

Ising ML

DETECCIÓN DE TRANSICIONES DE FASE

José David Bernal Restrepo

Nicolas Osorno Roa

Salomón Uran Parra



**UNIVERSIDAD
DE ANTIOQUIA**
Facultad de Ciencias Exactas y Naturales

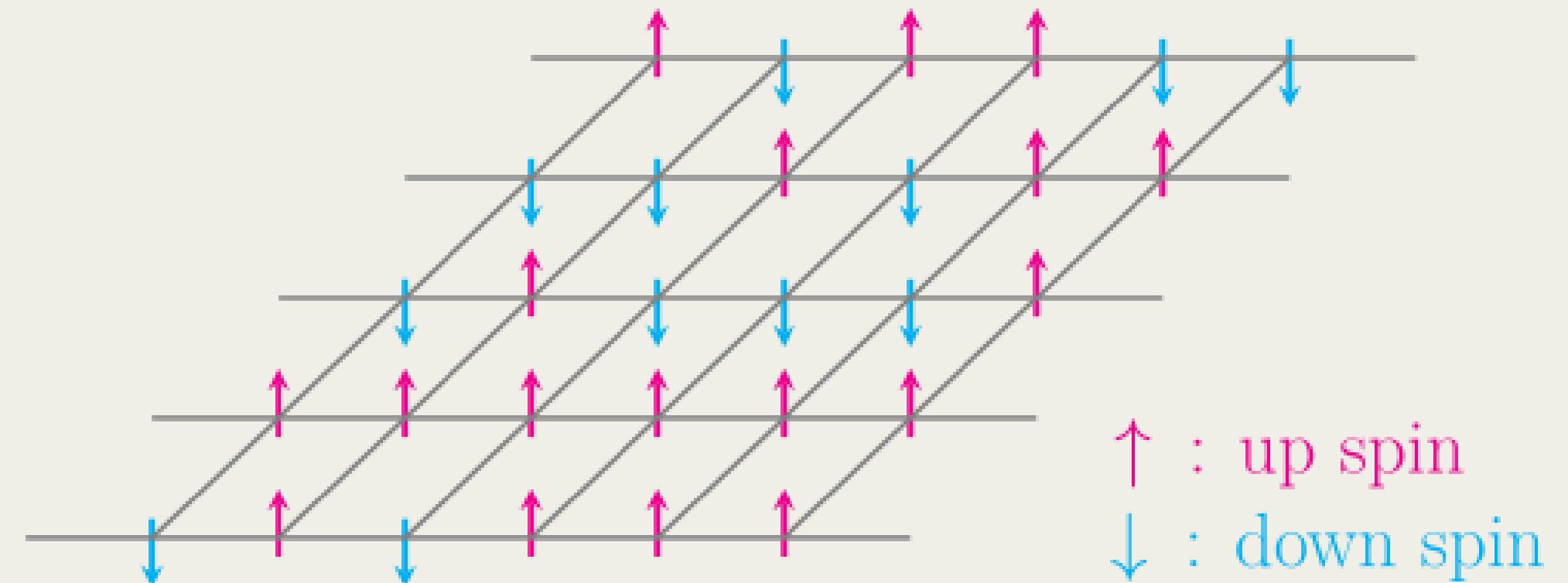
MODELO DE ISING

El modelo de Ising es un modelo matemático de ferromagnetismo que consiste en variables binarias llamadas spins (± 1) dispuestas en una red, que interactúan solo con sus vecinos más cercanos. Describe cómo estas interacciones locales pueden dar lugar a fenómenos colectivos como el orden magnético y las transiciones de fase.

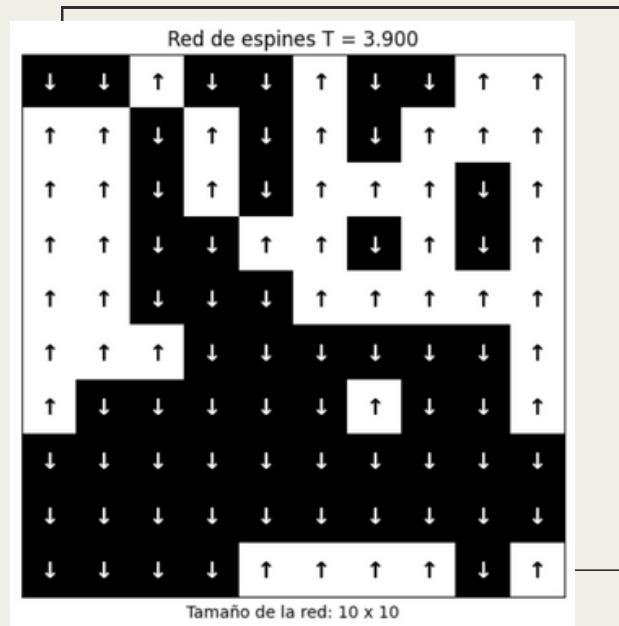
$$\mathcal{H} = -J \sum_{\langle i,j \rangle} \sigma_i^z \sigma_j^z - H \sum_i \sigma_i^z$$

$$\mathcal{M} \sim \sum_i \sigma_i^z$$

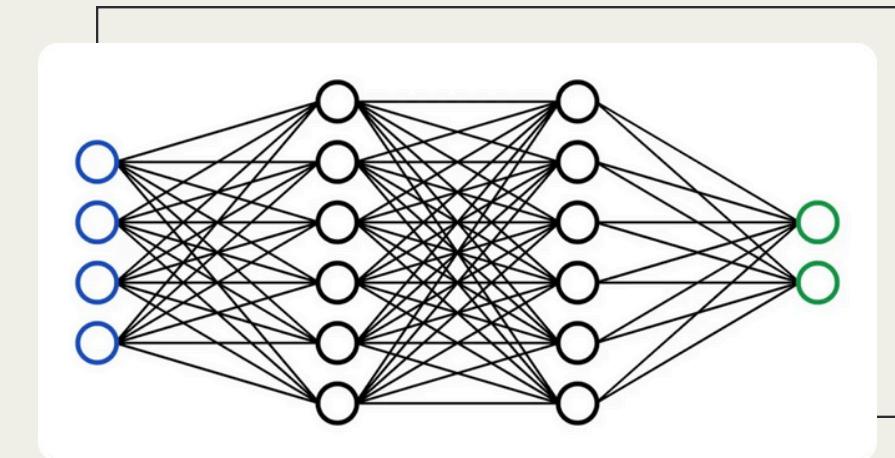
$$T_c = \frac{2J}{k_B \log(\sqrt{2} + 1)}$$



