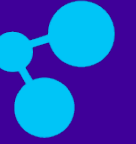




Inteligencia Artificial para profesionales TIC

Introducción a las redes neuronales y el DL



Índice

01. Background

02. Neurona artificial

03. Redes Neuronales Artificiales

04. Deep Learning



1. Background

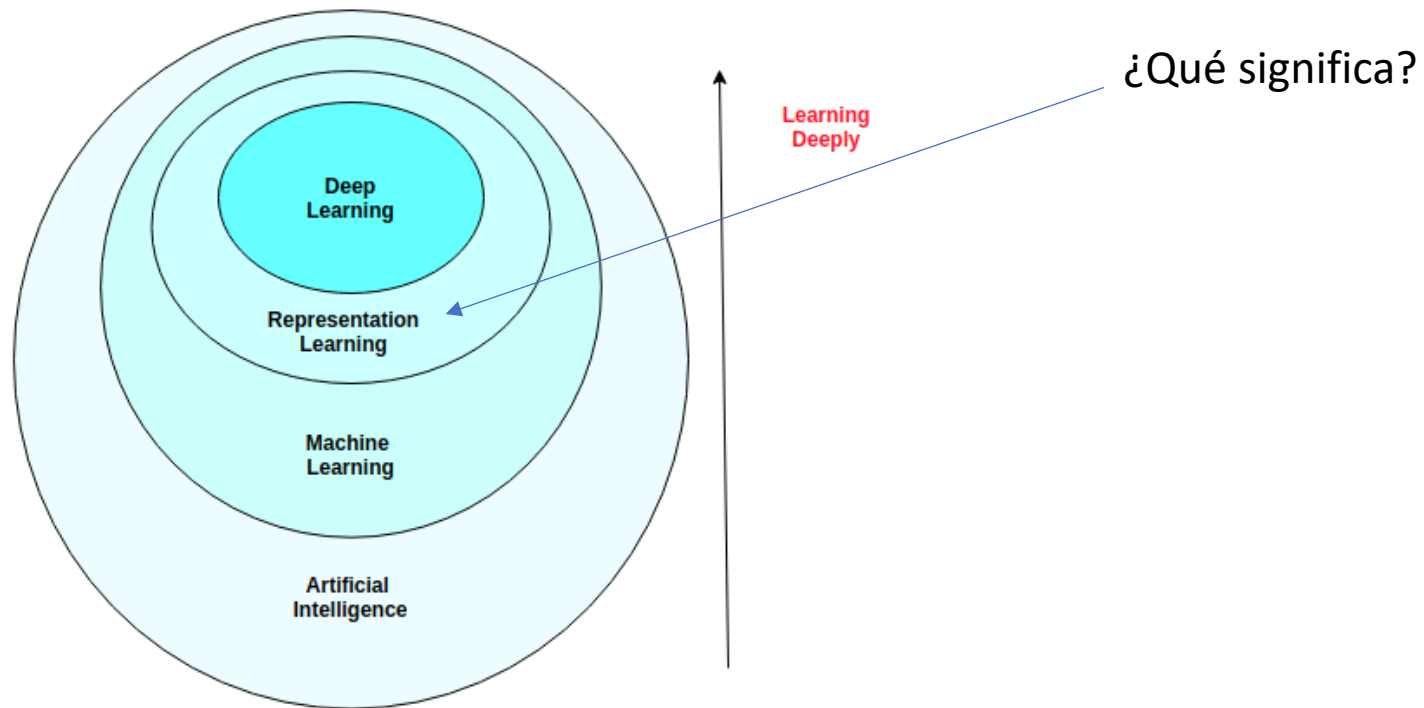
El viaje hacia el aprendizaje profundo

- En esta unidad vamos a presentar las **redes neuronales** y las grandes ventajas que podemos obtener del aprendizaje profundo
- En esta y futuras unidades trataremos los principios teóricos y los elementos prácticos de esta técnica avanzada de ML
- El **aprendizaje profundo** (basado en redes neuronales) es la técnica de ML más puntera del momento
- Esto no significa que sea la mejor opción siempre, especialmente con 'structured-data' donde podemos obtener excelentes resultados con modelos ensemble, incluso mejores resultados frente a pequeños conjuntos de datos (XGBT es una opción común entre los ganadores en algunos concursos en Kaggle)
- Donde DL va a ofrecer los mejores resultados, sin lugar a dudas, es frente a los 'datos no estructurados', como imágenes, vídeos, textos, gráficos, etc.
- También es interesante recordar, desde la introducción al Machine Learning, que las redes neuronales y el aprendizaje profundo dan los mejores resultados cuando tenemos grandes conjuntos de datos



