UNIVERSIDAD DEL VALLE DE GUATEMALA

Data Science

Sección 30

LABORATORIO 3

Deep Learning

José Emilio Reyes Paniagua, 22674
Michelle Angel de María Mejía Villela, 22596
Silvia Alejandra Illescas Fernández, 22376

Guatemala, 02 de agosto del 2025

Contenido

Introducción

El reconocimiento de caracteres manuscritos es un problema clásico en el campo del aprendizaje automático, cuya resolución ha evolucionado significativamente con el desarrollo de técnicas de Deep Learning. En este laboratorio, se trabajó con el dataset PolyMNIST, una versión extendida del clásico MNIST, que incluye múltiples modalidades con distintos fondos visuales para aumentar la complejidad del reconocimiento. El objetivo principal fue diseñar, entrenar y evaluar distintos modelos capaces de identificar correctamente los dígitos presentes en las imágenes, aplicando desde modelos de redes neuronales simples hasta redes convolucionales más complejas. Además, se implementaron técnicas de data augmentation y se probaron imágenes manuscritas generadas por los integrantes del grupo para evaluar la robustez de los modelos desarrollados. Este informe presenta el análisis exploratorio del dataset, el preprocesamiento aplicado, la arquitectura y desempeño de los modelos entrenados, y una comparación entre sus resultados.

Análisis Exploratorio

Para comprender la estructura y características del dataset PolyMNIST, se realizó un análisis exploratorio que permitió identificar la distribución de las clases, la organización por modalidades y la resolución de las imágenes. Este paso es fundamental para detectar posibles sesgos, verificar el balanceo de las clases y asegurarse de que el conjunto de datos está en condiciones óptimas para el entrenamiento de modelos de Deep Learning.

El dataset está organizado en cinco modalidades distintas (m0 a m4), cada una compuesta por imágenes de dígitos manuscritos superpuestos a diferentes tipos de fondos visuales. Como se observa en la Figura 1, cada modalidad contiene 60,000 imágenes de entrenamiento distribuidas uniformemente entre las diez clases de dígitos (del 0 al 9), lo cual confirma que el dataset está completamente balanceado. Esto es clave para evitar sesgos en el entrenamiento del modelo hacia alguna clase específica.

Asimismo, se comprobó que todas las imágenes tienen una resolución uniforme de 28×28 píxeles, lo cual facilita la preparación de los datos y permite utilizar arquitecturas estándar de redes convolucionales sin necesidad de redimensionamiento adicional.

