

Análisis exploratorio

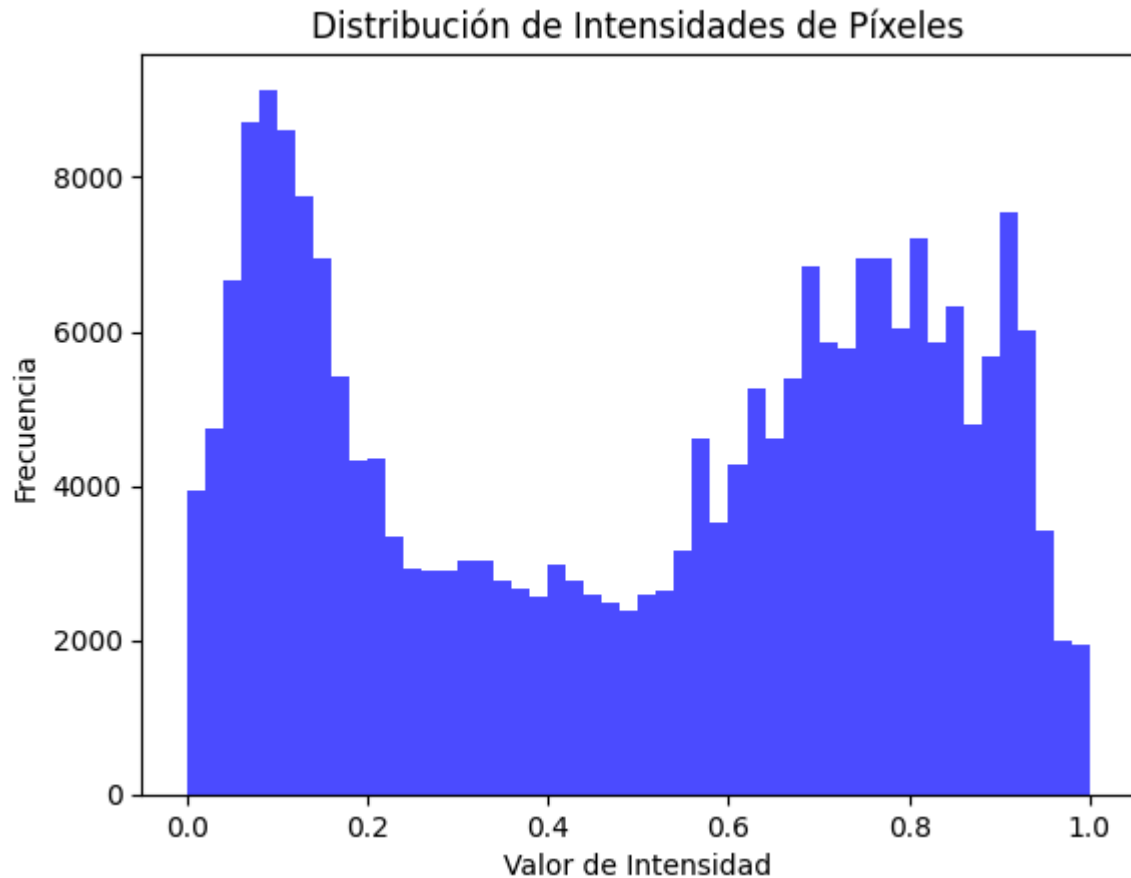
Las imágenes utilizadas para laboratorio poseían un tamaño 28x28 píxeles y contenían 3 canales de color. Las imágenes fueron almacenadas en formato PNG.

La distribución de imágenes se encontró perfectamente balanceada, cada modalidad tenía 60 mil imágenes distribuidas.

Preprocesamiento: Antes de ser alimentadas a los modelos de aprendizaje profundo, las imágenes fueron normalizadas dividiendo los valores de los píxeles entre 255. Esto con el fin de que los valores de los píxeles estén dentro del rango de 0 a 1.

Las imágenes están distribuidas en 5 categorías (m0, m1, m2, m3, m4), cada una representando un subconjunto de los datos. El conjunto de datos se dividió en entrenamiento y prueba.