ROADMAP

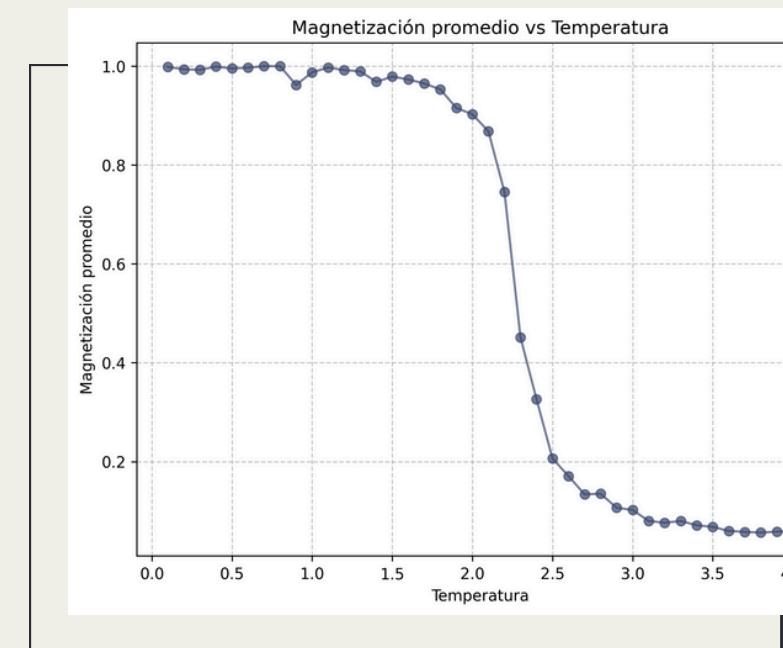


Análisis de datos

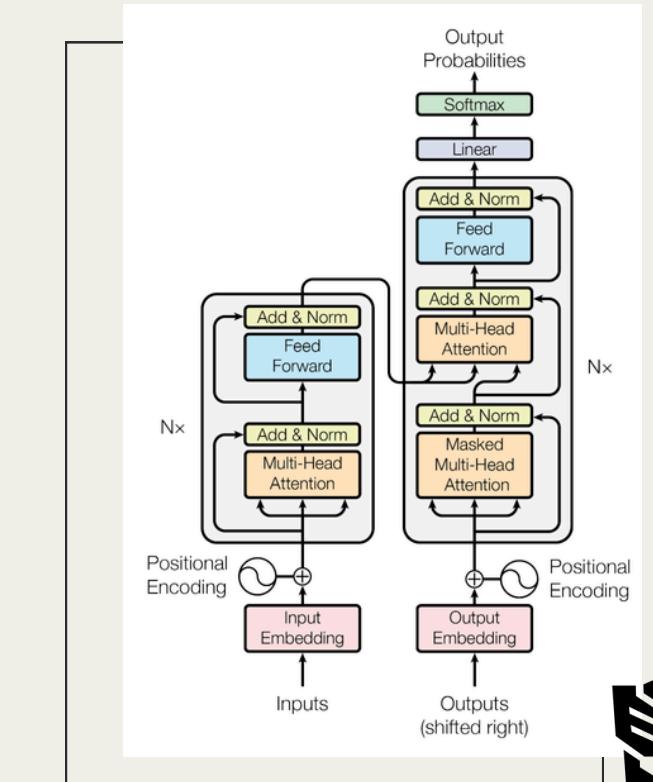


Modelo generativo

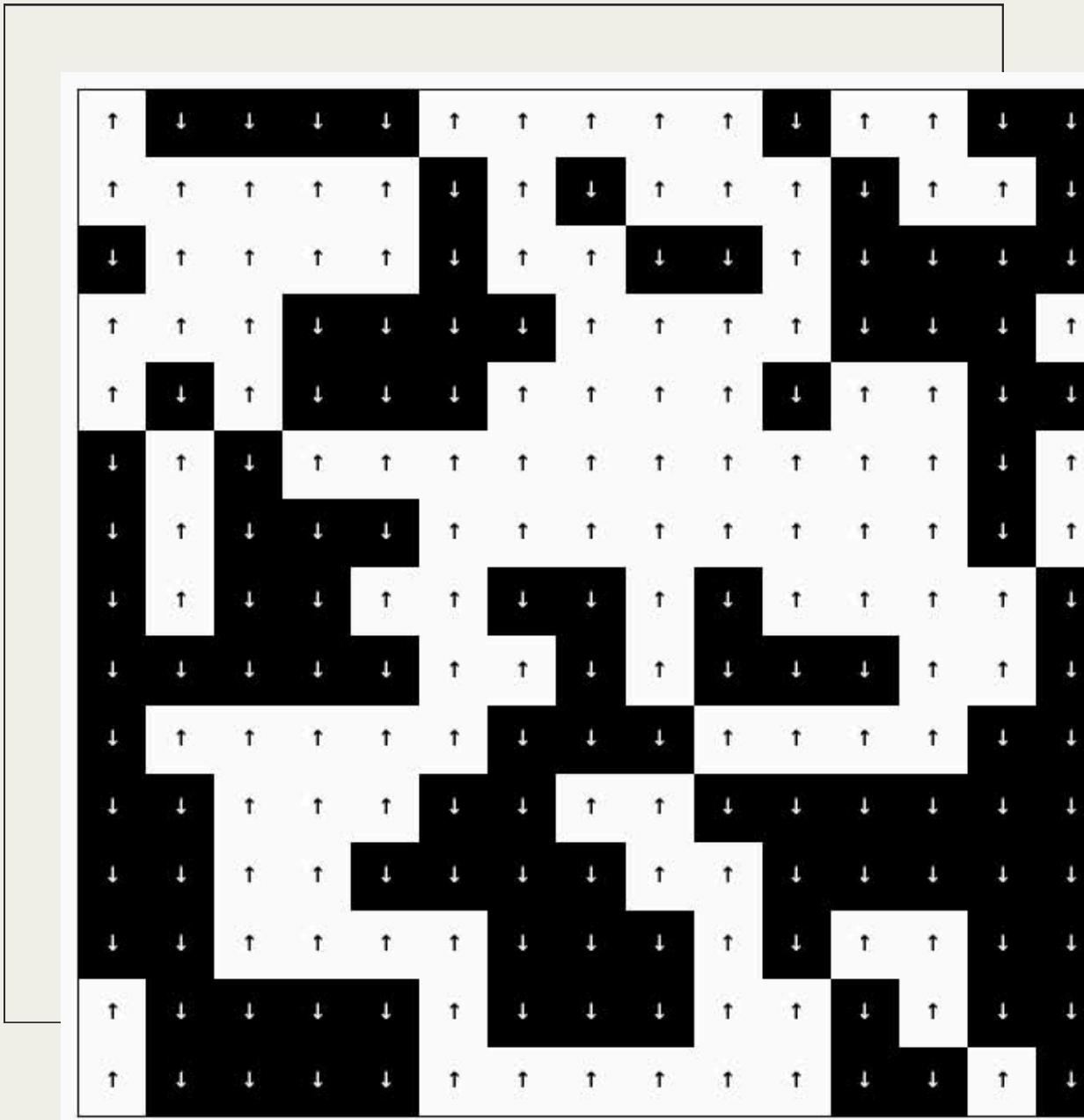
Generación del conjunto de datos



Modelo de clasificación



GENERACIÓN DE DATOS

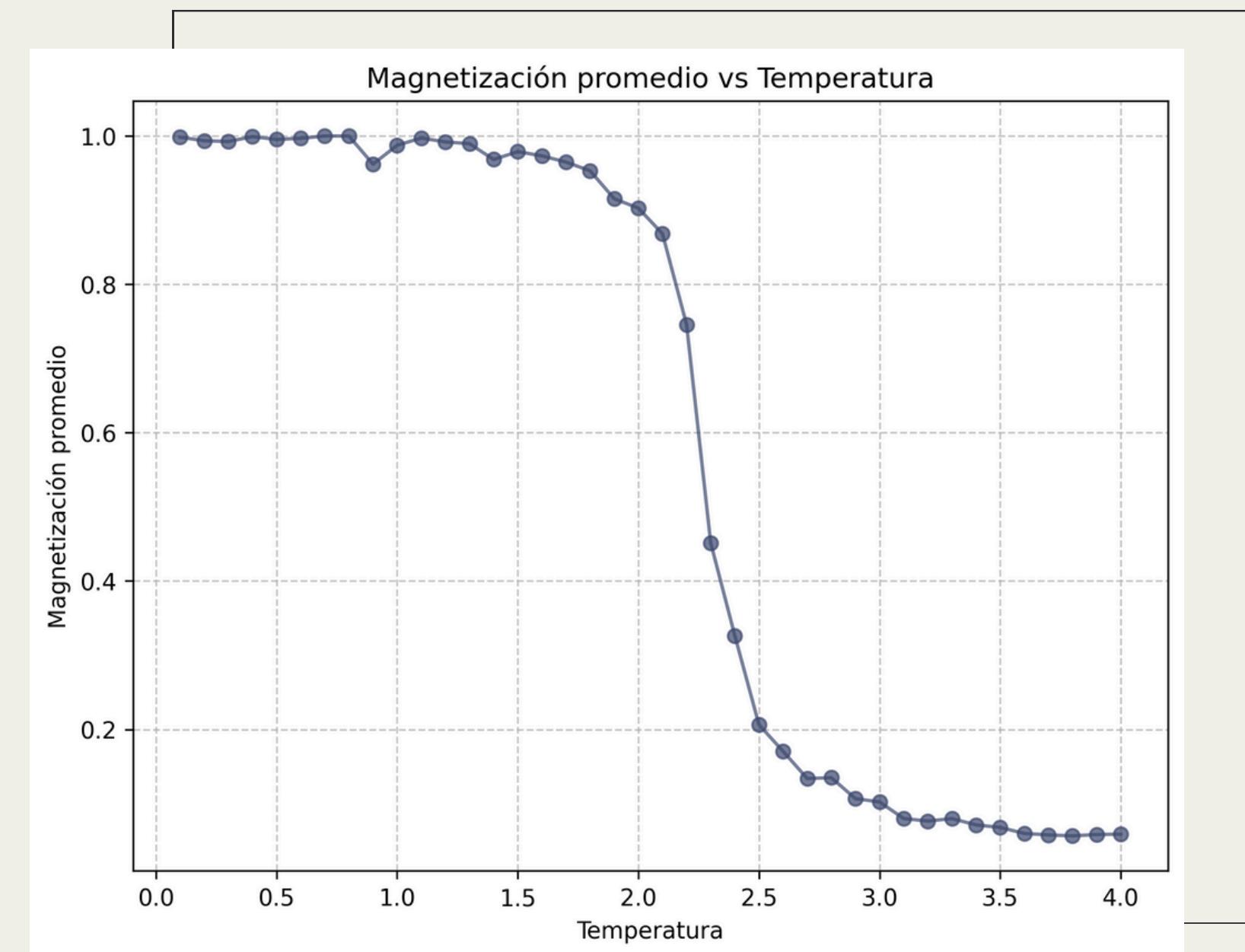


Algoritmo Metropolis-Hastings + simulated Annealing

- Definir temperatura T y parámetros del sistema
- Repetir por N barridos (después de termalizar el sistema):
 - Para un espín (i, j):
 - Calcular la variación de energía ΔE si el espín se volteea
 - Si $\Delta E \leq 0 \Rightarrow$ aceptar el cambio (minimiza energía)
 - Si $\Delta E > 0$ aceptar con probabilidad $\exp(-\Delta E / T)$ (Esto permite fluctuaciones térmicas controladas por T)
 - Además implementamos un proceso de enfriamiento para evitar caer en mínimos locales

ANÁLISIS DE DATOS

Después de generar los datos en un rango de temperatura fijo entre 0.1 y 4.0, distribuidos con un $\Delta T = 0.1$, obtenemos la distribución estadística de la energía y la magnetización para cada conjunto de datos. Finalmente verificamos que el modelo tenga el comportamiento esperado para la magnetización en función de la temperatura