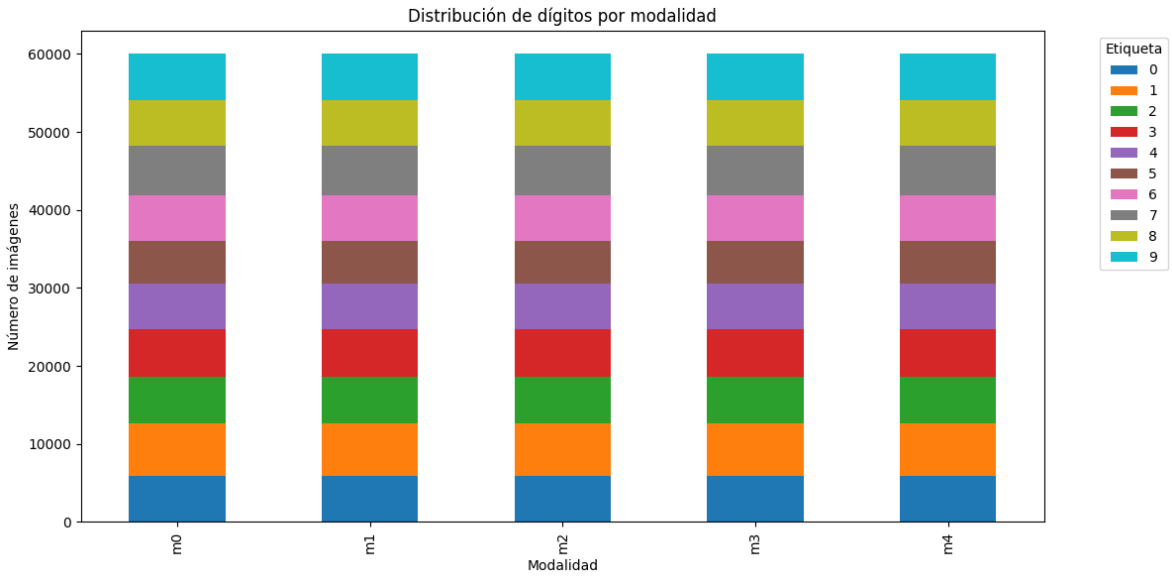


Figura 1. Distribución de dígitos por modalidad

Por otro lado, en la Figura 2 se muestra la distribución de imágenes entre los conjuntos de entrenamiento (train) y prueba (test) para cada modalidad. Se observa que cada modalidad cuenta con 60,000 imágenes para entrenamiento y 10,000 para prueba, lo que evidencia una división estándar de datos (aproximadamente 85%–15%) que favorece la evaluación objetiva del rendimiento del modelo sin sobreajuste.

Este análisis exploratorio permitió concluir que el conjunto PolyMNIST presenta características ideales para el desarrollo de modelos de clasificación robustos, ya que ofrece una gran cantidad de ejemplos por clase, equilibrio en la distribución de etiquetas y consistencia en las resoluciones. Además, la presencia de múltiples modalidades representa un reto interesante, pues obliga a los modelos a generalizar más allá del estilo visual del fondo, enfocándose en la forma del dígito.

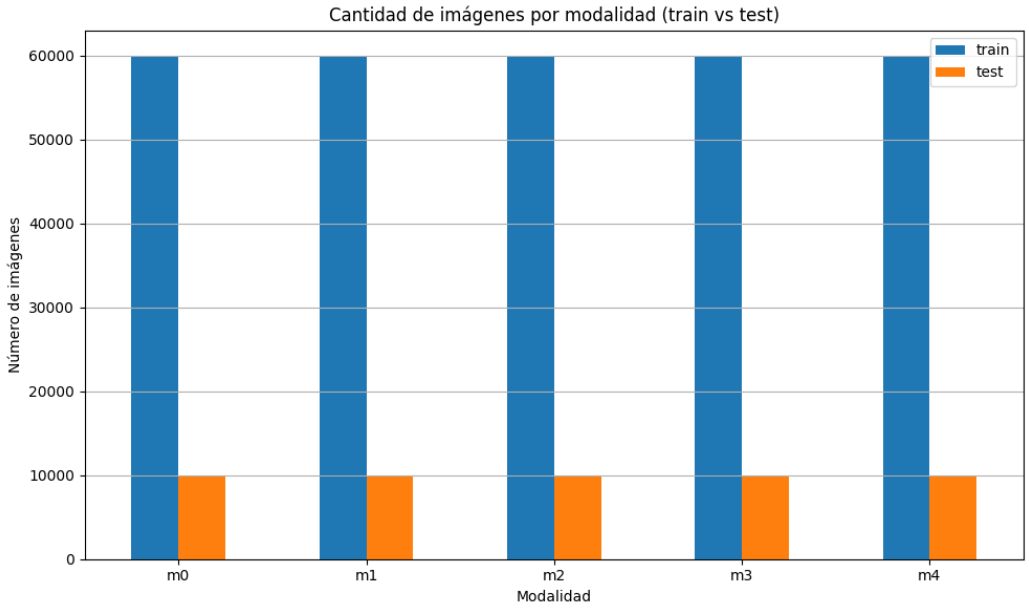


Figura 2. Cantidad de imágenes por modalidad

Adicionalmente, se calculó la intensidad promedio de píxeles por clase dentro de la modalidad m0. Todas las clases presentan una intensidad promedio cercana a 0.5 (en una escala de 0 a 1), lo cual sugiere que el nivel de brillo o saturación de los dígitos es bastante uniforme entre etiquetas. Esta homogeneidad es deseable, ya que indica que no hay clases sistemáticamente más oscuras o más claras que otras, lo cual podría introducir un sesgo en el entrenamiento.

