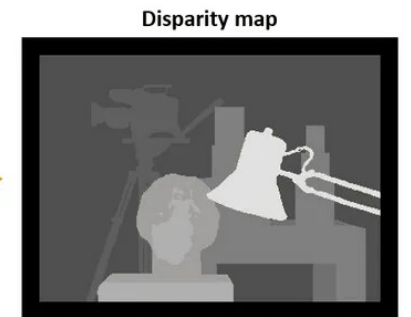
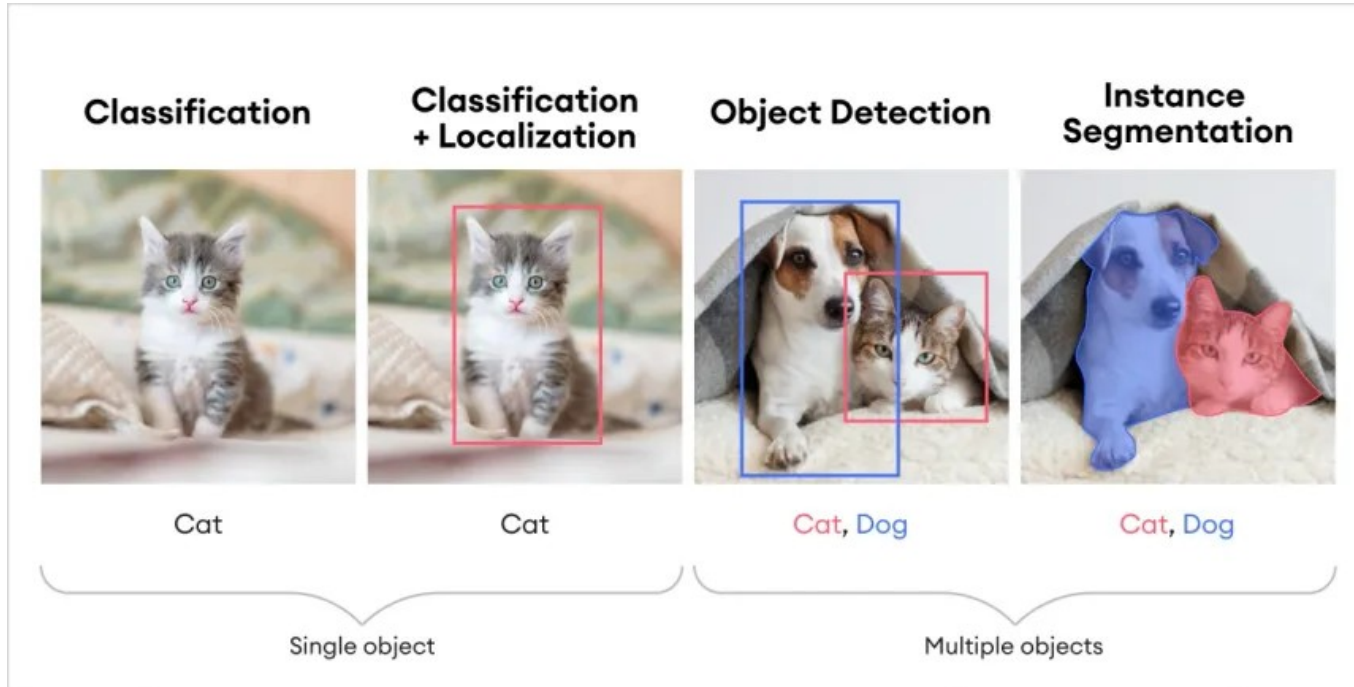




