

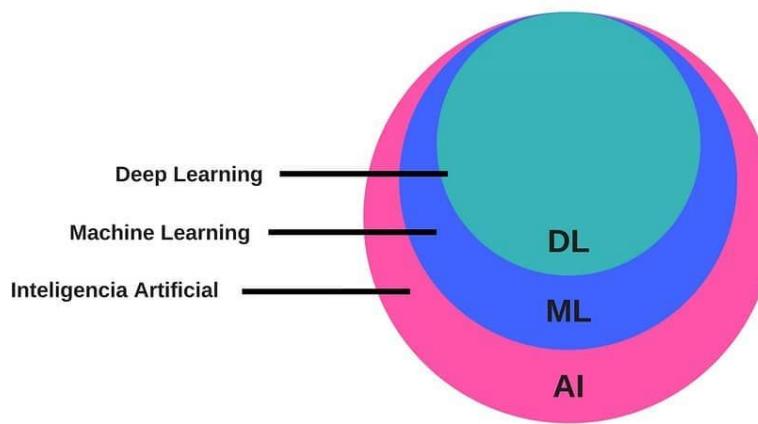
# Introducción a Deep Learning

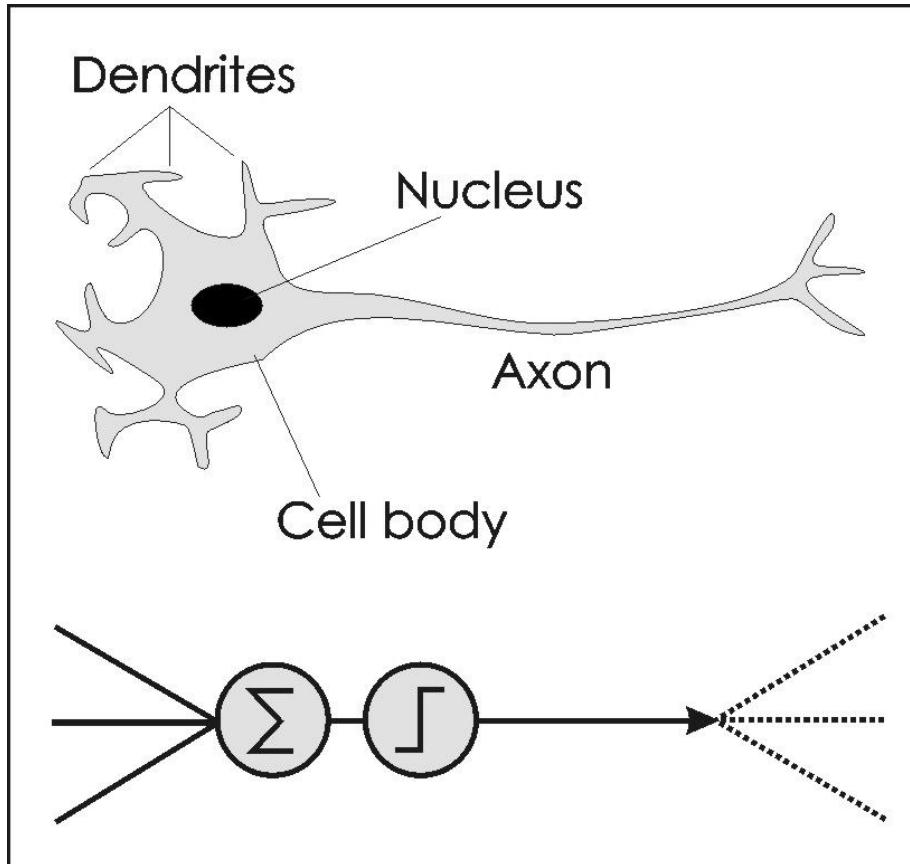
# INTELIGENCIA ARTIFICIAL, MACHINE LEARNING Y DEEP LEARNING

Se puede diferenciar Deep Learning, Machine Learning y la Inteligencia Artificial como un conjunto anidado entre sí, comenzando por el más pequeño.

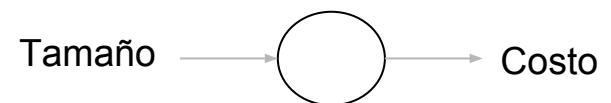
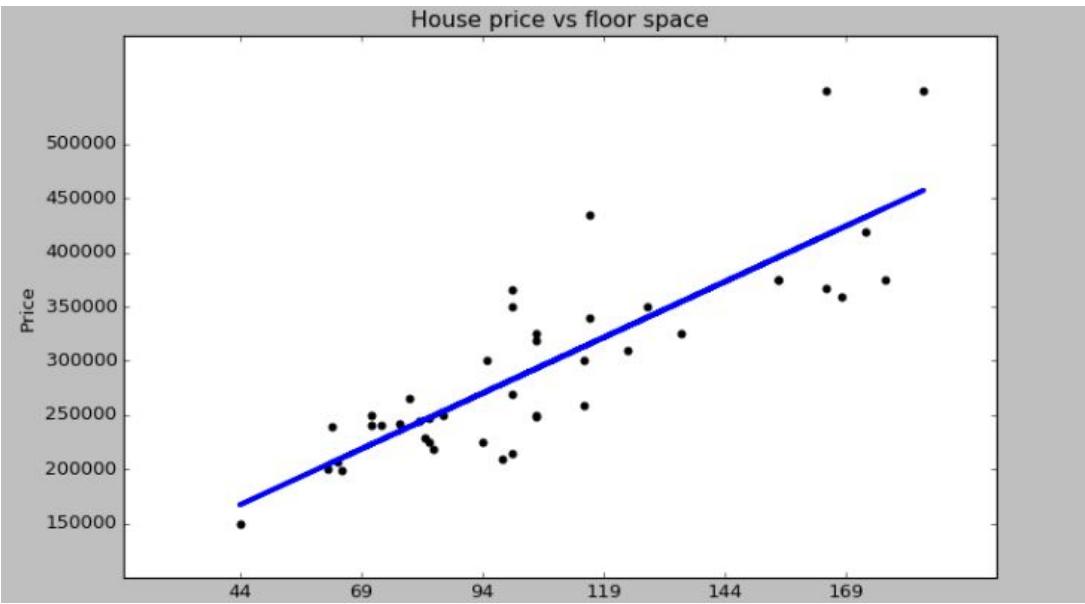
El aprendizaje profundo (DL) es un subconjunto del aprendizaje automático (ML), y el aprendizaje automático es un subconjunto de la inteligencia artificial (AI), que es un término genérico para cualquier programa informático que "actúe" inteligente.

En otras palabras, todo aprendizaje automático es IA, pero no todas las IA son aprendizaje automático, y así sucesivamente...