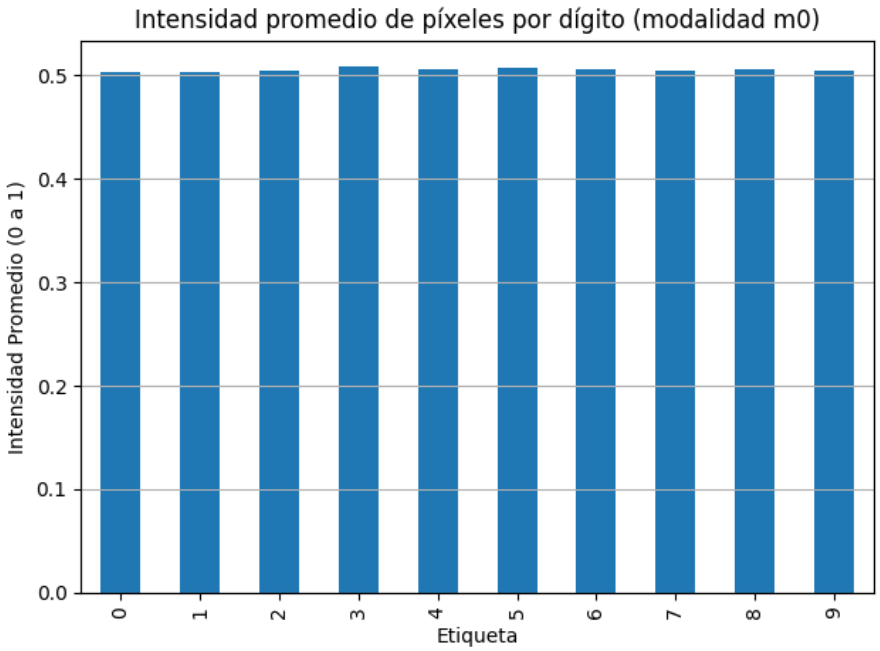


Figura 3. Intensidad promedio de pixeles

Finalmente, se analizó el tamaño promedio de los dígitos en términos de la cantidad de píxeles oscuros por clase en la modalidad m0. Los resultados muestran una distribución bastante uniforme, con valores que oscilan alrededor de los 610 píxeles. Esto indica que los dígitos ocupan áreas similares dentro de las imágenes, lo cual es favorable para el entrenamiento, ya que evita que el modelo aprenda patrones relacionados al tamaño en lugar de la forma del dígito.

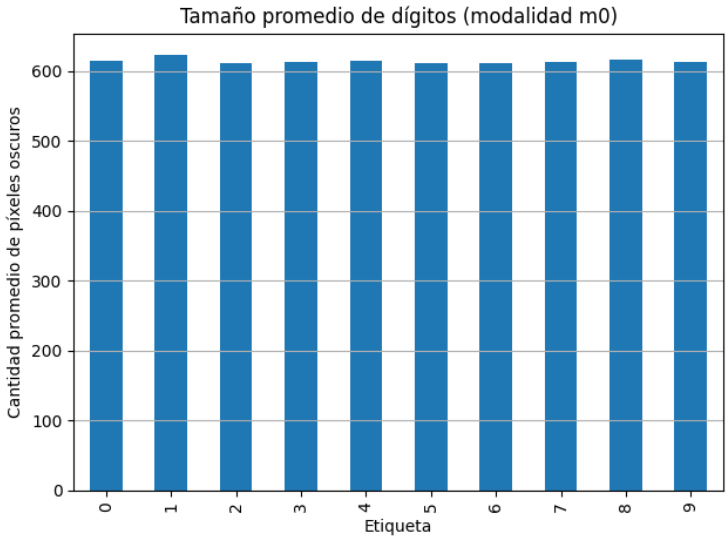


Figura 4. Tamaño promedio de dígitos

Preprocesamiento de las Imágenes

Antes de entrenar los modelos, fue necesario realizar un preprocesamiento de los datos con el objetivo de estandarizar las entradas y facilitar el aprendizaje. Uno de los primeros pasos fue la inspección visual de las distintas modalidades del dataset PolyMNIST. En la Figura 5 se muestran ejemplos del dígito '5' en cada una de las modalidades (m0 a m3), lo cual permite observar las variaciones de fondo, color, contraste e iluminación que podrían dificultar el reconocimiento por parte de modelos simples.