Cada imagen está etiquetada con un valor que representa el dígito manuscrito que contiene (de 0 a 9). Las etiquetas fueron convertidas a formato **one-hot encoding**. Cada etiqueta se representa como un vector binario de longitud 10, con un 1 en la posición correspondiente al dígito representado por la imagen.

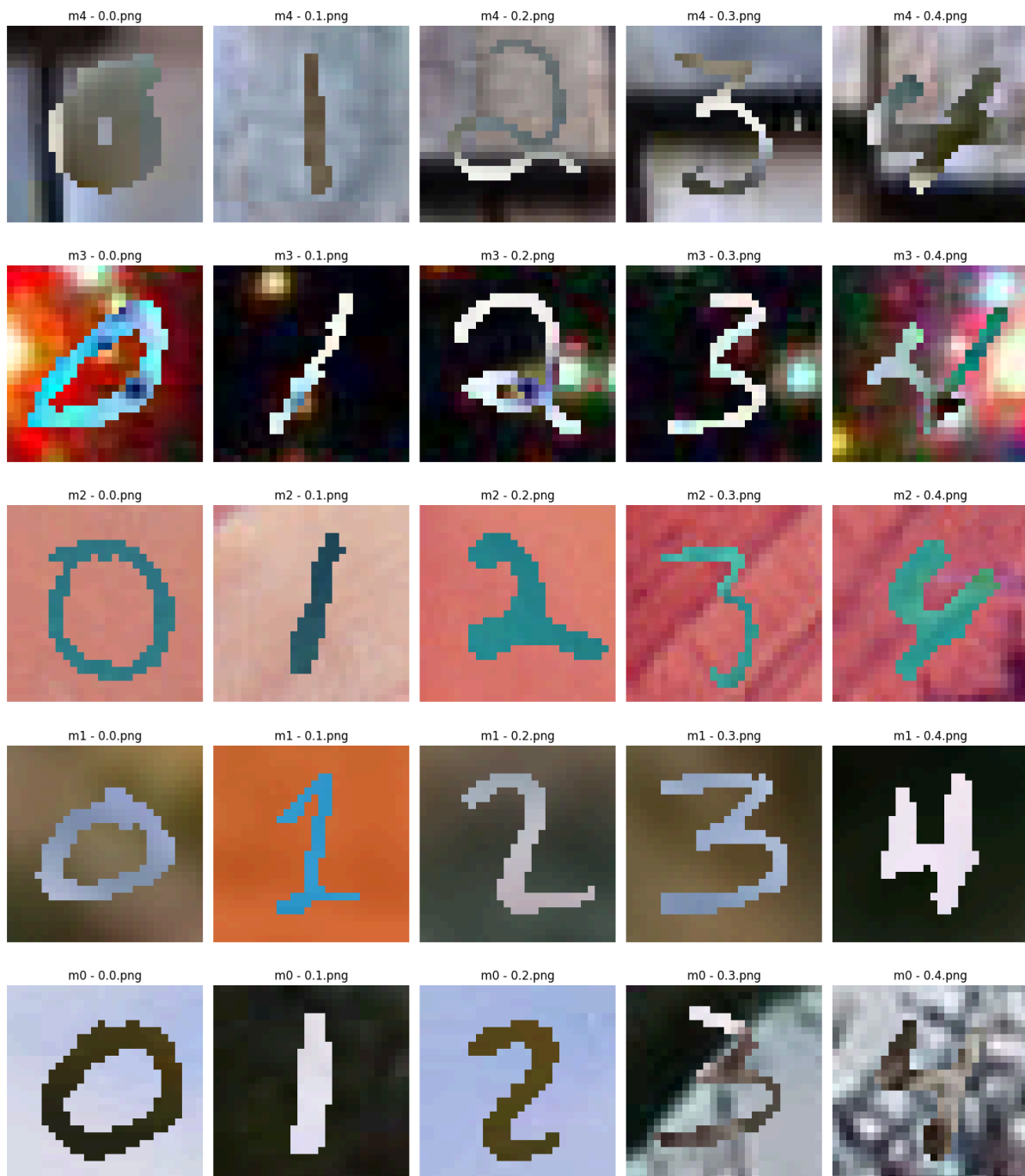


Valores bajos (cerca de 0): Vemos que la mayor parte de los píxeles en las imágenes tienen valores de intensidad bajos, es decir, lo que indica que hay muchas áreas oscuras en las imágenes.

