

Día 04 · Tema 2 — Ejercicios: convolución, pooling y transfer learning

Cómo se usa esta hoja. Hay dos tipos de ejercicio, y sólo dos.

El ejercicio del día. Uno por hoja. Es el obligatorio y es el que entra en el examen, con el cálculo incluido. Basta con hacer ese para seguir el curso con solvencia.

Los ejercicios de ampliación. El resto. Están para enseñar de qué es capaz el método, no para exigir: **no se piden calculados en el examen.** De ellos sólo puede preguntarse la idea, a nivel conceptual.

AMPLIACIÓN *opcional · no se pide calculado en el examen*

1. Recuento de parámetros: MLP frente a convolución

Considera una imagen de entrada de dimensiones $32 \times 32 \times 3$ (alto $\times$ ancho $\times$ canales RGB). Queremos obtener 16 mapas de características de salida.

- (a) Si utilizamos una capa *fully connected* (MLP) que conecta cada píxel aplanado con cada una de las 16 neuronas de salida, ¿cuántos parámetros tiene la capa (pesos + sesgos)?
- (b) Si utilizamos una capa convolucional con 16 filtros de tamaño 3×3 , ¿cuántos parámetros tiene? Recuerda: $\text{params} = k^2 \cdot C_{\text{in}} \cdot C_{\text{out}} + C_{\text{out}}$.
- (c) ¿Cuántas veces más parámetros tiene la capa MLP respecto a la convolucional? Comenta por qué esta diferencia es relevante para imágenes de mayor resolución.

EL EJERCICIO DEL DÍA *obligatorio · entra en el examen*

2. Convolución a mano (1 canal)

Sea la entrada $\mathbf{X} \in \mathbb{R}^{4 \times 4}$ (un solo canal) y el kernel $\mathbf{K} \in \mathbb{R}^{3 \times 3}$:

$$\mathbf{X} = \begin{pmatrix} 1 & 0 & 2 & 1 \\ 3 & 1 & 0 & 2 \\ 2 & 0 & 1 & 3 \\ 1 & 2 & 3 & 0 \end{pmatrix}, \quad \mathbf{K} = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}.$$

Sin padding ($p = 0$) y con stride $s = 1$:

- (a) ¿Cuáles son las dimensiones del mapa de salida?
- (b) Calcula cada elemento $Y_{i,j} = \sum_u \sum_v X_{i+u, j+v} K_{u,v}$ explícitamente.

AMPLIACIÓN *opcional · no se pide calculado en el examen*

3. Dimensiones de salida: conv + pooling

Una red recibe una imagen de $224 \times 224 \times 3$ y aplica las siguientes operaciones en secuencia:

- (a) Convolución con kernel 7×7 , stride $s = 2$ y padding $p = 3$. Calcula las dimensiones espaciales de salida usando

$$H' = \left\lfloor \frac{H + 2p - k}{s} \right\rfloor + 1.$$

- (b) Max-pooling 2×2 con stride 2. ¿Cuáles son las dimensiones finales?
- (c) ¿Reconoces estas operaciones como parte de alguna arquitectura famosa?

AMPLIACIÓN *opcional · no se pide calculado en el examen*

4. Convolución multicanal

Ahora la entrada tiene 2 canales y dimensión espacial 3×3 . Un único filtro de salida tiene kernel $2 \times 2 \times 2$ y sesgo $b = 0,5$. Los datos son:

Canal 0 de la entrada:

$$\mathbf{X}^{(0)} = \begin{pmatrix} 1 & 0 & 2 \\ 3 & 1 & 0 \\ 2 & 0 & 1 \end{pmatrix}, \quad \text{Canal 1: } \mathbf{X}^{(1)} = \begin{pmatrix} 0 & 1 & 1 \\ 2 & 0 & 3 \\ 1 & 1 & 0 \end{pmatrix}.$$

Kernel (canal 0) y kernel (canal 1):

$$\mathbf{K}^{(0)} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{K}^{(1)} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

Calcula el mapa de salida $\mathbf{Y} \in \mathbb{R}^{2 \times 2}$ usando $Y_{i,j} = \sum_c \sum_u \sum_v X_{i+u, j+v}^{(c)} K_{u,v}^{(c)} + b$.

