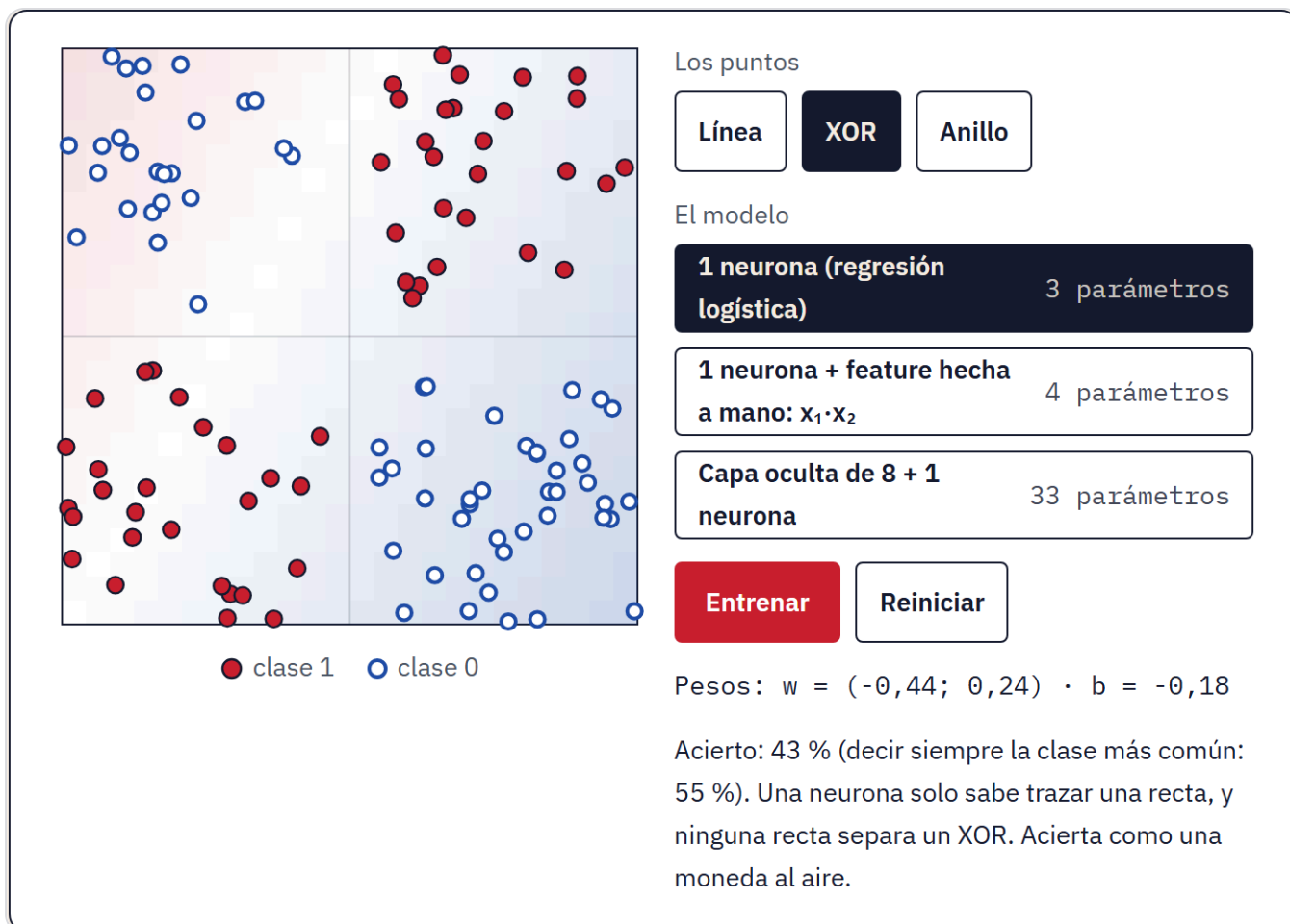


Si la balanza del final es la misma, ¿qué tiene de especial una red neuronal grande? Lo que pasa **antes** de la balanza.