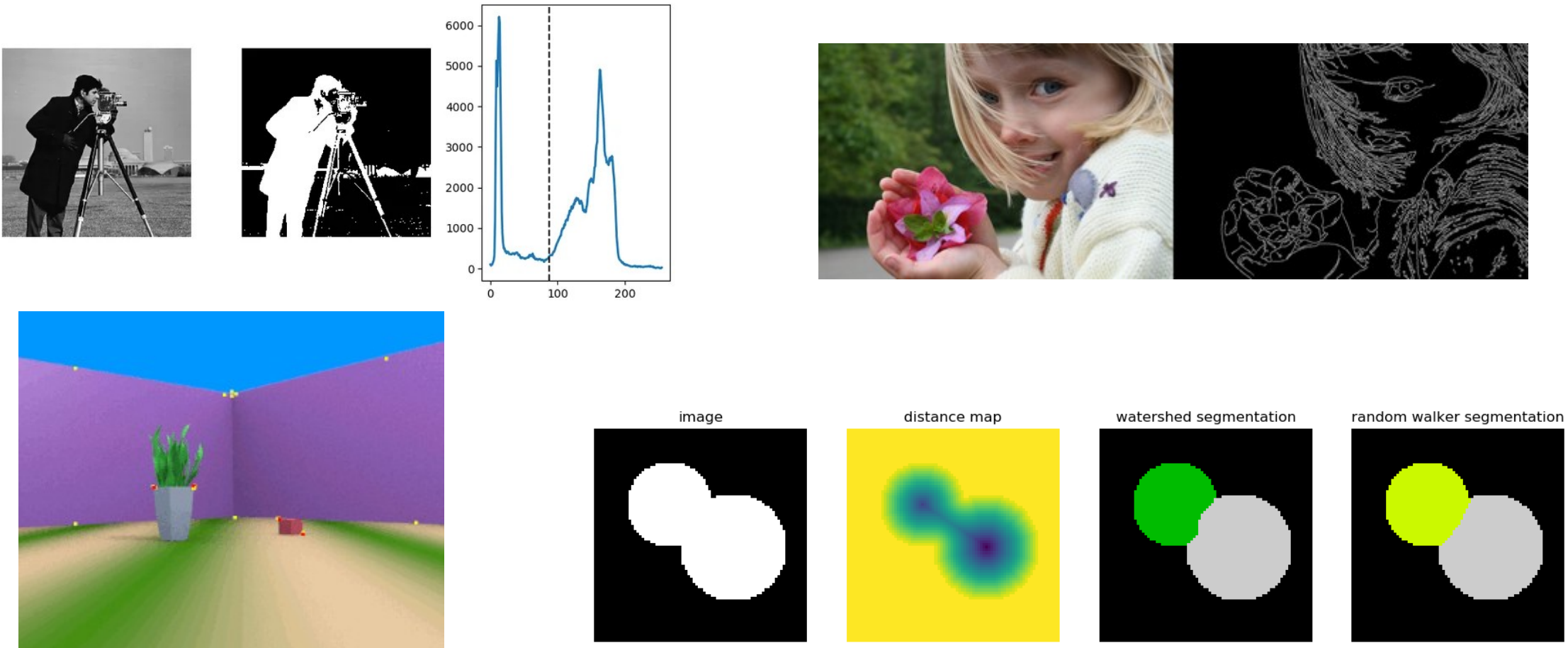
# Data Science

## Part-time Bootcamp

**Visión por computador** Los humanos hemos querido replicar muchas de las tareas del cerebro, y el trabajar con imágenes siempre ha sido un reto. ¿Cómo podemos obtener de las imágenes la información necesaria?



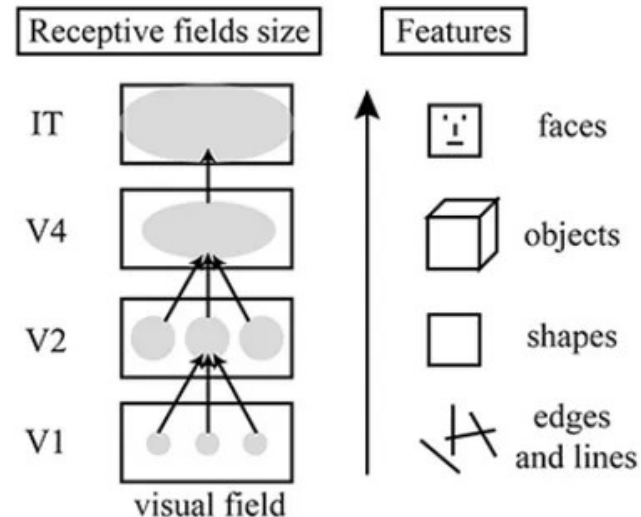
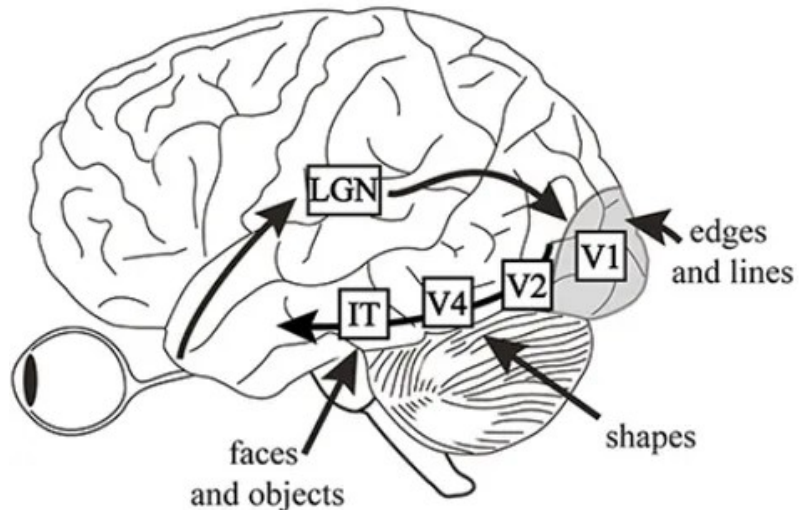
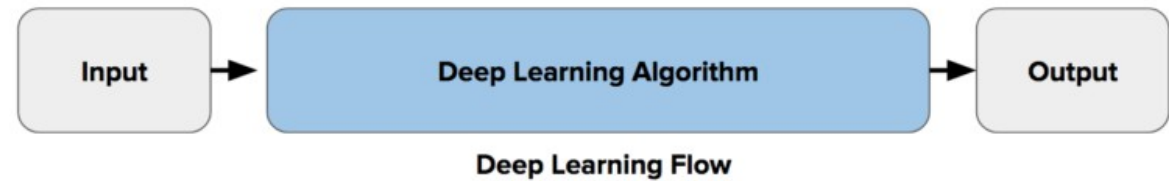
Las primeras aproximaciones requerían de un trabajo previo importante relativo a calibrado de las imágenes, detección de bordes, ... Se empleaban técnicas del dominio de la señal.



