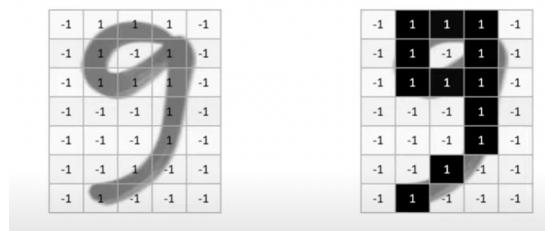


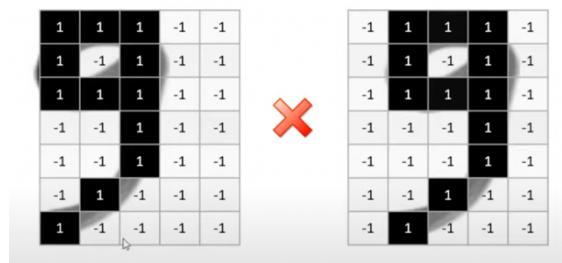
1 Convolutional Neural Network

1.1 Introducción

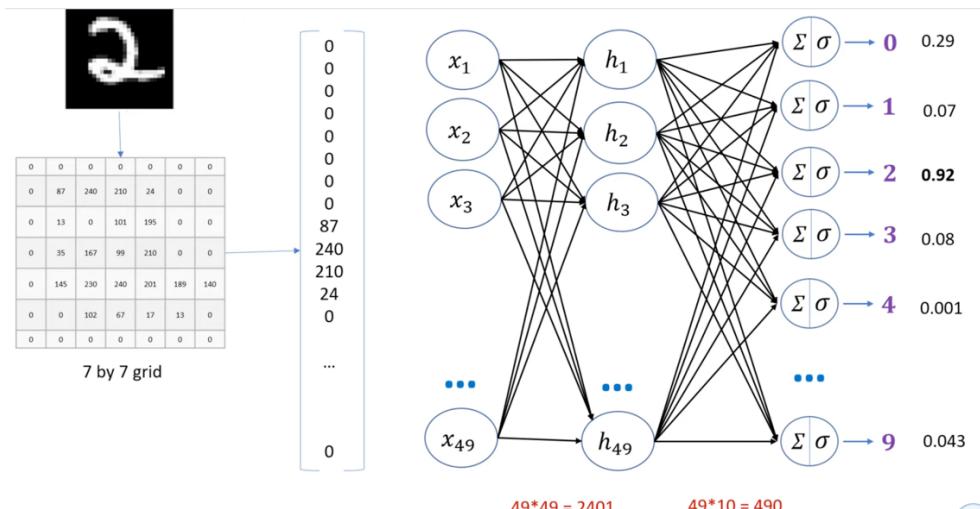
Imaginamos que queremos realizar una clasificación de dígitos numéricos como lo hemos hecho hasta ahora, con machine learning. Debemos tener muestras de cada número para poder identificarlo:



Pero puede suceder que el número esté desplazado, que tenga diferentes maneras de representación, etc. En estos casos el aprendizaje automático no nos soluciona el problema.



Para estos casos en los que queremos tener la generalización, usamos las redes neuronales artificiales. ANN. Por ejemplo para el caso de identificar números almacenados en una matriz 7x7:



Pero ... ¿qué sucede cuando tenemos imágenes mucho más grandes?

Ejemplo: imagen de 1920 x 1080 x 3 (canales)

1^a Capa de neuronas:

1920 x 1080 x 3 ≈ 6 Millones de caract.



Capa oculta de neuronas: Por ejemplo 4 M

Número de pesos entre la capa de input y la oculta: $6M \times 4M = 24 M$ de pesos

Por lo que **no es factible la realización de tantos cálculos para clasificación de imágenes.**

Desventajas del uso de ANN para la clasificación de imágenes:

- Mucho proceso
- Sensibilidad de dónde se encuentra el objeto/animal/persona/.... en la imagen (derecha, izquierda...)

