

Redes Neuronales Artificiales

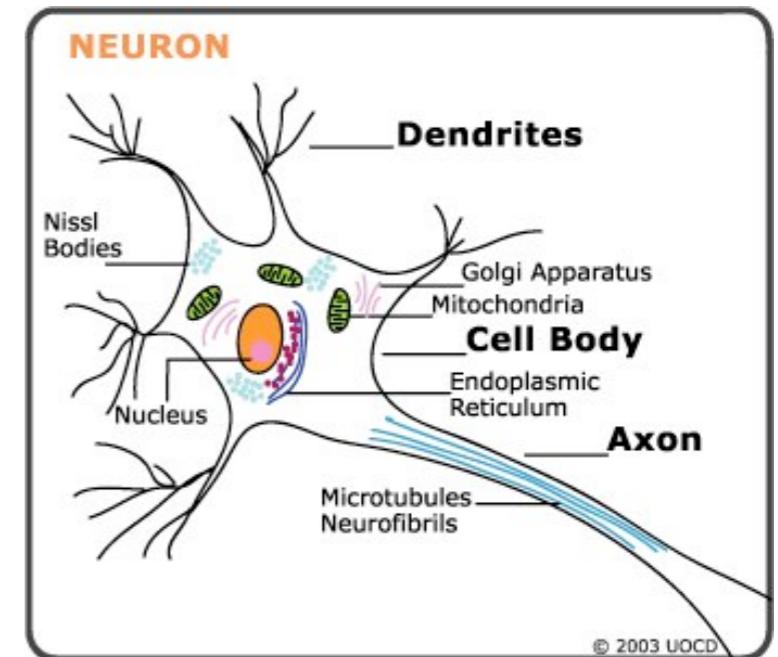


<https://github.com/biuse/2021 Neural Networks>

REDES NEURONALES ARTIFICIALES

Herramienta de *machine learning* que nació con dos objetivos:

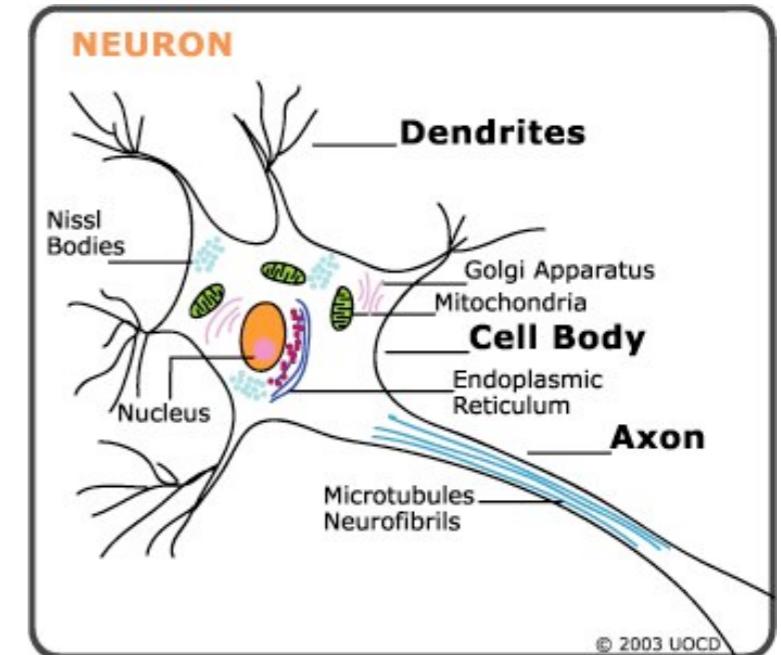
- Modelizar las neuronas biológicas
- Inteligencia artificial



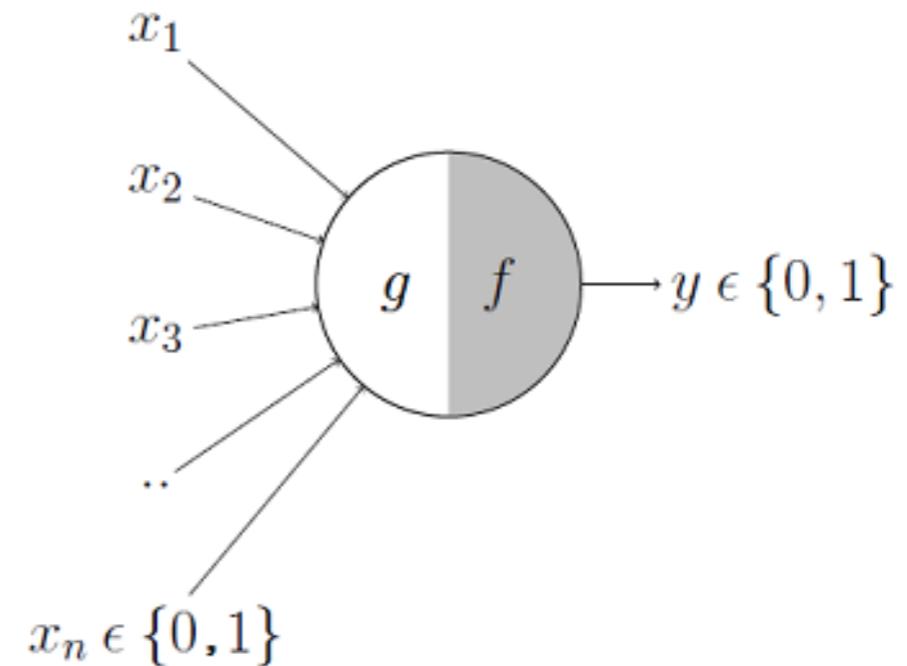
REDES NEURONALES ARTIFICIALES

Herramienta de *machine learning* que nació con dos objetivos:

- Modelizar las neuronas biológicas
- Inteligencia artificial



Neurona artificial recibe
una señal y devuelve 0 o 1

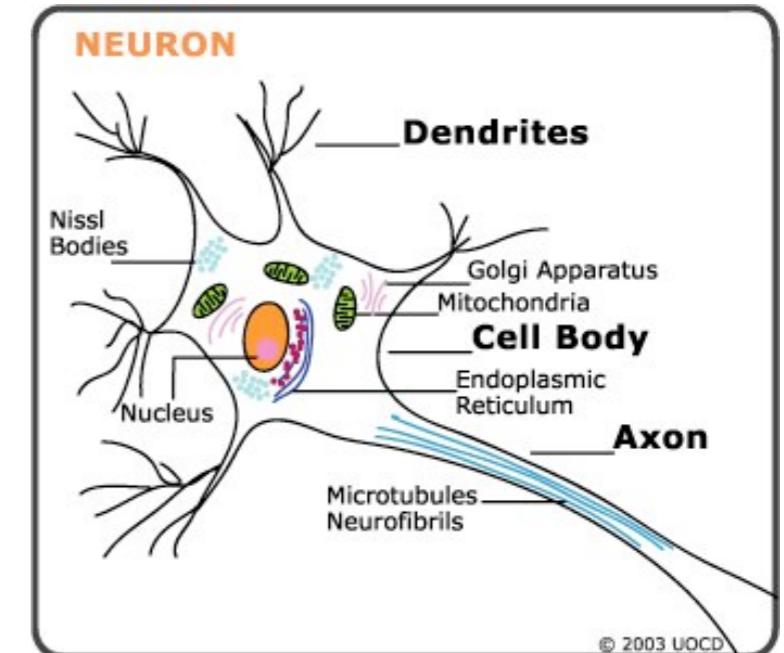


Modelo de McCulloch and Pitts 1943

REDES NEURONALES ARTIFICIALES

Herramienta de *machine learning* que nació con dos objetivos:

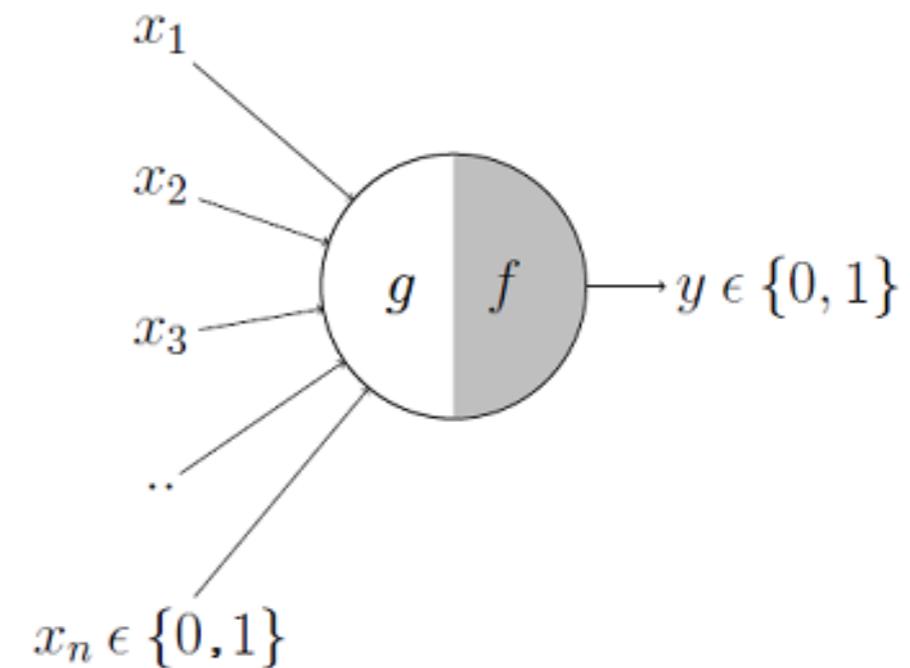
- Modelizar las neuronas biológicas
- Inteligencia artificial



Neurona artificial recibe
una señal y devuelve 0 o 1

$$\begin{aligned} f(g(\mathbf{x})) &= 1 \quad \text{if} \quad g(\mathbf{x}) \geq \theta \\ &= 0 \quad \text{if} \quad g(\mathbf{x}) < \theta \end{aligned}$$

$$g(\mathbf{x}) = \sum_{i=1}^n x_i$$

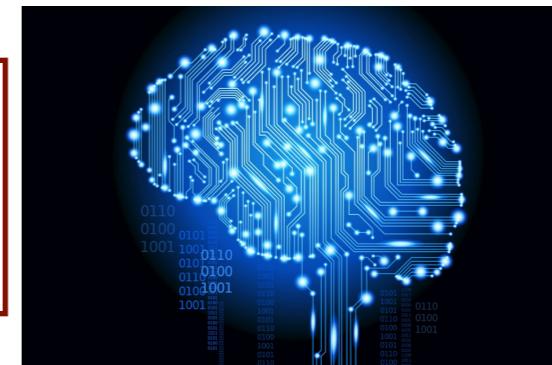


Modelo de McCulloch and Pitts 1943

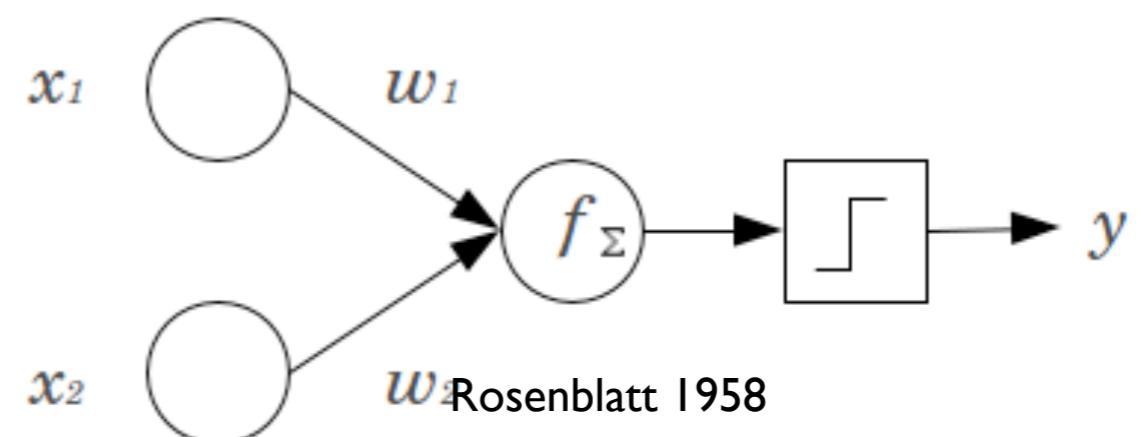
REDES NEURONALES

Herramienta de *machine learning* que nació con dos objetivos:

- modelizar las neuronas biológicas
- la inteligencia artificial



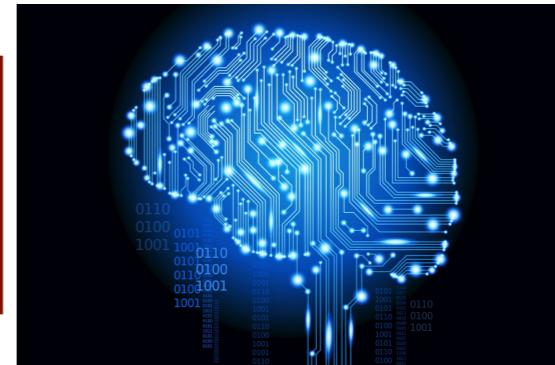
Se añadieron unos **pesos**
que podían ajustarse.



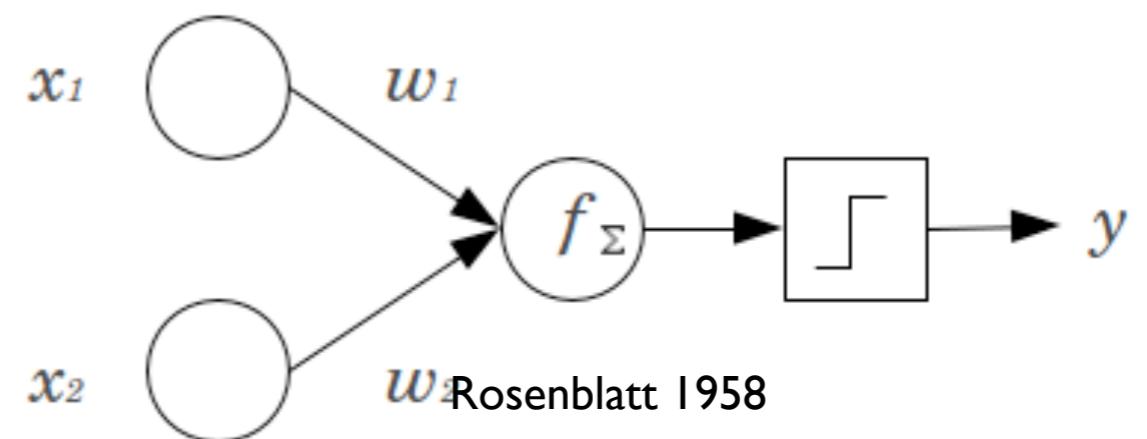
REDES NEURONALES

Herramienta de *machine learning* que nació con dos objetivos:

- modelizar las neuronas biológicas
- la inteligencia artificial



Se añadieron unos **pesos**
que podían ajustarse.

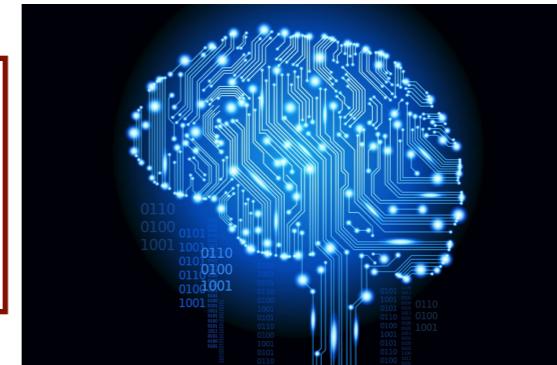


¡Y funcionó!

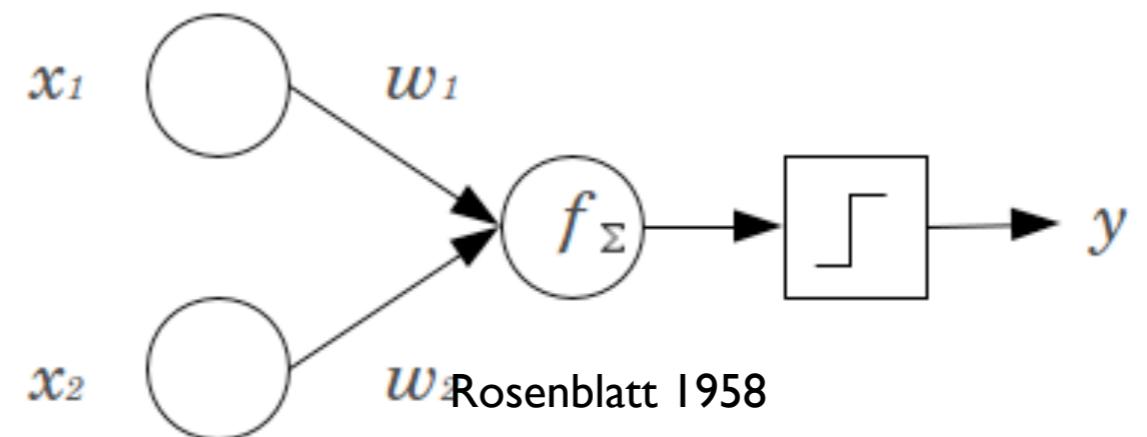
REDES NEURONALES

Herramienta de *machine learning* que nació con dos objetivos:

- modelizar las neuronas biológicas
- la inteligencia artificial



Se añadieron unos **pesos**
que podían ajustarse.



¡Y funcionó!

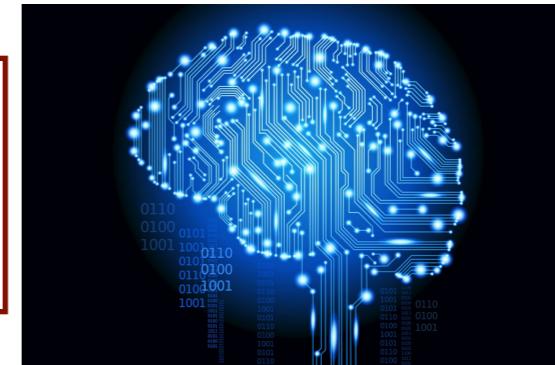
$$\begin{aligned} f(g(\mathbf{x})) &= 1 & \text{if} & \quad g(\mathbf{x}) \geq \theta \\ &= 0 & \text{if} & \quad g(\mathbf{x}) < \theta \end{aligned}$$

$$g(x) = \sum_i w_i x_i$$

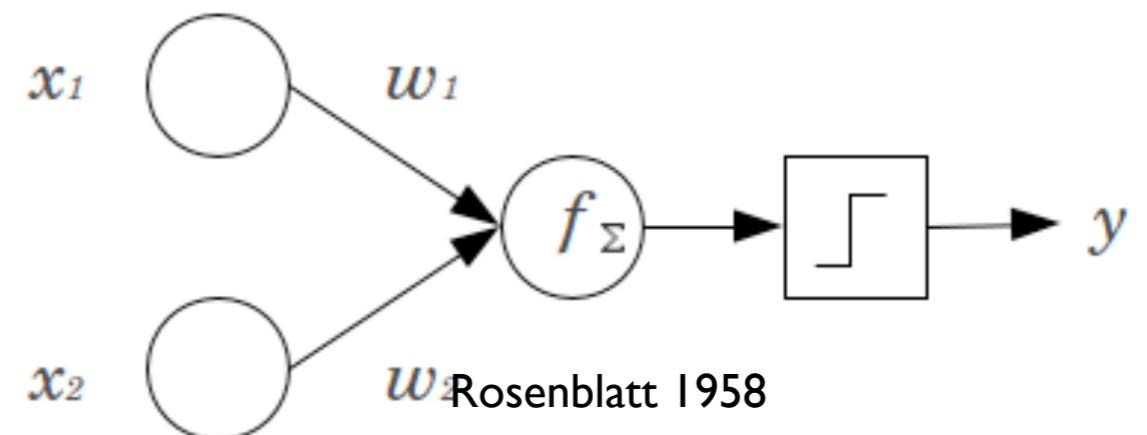
REDES NEURONALES

Herramienta de *machine learning* que nació con dos objetivos:

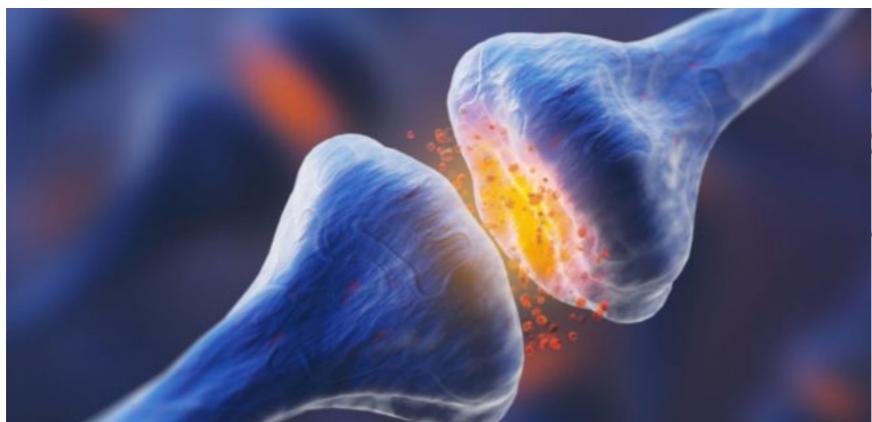
- modelizar las neuronas biológicas
- la inteligencia artificial



Se añadieron unos **pesos** que podían ajustarse.



¡Y funcionó!

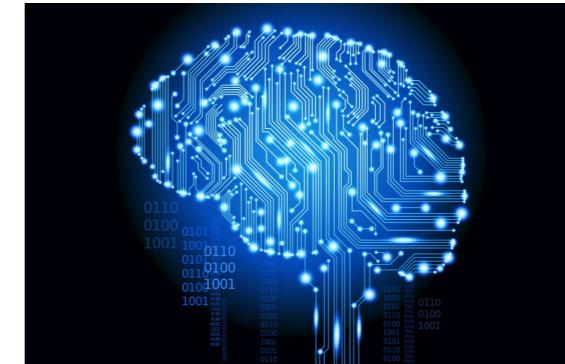


La **sinápsis**, la fortaleza de la conexión entre neuronas era fundamental para el aprendizaje

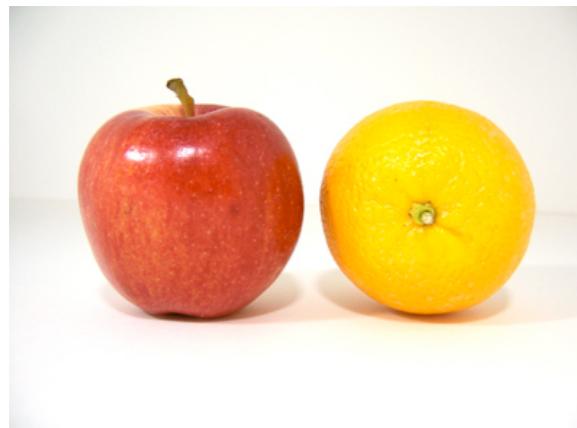
REDES NEURONALES

Herramienta de *machine learning* que nació con dos objetivos:

- modelizar las neuronas biológicas
- la inteligencia artificial



El perceptrón coge una señal, la procesa con una operación matemática que devuelve 0 o 1. Pero permite variar unos pesos, de tal manera que la máquina puede aprender.

