

Time Series Forecasting

Introducción

National University of Singapore (NUS)

Félix Fuentes

Curso 2020/2021 – Edición II

Fecha 27/03/2021

Presentación

- PhD en Machine Learning (Universidad Politécnica de Valencia, 2018)
 - Tesis desarrollada en el campo del aprendizaje no supervisado: "A system for modeling social traits in realistic faces with artificial intelligence"
- Investigador en la Universidad Nacional de Singapur (NUS)
 - Mejora de la calidad de imagen microscópica para agilizar su adquisición mediante métodos generativos
- Experiencia profesional en la industria (London/Cambridge, UK)
 - Detección de *facial landmarks*
 - Segmentación de rasgos faciales y de uñas
 - Generación y transferencia de maquillaje
 - ...todo ello en *edge devices* (dispositivos móviles y con pocos recursos disponibles)
 - Detección y segmentación de instrumentación quirúrgica
 - Aprendizaje semi/débilmente supervisado
- Experiencia en docencia
 - Machine Learning y Aprendizaje No Supervisado, en la Universidad Internacional de Valencia (VIU)
 - Aprendizaje Automático en la Universitat Oberta de Catalunya (UOC)
 - RNNs y Generative Adversarial Networks, en EDEM ☺

Presentación

A system for modeling social traits in realistic faces with artificial intelligence

Fuentes Hurtado, F.J. (2018). A system for modeling social traits in realistic faces with artificial intelligence [Tesis doctoral no publicada]. Universitat Politècnica de València. <https://doi.org/10.4995/Thesis/10251/101943>

Por favor, use este identificador para citar o enlazar este ítem: <http://hdl.handle.net/10251/101943>

Files in this item



[Open/Preview](#)

Name: Fuentes - A system ...
Size: 69.67Mb
Format: PDF



[Open/Preview](#)

Name: indice.pdf
Size: 42.02Kb
Format: PDF



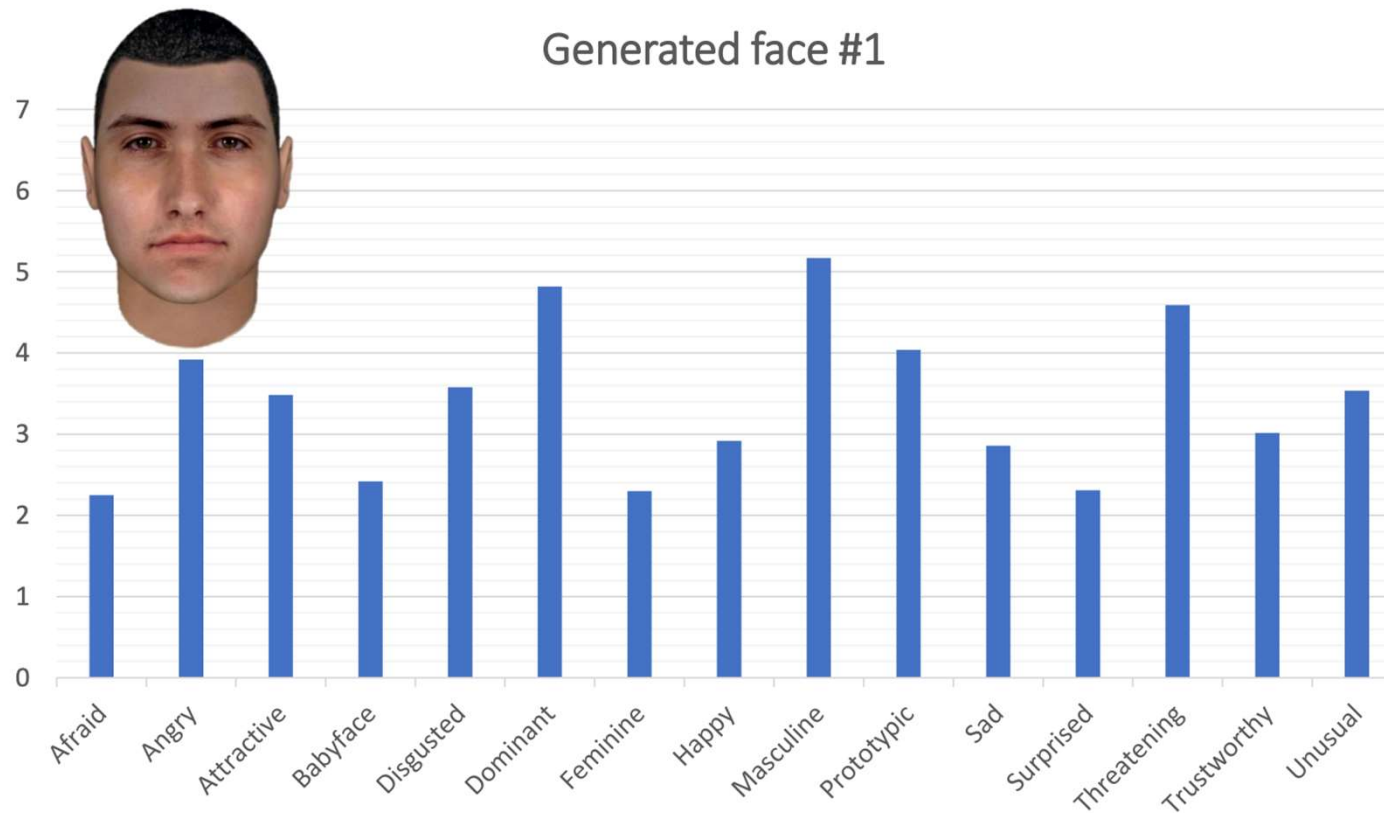
[Open/Preview](#)