Figura 5. Dígito 5 en cada modalidad

A pesar de que todas las imágenes mantienen una resolución uniforme de 28×28 píxeles, las diferencias en los fondos introducen ruido visual. Por esta razón, se aplicaron las siguientes técnicas de preprocesamiento:

Conversión a escala de grises: Aunque algunas modalidades contienen imágenes en color, convertirlas a escala de grises reduce la dimensionalidad sin afectar la forma del dígito, que es el rasgo más importante para la clasificación.

Normalización: Los valores de los píxeles fueron escalados al rango [0, 1], dividiendo cada valor entre 255. Esta práctica es estándar en modelos de aprendizaje profundo, ya que mejora la estabilidad numérica durante el entrenamiento.

Ajuste del canal de entrada: Se garantizó que cada imagen tuviera una forma compatible con las redes neuronales, transformándolas en tensores de tamaño (1, 28, 28) para imágenes en blanco y negro, o (3, 28, 28) en caso de usar color.

Revisión de etiquetas y nombres de archivo: Se extrajo la etiqueta del dígito a partir del nombre del archivo (formato XXXX.Y.PNG) para asegurar una correcta asociación imagen-etiqueta durante la carga.

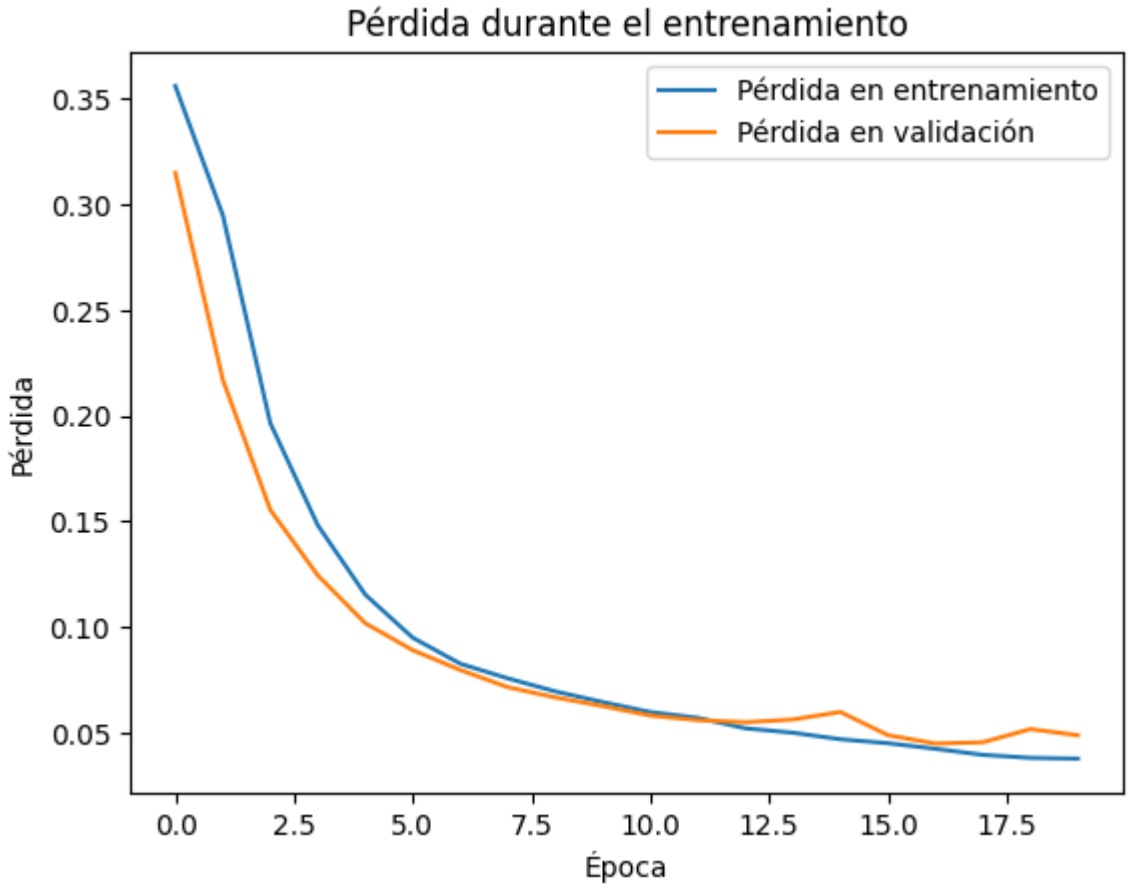
Estas transformaciones buscan homogeneizar el conjunto de datos de entrada, reducir el impacto del ruido introducido por los fondos aleatorios, y permitir que el modelo se enfoque en los patrones estructurales del dígito. Este proceso sienta las bases para un entrenamiento eficiente y un rendimiento más robusto en escenarios reales.