1. Background

La visión holística de la IA, de nuevo





1. Background

Un nuevo elemento: el aprendizaje de la representación

- La representación de datos es un elemento crítico para el éxito de una técnica de ML (**vector de características**, ingeniería de características)
- Hasta ahora, hemos sido **nosotros**, o los expertos en la materia, los que hemos definido los datos o características que utilizaremos en nuestra técnica de ML
- Este proceso siempre es un **reto**, incluso con información estructurada, y casi imposible con información no estructurada
- Gran parte de la actividad investigadora durante años se ha centrado en la "ingeniería de características" frente a muchos problemas diferentes, con incluso muchos problemas en los datos no estructurados, utilizando el procesamiento de imágenes, el procesamiento de señales, la estadística, la teoría de la información y muchos otros campos; siempre a cargo de científicos e investigadores



1. Background

Un nuevo elemento: el aprendizaje de la representación

- Un ejemplo clásico de dificultad de encontrar un conjunto de características es cuando trabajamos con imágenes
- En 1990 un estudio permitió predecir si era necesaria o no una cesárea mediante **regresión logística a partir de un vector de características que un médico** era capaz de obtener a partir de una ecografía
- Es un problema, por tanto, de ingeniería de características: ¿cuáles son las características correctas para medir desde el punto de vista del médico?, ¿son precisas?, ¿nos faltan características o patrones que pueden aportar elementos importantes en la decisión final, incluso no considerados por los mejores médicos?
- ¿No sería mejor si una técnica avanzada de ML pudiera encontrar **las mejores características** y patrones de las imágenes sin procesar?

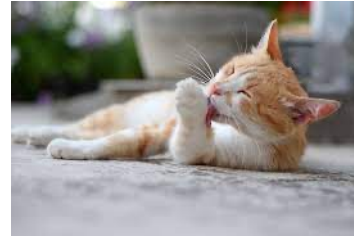




1. Background

Un nuevo elemento: el aprendizaje de la representación

- ¿Qué conjunto de rasgos obtenemos para caracterizar lo que es un gato? ¿Qué técnicas de preprocesado de imágenes debemos aplicar y qué elementos obtenemos de sus resultados? ¿Valores estadísticos? ¿Operadores morfológicos? ¿Técnicas de detección de bordes? ¿Qué podemos obtener de esos procesos y cómo los relacionamos con importantes vectores finales de características?

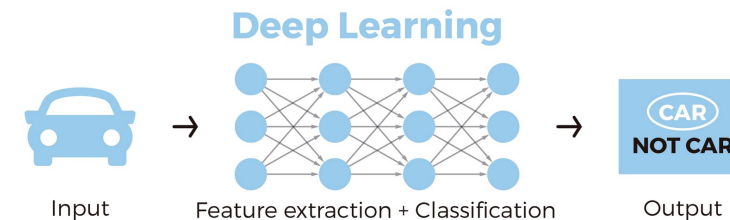
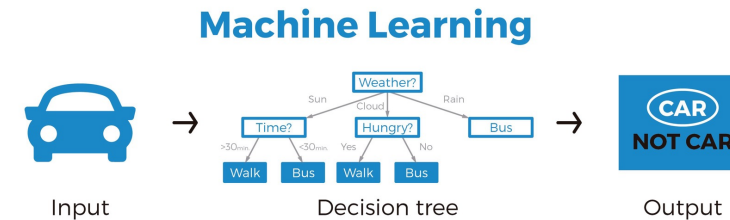




1. Background

Un nuevo elemento: el aprendizaje de la representación

- En el **"aprendizaje de representación"**, el modelo aprende patrones a partir de ejemplos y **también decide cuál es la mejor colección de características** a las que prestar atención en el proceso de aprendizaje
- El aprendizaje profundo es la herramienta de aprendizaje de representación

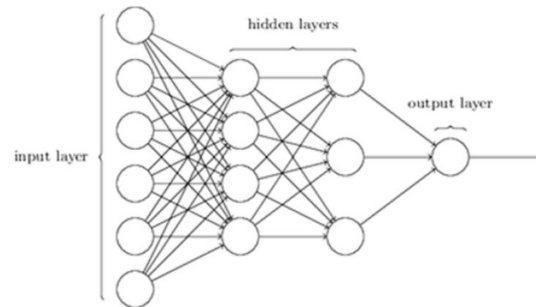




1. Background

Un poco de historia

- A partir de los **años 80** comienza un **renovado interés** por las redes neuronales, principalmente gracias al movimiento conexionista
- La idea del conexionismo es que un gran número de **pequeñas unidades de cómputo** (neuronas artificiales) pueden alcanzar un comportamiento inteligente a través de su conexión en una red
- El algoritmo de retropropagación (backpropagation), una pieza clave en el entrenamiento de redes neuronales, se concibió inicialmente en esta época
- En los años 90 se introducen las redes LSTM secuenciales, aunque las SVM siguen siendo la mejor opción
- Sobre 2006 muchos avances en el campo de Geoffrey Hinton, Yoshua Bengio y Yann LeCun dan un gran impulso a las redes neuronales.
- 2012 con **AlexNet**, comienza la explosión del aprendizaje profundo