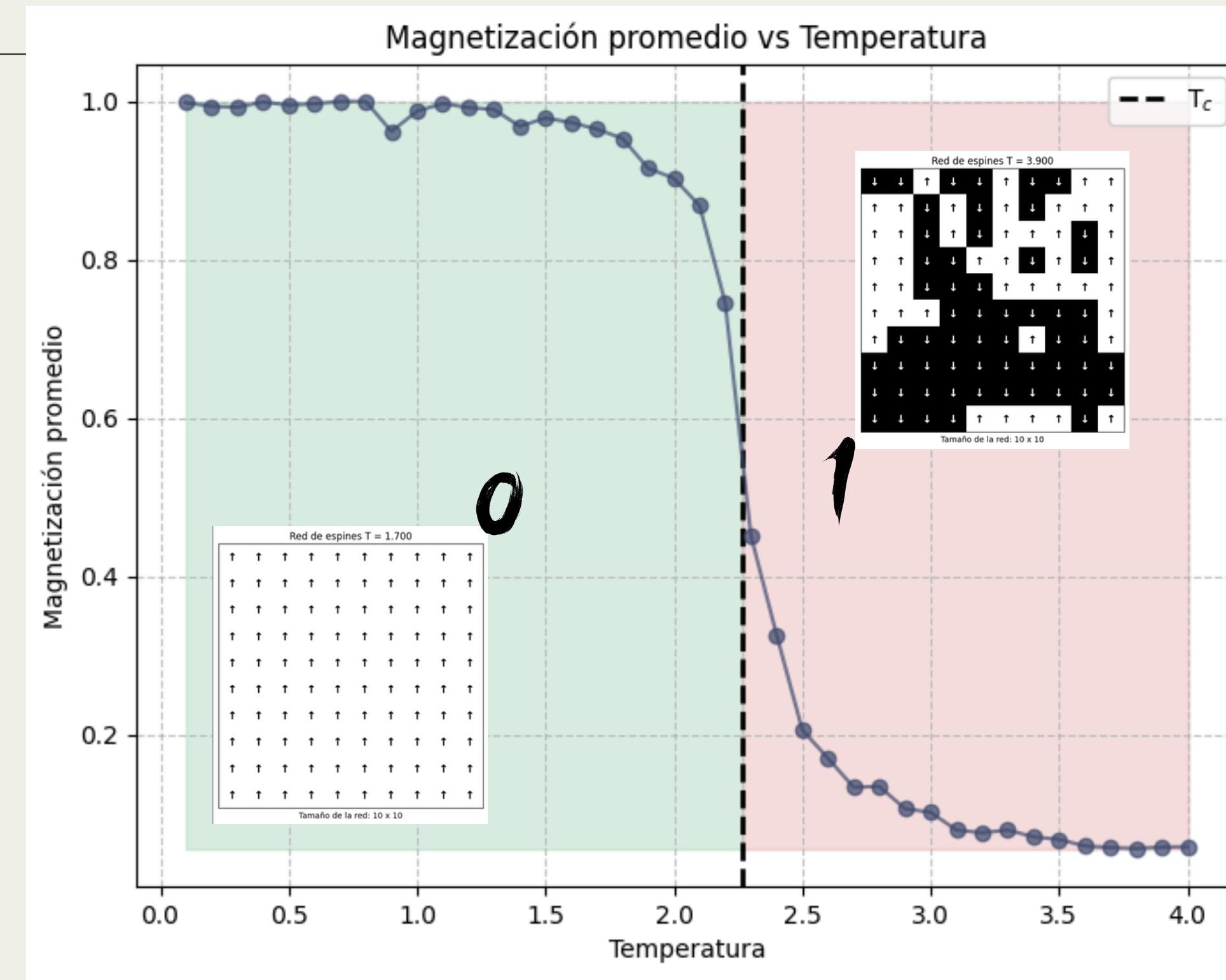


Magnetización promedio en función de la temperatura para una red de tamaño 30x30

Modelos implementados

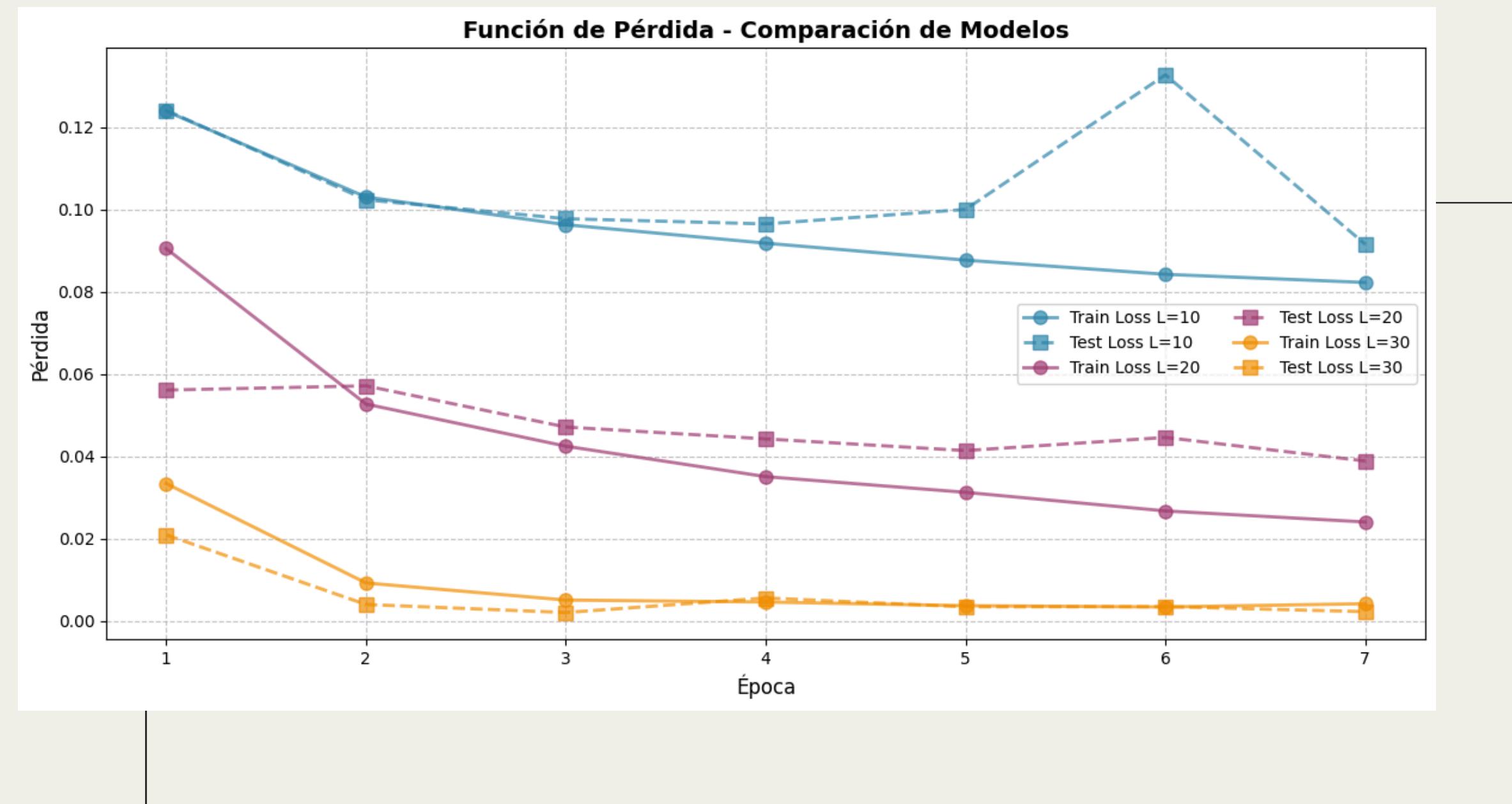
*El entrenamiento se realizó en un sistema **Windows 11** con una **CPU AMD Ryzen (Zen 4)** y **32 GB** de memoria **RAM**, utilizando **PyTorch 2.6.0 (versión CPU-only)** y **Python 3.12**. Todos los entrenamientos se llevaron a cabo bajo las mismas condiciones computacionales.

PROBLEMA DE CLASIFICACIÓN

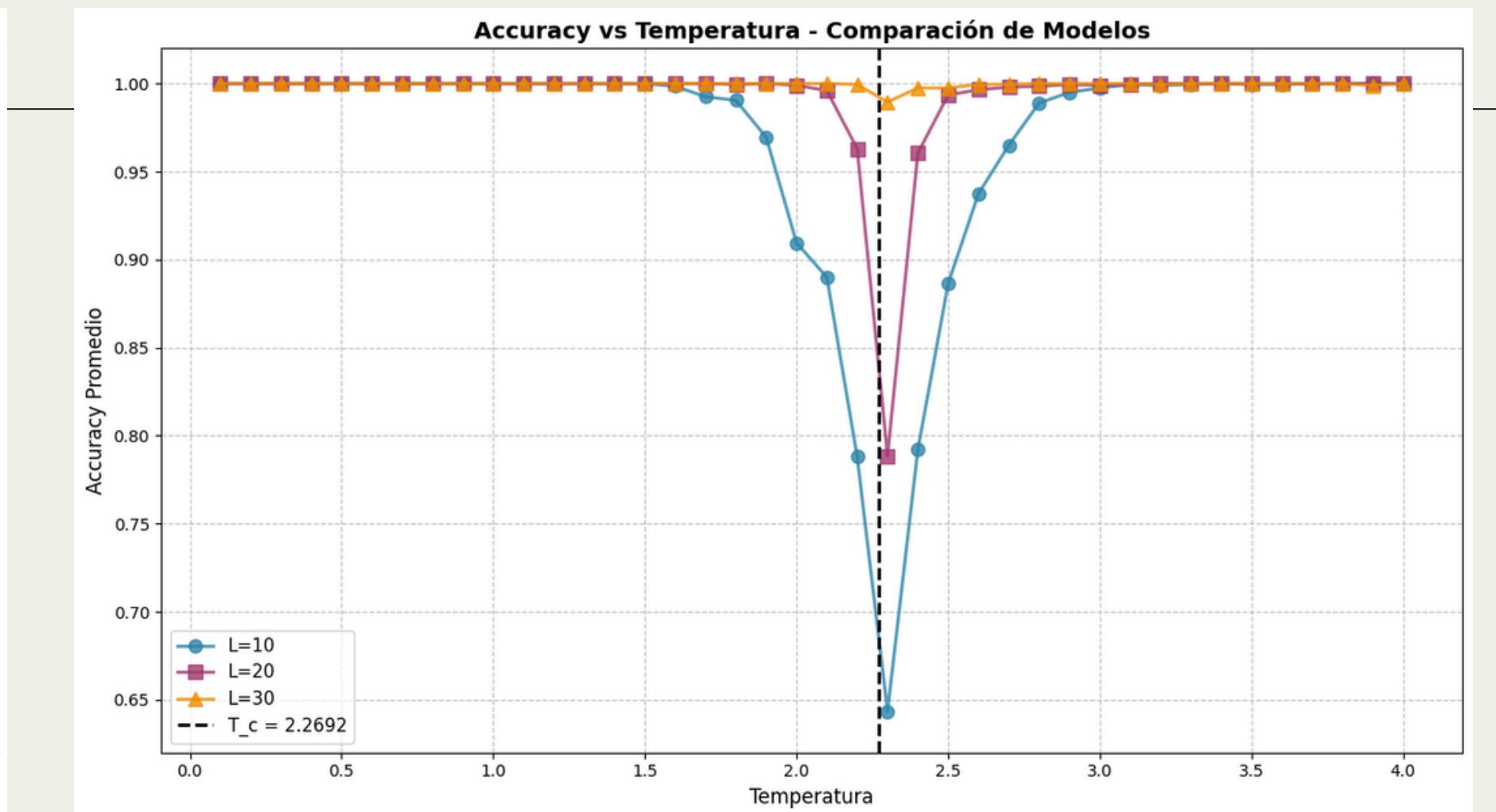
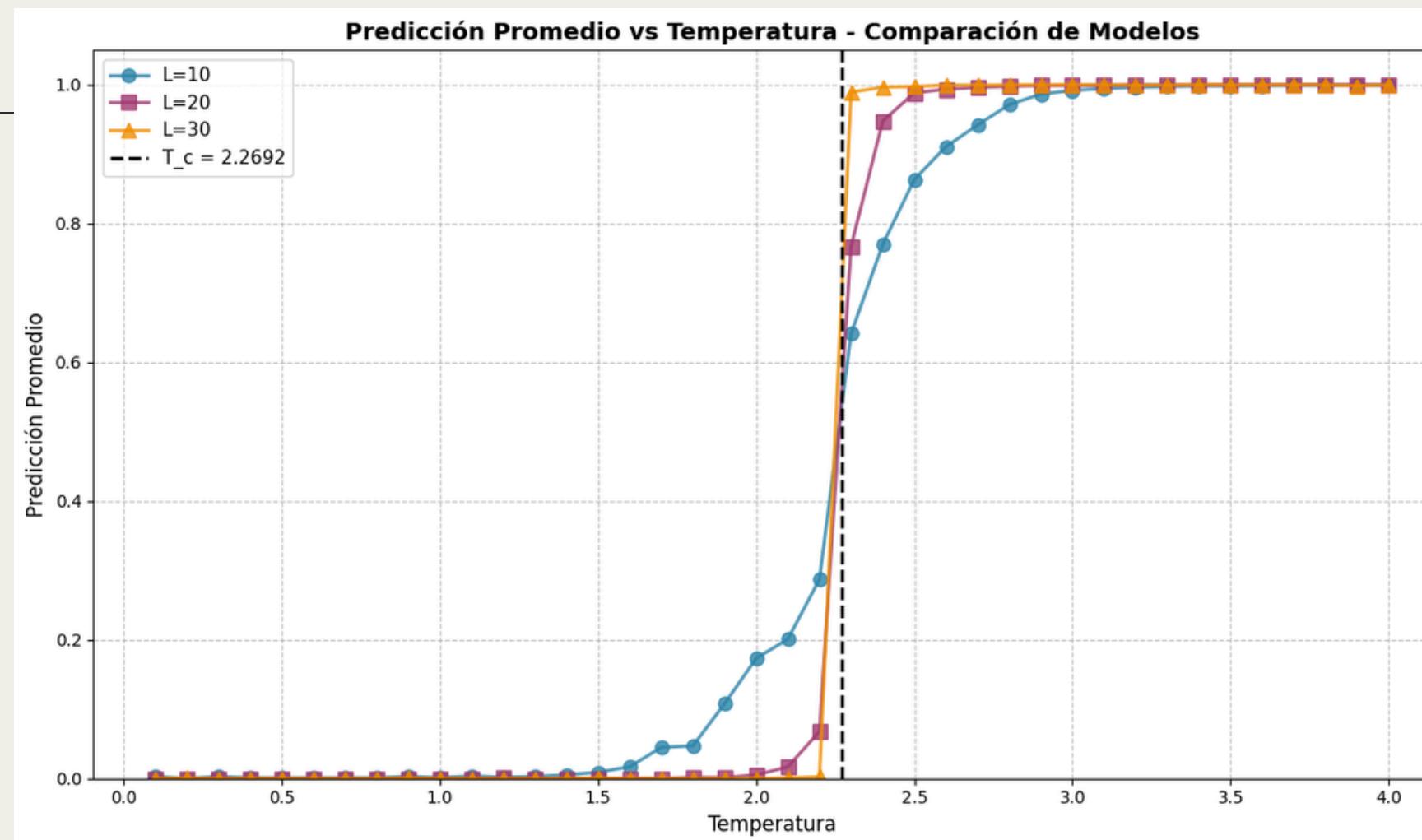