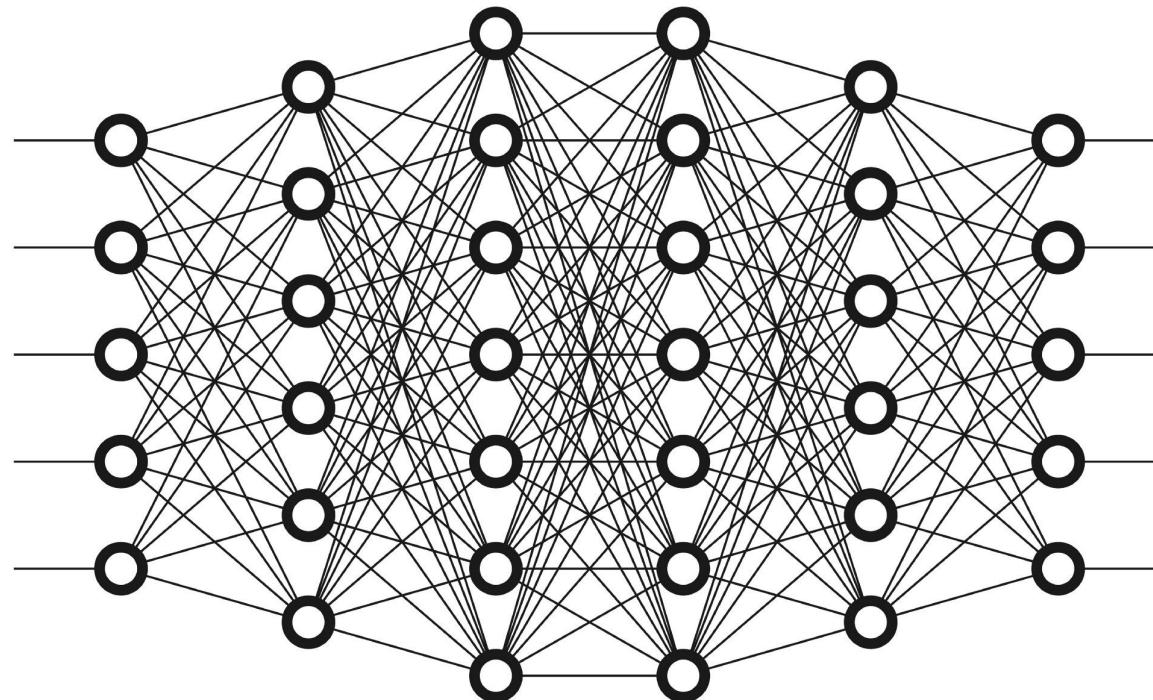




# Predicción de costos de casa



# Deep Learning

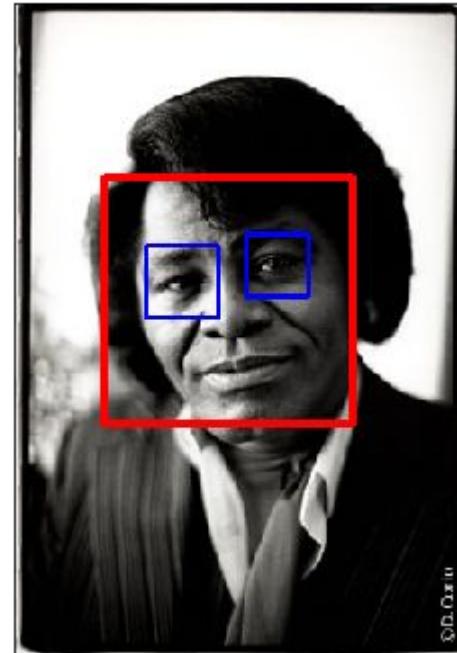


# Reconocimiento de cara

Image with Face Detections



Image with Face and Eye Detection



Downton

# Reconocimiento de cara

Original Image



Blurred face image



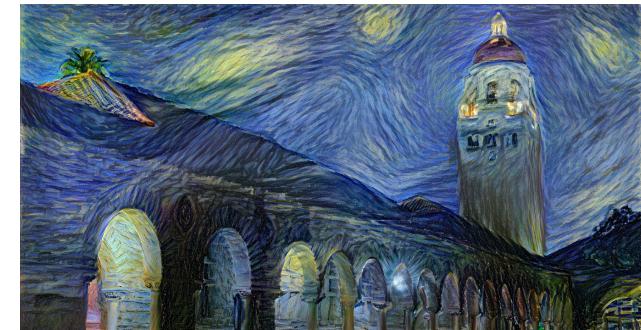
# Neural Style Transfer



+



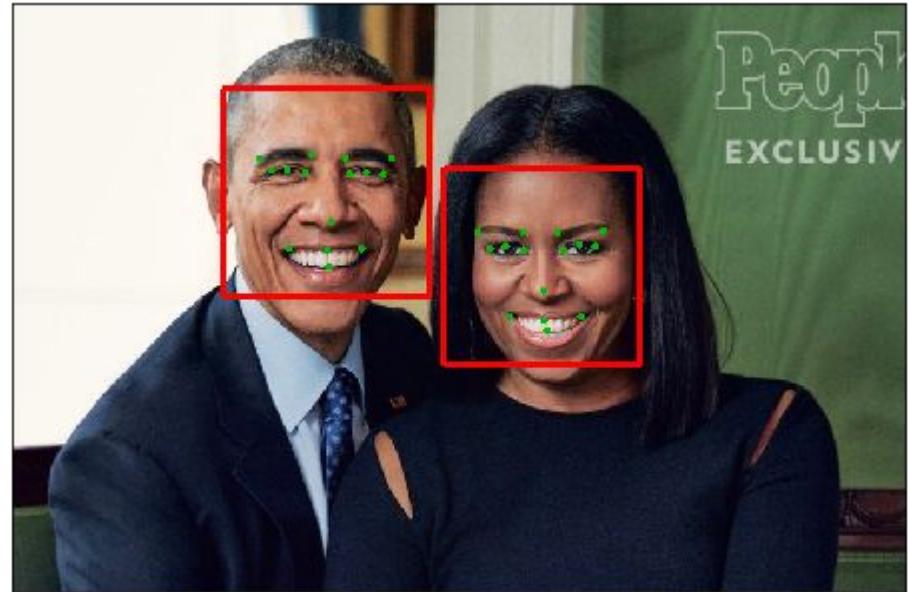
=



# Puntos importantes de la cara



Image with Face Detections and Face Keypoints



# Clasificación (CNN)



That's a dog. Breed: Cocker\_spaniel



That's a human, but it looks like a Cane\_corso



That's a human, but it looks like a Nova\_scotia\_duck\_tolling\_retriever

# ImageNet (CNN)

truck (truck)



truck (truck)



automobile (automobile)



airplane (airplane)

frog (frog)



automobile (automobile)

deer (deer)



bird (cat)



ship (ship)



deer (deer)



deer (bird)



horse (horse)



frog (cat)



airplane (airplane)



cat (cat)



frog (horse)



dog (dog)



frog (frog)



horse (bird)



horse (horse)



deer (deer)



airplane (airplane)



truck (truck)



ship (truck)

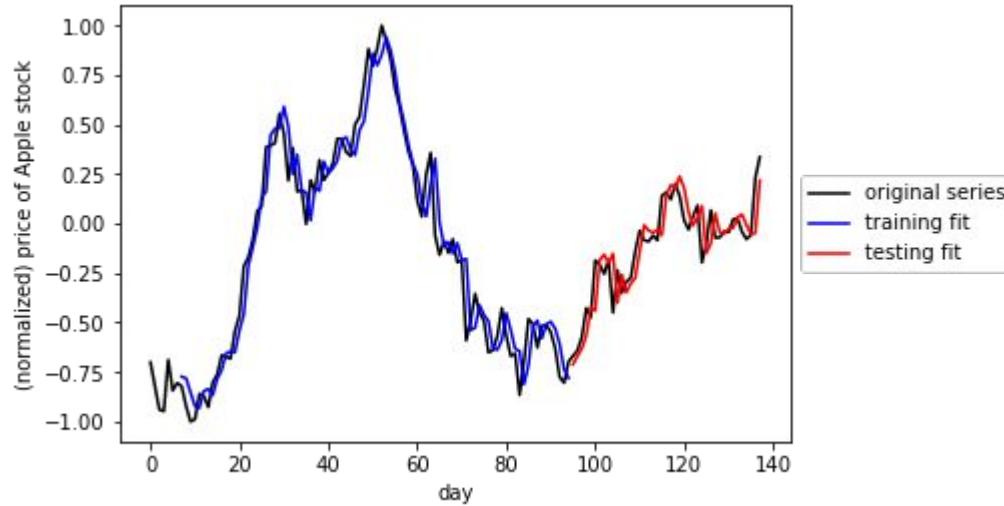


deer (cat)



frog (frog)

# Predictión de bolsas (RNNs)



# Reconocimiento de emociones



Mimic Me!



Score: 3 / 7

**Start** **Stop** **Reset**

## INSTRUCTIONS

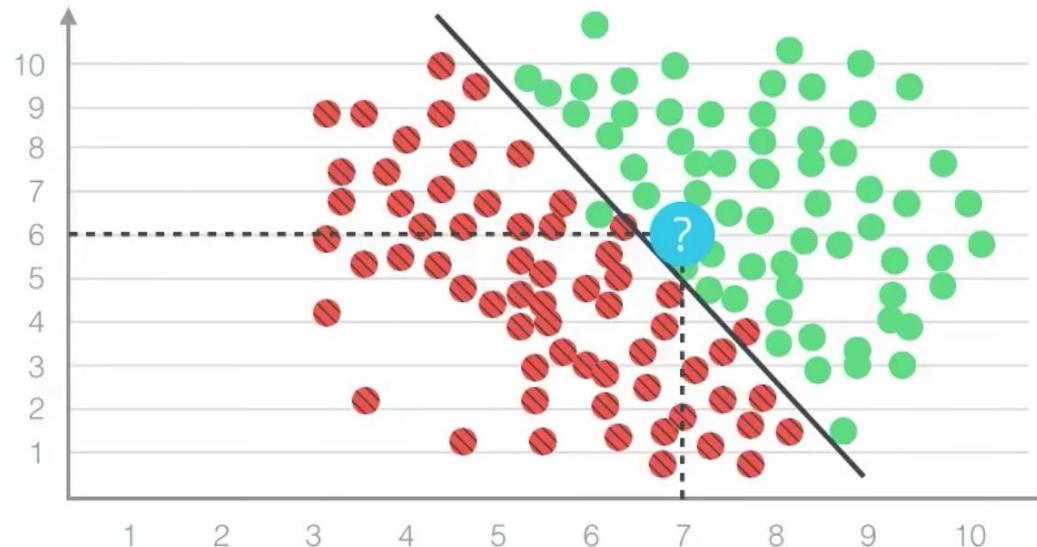
- Press **Start** to initialize the detector.
- Your current emoji will be shown.
- Mimic each emoji being displayed.
- Press **Stop** to end the detector.
- Watch the tracking results and more information.