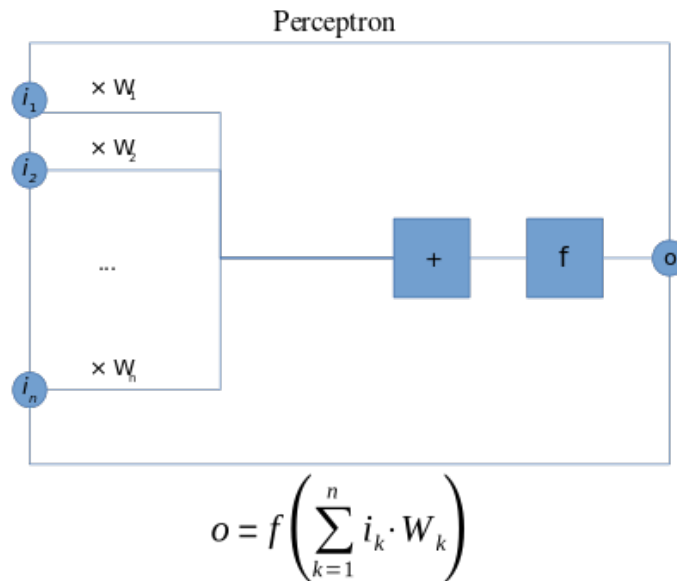
← Una red neuronal con más de 1 capa oculta -> aprendizaje profundo



1. Background

Un poco de historia

- El origen del concepto de neurona artificial fue introducido inicialmente en 1943 por los investigadores McCulloch-Pitts (Threshold Logic Unit, o Unidad de Umbral Lineal)
- En 1958 se dio a conocer el concepto de **perceptrón** (clasificador binario) de Frank Rosenblatt:



$$f(x) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} + b > 0 \\ 0 & \text{otherwise} \end{cases}$$



2. Neurona artificial

Estudiamos algo similar: la regresión logística

- Es obvia la similitud de lo que llamaremos una neurona artificial con la **regresión logística** y, por lo tanto, cómo el algoritmo de descenso de gradiente va a ser una parte esencial de su proceso de entrenamiento

$$z = w^T x + b$$

Linear weights Vector of features Bias (value of z for $x = 0$)

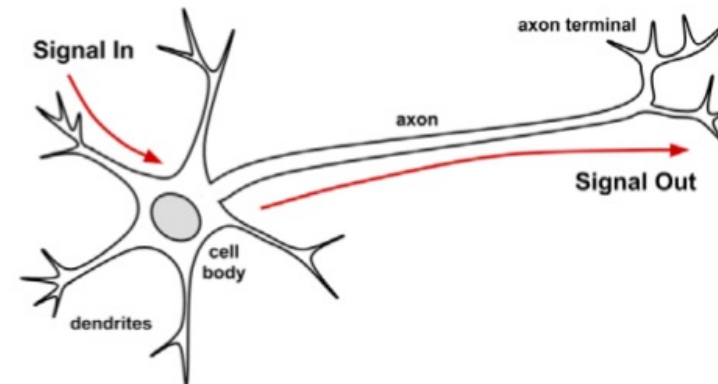
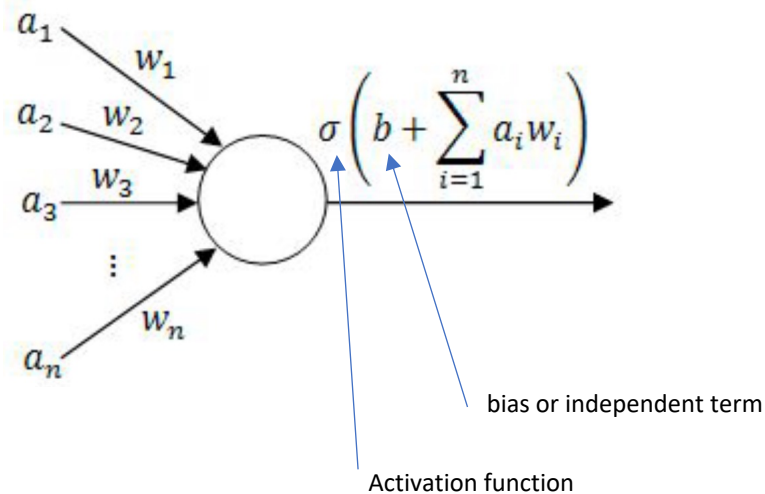
$$a = \frac{1}{1 + e^{-z}}$$
$$y_{predict} = a = \text{sigmoid}(z)$$



2. Neurona artificial

Una neurona o unidad única

- Una serie de **entradas** ($a_1, a_2, a_3 \dots a_n$)
- Una serie de **pesos** ($w_1, w_2, w_3 \dots w_n$)
- Un **bias** (b). Es una especie de predisposición, que permite a la neurona modelar un valor constante e independiente de las entradas que afecta a la salida de la neurona
- Una **función de activación** que se aplica a la suma del sesgo y el producto de cada entrada con su correspondiente peso