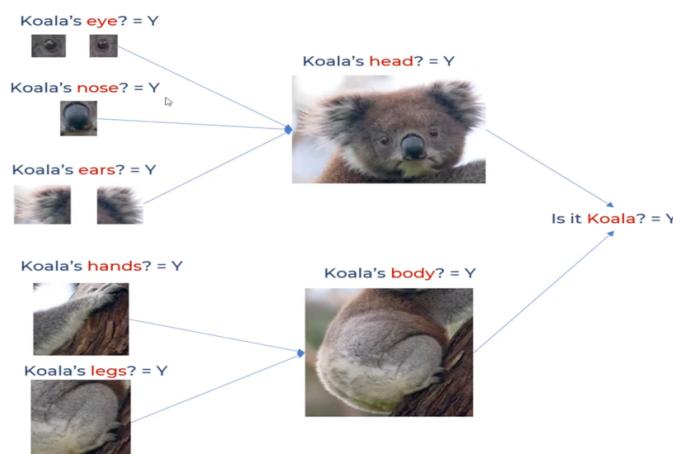
1.2 ¿Cómo detecta nuestro cerebro qué hay en una imagen?

Al ver la imagen (koala p.ej.) nuestro cerebro pregunta:

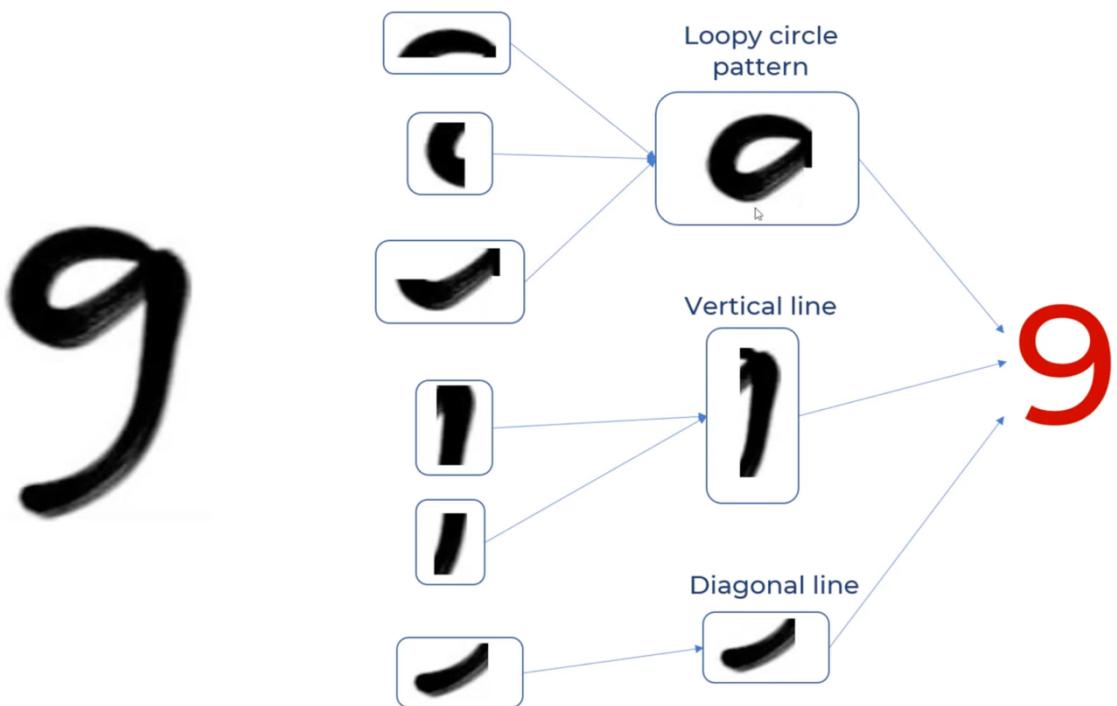
- ¿Tiene ojos de koala? Sí
- ¿Tiene nariz de koala? Sí → Entonces es cara de koala
- ¿Tiene orejas de koala? Sí

Por otro lado, otras neuronas preguntan:

- ¿Tiene pies de koala?
- ¿Tiene manos de koala? → Entonces es el cuerpo de un koala



