

# **APRENDIZAXE AUTOMÁTICA SUPERVISADA**

## **3º Grao Intelixencia Artificial**

**P3. Redes Neuronais Artificiales**  
**Curso 2025-26**

**José María Alonso Moral**  
**CiTIUS y Departamento de Electrónica e Computación**  
**Universidade de Santiago de Compostela**  
**[josemaria.alonso.moral@usc.es](mailto:josemaria.alonso.moral@usc.es)**

# Redes Neuronales Artificiales

- Multilayer Perceptron (MLP)
- Deep Learning:
  - Computer Vision:
    - Convolutinal Neural Networks (CNN)
      - Fashion-MNIST dataset

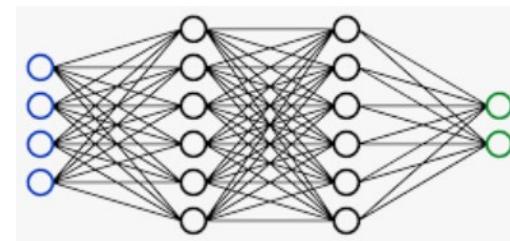
# MLP

- **Dataset:** Fashion-MNIST
  - Training: 60k images
  - Test: 10k images
  - 10 classes



<https://github.com/zalandoresearch/fashion-mnist>

- **ScikitLearn MLP:**
  - Input: all pixels
  - No need to optimize hyperparameters
  - FeatureSelection [+ PCA + OpenCV (manual extraction)]
  - Backpropagation
- **Keras/TensorFlow (Pytorch) – GOOGLE COLAB**



# Convolución

- Operación matemática (transformación espacio-temporal)

- $y = f * g$

$$(f * g)(t) \doteq \int_{-\infty}^{\infty} f(\eta)g(t - \eta)d\eta$$

$$f[m] * g[m] = \sum_n f[n]g[m - n]$$

- Propiedades:

**Commutatividad**

$$f * g = g * f$$

**Asociatividad**

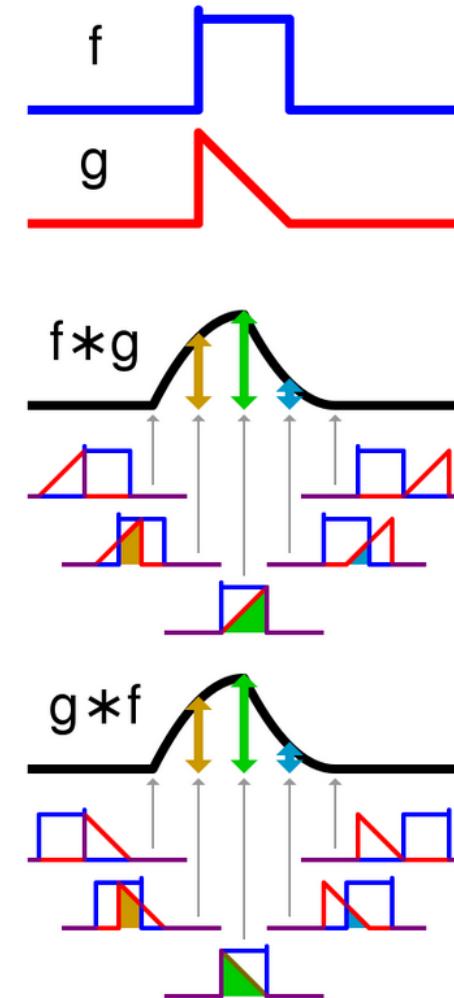
$$f * (g * h) = (f * g) * h$$

**Distributividad**

$$f * (g + h) = (f * g) + (f * h)$$

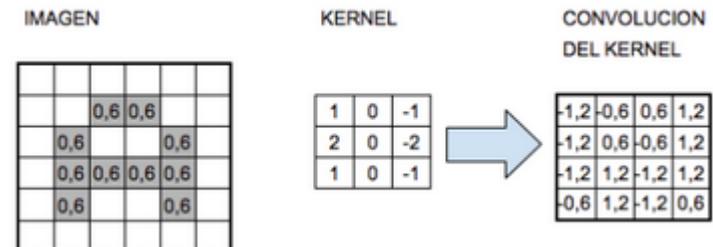
- Teorema de Transformada de Fourier:

$$\mathcal{F}(f * g) = (\mathcal{F}(f)) \cdot (\mathcal{F}(g))$$



# CNN

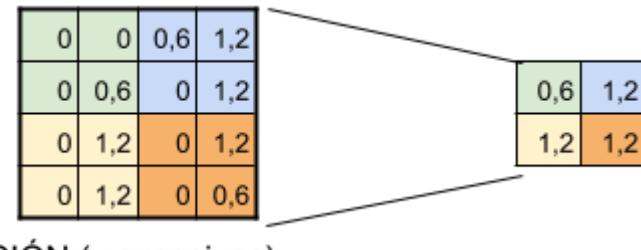
- Múltiples capas de filtros convolucionales de 1 o más dimensiones
- La imagen se interpreta como una matriz de píxeles
  - Cada pixel toma un valor 0..255 normalizado a 0..1
  - Si la imagen tiene 28 x 28 píxeles
    - Blanco y negro:  $28 \times 28 = 784$  neuronas
    - Color (RGB):  $28 \times 28 \times 3 = 2352$  neuronas
- Las convoluciones se aplican a grupos de píxeles cercanos
  - Ej: kernel 3 x 3
    - Se recorrerá la imagen de izquierda a derecha y de arriba abajo
  - Filtro: conjunto de kernels
    - cada filtro genera una matriz de salida



# CNN

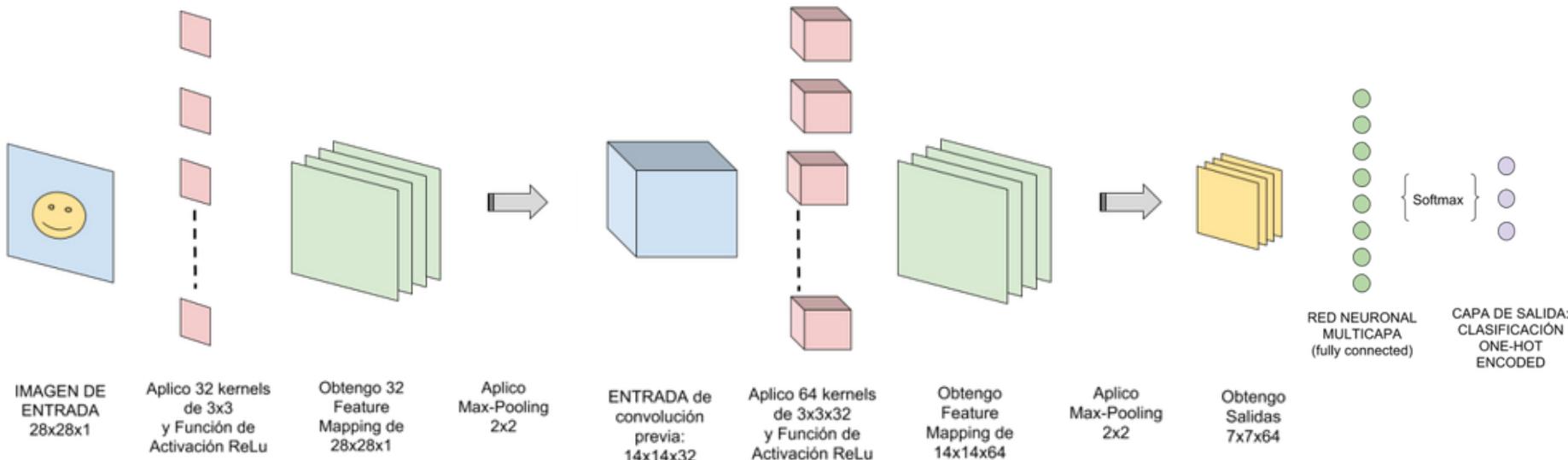
$$f(x) = \max(0, x).$$

- Capa de activación ReLU (Rectifier Linear Unit)
- Muestreo - Subsampling (Ej. Max-Pooling 2x2)
  - para reducir el tamaño de la próxima capa de neuronas
  - preservando las características más importantes que detectó cada filtro



PRIMERA CONVOLUCIÓN

SEGUNDA CONVOLUCIÓN (y sucesivas)



# CNN vs MLP

- **Capa de entrada:**
  - MLP: características de la imagen (no escalable por pixeles)
  - CNN: pixeles
- **Capas ocultas:**
  - MLP: número variable de neuronas por capa
  - CNN: capas jerárquicas de convolución (kernels) + muestreo
- **Capa de salida:**
  - MLP: 1 neurona por cada clase
  - CNN: aplanado en 1 o más capas tradicionales + SoftMax
- **Backpropagation:**
  - MLP: para ajustar los pesos de todas las interconexiones
  - CNN: para ajustar los pesos de los kernels