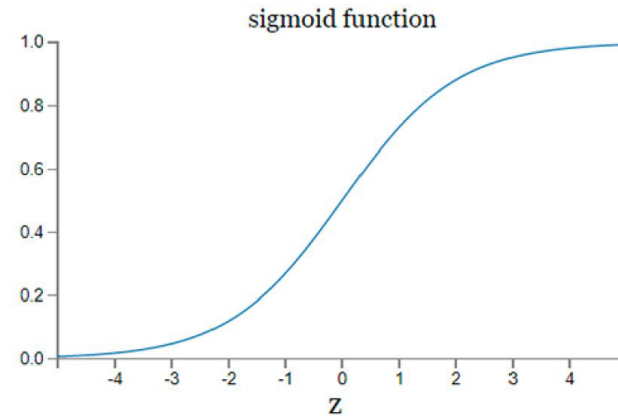


2. Neurona artificial

Una neurona o unidad única

- La función de activación recibe también el nombre de **no linealidad**
- Veremos muchas funciones de activación, donde la función sigmoide todavía se usa cuando queremos modelar probabilidades

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

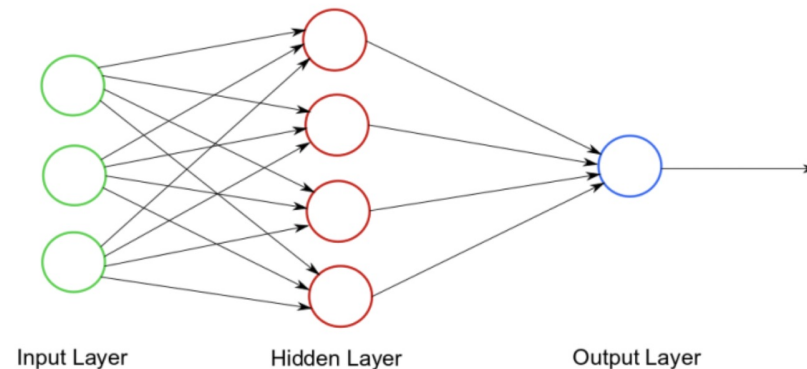




3. Redes neuronales artificiales

NN – redes neuronales

- Para obtener una breve descripción general de lo que son las Redes Neuronales (NN), son simplemente una **colección de neuronas**, que están conectadas a través de **varias capas**. Intenta aprender la asignación de los datos de entrada a los datos de salida, cuando se le proporciona un conjunto de entrenamiento.
- El entrenamiento de la red neuronal facilita posteriormente las **predicciones** realizadas por la misma sobre datos nuevos de la **misma distribución**. Este mapeo se logra mediante un conjunto de parámetros entrenables llamados pesos, distribuidos en diferentes capas. Los pesos son aprendidos por el algoritmo de retropropagación (**backpropagation**), cuyo objetivo es minimizar una función de pérdida. Una función de pérdida mide la distancia a la que se encuentran las predicciones realizadas por la red de los valores reales

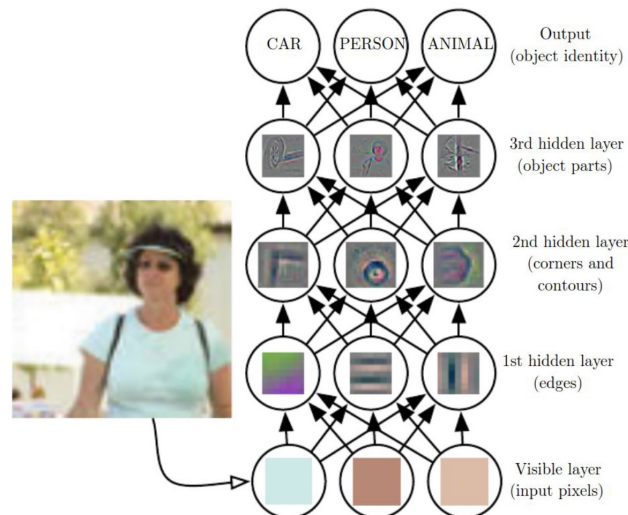




4. Deep learning

NN – neural networks

- En el contexto de imágenes, por ejemplo, aprender una única función directa directamente de los píxeles de un objeto podría ser casi imposible
- Una red neuronal divide esta tarea en una serie de problemas más simples
- Profundo no solo significa más de 1 capa oculta sino: **jerarquía profunda de conceptos**, con un grado creciente de complejidad o abstracción en cada nueva capa (**cada nueva capa es capaz de detectar conceptos más generales**)



