

Clasificación de imágenes de cáncer de cerebro mediante aprendizaje profundo

Lorena Pelegero Alonso

Bioinformática y Bioestadística

Àrea 4

Romina Astrid Rebrij

Antoni Pérez Navarro

24/12/2021

<Lorena Pelegero Alonso>



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

C) Copyright

© (Lorena Pelegero Alonso)

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Clasificación de imágenes de cáncer de cerebro mediante aprendizaje profundo</i>
Nombre del autor:	<i>Lorena Pelegero Alonso</i>
Nombre del consultor/a:	<i>Romina Astrid Rebrij</i>
Nombre del PRA:	<i>Antoni Pérez Navarro</i>
Fecha de entrega (mm/aaaa):	12/2021
Titulación:	<i>Máster Bioinformática y Bioestadística</i>
Área del Trabajo Final:	<i>Área 4, Subárea 2: Inteligencia artificial y desarrollo web</i>
Idioma del trabajo:	Español
Número de créditos:	15
Palabras clave	<i>Aprendizaje profundo; Red neuronal artificial; cáncer</i>
Resumen del Trabajo:	
<p>En este trabajo se realiza un estudio sobre el cáncer de cerebro con la finalidad de poder identificar el tipo de tumor a través de una red neuronal convolucional. Las aplicaciones de este trabajo son claves para la comunidad sanitaria debido a que la predicción del tipo de tumor es esencial para determinar las opciones de tratamiento. Para poder realizar las predicciones del tumor se han utilizado 3064 imágenes de 233 pacientes con 3 tipos de tumores diferentes: meningioma, gliomas y tumor pituitario. Para construir el modelo de la red neuronal se ha realizado previamente la exploración de los datos, seguido del preprocesamiento que ha consistido en normalizar los datos y bajar el tiempo de procesamiento de las imágenes disminuyendo la resolución, seguidamente se ha utilizado la librería "Keras" de Python para la construcción del modelo y por último, se ha realizado el análisis de resultados. Con la construcción de la red neuronal convolucional se ha obtenido una exactitud del 93% sobre la predicción del tipo de tumor. Estos resultados los consideramos óptimos debido a que se ha</p>	

obtenido una precisión elevada con un número muy limitado de imágenes de IMR. Podemos concluir que el modelo obtenido puede dar soporte a los profesionales sanitarios para la predicción de estos 3 tipos de tumores con una mayor seguridad.

Abstract:

In this work a study on brain cancer is carried out in order to be able to identify what type of tumor it is through a convolutional neural network. The applications of this work are key for the health community because the tumor type prediction is essential to determine different treatments. To make tumor predictions, 3064 images of 233 patients with 3 different types of tumors were used: meningioma, gliomas and pituitary tumor. To build the neural network model, the data exploration has been previously carried out, followed by the preprocessing that has consisted of normalizing the data and reducing the processing time of the images by decreasing the resolution, then the construction of the model was done with Python library "Keras" and finally, the result analysis was carried out. With the convolutional neural network, the accuracy on the prediction of the type of tumor obtained was up to 93%. We consider these results optimal because high precision has been obtained with a very limited number of IMR images. We can conclude that the model obtained can support healthcare professionals for the prediction of these 3 types of tumors with greater safety.

Índice

1	Resumen	2
2	Introducción	2
2.1	Contexto y justificación del Trabajo	3
2.2	Objetivos del Trabajo	3
2.3	Enfoque y método seguido	4
2.4	Planificación del Trabajo	7
2.5	Breve resumen de contribuciones y productos obtenidos	10
2.6	Breve descripción de los otros capítulos de la memoria	11
3	Estado del arte	11
3.1.	Cáncer de cerebro.	12
3.1.1.	Diagnóstico	12
3.1.2.	Clasificación	12
3.2.	Cáncer de cerebro y clasificación automática.	15
3.3	Aprendizaje automático.	16
3.4	Aprendizaje profundo.	19
4	Metodología	25
5	Resultados	29
6	Discusión	35
7	Valoración económica	35
8	Conclusiones	36
8.1	Líneas de futuro	37
8.2	Seguimiento de la planificación	37
9	Glosario	38
10	Bibliografía	39
	Anexos	41

Lista de figuras

Ilustración 1: esquema de la metodología de las ANN.	5
Ilustración 2: representación gráfica del diagrama de Gantt sobre la planificación del trabajo ¹³	9
Ilustración 3: imagen en IMR de un tumor tipo Glioma	13
Ilustración 4: imagen en IMR de un tumor tipo Meniingioma.	14
Ilustración 5: imagen en IMR de un tumor tipo pituitario.	15
Ilustración 6: esquema sobre el aprendizaje automática	16
Ilustración 7: esquema sobre las redes neuronales.....	20
Ilustración 8: representación grafica de la función de activación Sigmoide ..	23
Ilustración 9: representación gráfica de la función de activación tangente hiperbólica	23
Ilustración 10: representación gráfica de la función de activación Softmax. .	23
Ilustración 11: representación gráfica de la función de activación ReLU	24
Ilustración 12: representación gráfica del funcionamiento de gradiente descendente.....	25
Ilustración 13: representación esquemática del funcionamiento de las redes neuronales convolucionales.	25
Ilustración 14: esquema de la estructura de las carpetas para la creación del modelo.	26
Ilustración 15: esquema del modelo que se ha implementado	27
Ilustración 16: esquema sobre el funcionamiento del Max-Pooling.	28
Ilustración 17: representación gráfica de la precisión del primer modelo:.....	30
Ilustración 18: representación gráfica de la precisión del modelo mejorado. .	31
Ilustración 19: representación gráfica de la sensibilidad y especificidad vs los ciclos del modelo mejorado	32
Ilustración 20: matriz de Confusión con los resultados del modelo	33
Ilustración 21: representación gráfica de los TP, FN, FP y TN	34

Lista de tablas

Tabla 1: planificación del proyecto con la indicación de la fecha inicial y la final.	8
Tabla 2: descripción, severidad y probabilidad del análisis de riesgo.....	10
Tabla 3: resultados de la precisión del modelo inicial y mejorado	29
Tabla 4: resultados de la exactitud, precisión, sensibilidad y especificidad para cada clase.	34
Tabla 5: coste para la realización del proyecto.....	35

1 Resumen

Hoy en día el estudio de aprendizaje profundo es primordial en el ámbito de salud ya que gracias a la creación de modelos de inteligencia artificial se puede ayudar al personal sanitario a detectar ciertas enfermedades con una gran precisión y poder aplicar el tratamiento correcto. Existen numerosos estudios sobre el cáncer de cerebro y su detección, en este trabajo en concreto se ha construido un modelo de red neuronal convolucional con la capacidad de clasificar entre 3 tipos diferentes de tumores cerebrales: meningioma, gliomas y tumor pituitario. Los resultados obtenidos con el modelo son de una exactitud del 93%, este resultado aunque sea un valor elevado y haya cumplido con el objetivo de este estudio, puede no resultar totalmente convincente debido a que la finalidad de este estudio es aplicaciones médicas donde un valor más bajo del 99% puede poner en riesgo vidas de personas. Por tanto, estos resultados pueden servir para orientar y dar soporte al personal sanitario pero no para decidir por ellos.

2 Introducción

Un tumor cerebral es una masa o un crecimiento de células anormales en el cerebro que pueden ser tanto benignos, que no contienen células cancerosas, como malignos. Algunos tumores son primarios, es decir, se originan en el momento en el que sus células tienen mutaciones en su ADN, por el contrario, existen tumores metastásicos que originariamente aparecieron en una parte del cuerpo y que, por metástasis, han llegado al cerebro. Existen muchos tipos de tumores cerebrales pero en este estudio nos centraremos en 3 tipos: meningioma (surge de las meninges), glioma(se desarrolla en el cerebro y la médula espinal) y el tumor pituitario (se desarrollan en la glándula pituitaria).¹

No hay forma de prevenir el cáncer de cerebro, pero es primordial conseguir un correcto diagnóstico para su tratamiento. Actualmente, una de las maneras de los profesionales de la salud para reconocer el tipo de tumor es a través de imágenes por resonancia magnética (IRM).² Se trata de una técnica no invasiva que permite la visualización de estructuras internas del cuerpo, pero en muchos casos, es necesaria otras prácticas como la biopsia para confirmar el tipo de tumor, debido a que gran parte de los datos disponibles a través de las imágenes de resonancia magnética no se pueden detectar a simple vista, como es el caso de los detalles relacionados con la forma, la textura o la intensidad del tumor.

Tanto el índice de crecimiento como la ubicación del tumor cerebral determinan cómo afectará al sistema nervioso, por tanto, la predicción del tipo de tumor es esencial para determinar las opciones de tratamiento, pronóstico y prevenir la tasa de mortalidad humana, por eso, es importante el desarrollo de algoritmos de Inteligencia Artificial que ayudan a la detección y clasificación de los tumores.

2.1 Contexto y justificación del Trabajo

El aprendizaje automático es una rama de la inteligencia artificial que emplea técnicas estadísticas y de optimización que permiten generar modelos de predicción a partir de una base de datos.³ Estos modelos predictivos son cada vez más utilizados en aplicaciones médicas,⁴ como en el diagnóstico y detección del cáncer, esto es debido a que el método convencional para la clasificación es mediante inspección humana que puede llevar a controversia entre los diferentes profesionales debido a la subjetiva interpretación, en cambio, al utilizar sistemas de clasificación automática esta subjetividad se elimina y dan soporte a la toma de decisiones.

En este estudio, se emplea una red neuronal convolucional con técnicas de procesamiento de imágenes para implementar una clasificación de 3 tipos de tumores cerebrales. A día de hoy, hay muchos métodos de aprendizaje automático para la clasificación del cáncer, por eso, es necesario conocer cuál de estos modelos tiene un mayor rendimiento predictivo. Existen varios estudios desarrollados por otros autores para la clasificación de cáncer de clases múltiples donde se analiza el rendimiento predictivo entre varios métodos existentes⁵, como el estudio de Konstantina Kourou "*Machine learning applications in cancer prognosis and prediction*"⁴ donde realizan una revisión de los enfoques recientes de aprendizaje automático (Machine Learning) empleados en el modelado de la progresión del cáncer, además, presentan las publicaciones más recientes que emplean estas técnicas como objetivo para predecir el riesgo de cáncer.