Piensa en una olla. Si cocinas con ingredientes ya picados, los echas y listo. Así son los datos en forma de tabla: edad, importe, número de visitas. Cada columna ya es una pista con sentido, y la balanza las pesa directamente.

Una foto es una patata entera. Un punto de color suelto no te dice si hay un gato. Alguien tiene que picar la foto en pistas útiles, como “aquí hay unas orejas” o “esto parece un bigote”. Las **capas** de una red neuronal hacen justo eso: aprenden solas a picar. A esas pistas fabricadas los entendidos las llaman *features*. Al final, todo cae en la misma olla: la misma balanza de antes.

Pruébalo. El XOR y el anillo son los dos ejemplos clásicos de puntos que una recta no puede separar. Elige un conjunto y un modelo, y entrénalo:

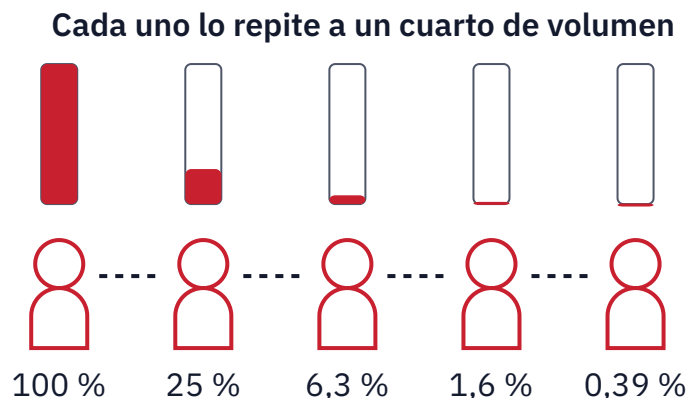


Widget interactivo. Pruébalo en la web: <https://kingofdumbs.com/es/lecciones/regresion-logistica-o-red-neuronal>

Lo que más me gusta de este experimento es la fila del medio. La misma neurona que fracasa con el XOR acierta en cuanto le das hecho a mano el producto de las dos coordenadas. La capa oculta llega al mismo sitio sin que nadie le diga qué buscar, a cambio de 33 números que ajustar en lugar de 4 y de que ya no puedas leer qué ha aprendido.



Por qué la sigmoide salió de las capas ocultas



Si una red es “muchas balanzas en fila”, lo natural sería usar la misma aguja, la de la curva en S, en todas. Así se hizo durante años, y salió mal.

Para aprender, una red recibe un aviso al final, “te has equivocado por tanto”, y ese aviso tiene que viajar hacia atrás, de la última capa a la primera, para que cada una ajuste sus pesos. Es como el teléfono escacharrado: cada capa con esa curva repite el aviso, como mucho, a **un cuarto del volumen**. Tras cinco personas, el mensaje llega a menos de la mitad de un uno por ciento. Con diez, no llega ni un susurro, y las primeras capas casi no aprenden.

La solución fue cambiar la aguja de las capas de en medio por otra que no baja el volumen cuando el aviso es positivo. La curva en S no se jubiló: se fue a trabajar de **portera**. Sigue al final de la red, para decir “88 % de que sí”, y dentro de los grandes modelos de lenguaje, decidiendo cuánta información deja pasar.