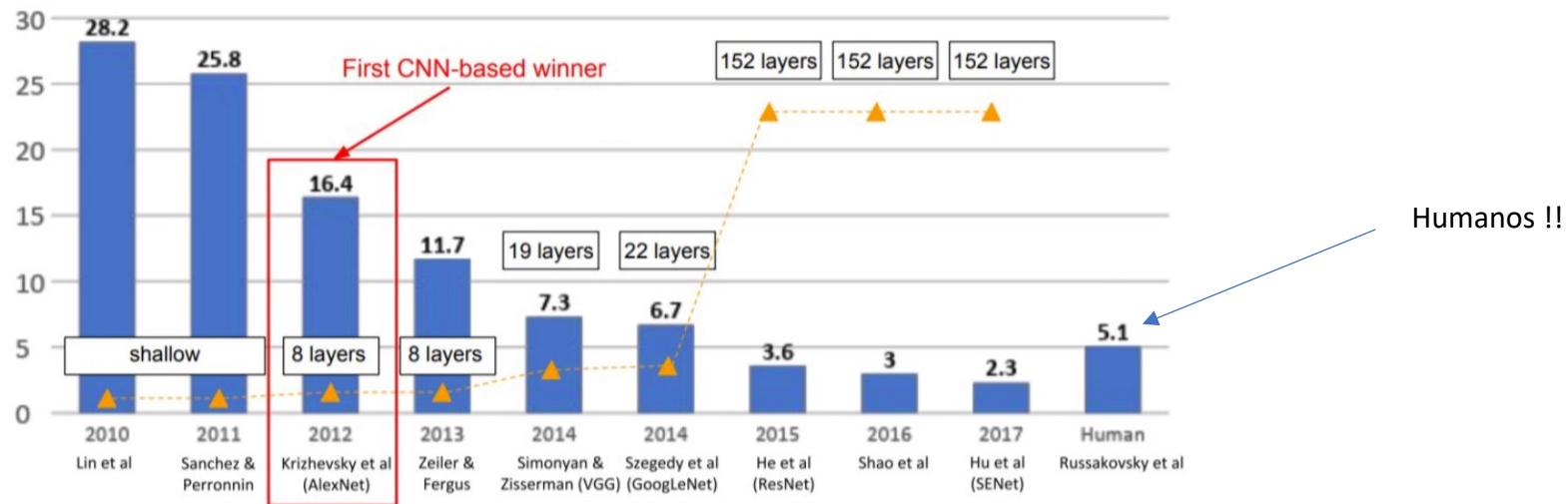
Jerarquía de conceptos: una representación de una complejidad creciente



4. Deep learning

El despegue deep learning (2012)

- **ImageNet contest:** 14 millones de imágenes con 20000 categorías anotadas a mano (como 'globo', 'fresa', etc.)
- Objetivo: minimizar el error utilizando varias estrategias de IA





4. Deep learning

Principales áreas actuales

- **Visión artificial** (redes neuronales convolucionales)
- **Reconocimiento de voz** (lo que permite el asistente de voz actual)
- **Aprendizaje por refuerzo** (mejora basada en deep learning, AlphaGo, AlphaFold 2, etc.)
- **Procesamiento del lenguaje natural (PLN)** como traducción, análisis de sentimientos, modelos de lenguaje (GPT-X)
- **Modelos generativos** (sí, DL puede generar texto, resolver problemas, arte, música, imágenes, películas, etc.)
- Etc.



4. Deep learning

¿Por qué ahora? Elementos clave del éxito del aprendizaje profundo

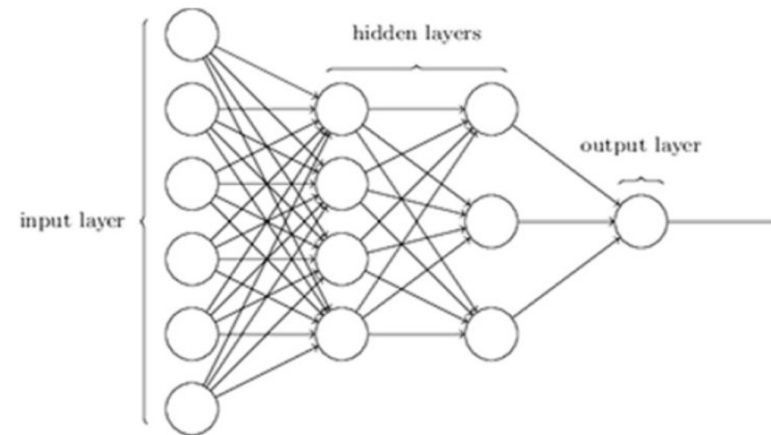
- Algunos avances en la mejora de su **entrenamiento y otras técnicas** (regularización, optimización, nuevas arquitecturas, etc.)
- Una **disponibilidad cada vez mayor de datos** (ahora se dispone de enormes conjuntos de datos, elemento clave para el éxito del entrenamiento de una red neuronal)
- Mejora en la **potencia de cálculo** (como GPU's y TPU's)



4. Deep learning

Neural networks

- El objetivo es poder aprender pesos y bias, **los parámetros de la red neuronal**, a partir de los datos de entrada, tratando de minimizar una función de pérdida/coste
- Con una sola neurona no somos capaces de hacer frente a muchos problemas reales
- Generalmente, cuando hablamos de una red neuronal, nos estamos refiriendo a una de sus clases principales: **red neuronal totalmente conectada feed-forward**, donde las neuronas de una capa están conectadas con todas las neuronas de la capa anterior