Otro reto al que se enfrenta este estudio es sobre la interpretabilidad de los resultados, debido a que el aprendizaje automático carece de la capacidad de interpretar su mecanismo de trabajo interno y explicar el razonamiento principal de sus predicciones. En este estudio proporcionamos razonamientos sobre las predicciones las cuales deben ser precisas e interpretables a partir de la investigación teórica previa del tema de estudio, y validaremos la calidad del modelo obtenido. Para ello, se realizará un estudio previo sobre el contexto del estudio y se realizará posteriormente una correcta interpretación.

2.2 Objetivos del Trabajo

El objetivo de este trabajo es determinar el tipo de tumor cerebral presente en diferentes imágenes a través de un modelo de red neuronal convolucional. Los datos utilizados para entrenar el modelo consisten en un conjunto de imágenes de resonancia magnética que muestran 3 tipos de tumores cerebrales: meningioma, glioma o tumor pituitario⁶. Los objetivos específicos son:

<Lorena Pelegero Alonso>

Objetivos generales:

- Determinar el tipo de tumor cerebral presente en diferentes imágenes a través de un modelo de red neuronal convolucional.
- Desarrollar una interfaz web para implementar el modelo donde se visualicen los resultados de las predicciones.

Objetivos específicos:

- Estudiar las diferencias fisiológicas entre los 3 tipos de tumores.
- Utilizar una Red Neuronal Convencional para la predicción del tumor.
- Conseguir una predicción del tipo de tumor con una precisión de mínimo 80%.
- Investigar la posible optimización del modelo escogido.
- Investigar la relevancia del estudio y las posibles aplicaciones en la comunidad.

2.3 Enfoque y método seguido

El estudio realizado contiene 3064 imágenes de 3 tipos de tumores cerebrales de 233 pacientes, 708 imágenes contienen meningioma, 1426 gliomas y 930 contiene el tumor pituitario. Los datos se han obtenido del estudio " Enhanced Performance of Brain Tumor Classification via Tumor Region Augmentation and Partition" ⁷ y "Retrieval of Brain Tumors by Adaptive Spatial Pooling and Fisher Vector Representation."⁸ de Cheng, Jun, a diferencia de este estudio que utilizaremos una red neuronal convolucional para la predicción del tumores, en los estudios mencionados utilizan pirámides espaciales (SPM) y tienen como objetivo recuperar tumores cerebrales en imágenes de resonancia magnética mejoradas con contraste ponderado.

Las metodologías para desarrollar un modelo de clasificación de imágenes siguen la siguiente esquemática⁹:

- Paso 1 - Recoger los datos: Importar los datos de las imágenes. Es importante utilizar un elevado número de imágenes incluyendo la mayor variabilidad posible.
- Paso 2 - Exploración y preparación de los datos: revisar la estructura de los datos para comprobar que los tipos de datos son los adecuados, es decir, que contienen las imágenes con una resolución adecuada y con suficiente variedad entre las clases. Luego se procede a la transformación de los datos en caso de ser necesario.
- Paso 3 - Entrenar el modelo con los datos: asignación de individuos de entrenamiento y validación. Entrenar una red neuronal consiste en ajustar cada uno de los pesos de las entradas de todas las neuronas que forman

parte de la red neuronal, para que las respuestas de la capa de salida se ajusten lo más posible a los datos que conocemos.

- Paso 4 - Evaluación del rendimiento del algoritmo a través de una matriz de confusión que nos indica los parámetros de exactitud, precisión, especificidad y sensibilidad. Se prueba el modelo con los datos de validación, y se obtiene la evaluación correspondiente.
- Paso 5 - Mejorar el modelo: mejorar la precisión de la validación y la curva de aprendizaje. Para ello se puede utilizar la técnica de ampliación del volumen de datos (data augmentation) que permite generar más imágenes a partir de algunas distorsiones de las imágenes que se tiene. Otra opción sería utilizar los pesos de una CNN pre-entrenada, a esto se le llama transferencia de conocimiento (Transfer Learning) y ajustar los pesos (parámetros de la red) a los datos propios.



Ilustración 1: esquema de la metodología de las ANN.

Para construir la red neuronal convolucional se pueden utilizar varios lenguajes de programación, los más utilizados son Python y R, aunque Python es el que tiene más peso debido a que las librerías estaban escritas originalmente solo con Python y posee una gran biblioteca para modelos de Machine learning de redes neuronales artificiales que proporciona un alto nivel frente a varios puntos de aprendizaje profundo. Hoy en día, se puede utilizar los dos lenguajes con la misma eficacia ya que todas las librerías se han traducido a R. En este estudio se utilizará Python debido a que tiene un tiempo inferior de procesamiento debido a que R usa Python cuando ejecuta funciones Keras y provoca que tarde más en procesar, además, Python administra mejor la memoria.

Las bibliotecas más utilizadas para realizar aprendizaje profundo son TensorFlow, Theano, deeplearning4j y Caffe. A continuación se exponen las ventajas y desventajas de cada biblioteca y las razones de nuestra elección para este estudio.

<Lorena Pelegero Alonso>

Deeplearning4j tiene muchas aplicaciones para redes neuronales profundas. Esta biblioteca funciona con Scala y también proporciona soporte de GPU incorporado. El mayor inconveniente es que solo admite el lenguaje de programación java y por tanto, queda descartada debido a que queremos utilizar Python.

Caffe permite configurar los hiper parámetros para una red profunda. La configuración de la capa es muy robusta y sofisticada. Su inconveniente es que utiliza un código abierto escrito en C ++ / CUDA y en este estudio se utilizará Python¹⁰.

Theano es la biblioteca de aprendizaje profundo de código abierto escrita en Python y CUDA. Theano es rápido debido a que utiliza vectores y matrices para sus funciones y provoca que se ejecute más rápido. Es similar a Tensorflow, pero la implementación y el uso son más complejos, por tanto, nos decantamos por Tensorflow¹¹.

Actualmente, **TensorFlow** es la biblioteca más utilizada debido a que se pueden desarrollar diferentes tipos de redes profundas y abordar la mayoría de las tareas y problemas en el campo del aprendizaje profundo. Tensorflow utiliza la librería, Keras que sirve para acceder a los diferentes paquetes y métodos fácilmente. En este estudio se utilizará Tensorflow debido a que es fácil de construir modelos complejos solo en unas pocas líneas de código lo cual nos permitirá desarrollar el modelo y verificarlo rápidamente. Además, está adaptada a Python y puede usar GPU para un ajuste rápido del modelo¹¹.

Para la implementación de la interfaz web del modelo se pueden utilizar lenguajes desarrollados específicamente para el desarrollo web como HTML, CSS y PHP, el inconveniente de estos lenguajes es que requieren mucha formación y necesitaríamos codificar cada detalle poco relevante para los objetivos de este trabajo. Como alternativa tenemos Django y Flask que permiten crear aplicaciones web sin tener que entrar en todos los detalles implicados. Estos programas utilizan marcos que ahorran mucho tiempo y permiten centrarse en la creación de aplicaciones web sin consumir mucho tiempo.

Django es un marco de alto nivel que se utiliza para desarrollar rápidamente aplicaciones, está diseñado para minimizar la codificación real durante el proceso de diseño de la aplicación. Django es rápido y tiene todas las funciones que puede necesitar la aplicación web pero como inconveniente tiene que no está hecho para proyectos pequeños ya que no hay forma de separar nada de ese paquete si solo necesita algunas características. En cambio Flask es un micro-marco, lo que significa que está diseñado para realizar un papel más limitado. Como micro-framework, Flask es más flexible a diferentes estilos de trabajo y enfoques para el desarrollo de aplicaciones web. Para nuestro estudio estamos buscando diseñar una aplicación web relativamente simple, por tanto, Flask es la mejor opción¹².

<Lorena Pelegero Alonso>

Una vez desarrollada la aplicación web se subirán al servidor werkzeug.

Para poder realizar el estudio se utilizará Google colab, que se trata de un entorno gratuito que no requiere configuración y que se ejecuta completamente en la nube. Permite a cualquier usuario escribir y ejecutar código de Python en el navegador, es utilizado para tareas de aprendizaje automático y análisis de datos debido a que ofrece acceso gratuito a recursos informáticos, como GPUs, de esta manera, conseguimos un rendimiento mayor al que tendríamos en nuestro PC.

2.4 Planificación del Trabajo

Descripción de tareas, fechas de inicio y fin y entregable en cada etapa.

Definición	Fecha inicio	Fecha Final	Hito
Definición de los contenidos del trabajo	15/09/21	22/09/21	Entrega de PEC 0
Plan de trabajo	23/09/21	04/10/21	Entrega de PEC 1
Fase 1 del desarrollo del trabajo	05/10/21	08/11/21	Entrega de PEC 2
Exploración de paquetes para la predicción de modelos con imágenes	28/09/21	01/10/21	
Practica en la interfaz Google Colab	28/09/21	01/10/21	
Extracción de los datos de imágenes	01/10/21	04/10/21	
Exploración y preparación de los datos	04/10/21	11/10/21	

<Lorena Pelegero Alonso>

Implementar un modelo CNN y entrenarlo	11/10/21	18/10/21	
Evaluación del rendimiento del algoritmo	18/10/21	25/10/21	
Documentar en la memoria	25/10/21	08/11/21	
Fase 2 del desarrollo del trabajo	09/11/21	09/12/21	Entrega de PEC 3
Mejorar el modelo	09/11/21	16/11/21	
Estudio sobre el desarrollo web	16/11/21	23/11/21	
Desarrollo interfaz web con el modelo mejorado.	24/11/21	09/12/21	
Comprobación de uso de la aplicación y subirlo al servidor werkzeug	09/12/21	12/12/21	
Documentar en la memoria	09/12/21	24/12/21	
Cierre de la memoria	10/12/21	24/12/21	Entrega de PEC 4
Elaboración de la presentación	27/12/21	03/01/22	Entrega de PEC 5
Estudio y práctica de la presentación	03/01/22	13/01/22	
Defensa pública	13/01/22	21/01/22	Entrega de PEC 5b

Tabla 1: planificación del proyecto con la indicación de la fecha inicial y la final.