Name: ResumenCastellano.pdf
Size: 21.23Kb
Format: PDF

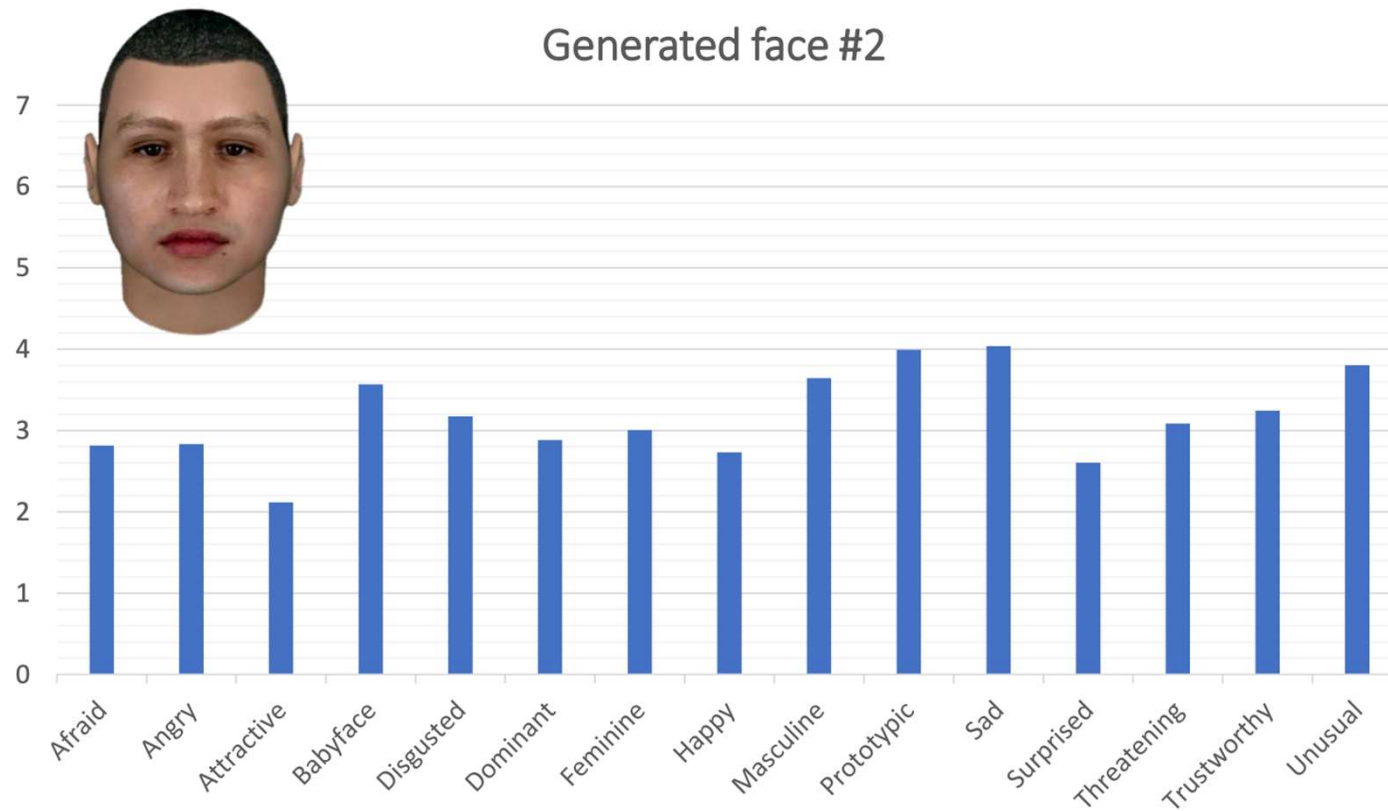
Item Metadata

Title:	A system for modeling social traits in realistic faces with artificial intelligence		
Author:	Fuentes Hurtado, Félix José		
Director(s):	 Diego Más, José Antonio	 Naranjo Ornedo, Valeriana	
UPV Unit:	Universitat Politècnica de València. Departamento de Comunicaciones - Departament de Comunicacions		
Read date / Event date:	2018-04-11	Issued date:	2018-05-14