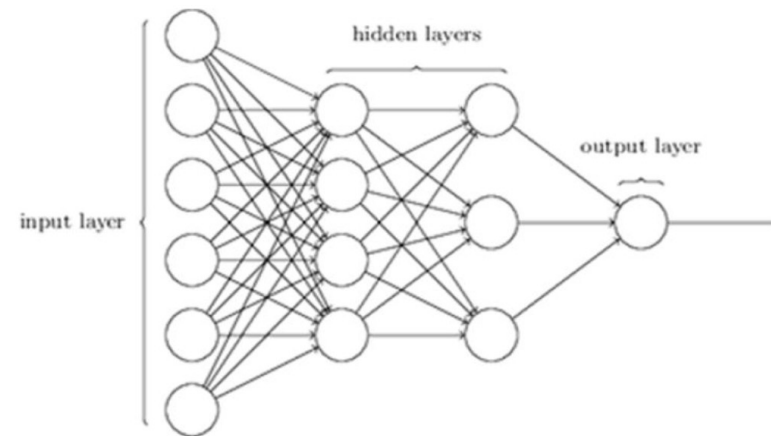




4. Deep learning

Neural networks

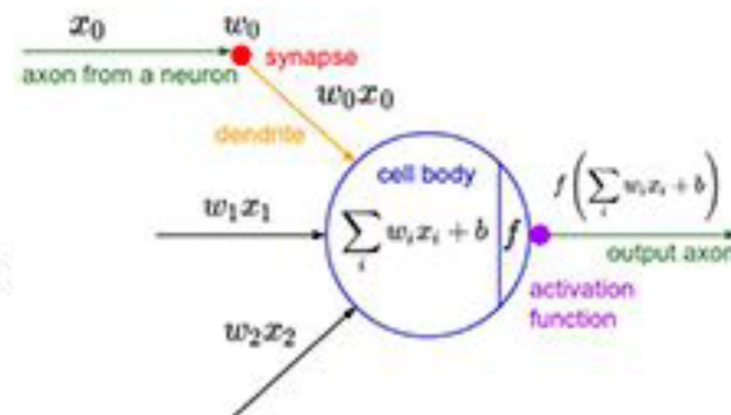
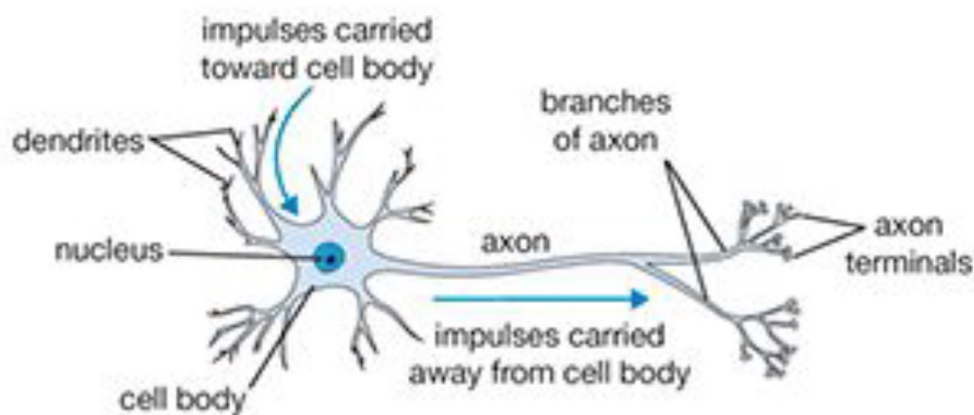
- **Input layer:** capa que recibe los datos con los que trabajar (en realidad no neuronas)
- **Hidden layers:** Capas internas
- **Output layer:** La salida de la red neuronal
- El entrenamiento de una red neuronal es el proceso con el que encontramos los pesos \mathbf{w} y los bias \mathbf{b} de las neuronas que mejor describen los datos de entrada en términos de la salida deseada



4. Deep learning

Inspiración 'biológica'

- Las neuronas biológicas reciben impulsos de otras neuronas, y de la misma manera transmiten un impulso (sinapsis) a otras neuronas
- Las similitudes terminan aquí, eso es todo

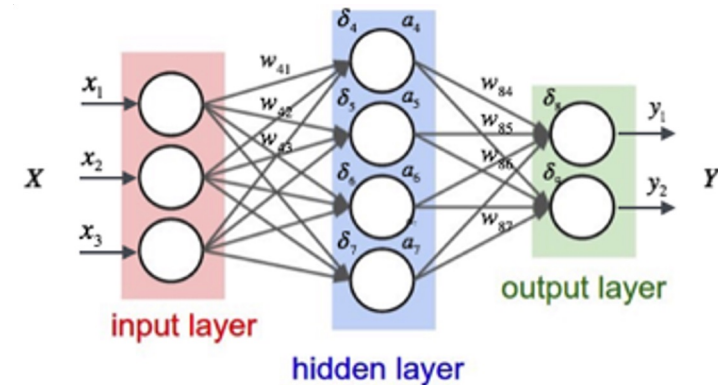
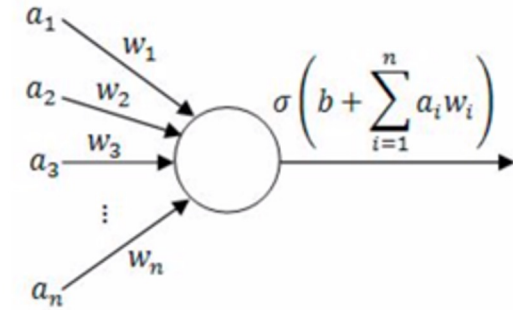




