

Uso de Redes Neuronales Convolucionales para la detección de tumores cancerígenos en el sistema cerebrovascular

Reconocimiento de patrones y aprendizaje de máquina

Diego Arturo Velázquez Trejo

Facultad de Ciencias, Universidad Nacional Autónoma de México, 22 de Mayo

Motivación:

Con el desarrollo de nuevas enfermedades y padecimientos, la detección y diagnóstico de enfermedades cancerígenas se ha vuelto uno de los problemas más difíciles y complejos de la medicina preventiva. Por otro lado, con el desarrollo de la Inteligencia Artificial y los avances en el aprendizaje de máquina, los problemas que hace 10 años parecían difíciles de resolver, ahora pueden ser resueltos por máquinas en cuestión de minutos. La medicina es uno de los campos que más se ha beneficiado de estos avances tecnológicos y matemáticos; las redes neuronales son empleadas para identificar tumores cancerígenos, para diagnosticar virus, enfermedades y cualquier tipo de padecimiento, para proporcionar tratamientos más efectivos, etcétera.

En el campo de la inteligencia artificial, existen varios métodos que son utilizados en las ciencias de la salud para los problemas antes mencionados: modelación con agentes, métodos heurísticos, redes bayesianas, árboles de decisión, regresiones, etcétera. Sin embargo, las redes neuronales han mostrado una mejor adaptación a este tipo de problemas, así como un mejor desempeño al momento de categorizar información.

En el presente trabajo, se desarrolló una red neuronal convolucional que es capaz de identificar tumores cancerígenos en tomografías cerebrales con un 97% de precisión. La base de datos de tomografías se extrajo de la página *kaggle*¹. Para la función de error se utilizó una función de entropía cruzada y se utilizó el algoritmo ADAM en backpropagation. Sobre las especificaciones técnicas hablaremos más adelante.

1) Espacio de Hipótesis

El espacio de hipótesis de nuestro problema corresponde a todas las imágenes que se pueden asociar con tomografías superiores del cerebro. Entre las categorías que se buscan identificar destacan: categoría de cerebro sano y categoría de cerebro con tumor. Por lo que, se distinguen las siguientes hipótesis al momento de predecir sobre una tomografía en particular: H_0 : La tomografía denota rasgos de un tumor cancerígeno v.s. H_A : La tomografía denota rasgos de un cerebro sano. Estas son las dos hipótesis sobre la cual se realizará la inferencia, y denotan el espacio de hipótesis del problema.

¹ <https://www.kaggle.com/code/purvisharma/brain-tumor-classification-98-4-accuracy/data?select=Brain+Tumor+Data+Set>

2) ¿Qué técnica se utilizó? ¿Cómo funciona?

Descripción de la técnica y del sistema:

El sistema que se implementó utiliza como núcleo a una red neuronal convolucional. Para precisar sobre su funcionamiento, es necesario describir los siguientes elementos:

a) **Convolutional Layer:**

Una capa de convolución o *convolutional layer* por el idioma inglés, es un bloque de neuronas que efectúa cierto tipo de operaciones. Es el núcleo de las redes convolucionales y su arquitectura es la que permite ir procesando la información a través de las neuronas. La arquitectura de estas capas permite a la red especializarse en identificar grandes o muy precisas características.

b) **Filtros:**

Los pesos sinápticos de las neuronas pueden representar una pequeña imagen que representa un conjunto de patrones. Los filtros o kernels, representan un bloque de números que efectúa ciertas operaciones sobre la matriz de la imagen para extraer ciertos rasgos.



En las imágenes anteriores se puede observar un tipo de filtro que se le aplica a la imagen. Los filtros no se tienen que definir de manera manual; en lugar, durante el entrenamiento de la capa convolucional, la red aprenderá aquellos filtros más útiles para la tarea dada.