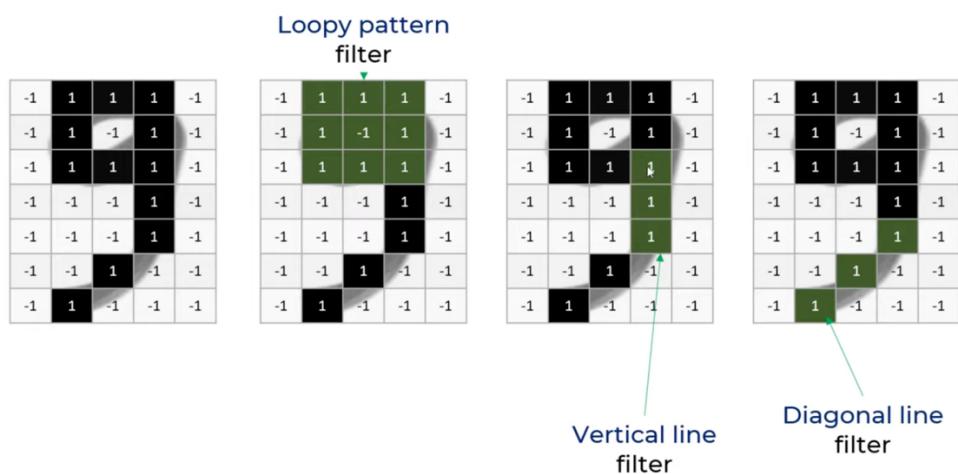
El mismo ejemplo orientado al reconocimiento del número 9 podemos ejemplificarlo de la siguiente manera:



¿Cómo podemos reconocer ciertas características computacionalmente?

En este caso se trabaja con "filtros". Podemos crear filtros para detección de determinadas características y aplicarlos a la imagen para saber si cumple con los filtros o no los cumple.

En el caso de reconocer el 9, podemos aplicar 3 filtros



1.2.1 ¿Cómo se aplican los filtros a las imágenes?

Recordemos la teoría de “convoluciones”. El método es exactamente el mismo.

Basándonos en la imagen del “9” y aplicando el filtro de “parte circular” obtenemos la tabla siguiente:

-1	1	1	1	-1
-1	1	-1	1	-1
-1	1	1	1	-1
-1	-1	-1	1	-1
-1	-1	-1	1	-1
-1	-1	1	-1	-1
-1	1	-1	-1	-1

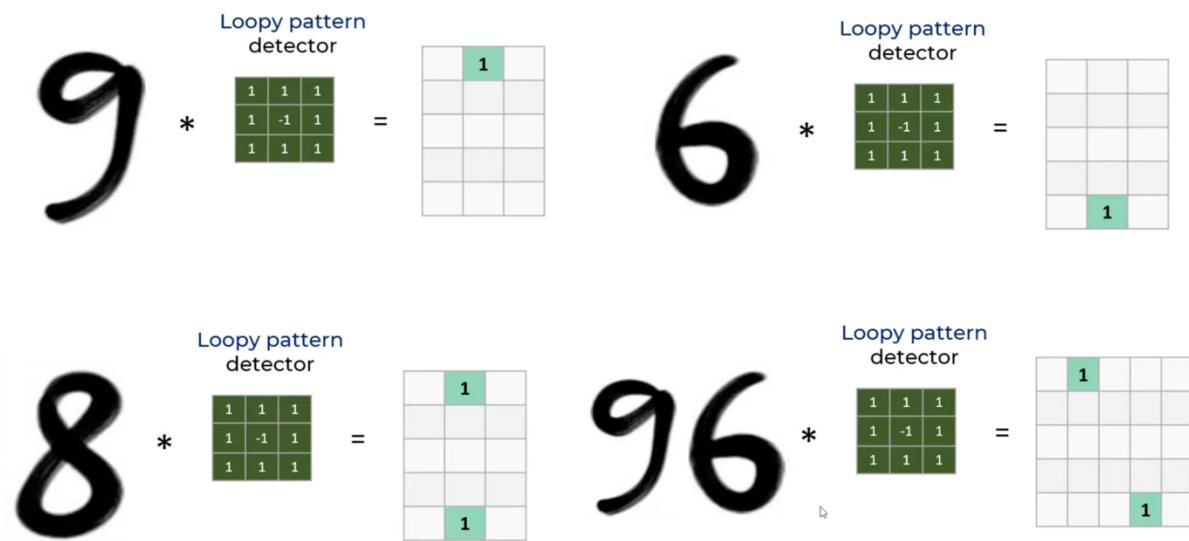
*

1	1	1
1	-1	1
1	1	1

-0.11	1	-0.11
-0.55	0.11	-0.33
-0.33	0.33	-0.33
-0.22	-0.11	-0.22
-0.33	-0.33	-0.33