NEURAL NETWORK (CLASIFICACIÓN)

- Red neuronal densa (**MLP**)
- Entrada: configuración de spins ($n \times n \rightarrow$ vector)
- Aplanado previo (Flatten)
- Capas **fully connected** con **ReLU**
- Sin tratamiento explícito de la estructura espacial

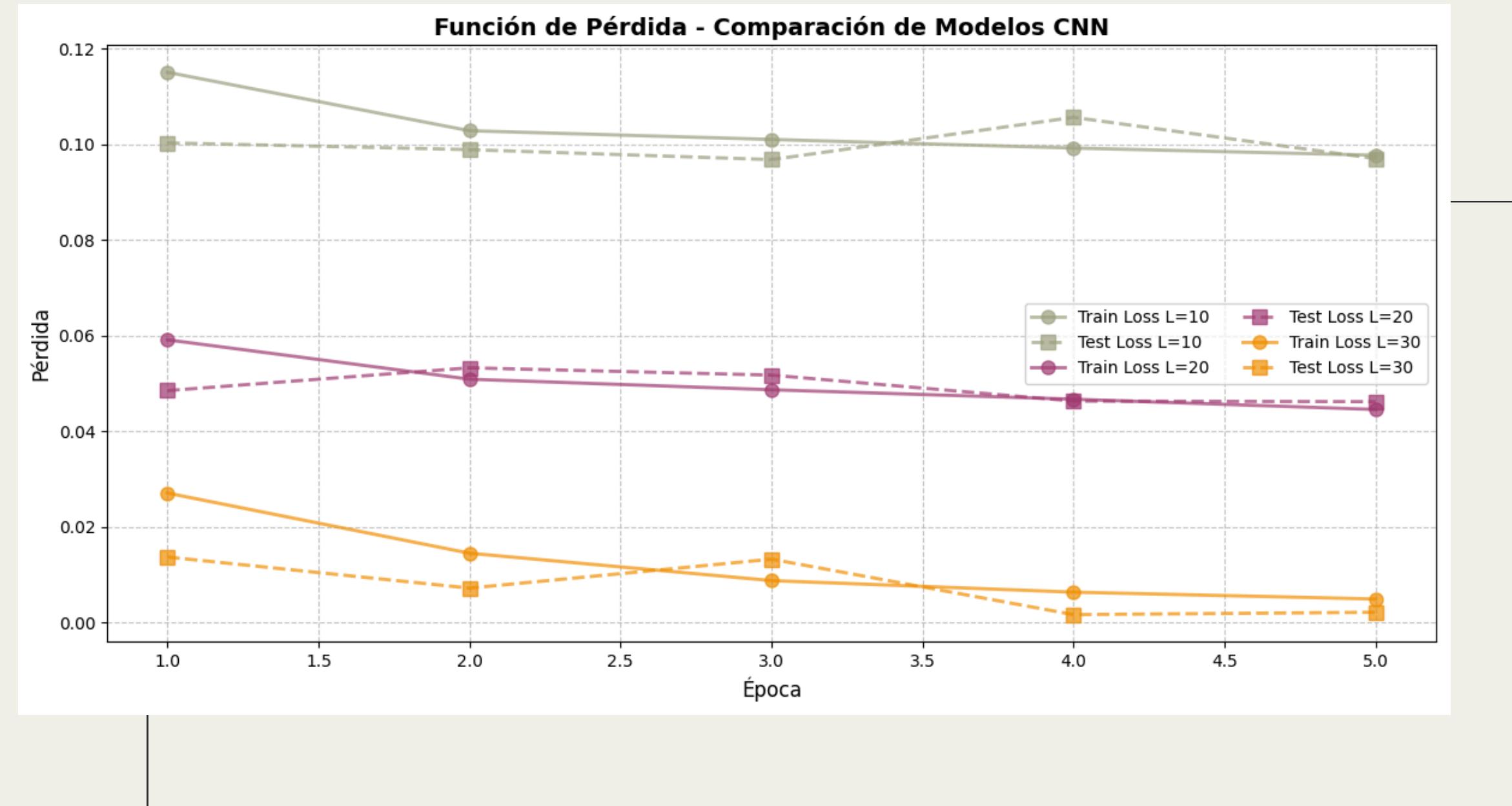


NEURAL NETWORK (CLASIFICACIÓN)