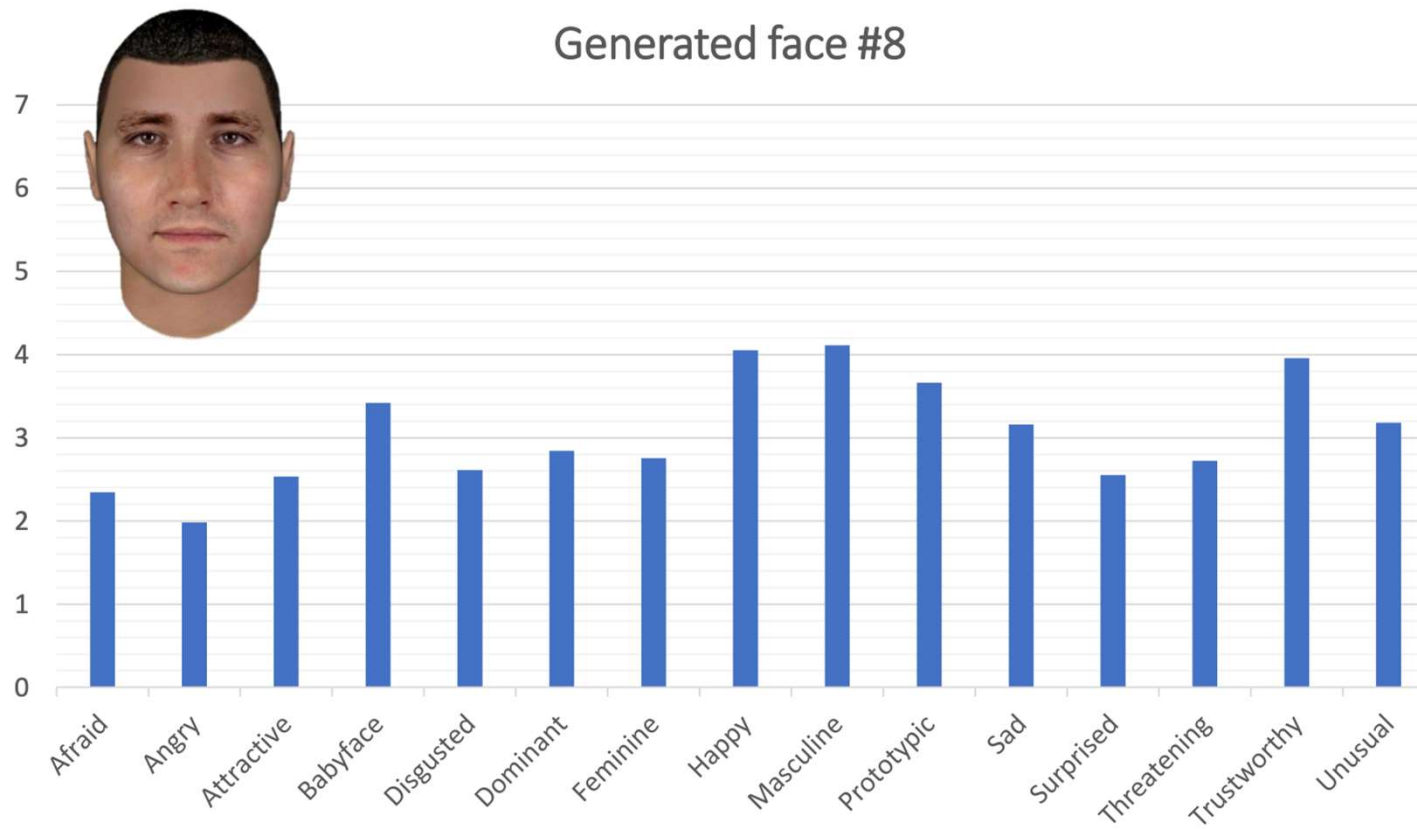
Presentación



Presentación



Presentación



Presentación

Líneas de investigación

- Aprendizaje débilmente supervisado (weakly supervised learning)
- Aprendizaje no supervisado
- Redes generativas (Generative Adversarial Networks)
- Investigación aplicada: detección de objetos en videos, clasificación de imágenes, segmentación y regresión, etc.

Perfil de Google Scholar: <https://scholar.google.es/citations?hl=es&user=XDrOZr0AAAAJ>

Perfil de LinkedIn: <https://www.linkedin.com/in/felixfuentesh/>

Presentación



Félix Fuentes-Hurtado

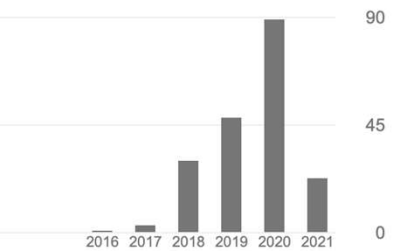
National University of Singapore (NUS)
Dirección de correo verificada de upv.es
deep learning machine learning computer vision

SEGUIR

<input type="checkbox"/>	TÍTULO	CITADO POR	AÑO
<input type="checkbox"/>	Consumer neuroscience-based metrics predict recall, liking and viewing rates in online advertising J Guixeres, E Bigné, JM Ausin Azofra, M Alcañiz Raya, ... Frontiers in psychology 8, 1808	58	2017
<input type="checkbox"/>	EvoDeep: a new evolutionary approach for automatic deep neural networks parametrisation A Martín, R Lara-Cabrera, F Fuentes-Hurtado, V Naranjo, D Camacho Journal of Parallel and Distributed Computing 117, 180-191	42	2018
<input type="checkbox"/>	A comparison of physiological signal analysis techniques and classifiers for automatic emotional evaluation of audiovisual contents A Colomer Granero, F Fuentes-Hurtado, V Naranjo Ornedo, ... Frontiers in computational neuroscience 10, 74	36	2016
<input type="checkbox"/>	Evolving deep neural networks architectures for android malware classification A Martín, F Fuentes-Hurtado, V Naranjo, D Camacho 2017 IEEE Congress on Evolutionary Computation (CEC), 1659-1666	25	2017
<input type="checkbox"/>	EasyLabels: weak labels for scene segmentation in laparoscopic videos F Fuentes-Hurtado, A Kadkhodamohammadi, E Flouty, S Barbarisi, ... International journal of computer assisted radiology and surgery 14 (7 ...	14	2019
<input type="checkbox"/>	Automatic classification of human facial features based on their appearance F Fuentes-Hurtado, JA Diego-Mas, V Naranjo, M Alcañiz PloS one 14 (1), e0211314	5	2019

Citado por

	Total	Desde 2016
Citas	195	195
Índice h	5	5
Índice i10	5	5



Coautores

EDITAR

	Valery Naranjo Universidad Politécnica de Valen...	>
	Prof. Mariano Alcañiz Raya Professor of Biomedical Enginee...	>
	Adrián Colomer Granero Researcher, Universidad Politéc...	>
	Alejandro Martín PhD in Computer Science, Assis...	>
	David Camacho	>