Estas imágenes tratadas luego eran enviadas aun modelo de clasificación para así entonces poder realizar la tarea en cuestión (detectar caras, identificar usuarios concretos,...)



¿Pero trabaja así el cerebro?



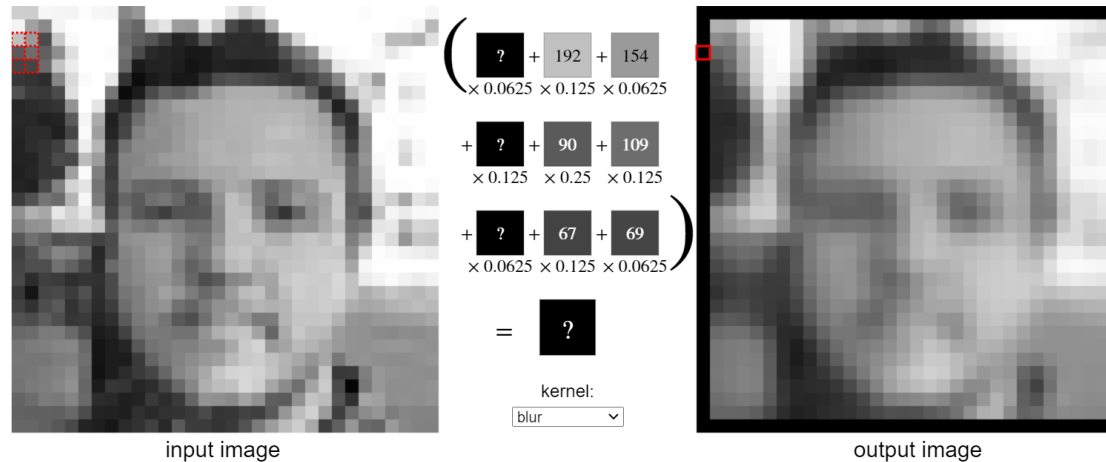
Detectamos unidades mínimas, orientadas, que posteriormente en capas componen formas hasta llegar a objetos reconocibles por nuestro cerebro.



Podemos del mismo modo, plantear una arquitectura de red neuronal que imite este proceso.

## Convoluciones

Las convoluciones son operaciones matemáticas que aplican un filtro (matriz o kernel) que se va moviendo por la imagen para generar una versión de menor resolución enfocada en los cambios en los bordes de la imagen.