Valores altos (cerca de 1): También se observa una alta frecuencia de valores cercanos a 1, lo que indica la presencia de áreas brillantes en las imágenes.

Valores intermedios: Hay una menor cantidad de píxeles con intensidades intermedias (entre 0.3 y 0.6).

Ejemplo de modalidades:



Modelos Deep Learning:

En el **modelo simple**, se diseñó una red convolucional con una arquitectura básica que incluye una capa convolucional con 32 filtros, seguida de una capa de *max pooling* para reducir las dimensiones y una capa densa de 128 neuronas después de aplanar los datos. Este modelo también incorpora una capa de *dropout* para prevenir el sobreajuste, reduciendo la complejidad del

modelo al eliminar conexiones de forma aleatoria durante el entrenamiento. Esta arquitectura fue suficiente para obtener buenos resultados en la clasificación de las imágenes, con una precisión considerable en el conjunto de validación, Modelo Simple - Precisión: 0.9748125076293945.

En el **modelo avanzado**, incrementé la complejidad añadiendo una segunda capa convolucional con 64 filtros después de la primera capa de 32 filtros, seguida nuevamente por una capa de *max pooling*. Además, se añadió una capa densa más grande de 256 neuronas, manteniendo la regularización con *dropout*. Esta red más profunda permite que el modelo detecte características más complejas en las imágenes, lo que resultó en una mayor precisión en comparación con el modelo simple, Modelo Avanzado - Precisión: 0.9910874962806702.

Red neuronal simple:

En el entrenamiento, se implementó una **red neuronal simple** basada en una arquitectura densa que fue entrenada durante 10 épocas. A medida que las épocas progresaron, el modelo fue mejorando tanto en términos de pérdida como de precisión. Inicialmente, el modelo comenzó con una precisión de **43.14%** y una pérdida de **1.62** en el conjunto de entrenamiento, con una precisión de **66.20%** y una pérdida de **1.05** en el conjunto de validación.

Conforme avanzaron las épocas, la precisión mejoró significativamente, alcanzando un valor de **75.37%** en el conjunto de entrenamiento y una precisión de **81.96%** en el conjunto de validación al final de la décima época. El modelo pudo aprender y generalizar mejor a los datos con cada iteración. La pérdida también disminuyó constantemente, indicando una reducción en los errores de clasificación del modelo. En la evaluación final, el modelo obtuvo una precisión de **81.91%**, lo cual es un resultado aceptable para una red neuronal simple.

Random Forest:

El informe de clasificación mostró un buen desempeño del modelo en un conjunto de datos de 80,000 ejemplos distribuidos en 10 clases, con una precisión global del 89%. Las métricas de precisión, recall y f1-score son