# Aplicaciones en una universidad

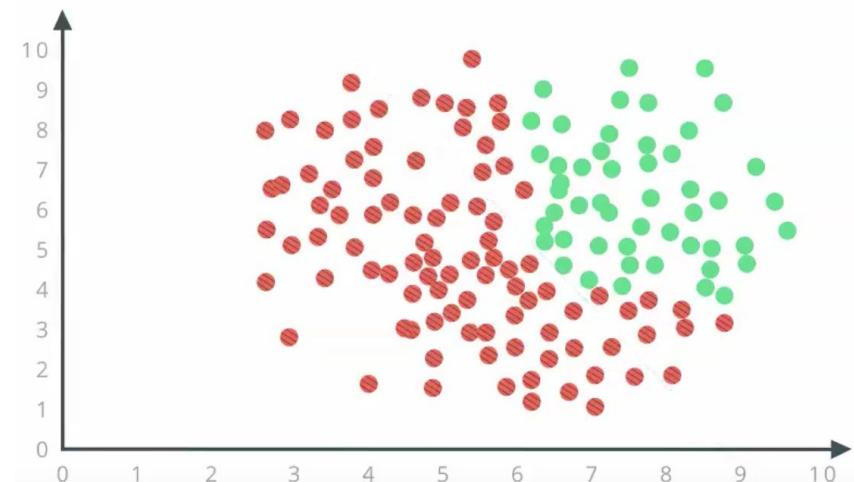
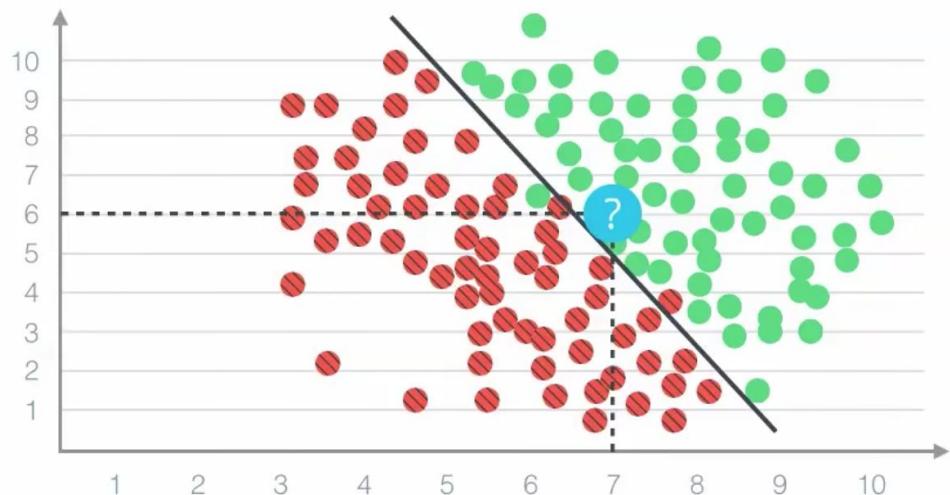
Features

	Promedio Colegio	Examen Admisión
A	10	9
B	8	9
C	6	6

# ¿Cómo sería en la vida real?

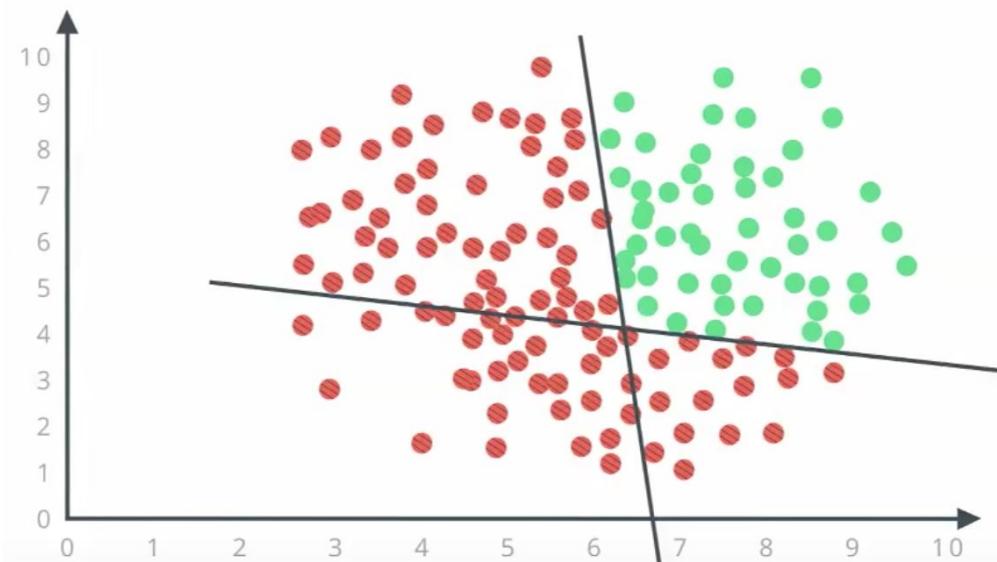


# Los problemas no suelen ser tan sencillos



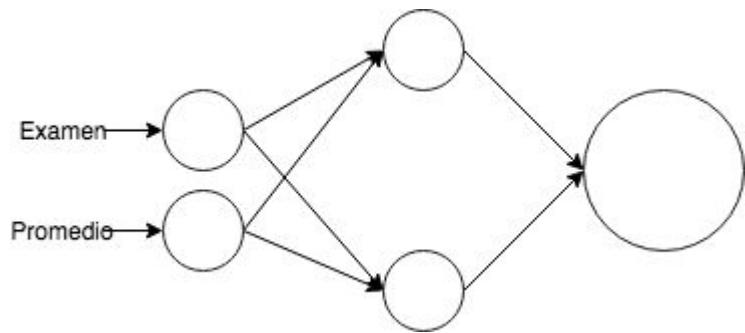
# Redes neuronales

- ¿El punto está arriba de la primera línea?
- ¿El punto está a la derecha de la otra línea?
- ¿La respuesta a las otras dos preguntas fue sí?



# Redes neuronales

- ¿El punto está arriba de la primera línea?
- ¿El punto está a la derecha de la otra línea?
- ¿La respuesta a las otras dos preguntas fue sí?



# Feature

- Propiedad individual que se puede medir.
- Feature engineering.

From: cheapsales@buystufffromme.com  
To: ang@cs.stanford.edu  
Subject: Buy now!

Deal of the week! Buy now!  
Rolex w4tchs - \$100  
Med1cine (any kind) - \$50  
Also low cost M0rgages available.

Spam

From: Alfred Ng  
To: ang@cs.stanford.edu  
Subject: Christmas dates?

Hey Andrew,  
Was talking to Mom about plans for Xmas. When do you get off work. Meet Dec 22?  
Alf

Non-spam

# ¿Qué harías?

- Tienes un día.
- Tienes que elegir 100 de 20,000 aplicantes.
- ¿Qué tal si hay más features?
  - Colegio al que fue
  - Calificación de cada materia de manera independiente
  - Actividades extracurriculares
- Tienes los datos de años previos.

# Información de entrenamiento (training set)

Promedio Colegio	Examen Admisión	Aceptado
9.8	8.9	1
8.7	9.2	1
6.3	7.2	0
7.5	9.3	1
9.8	7.2	0
8.8	8.4	1
8.3	8.9	1
9.5	9.4	1
7.3	6	0

# Features

- ¿Todos son igual de importantes?
- Podemos asignar un peso (W) a cada entrada/feature (X).

$$V = W_1 X_1 * W_2 X_2$$

- X1 calificación del colegio.
- W1 peso de la calificación del colegio.
- X2 calificación del examen.
- W2 peso de la calificación del examen.

$$\sum_{i=1}^m w_i x_i$$

# Ejemplo

- Si queremos que el examen sea más importante, podemos asignar pesos como los siguientes

$$V = 0.5X_1 + 2X_2$$

	Promedio Colegio	Examen Admisión	Valor
A	10	9	23
B	8	9	22
C	6	6	18

- Nosotros decidimos a partir de qué valor pasará el alumno (20?) - **bias**

# Perceptrón

- Se asigna un límite.
- Si el valor de la neurona es más alto que el límite, la neurona se enciende.

$$\text{output} = \begin{cases} 0 & \text{if } \sum_j w_j x_j \leq \text{threshold} \\ 1 & \text{if } \sum_j w_j x_j > \text{threshold} \end{cases}$$