La manera en la que se efectúan las operaciones matemáticas para las redes neuronales convolucionales no difieren de aquellas que se realizan para una red neuronal no convolucional, a continuación presentamos la ecuación de la salida de una red neuronal convolucional:

Equation 14-1. Computing the output of a neuron in a convolutional layer

$$z_{i,j,k} = b_k + \sum_{u=0}^{f_h-1} \sum_{v=0}^{f_w-1} \sum_{k'=0}^{f_{n'}-1} x_{i',j',k'} \cdot w_{u,v,k',k} \quad \text{with} \quad \begin{cases} i' = i \times s_h + u \\ j' = j \times s_w + v \end{cases}$$

- $z_{i,j,k}$ is the output of the neuron located in row i , column j in feature map k of the convolutional layer (layer l).
- As explained earlier, s_h and s_w are the vertical and horizontal strides, f_h and f_w are the height and width of the receptive field, and $f_{n'}$ is the number of feature maps in the previous layer (layer $l - 1$).
- $x_{i',j',k'}$ is the output of the neuron located in layer $l - 1$, row i' , column j' , feature map k' (or channel k' if the previous layer is the input layer).
- b_k is the bias term for feature map k (in layer l). You can think of it as a knob that tweaks the overall brightness of the feature map k .
- $w_{u,v,k',k}$ is the connection weight between any neuron in feature map k of the layer l and its input located at row u , column v (relative to the neuron's receptive field), and feature map k' .

La imagen muestra la ecuación que describe la propagación de una red neuronal convolucional.²

Los tensores representan la información contenida en la imagen y la arquitectura de la red debe de tener las siguientes capas:

- a) Flatten: Es aquella que se encarga de convertir una matriz de dimensiones $n \times m$ en un vector de dimensión $((n \times m) - k) \times 1$ vector.
- b) Capas intermedias: realizan distintas operaciones sobre los pesos sinápticos y son aquellas que se encargan de identificar rasgos precisos en los conjuntos de entrenamiento.
- c) Softmax (es una de las últimas capas de la red): se encarga de convertir los resultados a probabilidades, para que se pueda determinar la clase con mayor probabilidad, que corresponde a la clase seleccionada por la red neuronal.

Para el entrenamiento de la red neuronal se necesitan considerar dos elementos: un optimizador y una función de error. El optimizador es el método mediante el cual la red aprenderá (modificando los pesos sinápticos de las conexiones entre neuronas), para efectos de este proyecto, se utilizó como optimizador al algoritmo ADAM. La función de entropía cruzada se utiliza para obtener el error de la red para que después se proceda a minimizar, se utilizó la entropía cruzada categórica.

² Aurélien Géron, Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow, O'REILLY, 2da edición. (2019).

3) Cómo se conforma el conjunto de datos con los que se dispone

El conjunto de datos que se obtuvo es un repositorio de 4,600 imágenes en donde 2513 imágenes corresponden a tomografías cancerígenas, mientras que 2087 imágenes son tomografías de cerebros sanos. Se obtuvieron del repositorio [Kaggle](https://www.kaggle.com/purvisharma/brain-tumor-classification-98-4-accuracy/data?select=Brain+Tumor+Data+Set)³. Las únicas categorías que predispone la base de datos que se encontró son: cerebro sano y cerebro con tumor cancerígeno. Al descargarse del repositorio, las imágenes están contenidas en dos carpetas: *Brain Tumor* y *Healthy*. Para utilizar las imágenes en el sistema, se leyeron con ImageDataGenerator de *keras* y se almacenaron en los conjuntos de entrenamiento, prueba y validación. Estas imágenes fueron cambiadas de tamaño a 244 píxeles x 244 píxeles, cada una.

4) ¿Cómo se modeló la solución? Descripción de las imágenes que se busca identificar

Las tomografías que se extrajeron se consideran en formato RGB para el análisis, sin embargo, parecen venir en formato de escala de grises. Para modelar el problema, lo que se realizó fue un procesamiento de imágenes vago en donde se generalizó el tamaño de todas las imágenes a 244 píxeles x 244 píxeles, cada una. Las imágenes con tumores tienen una escala de grises más acentuados y no muestran una forma elíptica en el contorno del cerebro, a diferencia de las tomografías de cerebros sanos. Los cerebros con tumores tienen una forma más ovalada y continua sin deformaciones en el contorno, que aquellos cerebros sin tumor.