Feature Map

Aplicándolo el mismo filtro a diferentes números, vemos el resultado:



Volvemos al Koala ...



$$\text{koala} * \begin{matrix} \text{eye} \\ \text{detector} \\ \begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix} \end{matrix} = \begin{matrix} 1 & 1 \end{matrix}$$



$$\text{koala} * \begin{matrix} \text{eye} \\ \text{detector} \\ \begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix} \end{matrix} = \begin{matrix} 1 & 1 \end{matrix}$$



$$\text{koala} * \begin{matrix} \text{hands} \\ \text{detector} \\ \begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix} \end{matrix} = \begin{matrix} 1 \end{matrix}$$

Y con el 9:

$$9 * \begin{matrix} \text{Loopy pattern} \\ \text{detector} \\ \begin{matrix} 1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & 1 \end{matrix} \end{matrix} = \begin{matrix} 1 \end{matrix}$$

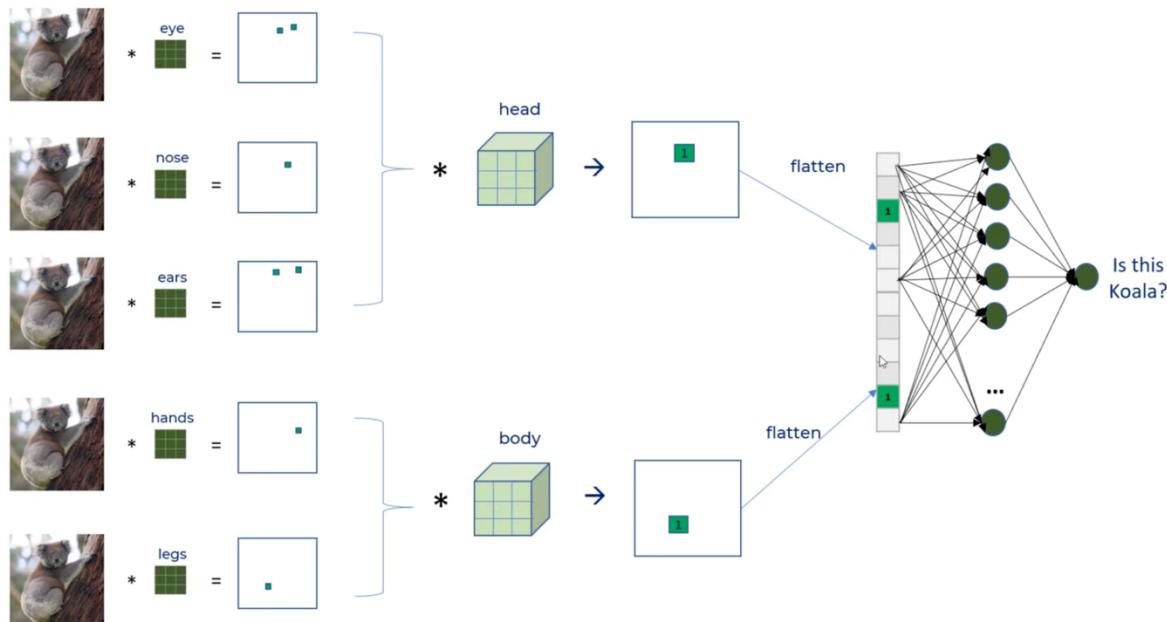
$$9 * \begin{matrix} \text{Vertical line} \\ \text{detector} \\ \begin{matrix} -1 & 1 & -1 \\ -1 & 1 & -1 \\ -1 & 1 & -1 \end{matrix} \end{matrix} = \begin{matrix} 1 \end{matrix}$$

$$9 * \begin{matrix} \text{Diagonal line} \\ \text{detector} \\ \begin{matrix} -1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & -1 \end{matrix} \end{matrix} = \begin{matrix} 1 \end{matrix}$$