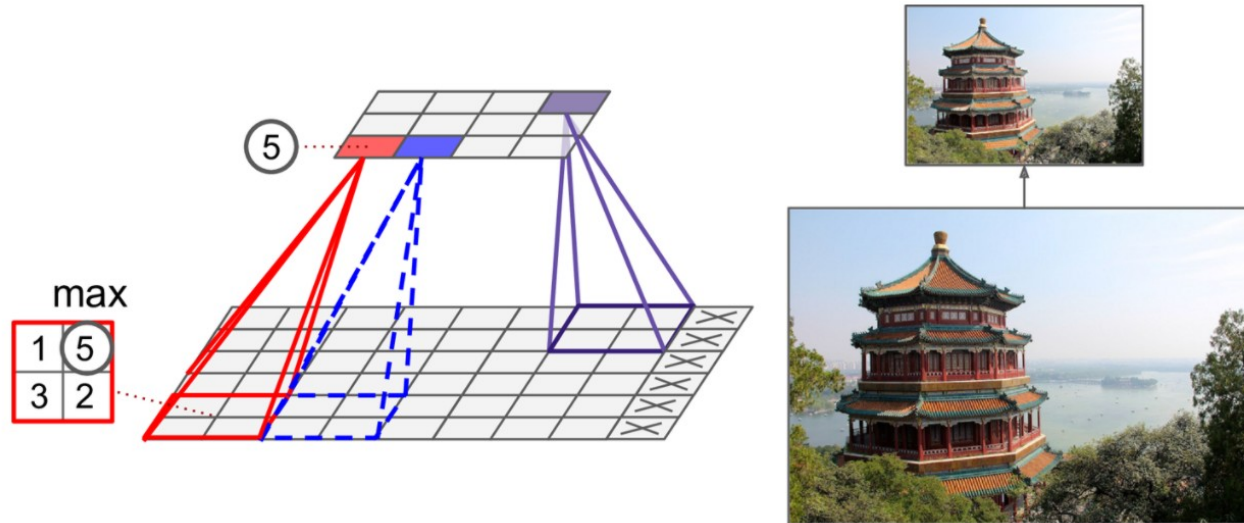


<https://setosa.io/ev/image-kernels/>

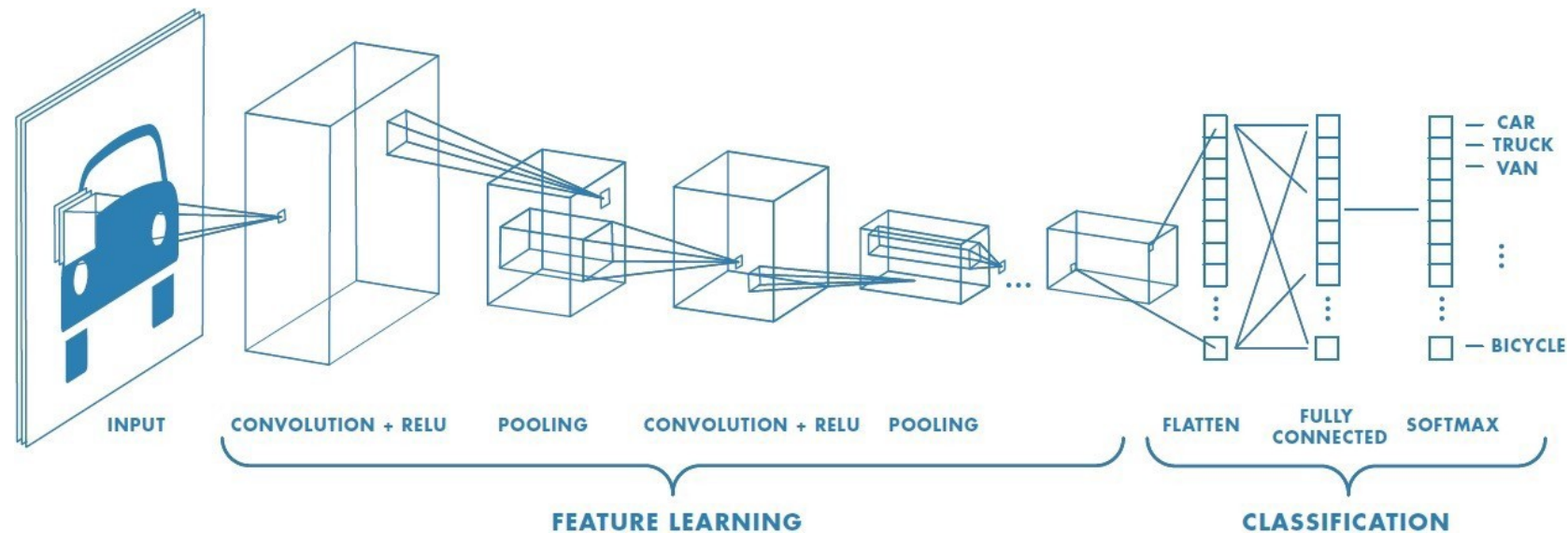
Podemos del mismo modo, plantear una arquitectura de red neuronal que imite este proceso.