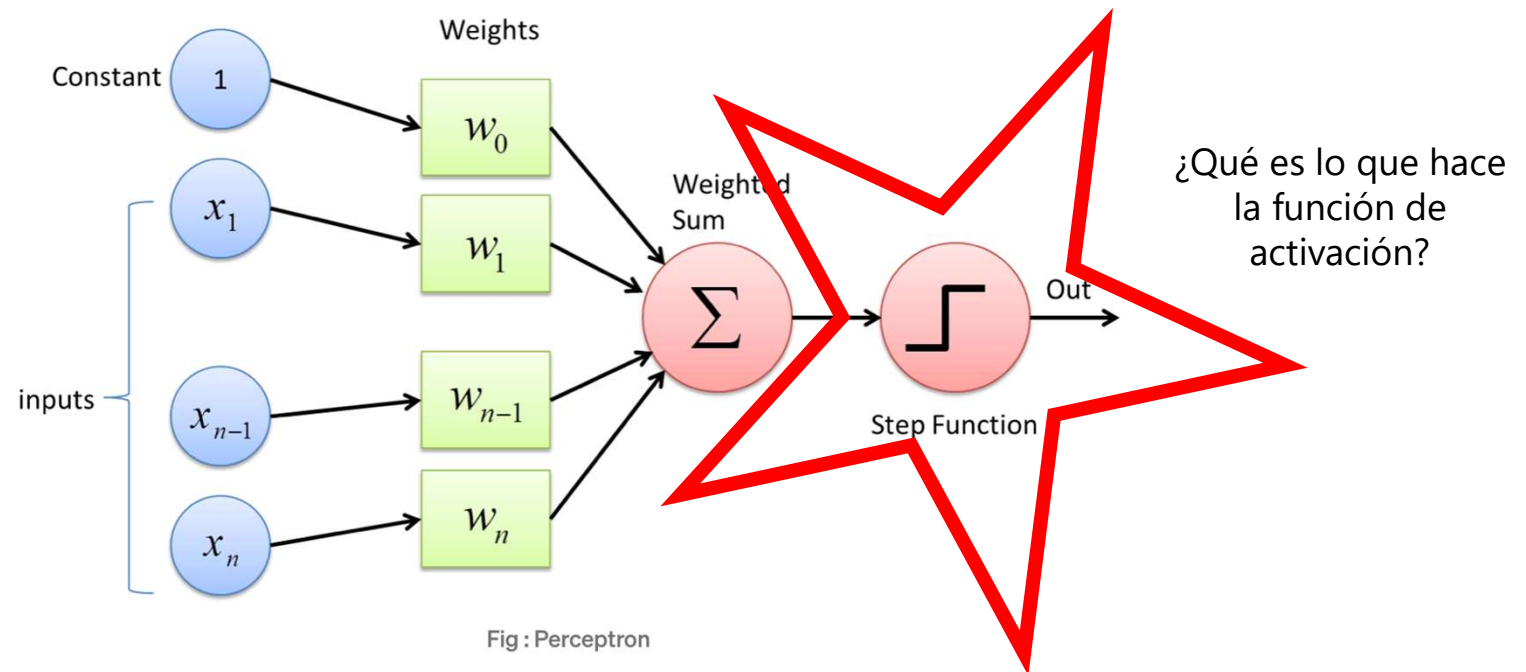
Índice

- Redes neuronales aplicadas a DL
 - Funciones de activación
 - Complejidad
 - Ejemplos prácticos
- Redes recurrentes
 - Introducción
 - Predicción de series temporales con una variable
 - Predicción de series temporales multivariantes
 - El problema de la memoria y la solución
 - Ejemplos prácticos con LSTMs y XGBoost
- Transformers
 - Introducción
 - Predicción de series temporales

Redes neuronales en DL

La red neuronal: ¿Por qué tienen esta configuración?

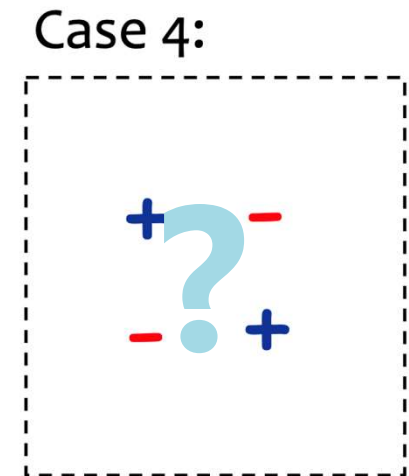
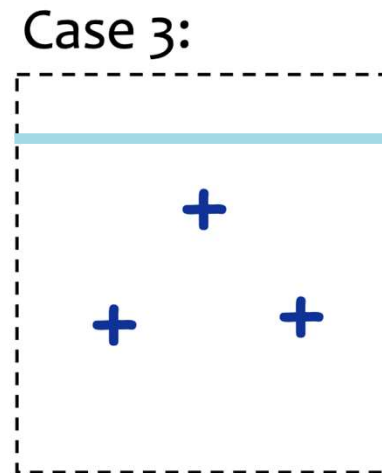
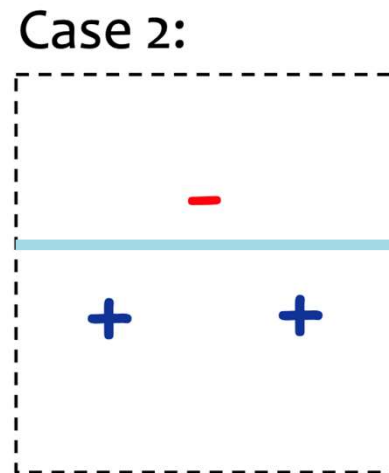
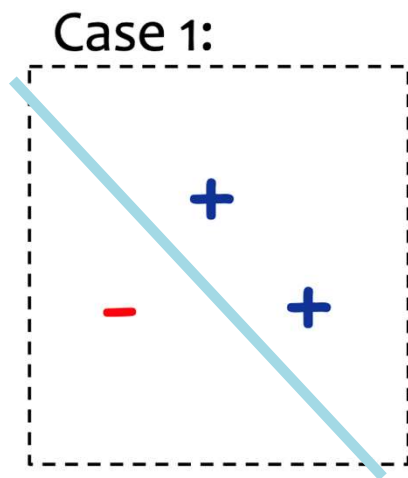
La versión básica (el Perceptrón) es muy similar a una regresión lineal. ¿En qué se diferencian?