- En nuestro caso, el threshold sería 20

# Perceptrón

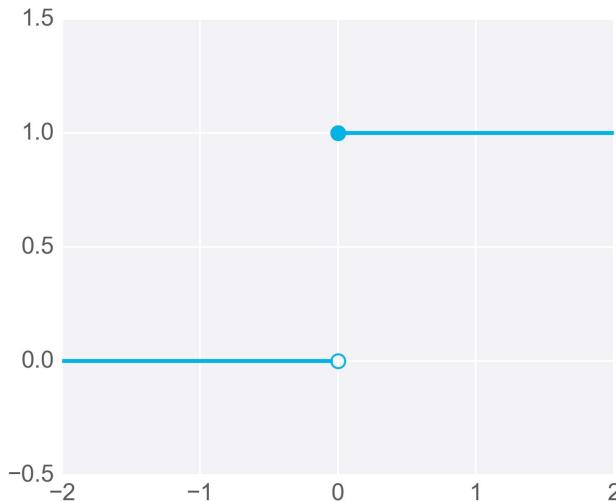
- Se asigna un bias.
- Si el valor de la neurona es más alto que el -bias, la neurona se enciende.

$$\text{output} = \begin{cases} 0 & \text{if } w \cdot x + b \leq 0 \\ 1 & \text{if } w \cdot x + b > 0 \end{cases}$$

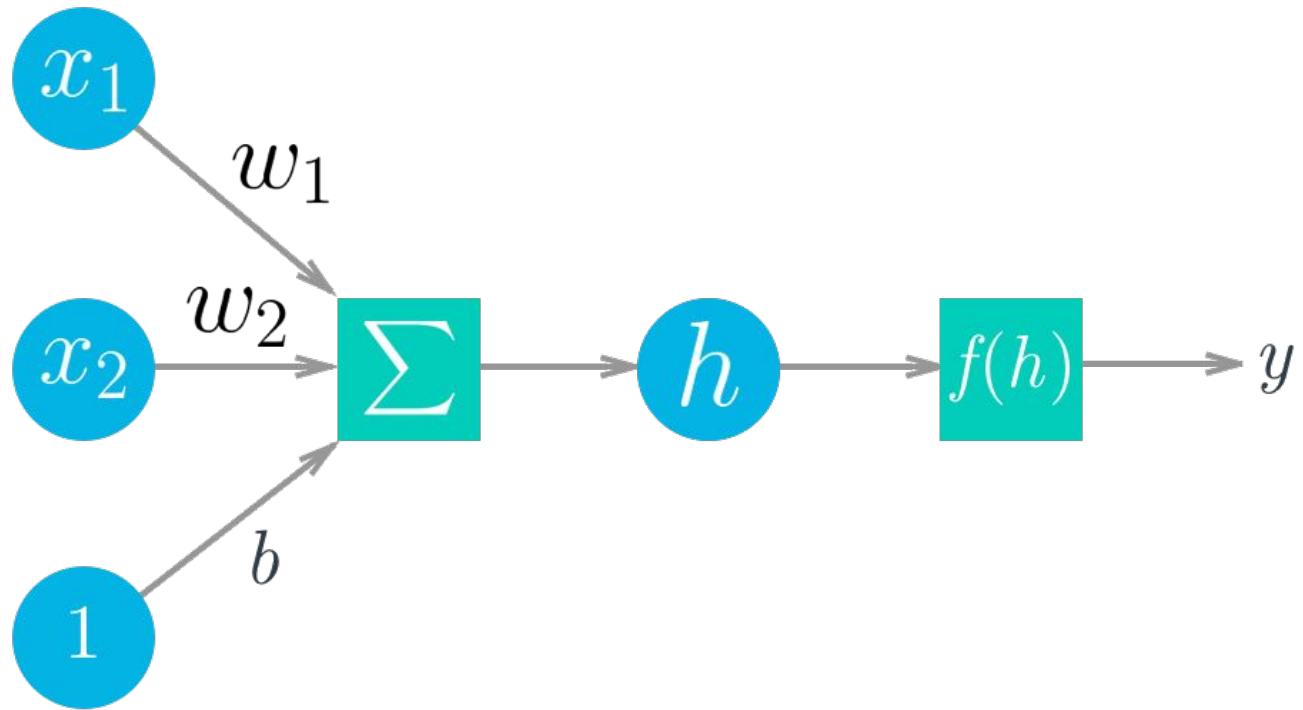
- En nuestro caso, b sería -20

# Perceptrón

- Aprende a través de ejemplos.
- Pesos ( $w$ ) y bias ( $b$ ) se van ajustando viendo los ejemplos
- Su **función de activación** es una función escalón

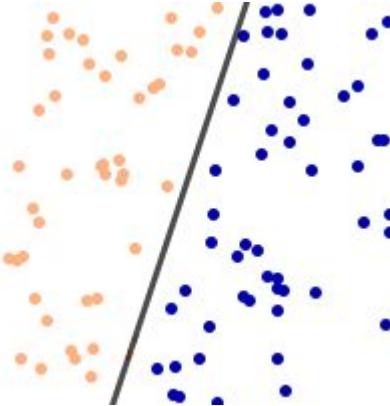


$$f(h) = \begin{cases} 0 & \text{if } h < 0 \\ 1 & \text{if } h \geq 0 \end{cases}$$



# Deep Learning

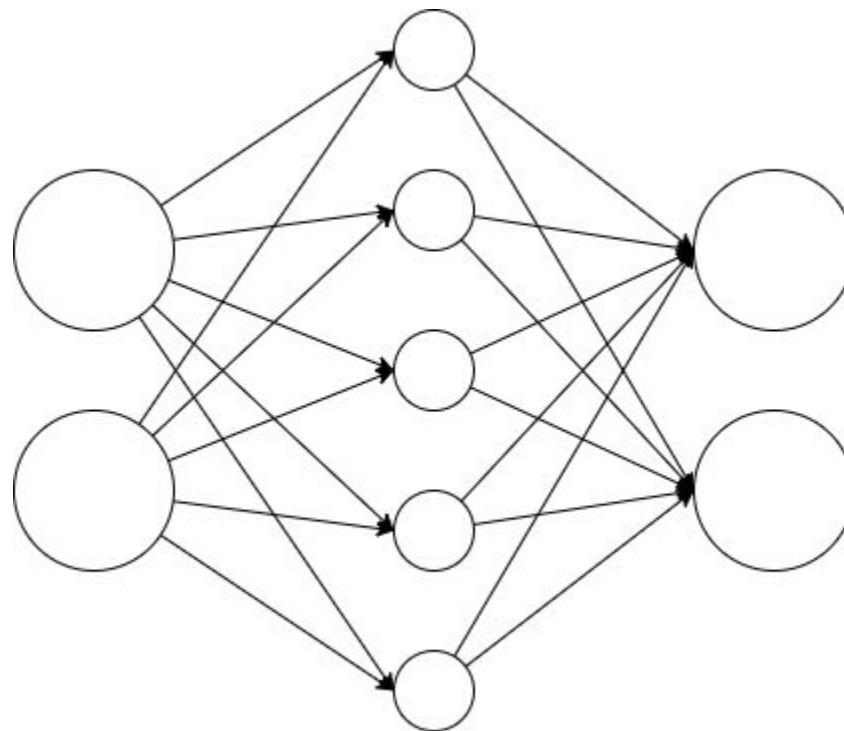
- Determina qué features son importantes y cuáles no sirven.
- A partir de muchos ejemplos, va determinando qué pesos importan y cuáles no importan.
- Muchos, muchos ejemplos.
- [Ejemplo 1](#)
- [Ejemplo 2](#)