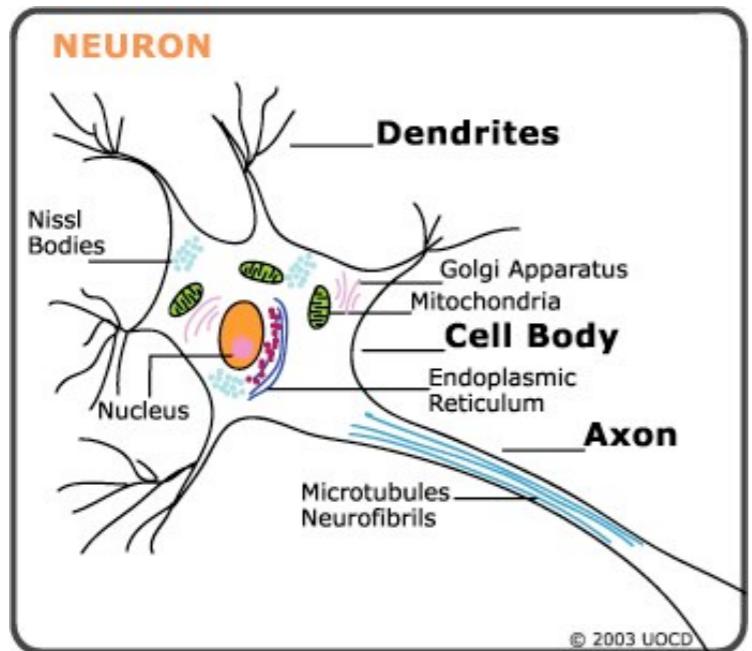


Nos podría, por ejemplo, clasificar manzanas (clase 0) y naranjas (clase 1)

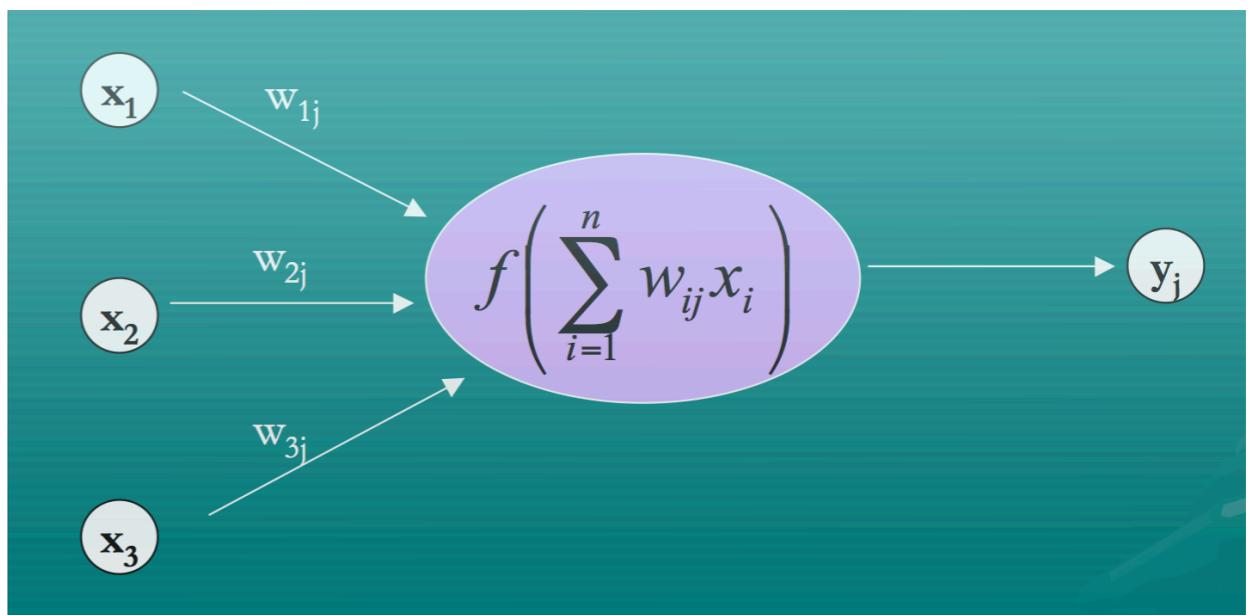
<https://towardsdatascience.com/perceptron-explanation-implementation-and-a-visual-example-3c8e76b4e2d1>

REDES NEURONALES

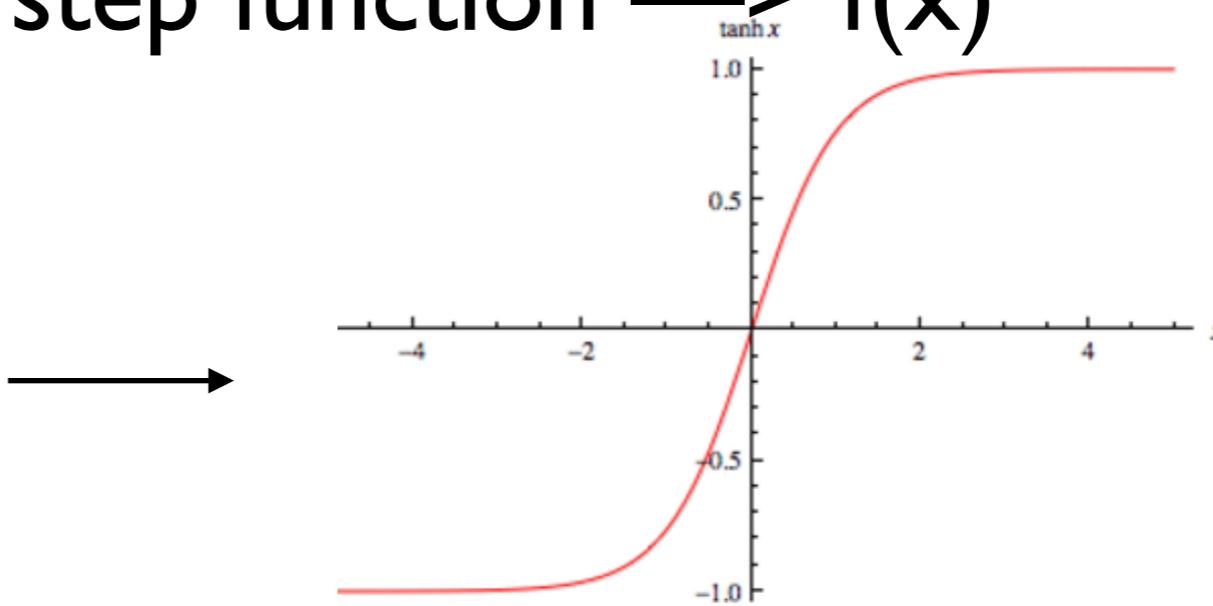
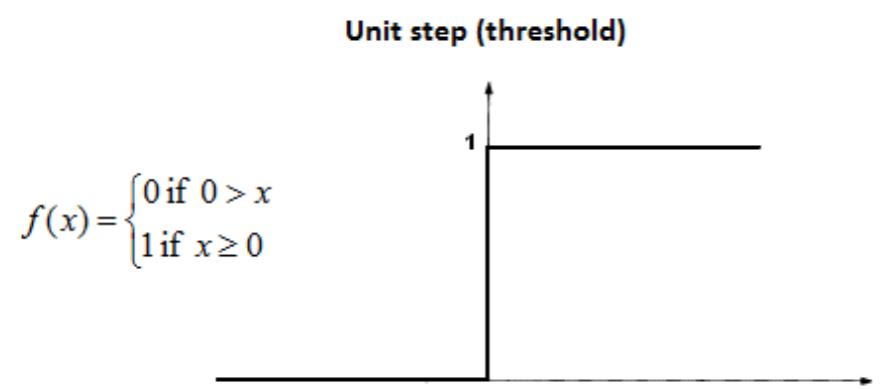
Neurona biológica (simplificación)



Neurona artificial

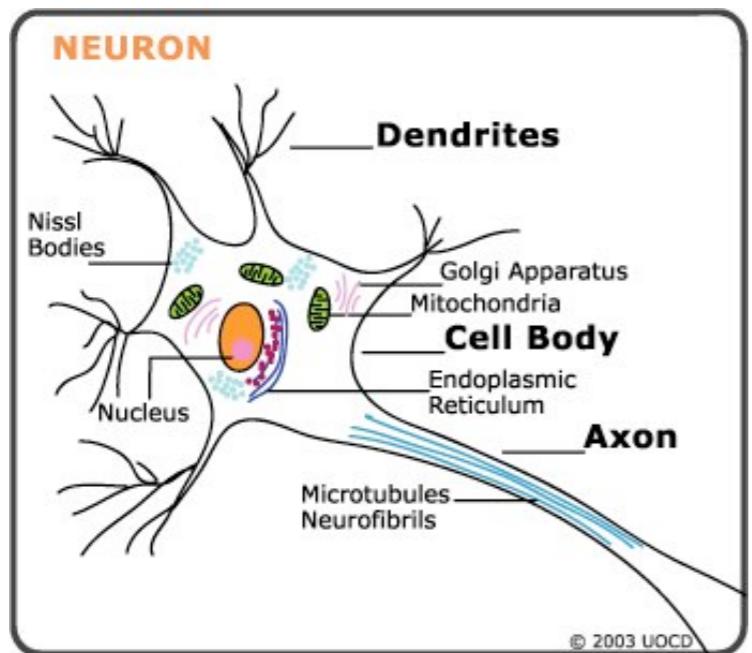


Se generalizó step function $\rightarrow f(x)$

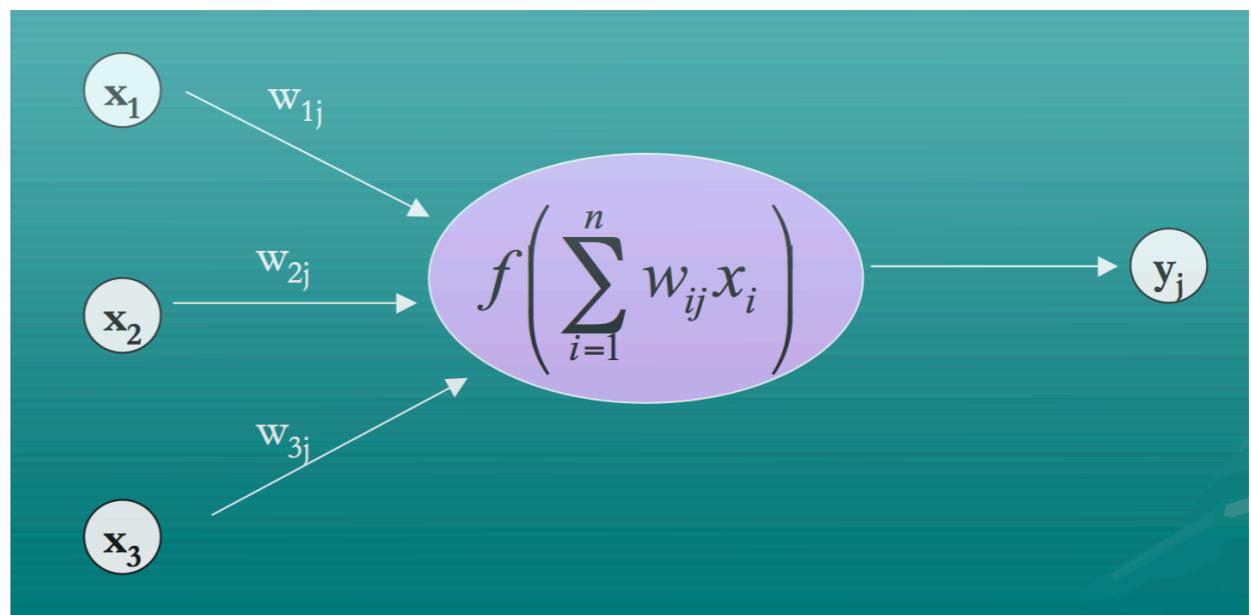


REDES NEURONALES

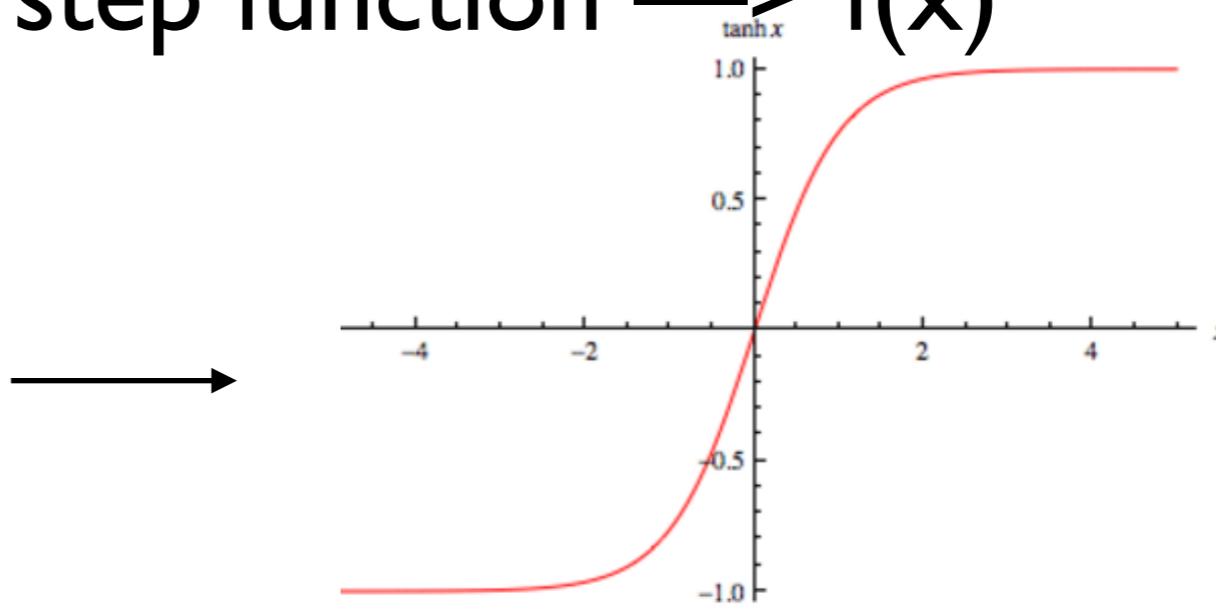
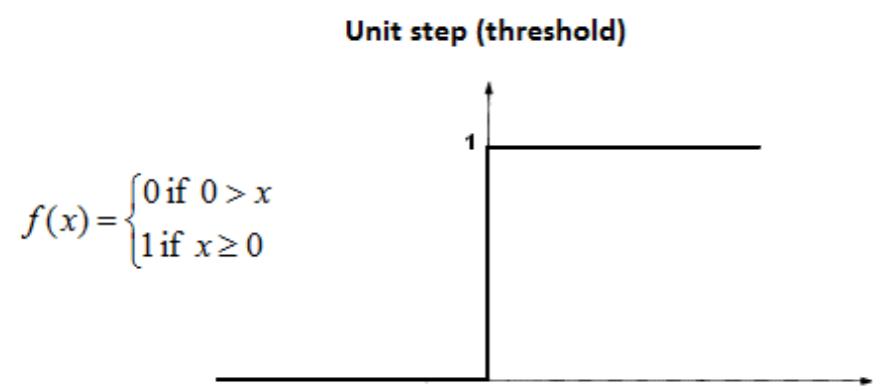
Neurona biológica (simplificación)



Neurona artificial



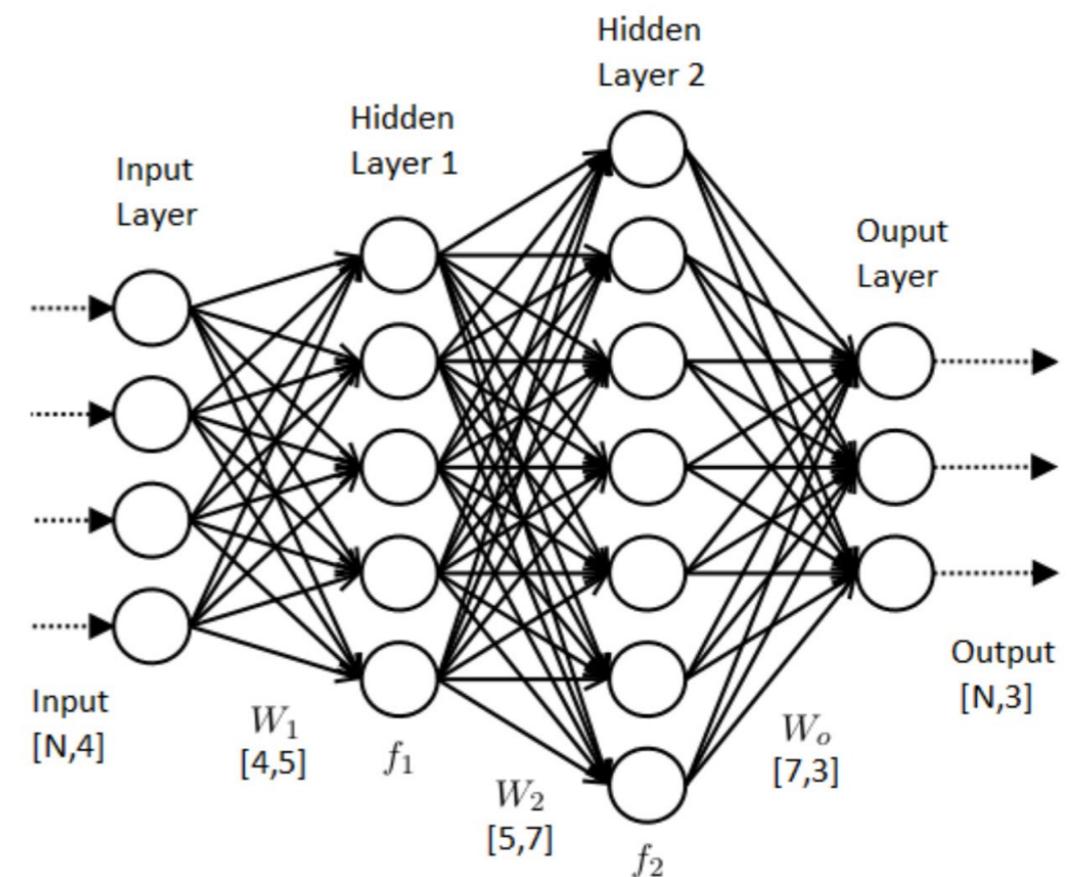
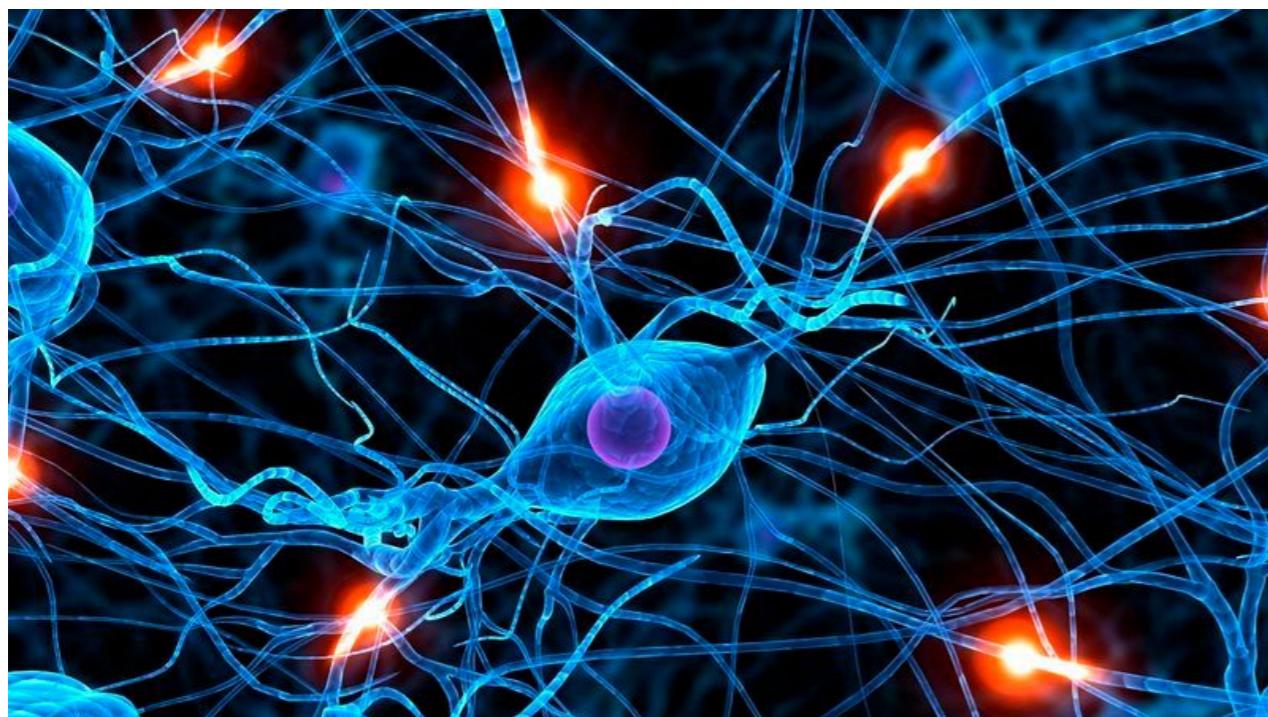
Se generalizó step function $\rightarrow f(x)$



REDES NEURONALES

Red neuronal biológica

Un cerebro humano tiene alrededor de 10^{11} neuronas, cada una de ellas con unos pocos de miles de conexiones.



Red neuronal artificial

Se organizan por capas de neuronas conectadas entre ellas.

Una red neuronal biológica es muy compleja, la ANN que presentamos aquí tiene sus limitaciones, pero es muy eficiente en algunos problemas.

REDES NEURONALES

TIPOS

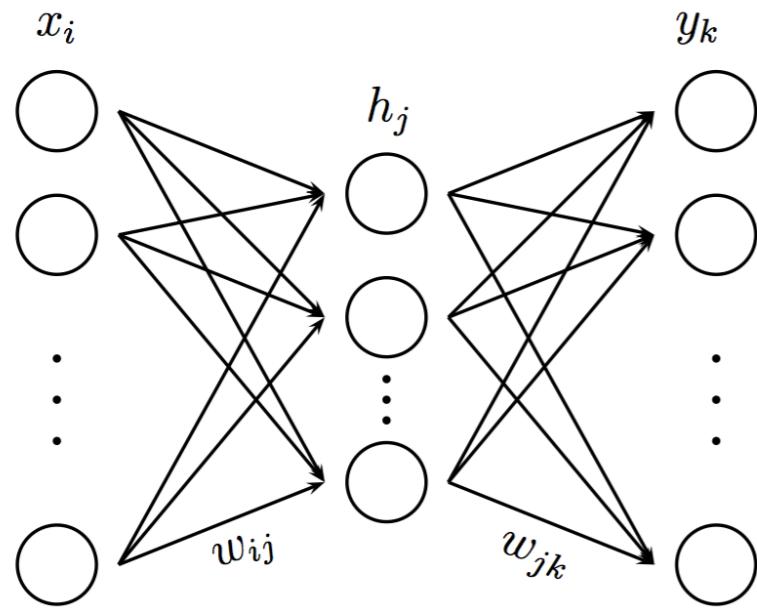
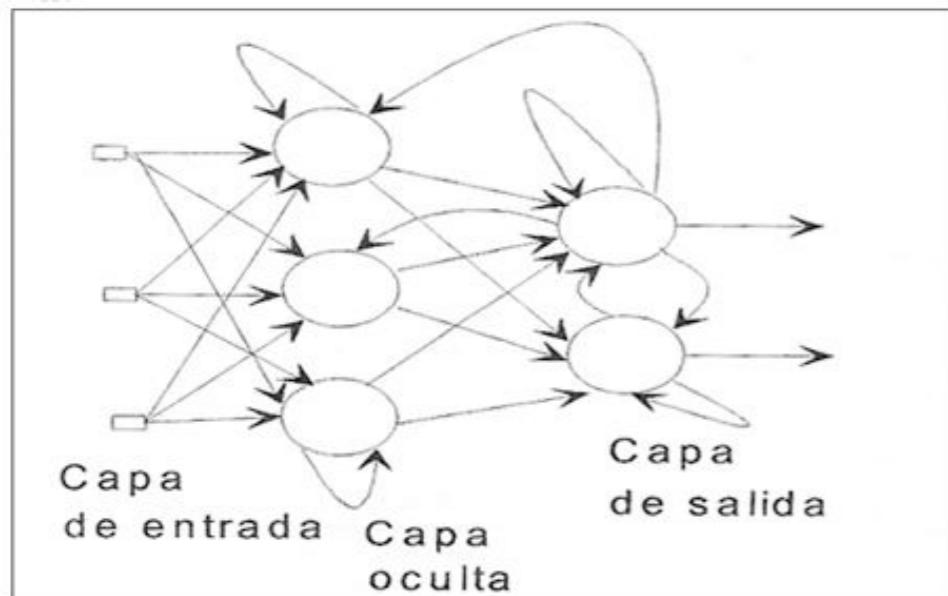
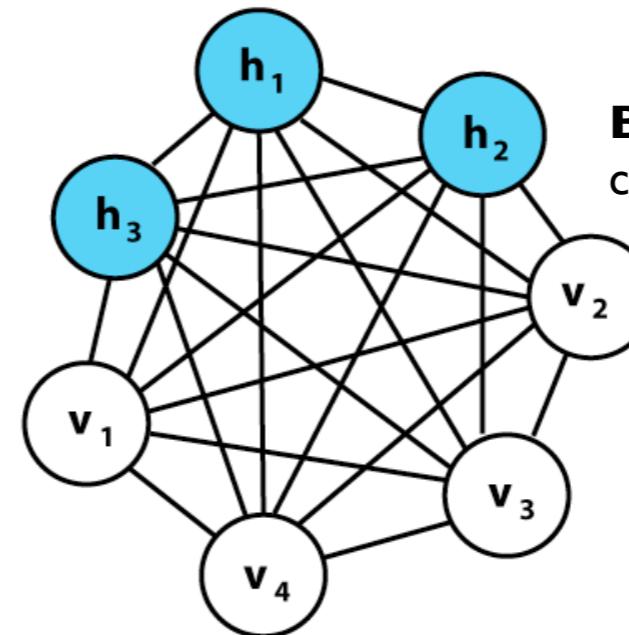


Figure 1. Schematic of a 3-layer feed-forward neural network.