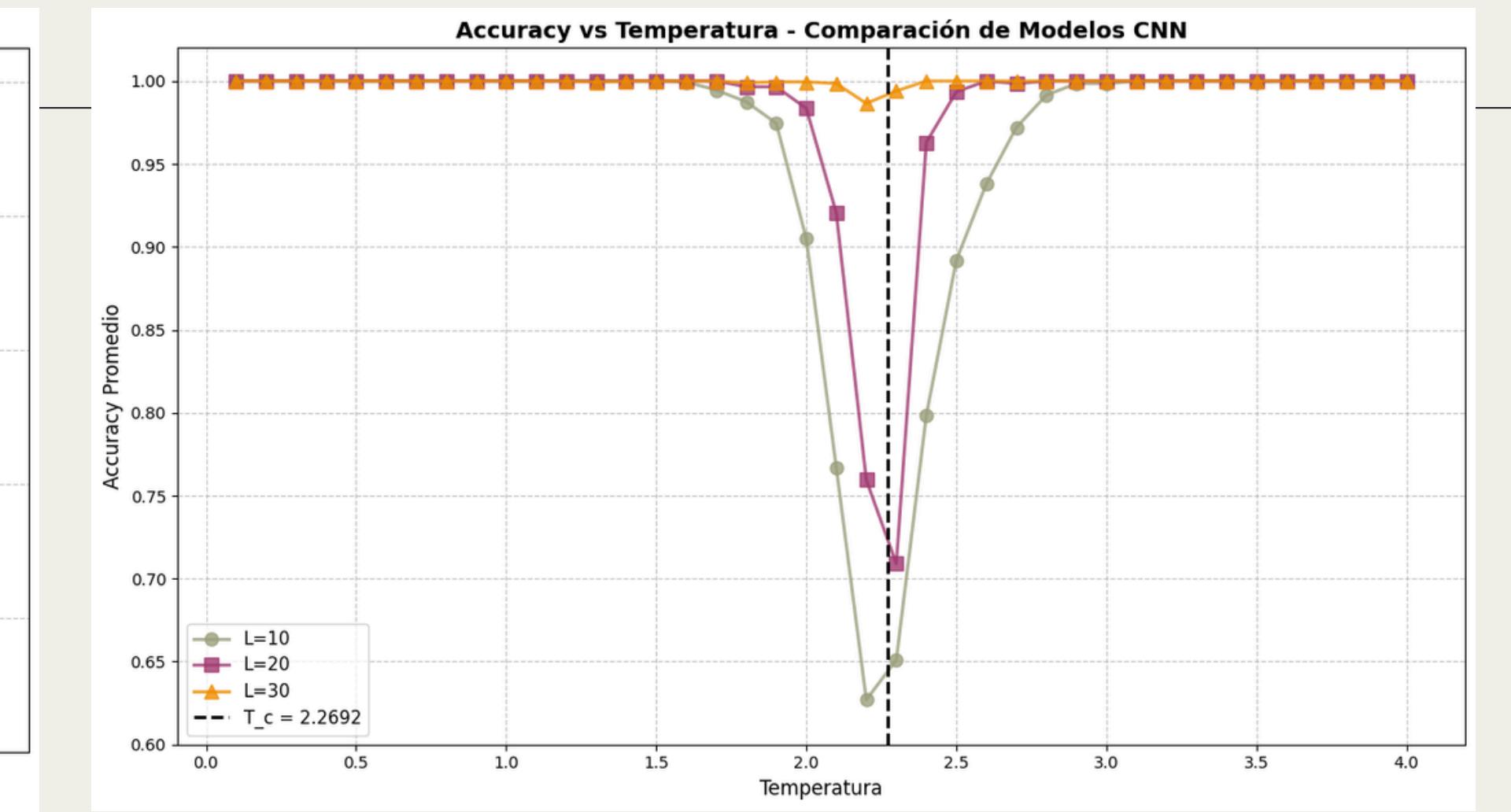
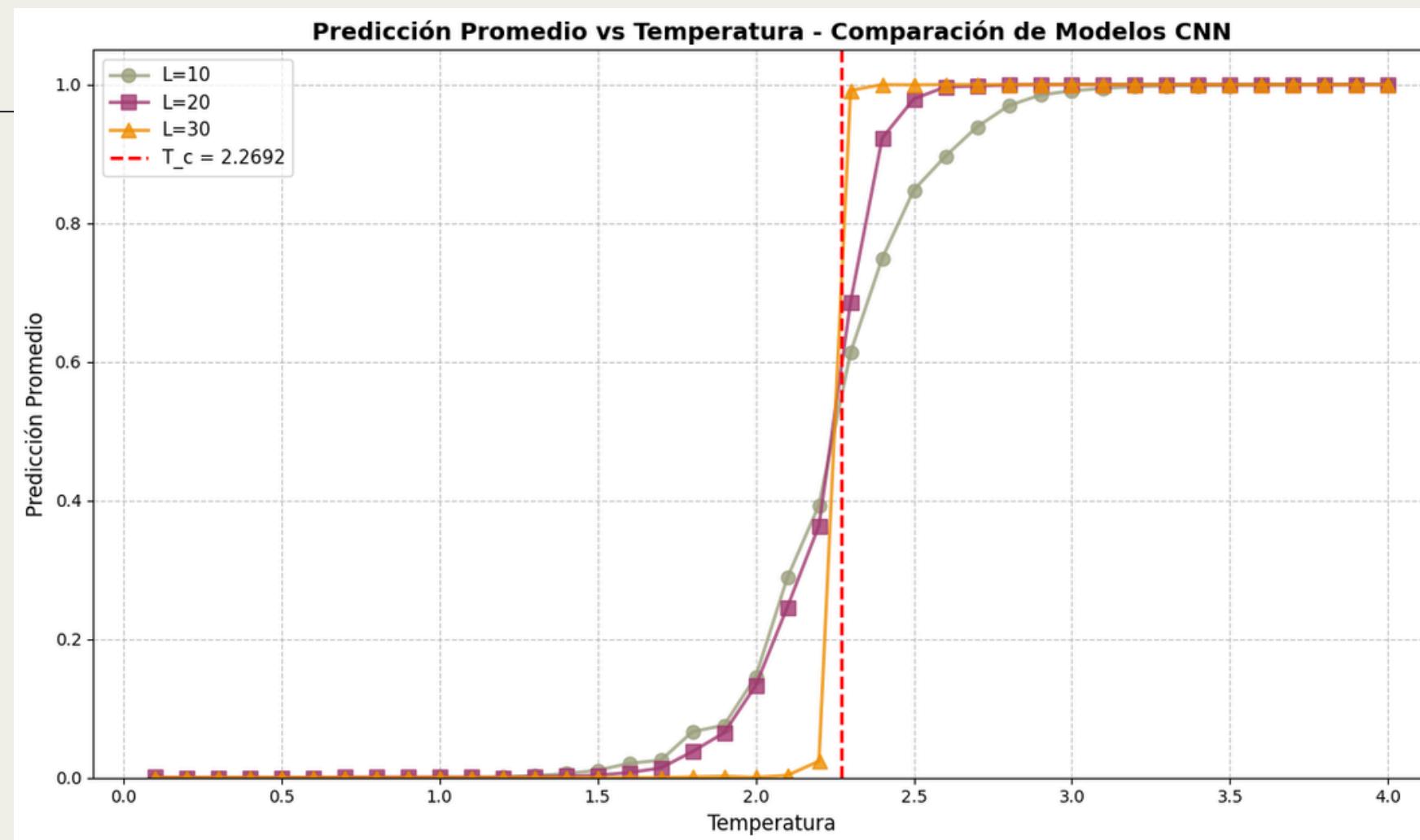


CNN (CLASIFICACIÓN)

- Red neuronal convolucional (CNN)
- Arquitectura tipo **AlexNet** (adaptada)
- Entrada: configuraciones de spins 2D ($n \times n$)
- Capas **Conv2D (kernels 3x3) + ReLU**
- **MaxPooling** para reducción espacial
- Capas **fully connected** finales para clasificación

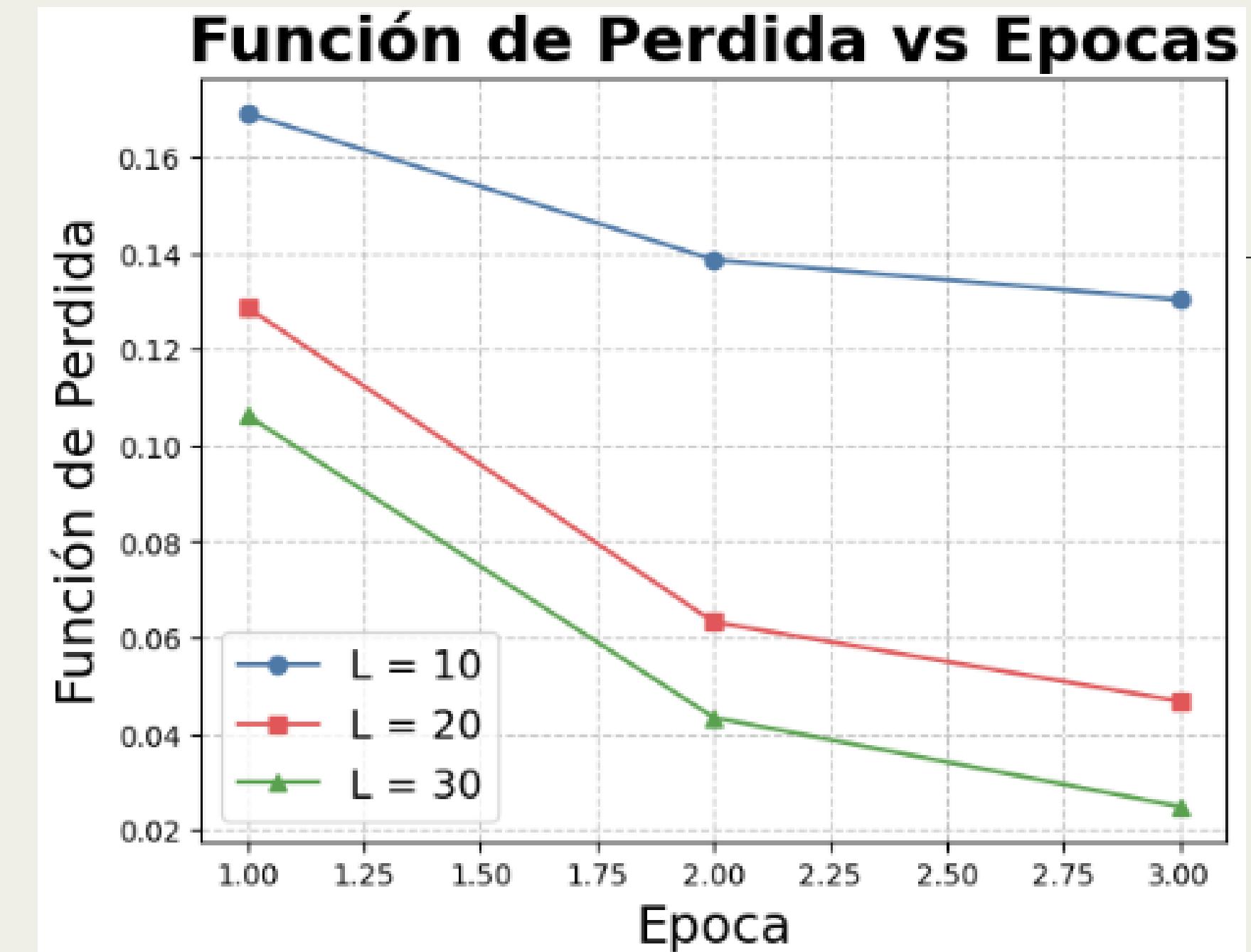


CNN (CLASIFICACIÓN)