Input Layer

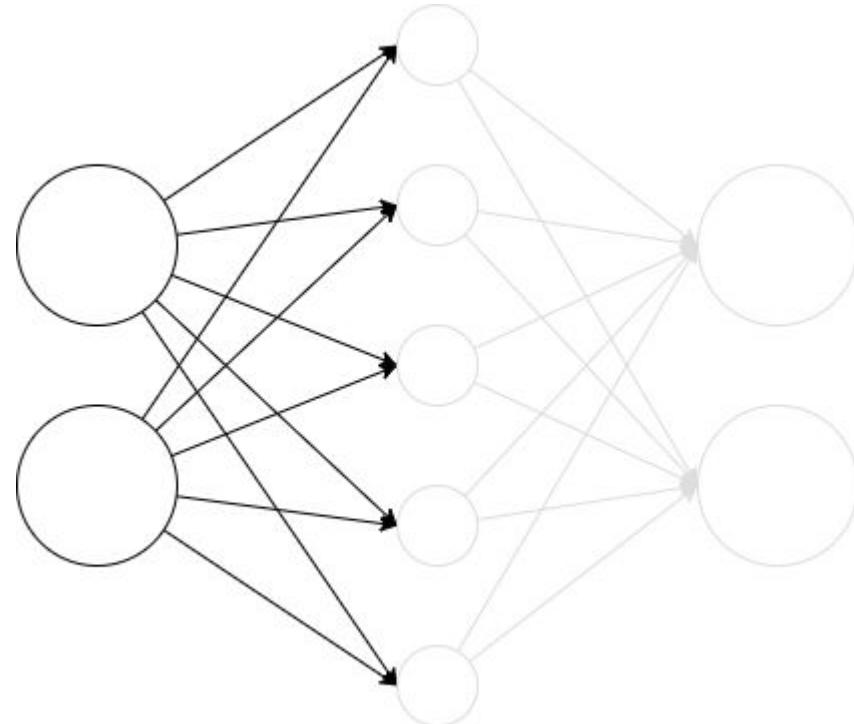
Hidden Layers

Output  
Layer



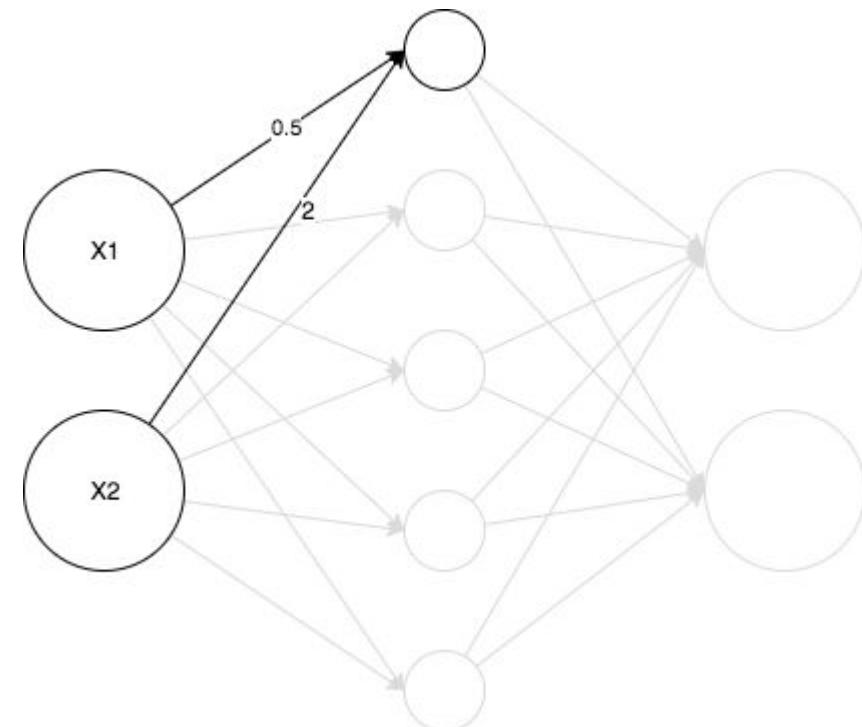
# Input Layer (Capa de entrada)

- Un nodo para cada entrada o feature.
- En este caso:
  - Calificación del colegio.
  - Calificación del examen.

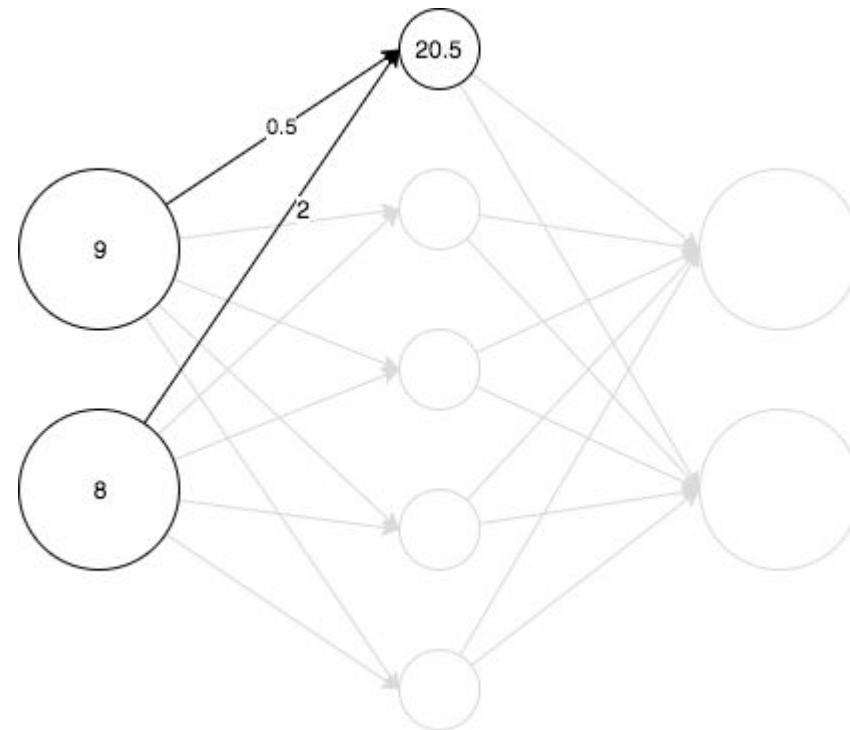


# Hidden Layer

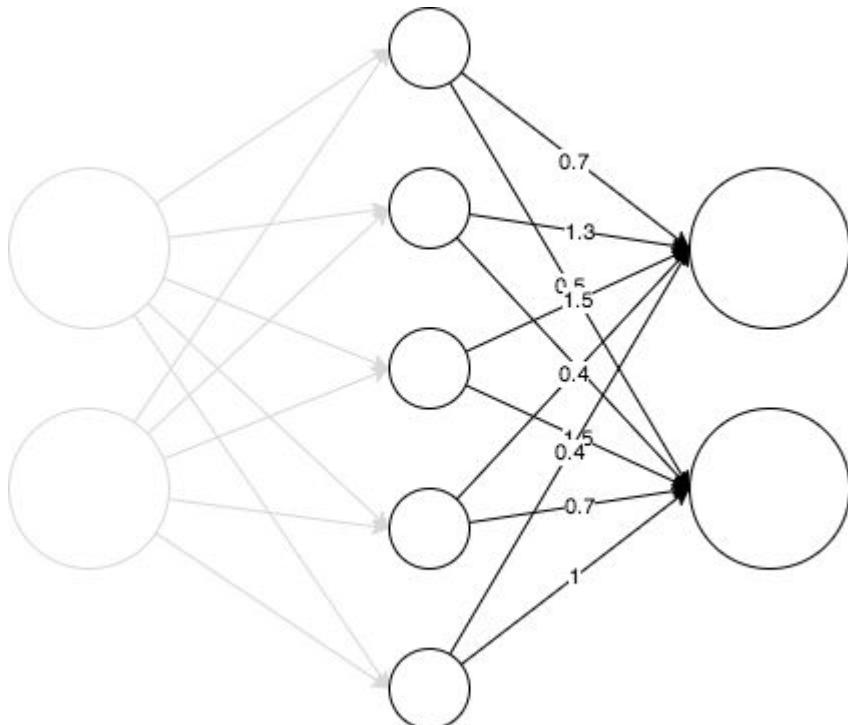
- Los pesos se ajustarán automáticamente en el entrenamiento.
- Tener más capas y más nodos permite entender comportamientos más complejos



# Ejemplo



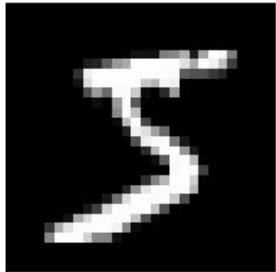
# Output Layer



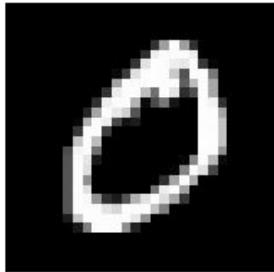
- Este es un problema de clasificación binaria
  - Dos resultados: aceptado o no aceptado.
- Tendremos dos neuronas de salida, cada una representando uno de estos casos.
- La neurona que se encienda más fuerte será la elegida.