Descripción de los Modelos

Se diseñaron dos modelos de redes neuronales convolucionales (CNN) para abordar el problema de clasificación de dígitos manuscritos utilizando únicamente la modalidad m0 del dataset PolyMNIST. Ambos modelos fueron construidos en Keras, utilizando la arquitectura secuencial y la función de pérdida categorical_crossentropy, con el optimizador Adam. La diferencia principal entre los dos modelos radica en la profundidad de la red y la cantidad de capas convolucionales utilizadas, lo cual se refleja directamente en su capacidad de extracción de características.

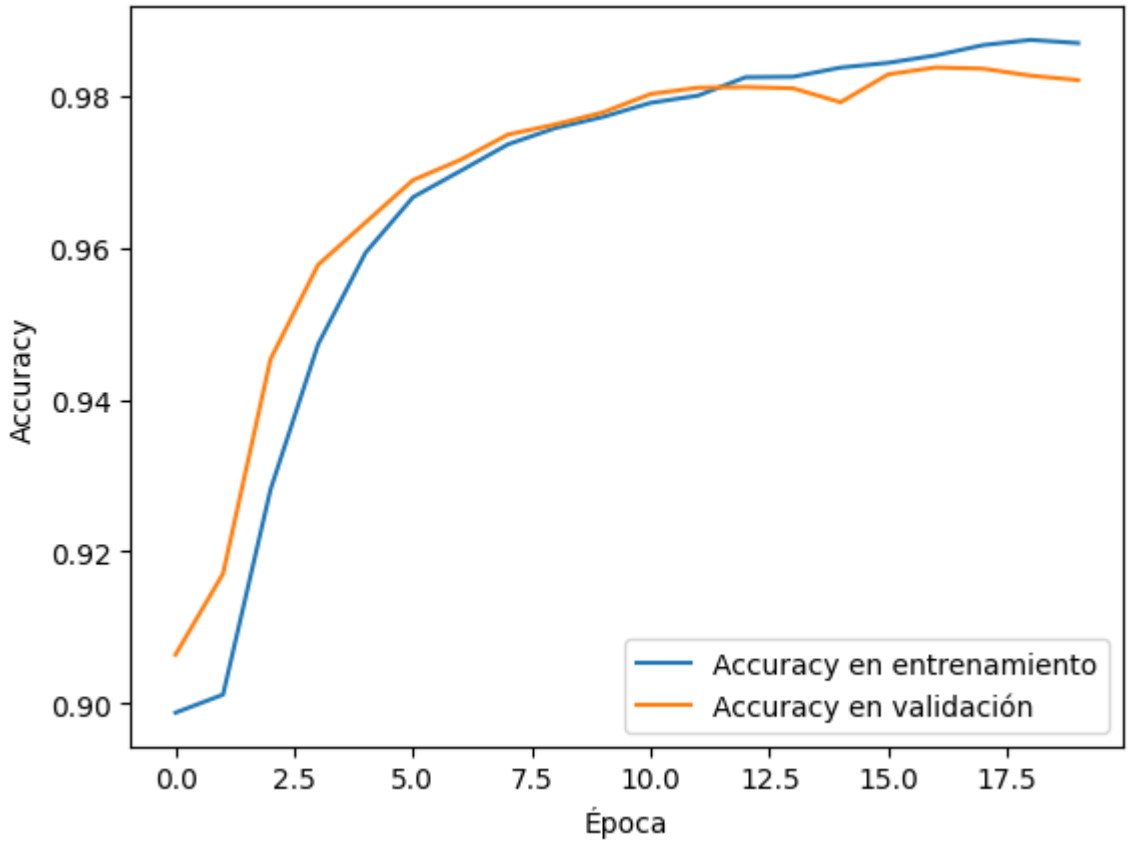
Modelo 1

El primer modelo implementado es una arquitectura CNN profunda compuesta por ocho capas convolucionales distribuidas en bloques, cada uno seguido de una capa de max pooling para reducir la dimensionalidad espacial. Esta red culmina en una capa fully connected densa de 128 neuronas antes de la capa de salida softmax. Esta arquitectura busca capturar patrones de distinta complejidad y fue diseñada para explotar al máximo las variaciones visuales presentes en las imágenes del conjunto de entrenamiento.