Calendario

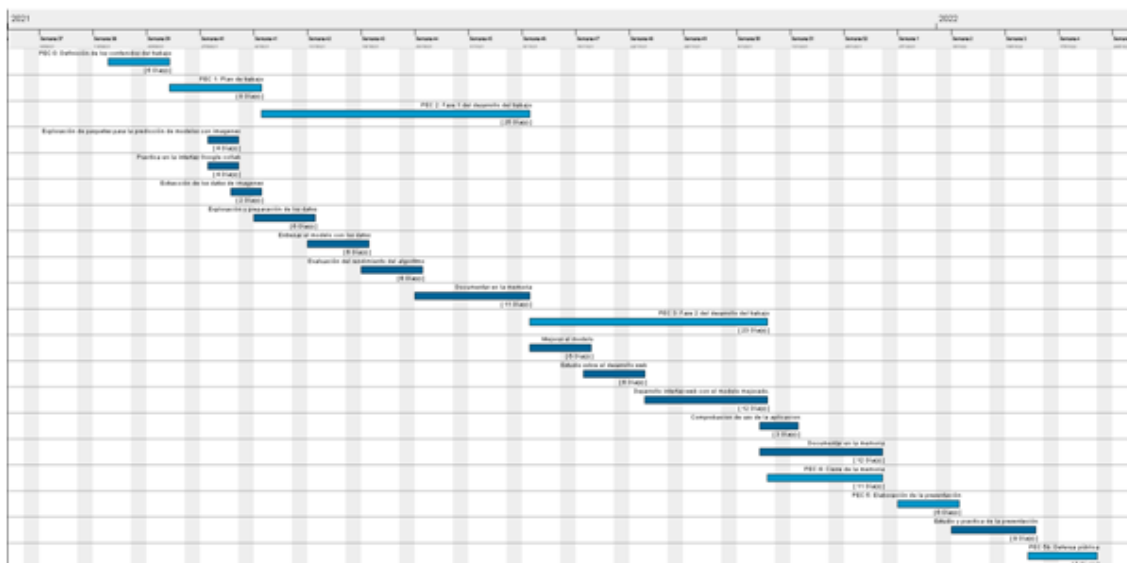


Ilustración 2: representación gráfica del diagrama de Gantt sobre la planificación del trabajo¹³

Hitos:

- Tener definidos los contenidos del trabajo para el 22/09/2021 (PEC0).
- Tener definido el Plan de trabajo para el 04/10/2021 (PEC1).
- Tener finalizada la Fase 1 de desarrollo para el 08/11/2021 (PEC2).
- Tener finalizada la Fase 2 del desarrollo para el 09/12/2021 (PEC3).
- Tener finalizada la Memoria para el 24/12/2021 (PEC4).
- Tener finalizada la Presentación para el 03/01/2022 (PEC5a)
- Realizar la defensa pública el 21/01/2022 (PEC5b).

Análisis de riesgos

A continuación se muestra una tabla con la descripción del riesgo, la severidad, la probabilidad y las mitigaciones.

Descripción del riesgo	Severidad	Probabilidad	Mitigación
No conseguir desarrollar la aplicación web	Alta	Media	Realizar varios cursos sobre Flask para conseguir mayores conocimientos
Obtener un modelo con eficacia menor del 80%	Media	Media	Investigación bibliográfica sobre la manera de optimizar los modelos
Alto tiempo de procesamiento de las imágenes	Baja	Alta	Utilizar google Collab para disminuir el tiempo computacional Emplear Kaggle como alternativa a google collab.
Dificultades a la hora de utilizar python	Media	Baja	Resolver a partir de la búsqueda de foros y páginas webs las dificultades que vaya encontrando. Emplear como alternativa R
No conseguir una conclusión clara al hecho de escoger un método respecto a los otros.	Baja	Media	Búsqueda bibliográfica sobre la comparativa de los diferentes métodos y adjuntar las referencias.

Tabla 2: descripción, severidad y probabilidad del análisis de riesgo.

2.5 Breve resumen de contribuciones y productos obtenidos

-Plan de trabajo: documento que contiene las tareas asignadas y la esquemática del calendario de trabajo con sus respectivas descripciones y duraciones, además, contiene el análisis de riesgo con las posibles situaciones adversas que pueden encontrarse a la hora de ejecutar el proyecto.

-Memoria: documento en formato pdf donde se encuentra el trabajo completo incluyendo el desarrollo de la temática escogida junto con los resultados obtenidos y conclusiones. Se incluye un enlace a Github con el script completo de python.

<Lorena Pelegero Alonso>

-Presentación virtual: realizada a través de power point o Canva

2.6 Breve descripción de los otros capítulos de la memoria

-Resumen: síntesis del trabajo donde se indica la finalidad, los objetivos, el contexto, el método utilizado y las conclusiones extraídas.

-Introducción: se explica el contexto y justificación del trabajo, se indican los objetivos, el enfoque y método seguido, se indica la planificación del trabajo y se hace un breve sumario de productos obtenidos.

-Estado del arte: contexto del trabajo y relevancia del trabajo donde se realiza un estudio bibliográfico sobre el cáncer de cerebro y sus clasificaciones.

-Metodología: explicación y justificación de la realización del modelo neuronal convolucional.

-Resultados: resultados obtenidos del modelo implementado y su interpretación.

-Discusión: se discuten los resultados en el contexto del proyecto y las posibles mejoras que se pueden implementar. Se muestra como los resultados dan respuesta a los problemas planteados.

-Valoración económica: este apartado indicará los gastos asociados al desarrollo y mantenimiento del trabajo, así como los beneficios económicos obtenidos.

-Conclusiones: se detallan las conclusiones obtenidas a partir de los resultados y la discusión. Una descripción de las conclusiones del trabajo con una reflexión crítica sobre el logro de los objetivos planteados inicialmente. Además, se indican las líneas de trabajo futuro que no se han podido explorar en este trabajo y han quedado pendientes. Y se realiza un análisis crítico del seguimiento de la planificación y metodología a lo largo del producto.

-Glosario: definición de los términos y acrónimos más relevantes utilizados dentro de la Memoria.

-Bibliografía: lista numerada de las referencias bibliográficas utilizadas dentro de la memoria.

-Anexos: listado de apartados que son demasiado extensos para incluir dentro de la memoria y tienen un carácter autocontenido

3 Estado del arte

El aprendizaje profundo ha demostrado que es una herramienta muy útil debido a su capacidad para manejar grandes cantidades de datos¹⁴. Una de las aplicaciones de la IA en la lucha contra el cáncer, es el diagnóstico de los pacientes con tumores cerebrales, donde el primer paso suele ser una cirugía donde se elimina la mayor cantidad de masa tumoral, pero, gracias a la aplicación combinada de una técnica avanzada de obtención de imágenes con

<Lorena Pelegero Alonso>

la inteligencia artificial, se puede diagnosticar con precisión el tumor cerebral y diferenciar con exactitud el tejido tumoral del sano.

3.1. Cáncer de cerebro.

Un tumor cerebral es un crecimiento de masa de células anormales que crece en el cerebro o alrededor del cerebro. Presenta un riesgo para el cerebro sano porque invade o destruye el tejido normal del cerebro comprimiendo y desplazando al cerebro. Debido a que el cerebro está contenido dentro del cráneo que tiene una cantidad limitada de espacio, un tumor que está creciendo puede causar un daño importante al cerebro llegando a destruir las células cerebrales sanas o dañar células sanas indirectamente mediante la colonización de otras partes del cerebro, causando inflamación, hinchazón del cerebro y presión dentro del cráneo.

3.1.1. Diagnóstico

La velocidad con la que crece y avanza un tumor cerebral varía en función del tipo y la persona, y su velocidad de crecimiento y ubicación determinarán cómo afecta al sistema nervioso¹⁵. Existen números tipos de tumores cerebrales:

En el caso de que se sospeche de la presencia de un tumor cerebral, el especialista puede ordenar la realización de una serie de pruebas para diagnosticar la presencia del tumor y comenzar el tratamiento inmediato. Las principales pruebas para diagnosticar el tumor son las siguientes:

- Examen neurológico: se controla la visión, la capacidad auditiva, el equilibrio o la coordinación. Si el paciente tiene dificultades en una o más áreas pueden dar pistas sobre la zona afectada.
- Pruebas o diagnóstico por imagen: normalmente se utilizan imágenes por resonancia magnética (RM) para ayudar con el diagnóstico¹⁵. A veces, se recomiendan otras pruebas como la tomografía computarizada.
- Pruebas para encontrar el cáncer en otras partes del cuerpo: en el caso de que el especialista considere que el tumor puede ser origen de otro cáncer que se ha expandido, se pueden hacer tomografías computarizadas o tomografía por emisión de positrones para buscar cáncer de pulmón.
- Recolección y análisis de una muestra: se trata de las biopsias. Se obtiene una muestra del tejido que se desea analizar y se envía a laboratorios.

3.1.2. Clasificación

Los tumores cerebrales pueden ser tanto benignos, es decir, no contiene células cancerosas, como malignos, que contienen células cancerosas de crecimiento muy rápido que invade o crece en zonas sanas del cerebro. También se dividen en dos categorías diferentes dependiendo del origen:

<Lorena Pelegero Alonso>

- Los **tumores cerebrales primarios** se originan dentro del cerebro incluyendo el parénquima del cerebro, la médula espinal o los nervios, se originan en el momento en el que sus células tienen mutaciones en su ADN. Pueden ser tanto benignos o malignos.
- Un **tumor metastásico** se forma cuando las células cancerosas situadas en cualquier parte del cuerpo con mayor frecuencia pulmones, mama y colon se han extendido a otras partes del cuerpo, y viajan hasta el cerebro.