## Pooling

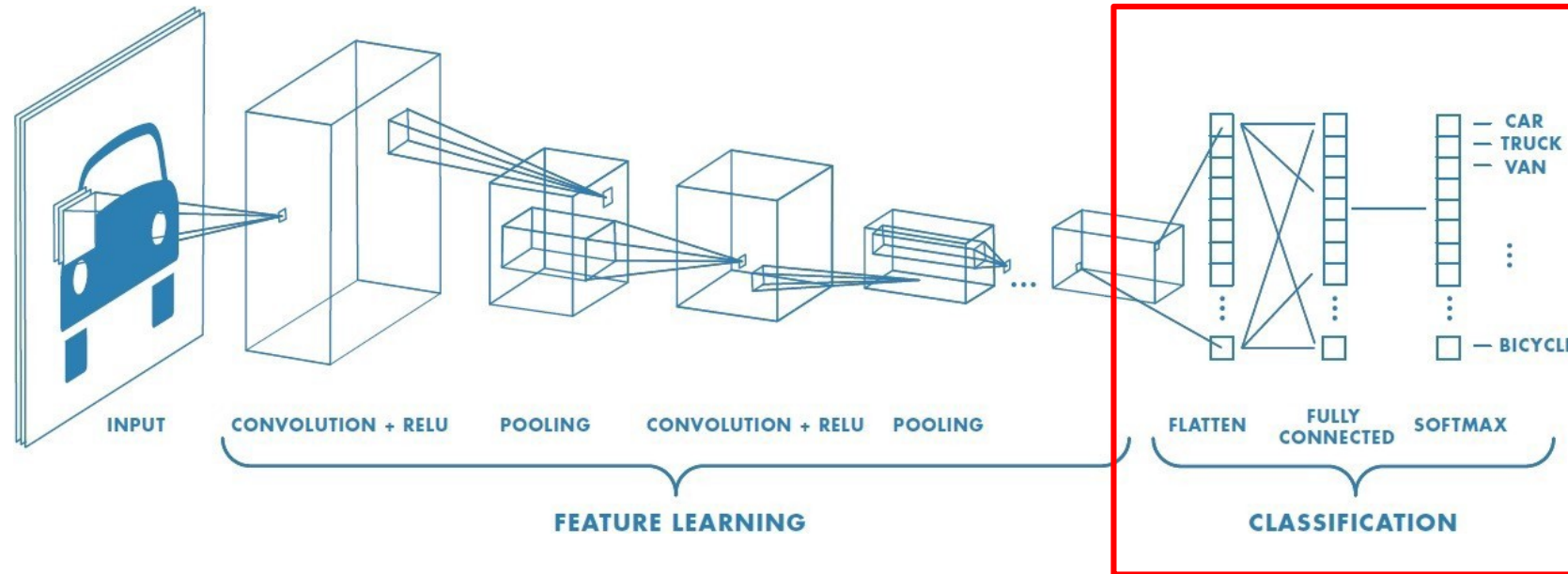
Para concentrar la información de varios pixeles en una imagen de menor tamaño de modo que concentremos su contenido.



Con estas herramientas es cuestión de montar redes que simulen una estructura candidata de la corteza visual nuestro cerebro. Una vez extraídas las características clave, podemos enfocarnos en la tarea en cuestión en un ejercicio de **entrenamiento supervisado**.



Dado lo pesado de entrenar las primeras capas convolucionales, podemos tomar redes ya entrenadas y sustituir la capa final de clasificación de manera que solo necesitamos detallar un nuevo objetivo.



Esta tarea se conoce como **transfer learning** y ahorra mucho tiempo en disponer modelos para tareas con datos de entrada parecidos para tareas de distinta naturaleza.



## Referencias:

- [https://dcain.etsin.upm.es/~carlos/bookAA/05.7\\_RRNN\\_Convoluciones\\_CIFAR\\_10\\_INFORMATIVO.html](https://dcain.etsin.upm.es/~carlos/bookAA/05.7_RRNN_Convoluciones_CIFAR_10_INFORMATIVO.html)
- <https://www.analyticsvidhya.com/blog/2021/10/understanding-transfer-learning-for-deep-learning/>
- [https://proceedings.neurips.cc/paper\\_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf)