Redes neuronales recurrentes. Pueden haber conexiones en varias direcciones



Feed-forward. Solo conexiones hacia delante



Boltzmann machine. Todas conectadas con todas

Redes neuronales convolucionales. deep learning

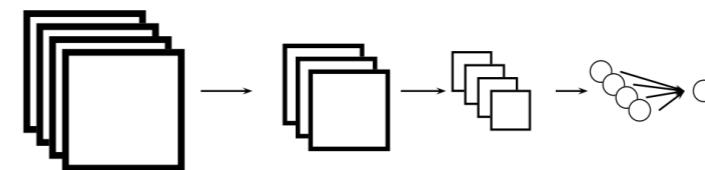


Figure 2. Schematic convolutional neural network example, with 2 hidden layers in the CNN part and 1 hidden layer in the FC part. In this example, the input depth is $k = 4$, the first layer depth $m_1 = 3$ and the second layer depth is $m_2 = 4$, the number of hidden nodes in the FC hidden layer is 4, and only one output is present.

REDES NEURONALES

TIPOS

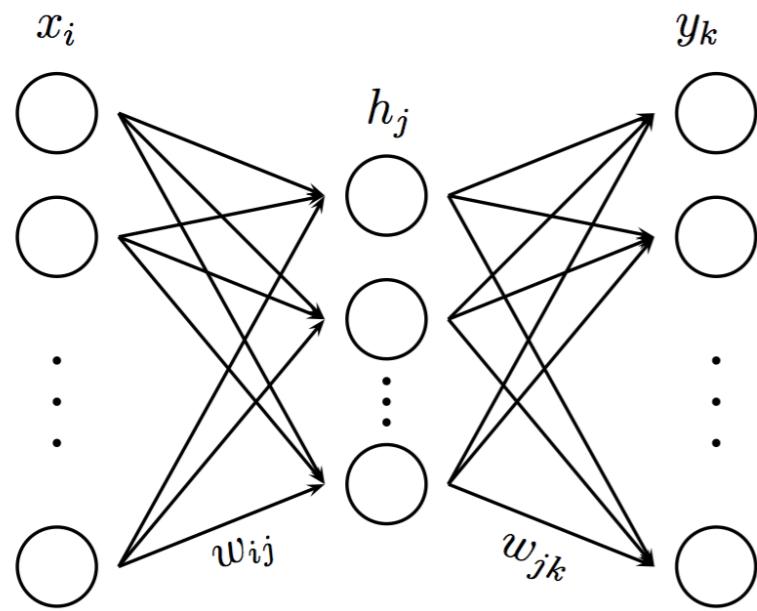
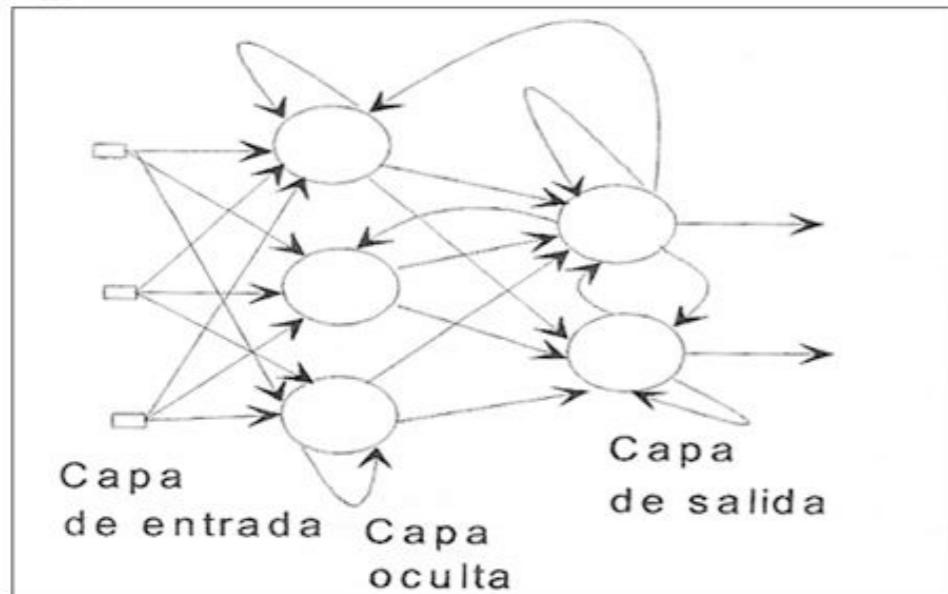
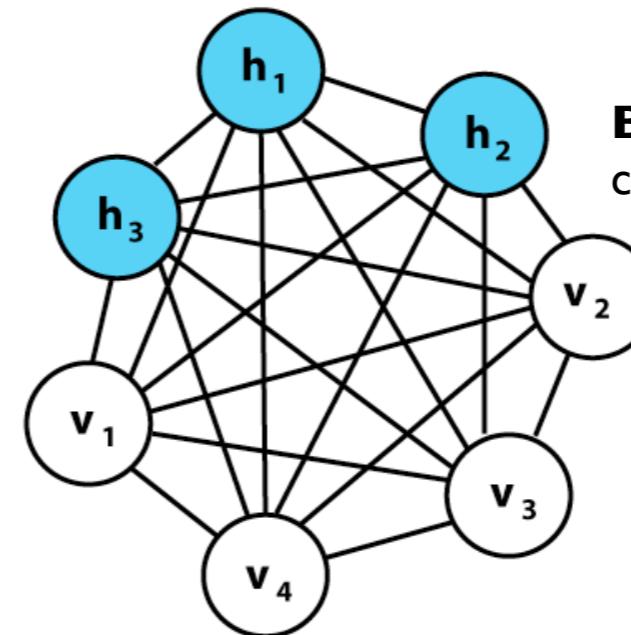


Figure 1. Schematic of a 3-layer feed-forward neural network.

Redes neuronales recurrentes. Pueden haber conexiones en varias direcciones



Feed-forward. Solo conexiones hacia delante



Boltzmann machine. Todas conectadas con todas

Redes neuronales convolucionales. deep learning

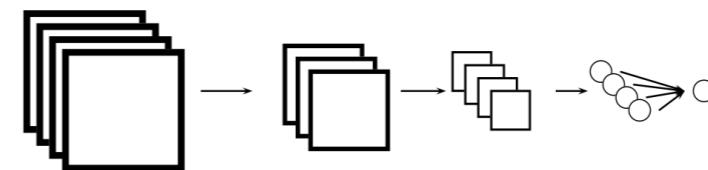
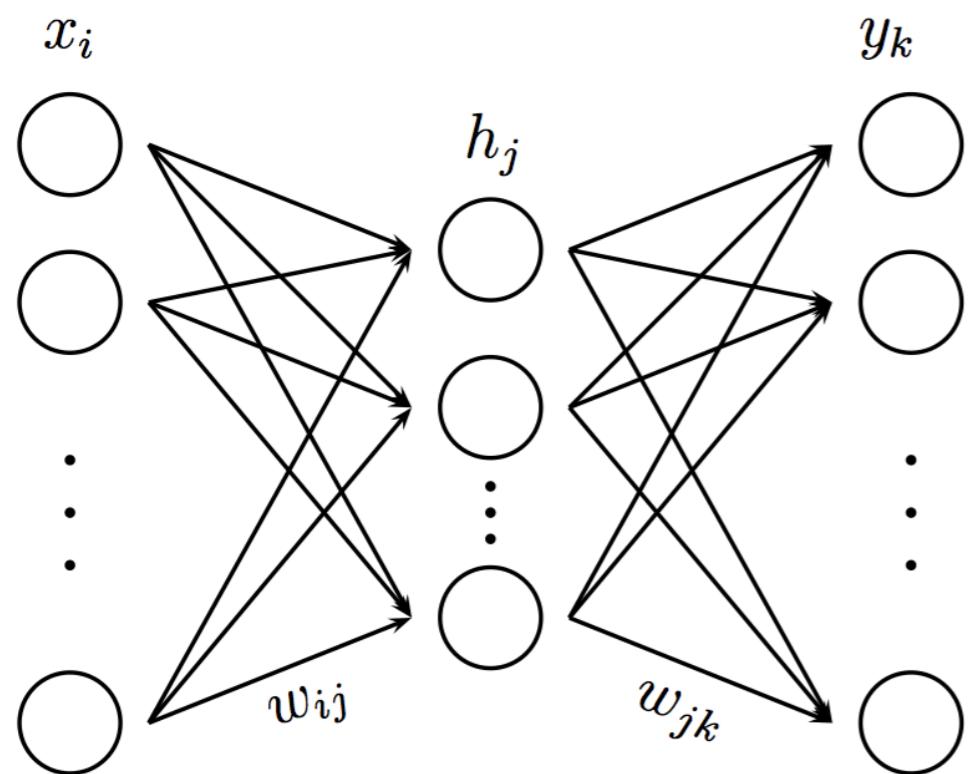


Figure 2. Schematic convolutional neural network example, with 2 hidden layers in the CNN part and 1 hidden layer in the FC part. In this example, the input depth is $k = 4$, the first layer depth $m_1 = 3$ and the second layer depth is $m_2 = 4$, the number of hidden nodes in the FC hidden layer is 4, and only one output is present.

REDES NEURONALES



Una red neuronal feed forward se organiza por capas conectadas las unas con las otras.

Figure 1. Schematic of a 3-layer feed-forward neural network.

REDES NEURONALES

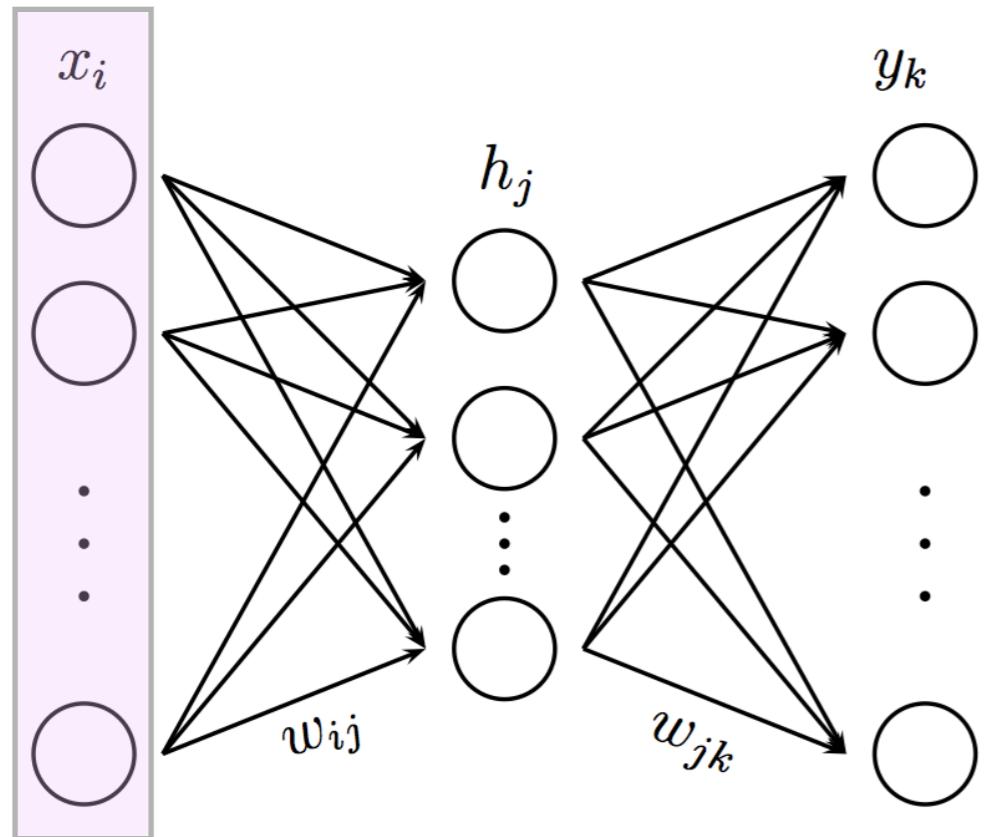


Figure 1. Schematic of a 3-layer feed-forward neural network.

Una red neuronal feed forward se organiza por capas conectadas las unas con las otras.

Están la capa inicial de **inputs** x

REDES NEURONALES

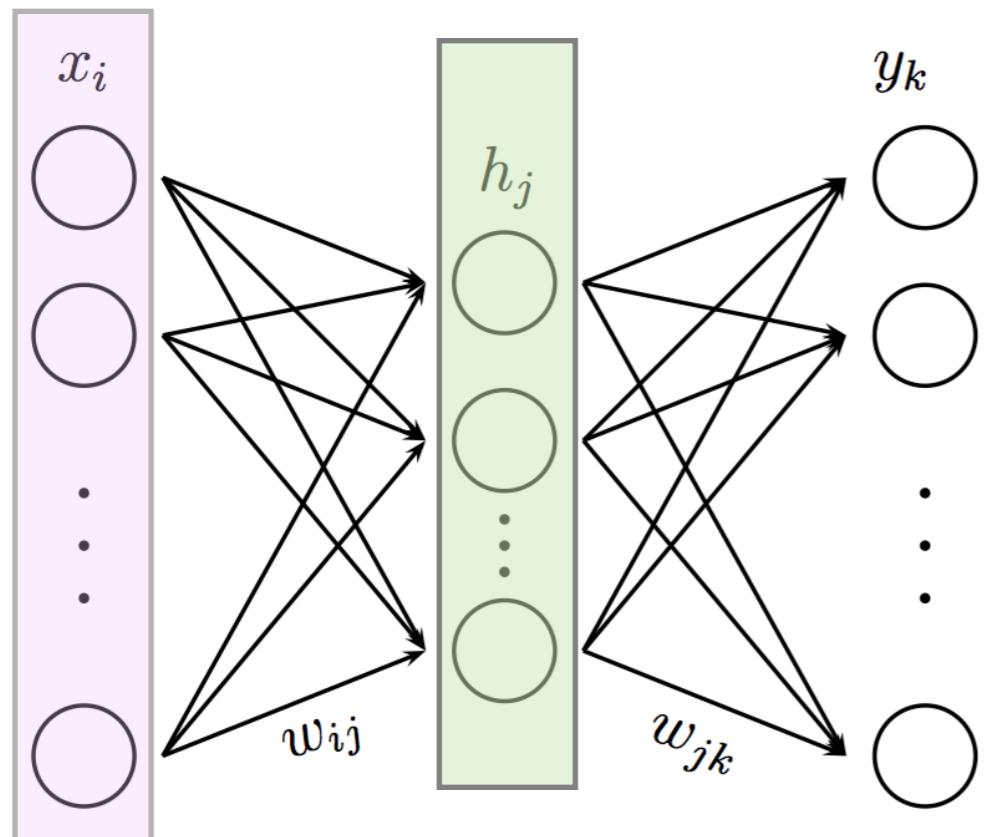


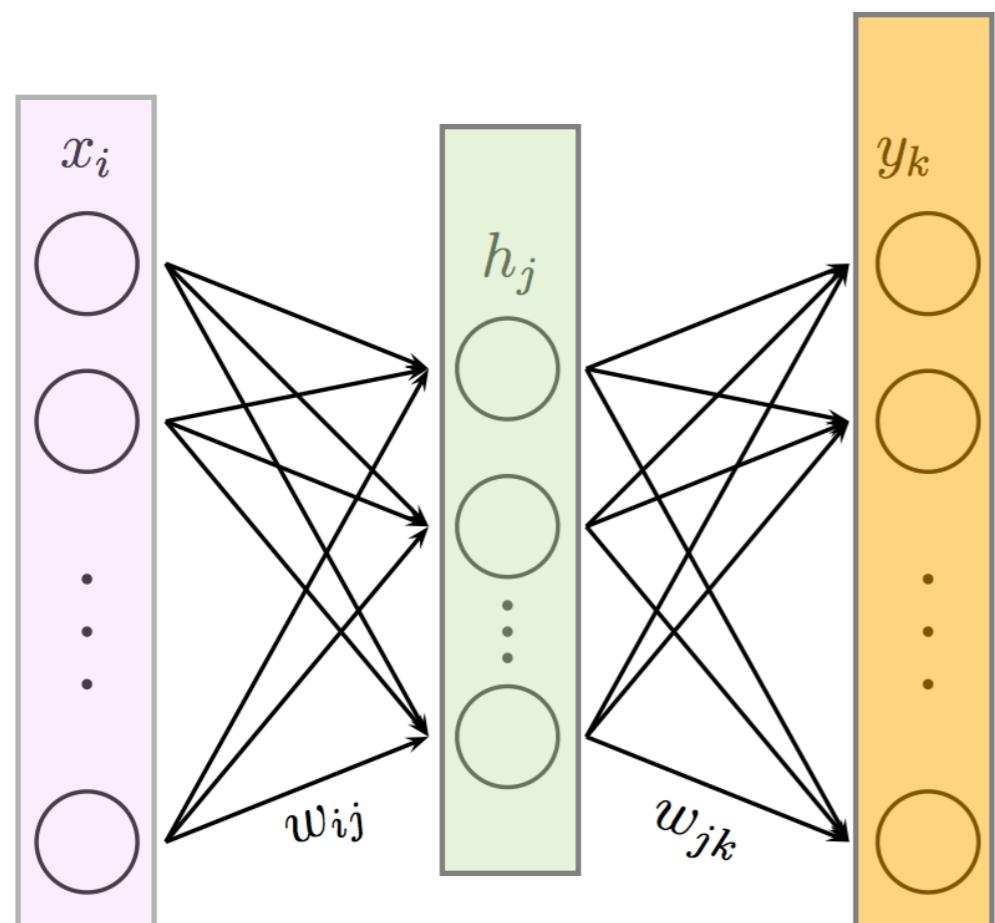
Figure 1. Schematic of a 3-layer feed-forward neural network.

Una red neuronal feed forward se organiza por capas conectadas las unas con las otras.

Están la capa inicial de **inputs** x

Las capas de neuronas (h) que se suelen llamar **capas ocultas**,

REDES NEURONALES



Una red neuronal feed forward se organiza por capas conectadas las unas con las otras.

Están la capa inicial de **inputs** x

Las capas de neuronas (h) que se suelen llamar **capas ocultas**,

La capa de **outputs** (y) con el resultado.

Figure 1. Schematic of a 3-layer feed-forward neural network.

REDES NEURONALES

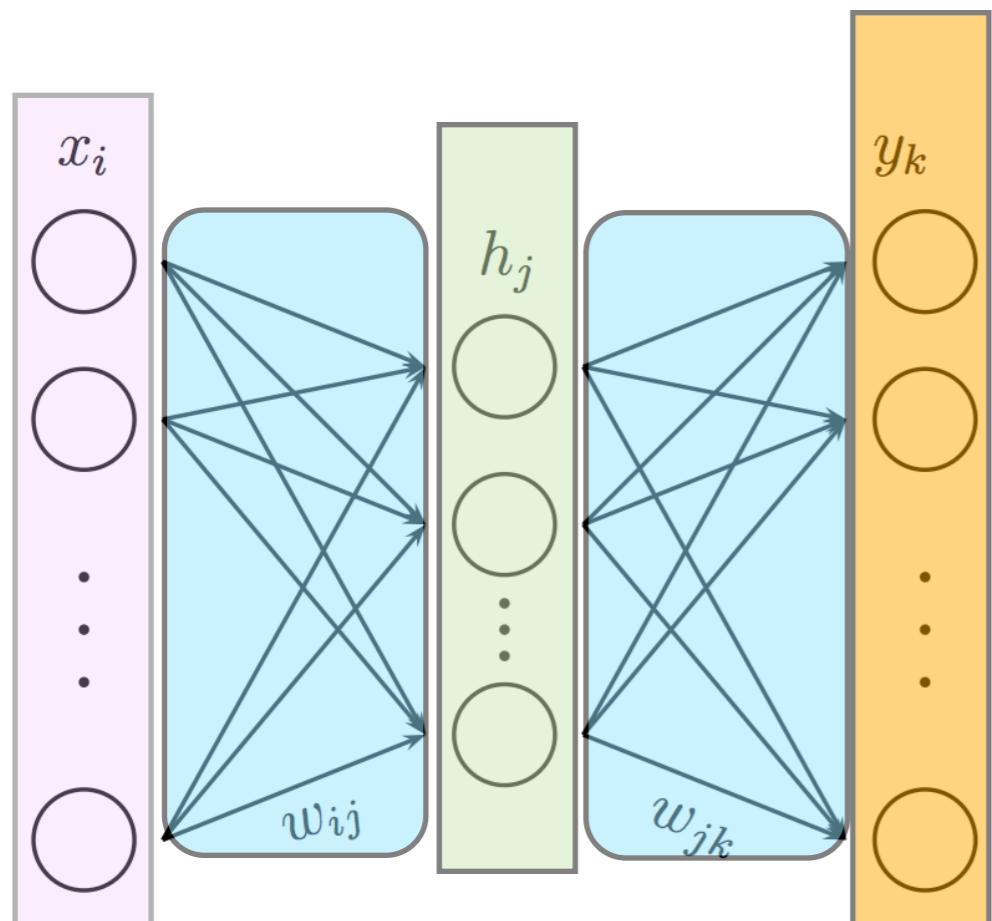


Figure 1. Schematic of a 3-layer feed-forward neural network.

Una red neuronal feed forward se organiza por capas conectadas las unas con las otras.

Están la capa inicial de **inputs** x

Las capas de neuronas (h) que se suelen llamar **capas ocultas**,

La capa de **outputs** (y) con el resultado.