# Dígitos escritos a mano (MNIST)

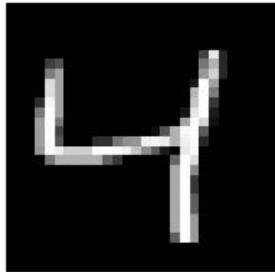
5



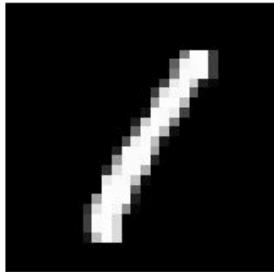
0



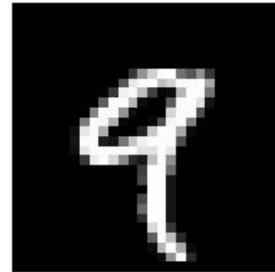
4



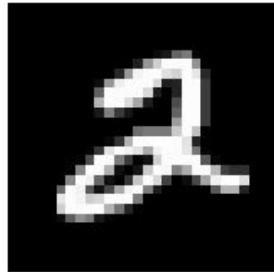
1



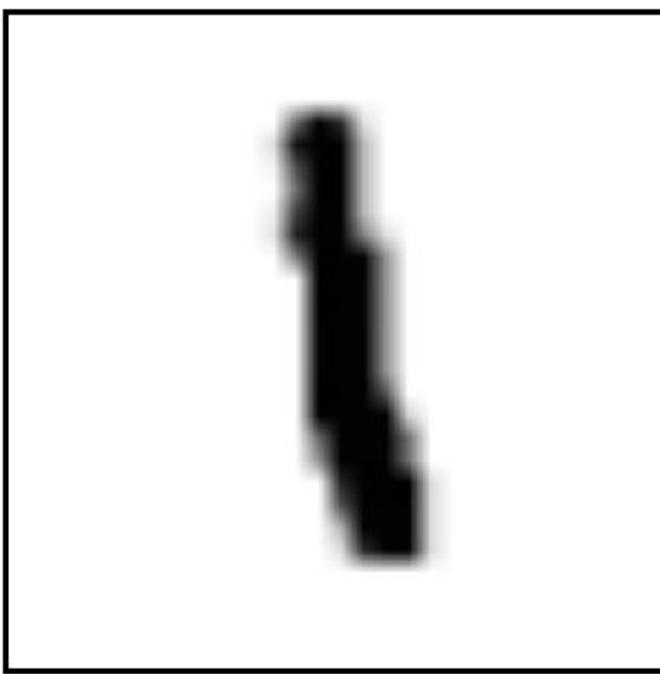
9



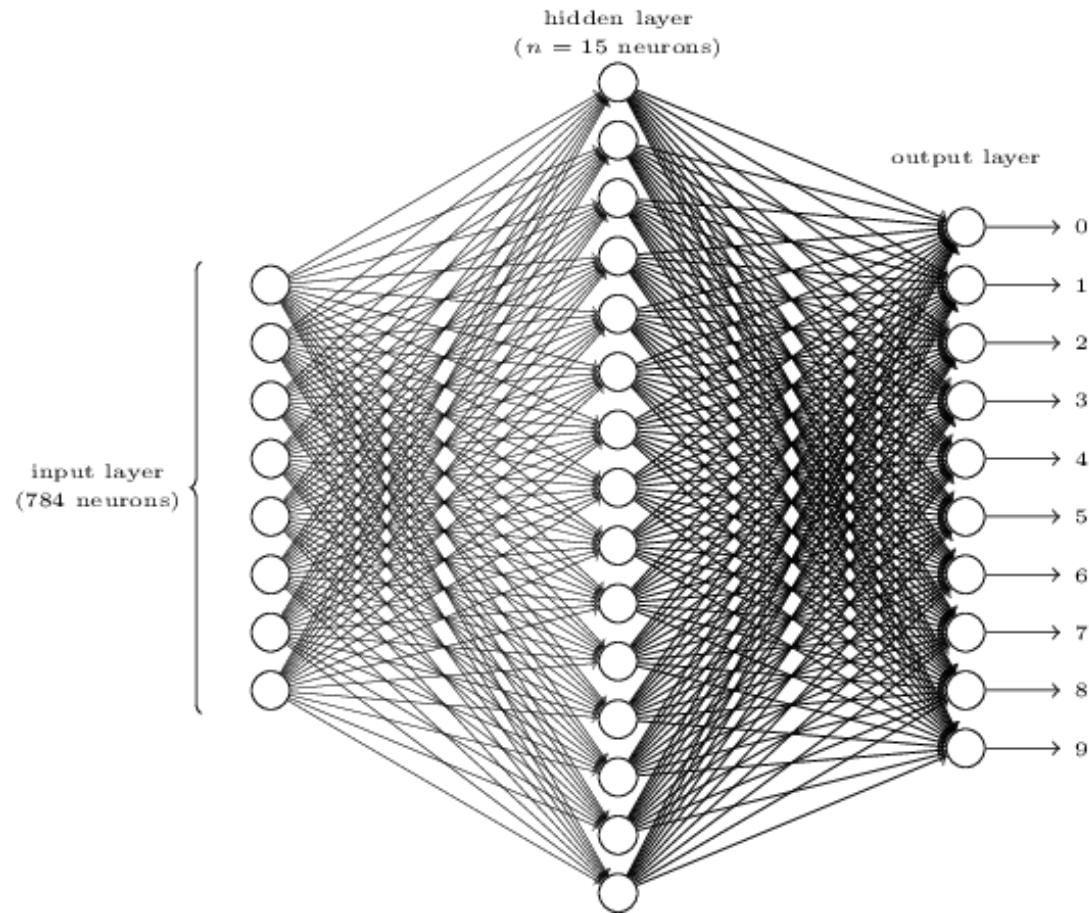
2



# Features (748 pixels)



2



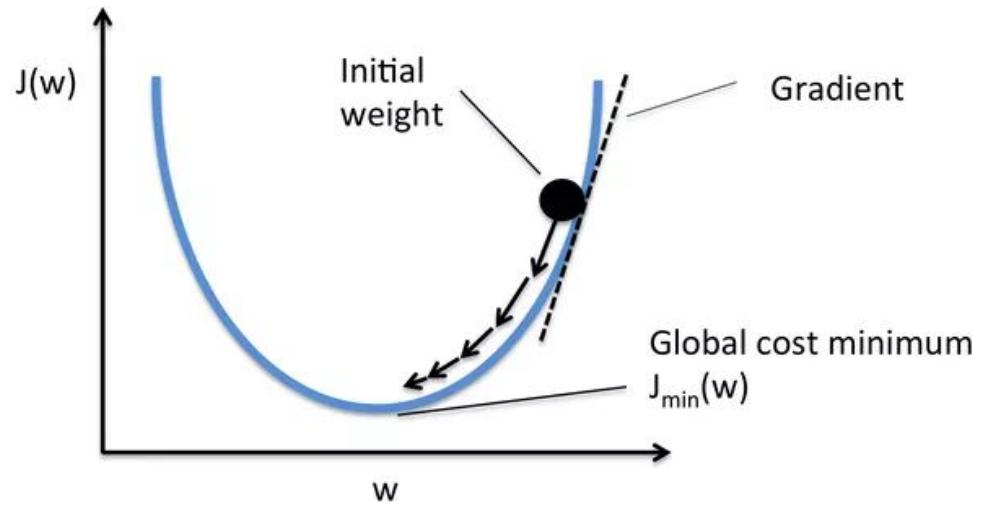
# One-hot encoding

	admit	gre	gpa	rank
0	0	380	3.61	3
1	1	660	3.67	3
2	1	800	4.00	1
3	1	640	3.19	4
4	0	520	2.93	4
5	1	760	3.00	2
6	1	560	2.98	1
7	0	400	3.08	2
8	1	540	3.39	3
9	0	700	3.92	2

	rank_1	rank_2	rank_3	rank_4
0	0	0	1	0
1	0	0	1	0
2	1	0	0	0
3	0	0	0	1
4	0	0	0	1

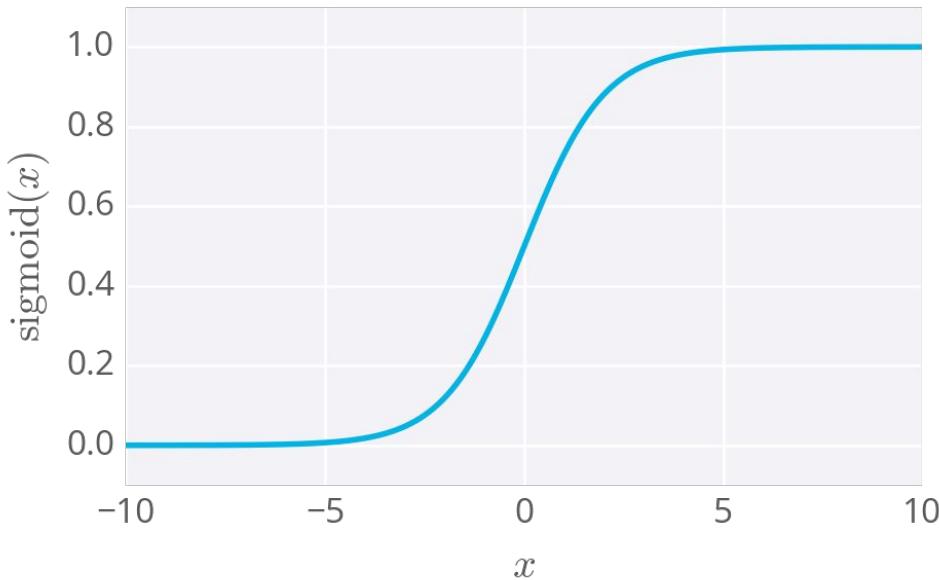
# Optimizadores

- SGD - Stochastic Gradient Descent
  - Momentum
  - Learning Rate Decay
- RMSProp
- Adagrad
- Adam
- Nadam



# Función de activación

- Aquí usamos una función de activación sencilla, pero realmente puede **cualquier** función (derivable).