Existen distintos tipos de tumores primarios, en este estudio nos centraremos en los 3 siguientes:

-Gliomas: se trata de tumores que empiezan en el cerebro o en la médula. Los gliomas representan el tipo más frecuente de tumor cerebral en los adultos, de modo que son responsables de aproximadamente el 42% de los tumores cerebrales en ellos.¹⁶

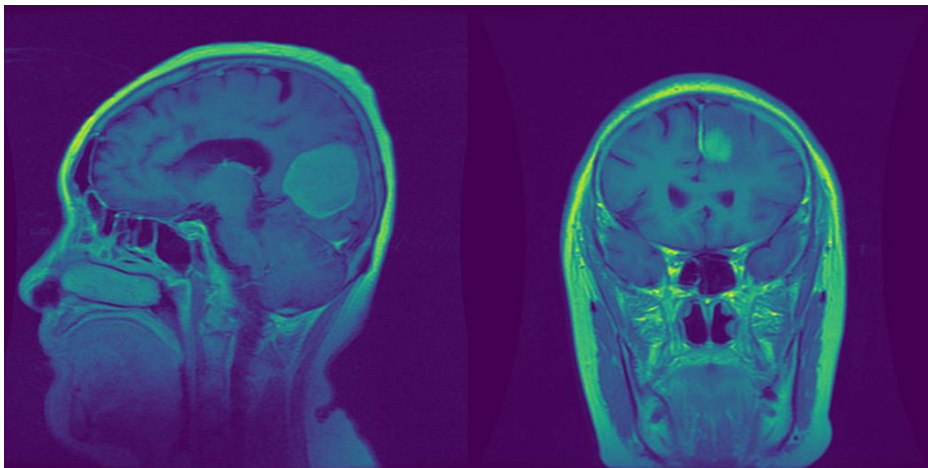


Ilustración 3: imagen en IMR de un tumor tipo Glioma

Los gliomas se subclasifican en función de los tipos celulares a los que afectan:

- **Astrocitomas:** células en forma de estrella que protegen las neuronas. Los tumores de estas células pueden extenderse desde el foco primario a otras zonas del cerebro, pero rara vez se diseminan fuera del sistema nervioso central. Los astrocitomas se gradúan del I al IV en función de la velocidad de progresión siendo el I el más lento y el IV el más rápido.
- **Oligodendroglioma:** estas células producen mielina, una sustancia grasa que forma una vaina protectora alrededor de las células nerviosas. Los oligodendrogliomas, que constituyen el 4% de los tumores cerebrales, afectan principalmente a personas mayores de 45 años.
- **Ependimoma:** estos tumores afectan a los ependimocitos, que revisten las vías que llevan el líquido cefalorraquídeo por todo el cerebro y la médula

<Lorena Pelegero Alonso>

espinal. Los ependimomas son infrecuentes y representan el 2% de todos los tumores cerebrales; no obstante, son los tumores cerebrales más habituales en los niños.

-Meningiomas: surge de las membranas que se forman en el tejido que forma la envoltura externa protectora del cerebro y la médula espinal. Una cuarta parte de todos los tumores cerebrales y medulares son meningiomas, de los que hasta el 85% son benignos. Los meningiomas pueden desarrollarse a cualquier edad, pero su incidencia aumenta significativamente en las personas mayores de 65 años. Las mujeres tienen dos veces más probabilidades que los varones de padecer meningiomas. Normalmente crecen muy lentamente y no suelen causar síntomas¹⁶.

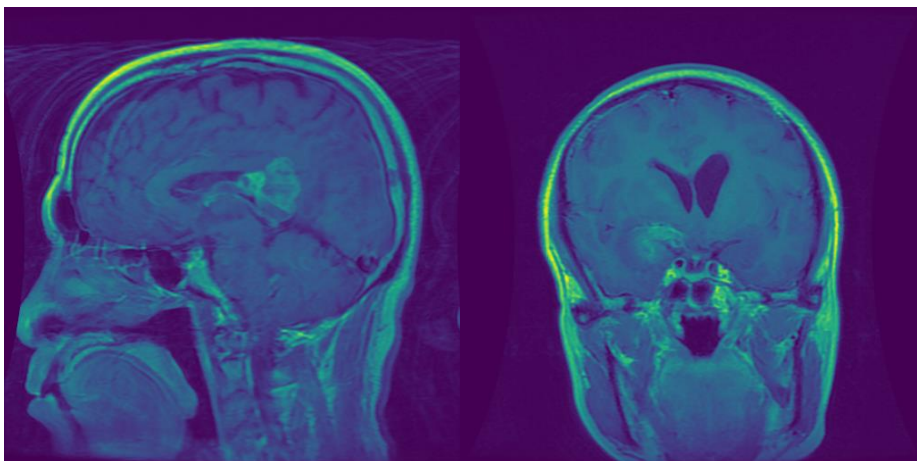


Ilustración 4: imagen en IMR de un tumor tipo Meniiongima.

-Tumor pituitario: son crecimientos anormales que se desarrollan en la glándula pituitaria que está ubicada en la base del cerebro, detrás de la nariz. Algunos tumores pituitarios generan demasiadas hormonas que regulan las funciones importantes del cuerpo o producen niveles más bajos de hormonas. La mayoría de los tumores pituitarios son crecimientos benignos, también llamados adenomas. Estos permanecen en la glándula pituitaria o en los tejidos que la rodean y no se diseminan a otras partes del cuerpo¹⁶.

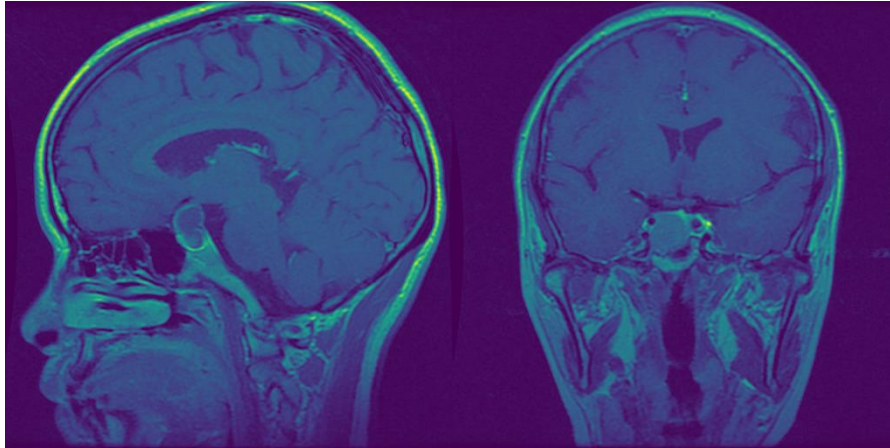


Ilustración 5: imagen en IMR de un tumor tipo pituitario.

3.2. Cáncer de cerebro y clasificación automática.

La Inteligencia Artificial es un conjunto de tecnologías que hacen que las máquinas puedan percibir, comprender, actuar y aprender de la misma forma que lo hace la mente humana. Estas tecnologías comprenden procesos desde el aprendizaje automático (machine learning) hasta el procesamiento del lenguaje natural y gracias a técnicas como el aprendizaje profundo (deep learning), se podrían mejorar la prevención y el diagnóstico de cáncer, ya que, a través de un conjunto de algoritmos, es capaz de aprender a identificar patrones complejos en grandes cantidades de datos y funciona de forma similar a las neuronas del cerebro humano, lo que aporta grandes beneficios tanto para el paciente como para el profesional sanitario.

Hay diversos proyectos e investigaciones que están comprobando cómo esta tecnología puede aportar mucho en la prevención, diagnóstico y tratamiento del cáncer. Algunas de las aplicaciones que han dado resultados positivos, permiten que, junto al conocimiento del profesional médico, se obtengan diagnósticos más precisos, se mejore la calidad de las pruebas radiológicas, evitando repeticiones innecesarias o pruebas más invasivas como las biopsias y se apliquen tratamientos más adecuados.

Uno de los proyectos es el de «Identificación y clasificación basada en resonancia magnética de los principales tipos de tumores intracraneales mediante una red neuronal convolucional 3D: un análisis retrospectivo multiinstitucional»¹⁷. Colaborando con Satrajit Chakrabarty y los Dres. Sotiras y Marcus fueron Mikhail Milchenko, Ph.D., Pamela LaMontagne, Ph.D. y Michael Hileman, B.S. En este estudio se ha desarrollado un modelo de deep learning que es capaz de clasificar un tumor cerebral como uno de los seis tipos comunes utilizando un solo escaneo de resonancia magnética 3D. Los investigadores dividieron un total de 2.105 exploraciones en tres subconjuntos de datos: 1.396

para entrenamiento, 361 para pruebas internas y 348 para pruebas externas. Con los datos de las pruebas internas, el modelo logró una precisión del 93,35% (337 de 361) en siete clases de imágenes (una clase sana y seis clases de tumores).

Otro estudio para recalcar es el “Deep Learning in Breast Cancer Detection and Classification”¹⁸ En este artículo, presentan los modelos más recientes de detección y clasificación del cáncer de mama que son modelos basados en el aprendizaje automático donde se analizan en forma de estudio comparativo.

3.3 Aprendizaje automático.

En función del tipo de aprendizaje se clasifican en 3 tipos que tenemos representados en el esquema siguiente:



Ilustración 6: esquema sobre el aprendizaje automática

- **Supervisado:** se tratan de datos que contienen etiquetas, es decir, se disponen de los datos etiquetados como un conjunto de filas y columnas donde cada fila son los datos de una sola observación de un objeto de estudio y las columnas son características observadas¹⁸. Los algoritmos más utilizados con este tipo de aprendizaje son:

-Clasificación: es un método en el que un algoritmo aprende a clasificar nuevas observaciones a partir de ejemplos de datos etiquetados, es el proceso de predecir una categoría o etiqueta de clase discreta¹⁹. Como ejemplo tenemos:

-Clasificación de Naïve Bayes: están basados en el teorema de Bayes y clasifican cada valor como independiente de cualquier otro. Lo que permite predecir una clase o categoría en función de un conjunto dado de características, utilizando la probabilidad.