Redes neuronales en DL

¿Qué problemas podemos resolver con una red neuronal?

Con un perceptrón podríamos hallar una frontera de decisión lineal



Pero... ¿es suficiente?

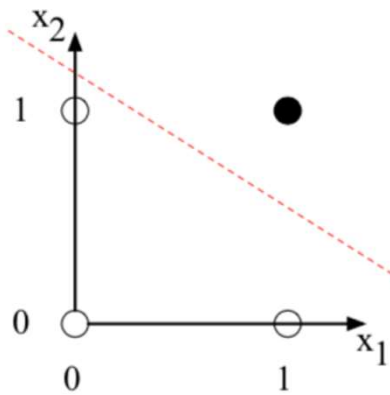
¿Cómo lo arreglamos?

Redes neuronales en DL

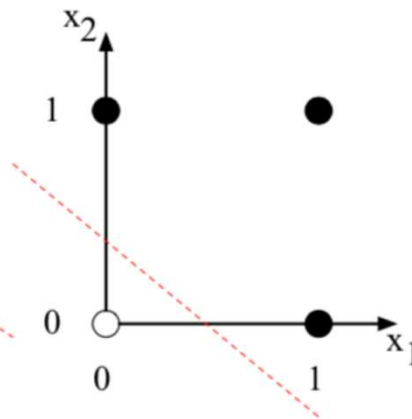
Problemas separables y no separables linealmente

El perceptrón puede reproducir el comportamiento de los operadores AND y OR porque son datos linealmente separables.

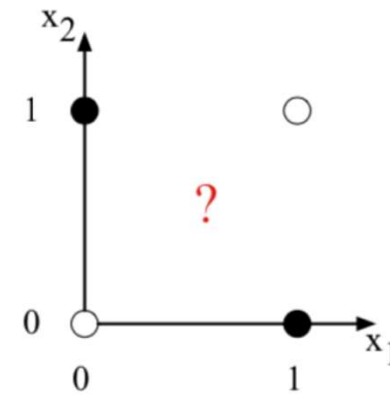
Sin embargo, en el caso de XOR, los datos no son linealmente separables, y no es posible encontrar una recta que los separe.



a) x_1 AND x_2



b) x_1 OR x_2

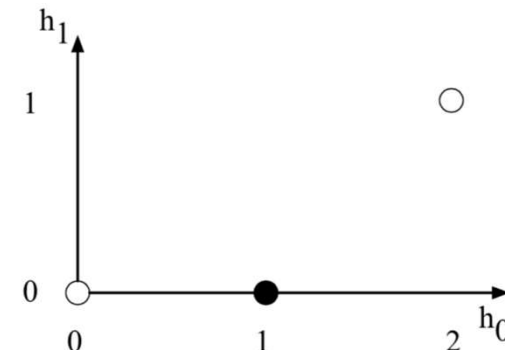
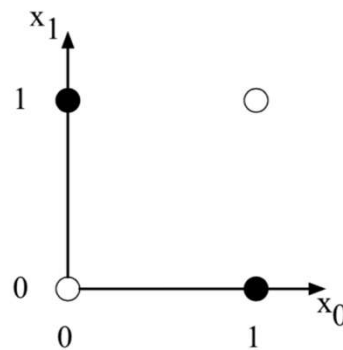
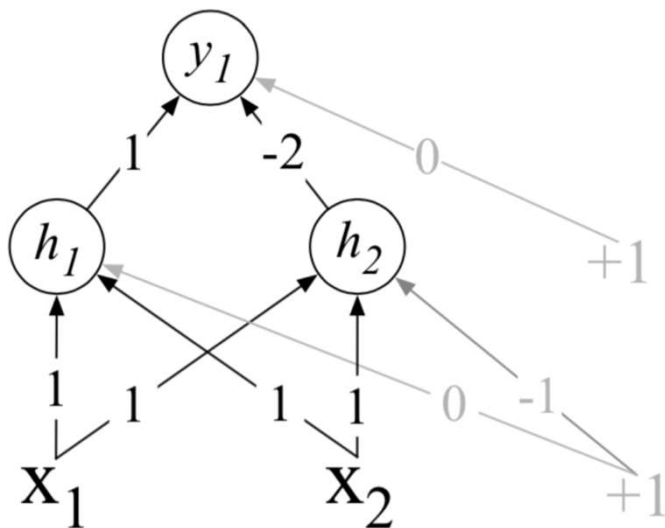


