

Creación de una CNN para detección de casos de neumonía

El objetivo de esta práctica será programar una red neuronal convolucional (CNN) en Python que trabaje sobre radiografías de tórax y las clasifique como neumonía o normal. Por tanto, estaremos hablando de un clasificador **binario**.

Vamos a partir de un dataset público, disponible en la plataforma [Kaggle](#) (**atención**: no es necesario que os lo descarguéis). Las imágenes estarán ya divididas en tres datasets: train, validation y test, y a su vez, cada una estará dividida en “NORMAL” (exámenes normales) y “OPACITY” (exámenes con neumonía). Como vamos a entrenar una red neuronal a partir de datos etiquetados, estamos hablando de **aprendizaje supervisado**.

Trabajaremos con [Google Colaboratory](#) (Colab), una herramienta que ofrece Google para poder utilizar los recursos de máquinas remotas más potentes. Podéis acceder al notebook a través del siguiente enlace:

Notebook de Colab:

<https://colab.research.google.com/drive/1RE2HO-WAy79z9bamS4WWc1vacu5WlIKD?usp=sharing>

Veréis que no tendréis permisos para editar el notebook. Lo recomendable es que accedáis al notebook, os lo descarguéis en vuestra carpeta de Google Drive, y trabajéis con una copia. El único requisito para acceder a Colab es tener una cuenta de google drive. Todo el cuaderno está explicado a lo largo del vídeo, por lo que lo ideal es que vayáis viendo el vídeo a medida que vayáis trabajando con vuestra copia.

Se pide que se analice el código indicado anteriormente y se realice un pequeño informe (máximo 2 páginas) dónde se responda a las siguientes preguntas

1. Describa brevemente cada uno de los pasos que se han seguido hasta la evaluación del modelo.
2. ¿Qué tipo de algoritmo de IA estamos programando?
3. En este caso ya tenemos los datos separados en datasets de entrenamiento, validación y test. ¿Qué otra herramienta podríamos usar si este no fuera el caso?
4. Describa la arquitectura de la red que se ha programado.

5. ¿Qué función de activación hemos usado en cada una de las capas convolucionales?
¿Por qué se ha usado?
6. ¿Cuál es la función de las capas densas al final de la red?
7. ¿Qué nos indica la matriz de confusión con respecto a la sensibilidad y especificidad?
¿Y la curva ROC?
8. Ejecute el modelo con alguna variación (por ejemplo, añadiendo algún parámetro de regularización, cambiando la función de activación de alguna de las capas, modificando alguna capa, modificando algún hiperparámetro, etc.). Explique qué variación/es ha introducido y cómo cambia el resultado del modelo, adjuntando además una imagen de la matriz de confusión y la curva ROC.
9. ¿Se observa *overfitting/underfitting* con el nuevo modelo?