Las conexiones que están representadas por pesos W

REDES NEURONALES

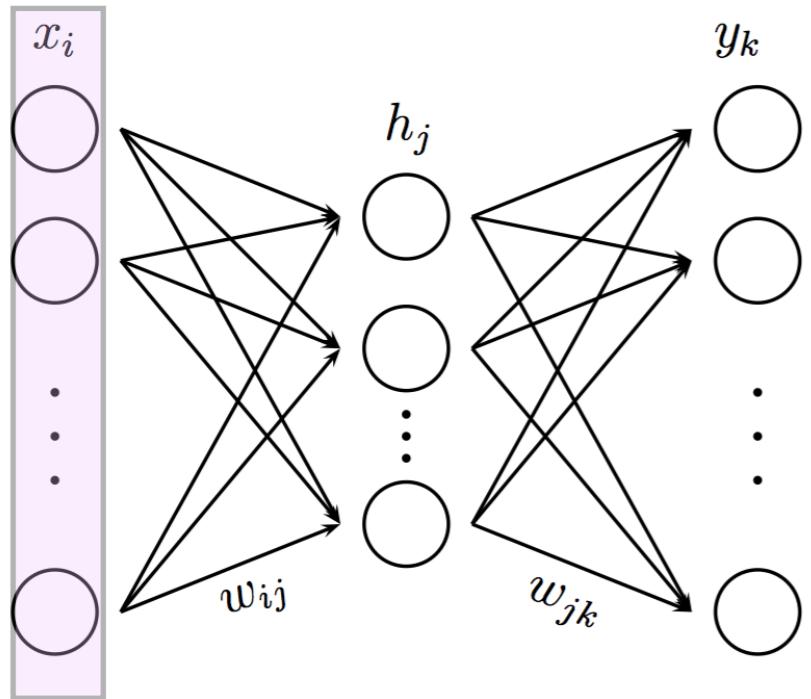


Figure 1. Schematic of a 3-layer feed-forward neural network.

En la capa de los inputs no pasa nada. Están ahí los valores de nuestras variables.

REDES NEURONALES

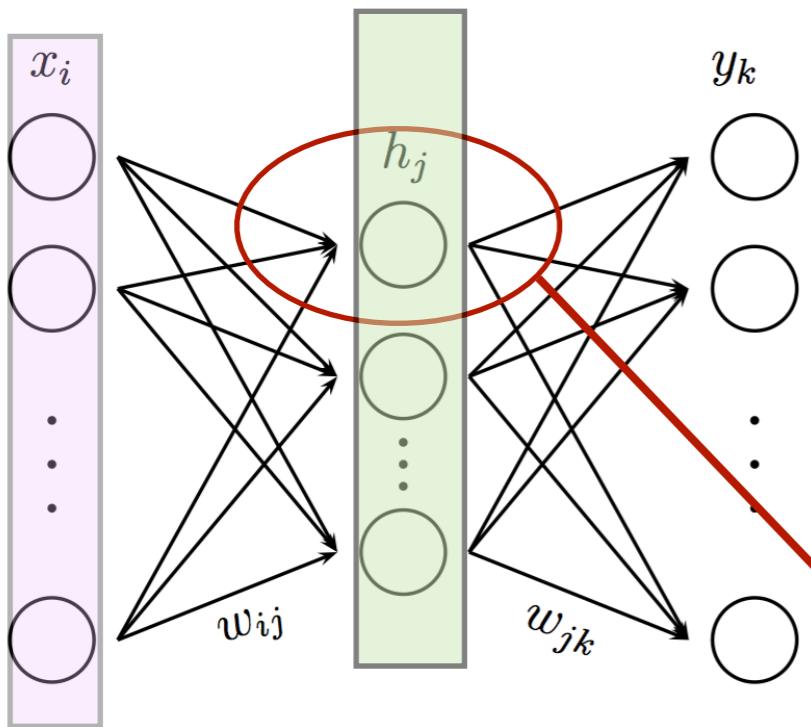
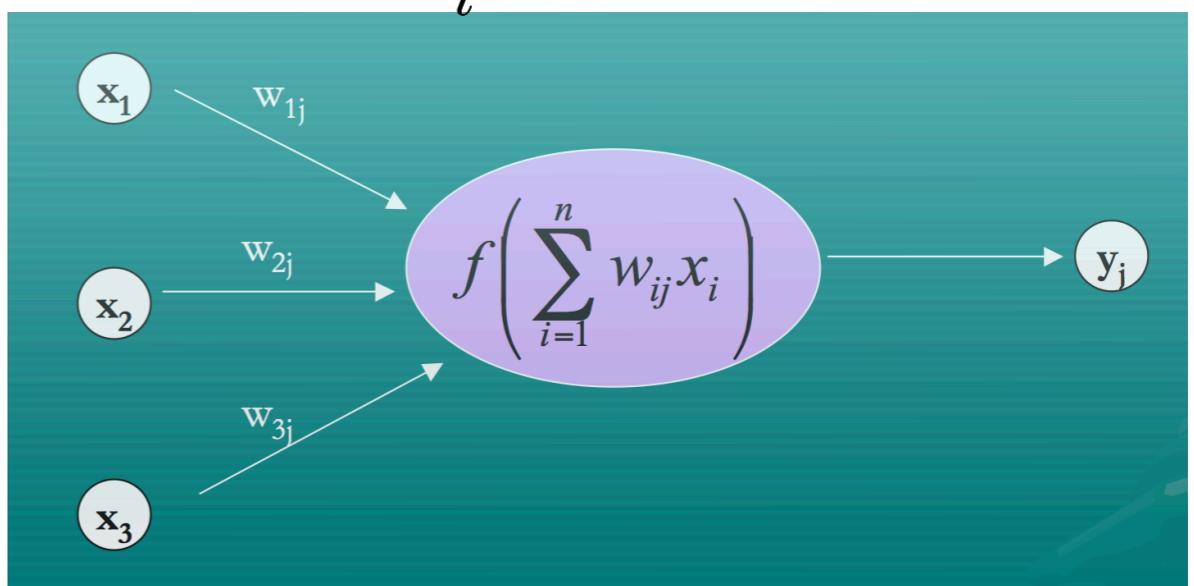


Figure 1. Schematic of a 3-layer feed-forward neural network.

En la capa de los inputs no pasa nada. Están ahí los valores de nuestras variables.

En la siguiente capa, en cada una de las neuronas se hace una operación matemática:

$$h_j = f\left(\sum_i w_{ij}x_i + \theta_j\right)$$



REDES NEURONALES

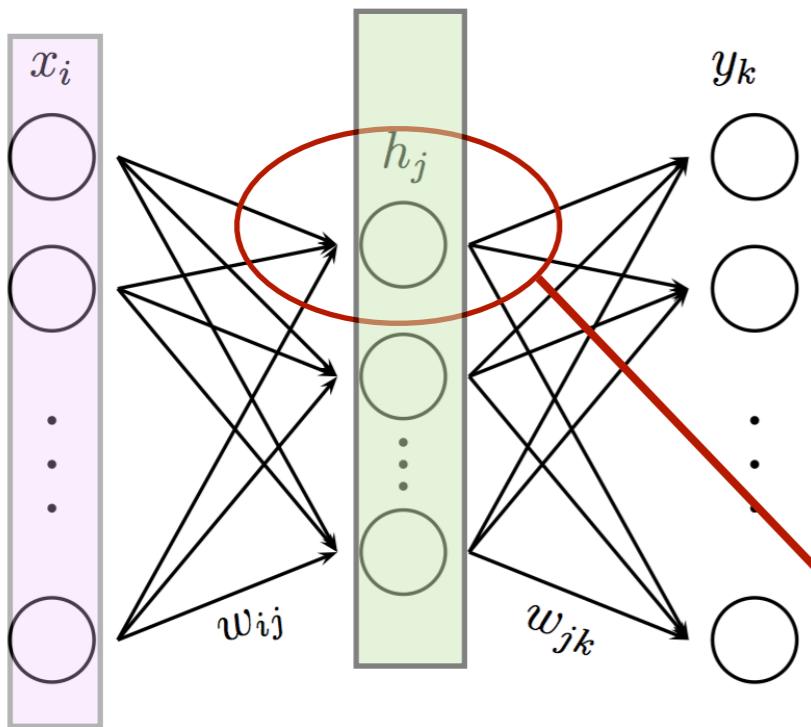


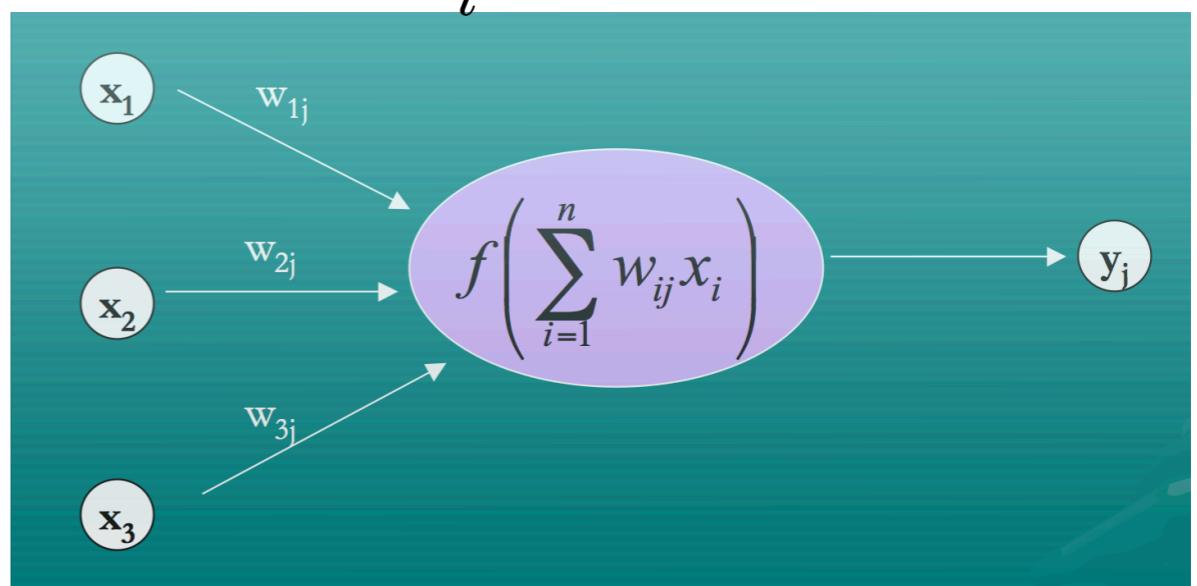
Figure 1. Schematic of a 3-layer feed-forward neural network.

Para cada neurona de la capa oculta hacemos este cálculo y los h_i serían los valores de entrada para la siguiente capa.

En la capa de los inputs no pasa nada. Están ahí los valores de nuestras variables.

En la siguiente capa, en cada una de las neuronas se hace una operación matemática:

$$h_j = f\left(\sum_i w_{ij}x_i + \theta_j\right)$$



REDES NEURONALES

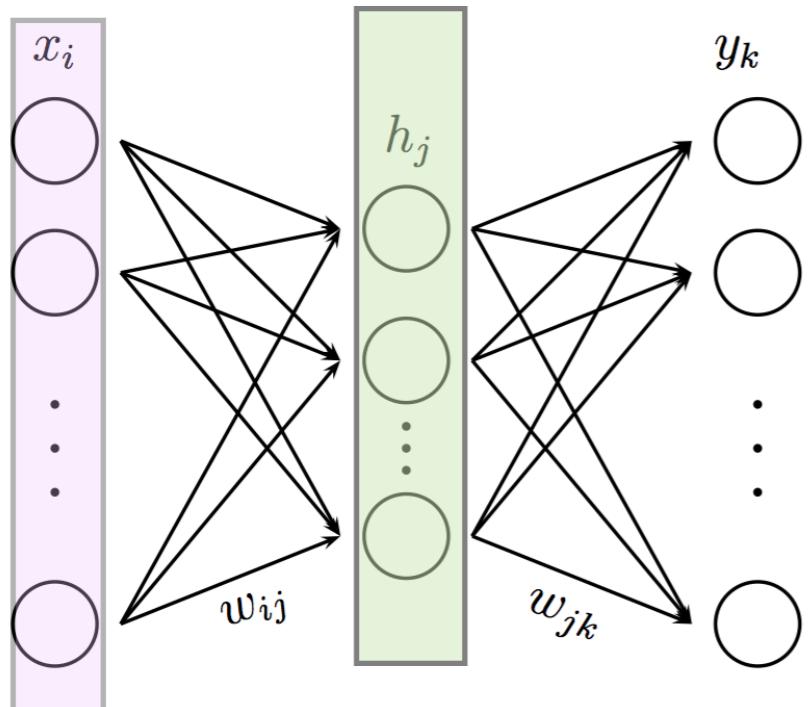


Figure 1. Schematic of a 3-layer feed-forward neural network.

Si hubiera una siguiente capa oculta vuelve a pasar lo mismo.

$$h_{j+1} = g(\sum_j w_{jj+1} h_j + \theta_{j+1})$$

La función que aplicamos no tiene porque ser la misma

REDES NEURONALES

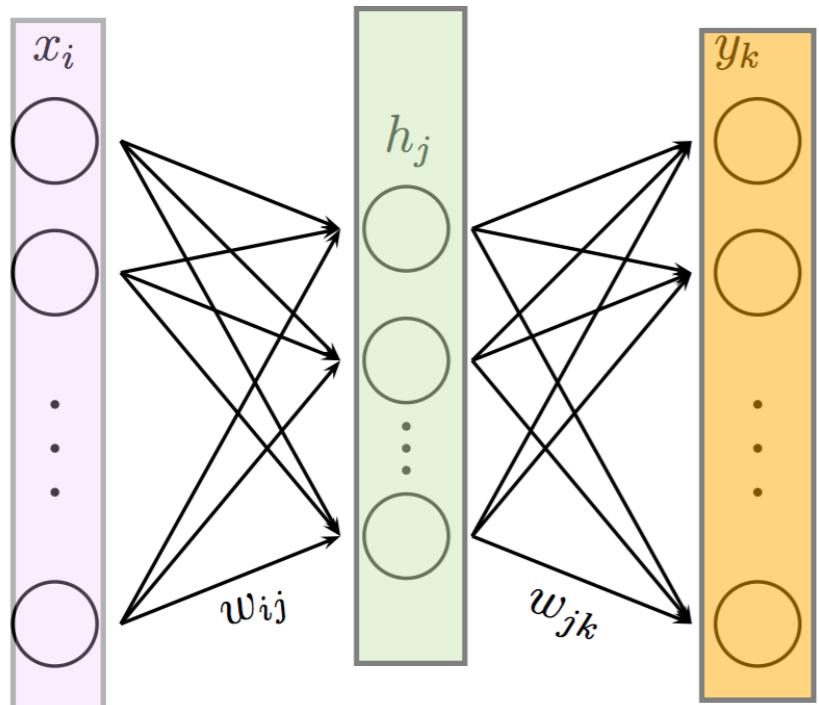


Figure 1. Schematic of a 3-layer feed-forward neural network.

Si hubiera una siguiente capa oculta vuelve a pasar lo mismo.

$$h_{j+1} = g(\sum_j w_{jj+1} h_j + \theta_{j+1})$$

La función que aplicamos no tiene porque ser la misma

Y así hasta la última capa

$$y_k = g\left(\left(\sum_j w_{jk} \dots f\left(\sum_i w_{ij} x_i \right) \right) \right)$$

REDES NEURONALES

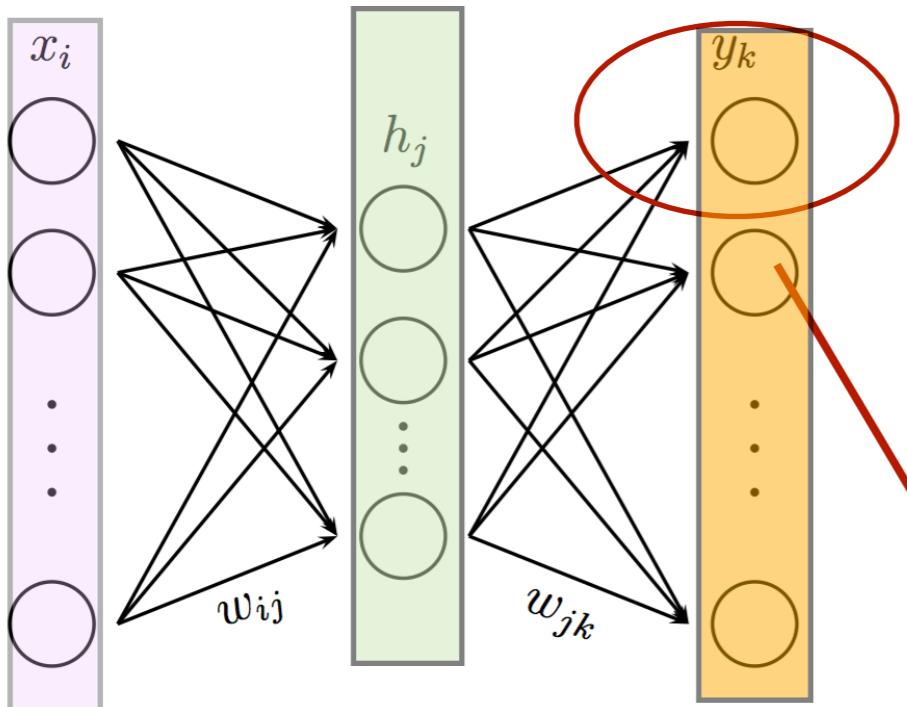


Figure 1. Schematic of a 3-layer feed-forward neural network.

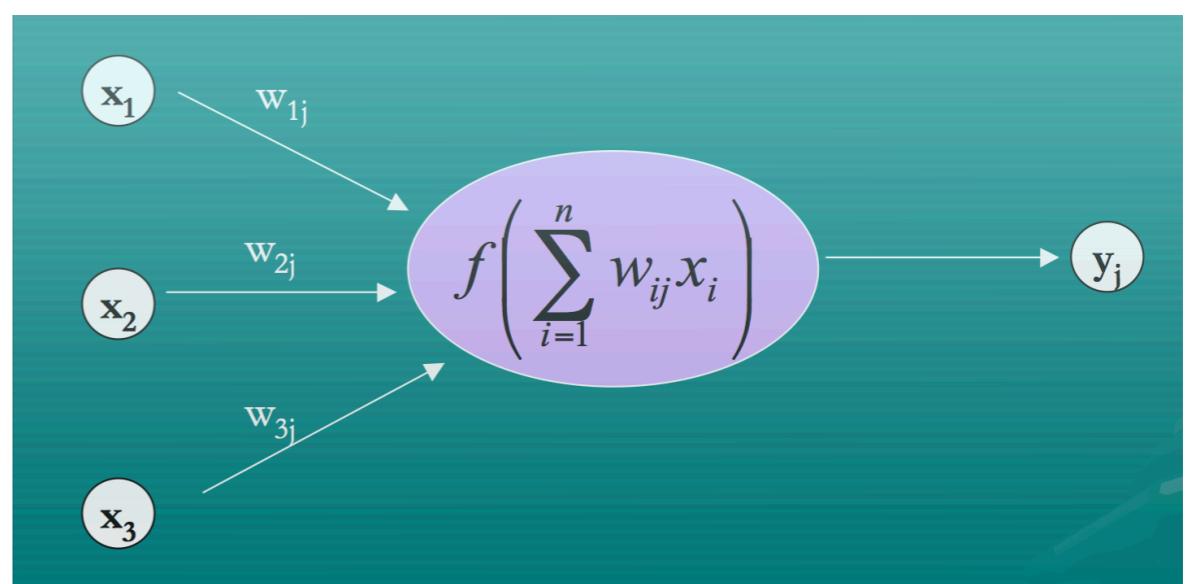
Y así hasta la última capa

$$y_k = g \left(\left(\sum_j w_{jk} \dots f \left(\sum_i w_{ij} x_i \right) \right) \right)$$

Si hubiera una siguiente capa oculta vuelve a pasar lo mismo.

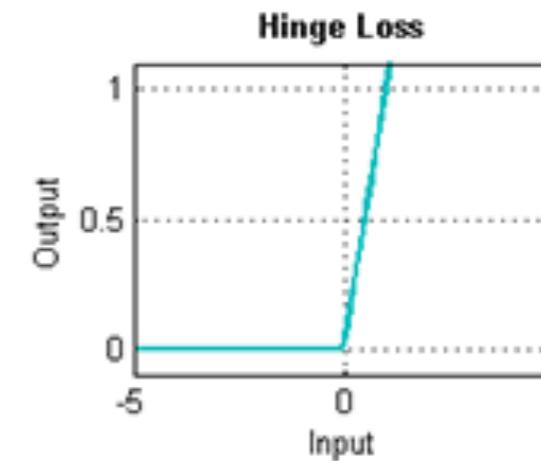
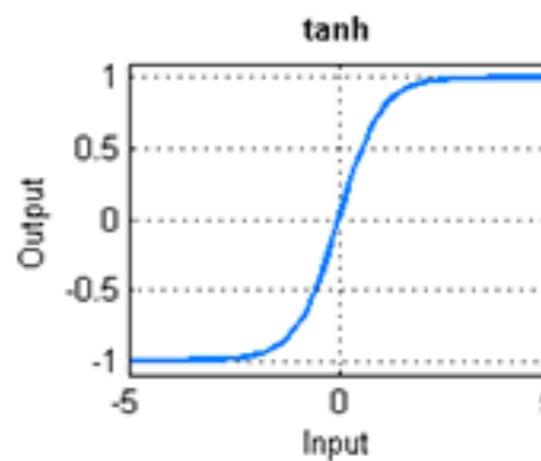
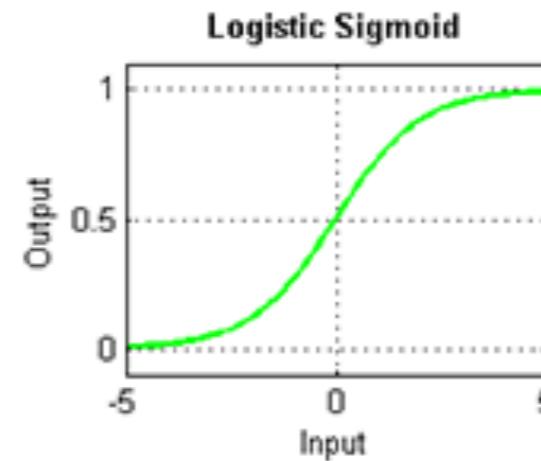
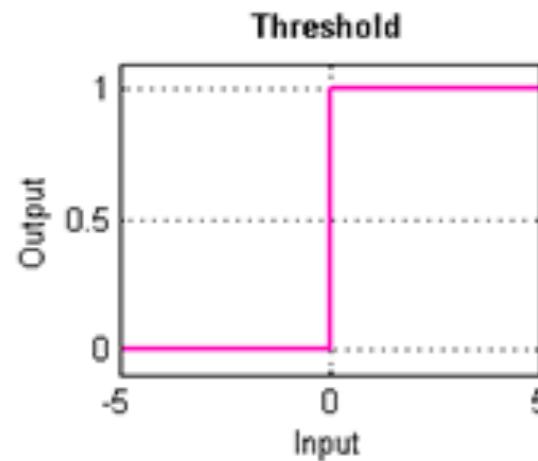
$$h_{j+1} = g \left(\sum_j w_{jj+1} h_j + \theta_{j+1} \right)$$

La función que aplicamos no tiene porque ser la misma



REDES NEURONALES

FUNCIONES DE ACTIVACIÓN



Función de Heaviside o función escalón.

$$H[n] = \begin{cases} 0, & n < 0, \\ 1, & n \geq 0, \end{cases}$$

Originalmente se creía que las neuronas biológicas se **activaban o no**. Por eso esta fue la función escogida en los primeros modelos.

Tangente hiperbólica o logistic sigmoid. Funciones derivables hacen que el problema sea más fácil de tratar matemáticamente. Siguen siendo funciones de activación, pero continuas, la más usada es tanh

$$f(x) = \frac{1}{1+e^{-x}} \quad f(x) = \frac{e^{2x}-1}{e^{2x}+1}$$

Recifier (ReLU). Se usa en problemas de clasificación, a veces se prefieren versiones más suaves.