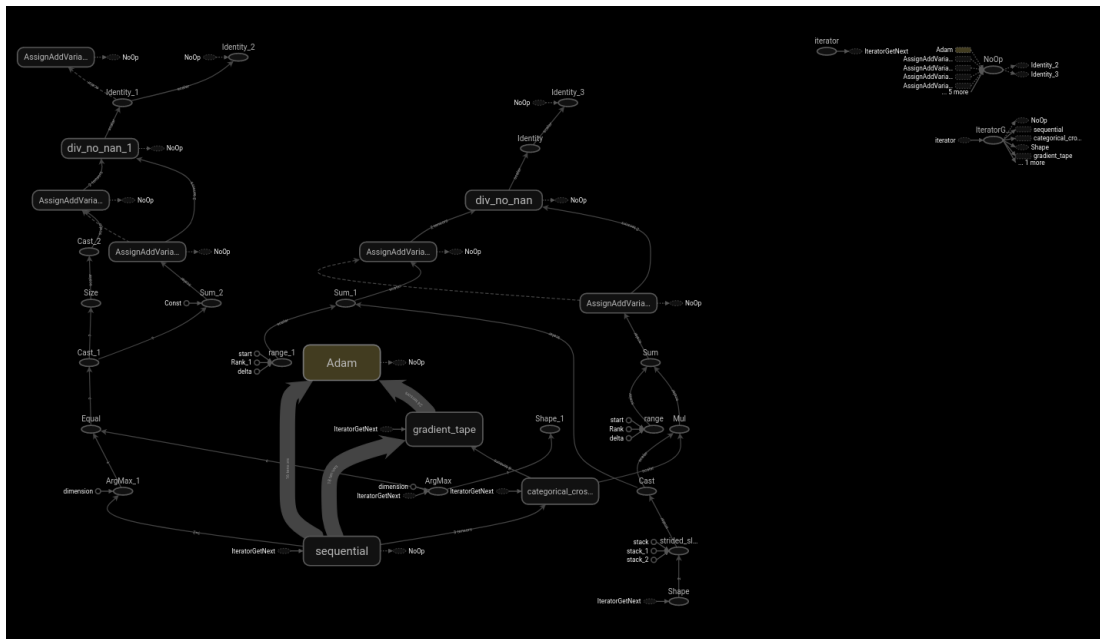
Para el procesamiento de imágenes, por otro lado, para asignarle la categoría a cada imagen, se utilizaron funciones en python para relacionar el nombre del directorio en donde cada imagen estaba almacenada con su respectiva categoría.

5) Resultados de la aplicación de la técnica al problema seleccionado

Notamos que cuando se agregaron dos capas ocultas, la precisión de la red neuronal mejoró en unas cuantas unidades. Por otro lado, para mostrar resultados, se utilizó Tensor Board que instanció una página html con la visualización de los parámetros de la red neuronal.

Los parámetros que se utilizaron fueron: 10 épocas, 5 capas ocultas, una capa conv2D, y maxPooling2D.