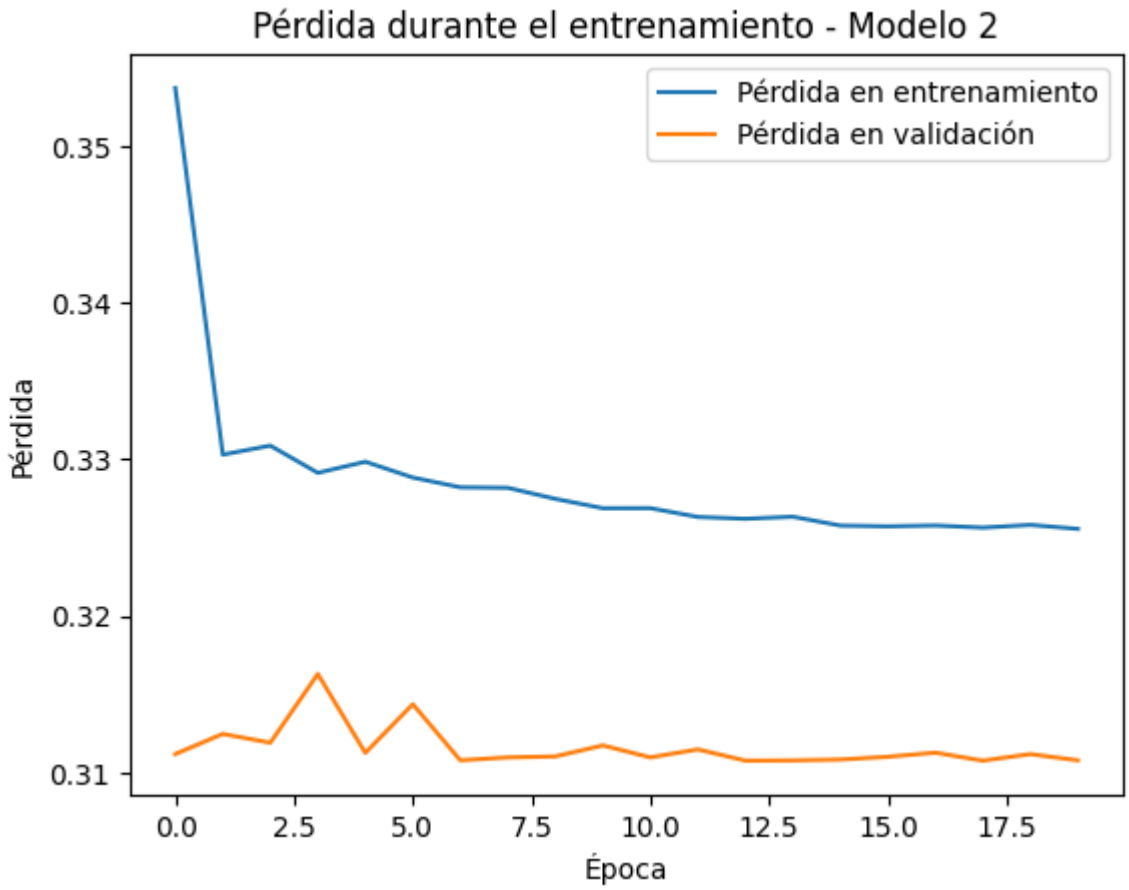
Efectividad

Durante el entrenamiento, el modelo presentó un desempeño altamente positivo. La precisión en entrenamiento y validación aumentó progresivamente a lo largo de las épocas, alcanzando valores cercanos al **98.5% en entrenamiento** y **98.2% en validación**, como se muestra en la figura correspondiente. La diferencia entre ambas curvas se mantuvo pequeña, lo cual indica que el modelo generaliza bien y no está sobreajustando. Este comportamiento también se refleja en la **precisión final sobre el conjunto de prueba**, que fue de **98.18%**, consolidando este modelo como una arquitectura robusta y eficaz para el reconocimiento de dígitos manuscritos en condiciones controladas.