Función lineal! Se suele usar en la capa final, así los outputs pueden tener cualquier valor.

$$f(x) = x$$

REDES NEURONALES

UNA NEURONA

Ejemplo:

Una sola capa oculta, con función de activación tanh y función de activación lineal para capa de los outputs:

En una red completa de 1 capa:

capa oculta con tanh:

$$h_j = \tanh\left(\sum w_{ij}x_i + \theta_j\right)$$

última capa con f. de activación lineal

$$y_k = \sum_j w_{jk}h_j + \theta_k$$

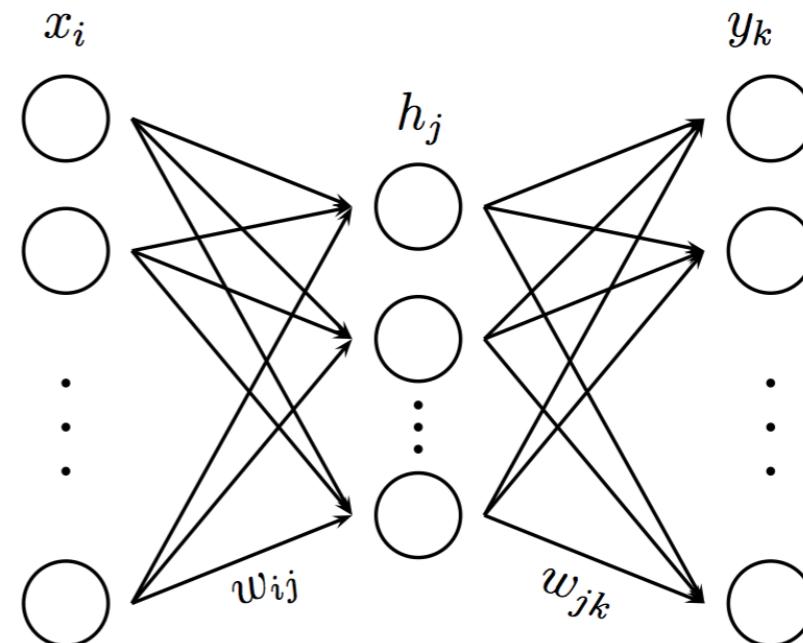


Figure 1. Schematic of a 3-layer feed-forward neural network.

Podemos poner los outputs en función de los inputs y de los pesos



$$y_k = \sum_j w_{jk} \tanh\left(\sum_i w_{ij}x_i + \theta_j\right) + \theta_k$$

REDES NEURONALES

**Podemos poner los outputs en función de
los inputs y de los pesos**

Forma general:

$$y_k = g \left(w_{vk} + \dots + f \left(\sum_i w_{ij} x_i + \theta_j \right) + w_v \right)$$

REDES NEURONALES

Podemos poner los outputs en función de los inputs y de los pesos

Forma general:

$$y_k = g \left(w_{vk} + \dots + f \left(\sum_i w_{ij} x_i + \theta_j \right) + w_v \right)$$

En el fondo estamos construyendo una familia de funciones con mucha flexibilidad.

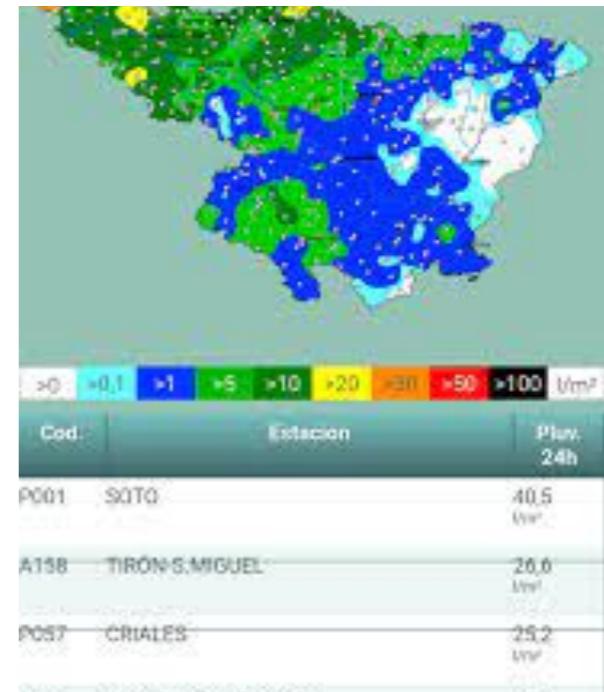
REDES NEURONALES

- ¿Y cómo puede aprender la máquina?

REDES NEURONALES

Imaginad que tenemos unos datos: metros(mm^3), y queremos saber si puedo predecir la lluvia de la siguiente semana pero NO tenemos un modelo fiable.

TABLA DE DATOS HISTÓRICOS DE PLUVIOMETRÍA MEDIA DE LOS PLUVIÓMETROS PENINSULARES (Fuente: AEMET) (Considerando todos los pluviómetros peninsulares la media total se eleva a 643,3 mm.)													
AÑOS	OCT	NOV	DIC	ENE	FEB	MAR	ABR	MAY	JUN	JUL	AGO	SEPT	TOTALES
MESES													
Media Hist.	76,6	80,0	78,8	63,5	52,7	47,1	64,6	60,9	31,4	20,2	22,9	44,6	643,3
1981/1982	55,7	72,0	27,8	21,9	36,3	45,3	82,7	39,4	25,6	14,5	19,0	43,5	484,0
1982/1983	33,0	2,0	131,6	59,1	54,0	37,3	32,6	45,6	29,9	24,1	25,0	44,8	520,0
1983/1984	73,4	109,8	56,4	4,8	48,2	25,6	82,2	39,8	23,5	22,2	64,9	7,6	557,9
1984/1985	23,2	132,2	81,1	56,7	43,2	79,0	45,5	120,4	40,0	5,7	18,8	29,0	674,8
1985/1986	56,4	167,9	33,0	88,8	82,9	40,0	63,7	67,2	23,1	17,8	5,7	10,8	657,3
1986/1987	14,0	85,5	86,1	64,7	97,4	36,0	64,7	29,8	17,3	12,8	12,2	75,7	566,2
1987/1988	72,8	46,8	37,0	96,8	73,1	26,6	62,6	24,0	22,5	47,7	19,8	35,9	565,6
1988/1989	124,5	74,3	109,3	111,5	39,8	18,5	102,2	79,9	24,0	101,4	6,5	14,1	806,0
1989/1990	68,7	66,5	10,6	25,8	62,7	43,9	89,7	64,6	13,1	15,9	25,9	52,3	549,7
1990/1991	40,9	164,4	161,7	54,5	13,6	23,5	79,2	39,2	28,5	14,0	21,3	32,9	674,7
1991/1992	107,3	64,2	43,3	45,7	73,2	95,0	50,7	30,4	17,1	15,2	10,9	60,9	614,0
1992/1993	73,1	63,8	29,2	24,4	34,5	43,2	48,4	60,1	94,8	16,2	39,0	43,3	570,0
1993/1994	115,8	13,6	66,2	12,2	38,2	37,8	73,0	87,8	45,1	10,3	23,8	53,2	586,7
1994/1995	139,3	60,3	36,0	69,0	64,7	9,1	46,3	78,2	12,8	9,2	12,0	54,1	589,0
1995/1996	86,6	64,1	48,9	53,1	53,9	27,9	23,5	27,9	23,5	17,4	26,4	38,5	515,2
1996/1997	22,4	97,6	168,8	183,3	60,4	49,3	43,0	85,3	10,2	22,0	32,7	60,5	846,5
1997/1998	38,2	106,1	192,2	130,5	10,1	3,0	48,7	84,9	59,1	42,2	47,8	53,5	816,3
1998/1999	55,0	172,1	133,5	54,5	39,3	26,6	88,1	91,9	24,5	10,3	19,0	67,1	781,9
1999/2000	32,8	33,0	53,0	47,1	30,6	61,3	53,0	52,9	13,7	25,7	19,8	85,0	517,9
2000/2001	126,6	46,3	56,5	24,8	12,1	32,3	132,3	70,8	23,8	16,2	12,3	29,0	583,1
2001/2002	92,4	119,6	151,1	127,0	50,6	128,2	27,7	55,2	10,7	27,1	16,3	48,5	862,5
2002/2003	88,5	48,0	34,0	48,7	26,7	66,8	71,5	65,9	36,1	14,7	35,8	59,6	596,3
2003/2004	75,6	111,2	107,0	91,7	82,8	50,1	63,3	46,0	14,3	12,8	27,2	49,0	741,0
2004/2005	150,1	100,4	70,8	43,4	63,3	77,4	66,5	79,1	18,1	17,4	30,3	28,8	745,6
2005/2006	100,8	30,1	55,1	14,6	40,1	34,9	40,3	39,3	30,1	7,8	14,5	33,1	430,8
2006/2007	104,0	67,9	51,7	58,2	53,9	71,3	44,9	30,3	37,2	19,1	19,5	61,6	619,6
2007/2008	115,3	104,6	47,4	38,9	76,0	56,0	92,9	76,0	37,5	8,8	30,7	40,1	722,2
2008/2009	54,5	39,7	28,2	50,2	42,2	41,3	111,0	125,1	41,3	13,3	13,8	43,5	604,1
2009/2010	105,8	64,6	73,1	82,3	49,1	47,4	56,0	25,8	27,7	12,8	16,5	45,8	606,6
2010/2011	53,0	61,2	183,5	110,8	79,8	53,2	55,2	71,8	12,1	18,7	37,4	867,5	
2011/2012	90,1	78,8	130,5	59,6	53,3	77,0	64,0	59,8	26,7	18,1	18,9	16,3	693,1
2012/2013	52,7	104,7	27,4	22,0	15,5	25,4	94,2	41,9	21,1	13,9	10,0	60,4	489,2
2013/2014	103,5	112,6	49,6	65,0	75,5	166,1	70,4	51,1	31,8	21,5	22,2	41,1	840,4
2014/2015	68,4	54,1	72,4	99,0	94,4	52,1	52,7	33,4	29,2	27,2	18,2	65,2	666,3
2015/2016	61,5	144,5	43,0	64,5	59,3	63,9	45,8	17,7	42,7	21,4	26,0	43,2	633,5
2016/2017	67,3	46,7	19,1	87,6	93,5	67,2	73,8	67,8	16,4	11,5	7,7	27,8	586,2
2017/2018	51,7	102,5	42,6	41,3	64,6	56,6	23,3	44,1	32,4	21,4	29,4	19,7	529,4
2018/2019	26,7	47,5	66,9	71,0	73,2	131,6	83,8	52,1	50,5	24,8	18,6	28,4	675,0
Precipitación superior a la media de la serie histórica 1981-2010													

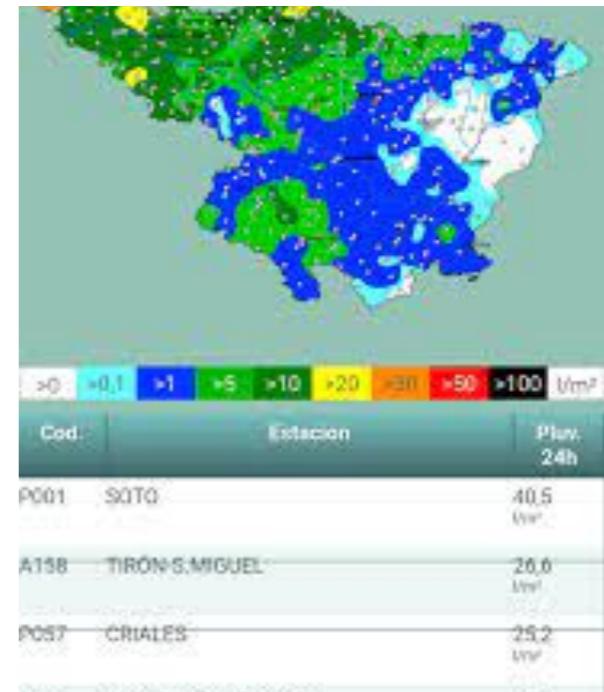


TENEMOS UNOS DATOS

REDES NEURONALES

Imaginad que tenemos unos datos: metros(mm^3), y queremos saber si puedo predecir la lluvia de la siguiente semana pero NO tenemos un modelo fiable.

TABLA DE DATOS HISTÓRICOS DE PLUVIOMETRÍA MEDIA DE LOS PLUVIÓMETROS PENINSULARES (Fuente: AEMET) (Considerando todos los pluviómetros peninsulares la media total se eleva a 643,3 mm.)													
AÑOS	OCT	NOV	DIC	ENE	FEB	MAR	ABR	MAY	JUN	JUL	AGO	SEPT	TOTALES
MESES													
Media Hist.	76,6	80,0	78,8	63,5	52,7	47,1	64,6	60,9	31,4	20,2	22,9	44,6	643,3
1981/1982	55,7	72,0	27,8	21,9	36,3	45,3	82,7	39,4	25,6	14,5	19,0	43,5	484,0
1982/1983	33,0	2,0	131,6	59,1	54,0	37,3	32,6	45,6	29,9	24,1	25,0	44,8	520,0
1983/1984	73,4	109,8	56,4	4,8	48,2	25,6	82,2	39,8	23,5	22,2	64,9	7,6	557,9
1984/1985	23,2	132,2	81,1	56,7	43,2	79,0	45,5	120,4	40,0	5,7	18,8	29,0	674,8
1985/1986	56,4	167,9	33,0	88,8	82,9	40,0	63,7	67,2	23,1	17,8	5,7	10,8	657,3
1986/1987	14,0	85,5	86,1	64,7	97,4	36,0	64,7	29,8	17,3	12,8	12,2	75,7	566,2
1987/1988	72,8	46,8	37,0	96,8	73,1	26,6	62,6	24,0	22,5	47,7	19,8	35,9	565,6
1988/1989	124,5	74,3	109,3	111,5	39,8	18,5	102,2	79,9	24,0	101,4	6,5	14,1	806,0
1989/1990	68,7	66,5	10,6	25,8	62,7	43,9	89,7	64,6	13,1	15,9	25,9	52,3	549,7
1990/1991	40,9	164,4	161,7	54,5	13,6	23,5	79,2	39,2	28,5	14,0	21,3	32,9	674,7
1991/1992	107,3	64,2	43,3	45,7	73,2	95,0	50,7	30,4	17,1	15,2	10,9	60,9	614,0
1992/1993	73,1	63,8	29,2	24,4	34,5	43,2	48,4	60,1	94,8	16,2	39,0	43,3	570,0
1993/1994	115,8	13,6	66,2	12,2	38,2	37,8	73,0	87,8	45,1	10,3	23,8	53,2	586,7
1994/1995	139,3	60,3	36,0	69,0	64,7	9,1	46,3	78,2	12,8	9,2	12,0	54,1	589,0
1995/1996	86,6	64,1	48,9	53,1	53,9	27,9	23,5	27,9	23,5	17,4	26,4	38,5	515,2
1996/1997	22,4	97,6	168,8	183,3	60,4	49,3	43,0	85,3	10,2	22,0	32,7	60,5	846,5
1997/1998	38,2	106,1	192,2	130,5	10,1	3,0	48,7	84,9	59,1	42,2	47,8	53,5	816,3
1998/1999	55,0	172,1	133,5	54,5	39,3	26,6	88,1	91,9	24,5	10,3	19,0	67,1	781,9
1999/2000	32,8	33,0	53,0	47,1	30,6	61,3	53,0	52,9	13,7	25,7	19,8	85,0	517,9
2000/2001	126,6	46,3	56,5	24,8	12,1	32,3	132,3	70,8	23,8	16,2	12,3	29,0	583,1
2001/2002	92,4	119,6	151,1	127,0	50,6	128,2	27,7	55,2	10,7	27,1	16,3	48,5	862,5
2002/2003	88,5	48,0	34,0	48,7	26,7	66,8	71,5	65,9	36,1	14,7	35,8	59,6	596,3
2003/2004	75,6	111,2	107,0	91,7	82,8	50,1	63,3	46,0	14,3	12,8	27,2	49,0	741,0
2004/2005	150,1	100,4	70,8	43,4	63,3	77,4	66,5	79,1	18,1	17,4	30,3	28,8	745,6
2005/2006	100,8	30,1	55,1	14,6	40,1	34,9	40,3	39,3	30,1	7,8	14,5	33,1	430,8
2006/2007	104,0	67,9	51,7	58,2	53,9	71,3	44,9	30,3	37,2	19,1	19,5	61,6	619,6
2007/2008	115,3	104,6	47,4	38,9	76,0	56,0	92,9	76,0	37,5	8,8	30,7	40,1	722,2
2008/2009	54,5	39,7	28,2	50,2	42,2	41,3	111,0	125,1	41,3	13,3	13,8	43,5	604,1
2009/2010	105,8	64,6	73,1	82,3	49,1	47,4	56,0	25,8	27,7	12,8	16,5	45,8	606,6
2010/2011	53,0	61,2	183,5	110,8	79,8	53,2	55,2	71,8	12,1	18,7	37,4	867,5	
2011/2012	90,1	78,8	130,5	59,6	53,3	77,0	64,0	59,8	26,7	18,1	18,9	16,3	693,1
2012/2013	52,7	104,7	27,4	22,0	15,5	25,4	94,2	41,9	21,1	13,9	10,0	60,4	489,2
2013/2014	103,5	112,6	49,6	65,0	75,5	166,1	70,4	51,1	31,8	21,5	22,2	41,1	840,4
2014/2015	68,4	54,1	72,4	99,0	94,4	52,1	52,7	33,4	29,2	27,2	18,2	65,2	666,3
2015/2016	61,5	144,5	43,0	64,5	59,3	63,9	45,8	17,7	42,7	21,4	26,0	43,2	633,5
2016/2017	67,3	46,7	19,1	87,6	93,5	67,2	73,8	67,8	16,4	11,5	7,7	27,8	586,2
2017/2018	51,7	102,5	42,6	41,3	64,6	56,6	23,3	44,1	32,4	21,4	29,4	19,7	529,4
2018/2019	26,7	47,5	66,9	71,0	73,2	131,6	83,8	52,1	50,5	24,8	18,6	28,4	675,0
Precipitación superior a la media de la serie histórica 1981-2010													

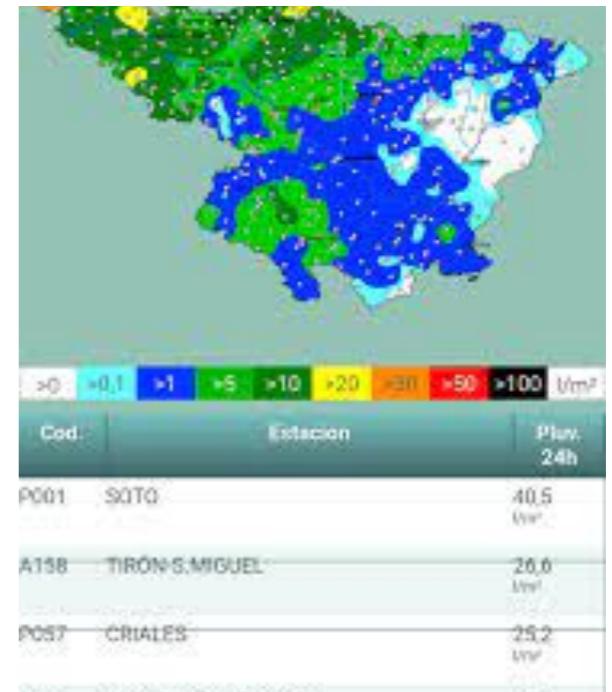


TENEMOS UNOS DATOS

REDES NEURONALES

Imaginad que tenemos unos datos: metros(mm^3), y queremos saber si puedo predecir la lluvia de la siguiente semana pero NO tenemos un modelo fiable.

TABLA DE DATOS HISTÓRICOS DE PLUVIOMETRÍA MEDIA DE LOS PLUVIÓMETROS PENINSULARES (Fuente: AEMET) (Considerando todos los pluviómetros peninsulares la media total se eleva a 643,3 mm.)													
AÑOS	OCT	NOV	DIC	ENE	FEB	MAR	ABR	MAY	JUN	JUL	AGO	SEPT	TOTALES
MESES													
Media Hist.	76,6	80,0	78,8	63,5	52,7	47,1	64,6	60,9	31,4	20,2	22,9	44,6	643,3
1981/1982	55,7	72,0	27,8	21,9	36,3	45,3	82,7	39,4	25,6	14,5	19,0	43,5	484,0
1982/1983	33,0	2,0	131,6	59,1	54,0	37,3	32,6	45,6	29,9	24,1	25,0	44,8	520,0
1983/1984	73,4	109,8	56,4	4,8	48,2	25,6	82,2	39,8	23,5	22,2	64,9	7,6	557,9
1984/1985	23,2	132,2	81,1	56,7	43,2	79,0	45,5	120,4	40,0	5,7	18,8	29,0	674,8
1985/1986	56,4	167,9	33,0	88,8	82,9	40,0	63,7	67,2	23,1	17,8	5,7	10,8	657,3
1986/1987	14,0	85,5	86,1	64,7	97,4	36,0	64,7	29,8	17,3	12,8	12,2	75,7	566,2
1987/1988	72,8	46,8	37,0	96,8	73,1	26,6	62,6	24,0	22,5	47,7	19,8	35,9	565,6
1988/1989	124,5	74,3	109,3	111,5	39,8	18,5	102,2	79,9	24,0	101,4	6,5	14,1	806,0
1989/1990	68,7	66,5	10,6	25,8	62,7	43,9	89,7	64,6	13,1	15,9	25,9	52,3	549,7
1990/1991	40,9	164,4	161,7	54,5	13,6	23,5	79,2	39,2	28,5	14,0	21,3	32,9	674,7
1991/1992	107,3	64,2	43,3	45,7	73,2	95,0	50,7	30,4	17,1	15,2	10,9	60,9	614,0
1992/1993	73,1	63,8	29,2	24,4	34,5	43,2	48,4	60,1	94,8	16,2	39,0	43,3	570,0
1993/1994	115,8	13,6	66,2	12,2	38,2	37,8	73,0	87,8	45,1	10,3	23,8	53,2	586,7
1994/1995	139,3	60,3	36,0	69,0	64,7	9,1	46,3	78,2	12,8	9,2	12,0	54,1	589,0
1995/1996	86,6	64,1	48,9	53,1	53,9	27,9	23,5	27,9	23,5	17,4	26,4	38,5	515,2
1996/1997	22,4	97,6	168,8	183,3	60,4	49,3	43,0	85,3	10,2	22,0	32,7	60,5	846,5
1997/1998	38,2	106,1	192,2	130,5	10,1	3,0	48,7	84,9	59,1	42,2	47,8	53,5	816,3
1998/1999	55,0	172,1	133,5	54,5	39,3	26,6	88,1	91,9	24,5	10,3	19,0	67,1	781,9
1999/2000	32,8	33,0	53,0	47,1	30,6	61,3	53,0	52,9	13,7	25,7	19,8	85,0	517,9
2000/2001	126,6	46,3	56,5	24,8	12,1	32,3	132,3	70,8	23,8	16,2	12,3	29,0	583,1
2001/2002	92,4	119,6	151,1	127,0	50,6	128,2	27,7	55,2	10,7	27,1	16,3	48,5	862,5
2002/2003	88,5	48,0	34,0	48,7	26,7	66,8	71,5	65,9	36,1	14,7	35,8	59,6	596,3
2003/2004	75,6	111,2	107,0	91,7	82,8	50,1	63,3	46,0	24,3	12,8	27,2	49,0	741,0
2004/2005	150,1	100,4	70,8	43,4	63,3	77,4	66,5	79,1	18,1	17,4	30,3	28,8	745,6
2005/2006	100,8	30,1	55,1	14,6	40,1	34,9	40,3	39,3	30,1	7,8	14,5	33,1	430,8
2006/2007	104,0	67,9	51,7	58,2	53,9	71,3	44,9	30,3	37,2	19,1	19,5	61,6	619,6
2007/2008	115,3	104,6	47,4	38,9	76,0	56,0	92,9	76,0	37,5	8,8	30,7	40,1	722,2
2008/2009	54,5	39,7	28,2	50,2	42,2	41,3	111,0	125,1	41,3	13,3	13,8	43,5	604,1
2009/2010	105,8	64,6	73,1	82,3	49,1	47,4	56,0	25,8	27,7	12,8	16,5	45,8	606,6
2010/2011	53,0	61,2	183,5	110,8	79,8	53,2	55,2	71,8	12,1	18,7	37,4	867,5	
2011/2012	90,1	78,8	130,5	59,6	53,3	77,0	64,0	59,8	26,7	18,1	18,9	16,3	693,1
2012/2013	52,7	104,7	27,4	22,0	15,5	25,4	94,2	41,9	21,1	13,9	10,0	60,4	489,2
2013/2014	103,5	112,6	49,6	95,0	75,5	166,1	70,4	51,1	31,8	21,5	22,2	41,1	840,4
2014/2015	68,4	54,1	72,4	99,0	94,4	52,1	52,7	33,4	29,2	27,2	18,2	65,2	666,3
2015/2016	61,5	144,5	43,0	64,5	59,3	63,9	45,8	17,7	42,7	21,4	26,0	43,2	633,5
2016/2017	67,3	46,7	19,1	87,6	93,5	67,2	73,8	67,8	16,4	11,5	7,7	27,8	586,2
2017/2018	51,7	102,5	42,6	41,3	64,6	56,6	23,3	44,1	32,4	21,4	29,4	19,7	529,4
2018/2019	26,7	47,5	66,9	71,0	73,2	131,6	83,8	52,1	50,5	24,8	18,6	28,4	675,0
Precipitación superior a la media de la serie histórica 1981-2010													

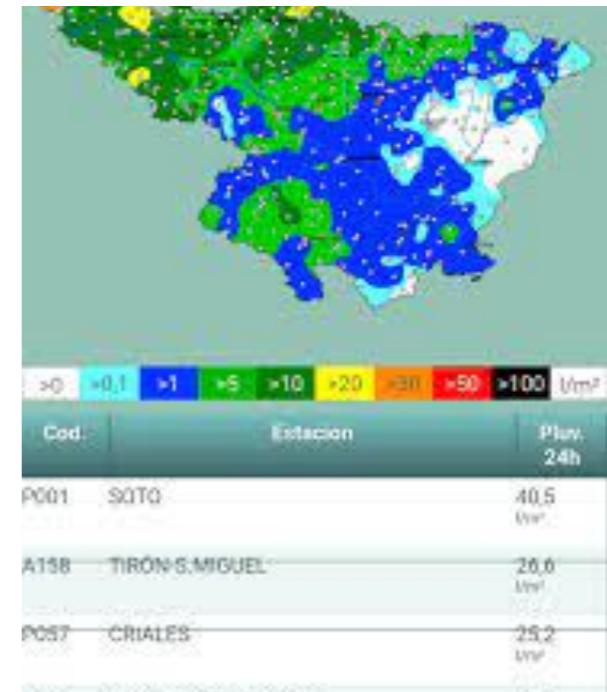


TENEMOS UNOS DATOS

REDES NEURONALES

Imaginad que tenemos unos datos: metros(mm^3), y queremos saber si puedo predecir la lluvia de la siguiente semana pero NO tenemos un modelo fiable.