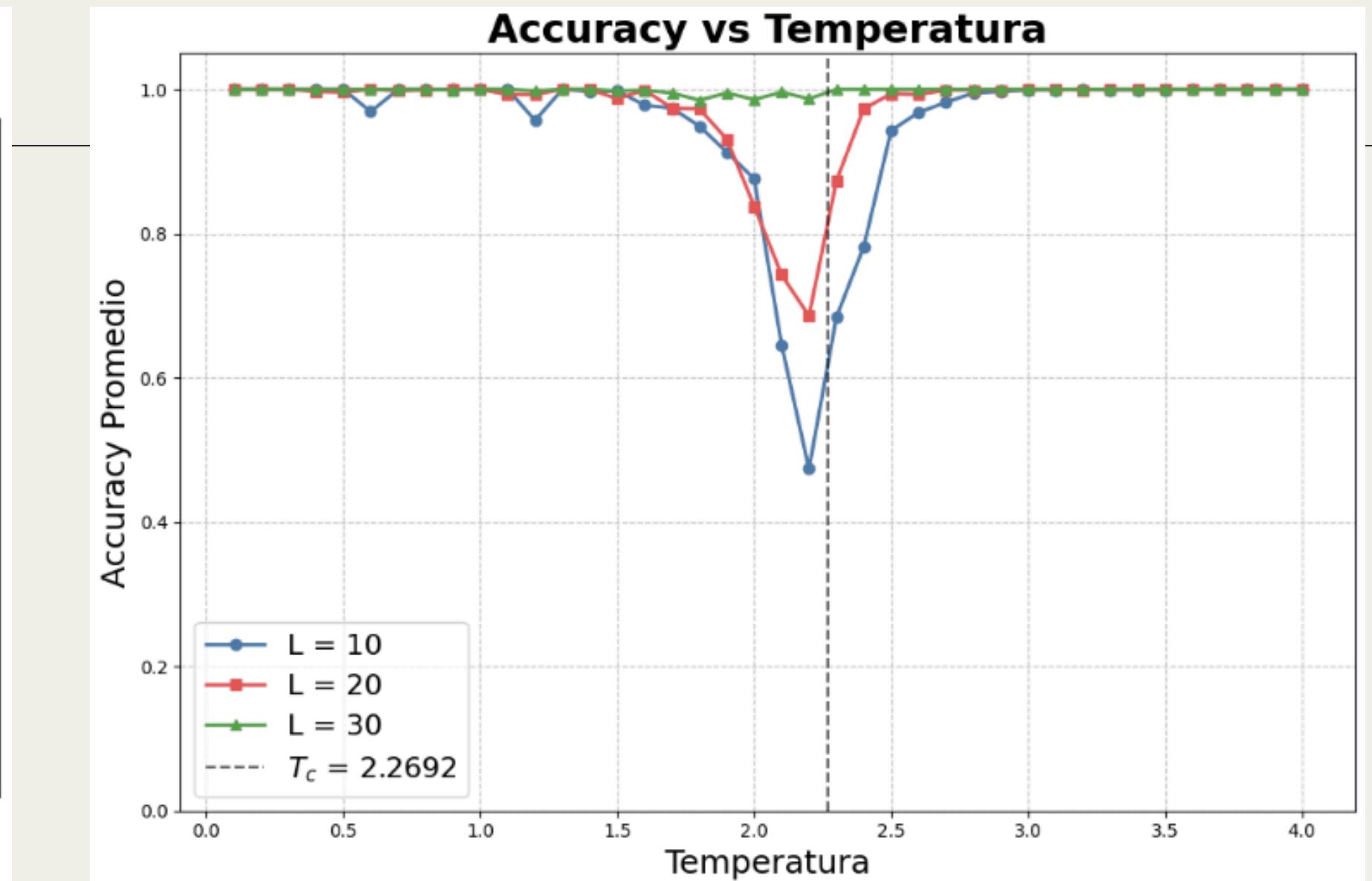
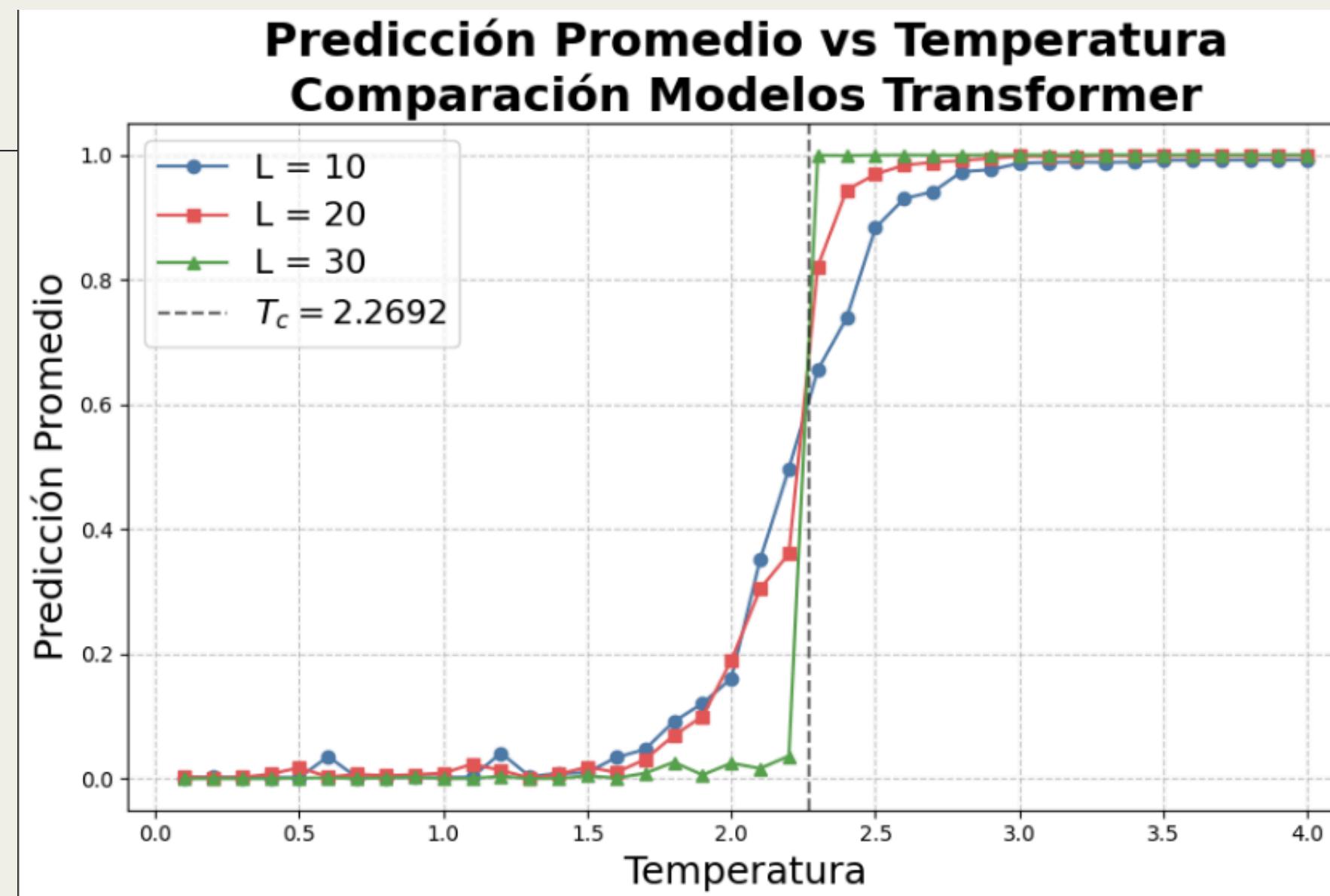


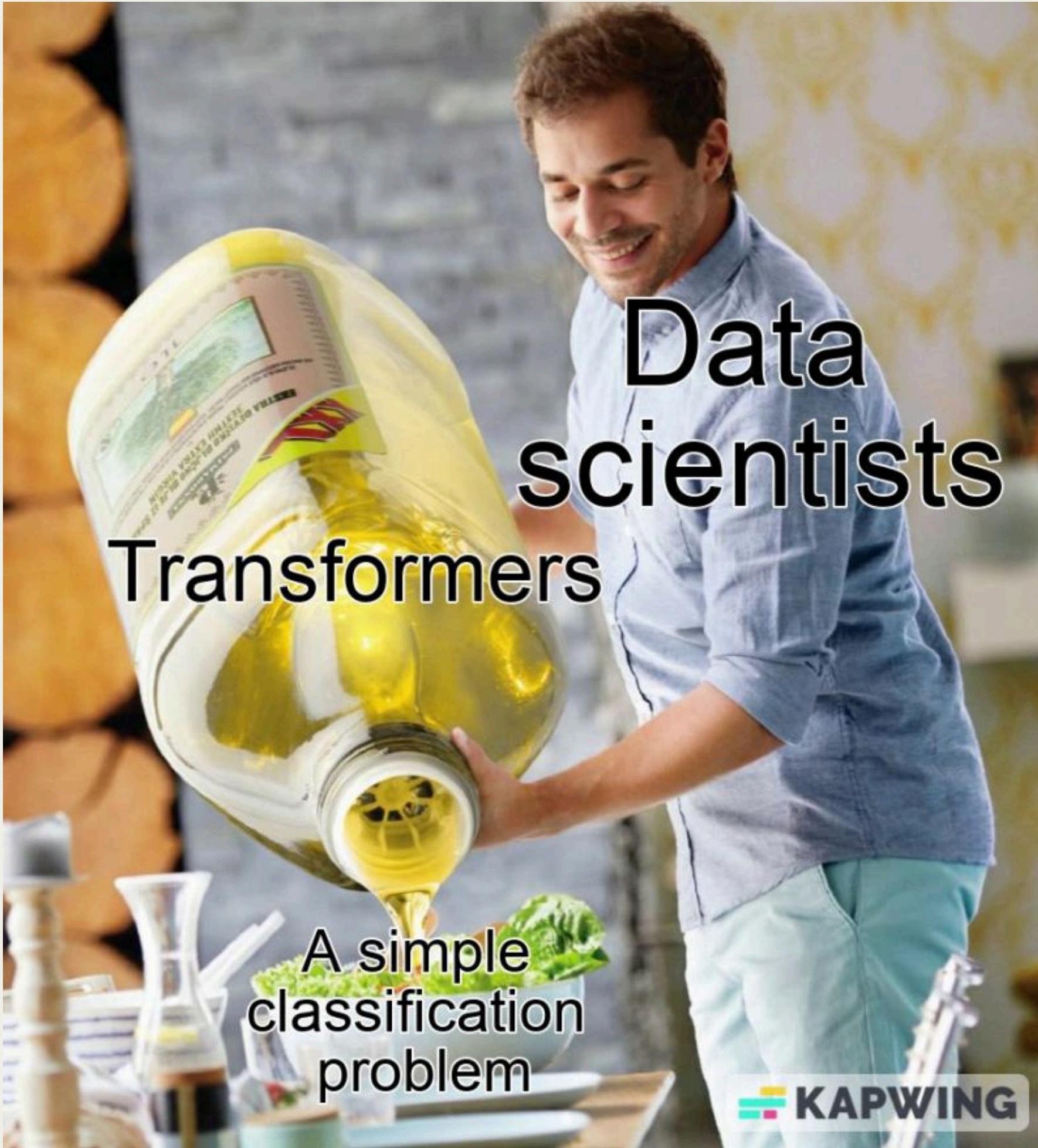
TRANSFORMER (CLASIFICACIÓN)

- **Vision Transformer (ViT):**
- División de las configuraciones de la red en parches 10x10
- **Embedding Convolucional + Positional Embedding** sobre los parches de cada configuración da el conjunto de entrada al modelo.
- El modelo es **Transformer Encoder** compuesto por dos capas.
- **Global Average Pooling** para la clasificación usando modelo lineal.
- Los parámetros se optimizan conjuntamente con el optimizador Adam, minimizando la función de pérdida de **Cross Entropy**.