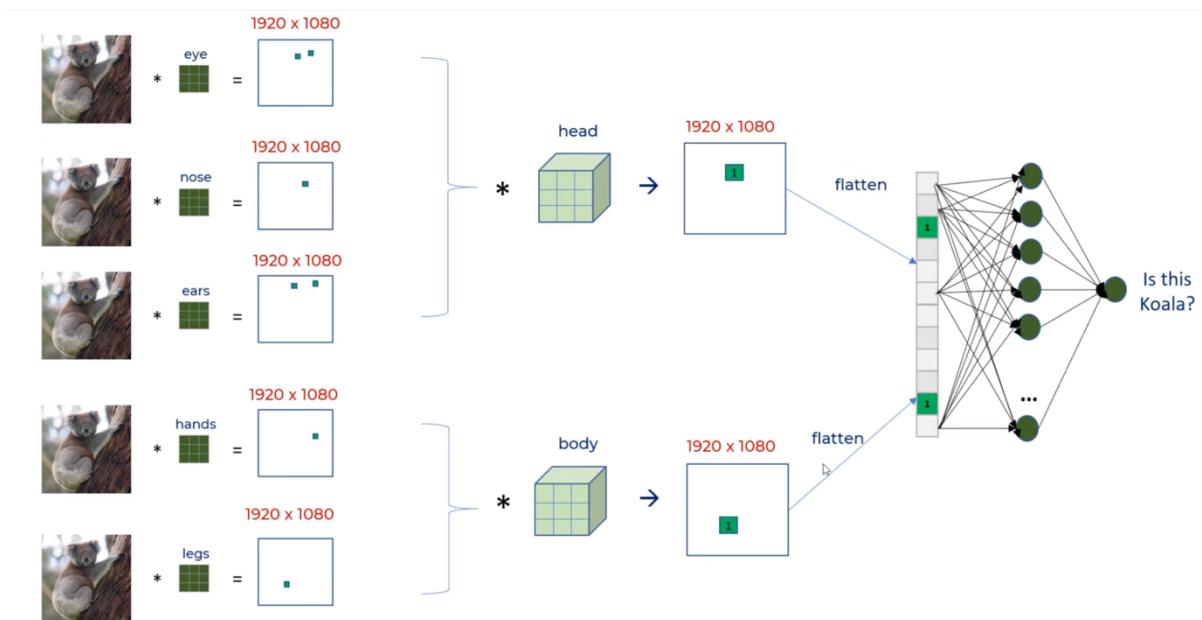
$$9 * \begin{matrix} \text{Filters} \\ \begin{matrix} 1 & 1 & 1 \\ -1 & -1 & 1 \\ -1 & 1 & -1 \\ -1 & 1 & -1 \\ -1 & 1 & -1 \end{matrix} \end{matrix} = \begin{matrix} 1 \\ 1 \end{matrix}$$

03008915 C/ Ferrocarril, 22, 03570 La Vila Joiosa Tel 966870140 Fax 966870141 http://portal.edu.gva.es/iesmarcoszaragoza

Esos mapas de características se unen en un array de 1 dimensión que pasará a ser la entrada de la red neuronal artificial tal como la conocemos hasta ahora.