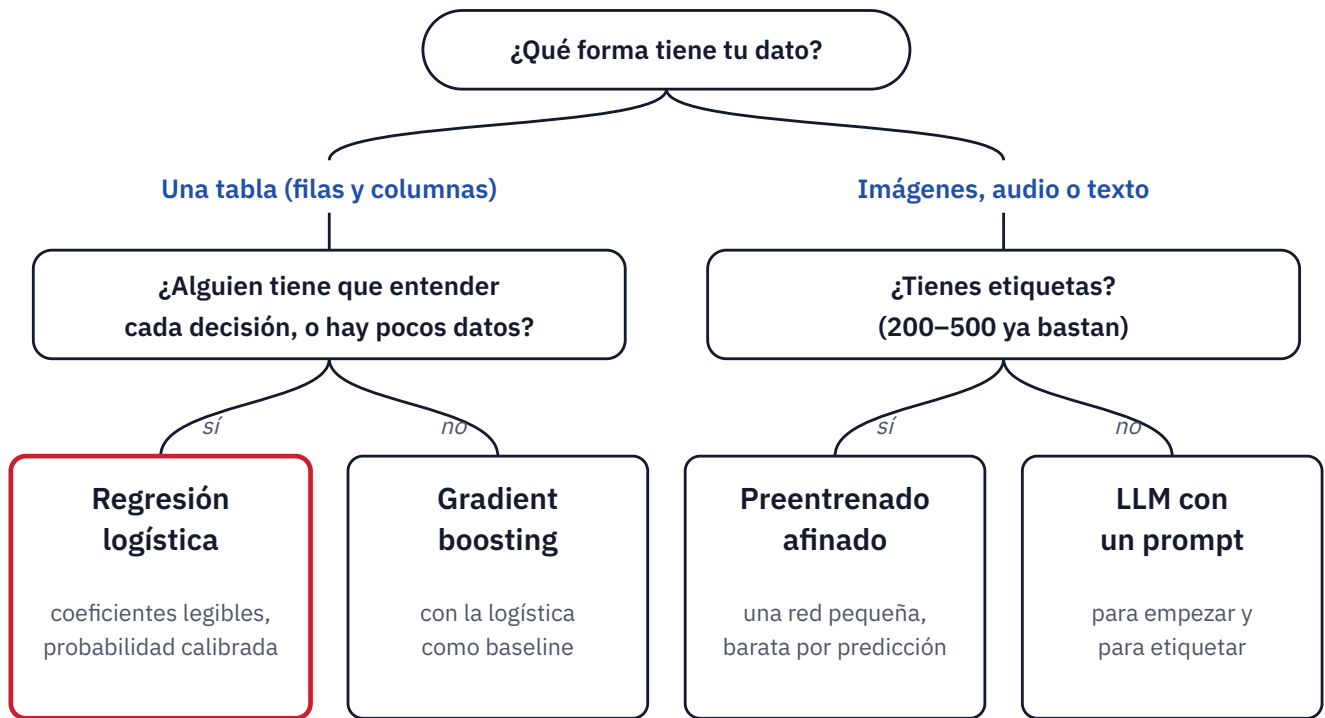
El Rey de los Tontos dice:

La sigmoide es como yo: la echaron de las capas ocultas y acabó de portera en todos los modelos de lenguaje.

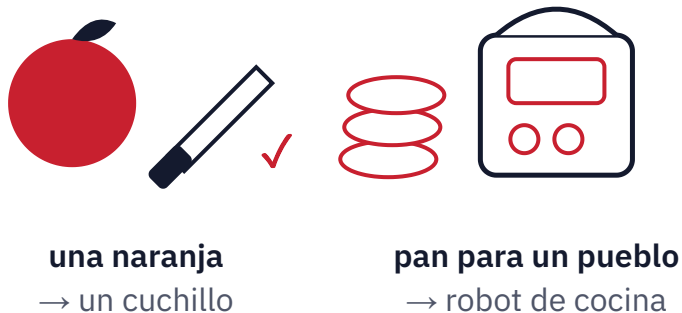


Cuándo usar cada cosa

La decisión casi nunca va de “lo complejo que es el problema”. Va de qué forma tiene tu dato, cuántos ejemplos con la respuesta correcta tienes y quién tiene que entender cada decisión. Este árbol vale para los tres niveles:



Para predecir una etiqueta. Bajar de complejidad es barato; subir, caro: empieza por la izquierda.



Para pelar una naranja no te compras un robot de cocina. El robot es caro, tarda en montarse y, si algo sale mal, no sabes por qué. Compensa cuando tienes que hacer pan para un pueblo entero.

Con los modelos pasa igual. Si tus datos son una tabla con columnas que ya significan algo, la balanza de toda la vida suele bastar, y además puedes explicar cada decisión: “te he dicho que no por esta columna”. Si tus datos son fotos, sonido o texto, hace falta quien pique los ingredientes, y ahí sí compensa una red. Un consejo que sirve siempre: prueba primero la balanza. Tardas minutos y te dice cuánto tendría que ganar el robot para que merezca la pena.