TRANSFORMER (CLASIFICACIÓN)



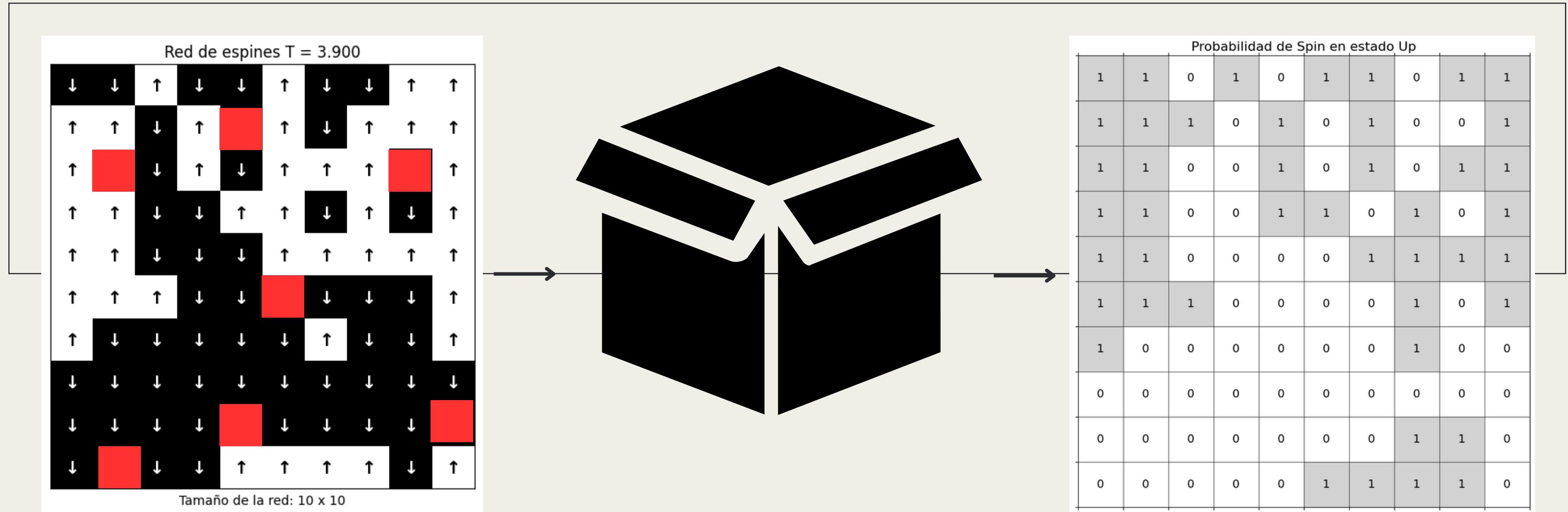


Data
scientists
Transformers

A simple
classification
problem

 KAPWING

PROBLEMA GENERATIVO

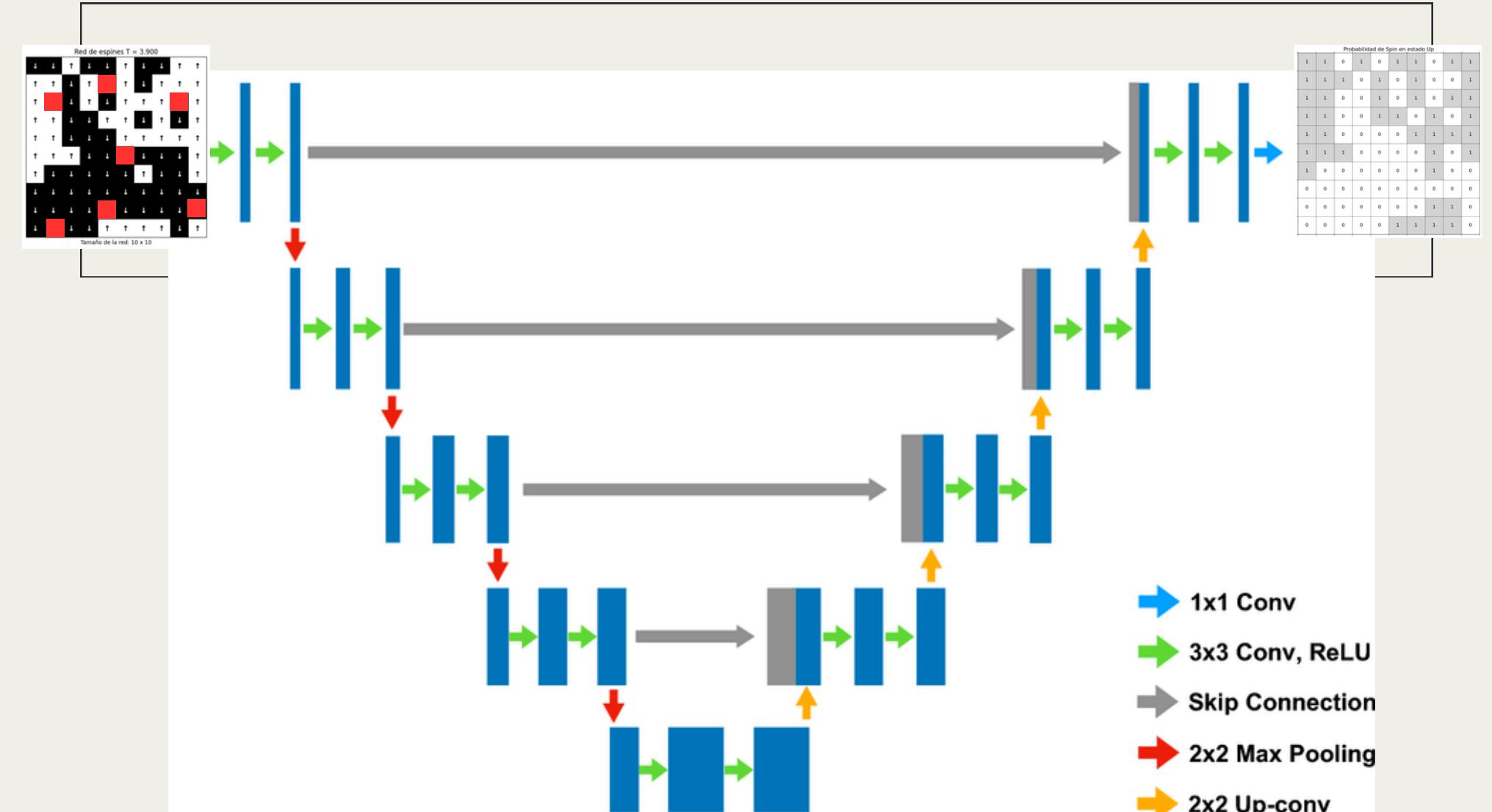


Espines imputados de manera aleatoria
(5%)

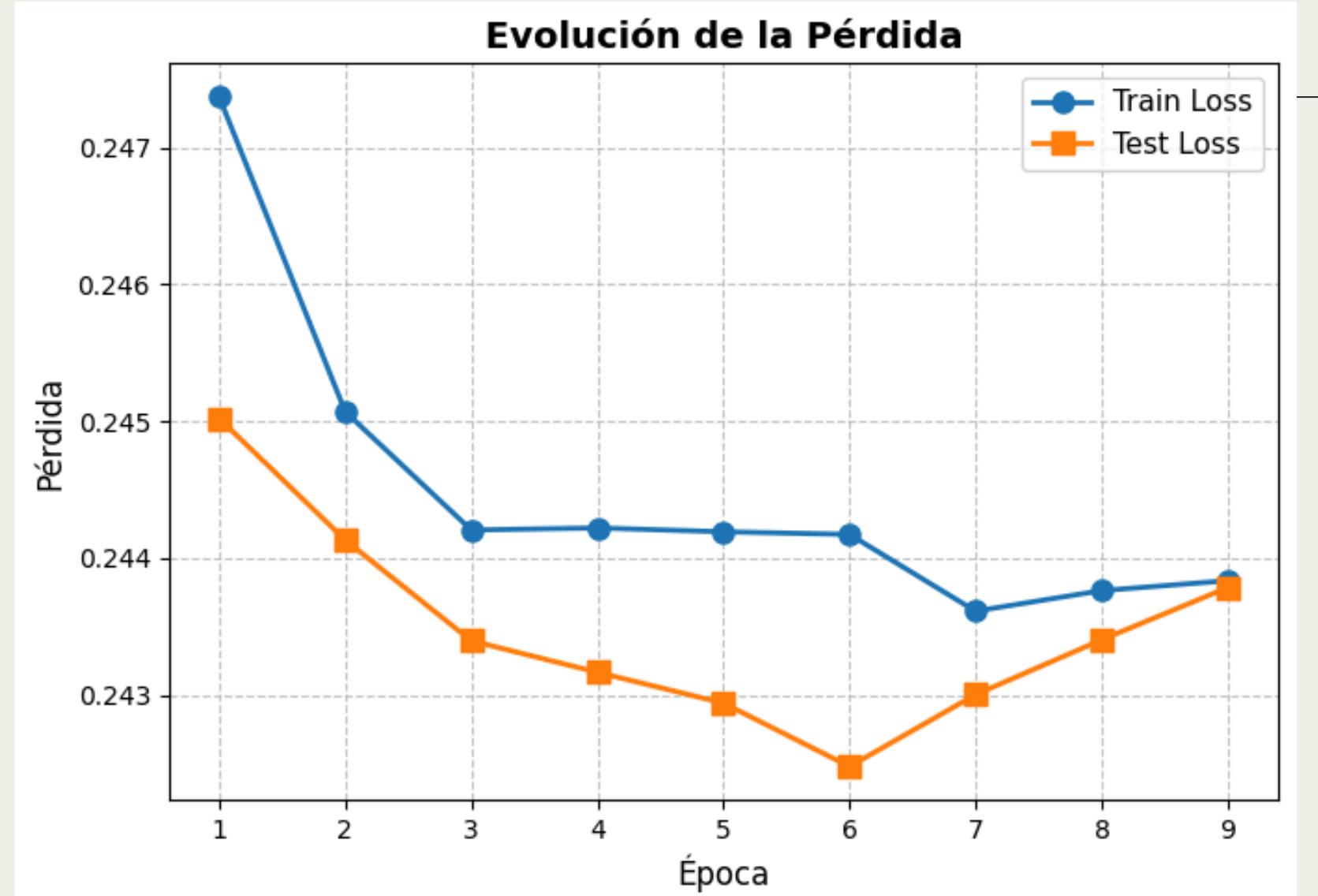
Modelo

Probabilidad de obtener Spin en estado
“Up”