-Árboles de decisión: es un algoritmo con estructura de árbol similar a un diagrama de flujo que utiliza un método de bifurcación para ilustrar cada resultado posible de una decisión. Cada nodo dentro del árbol representa una prueba en una variable específica, y cada rama es el resultado de esa prueba.

-Support Vector Machines (SVM): son modelos que analizan datos para clasificación y análisis de regresión. El objetivo de la optimización es establecer una línea de decisión que separe las clases maximizando el margen entre esta línea y los puntos de muestra cercanos a este hiperplano.

-KNN: son algoritmos basados en instancia, que se caracterizan por memorizar el conjunto de datos de entrenamiento. El algoritmo encuentra las k muestras que son más cercanas al punto que se quiere clasificar, basando sus predicciones en la distancia métrica.

-Regresión Logística: es un tipo de análisis de regresión utilizado para predecir el resultado de una variable categórica en función de las variables independientes o predictoras. Se utiliza para modelar la probabilidad de un evento ocurriendo en función de otros factores.

-Redes Neuronales: Una red neuronal artificial (RNA) comprende unidades dispuestas en una serie de capas, cada una de las cuales se conecta a las capas anexas. Las RNA se inspiran en los sistemas biológicos, como el cerebro, y en cómo procesan la información.

-Regresión: describen la relación entre una variable de respuesta (salida) y una o varias variables de predicción (entrada). El

modelo de aprendizaje automático debe estimar y comprender las relaciones entre las variables. Es el proceso de predicción de un valor continuo en contraposición a la predicción un valor categórico en Clasificación¹⁹. Como ejemplo tenemos:

-Regresión por mínimos cuadrados: este método consiste en minimizar la suma de cuadrados del error de predicción del modelo. Se trata de encontrar un método para hallar una recta que se ajuste de una manera adecuada a la nube de puntos definida por todos los pares de valores muestrales

-Regresión Ridge/Lasso: es un método estadístico que trata de modelar la relación entre una variable continua y una o más variables independientes mediante el ajuste de una ecuación lineal. La regularización Lasso penaliza la suma del valor absolutos de los coeficientes de regresión, en cambio la de Ridge penaliza la suma de los coeficientes elevados al cuadrado.

- **No supervisado:** Se trata de datos que no contienen etiquetas, se basa en algoritmos que buscan patrones, relaciones, tendencias y/o características, de manera que el modelo trabaje por su cuenta para descubrir información que puede que no sea visible para el ojo humano¹⁸. Los algoritmos más utilizados con este tipo de aprendizaje son:

-**Reconocimiento de patrones:** funciona mediante la clasificación de los datos de entrada en objetos o clases en función de características clave, ya sea mediante la clasificación supervisada o no supervisada¹⁹. Como ejemplo tenemos euclat, aprior y FP-Growth.

-**Clustering:** se emplea en redes neuronales para el análisis de datos exploratorio a fin de localizar patrones ocultos o agrupaciones de datos. Este proceso implica la agrupación de datos por similitud¹⁹. Como ejemplo tenemos:

-K-Medias: El algoritmo funciona mediante la búsqueda de grupos dentro de los datos, con el número de grupos representados por la variable K. Funciona de manera iterativa para asignar cada punto de datos a uno de los k grupos según las características proporcionadas

-**Reducción de dimensiones:** se define como una forma de convertir un conjunto de datos de dimensiones elevadas en un conjunto de datos de dimensiones menores, asegurando que la información que proporciona es similar en ambos casos¹⁹. Como ejemplo tenemos:

-PCA: El análisis de componentes principales es una técnica que se utiliza para reducir el número de variables de entrada en el conjunto de datos de entrenamiento. Ayuda a encontrar las características más significativas en un conjunto de datos y facilita el trazado de datos en 2D y 3D, además, ayuda a encontrar una secuencia de combinaciones lineales de variables.

-LDA: El Análisis Discriminante Lineal es una técnica de reducción de dimensionalidad con el objetivo de proyectar las características en el espacio dimensional superior en un espacio de dimensión inferior para evitar la dimensionalidad y también reducir los recursos y los costos computacionales.

- **Aprendizaje por refuerzo:** no es un tipo de aprendizaje supervisado, porque no se basa estrictamente en un conjunto de datos etiquetados, sino en la monitorización de la respuesta a las acciones tomadas. Tampoco es un aprendizaje no supervisado, ya que, cuando modelamos ya sabemos cuál es la respuesta esperada. En este caso, la máquina o el programa aprenden mediante prueba y error. El sistema prueba distintas soluciones a un mismo problema hasta que acaba por comprender cuál es la mejor decisión para obtener el resultado deseado y con la mejor calidad posible. En consecuencia, este sistema enseña la máquina a través del proceso de ensayo y error. Como ejemplo tenemos:
 - Q-Learning: es un algoritmo de aprendizaje basado en valores donde se actualiza la función de valor basada en una ecuación (particularmente la ecuación de Bellman).

3.4 Aprendizaje profundo.

Una neurona artificial está formada por un conjunto de entradas que son los enlaces o interconexiones por donde reciben información del exterior de la red o de otras neuronas.²⁰ Cada una de las entradas está ponderada por un peso que determina la importancia de la información que recibe por esa interconexión y que es un valor que se va ajustando durante el entrenamiento de la red neuronal para minimizar el error de la red y que los resultados sean óptimos. La capa de entrada corresponde a nuestros predictores/entidades y la capa de salida a nuestras variables de respuesta más fiables.

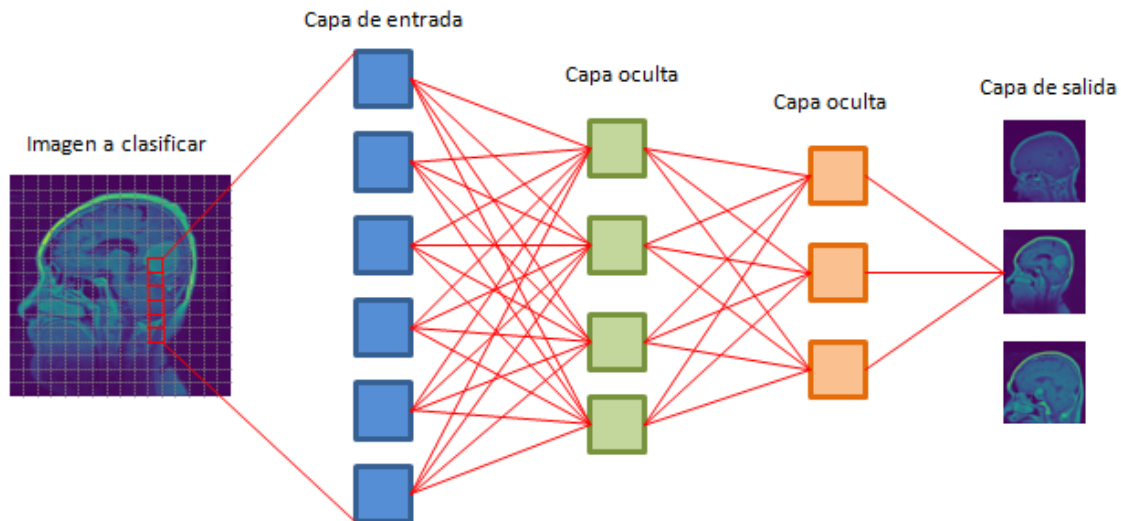


Ilustración 7: esquema sobre las redes neuronales

Las redes neuronales, son estructuras computacionales utilizadas en el campo del aprendizaje automático y están en el núcleo de los algoritmos de deep learning. Su nombre y estructura se inspiran en el cerebro humano, e imitan la forma en la que las neuronas biológicas se señalan entre sí. Crean modelos para encontrar la combinación que mejor se ajuste y poder entrenar la red neuronal. Una red ya entrenada se puede usar luego para hacer predicciones o clasificaciones, es decir, para "aplicar" la combinación.

En base a su estructura las redes neuronales pueden ser:

- **Monocapa:** la red neuronal está formada por una única capa de neuronas que recibe valores del exterior de la red, las procesa y genera unos valores de salida que entrega como resultado. Las neuronas de la capa no se interconectan entre ellas.
- **Multicapa:** consta de una capa de entrada que recibe valores del exterior, una serie de capas intermedias u ocultas que van procesando la información y una capa de salida que entrega los resultados de la red neuronal.²⁰ Las neuronas de una capa se interconectan con todas las neuronas de la capa siguiente, pero no entre ellas.
- **Convolutional:** son similares a las redes neuronales multicapa, pero la diferencia radica en que en estas las neuronas de cada capa no se interconectan con todas las de la capa siguiente, si no con un subconjunto de estas, y no se interconectan entre las neuronas de la misma capa. En este caso se habla de especialización de las neuronas y se reduce la cantidad de unidades necesarias y la complejidad de los sistemas que utilizan este tipo de redes.
- **Concurrentes:** en estas redes las neuronas no se organizan en capas, sino que las neuronas están interconectadas entre ellas de manera no estructurada en capas. Esto permite que estas redes tengan memoria, es decir, que la información generada en iteraciones anteriores afecte al resultado del procesamiento en un tiempo futuro.

- **Red totalmente conectada (FCN)** en nuestra última capa, que es una red totalmente conectada, se envían los datos aplanados a una red totalmente conectada, básicamente se transforman los datos para hacer clases que se necesitan obtener de la red como salida.

Una red neuronal es un conjunto de nodos, llamados neuronas, conectadas entre sí y que forman una estructura de capas, llamadas capas de aprendizaje. A través de estas capas, la red procesa unos datos numéricos de entrada para dar un resultado. Es necesario conocer la estructura de las capas al detalle para conseguir el modelo óptimo. Existen diferentes tipos de redes neuronales, pero las que se utilizan normalmente para realizar clasificación de imágenes son las CNN, por otra parte, tenemos las RNN que se utilizan para aplicaciones como el modelado del lenguaje.

La Red Neuronal Artificial (ANN) es capaz de aprender cualquier función no lineal debido a su función de activación, esto ayuda a la red a aprender cualquier relación compleja entre entrada y salida²⁰.