c) x_1 XOR x_2

Redes neuronales en DL

Problemas separables y no separables linealmente

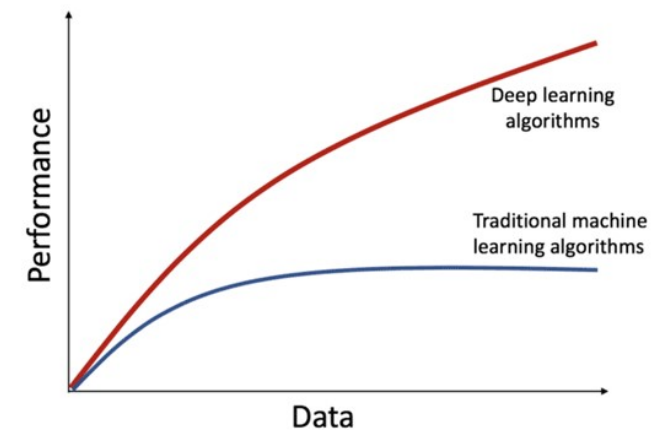
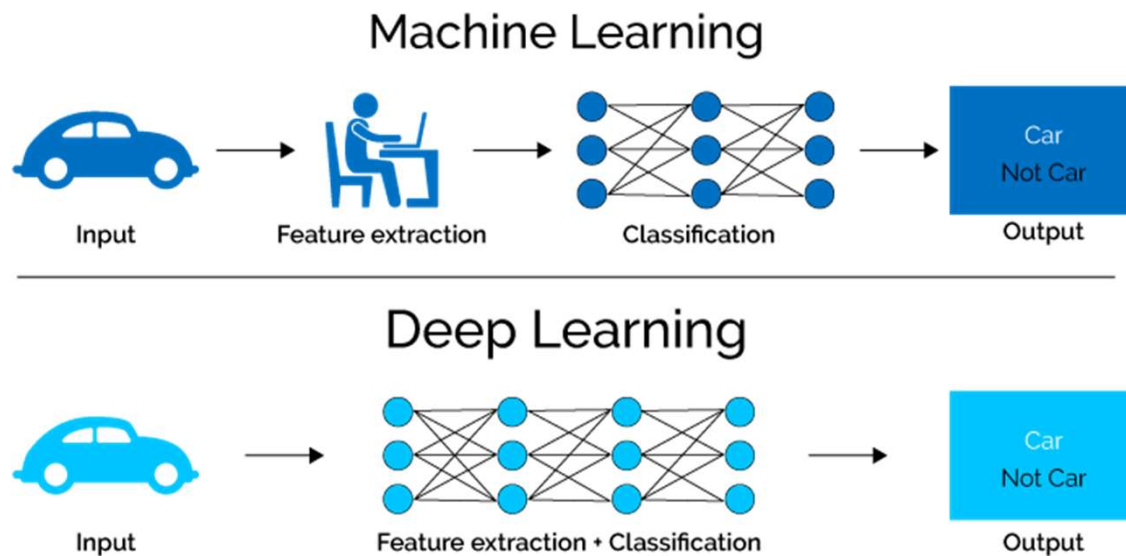
El caso de la XOR se puede resolver añadiendo una capa interna y funciones de activación que introduzcan no linealidades en nuestro modelo.



x_1	x_2	h_1	h_2	y_1
0	0	$0 \cdot 1 + 0 \cdot 1 + 1 \cdot 0 = 0 \rightarrow 0$	$0 \cdot 1 + 0 \cdot 1 + 1 \cdot (-1) = -1 \rightarrow 0$	$0 \cdot 1 + 0 \cdot (-2) = 0 \rightarrow 0$
0	1	$0 \cdot 1 + 1 \cdot 1 + 1 \cdot 0 = 1 \rightarrow 1$	$0 \cdot 1 + 1 \cdot 1 + 1 \cdot (-1) = 0 \rightarrow 0$	$1 \cdot 1 + 0 \cdot (-2) = 1 \rightarrow 1$
1	0	$1 \cdot 1 + 0 \cdot 1 + 1 \cdot 0 = 1 \rightarrow 1$	$1 \cdot 1 + 0 \cdot 1 + 1 \cdot (-1) = 0 \rightarrow 0$	$1 \cdot 1 + 0 \cdot (-2) = 1 \rightarrow 1$
1	1	$1 \cdot 1 + 1 \cdot 1 + 1 \cdot 0 = 2 \rightarrow 2$	$1 \cdot 1 + 1 \cdot 1 + 1 \cdot (-1) = -1 \rightarrow 1$	$2 \cdot 1 + 1 \cdot (-2) = 0 \rightarrow 0$