TABLA DE DATOS HISTÓRICOS DE PLUVIOMETRÍA MEDIA DE LOS PLUVIÓMETROS PENINSULARES (Fuente: AEMET) (Considerando todos los pluviómetros peninsulares la media total se eleva a 643,3 mm.)														
AÑOS	MESES	OCT	NOV	DIC	ENE	FEB	MAR	ABR	MAY	JUN	JUL	AGO	SEPT	TOTALES
Media Hist.		76,6	80,0	78,8	63,5	52,7	47,1	64,6	60,9	31,4	20,2	22,9	44,6	643,3
1981/1982		55,7	72,0	27,8	21,9	36,3	45,3	82,7	39,4	25,8	14,5	19,0	43,5	484,0
1982/1983		33,0	2,0	131,6	59,1	54,0	37,3	32,6	46,6	29,9	24,1	25,0	44,8	520,0
1983/1984		73,4	109,8	56,4	4,8	48,2	25,6	82,2	39,8	13,5	22,2	64,9	7,6	557,9
1984/1985		23,2	132,2	81,1	56,7	43,2	79,0	45,5	120,4	40,0	5,7	18,8	29,0	674,8
1985/1986		56,4	167,9	33,0	88,8	82,9	40,0	63,7	67,2	23,1	17,8	5,7	10,8	657,3
1986/1987		14,0	85,5	86,1	64,7	97,4	36,0	64,7	29,8	17,3	12,8	12,2	75,7	566,2
1987/1988		72,8	46,8	37,0	96,8	73,1	26,6	62,6	24,0	22,5	47,7	19,8	35,9	565,6
1988/1989		124,5	74,3	109,3	111,5	39,8	16,5	102,2	79,9	24,0	101,4	6,5	14,1	806,0
1989/1990		68,7	66,5	10,6	25,8	62,7	43,9	89,7	64,6	13,1	15,9	25,9	52,3	549,7
1990/1991		40,9	164,4	161,7	54,5	13,6	23,5	79,2	39,2	28,5	14,0	21,3	32,9	674,7
1991/1992		107,3	64,2	43,3	45,7	73,2	95,0	50,7	30,4	17,1	15,2	10,9	60,9	614,0
1992/1993		73,1	63,8	29,2	24,4	34,5	43,2	48,4	60,1	94,8	16,2	39,0	43,3	570,0
1993/1994		115,8	13,6	66,2	12,2	38,2	37,8	73,0	87,8	45,1	10,3	23,8	53,2	586,7
1994/1995		139,3	60,3	36,0	69,0	64,7	9,1	46,3	78,2	12,8	9,2	12,0	54,1	589,0
1995/1996		86,6	64,1	48,9	53,1	53,9	27,9	23,5	27,2	37,7	17,4	26,4	38,5	515,2
1996/1997		22,4	97,6	168,8	183,3	60,4	49,3	43,0	85,3	10,2	22,0	32,7	60,5	846,5
1997/1998		38,2	106,1	192,2	130,5	10,1	3,0	48,7	84,9	59,1	42,2	47,8	53,5	816,3
1998/1999		55,0	172,1	133,5	54,5	39,3	26,6	88,1	91,9	24,5	10,3	19,0	67,1	781,9
1999/2000		32,8	33,0	53,0	47,1	30,6	61,3	53,0	52,9	13,7	25,7	19,8	85,0	517,9
2000/2001		126,6	46,3	56,5	24,8	12,1	32,3	132,3	70,8	23,8	16,2	12,3	29,0	583,1
2001/2002		92,4	119,6	151,1	137,0	50,6	128,2	27,7	55,2	10,7	27,1	16,3	49,5	862,5
2002/2003		88,5	48,0	34,0	48,7	26,7	66,8	71,5	65,9	36,1	14,7	35,8	59,6	596,3
2003/2004		75,6	111,2	107,0	91,7	82,8	50,1	63,3	46,0	14,3	12,8	27,2	49,0	741,0
2004/2005		150,1	100,4	70,8	43,4	63,3	77,4	66,5	79,1	18,1	17,4	30,3	28,8	745,6
2005/2006		100,8	30,1	55,1	14,6	40,1	34,9	40,3	39,3	30,1	7,8	14,5	33,1	430,8
2006/2007		104,0	67,9	51,7	58,2	53,9	71,3	44,9	30,3	37,2	19,1	19,5	61,6	619,6
2007/2008		115,3	104,6	47,4	38,9	76,0	56,0	92,9	76,0	37,5	8,8	30,7	40,1	722,2
2008/2009		54,5	39,7	28,2	50,2	42,2	41,3	111,0	125,1	41,3	13,3	13,8	43,5	604,1
2009/2010		105,8	64,6	73,1	82,3	49,1	47,4	56,0	25,8	27,7	12,8	16,5	45,8	606,6
2010/2011		53,0	61,2	183,5	110,8	79,8	53,2	55,2	71,8	12,1	18,7	37,4	867,5	
2011/2012		90,1	78,8	130,5	59,6	53,3	77,0	64,0	59,8	26,7	18,1	18,9	16,3	693,1
2012/2013		52,7	104,7	27,4	22,0	15,5	25,4	94,2	41,9	21,1	13,9	10,0	60,4	489,2
2013/2014		103,5	112,6	49,6	65,0	75,5	166,1	70,4	51,1	31,8	21,5	22,2	41,1	840,4
2014/2015		68,4	54,1	72,4	99,0	94,4	52,1	52,7	33,4	29,2	27,2	18,2	65,2	666,3
2015/2016		61,5	144,5	43,0	64,5	59,3	63,9	45,8	17,7	42,7	21,4	26,0	43,2	633,5
2016/2017		67,3	46,7	19,1	87,6	93,5	67,2	73,8	67,8	16,4	11,5	7,7	27,8	586,2
2017/2018		51,7	102,5	42,6	41,3	64,6	56,6	23,3	44,1	32,4	21,4	29,4	19,7	529,4
2018/2019		26,7	47,5	66,9	71,0	73,2	131,6	83,8	52,1	50,8	24,8	18,6	28,4	675,0
		92,3	-	-	-	-	-	-	-	-	-	-	-	91,3



TENEMOS UNOS DATOS

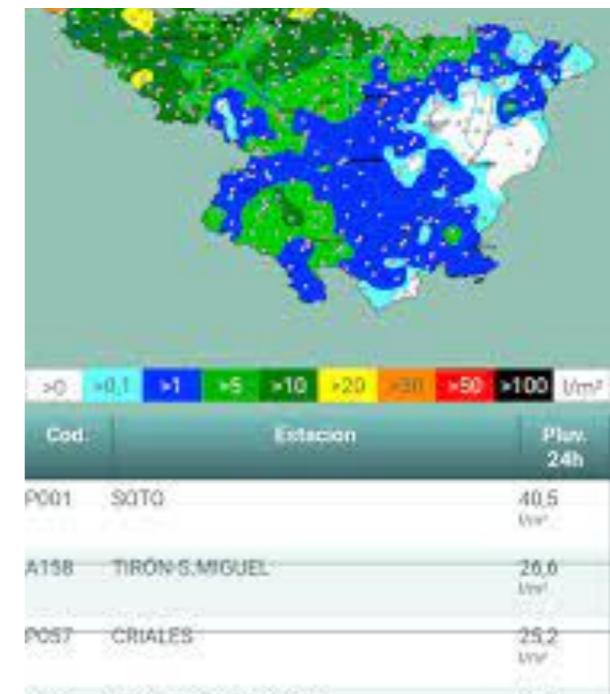
Y hacemos un ajuste → **entrenamiento**

REDES NEURONALES

Imaginad que tenemos unos datos: metros(mm^3), y queremos saber si puedo predecir la lluvia de la siguiente semana pero NO tenemos un modelo fiable.

TABLA DE DATOS HISTÓRICOS DE PLUVIOMETRÍA MEDIA DE LOS PLUVIÓMETROS PENINSULARES (Fuente: AEMET) (Considerando todos los pluviómetros peninsulares la media total se eleva a 643,3 mm.)													
AÑOS	OCT	NOV	DIC	ENE	FEB	MAR	ABR	MAY	JUN	JUL	AGO	SEPT	TOTALES
MESES													
Media Hist.	76,6	80,0	78,8	63,5	52,7	47,1	64,6	60,9	31,4	20,2	22,9	44,6	643,3
1981/1982	55,7	72,0	27,8	21,9	36,3	45,3	82,7	39,4	25,8	14,5	19,0	43,5	484,0
1982/1983	33,0	2,0	131,6	59,1	54,0	37,3	32,6	46,6	29,9	24,1	25,0	44,8	520,0
1983/1984	73,4	109,8	56,4	4,8	48,2	25,6	82,2	39,8	23,5	22,2	64,9	7,6	557,9
1984/1985	23,2	132,2	81,1	56,7	43,2	79,0	45,5	120,4	40,0	5,7	18,8	29,0	674,8
1985/1986	56,4	167,9	33,0	88,8	82,9	40,0	63,7	67,2	23,1	17,8	5,7	10,8	657,3
1986/1987	14,0	85,5	86,1	64,7	97,4	36,0	64,7	29,8	17,3	12,8	12,2	75,7	566,2
1987/1988	72,8	46,8	37,0	96,8	73,1	26,6	62,6	24,0	22,5	47,7	19,8	35,9	565,6
1988/1989	124,5	74,3	109,3	111,5	39,8	16,5	102,2	79,9	24,0	101,4	6,5	14,1	806,0
1989/1990	68,7	66,5	10,6	25,8	62,7	43,9	89,7	64,6	13,1	15,9	25,9	52,3	549,7
1990/1991	40,9	164,4	161,7	54,5	13,6	23,5	79,2	39,2	28,5	14,0	21,3	32,9	674,7
1991/1992	107,3	64,2	43,3	45,7	73,2	95,0	50,7	30,4	17,1	15,2	10,9	60,9	614,0
1992/1993	73,1	63,8	29,2	24,4	34,5	43,2	48,4	60,1	94,8	16,2	39,0	43,3	570,0
1993/1994	115,8	13,6	66,2	12,2	38,2	37,8	73,0	87,8	45,1	10,3	23,8	53,2	586,7
1994/1995	139,3	60,3	36,0	69,0	64,7	9,1	46,3	78,2	12,8	9,2	12,0	54,1	589,0
1995/1996	86,6	64,1	48,9	53,1	53,9	27,9	23,5	37,2	37,7	17,4	26,4	38,5	515,2
1996/1997	22,4	97,6	168,8	183,3	60,4	49,3	43,0	85,3	10,2	22,0	32,7	60,5	846,5
1997/1998	38,2	106,1	192,2	130,5	10,1	3,0	48,7	84,9	59,1	42,2	47,8	53,5	816,3
1998/1999	55,0	172,1	133,5	54,5	39,3	26,6	88,1	91,9	24,5	10,3	19,0	67,1	781,9
1999/2000	32,8	33,0	53,0	47,1	30,6	61,3	53,0	52,9	13,7	25,7	19,8	85,0	517,9
2000/2001	126,6	46,3	56,5	24,8	12,1	32,3	132,3	70,8	23,8	16,2	12,3	29,0	583,1
2001/2002	92,4	119,6	191,1	197,0	50,6	128,2	27,7	55,2	10,7	27,1	16,3	48,5	862,5
2002/2003	88,5	48,0	34,0	48,7	26,7	66,8	71,5	65,9	36,1	14,7	35,8	59,6	596,3
2003/2004	75,6	111,2	107,0	91,7	82,8	50,1	63,3	46,0	14,3	12,8	27,2	49,0	741,0
2004/2005	150,1	100,4	70,8	43,4	63,3	77,4	66,5	79,1	18,1	17,4	30,3	28,8	745,6
2005/2006	100,8	30,1	55,1	14,6	40,1	34,9	40,3	39,3	30,1	7,8	14,5	33,1	430,8
2006/2007	104,0	67,9	51,7	58,2	53,9	71,3	44,9	30,3	37,2	19,1	19,5	61,6	619,6
2007/2008	115,3	104,6	47,4	38,9	76,0	56,0	92,9	76,0	37,5	8,8	30,7	40,1	722,2
2008/2009	54,5	39,7	28,2	50,2	42,2	41,3	111,0	125,1	41,3	13,3	13,8	43,5	604,1
2009/2010	105,5	64,6	73,1	82,3	49,1	47,4	56,0	25,8	27,7	12,8	16,5	45,8	606,6
2010/2011	90,1	78,8	130,5	59,6	53,3	77,0	64,0	59,8	26,7	18,1	18,9	16,3	693,1
2011/2012	52,7	104,7	27,4	22,0	15,5	25,4	94,2	41,9	21,1	13,9	10,0	60,4	489,2
2012/2013	103,5	112,6	49,6	95,0	75,5	166,1	70,4	51,1	31,8	21,5	22,2	41,1	840,4
2013/2014	68,4	54,1	72,4	99,0	94,4	52,1	52,7	33,4	29,2	27,2	18,2	65,2	666,3
2014/2015	61,5	144,5	43,0	64,5	59,3	63,9	45,8	17,7	42,7	21,4	26,0	43,2	633,5
2015/2016	67,3	46,7	19,1	87,6	93,5	67,2	73,8	67,8	16,4	11,5	7,7	27,8	586,2
2016/2017	51,7	102,5	42,6	41,3	64,6	56,6	23,3	44,1	32,4	21,4	29,4	19,7	529,4
2017/2018	26,7	47,5	66,9	71,0	73,2	151,6	83,8	52,1	50,5	24,8	18,6	28,4	675,0
2018/2019	92,3	-	-	-	-	-	-	-	-	-	-	-	91,3

Precipitación superior a la media de la serie histórica 1981-2010



$$y_k = g \left(w_{vk} \cdot \dots \cdot f(\sum_i w_{ij} x_i + \theta_j) + w_v \right)$$

TENEMOS UN MODELO

TENEMOS UNOS DATOS

Y hacemos un ajuste → **entrenamiento**

REDES NEURONALES

Ejemplo: predecir precio vivienda

Training set.

Datos de ventas de los últimos años para características de la casa

Antigüedad	Tamaño	Tamaño jardín	Conflictividad	Precio
20	180	0	0.1	120000
10	130	100	0.2	200000
25	200	20	0.5	130000
2	100	0	0.2	190000
...
5	200	20	0.3	?



REDES NEURONALES

Ejemplo: predecir precio vivienda

Training set.

Datos de ventas de los últimos años para características de la casa

Antigüedad	Tamaño	Tamaño jardín	Conflictividad	Precio
20	180	0	0.1	120000
10	130	100	0.2	200000
25	200	20	0.5	130000
2	100	0	0.2	190000
...
5	200	20	0.3	?



EMPEZAMOS A DEFINIR LA RED QUE VAMOS A USAR

- 1) Es una red de regresión porque queremos saber un valor.
- 2) Primera capa 4 neuronas última l.
- 3) Última capa función de activación lineal

REDES NEURONALES

Ejemplo: predecir precio vivienda

Acabamos de decidir cuantas neuronas
y ya tenemos la función:

$$y_k = \sum_j w_{jk} \tanh(\sum_i w_{ij} x_i + \theta_j) + \theta_k$$



Y ahora hacemos un ajuste minimizando una loss function, por ejemplo la distancia cuadrática entre el **target** (en este caso el precio de la vivienda) y la función $f(x)$ que hemos construido:

$$E = \sum_n \frac{(t^n - y^n)^2}{N}$$

En el ajuste encontramos los parámetros del modelo que son los pesos y los bias w_{ij}, θ_j

REDES NEURONALES

Ejemplo: predecir precio vivienda

Acabamos de decidir cuantas neuronas
y ya tenemos la función:

$$y_k = \sum_j w_{jk} \tanh(\sum_i w_{ij} x_i + \theta_j) + \theta_k$$



Y ahora hacemos un ajuste minimizando una loss function, por ejemplo la distancia cuadrática entre el **target** (en este caso el precio de la vivienda) y la función $f(x)$ que hemos construido:

$$E = \sum_n \frac{(t^n - y^n)^2}{N}$$

En el ajuste encontramos los parámetros del modelo que son los pesos y los bias w_{ij}, θ_j

REDES NEURONALES

Ejemplo: predecir precio vivienda

Acabamos de decidir cuantas neuronas
y ya tenemos la función:

$$y_k = \sum_j w_{jk} \tanh(\sum_i w_{ij} x_i + \theta_j) + \theta_k$$



Y ahora hacemos un ajuste minimizando una loss function, por ejemplo la distancia cuadrática entre el **target** (en este caso el precio de la vivienda) y la función $f(x)$ que hemos construido:

$$E = \sum_n \frac{(t^n - y^n)^2}{N}$$

en este caso $k=1$, pero si hay más de un output, la función objetivo general quedaría (K número de outputs, N número set training) :

En el ajuste encontramos los parámetros del modelo que son los pesos y los bias w_{ij}, θ_j

REDES NEURONALES

Ejemplo: predecir precio vivienda

Una vez la red está entrenada tendremos un modelo que relaciona las características con el precio

Antigüedad	Tamaño	Tamaño jardín	Conflictividad	Precio
20	180	0	0.1	120000
10	130	100	0.2	200000
25	200	20	0.5	130000
2	100	0	0.2	190000
...
5	200	20	0.3	



$$y_k = \sum_j w_{jk} \tanh(\sum_i w_{ij} x_i + \theta_j) + \theta_k$$

REDES NEURONALES

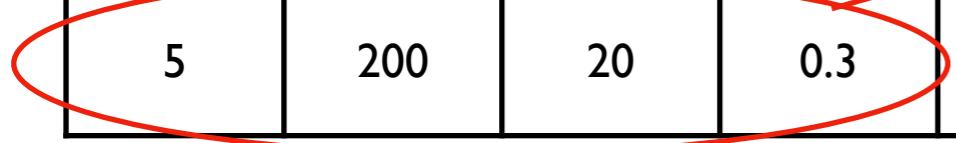
Ejemplo: predecir precio vivienda

Una vez la red está entrenada tendremos un modelo que relaciona las características con el precio

Antigüedad	Tamaño	Tamaño jardín	Conflictividad	Precio
20	180	0	0.1	120000
10	130	100	0.2	200000
25	200	20	0.5	130000
2	100	0	0.2	190000
...
5	200	20	0.3	



$$y_k = \sum_j w_{jk} \tanh(\sum_i w_{ij} x_i + \theta_j) + \theta_k$$



REDES NEURONALES

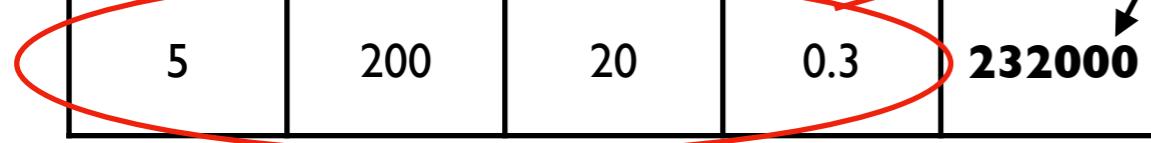
Ejemplo: predecir precio vivienda

Una vez la red está entrenada tendremos un modelo que relaciona las características con el precio

Antigüedad	Tamaño	Tamaño jardín	Conflictividad	Precio
20	180	0	0.1	120000
10	130	100	0.2	200000
25	200	20	0.5	130000
2	100	0	0.2	190000
...
5	200	20	0.3	232000



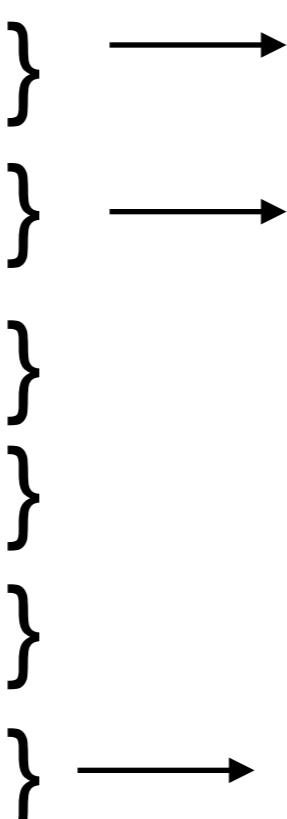
$$y_k = \sum_j w_{jk} \tanh(\sum_i w_{ij} x_i + \theta_j) + \theta_k$$



REDES NEURONALES

ENTRENAMIENTO

El ajuste que se hace en una red neuronal tiene ciertas características. Como se suele usar en casos de muchos datos, este ajuste se suele hacer a trozos y se van actualizando los