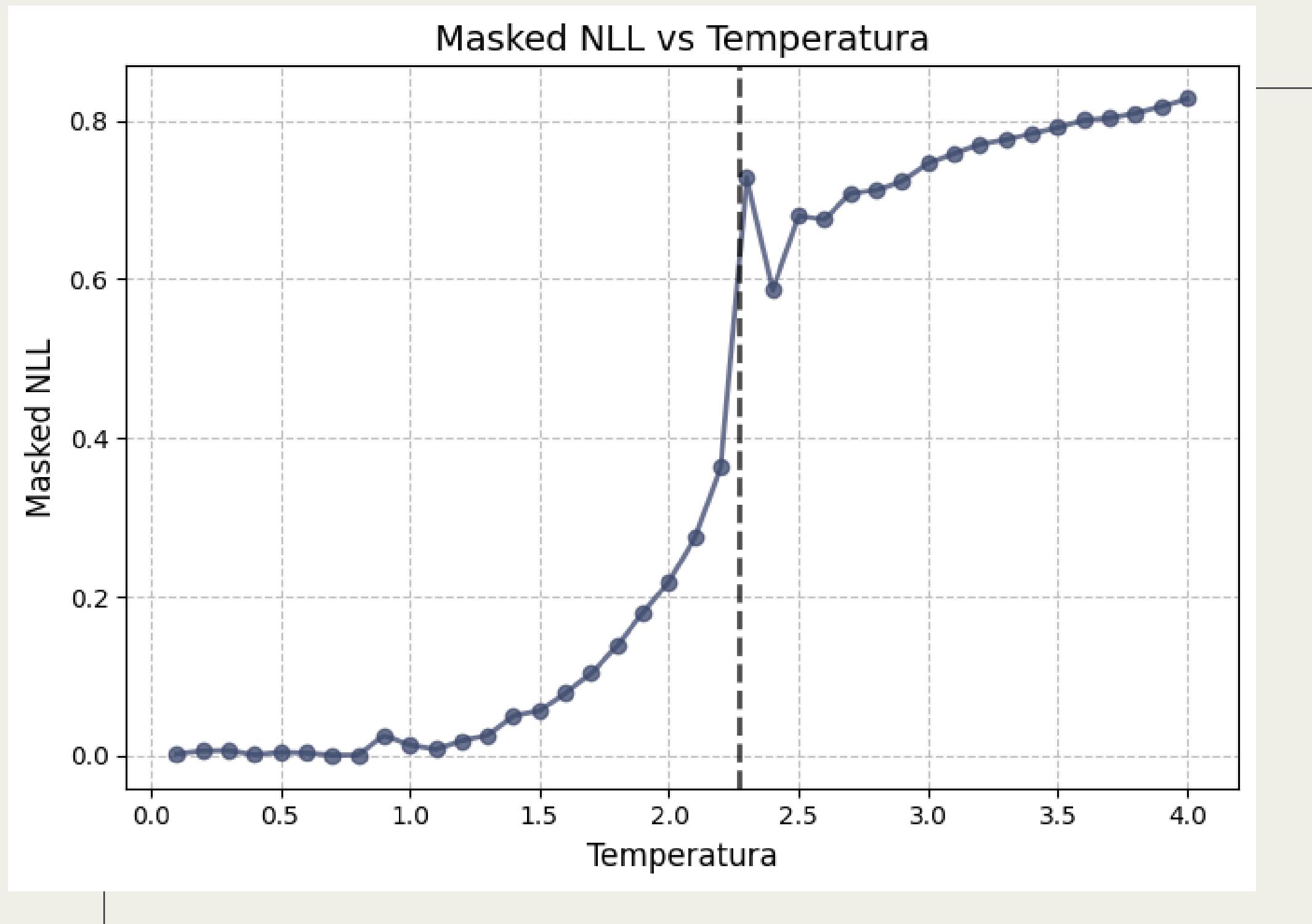
U - N E T



Red neuronal convolucional (CNN) tipo U-NET

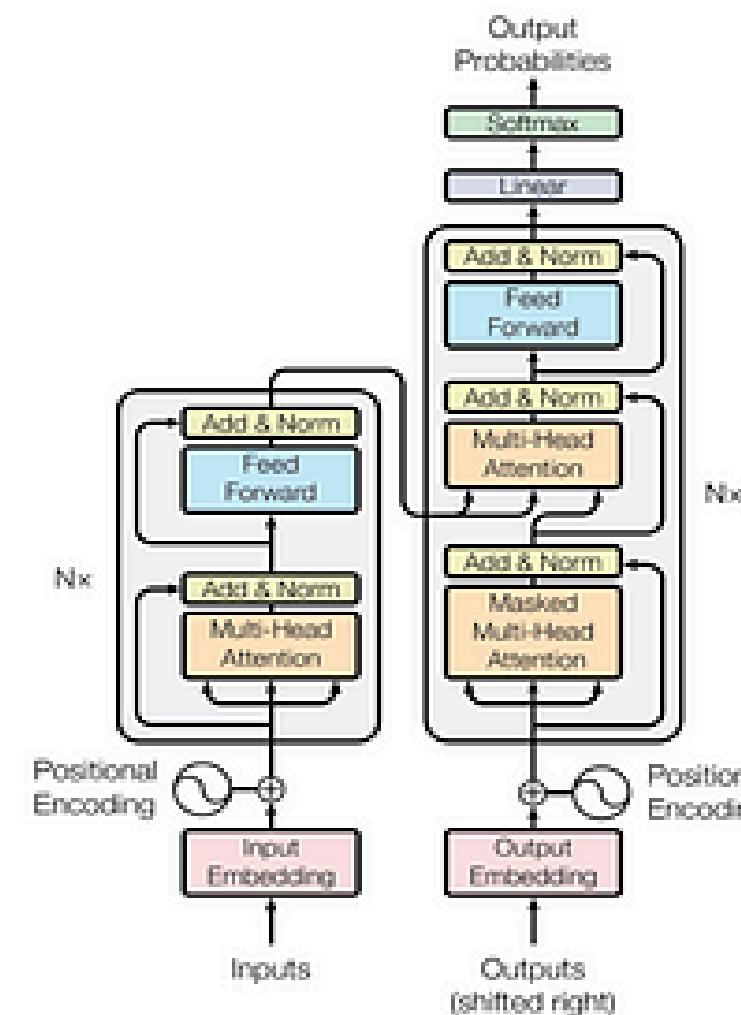


U - N E T



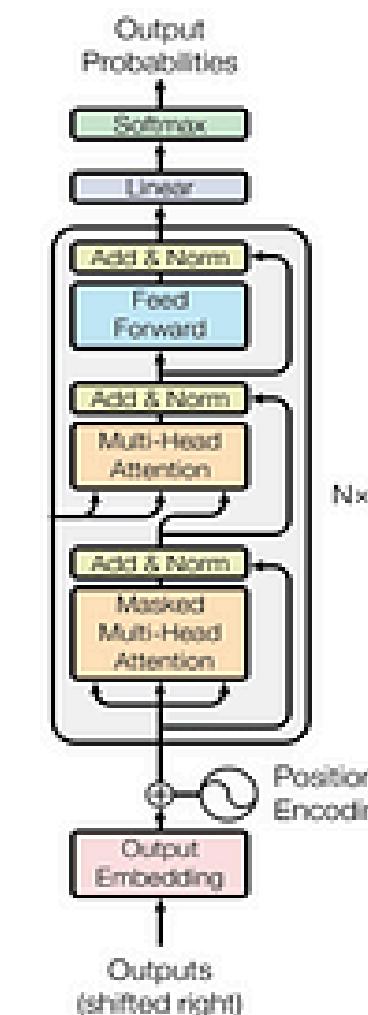
TRANSFORMER

Transformer



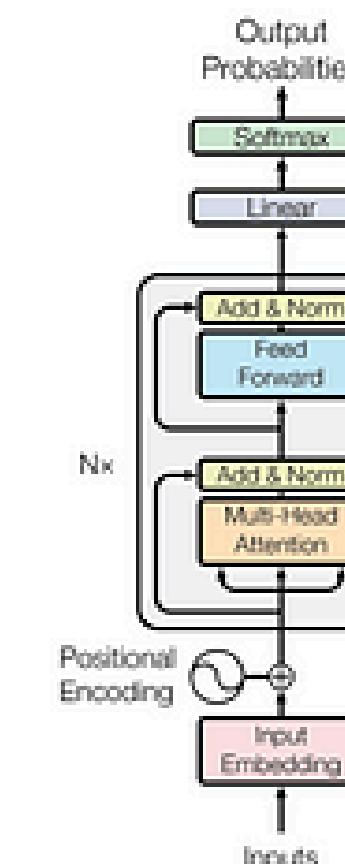
Encoder

GPT*



Decoder-only

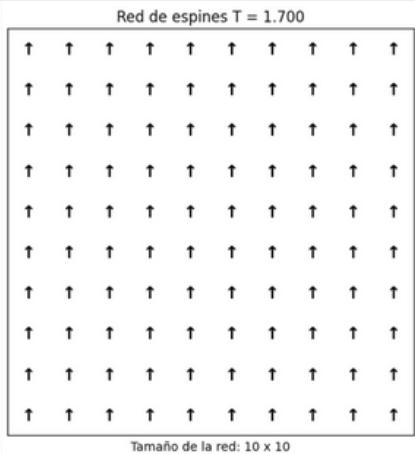
BERT*



Encoder-only

*Illustrative example, exact model architecture may vary slightly

TRANSFORMER



Soy un estudiante de ingeniería de 24 años que quiere entrar al campo de la ciencia de datos. Tengo Python intermedio, estadística básica y pocos recursos de hardware. ¿Qué tecnologías, lenguajes y herramientas me recomiendas aprender y en qué orden?

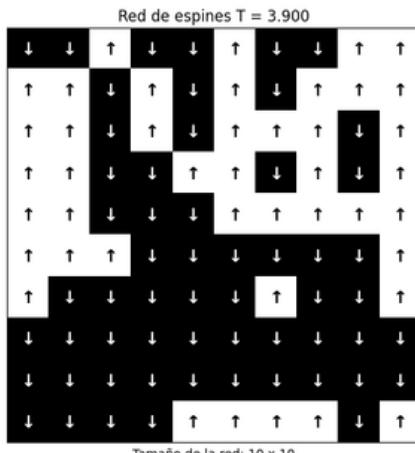
+

¡Buen perfil para empezar en ciencia de datos! 🎉 Con Python intermedio, estadística básica y poco hardware, lo ideal es **priorizar fundamentos, herramientas eficientes y trabajo en la nube gratuita**. Te propongo un **caminio realista y ordenado**, pensado para empleabilidad.

1 Fundamentos imprescindibles (reforzar primero)

⌚ 1–2 meses

📊 Estadística y matemáticas (orientadas a DS)



astronomia pan guitarra codigo selva memoria quantum bicicleta historia cafe algoritmo nube
montaña susurro economia ritual pixel viento teatro luna. neuronas oceano cortafuegos poema receta
galaxia mercado paciencia depurar reloj tinta bosque learning marmol justicia arquitectura energia

+

Bajo la luna, la **astronomía** afina su **guitarra**,
un **código** antiguo despierta en la **selva** de la **memoria quantum**.
Una **bicicleta** cruza la **historia** con aroma de **café**,
mientras un **algoritmo** sueña en la **nube** sobre la **montaña**.

Muchas gracias!

¿ALGUNA PREGUNTA?