Redes neuronales en DL

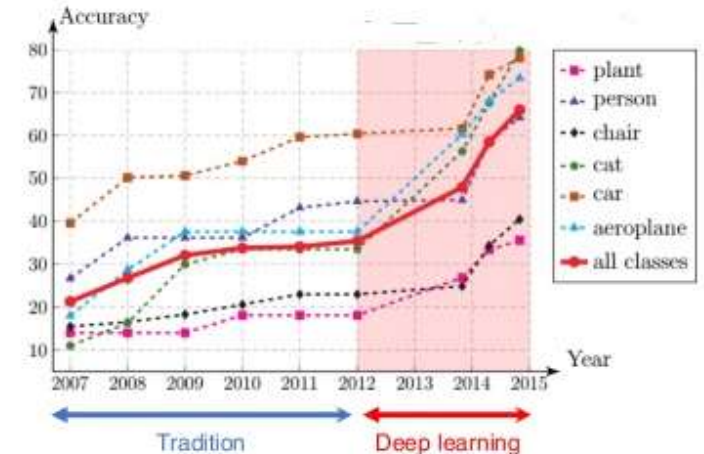
Deep Learning vs Machine Learning



Traditional approach vs. Deep learning

ImageNet: 1.2 million images with 1000 object categories

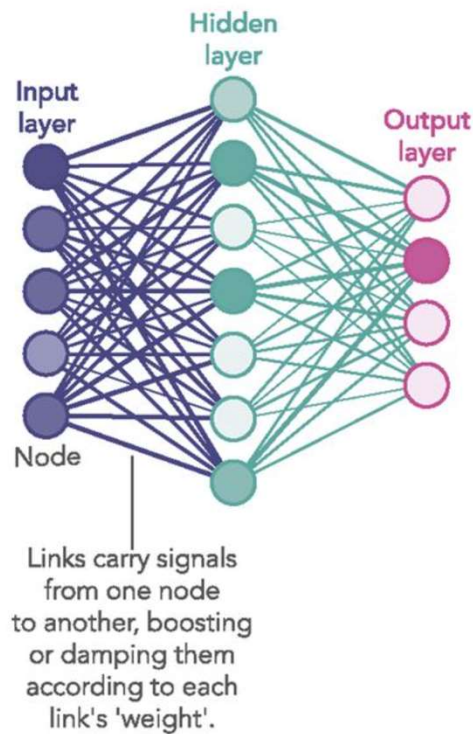
IMAGENET



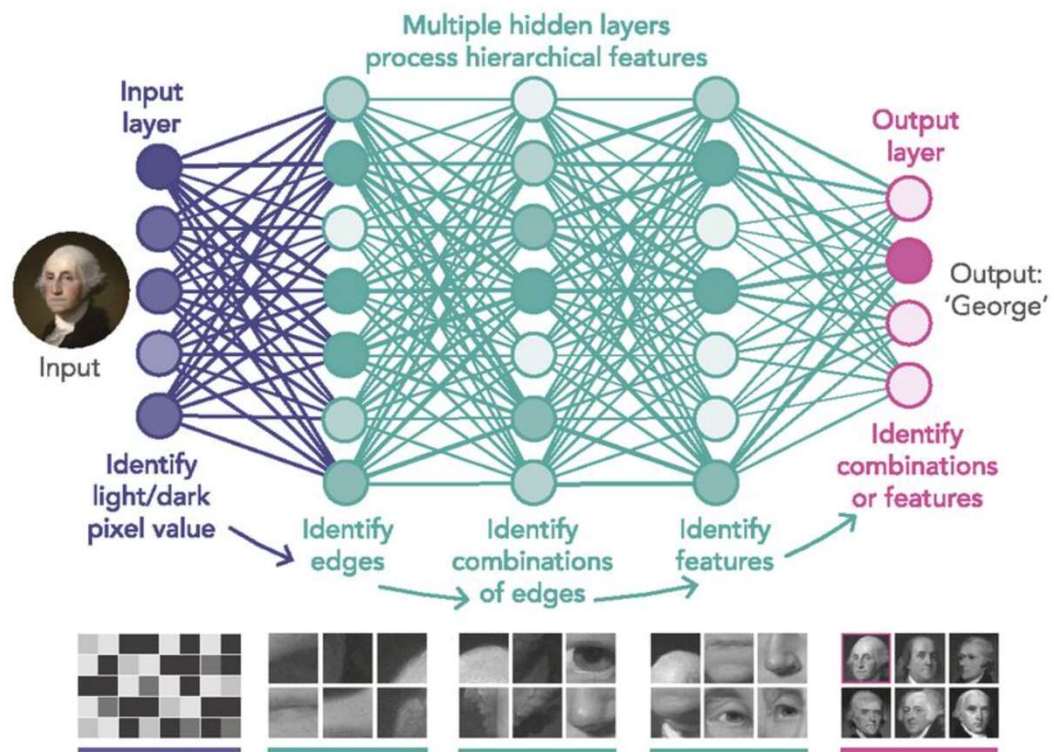
Redes neuronales en DL

¿Por qué “profundas”?

1980S-ERA NEURAL NETWORK



DEEP LEARNING NEURAL NETWORK

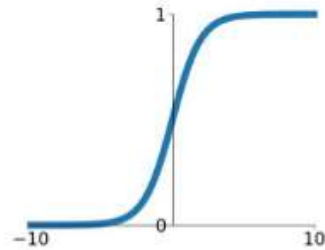


Redes neuronales en DL

Funciones de activación

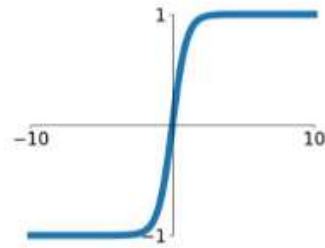
Sigmoid

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



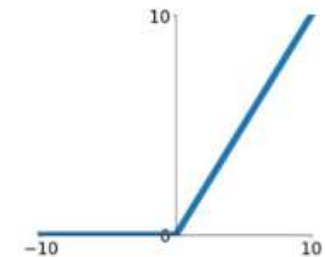
tanh

$$\tanh(x)$$



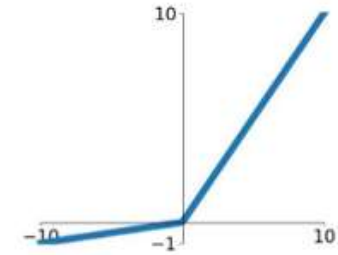
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

