

Inteligencia Artificial

Practica 3: Redes Neuronales

Profesor: Enrique Francisco Soto Astorga
Ayudante de Laboratorio: Ricardo Enrique Pérez Villanueva

March 27, 2025

1 Proyecto

Vamos a implementar 2 proyectos en esta practica: una red neuronal que pueda identificar imágenes de MRI de cerebros. Usando estos datos, vamos a crear una clasificación entre que imágenes presentan un diagnostico de tumor, y cuales no.

1.1 Diagnostico de Tumores

Vamos a utilizar en Dataset que tiene imágenes de tumores, etiquetadas de forma en que hay una carpeta donde hay fotos de diagnósticos positivos de tumores, y una carpeta con fotos de diagnósticos negativos. Usando esta imágenes, vamos a convertirlas a tensores, asignarles una etiqueta dentro del código y crear una red neuronal para clasificar los resultados en 2; diagnostico positivo y negativo.

2 Implementación

La parte mas importante de este proyecto es como tratar los datos de entrada. Primero, hay que "aplanar" las imágenes, es decir, transformar las imágenes en una matriz de números, que representen el contraste de un pixel en la imagen. Cambien hay que normalizar los valores, para que en vez de ir de 0 a 255, vayan de 0 a 1, en punto flotante. También, el tamaño de cada imagen es diferente entre ellas, es decir, el no tienen el mismo numero de pixeles, por lo que tendrán que estandarizar de alguna forma las imágenes para poder usar el mismo numero de inputs para su red neuronal.

Después de eso hay que asignarle a cada arreglo una etiqueta, esto es relativamente sencillo ya que las imágenes estaban separadas en los casos negativos y positivos. También hay que hacer un "train-test Split", es decir, separar el Dataset en un set de datos para entrenar la red neuronal y otro para probar la red neuronal. La implementación del optimizador, función de perdida, épocas y batches se dejan al diseño del alumno. Recuerden calcular la precisión del modelo y poner un ejemplo de clasificación usando el Dataset de test.

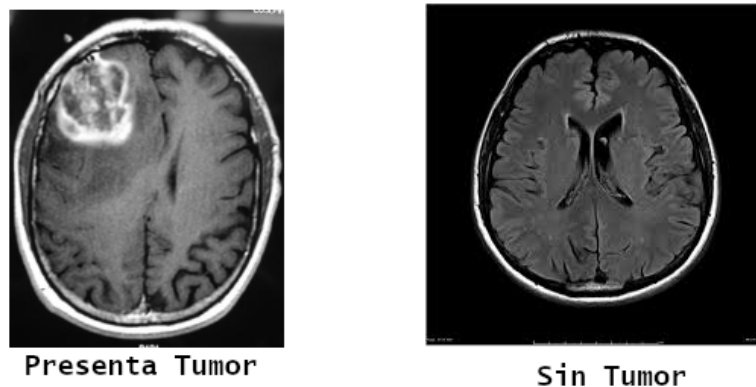


Figure 1: Ejemplo de imagenes para ambas categorias.

3 Entrega

Se va a calificar que la red neuronal funcione bien, y tenga muy pocos errores. Se va a evaluar que la función de pérdida al final del entrenamiento tenga un valor de 0.2 o menor, es decir, que su modelo sea preciso.

- Se recomienda que se haga mediante un Notebook en Google Collab
- Se recomienda que utilicen la implementación que se vio en clase, es decir, compartir con un Link en el Classroom su archivo de Google Collab donde hicieron la Red Neuronal, recuerden dar permisos de ejecución.
- Se va a calificar la precisión del modelo, como se menciona arriba, pérdida al final tenga un valor de 0.15f o menor es un buen comienzo. Esta vez, si se calificara la precisión del modelo y mientras mas preciso sean, mejor será la calificación. Recuerden que estamos tratando con diagnostico de enfermedades, por lo que la precisión debe ser nuestra prioridad en este modelo.
- Son libre de escoger el optimizador, función de pérdida y formación de las capas.

4 Documentación y materiales útiles:

A continuación se les ofrece fuentes de información que les van a ayudar a realizar su practica

- Ejemplo de una Red Neuronal para MNIST: <https://medium.com/@ombaval/building-a-simple-neural-network-from-scratch-for-mnist-digit-re>
- Visualización de una red neuronal: <https://playground.tensorflow.org/>
- Optimizadores de redes neuronales: <https://pytorch.org/docs/stable/optim.html>
- Leer Imágenes de diferentes tamaños con Pandas: <https://www.kaggle.com/code/muhammahmedansari/importing-images-of-different-size-shape-in-pandas>