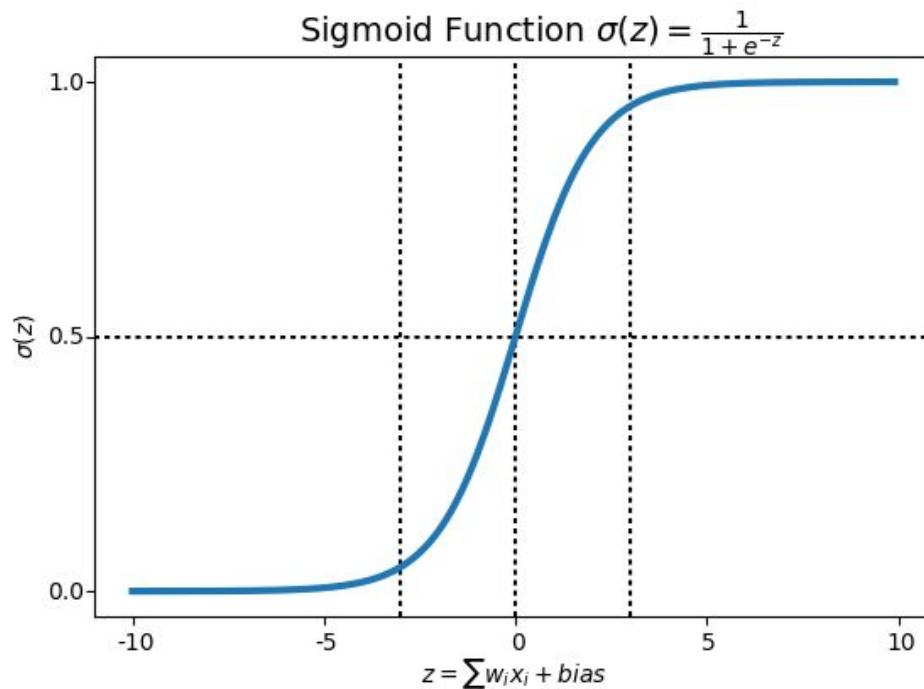


$$S(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}.$$

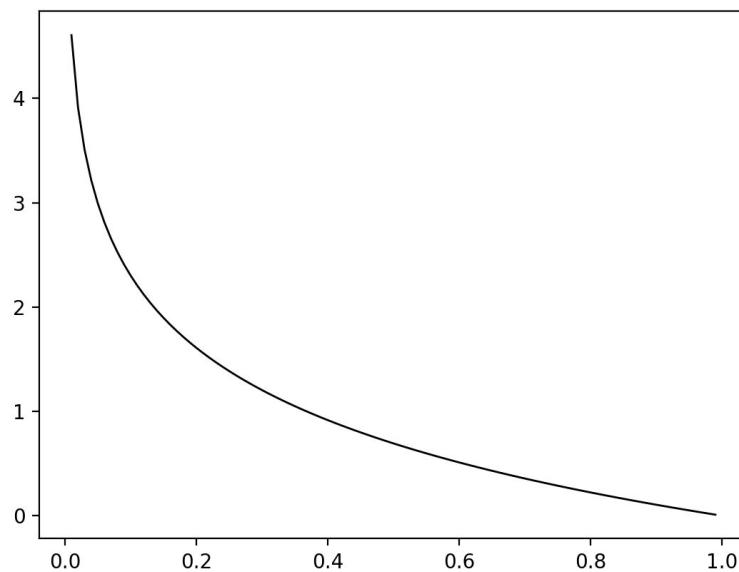
Su output puede ser cualquier número entre 0 y 1.

# Funciones de Activación

- Sigmoid
- Linear
- Softmax
- ReLU
- Tanh



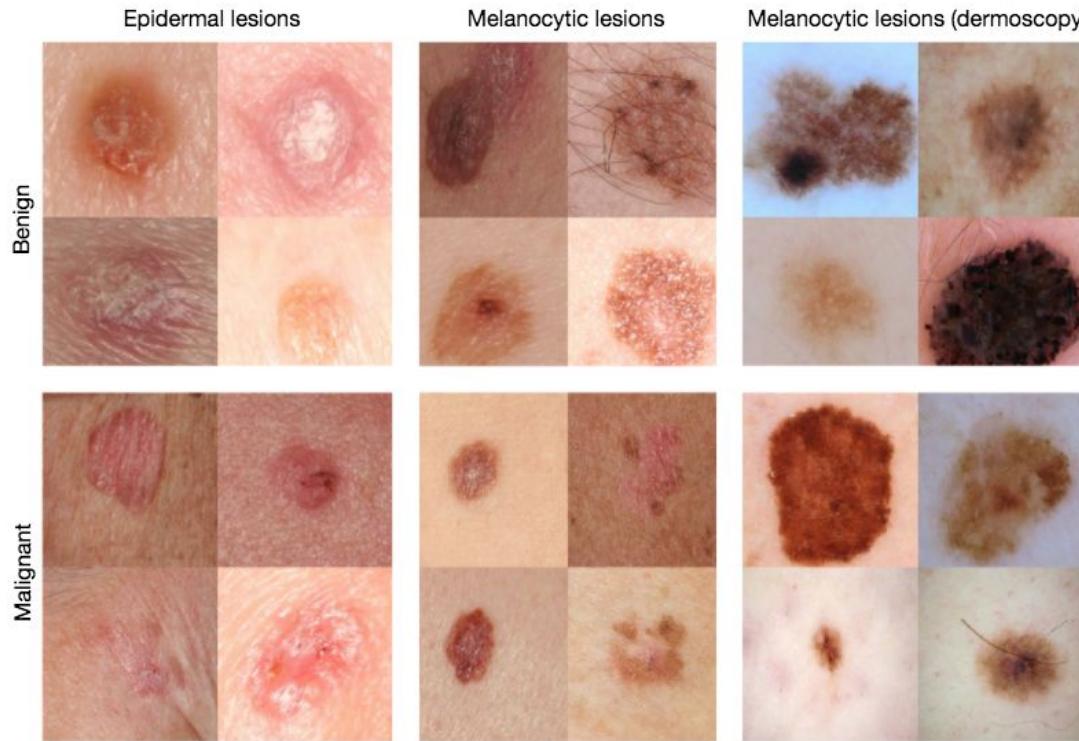
# Loss function



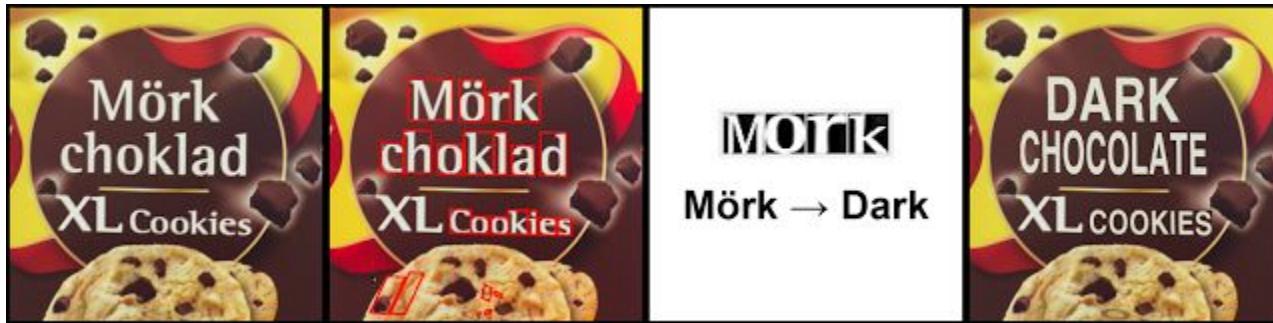
# Hyperparameters

- Input layer.
- Hidden layers.
- Activation function.
- Optimizer (gradient descent).
- Loss function (cross entropy).
- Learning rate.
- Epochs.
- Validation size.
- Batch size.

# Detección de cáncer de piel



# Traducir texto en imágenes



# Mejora de calidad de imágenes



# Subtitulado de imágenes



"man in black shirt is playing guitar."