Reto final

1. Una regresión logística y una neurona con sigmoide, entrenadas con el mismo castigo, ¿qué son?

- a. Primas: se parecen, pero dan resultados distintos
- b. El mismo modelo con dos nombres
- c. La neurona siempre acierta más, porque es deep learning

2. ¿Por qué dejó de usarse la sigmoide en las capas ocultas?

- a. Porque es lenta de calcular
- b. Porque el aviso del error se encoge en cada capa y las primeras casi no aprenden
- c. Porque los números entre 0 y 1 no sirven para nada

3. Tienes una tabla de 5.000 filas (edad, importe, visitas) y alguien tiene que entender cada decisión. ¿Por dónde empiezas?

- a. Un modelo de lenguaje con un prompt
- b. Una red profunda de diez capas
- c. Una regresión logística



En tres frases

1. Una regresión logística es una red neuronal de una sola neurona: la misma suma con pesos, la misma curva en S (la sigmoide) y el mismo castigo.
2. Lo que añade el deep learning son las capas de antes, que fabrican las features; la sigmoide salió de esas capas porque apaga el aviso del error, y hoy trabaja en la salida y en las puertas.
3. Si tus datos son una tabla con columnas que ya significan algo, empieza por la logística, que tarda minutos y te dice cuánto tiene que ganar lo complejo; si son píxeles, audio o texto,

alguien tiene que fabricar las features.



Fuentes

- Regresión logística como modelo lineal generalizado y sigmoide como enlace canónico: [Stanford CS229, apuntes](#). Convexidad de la log-loss: Boyd y Vandenberghe, [Convex Optimization](#), §7.1.1.
- Valores por defecto de LogisticRegression (L2, C=1.0): [documentación de scikit-learn](#).
- Gradiente que se desvanece y derivada máxima de la sigmoide: Hochreiter, tesis de 1991 ([PDF](#), en alemán) y artículo de 1998 ([PDF](#)).
- Error de test con sigmoide, tanh y tanh con inicialización normalizada: Glorot y Bengio, *AISTATS* 2010, [Tabla 1](#).
- SwiGLU: Shazeer, [arXiv:2002.05202](#); su uso en [LLaMA](#) y [PaLM](#). Puertas de LSTM: [Hochreiter y Schmidhuber, 1997](#).
- Machine learning frente a regresión logística en predicción clínica: Christodoulou et al., *J Clin Epidemiol* 110, 12 (2019), [doi:10.1016/j.jclinepi.2019.02.004](#). Leído el resumen.
- Árboles frente a redes en datos tabulares: Grinsztajn et al., [arXiv:2207.08815](#).
- Modelos afinados frente a LLM sin ejemplos: Bucher y Martini, [arXiv:2406.08660](#), Tablas 1 a 4 y §5.5.
- Calibración de redes modernas: Guo et al., *ICML* 2017, [arXiv:1706.04599](#), Tabla 1.
- BCEWithLogitsLoss: [documentación de PyTorch](#).

Cómo está hecha esta lección: a partir de mis notas de machine learning, con ayuda de Claude para ordenarlas y redactar los tres niveles. Las cifras se comprobaron en las tablas originales de cada artículo el 4 de octubre de 2026. El laboratorio de la neurona entrena en tu navegador con descenso de gradiente, igual que el código del nivel ♠. Los dibujos son propios.

Soluciones

1. **b)** El mismo modelo con dos nombres. Eso es. Mismas tres piezas, mismos pesos al final.
2. **b)** Porque el aviso del error se encoge en cada capa y las primeras casi no aprenden. Exacto: como mucho 0,25 por capa. Con diez capas no llega ni un susurro.
3. **c)** Una regresión logística. Bien. Las columnas ya significan algo y la decisión se tiene que poder explicar: la logística es el punto de partida.

Lección de King of Dumbs, kingofdumbs.com. Versión para leer sin conexión; la original, con widgets que se tocan, está en: <https://kingofdumbs.com/es/lecciones/regresion-logistica-o-red-neuronal>

Actualizada el 4 de octubre de 2026.