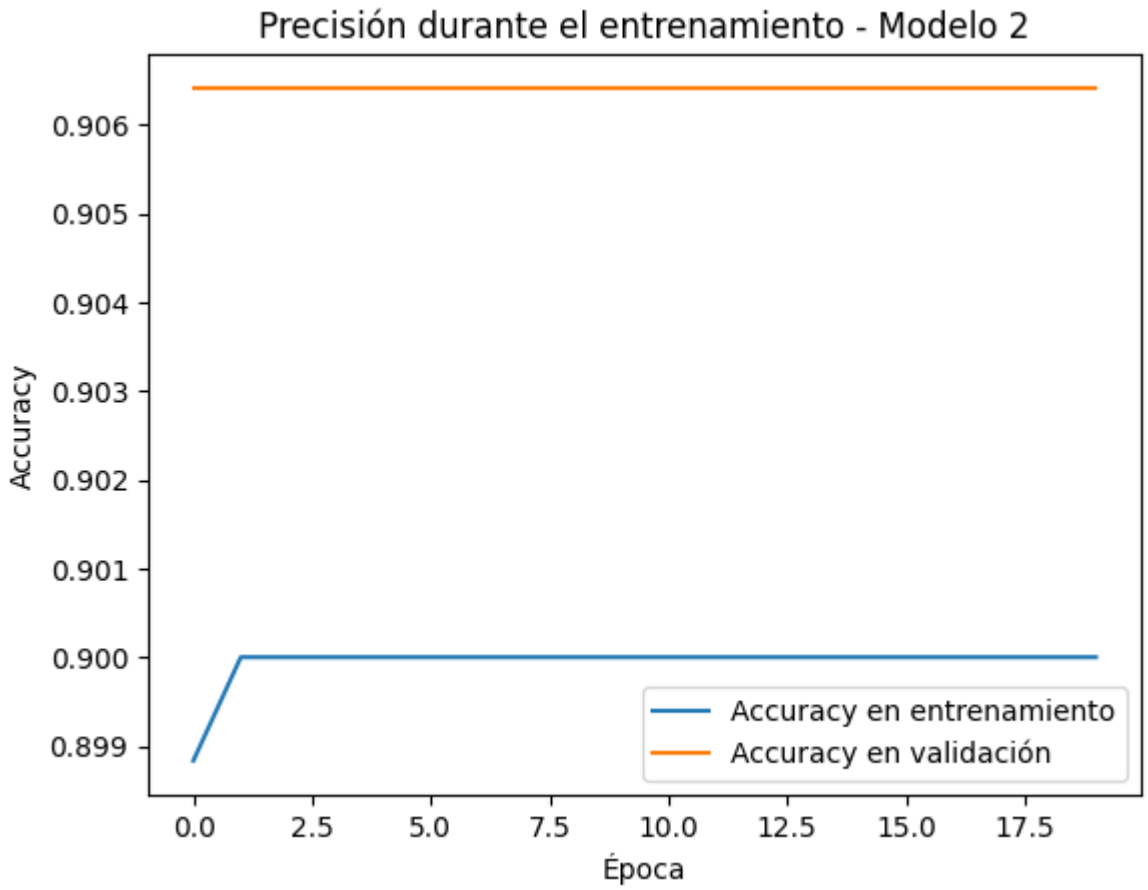
Modelo 2

El segundo modelo también es una red convolucional, pero con una arquitectura diferente enfocada en aumentar la profundidad y aplicar técnicas de regularización. Incluye cuatro capas convolucionales, seguidas por dos capas de pooling, una capa densa intermedia con 256 neuronas, y una capa de abandono (Dropout) con tasa 0.5 antes de la salida softmax. Esta modificación buscaba mejorar la generalización e incorporar más capacidad de representación, manteniendo una estructura más compacta que el modelo 1.



Efectividad

Durante el entrenamiento, el modelo mostró un comportamiento estable pero limitado. La pérdida se mantuvo constante en ambas curvas, y la precisión de validación se estancó en torno al 90.7%, mientras que la precisión final sobre el conjunto de prueba fue de 90.2%. La falta de mejora progresiva en las métricas y la baja diferencia entre entrenamiento y validación indican que el modelo probablemente llegó a su techo de rendimiento muy temprano, sin lograr explotar toda la información del conjunto de datos.



Comparación

Comparando ambos modelos, el modelo 1 superó ampliamente al modelo 2 tanto en precisión como en capacidad de aprendizaje. Mientras que el modelo 1 alcanzó una precisión de 98.2%, el modelo 2 quedó limitado al 90.2%, lo que representa una diferencia significativa en tareas de clasificación. Esto puede atribuirse a la menor cantidad de capas en el modelo 2, su arquitectura menos profunda, y posiblemente a una combinación de hiperparámetros menos óptimos. A pesar de incorporar Dropout, el modelo 2 no mostró signos de overfitting, lo que sugiere que su capacidad fue insuficiente para capturar patrones más complejos.