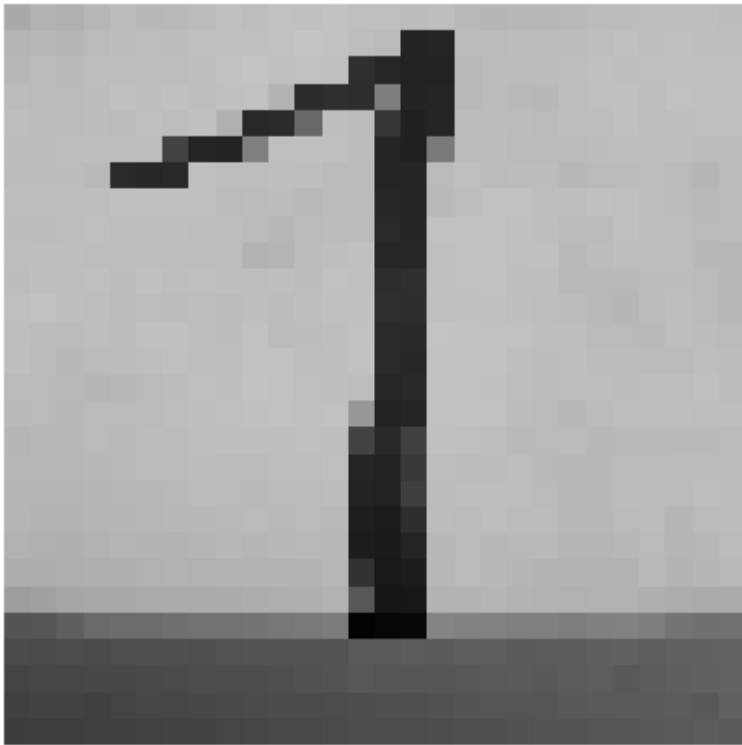
consistentemente altas en la mayoría de las clases, con valores que oscilan entre 0.83 y 0.94, el modelo es eficaz en la identificación de la mayoría de los dígitos. Aunque hay algunas variaciones en el desempeño de clases específicas, como la clase '8', que tiene un f1-score más bajo, el modelo demuestra un buen rendimiento en general.

Modelo con transformaciones

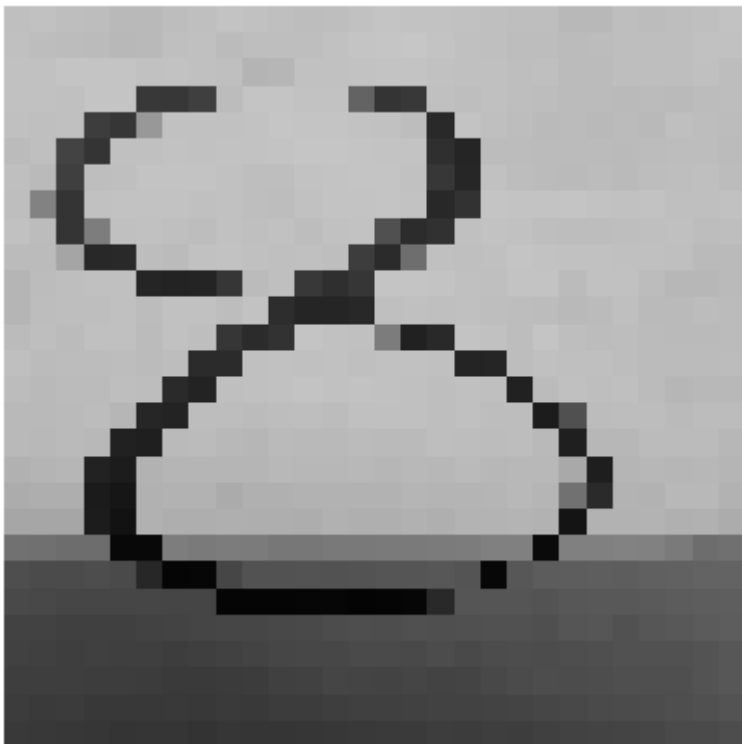
Se aplicó una técnica de *image augmentation* utilizando un generador de datos para realizar transformaciones aleatorias en las imágenes de entrenamiento, como rotaciones, desplazamientos y zoom, con el objetivo de mejorar la robustez del modelo. El modelo fue entrenado en un conjunto de imágenes divididas en 5 clases, utilizando augmentación en las imágenes de entrenamiento y sin augmentación en las imágenes de validación.

Los resultados indican que el modelo no logró aprender de manera efectiva. La precisión del modelo se mantuvo alrededor del 20% durante todo el entrenamiento, tanto en el conjunto de entrenamiento como en el de validación, el modelo no estaba diferenciando correctamente entre las clases. Las pérdidas no mejoraron significativamente, el modelo estaba "atascado" en una situación donde no lograba optimizar su rendimiento.