Minimizamos loss function

$$w_{ij}^0 = \text{random numbers}$$

Minimizamos E

$$w_{ij}^0 = w_{ij}^{\text{iter1}}$$

Minimizamos E

$$w_{ij}^0 = w_{ij}^{\text{iterN-1}}$$

$$w_{ij}^{\text{iter1}}$$

$$w_{ij}^{\text{iter2}}$$

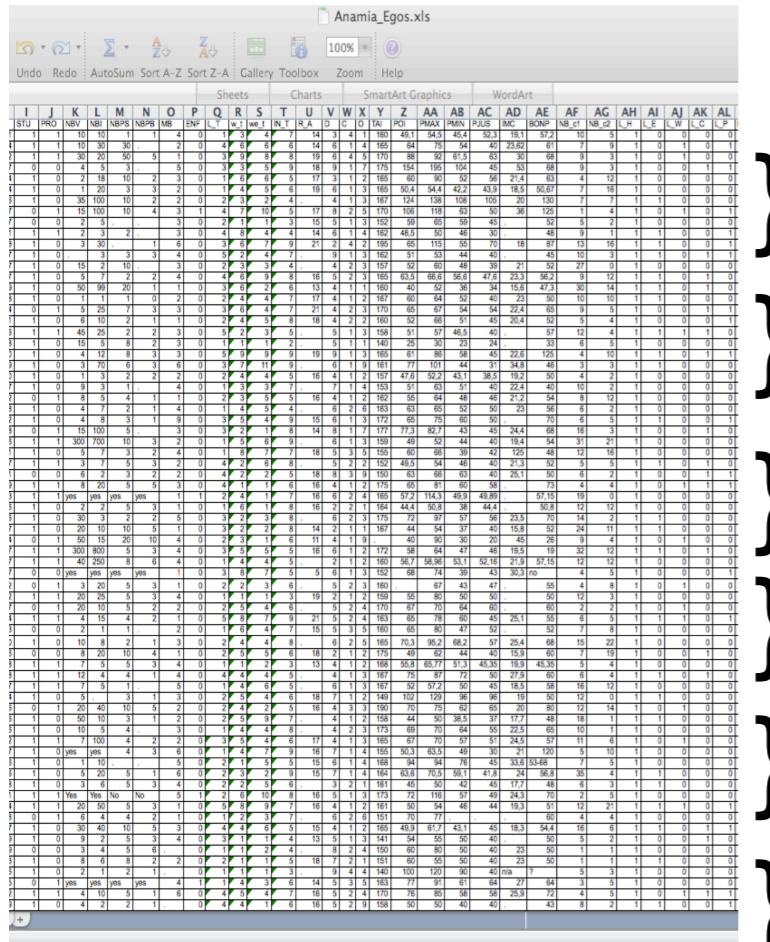
$$w_{ij}^{\text{iterN}}$$

Batch_size =tamaño de la muestra

REDES NEURONALES

ENTRENAMIENTO

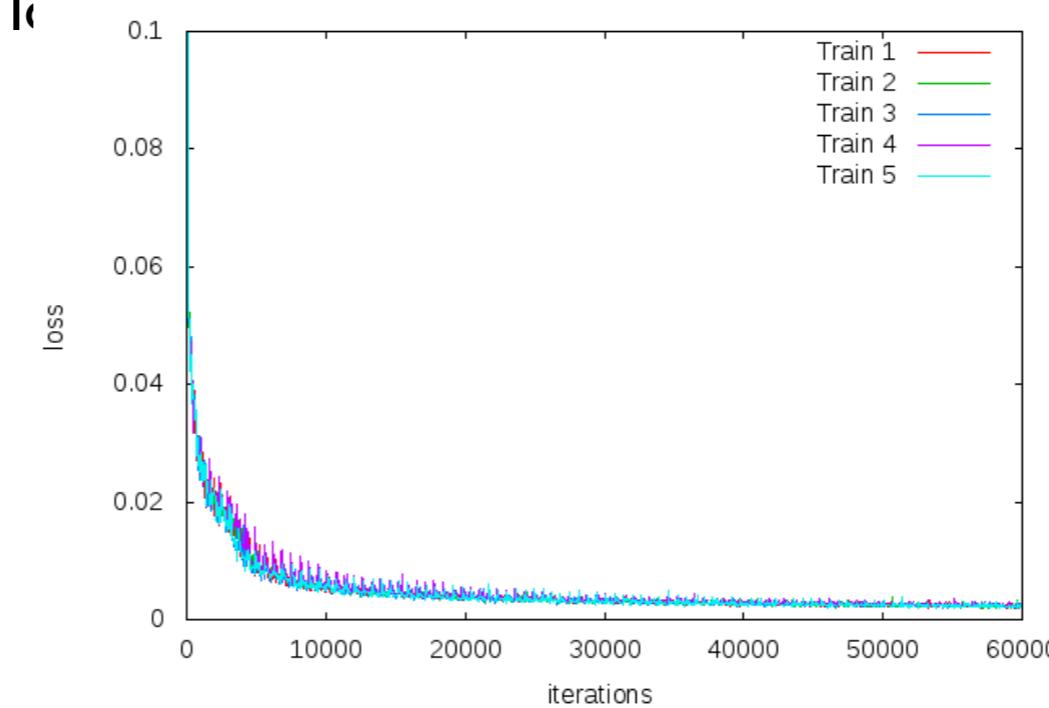
El ajuste que se hace en una red neuronal tiene ciertas características. Como se suele usar en casos de muchos datos, este ajuste se suele hacer a trozos y se van actualizando los



Iteración. Con los pesos calculados con los primeros datos como punto de partida, volvemos a hacer un ajuste con los 50 siguientes (o dependiendo del batch_size) y así hasta terminar los datos.

Época. Cuando hemos hecho un ajuste con cada uno de

Training loss vs. iterations

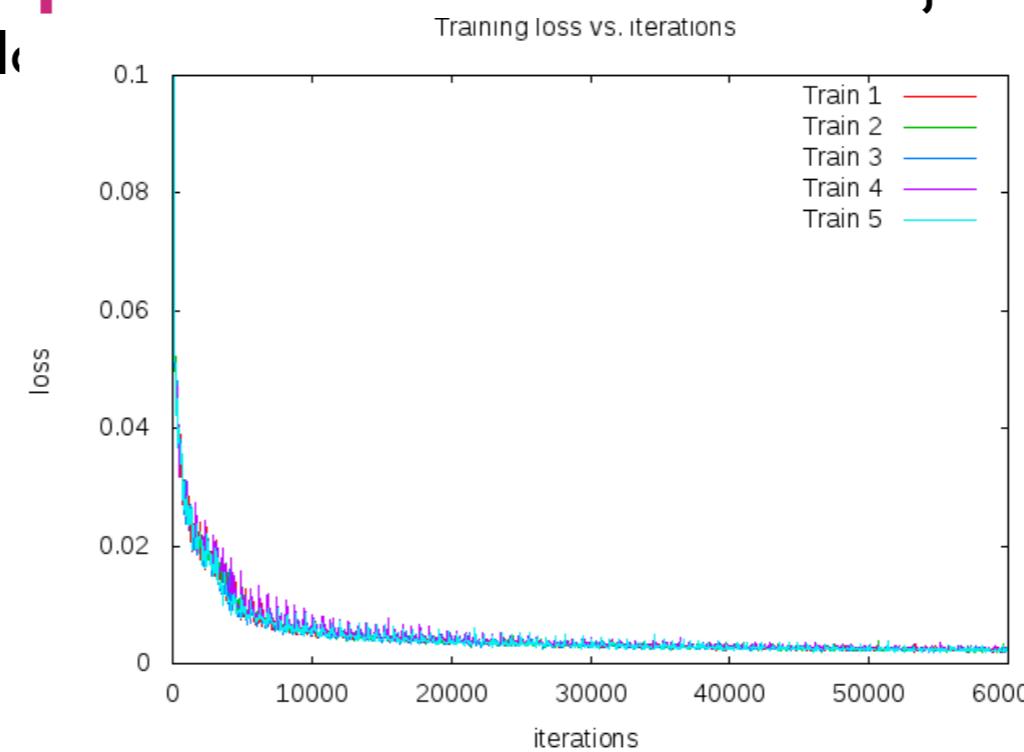


REDES NEURONALES

ENTRENAMIENTO

El ajuste que se hace en una red neuronal tiene ciertas características. Como se suele usar en casos de muchos datos, este ajuste se suele hacer a trozos y se van actualizando los

Iteración. Con los pesos calculados con los primeros datos como punto de partida, volvemos a hacer un ajuste con los 50 siguientes (o dependiendo del batch_size) y así hasta terminar los datos.



REDES NEURONALES

ENTRENAMIENTO

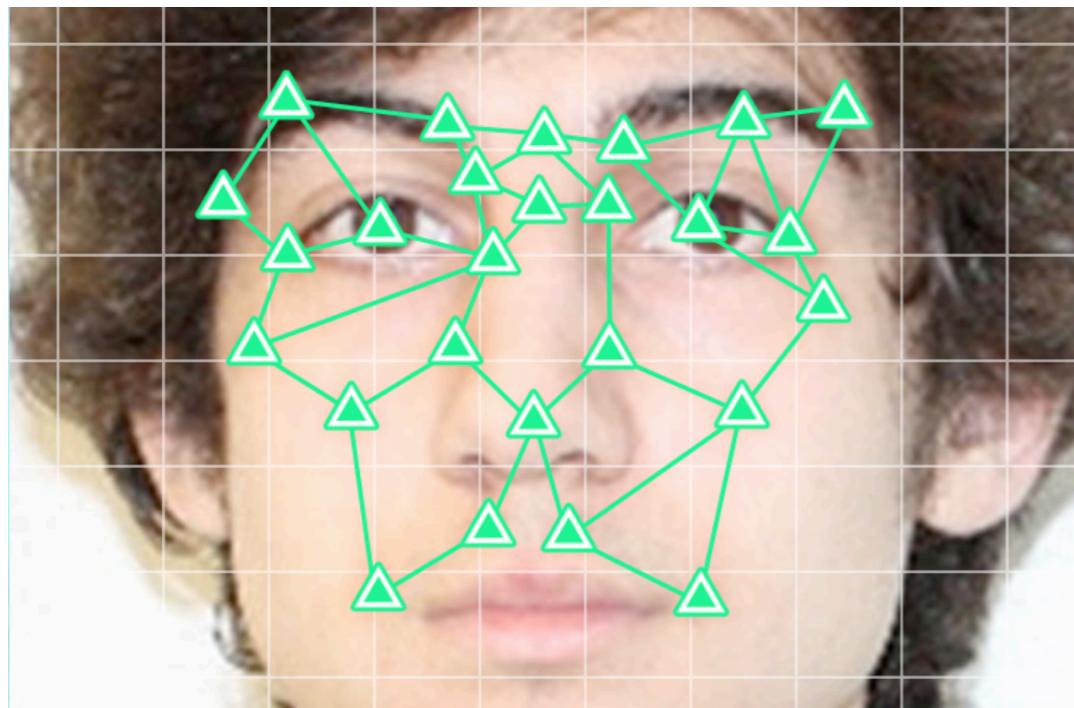
Otras cosas que se tienen que escoger al hacer el entrenamiento:

I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL							
1	1	10	10	1	1	4	0	1	3	7	14	3	4	1	160	48.1	54.5	45.4	52.3	16.1	37.2	10	5	1	0	0	0	0	0	0						
1	1	10	30	36	2	0	4	6	6	14	6	1	4	160	64	75	54	40	23.82	61	7	9	1	0	1	0	0	1	1							
1	0	2	10	30	5	1	0	1	3	7	10	6	4	5	170	88	92	61.5	63	30	68	9	3	1	0	1	0	0	0	0	0					
1	0	2	18	10	2	3	0	1	8	7	5	17	3	1	2	160	60	90	52	56	21.41	63	4	12	1	0	0	0	0	0	0	0				
1	0	11	20	3	3	2	0	1	4	6	10	6	1	3	160	90.4	54.4	42.2	43.9	18.5	50.67	7	16	1	0	0	0	0	0	0	0					
1	0	38	100	0	2	2	0	2	3	7	4	1	3	187	124	138	108	105	20	130	7	7	1	1	0	0	0	0	0	0						
1	0	15	100	10	4	3	0	1	7	10	2	1	2	160	100	100	36	100	20	100	9	1	0	0	0	0	0	0	0	0						
1	1	2	3	2	3	1	0	4	8	4	14	6	1	4	160	48.5	50	48	30	48	9	1	1	1	0	0	0	0	0	0	0					
1	0	51	30	1	6	0	0	1	6	9	21	2	4	2	195	65	115	55	70	18	87	13	16	1	1	0	0	0	0	0	0					
1	0	3	3	3	4	0	0	2	7	1	1	0	1	0	3	182	51	53	40	45	10	3	1	0	0	0	0	0	0	0	0	0				
1	0	18	100	10	4	3	0	1	7	10	2	1	2	160	100	100	36	100	20	100	9	1	0	0	0	0	0	0	0	0	0					
1	0	53	7	2	2	4	0	4	8	7	18	5	2	3	160	63.5	68.6	56.6	47.6	23.3	58.2	9	12	1	1	0	0	0	0	0	0	0				
1	0	50	99	26	11	1	0	2	8	6	13	4	1	1	160	40	52	36	34	15.6	47.3	30	14	1	1	0	0	0	0	0	0	0				
1	0	91	1	1	0	2	0	2	4	7	17	4	1	2	187	60	64	32	40	23	50	10	10	1	1	0	0	0	0	0	0	0				
1	0	81	5	2	1	1	0	2	4	7	17	4	1	2	187	60	64	32	40	23	50	10	10	1	1	0	0	0	0	0	0	0				
1	0	61	1	2	1	1	0	2	4	7	17	4	1	2	160	62	65	35	43	20.4	62	51	60	0	0	0	0	0	0	0	0	0	0			
1	1	45	25	0	2	3	0	3	2	5	1	0	1	3	158	91	67	40.5	40	57	12	41	1	1	1	1	0	0	0	0	0	0	0			
1	0	15	5	8	2	3	0	1	7	10	2	1	2	160	40	57	12	41	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0		
1	0	41	12	8	3	3	0	1	7	10	2	1	2	160	61	88	58	45	22.6	725	4	10	1	1	0	0	0	0	0	0	0	0	0			
1	0	3	4	2	1	1	0	2	4	7	17	4	1	2	160	62	64	32	40	23	50	10	10	1	1	0	0	0	0	0	0	0	0			
1	0	1	3	2	2	2	0	2	4	5	16	4	1	2	157	47.6	59.2	43.1	38.5	18.2	55	41	21	1	0	0	0	0	0	0	0	0	0			
1	0	91	3	1	4	0	0	1	7	1	4	153	51	63	51	40	22.4	40	10	21	1	0	0	0	0	0	0	0	0	0	0	0	0			
0	1	81	5	4	1	1	0	2	4	7	17	4	1	2	162	55	64	48	46	21.2	54	8	12	1	0	0	0	0	0	0	0	0	0	0		
0	1	4	2	1	2	1	0	2	4	7	17	4	1	2	160	62	65	32	40	23	50	8	21	1	0	0	0	0	0	0	0	0	0	0		
0	1	4	8	1	2	1	0	2	4	7	17	4	1	2	160	62	65	32	40	23	50	8	21	1	0	0	0	0	0	0	0	0	0	0		
0	1	15	100	5	1	3	0	2	4	7	17	4	1	2	177	77.3	82.7	43	45	24.4	68	16	31	1	0	0	0	0	0	0	0	0	0	0		
1	1	300	700	10	3	2	0	1	5	9	6	1	3	159	49	52	44	40	18.4	54	31	21	1	0	0	0	0	0	0	0	0	0	0	0		
1	0	51	7	3	2	4	0	1	8	7	18	5	3	5	150	60	68	39	42	12.5	48	12	18	1	0	0	0	0	0	0	0	0	0	0	0	
1	0	6	2	3	2	2	0	2	4	5	16	4	1	2	157	47.6	59.2	43.1	38.5	18.2	55	41	21	1	0	0	0	0	0	0	0	0	0	0	0	
1	1	81	20	5	5	3	0	2	4	7	18	4	1	2	175	65	81	66	58	73	4	4	1	0	0	0	0	0	0	0	0	0	0	0		
1	0	1	1	1	1	2	0	2	4	7	18	4	1	2	160	62	65	32	40	23	50	8	21	1	0	0	0	0	0	0	0	0	0	0	0	
1	0	2	2	1	2	1	0	2	4	7	18	4	1	2	160	62	65	32	40	23	50	8	21	1	0	0	0	0	0	0	0	0	0	0	0	
1	0	20	10	10	5	1	0	2	2	7	18	4	1	1	167	44	44	37	40	18.8	52	24	11	1	1	0	0	0	0	0	0	0	0	0	0	0
0	1	50	15	26	10	4	0	2	2	7	18	4	1	1	160	40	90	30	20	45	62	9	41	0	1	0	0	0	0	0	0	0	0	0	0	0
1	1	40	20	5	3	4	0	1	5	5	16	6	1	2	172	58	64	47	46	18.5	19	32	12	1	1	0	0	0	0	0	0	0	0	0	0	0
0	1	1	20	20	5	3	1	0	2	2	7	6	1	2	152	68	74	35	43	20.3	70	4	12</td													

REDES NEURONALES

- Aplicaciones

Reconocimiento de patrones

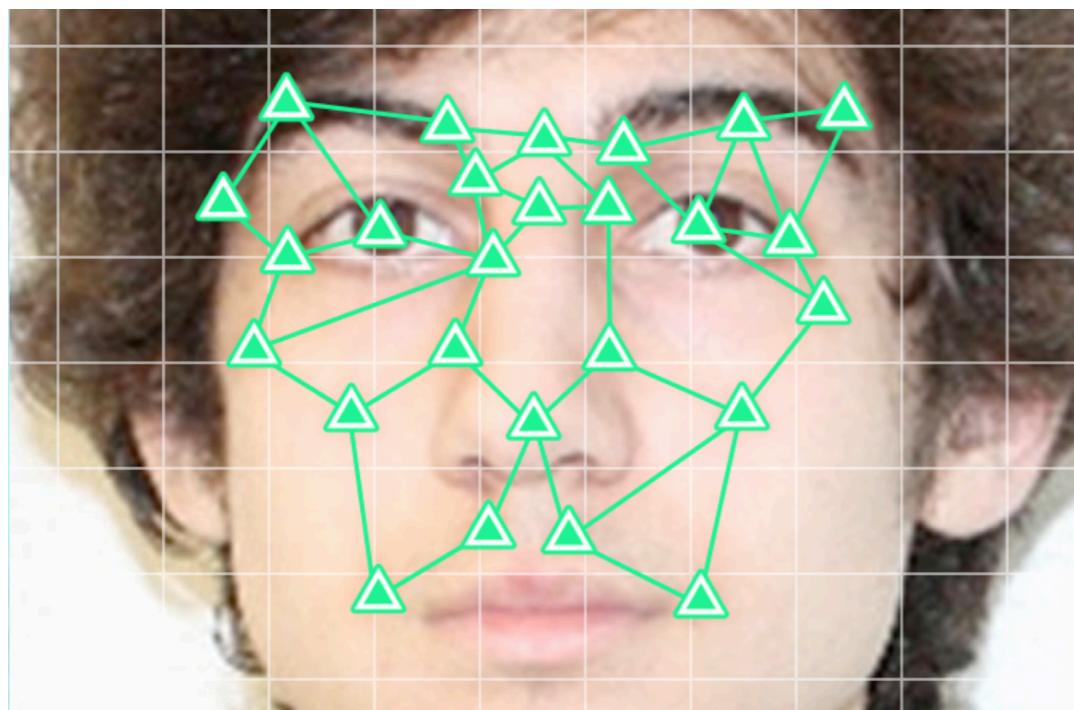


Lo trasladamos al lenguaje
máquina:

But the camera sees this:

194	210	201	212	199	213	215	195	178	158	182	209
180	189	190	221	209	205	191	167	147	115	129	163
114	126	140	188	176	165	152	140	170	106	78	88
87	103	115	154	143	142	149	153	173	101	57	57
102	112	106	131	122	138	152	147	128	84	58	66
94	95	79	104	105	124	129	113	107	87	69	67
68	71	69	98	89	92	98	95	89	88	76	67
41	56	68	99	63	45	60	82	58	76	75	65
20	43	69	75	56	41	51	73	55	70	63	44
50	50	57	69	75	75	73	74	53	68	59	37
72	59	53	66	84	92	84	74	57	72	63	42
67	61	58	65	75	78	76	73	59	75	69	50

Reconocimiento de patrones



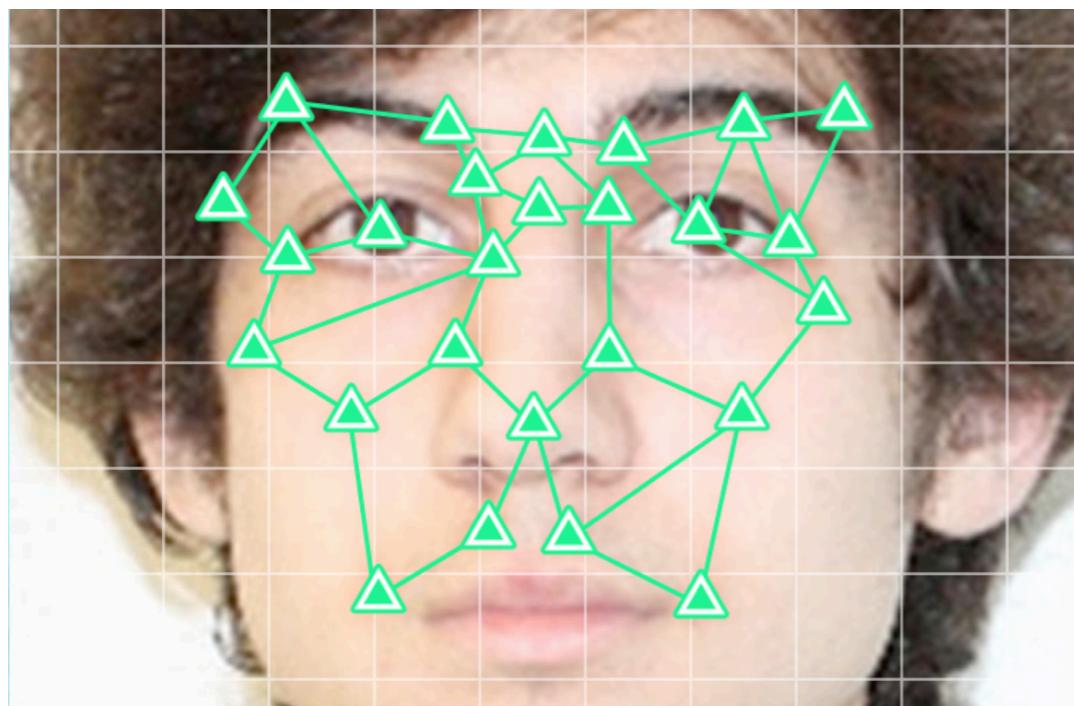
Si le damos un set de entrenamiento lo suficientemente grande, será capaz de detectar rostros

Lo trasladamos al lenguaje máquina:

But the camera sees this:

194	210	201	212	199	213	215	195	178	158	182	209
180	189	190	221	209	205	191	167	147	115	129	163
114	126	140	188	176	165	152	140	170	106	78	88
87	103	115	154	143	142	149	153	173	101	57	57
102	112	106	131	122	138	152	147	128	84	58	66
94	95	79	104	105	124	129	113	107	87	69	67
68	71	69	98	89	92	98	95	89	88	76	67
41	56	68	99	63	45	60	82	58	76	75	65
20	43	69	75	56	41	51	73	55	70	63	44
50	50	57	69	75	75	73	74	53	68	59	37
72	59	53	66	84	92	84	74	57	72	63	42
67	61	58	65	75	78	76	73	59	75	69	50

Reconocimiento de patrones



Si le damos un set de entrenamiento lo suficientemente grande, será capaz de detectar rostros