Para realizar clasificación de imágenes con ANN, el primer paso es convertir una imagen de 2 dimensiones en un vector de 1 dimensión antes de entrenar el modelo. Esto provoca que el número de parámetros entrenables aumente drásticamente con un aumento en el tamaño de la imagen. Además, la ANN pierde las características espaciales de una imagen, es decir, la disposición de los píxeles en una imagen y tampoco puede capturar información secuencial en los datos de entrada que se requiere para tratar con datos de secuencia.

La Red neuronal recurrente (RNN) captura la información secuencial presente en los datos de entrada, es decir, este método es ideal para el análisis de texto y voz debido a que captura la dependencia entre las palabras en el texto mientras hace predicciones²⁰. Los RNN comparten los parámetros en diferentes pasos de tiempo, esto da como resultado menos parámetros para entrenar y disminuye el costo computacional. Su problemática a la hora de clasificar imágenes es que sufren el problema del gradiente de desaparición y explosión.

Por último, tenemos las **redes neuronales convolucionales (CNN)**²⁰ que son las que se han escogido para realizar este estudio debido a que son muy efectivas para reducir el número de parámetros sin perder la calidad de los modelos. Las imágenes tienen una alta dimensionalidad, ya que cada píxel se considera una característica, por esto es tan importante el concepto de reducción de dimensionalidad donde se adapta a la gran cantidad de parámetros en una imagen. Además, las CNN capturan las características espaciales de una imagen, esto permite que el modelo identifique el objeto con precisión, la ubicación de un objeto, así como su relación con otros objetos en una imagen.

Las Redes neuronales convolucionales (CNN) son un tipo de red neuronal artificial (ANN) con aprendizaje supervisado, utilizada en el reconocimiento y procesamiento de imágenes que está especialmente diseñada para procesar datos (píxeles), toma cualquier imagen de entrada y encuentra un patrón en la imagen, la procesa y la clasifica en varias categorías, en este estudio se utilizan para clasificar las imágenes en 3 tipos de cáncer de cerebro.

Recordemos que la red neuronal deberá aprender por sí sola a reconocer una diversidad de objetos dentro de imágenes y para ello se necesita una gran cantidad de imágenes, en este estudio se utilizan 3064 imágenes que contienen radiografías de tumores de cerebros, de esta manera, la red podrá captar sus características únicas y reconocer el tipo de tumor en cualquier ángulo. Para ello, la CNN contiene varias capas ocultas especializadas y con una jerarquía, esto quiere decir que las primeras capas pueden detectar líneas, curvas y se van especializando hasta llegar a capas más profundas que reconocen formas complejas como todos los detalles de una radiografía con IMR.

Para construir el modelo se utilizan las redes neuronales de convolución que requieren de una operación matemática entre sus capas llamada convolución. El objetivo principal de la capa convolucional es extraer características de las imágenes y aprender todas las características de la imagen que ayudarían en las técnicas de detección de objetos. La capa de entrada contendrá algunos valores de píxeles con cierto peso y altura, nuestros núcleos o filtros se contornean alrededor de la capa de entrada y darán resultados que recuperarán todas las entidades con menos dimensiones. El proceso para realizarlo sería primero que las neuronas de la capa de entrada reciben información entrante de los datos que procesan y distribuyen a las capas ocultas, a la vez, esa información es procesada por capas ocultas y pasa a las neuronas de salida. La información en esta red neuronal se procesa en términos de una función de activación que imita a las neuronas cerebrales. Cada neurona contiene un valor de funciones de activación y un valor umbral que es el valor mínimo que debe poseer la entrada para que pueda ser activada. La tarea de la neurona es realizar una suma ponderada de todas las señales de entrada y aplicar la función de activación sobre la suma antes de pasarla a la siguiente capa (oculta o de salida).

Se debe agregar una función de activación a la red en cualquier lugar entre dos capas convolucionales y al final de la red para ayuda a tomar la decisión sobre qué información debe seguir hacia adelante y cuál no. Como el modelo no sigue una relación lineal se debe utilizar funciones de activación no lineales. Las funciones más utilizadas son las siguientes:

Función de activación sigmoide tiene una gran eficiencia²¹, se trata de un enfoque probabilístico hacia la toma de decisiones y produce una salida entre 0 y 1, por lo que cuando tenemos que tomar una decisión o predecir una salida utilizamos esta función de activación debido a que el rango es el mínimo, por lo tanto, la predicción será más precisa.

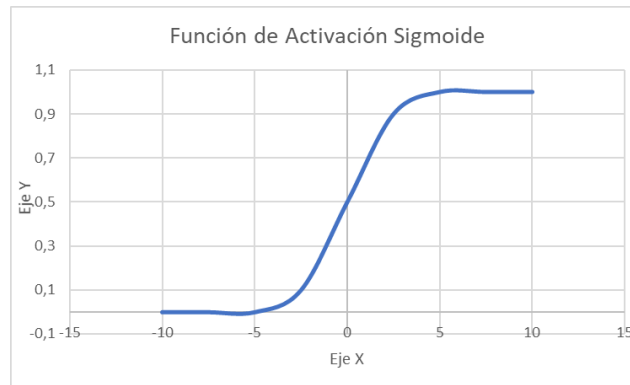


Ilustración 8: representación gráfica de la función de activación Sigmoide

Función de activación tangente hiperbólica (Tanh) esta función de activación es un poco mejor que la función sigmoide, también se usa para predecir o diferenciar entre dos clases²¹, pero mapea la entrada negativa solo en cantidad negativa y oscila entre -1 y 1.

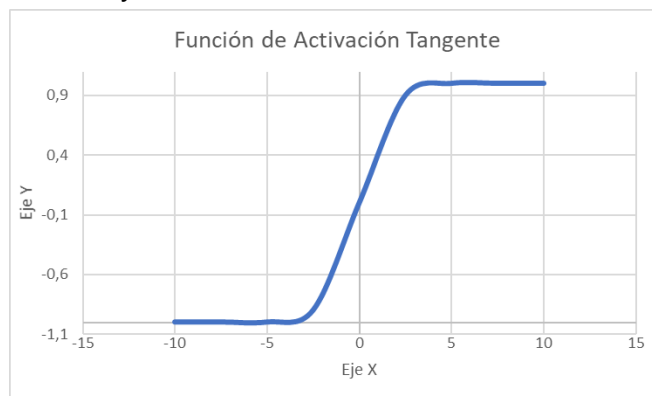


Ilustración 9: representación gráfica de la función de activación tangente hiperbólica

Función de activación de Softmax se utiliza en la última capa normalmente, es decir, la capa de salida para la toma de decisiones²¹, el softmax básicamente da valor a la variable de entrada de acuerdo con su peso, y la suma de estos pesos es eventualmente uno.

Para los problemas de clasificación de múltiples clases la función de softmax es más utilizada. También produce una salida entre 0 y 1.

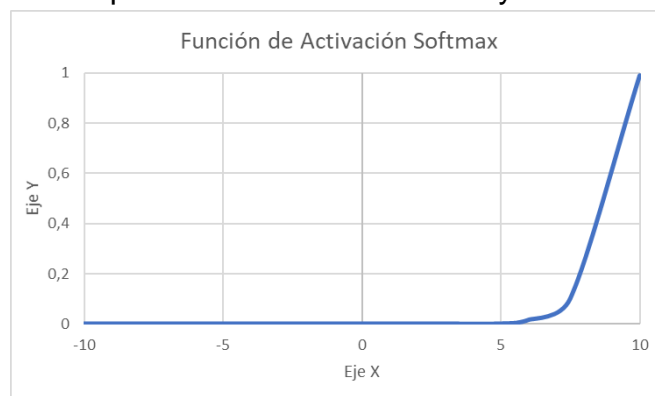


Ilustración 10: representación gráfica de la función de activación Softmax.

Función de activación ReLU (unidad lineal rectificada)²¹ es la función de activación más utilizada para este tipo de redes neuronales y es la que utilizaremos para este estudio junto con la de Softmax, consiste en $f(x)=\max(0,x)$. Convierte todos los valores negativos en 0, por tanto, todos los valores van de 0 a infinito.

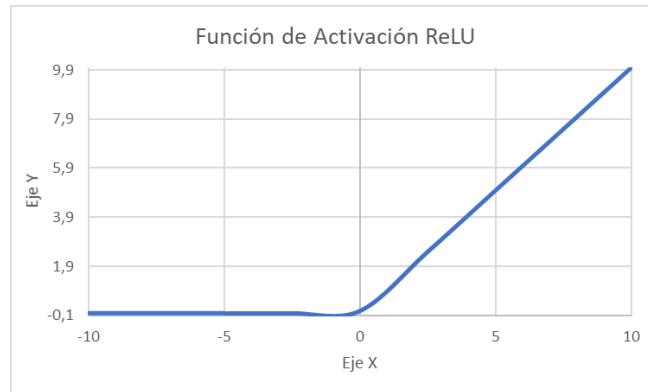


Ilustración 11: representación gráfica de la función de activación ReLU

Para comparar cómo es el resultado del modelo contra el resultado real se utiliza la “Función Coste”. Idealmente queremos que nuestro coste sea cero, es decir cuando el valor de la predicción es igual al resultado real. A medida que se entrena el modelo se irán ajustando los pesos de inter-conexiones de las neuronas de manera automática hasta obtener buenas predicciones. A ese proceso de “ir y venir” por las capas de neuronas se le conoce como Back-Propagation. Para poder ajustar los pesos de las conexiones entre neuronas haciendo que el coste se aproxime a cero usaremos una técnica llamada Gradient Descent²². Esta técnica permite encontrar el mínimo de una función. En nuestro caso, buscaremos el mínimo en la Función Coste.

Funciona cambiando los pesos en pequeños incrementos luego de cada iteración del conjunto de datos. Al calcular la derivada (o gradiente) de la Función Coste en un cierto conjunto de pesos, podremos ver en qué dirección desciende hacia el mínimo global. Para minimizar la función de coste necesitaremos iterar por el conjunto de datos muchas veces, para ello necesitamos una gran capacidad computacional.