Discusión

Ambos modelos fueron entrenados utilizando únicamente la modalidad m0, lo que permitió evaluar su rendimiento en un entorno relativamente controlado. El modelo 1 demostró ser altamente efectivo para el reconocimiento de dígitos manuscritos, alcanzando resultados consistentes y generalizables. Por otro lado, el modelo 2, a pesar de tener más regularización, no logró superar el umbral del 91%, posiblemente por un diseño menos adecuado o una arquitectura subóptima para la complejidad del problema. Esta comparación resalta la

importancia de experimentar con distintas configuraciones arquitectónicas y no asumir que una mayor profundidad o regularización conducirá automáticamente a un mejor rendimiento.

Conclusiones

El modelo convolucional profundo (modelo 1) demostró ser significativamente más efectivo que el modelo regularizado (modelo 2) para la tarea de clasificación de dígitos manuscritos en la modalidad m0 del dataset PolyMNIST. Con una precisión superior al 98%, este modelo logró generalizar correctamente a nuevos datos, manteniendo una baja pérdida y sin señales de sobreajuste.

La arquitectura y profundidad de una red neuronal tienen un impacto directo en su capacidad de aprendizaje. Aunque el modelo 2 incorporó técnicas como Dropout para evitar el sobreajuste, su menor capacidad de representación limitó su desempeño, mostrando la importancia de equilibrar regularización, profundidad y complejidad en el diseño de modelos de Deep Learning.

Referencias

Chollet, F. (2018). Deep Learning with Python (1st ed.). Manning Publications.

Link al repositorio: https://github.com/JEmilioRey1021/Laboratorio-3-Deep-Learning.git

Link al documento Versionado: Laboratorio 3 - Data Science.docx