4. Deep learning

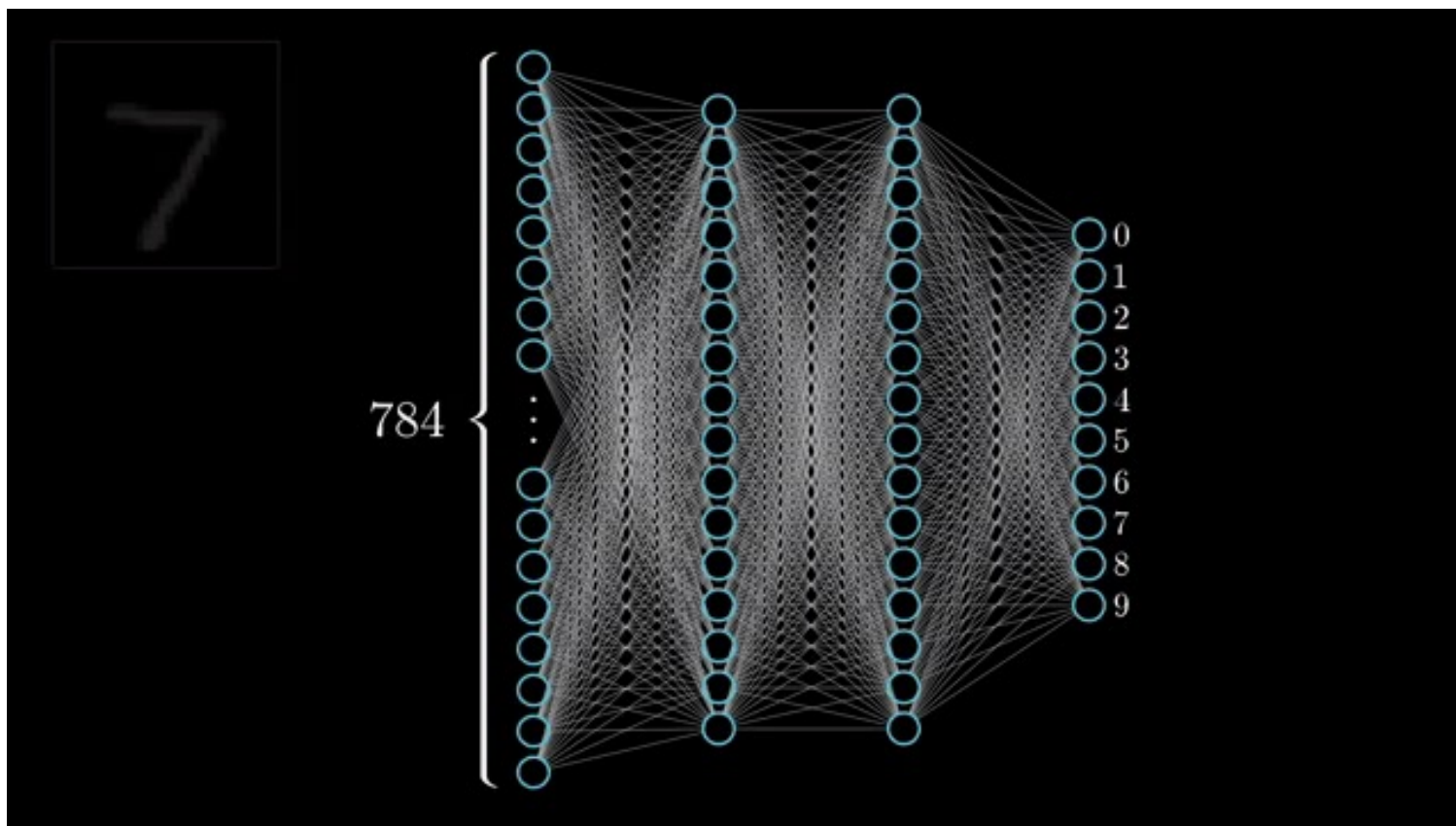
Breve descripción de su proceso de formación

- Primero se **inician aleatoriamente los parámetros**: se puede ver un peso como la importancia de este parámetro para la neurona, y el conjunto final de valores de todos los pesos y bias es el conocimiento "real" que la red neuronal finalmente ha aprendido del conjunto de datos de entrada
- Se introducen ejemplos (multiplicando por los pesos, la función de activación a través de la red, etc.) hasta que se produce un resultado
- Se calcula el gradiente de la función de pérdida y un algoritmo llamado **backpropagation** envía gradientes hacia atrás y se actualizan los pesos y el bias de las neuronas (el algoritmo de descenso de gradiente será la clave)
- Este proceso se repite con todos los ejemplos (cuantos más, mejor), y se evalúa continuamente (precisión) hasta conseguir los resultados esperados