AMPLIACIÓN *opcional · no se pide calculado en el examen*

5. Max-pooling a mano

Dado el mapa de características $\mathbf{M} \in \mathbb{R}^{4 \times 4}$:

$$\mathbf{M} = \begin{pmatrix} 1 & 3 & 2 & 4 \\ 5 & 6 & 7 & 8 \\ 3 & 2 & 1 & 0 \\ 9 & 4 & 6 & 5 \end{pmatrix}.$$

- Aplica max-pooling 2×2 con stride 2 y escribe la salida 2×2 .
- ¿Cuántos parámetros entrenables introduce esta capa de pooling?
- Repite el apartado (a) usando average-pooling en lugar de max-pooling.

AMPLIACIÓN *opcional · no se pide calculado en el examen*

6. Recuento de parámetros de una arquitectura completa

Considera la siguiente red para clasificar imágenes $32 \times 32 \times 3$ en 10 clases:

Capa	Configuración	Tras capa	Tras pooling
Conv1	$3 \rightarrow 16, k = 3, p = 1, s = 1$	$? \times ? \times 16$	MaxPool 2×2
Conv2	$16 \rightarrow 32, k = 3, p = 1, s = 1$	$? \times ? \times 32$	MaxPool 2×2
FC	$32 \cdot 8 \cdot 8 \rightarrow 10$	—	—

- Completa las dimensiones espaciales tras cada convolución y tras cada max-pool.
- Calcula el número de parámetros de Conv1, Conv2 y FC por separado.
- Calcula el total de parámetros de la red. ¿Qué capa domina el recuento?

AMPLIACIÓN *opcional · no se pide calculado en el examen*

7. Bloque residual (ResNet)

En un bloque residual la salida es $\mathbf{y} = F(\mathbf{x}; W) + \mathbf{x}$, donde F es la transformación aprendida.

Sea $\mathbf{x} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$ y $F(\mathbf{x}; W) = W\mathbf{x}$ con $W = \begin{pmatrix} 0,5 & -0,3 \\ 0,1 & 0,8 \end{pmatrix}$.

- Calcula $F(\mathbf{x})$.
- Calcula $\mathbf{y} = F(\mathbf{x}) + \mathbf{x}$.
- ¿Qué aprende la rama F si la inicializamos cerca de cero? ¿Qué ventaja tiene esto para el entrenamiento de redes profundas?

AMPLIACIÓN *opcional · no se pide calculado en el examen*

8. Coste asimétrico: transfer learning en detección de fraude

Un modelo de transfer learning (fine-tuning de las últimas capas de una red preentrenada) se aplica a detección de fraude bancario. Las probabilidades predichas y las etiquetas reales para 4 transacciones son:

Transacción	$\hat{y}$ (prob. fraude)	y (etiqueta real)
1	0,8	1 (fraude)
2	0,7	0 (legítima)
3	0,4	1 (fraude)
4	0,9	1 (fraude)

El banco asigna $c_{FP} = 10$ € (coste de investigar una transacción legítima) y $c_{FN} = 50$ € (coste de no detectar un fraude).

- Usando un umbral de clasificación $\tau = 0,5$, clasifica cada transacción, identifica FP y FN y calcula el coste total.
- Repite con $\tau = 0,75$. ¿Cambia el coste? ¿Qué umbral es preferible dados los costes asimétricos?
- ¿Por qué tiene sentido usar *transfer learning* en este problema en lugar de entrenar una red desde cero? Relaciona tu respuesta con el tamaño típico de los datasets de fraude.