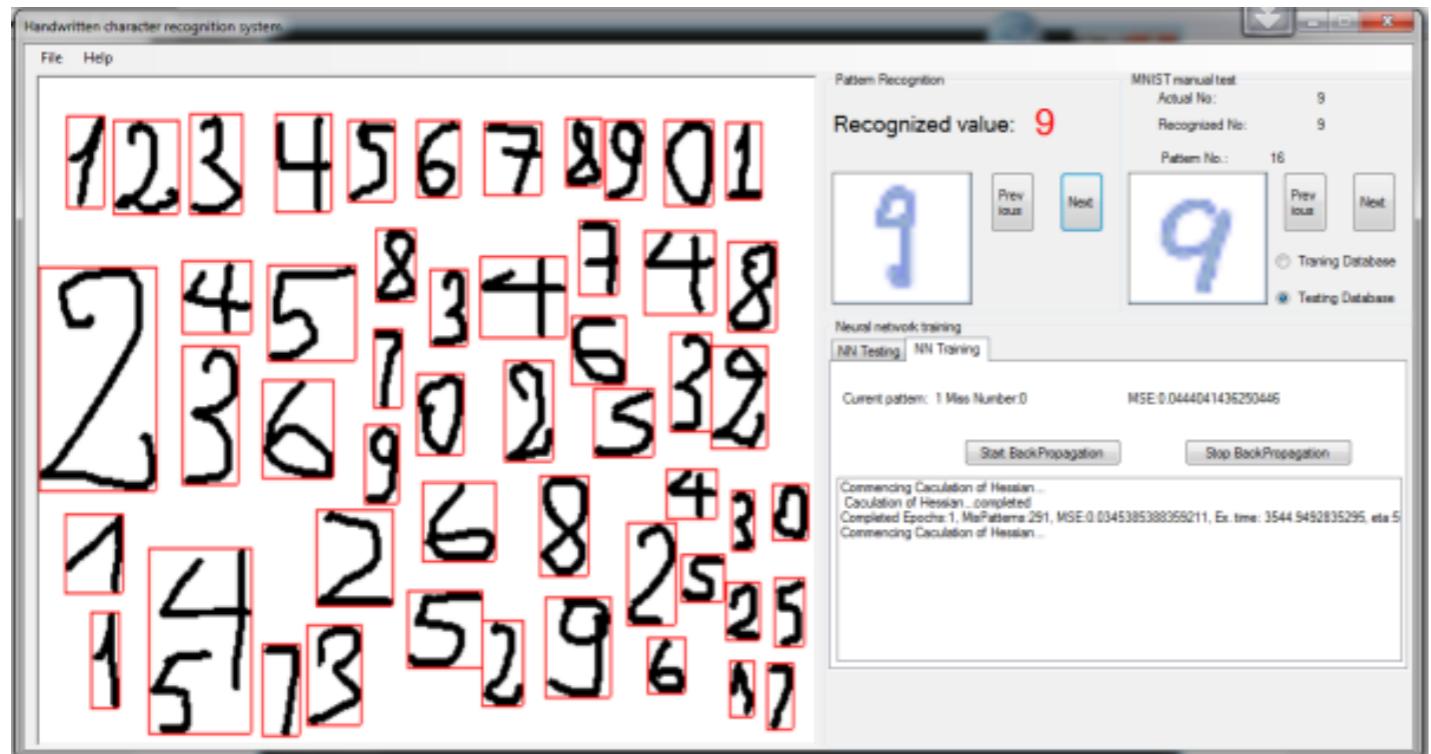
Lo trasladamos al lenguaje máquina:

But the camera sees this:

194	210	201	212	199	213	215	195	178	158	182	209
180	189	190	221	209	205	191	167	147	115	129	163
114	126	140	188	176	165	152	140	170	106	78	88
87	103	115	154	143	142	149	153	173	101	57	57
102	112	106	131	122	138	152	147	128	84	58	66
94	95	79	104	105	124	129	113	107	87	69	67
68	71	69	98	89	92	98	95	89	88	76	67
41	56	68	99	63	45	60	82	58	76	75	65
20	43	69	75	56	41	51	73	55	70	63	44
50	50	57	69	75	75	73	74	53	68	59	37
72	59	53	66	84	92	84	74	57	72	63	42
67	61	58	65	75	78	76	73	59	75	69	50

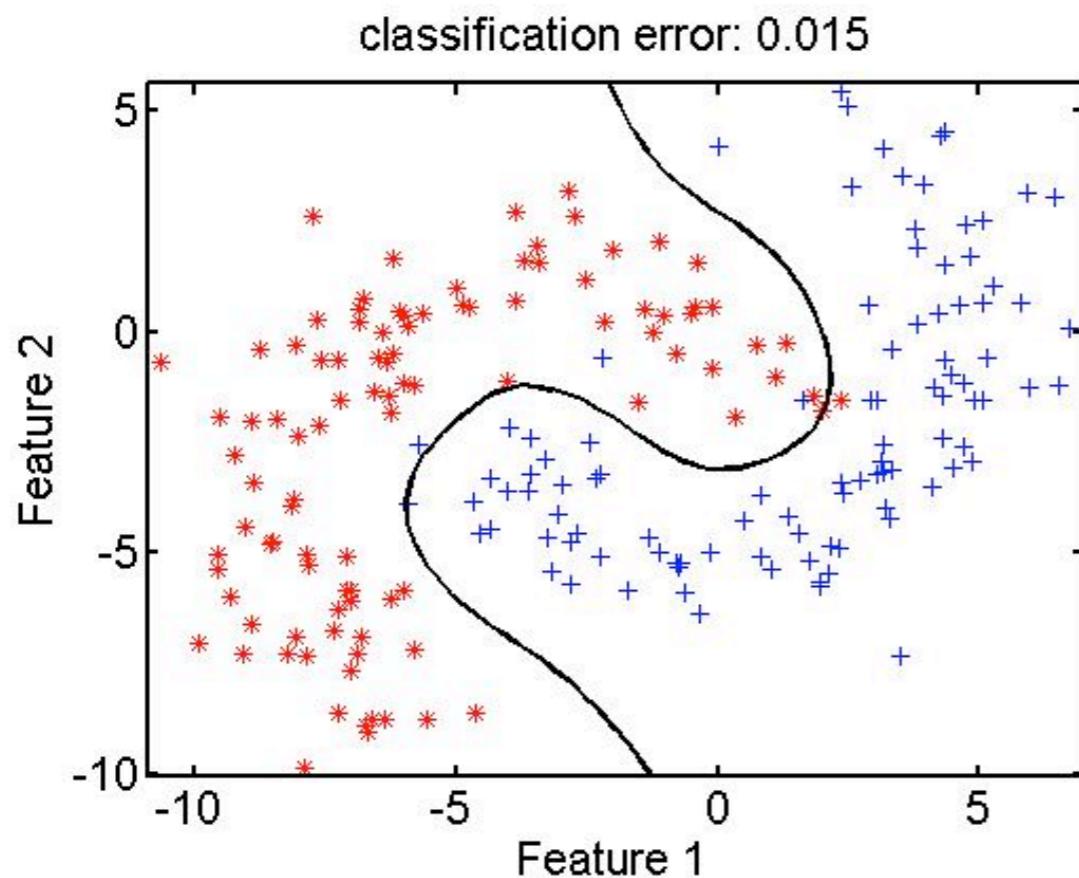
CNN deep learning,
resultados impresionantes!

Reconocimiento de patrones



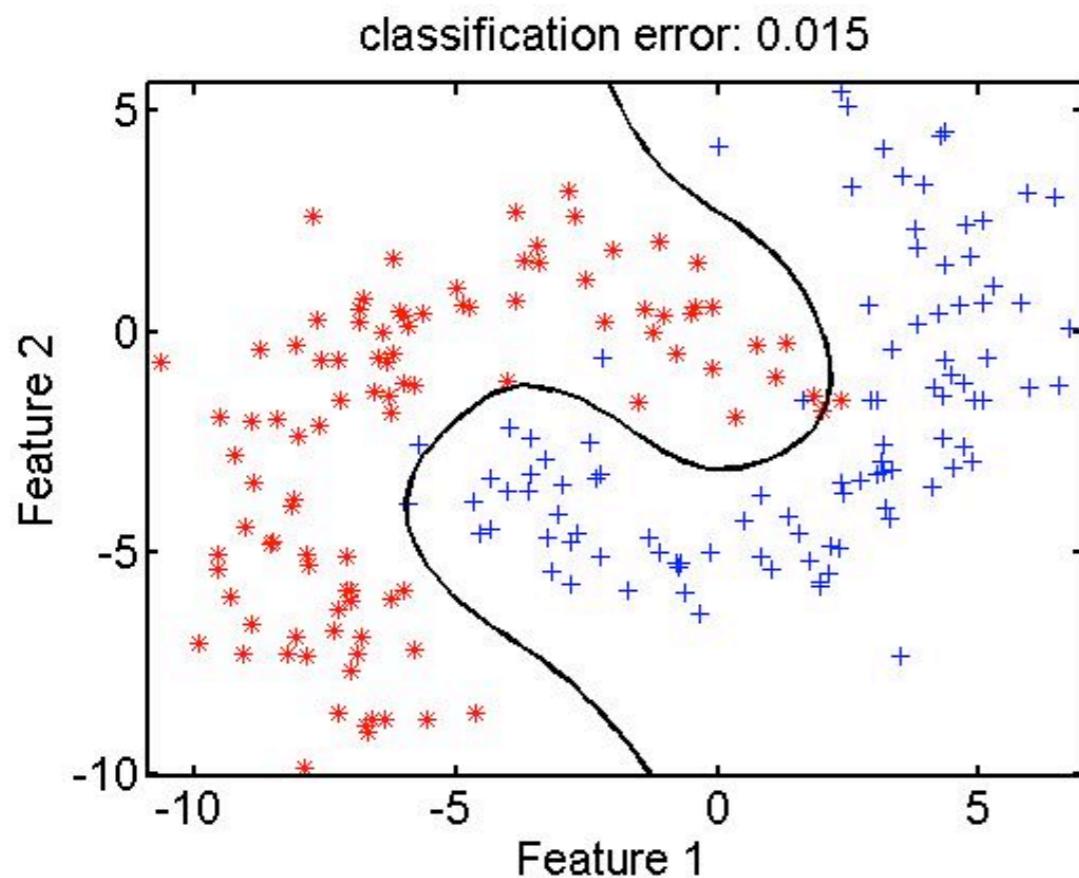
1. Se pasa lenguaje escrito a imágenes de 0 y 1
2. Se entrena la red con muchos ejemplos
3. Una vez entrenada la red es capaz de leer algo escrito a mano (o una matrícula de coche, ...)

Clasificación



- Es capaz de separar objetos en distintas clases
- Problemas de muchas dimensiones y altamente no lineal

Clasificación



- Es capaz de separar objetos en distintas clases
- Problemas de muchas dimensiones y altamente no lineal

[World Congress on Medical Physics and Biomedical Engineering 2018](#) pp 809-813 | [Cite as](#)

Baby Cry Recognition Using Deep Neural Networks

Clasificación

1. Graban sonido.
2. Lo pasan a serie temporal $f(t)$
3. Lo clasifican (saben en ese caso lo que le pasa el niño)
(clase 1, Hambre, clase 2 Dolor, clase 3 sueño,
clase 4 mamá...)
4. Entrenan la red
5. Grabas a tu hijo y la red te dirá que le pasa (en realidad te dice probabilidad de que
le pase 1, 2, 3 o 4,...)



Clasificación

[World Congress on Medical Physics and Biomedical Engineering 2018](#) pp 809-813 | [Cite as](#)

Baby Cry Recognition Using Deep Neural Networks

1. Graban sonido.
2. Lo pasan a serie temporal $f(t)$
3. Lo clasifican (saben en ese caso lo que le pasa el niño)
(clase 1, Hambre, clase 2 Dolor, clase 3 sueño,
clase 4 mamá...)
4. Entrenan la red
5. Grabas a tu hijo y la red te dirá que le pasa (en realidad te dice probabilidad de que
le pase 1, 2, 3 o 4,...)



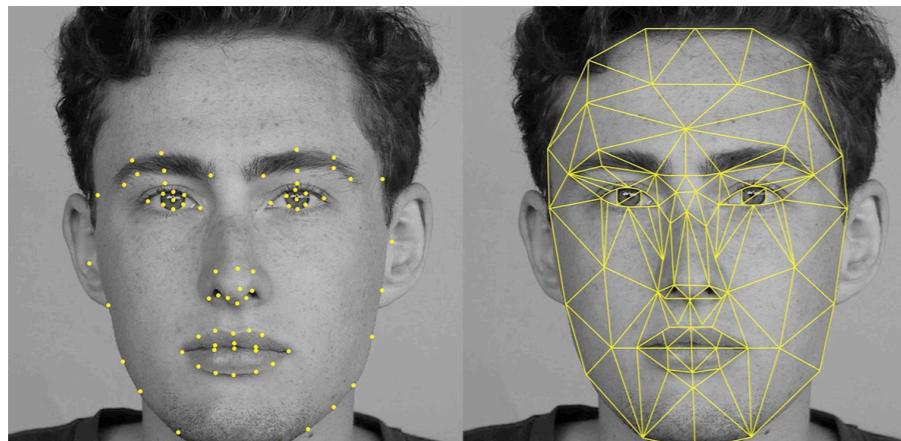
REDES NEURONALES

APLICACIONES

Reconocimiento imágenes. Una máquina lee una imagen como la intensidad en cada pixel.
Eso se puede dar como input a una red neuronal



Identificación de plantas, animales



Reconocimiento facial

Reconocimiento de etiquetas, texto,...



Clasificación de Galaxias

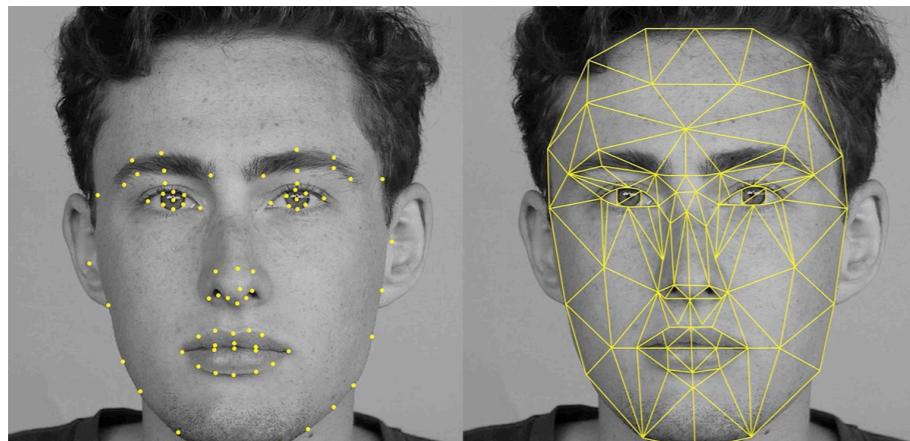
REDES NEURONALES

APLICACIONES

Reconocimiento imágenes. Una máquina lee una imagen como la intensidad en cada pixel.
Eso se puede dar como input a una red neuronal



Identificación de plantas, animales



Reconocimiento facial

Reconocimiento de etiquetas, texto,...



Clasificación de Galaxias

DEEP LEARNING

REDES NEURONALES

- Evaluación del training y otras consideraciones

REDES NEURONALES

EVALUACIÓN: SET DE VALIDACIÓN

Durante el entrenamiento vamos evaluando con la loss función a cada iteración. Minimos cuadrados para regresión y cross entropy para

Clasificación *loss function*:

$$H(p, q) = - \sum_{x \in \mathcal{X}} p(x) \log q(x)$$

Regresión *loss function*:

$$E = \sum_n \frac{(t^n - y^n)^2}{N}$$

REDES NEURONALES

EVALUACIÓN: SET DE VALIDACIÓN

Durante el entrenamiento vamos evaluando con la loss función a cada iteración. Minimos cuadrados para regresión y cross entropy para

Clasificación *loss function*:

$$H(p, q) = - \sum_{x \in \mathcal{X}} p(x) \log q(x)$$

Regresión *loss function*:

$$E = \sum_n \frac{(t^n - y^n)^2}{N}$$

REDES NEURONALES

EVALUACIÓN: SET DE VALIDACIÓN

Durante el entrenamiento vamos evaluando con la loss función a cada iteración. Minimos cuadrados para regresión y cross entropy para

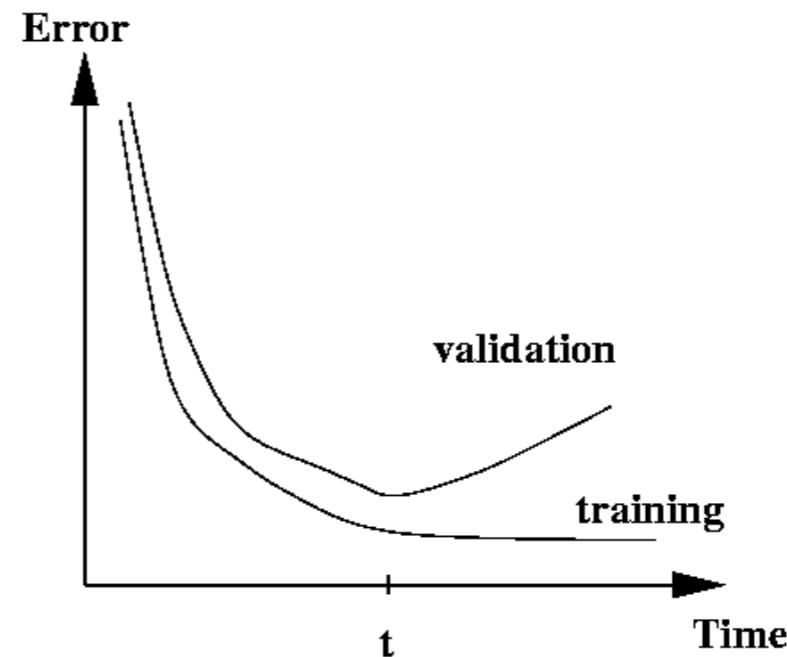
Clasificación *loss function*:

$$H(p, q) = - \sum_{x \in \mathcal{X}} p(x) \log q(x)$$

También calculamos la *loss function* para un **set de validación**, datos conocidos que no hemos usado en el training

Regresión *loss function*:

$$E = \sum_n \frac{(t^n - y^n)^2}{N}$$



REDES NEURONALES

EVALUACIÓN TEST VALIDACIÓN

Una vez terminado el entrenamiento se usan también otros indicadores sobre el set de validación para ver el desempeño de la red. Por ejemplo:

Regresión:

$$RelErr = \sum \frac{\frac{|t^n - y^n|}{t^n}}{N}$$
$$\rho = \frac{\sigma_{ty}^n}{\sigma_t \sigma_y}$$

REDES NEURONALES

EVALUACIÓN TEST VALIDACIÓN

Una vez terminado el entrenamiento se usan también otros indicadores sobre el set de validación para ver el desempeño de la red. Por ejemplo:

Clasificación

Precisión (o TPR)

$$TPR = \frac{TP}{TP+FP}$$

Compleitud (o recall)

$$Rec = \frac{TP}{TP+FN}$$

Regresión:

$$RelErr = \sum \frac{|t^n - y^n|}{N}$$

$$\rho = \frac{\sigma_{ty}}{\sigma_t \sigma_y}$$

REDES NEURONALES

ENTRENAMIENTO

Algunas observaciones:

- * En datos con muchas variables de magnitudes distintas es importante **estandarizar** los inputs y los outputs para que pesen por igual todos

$$x_i = \frac{x_i - \langle x_i \rangle}{\sigma_i}$$

- * Sólo podremos generalizar en el **rango del entrenamiento**. Siendo menos efectivo en los bordes (*edge problem*)

- * En ocasiones puede ser útil entrenar varias redes con training sets distintos y hacer promedio de los resultados.

REDES NEURONALES

ARQUITECTURA

¿Cuántas capas ocultas hay que poner? Y cuantas neuronas en cada capa?

No hay muchas pistas, hay que probar bastantes configuraciones.

Pista 1. Universal approximation theorem (Cybenko 1989): Una red neuronal feed-forward de una **sola capa** con un número finito de neuronas puede aproximar cualquier función continua real.

Pista 2. Pocas neuronas pueden hacer que no lleguemos a trazar bien el problema y demasiadas puede ser que sobreajustemos (overfitting). Un número para empezar $(N_{in} + N_{out})/2$, **pero no es una regla fiable!!**

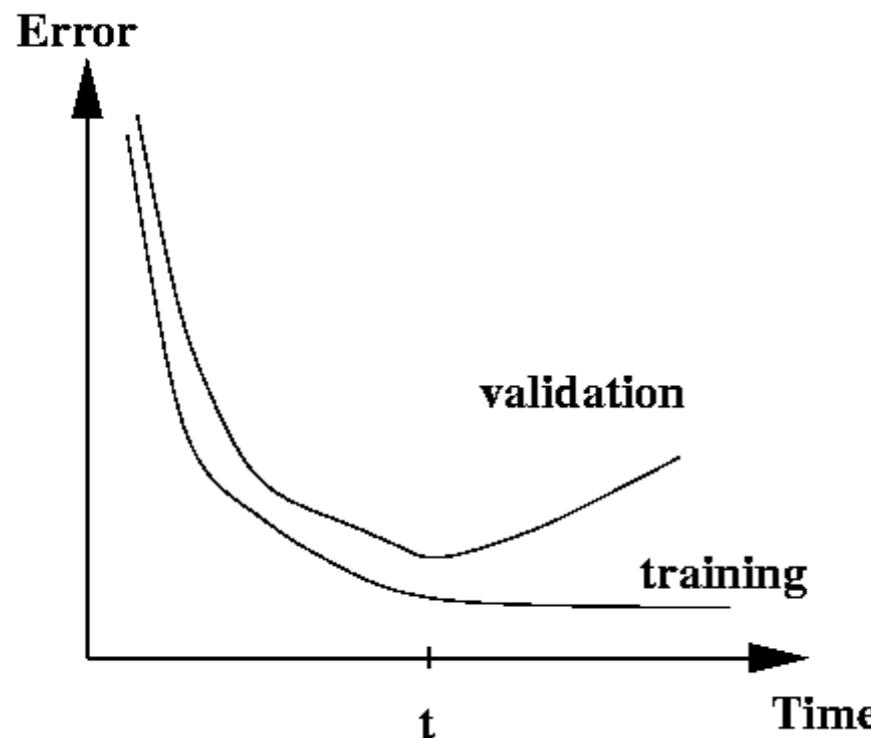
¡Prueba y error!

REDES NEURONALES

ENTRENAMIENTO

Vigilar con el overfitting.

Posibles soluciones:



- 1) Estamos usando demasiados parámetros:**
→ reducir capas o número de neuronas por capa
- 2) Early stoppin** → Paramos entrenamiento en iteración donde empieza overfitting
- 3) Optimizer options:** Paramos entrenamiento en iteración donde empieza overfitting

REDES NEURONALES

GENERALIZACIÓN

Una vez entrenada la red ponemos cualquier input x y nos dará el resultado que predice la red.

REDES NEURONALES

GENERALIZACIÓN

Una vez entrenada la red ponemos cualquier input x y nos dará el resultado que predice la red.

$$y_k = g\left(\left(\sum_j w_{jk} \dots f\left(\sum_i w_{ij} x_i\right)\right)\right)$$

REDES NEURONALES

PREDICTIVE ANALYSES

- Con la digitalización y la tecnología Big Data se dan las condiciones perfectas para sacar el máximo provecho de estas herramientas.
- con DEEP Learning (redes neuronales convolucionales) se obtienen resultados sorprendentemente buenos de clasificación, reconocimiento de imágenes, detección de objetos.



NN con Python

(vemos ejemplo con notebook)

sklearn
(otros Keras, pylearn,...)



mlp=MLPClassifier(hidden_layer_sizes=,activation=,batch_size=,verbose=...)

Objeto donde definimos arquitectura de la red para clasificación

mlp=MLPRegressor(hidden_layer_sizes=,max_iter=,verbose=True,...)

Objeto donde definimos arquitectura de la red para regresión

mlp.fit(x,y) Entrena la red

mlp.predict(xnew) Hace predicciones, nos da ynew para xnew

confusion_matrix(t,predictions) True positives and false positives de cada clase

classification_report(t,predictions) precision and recall for classification problems