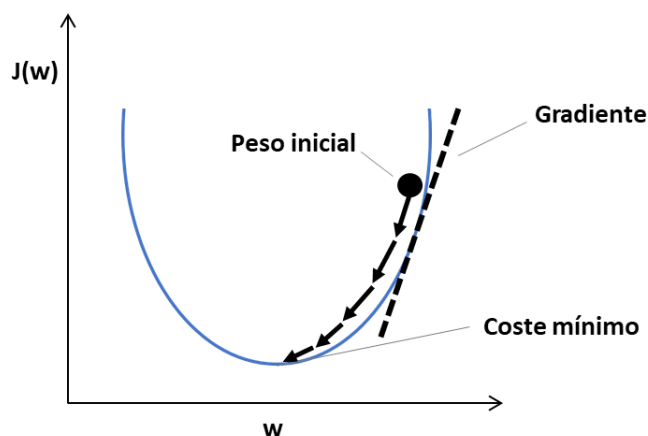


Ilustración 12: representación gráfica del funcionamiento de gradiente descendente.

4 Metodología

El objetivo del estudio es distinguir radiografías de tumores de cabeza, lo primero que se debe hacer es alimentar los píxeles de la imagen en forma de matrices a la capa de entrada de la red neuronal. Las capas ocultas llevan la extracción de características mediante la realización de varios cálculos y operaciones. Hay varias capas ocultas como la convolución, la ReLU y la capa de agrupación que realiza la extracción de características de la imagen. Entonces, finalmente, hay una capa totalmente conectada que identifica el objeto exacto en la imagen. Se puede entender muy fácilmente a partir de la siguiente figura:

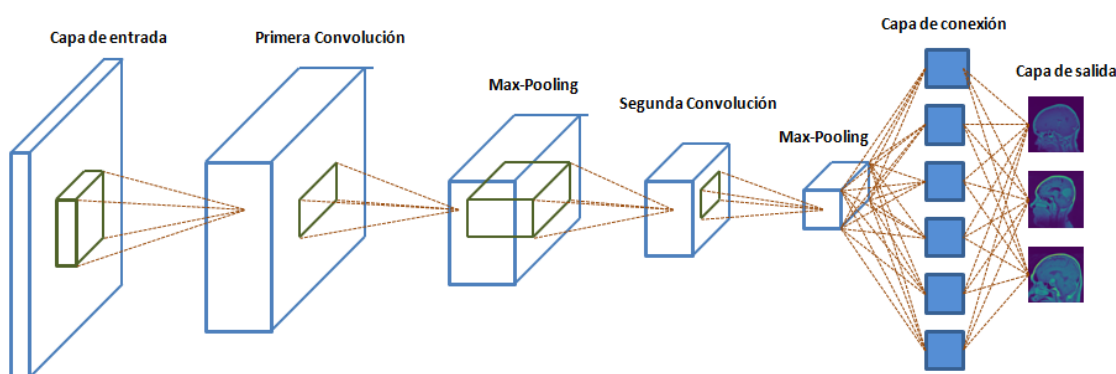


Ilustración 13: representación esquemática del funcionamiento de las redes neuronales convolucionales.

4.1. Recolección de datos.

Primero de todo se importan las imágenes a la web Google Colab para poder realizar un análisis exploratorio de las imágenes. En nuestro caso tenemos 3 carpetas con cada clase de tumor, la 1 contiene las radiografías con el tumor Meningioma, la carpeta 2 tumores de glioma y la carpeta 3 tumores pituitarios.²³ Nuestras imágenes contienen 512x512 píxeles de alto y ancho, además necesitamos 3 canales debido a que la imagen contiene color (rojo, verde, azul), por tanto, esto equivaldría $512 \times 512 \times 3 = 262.144$ neuronas, es decir, nuestra capa

<Lorena Pelegero Alonso>

de entrada sería de esta cantidad de neuronas. Al tratarse de imágenes con tanta resolución provocaría un tiempo de procesamiento muy largo, además, al exportar las imágenes de tal tamaño agotamos toda la memoria RAM disponible del ordenador, por tanto, la solución a este problema es importar las imágenes a 200x200 píxeles, de esta manera, toda la información de las imágenes se mantendrá y se ahorra tiempo de procesamiento innecesario.

4.2. Exploración de los datos.

Para poder entrenar el modelo en los siguientes pasos, se debe tener estructurado el directorio de la siguiente manera:

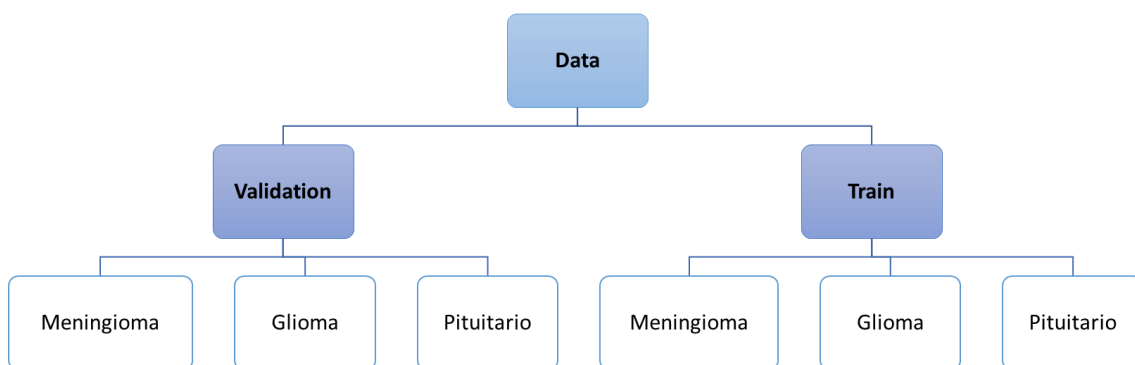


Ilustración 14: esquema de la estructura de las carpetas para la creación del modelo.

Los datos de entrenamiento son los datos que usamos para entrenar un modelo. La calidad de nuestro modelo dependerá de la calidad de los datos que le estamos dando. En cambio, los datos de validación son los datos que nos guardamos para comprobar si el modelo que hemos generado a partir de los datos de entrenamiento funciona correctamente, es decir, si las predicciones que realiza el modelo son acertadas o no. Es importante que el conjunto de datos de validación tenga un volumen suficiente como para generar resultados estadísticamente significativos y representativos.

Debido a que se tiene diferente número de imágenes de cada clase, se cogen únicamente 708 de cada una y se separan cogiendo el 20% de las imágenes para Validation (140 imágenes) y el 80% para la carpeta Train (568 imágenes).

4.3. Preprocesamiento de los datos.

Antes de poder utilizar los datos se normalizan los valores. Los colores de los píxeles tienen valores que van de 0 a 255, se hará una transformación de cada pixel: "valor/255" y se quedará siempre un valor entre 0 y 1.

4.4. Creación del modelo

Una vez se tienen los datos de entrenamiento listos se puede generar el modelo de la siguiente manera:

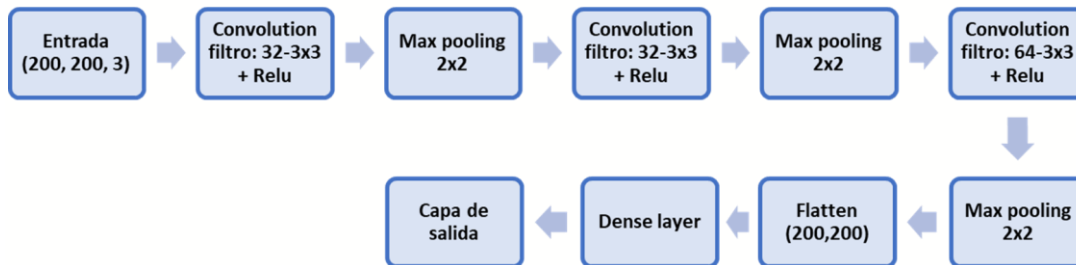


Ilustración 15: esquema del modelo que se ha implementado

Se crea una Red Neuronal con los valores de entrada (Input Layer) y con 3 neuronas de Salida (Output Layer). Las neuronas que hay en medio se llaman capas escondidas (Hidden Layers) y se pueden tener muchas, cada una con una distinta cantidad de neuronas. Todas las neuronas están interconectadas unas con otras en las distintas capas.

Para poder lanzar los datos a la capa de entrada se debe indicar el ancho de las imágenes de entrada, la altura de las imágenes de entrada y la profundidad, es decir, el número de canales en nuestras imágenes de entrada. En este caso 3 para imágenes RGB estándar ya que son imágenes a color.

La capa de entrada recibe los datos de entrada, cada conexión de nuestra red neuronal está asociada a un peso y este peso dictamina la importancia que tendrá esa relación en la neurona al multiplicarse por el valor de entrada. Una vez pase a la primera capa oculta, ahí se realizarán cálculos matemáticos con nuestras entradas para establecer si se debe saltar a la siguiente capa. Quien decide si pasa a la siguiente capa es la función de Activación, en nuestro modelo hemos utilizado la función de activación Relu en las capas escondidas debido a que manejamos una no linealidad, pero, en la capa de salida hemos utilizado la función Softmax debido a que se trata de una clasificación multiclase. Si la suma de sus valores recibidos, que previamente se han multiplicado por el peso de la conexión, supera un umbral entonces la neurona se activa y dispara un valor hacia la siguiente capa conectada.

Seguidamente se realiza una toma de muestra (subsampling) para poder reducir la cantidad de neuronas antes de la nueva convolución donde deberán prevalecer las características más importantes que detectó cada filtro, para realizarlo se utiliza la función "MaxPooling" donde se obtiene el valor máximo del

<Lorena Pelegero Alonso>

píxel, de esta manera, se reduce el tamaño de la imagen de entrada. Para evitar el sobreajuste se utiliza la función Dropout y para mejorarlo se usa el RMSprop que usa momento para evitar el problema de quedarte atascado en un mínimo local.