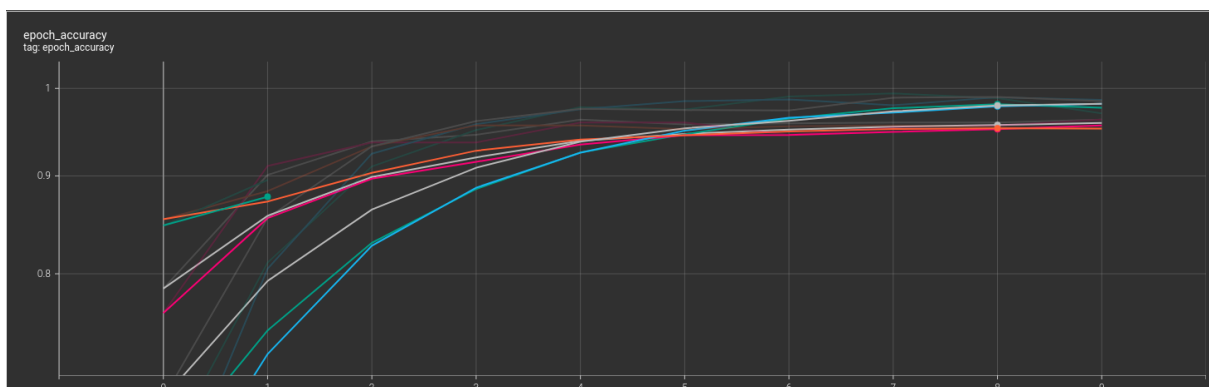
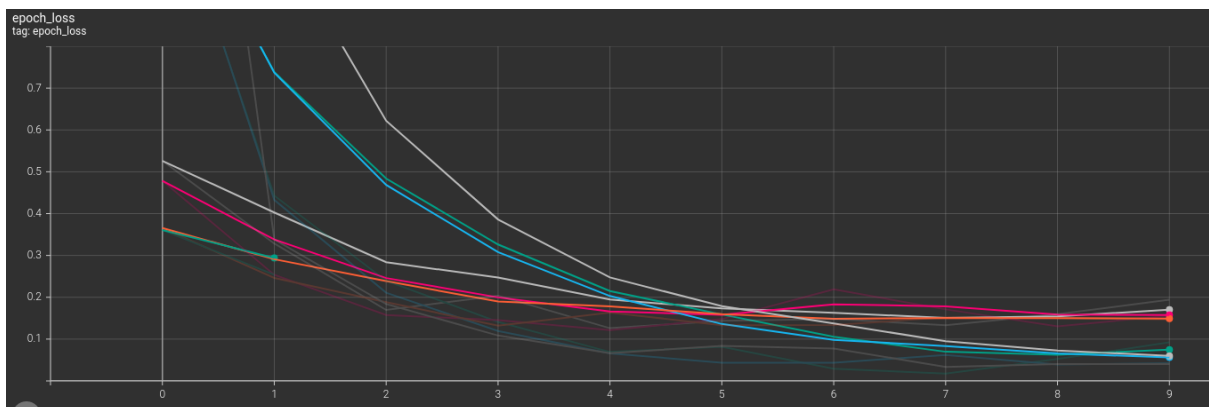
³ <https://www.kaggle.com/purvisharma/brain-tumor-classification-98-4-accuracy/data?select=Brain+Tumor+Data+Set>