4. Deep learning

Predicción

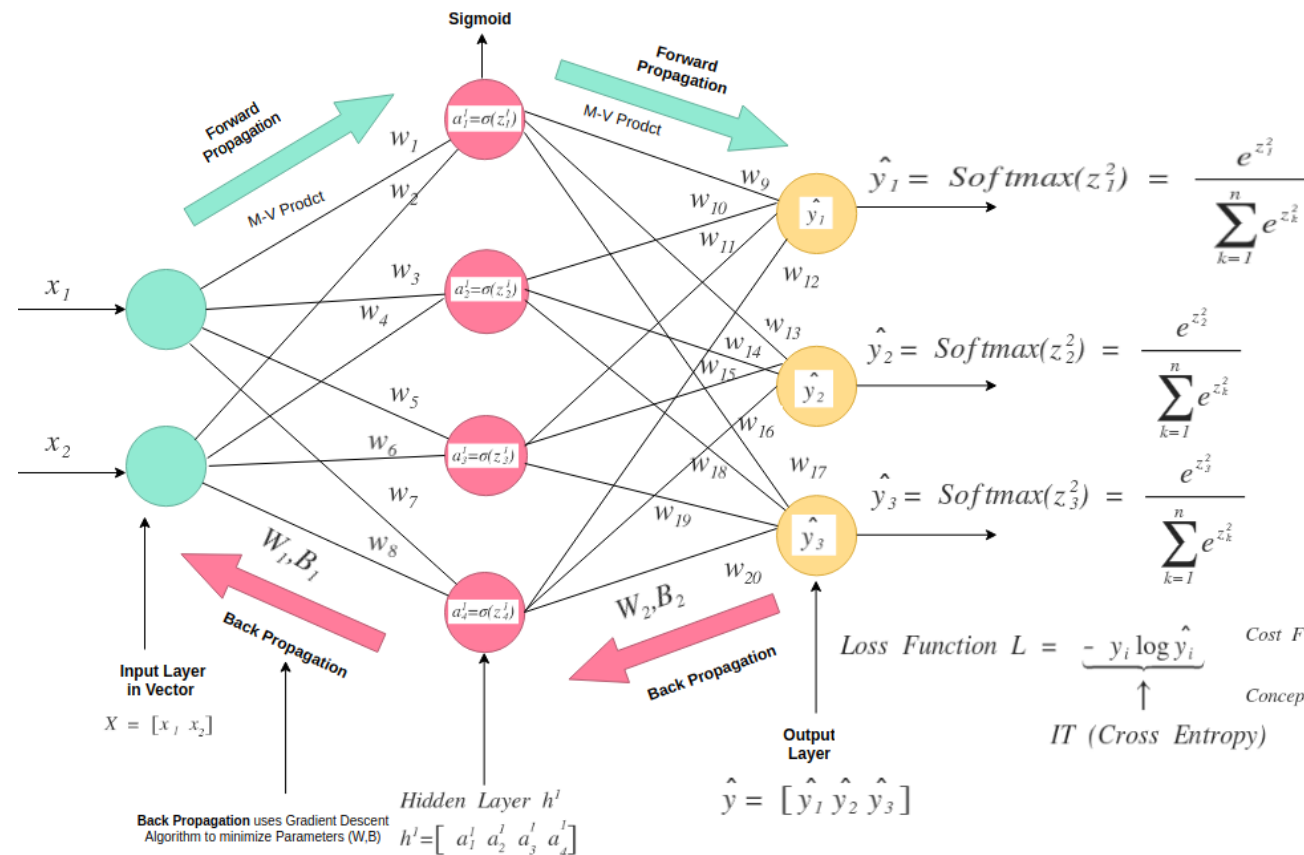


<https://www.youtube.com/watch?v=aircAruvnKk>



4. Deep learning

Entrenar a una NN es interesante



Mathematics & Statistical subjects used in Neural Network

M-V Product - **Linear Algebra** - Matrix -Vector Product

Sigmoid - **Statistics** - Sigmoid Probability Distribution

IT - **Information Theory** Cross Entropy Function used

Concept taken from OR (**Operational Research**)

Softmax Function uses **Probability** concept

Forward Propagation uses product and addition rules of LA

Back Propagation uses **Real Analysis** (Gradient Descent),
Numerical Optimization (Minima), **Convex Optimization**,
All the required **Basic Functions** (Graphs, Curves, etc.,)

Notation

Element^{Layer}_{Node} eg: z_2^1

Where z - Intermediate Variable; 1-> layer & node 2

Vector^{Layer} eg: h^1 h - First hidden Layer

\hat{y} -> Estimated output y denoted by \hat{y}

4. Deep learning

Trade-offs