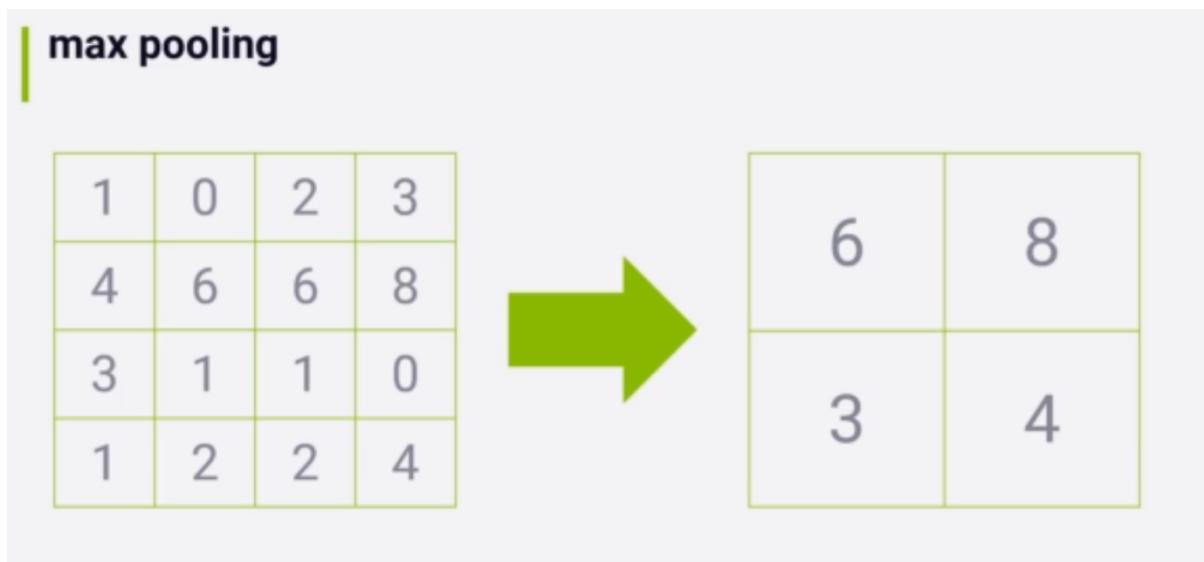
1.3 ¿Cómo solucionar el problema de alta computación?



1.3.1 Pooling

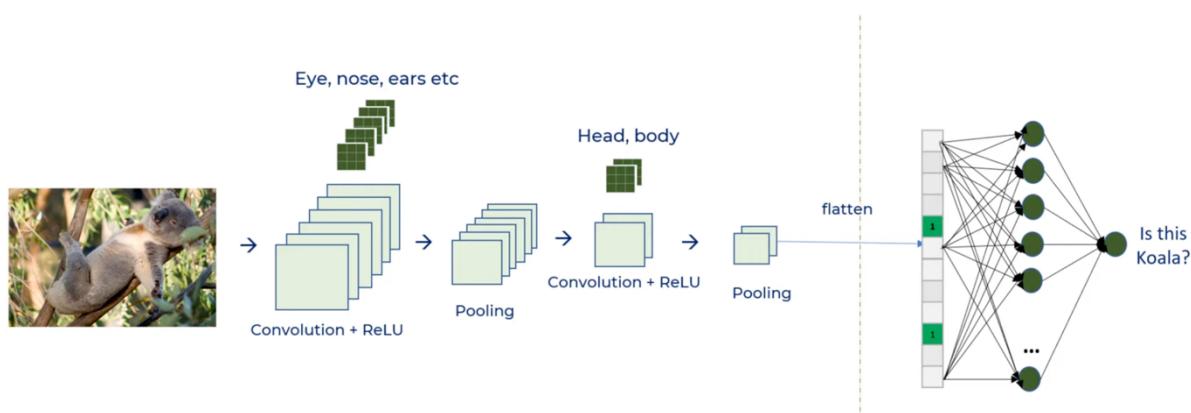
El método denominado “Pooling” sirve para reducir el tamaño de las imágenes.

El más usado es el denominado “Max Pooling” y la idea es reducir la imagen agrupando los pixels y eligiendo el mayor valor del grupo:



Podemos pensar que esto reduciría las predicciones de la red neuronal, pero realmente no es así ya que la información básica para la extracción de características se mantiene (pensemos que estamos hablando de imágenes grandes)

En resumen, una CNN se diseña de la siguiente manera:





¡Ojo! Pero una CNN (Red Neuronal Convolucional) no tiene en cuenta la rotación ni el escalado.

Se necesitan tener imágenes que tengan rotación y/o escalado. Para ello (si no tenemos esas muestras), tenemos el método de “Data Augmentation” (ya visto anteriormente) el cual consiste en generar nosotros, a partir de una imagen, otras imágenes variando la rotación, escalado .. etc. De esta manera aumentamos el dataset para aumentar la generalización de las clasificaciones.