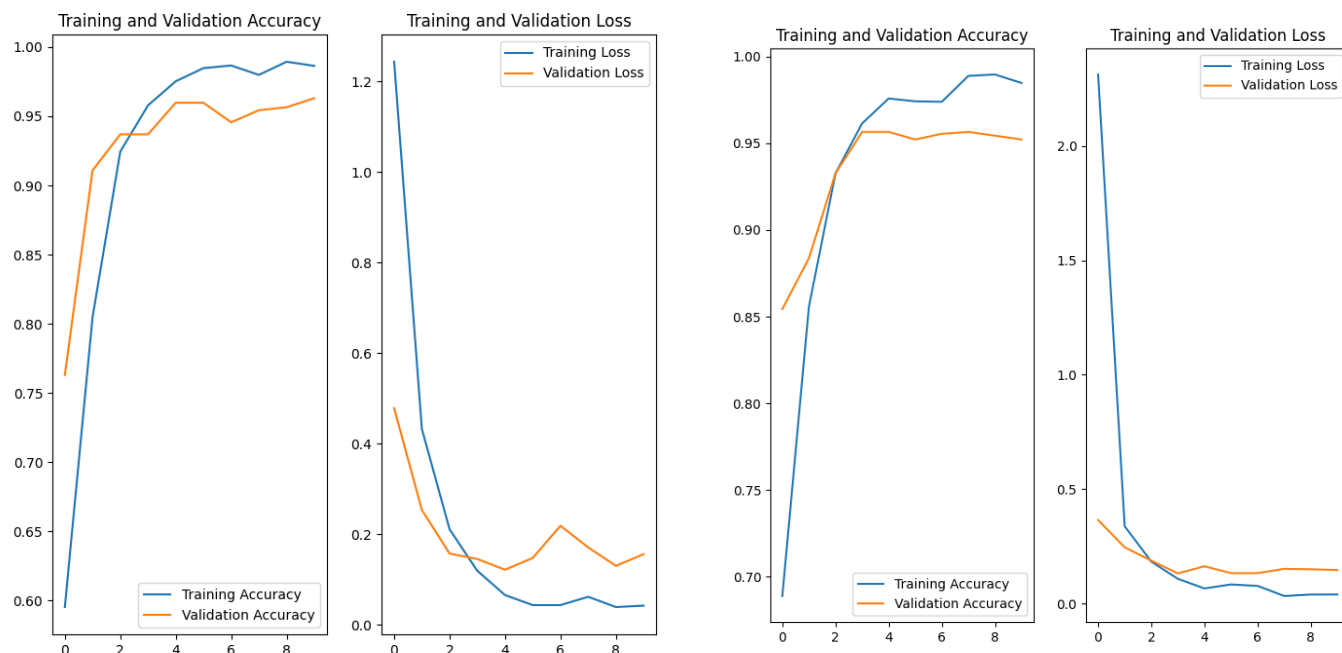


Arquitectura de la red convolucional que se diseñó

Resultados:

Observamos que, no se necesitaron más de 3 épocas para que el error fuera menor a las .3 décimas. De igual manera, cuando se estaba probando la red con más capas, a pesar de que el entrenamiento tardó más tiempo, el error disminuyó de mejor manera. En las gráficas a continuación se muestra el error y la precisión, respectivamente, a lo largo de las épocas del entrenamiento de la red neuronal:





En las gráficas antes mencionadas, podemos observar dos ejecuciones diferentes. La primera se realizó con dos capas menos que la segunda. Podemos ver cómo es que en la segunda época, la precisión en ambas arquitecturas, estaba convergiendo ya a un número mayor a las 0.9 unidades. Por otro lado, para verificar que la red neuronal no se estuviera sobreentrenando, bastó con observar que la curva azul (conjunto de entrenamiento), se encuentra muy próxima a la curva naranja que denota el conjunto de validación.

Para visualizar los resultados, en la ejecución del programa, se abrirá una instancia en <http://localhost:6006> para ver a más detalle los resultados presentados.

6) Ventajas y Desventajas de la propuesta

Las redes neuronales convolucionales resultan ser una herramienta bastante efectiva al momento de atacar problemas de clasificación de imágenes y de detección de distintas características. Sin embargo, para que un modelo sea muy preciso, se necesita una gran cantidad de fotografías/información del fenómeno en cuestión. Obtener las imágenes suele llegar a ser caro y tardado, por lo que esta es una de sus desventajas. Por otro lado, de acuerdo a la parametrización del modelo, se puede tener un mejor o peor resultado, por lo que también se necesitan ir estudiando los parámetros del modelo para encontrar los más efectivos. Las ventajas radican en que estas redes convolucionales tienen la capacidad de

generalizar a los fenómenos, de este modo, cuando se estén utilizando sobre nueva información al momento de predecir, se podrán realizar predicciones certeras. Por último, el aprendizaje heredado es una gran ventaja; ya que se cuentan con los parámetros de los pesos para identificar tumores cancerígenos, si se deseara identificar nuevos rasgos sobre la imagen con inflamación, dilatación, etcétera, se pueden reutilizar los pesos ya encontrados en el modelo previo y así, durante el entrenamiento de la nueva red neuronal, esta se centrará únicamente en la nueva categoría que se acaba de agregar, por lo que es muy fácil heredar conocimiento y ampliar el espacio de hipótesis y de trabajo.

Bibliografías:

- a) Aurélien Géron, Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow, O'REILLY, 2da edición. (2019).
- b) <https://www.tensorflow.org/tutorials/images/cnn?hl=es-419>
- c) <https://www.tensorflow.org/tutorials/images/classification?hl=es-419>
- d) <https://docs.microsoft.com/es-es/learn/modules/intro-computer-vision-tensorflow/5-convolutional-networks>