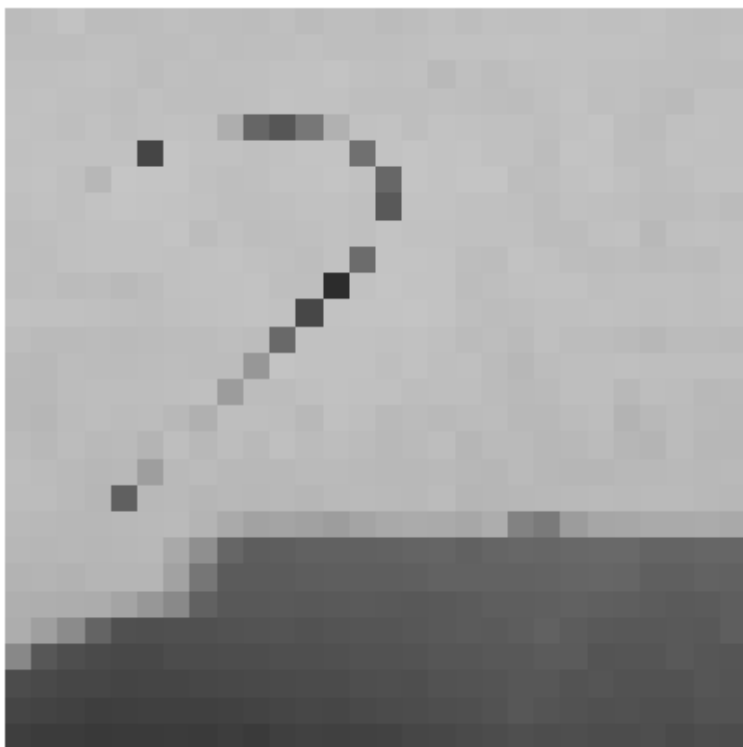
Pruebas con el mejor modelo



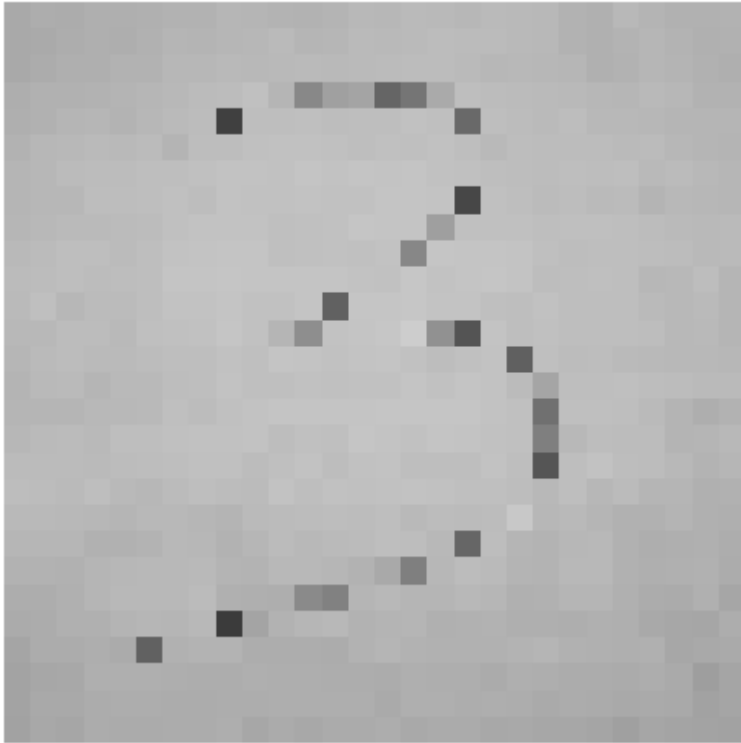
El dígito predicho es: 1



El dígito predicho es: 3



El dígito predicho es: 2



El dígito predicho es: 3

El modelo acertó en la mayoría de pruebas, demostrando que en realidad tenía una precisión casi perfecta, sin embargo, se equivocó con un 8 al decir que es un 3, el motivo de esto podría ser que el 8 y el 3 poseen una forma similar, el modelo en general clasificó de manera correcta las pruebas.