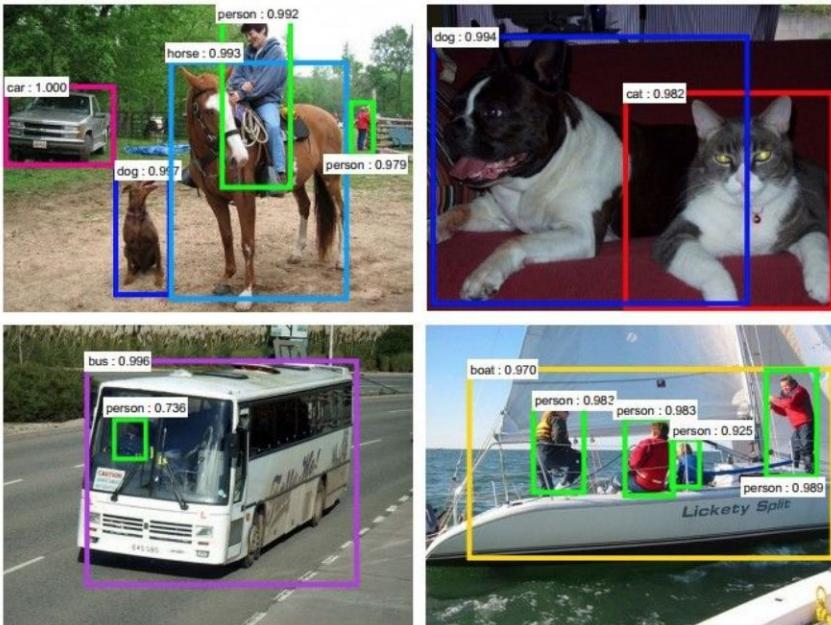
"construction worker in orange safety vest is working on road."



"two young girls are playing with lego toy."

# Detección y segmentación

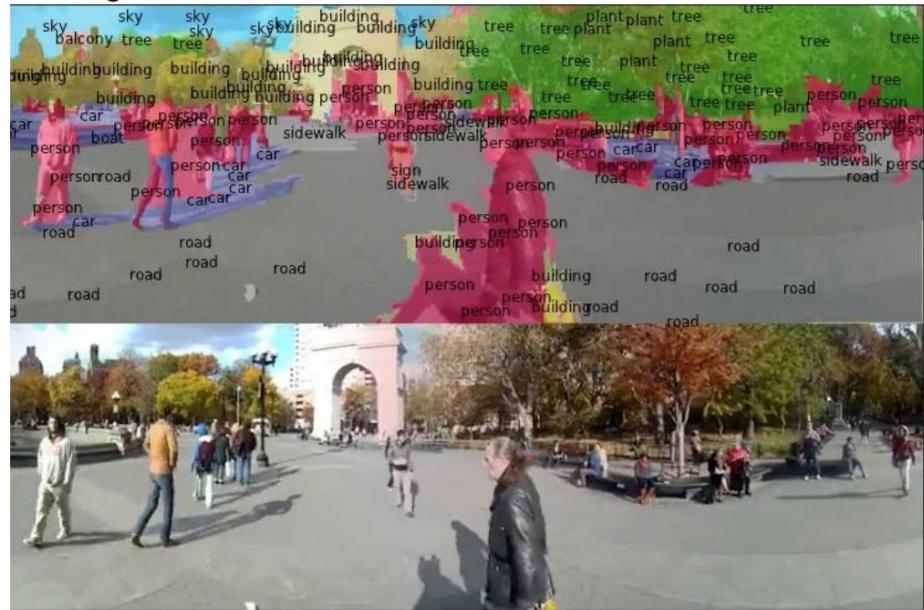
## Detection



Figures copyright Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, 2015. Reproduced with permission.

[Faster R-CNN: Ren, He, Girshick, Sun 2015]

## Segmentation

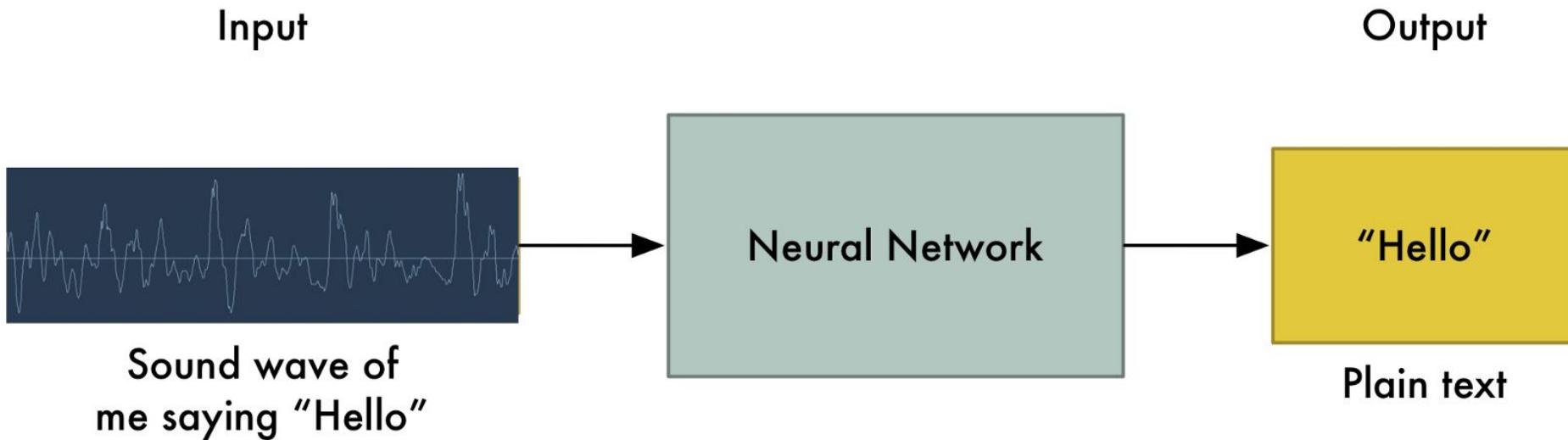


Figures copyright Clement Farabet, 2012.

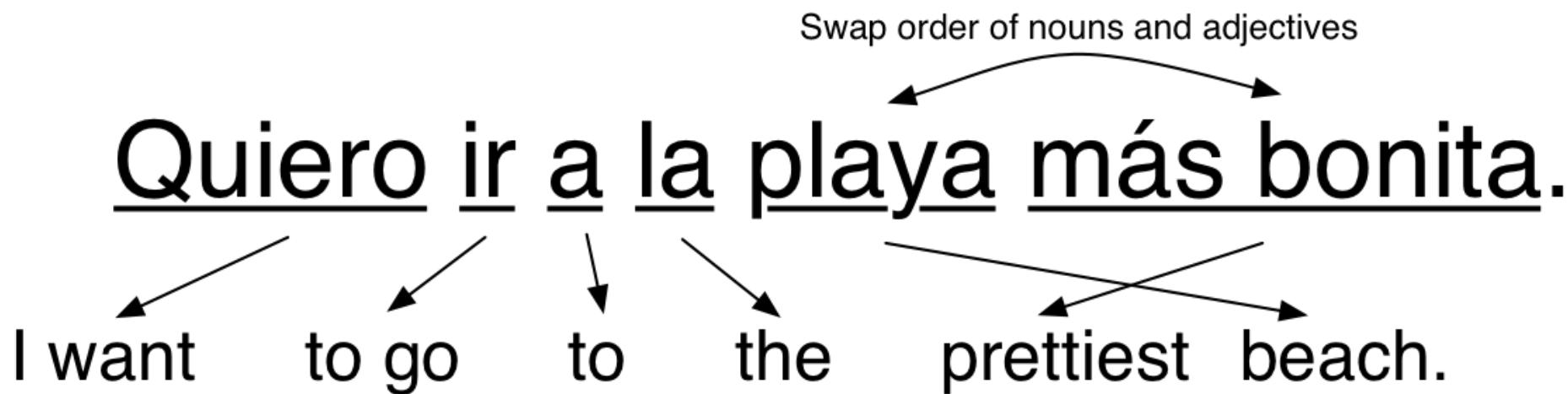
Reproduced with permission.

[Farabet et al., 2012]

# Speech Recognition



# Machine Translation



# Conclusión

- Nosotros vamos a definir cuántas neuronas hay y cómo están conectadas.
- Mientras más capas de neuronas, la red puede comprender información más compleja (Deep Learning).
- Las conexiones tienen un peso.
- Las neuronas ocultas tienen un bias.
- Los pesos y bias se asignan aleatoriamente y se van ajustando con entrenamiento (ejemplos).
- Nosotros definimos cómo las neuronas procesan la información (funciones de activación)
- **Una red neuronal puede aprender cualquier cosa con una buena arquitectura, suficiente información, y suficiente entrenamiento.**