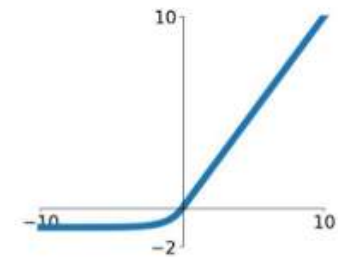


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

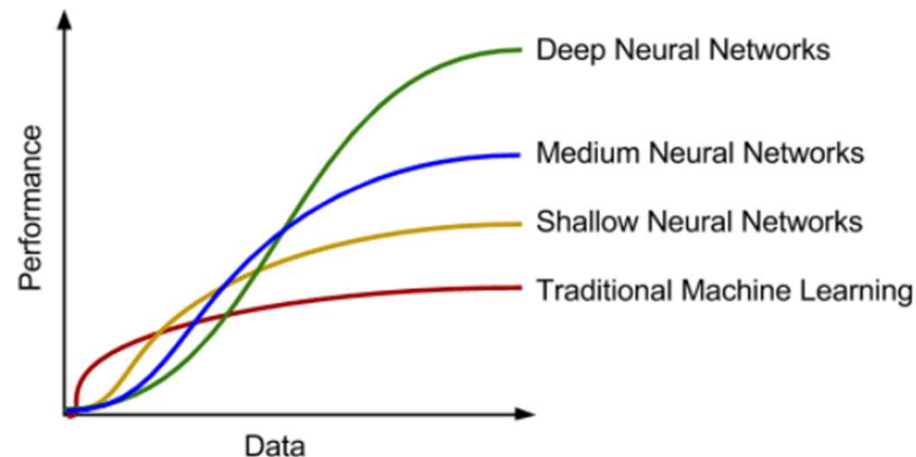
$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Redes neuronales en DL

Complejidad de la red

El número de capas de una red neuronal estará directamente relacionado con la complejidad de los datos y la tarea a realizar.



Según Lippmann¹, un MLP con 2 capas ocultas (más la de entrada y la de salida) es suficiente para crear una "región" de clasificación de cualquier forma deseada.

Según el **Teorema de aproximación universal**^{2,3}: un MLP con una capa oculta es capaz de aproximar cualquier función (pero... ¿cuantos nodos necesitará?).

1. Lippmann in the 1987 paper "[An introduction to computing with neural nets](#)"
2. Deep Learning book (<https://www.deeplearningbook.org>)
3. <https://ai.stackexchange.com/questions/5493/what-is-the-purpose-of-an-activation-function-in-neural-networks>

Redes neuronales en DL

¿Cómo escoger correctamente la función de activación y la complejidad?

- El mejor método consiste en emplear **arquitecturas pre-definidas** con mucha investigación detrás, que han probado ser eficientes para un amplio rango de problemas.
- A partir de estas arquitecturas, podemos probar **modificaciones** que adapten la red a nuestro problema, de forma que consigamos mejores resultados.
- No hay reglas escritas. ¿Entonces qué hacemos?
- Experimentar
- Usar la intuición
- Ante la duda, utilizar redes más profundas con activación de tipo ReLU o similar
- Coger prestadas ideas (basarnos en arquitecturas pre-definidas o trabajos científicos existentes)
- Network Architecture Search (NAS): random search, grid search, heurística, etc.

“[Neural Smithing](#)”, Section 4.4. Capacity versus size”.

Redes neuronales en DL

Intuiciones para decidir la función de activación

- **Sigmoide** y **tanh** no son las más eficientes debido al problema del desvanecimiento del gradiente
- **Softplus** y **Softsign** se deben dejar de lado en favor de ReLU
- **ReLU** debe ser la elección favorita para las capas ocultas. Si sufrimos del problema de “dying relu” podemos utilizar alguna modificación como la **leaky ReLU**, ELU, SELU, etc.
- Para redes muy profundas, **Swish** funciona mejor que ReLU.
- Para la capa final:
 - si estamos trabajando con regresión, utilizar activación **lineal**
 - Si estamos trabajando con clasificación, utilizar activación **sigmoide** o **softmax** (binaria o multiclase)

Redes neuronales en DL

Intuiciones para decidir el número de capas y neuronas

Table: Determining the Number of Hidden Layers

Num Hidden Layers	Result
none	Only capable of representing linear separable functions or decisions.
1	Can approximate any function that contains a continuous mapping from one finite space to another.
2	Can represent an arbitrary decision boundary to arbitrary accuracy with rational activation functions and can approximate any smooth mapping to any accuracy.
>2	Additional layers can learn complex representations (sort of automatic feature engineering) for layer layers.

I have a few rules of thumb that I use to choose hidden layers. There are many rule-of-thumb methods for determining an acceptable number of neurons to use in the hidden layers, such as the following:

- The number of hidden neurons should be between the size of the input layer and the size of the output layer.
- The number of hidden neurons should be $\frac{2}{3}$ the size of the input layer, plus the size of the output layer.
- The number of hidden neurons should be less than twice the size of the input layer.

Redes neuronales en DL

¿Cómo escoger correctamente la función de activación y la complejidad?

Lugares donde inspirarse y aprender más:

- [Activation functions demystified!](#)
- [Everything you need to know about "Activation Functions" in Deep learning models](#)
- [How to Choose an Activation Function for Deep Learning](#)
- [Fundamentals of Deep Learning - Activation Functions and when to use them?](#)
- [A Quick Guide to Activation Functions In Deep Learning](#)

- ["Neural Smithing"](#), Section 4.4. Capacity versus size.
- [Heaton Research: The Number of Hidden Layers](#)
- [Beginners Ask "How Many Hidden Layers/Neurons to Use in Artificial Neural Networks?"](#)
- [How to choose the number of hidden layers and nodes in a feedforward neural network?](#)
- [How to determine the number of layers and nodes of a neural network](#)

Redes neuronales en DL

Resumen

Se trata de un mundo en constante estudio y evolución, donde las únicas técnicas infalibles son la experimentación y posterior validación.

Para empezar: utilizar cosas ya hechas. ¡Os ahorrará mucho tiempo!

Cuando tengáis claros los conceptos básicos y entendáis esas arquitecturas, probad a modificar cosas y estudiar el comportamiento.

Con mucha experiencia y experimentación, conseguiréis una mejor intuición.

Redes neuronales en DL

¡Manos a la obra!

[Notebook 1. Funciones de activación](#)

[Notebook 2. Complejidad de la red](#)