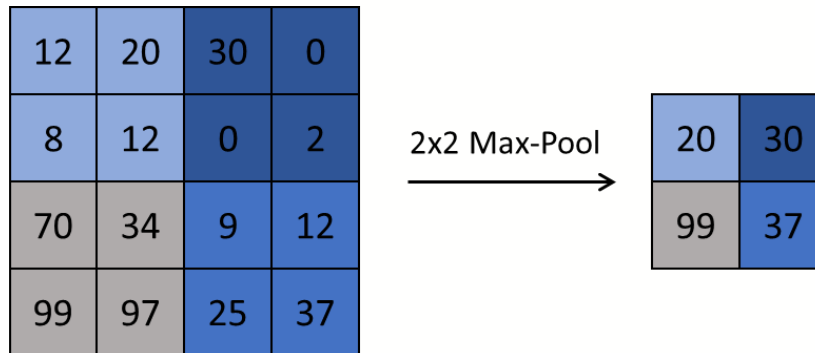


Ilustración 16: esquema sobre el funcionamiento del Max-Pooling.

La capa de Salida devuelve la predicción realizada, para transformarla de matriz a vector se utiliza la función Flatten, en nuestro caso de 3 resultados discretos las salidas podrán ser “1 0 0” para Meningioma, “0 1 0” para Glioma y “0 0 1” para Pituitario.

Para realizar las convoluciones se toman grupos de píxeles cercanos de la imagen de entrada y se operará matemáticamente contra una matriz llamada kernel que recorre todas las neuronas de entrada de izquierda-derecha, de arriba-abajo y genera una nueva matriz de salida que se convertirá en nuestra nueva capa de neurona oculta. Como en este estudio se trabajan con imágenes a color tendremos 3 matrices de kernel y luego se suman y formarán una salida. Los conjuntos de kernel se denominan filtros. Una vez obtenido un conjunto de filtros y una entrada se generará un mapa de características y habremos logrado la primera capa de convolución capaz de detectar características primitivas como líneas o curvas. A medida que hagamos más capas con las convoluciones, los mapas de características serán capaces de reconocer formas más complejas. Al realizar la predicción con este modelo se obtiene un rendimiento del 0.4727.

MEJORAR EL MODELO

Observamos que en la primera implementación del modelo se obtiene una precisión muy baja, esto se puede deber a diferentes factores.

- Que el modelo es insuficiente (underfit), es decir, el modelo no es lo suficientemente complejo como para aprender los patrones en los datos. La solución sería aumentar el número de parámetros entrenables aumentando las capas y el tamaño de cada capa en ese caso.
- Que el modelo está sobreajustado (overfit) Esto significa que el modelo es tan complejo que se ha sobreajustado en los datos de entrenamiento.

<Lorena Pelegero Alonso>

y no se ha generalizable en los datos de prueba. La solución sería reducir el número de parámetros de entrenamiento.

- El diseño del modelo no es adecuado, es decir, el modelo no se ajusta a los datos que se debe clasificar.
- No se han seleccionado un optimizador o funciones de activación adecuadas para nuestro modelo. La solución sería modificarlos para que encaje con nuestros parámetros.

En este caso, el modelo es insuficiente, debido a que se dispone de un número insuficiente de imágenes de entrenamiento, por tanto, se utilizará la función ImageDataGenerator para generar más imágenes a partir de las que ya se tienen, de esta manera, el modelo podrá detectar más patrones para poder clasificarlas.

5 Resultados

Para evaluar los resultados de los dos modelos se definen los conceptos que se van a utilizar.

-La precisión (accuracy) es la cantidad de clasificaciones correctas respecto a la cantidad total de clasificaciones.

-La precisión de entrenamiento (Train accuracy) es la precisión del modelo construido con los datos de entrenamiento.

-La precisión de prueba (Test accuracy) es la precisión del modelo para datos que no ha visto nunca.

Los resultados de los dos modelos:

	Modelo inicial	Modelo Mejorado
Precisión de entrenamiento	0.4696	0.929
Precisión de prueba	0.4696	0.928

Tabla 3: resultados de la precisión del modelo inicial y mejorado

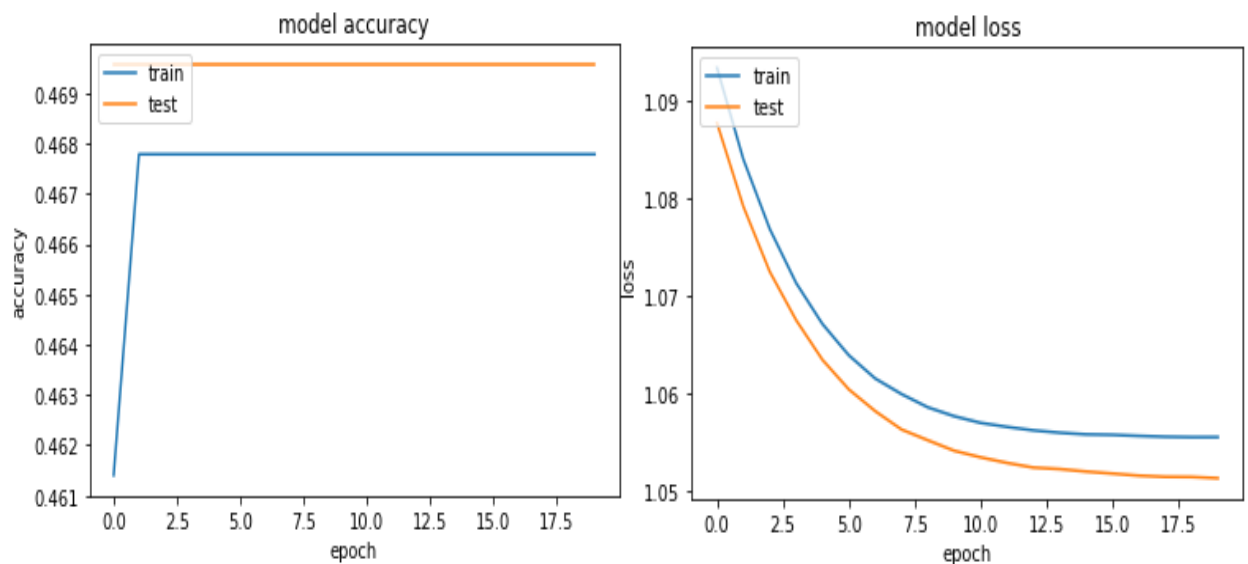


Ilustración 17: representación gráfica de la precisión del primer modelo:

Para analizar los resultados se representa gráficamente la precisión y la pérdida vs los ciclos del ajuste del modelo.

La curva de aprendizaje y los gráficos de pérdida representa la precisión del modelo de red neuronal con respecto al conjunto de datos de entrenamiento y validación después de que se ejecuta cada época de entrenamiento. La curva de aprendizaje ayuda a determinar el valor óptimo de los hiper parámetros para crear la red neuronal óptima a fin de evitar el sobreajuste y ayudar a lograr una mayor generalización al realizar predicciones correctas. A partir del gráfico de precisión, se puede ver que el modelo no está bien ajustado, debido a que se mantiene la precisión constante a partir del primer ciclo. A partir de la gráfica de pérdida, se puede ver que el modelo tiene una buena tasa de aprendizaje tanto en los datos de entrenamiento como en el conjunto de datos de validación. Como se comentó en el apartado anterior, este modelo no tiene datos suficientes de entrenamiento y por tanto, el modelo no se ajusta correctamente.

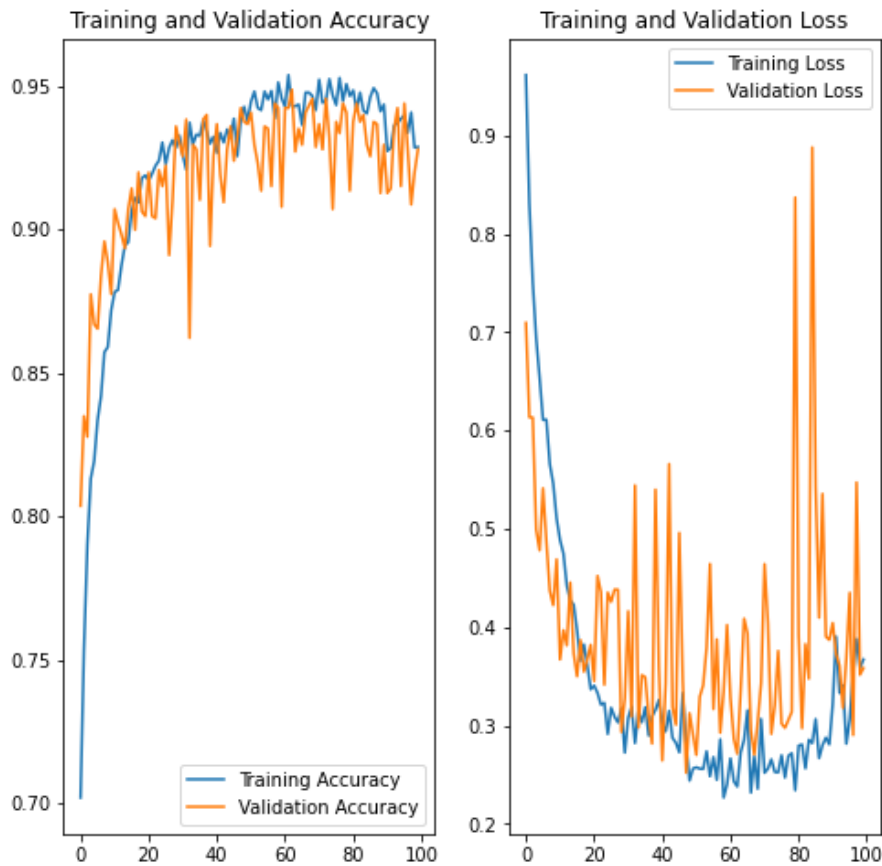


Ilustración 18: representación gráfica de la precisión del modelo mejorado

A partir del gráfico de precisión, se puede ver que el modelo mejorado si que está suficientemente entrenado ya que la tendencia decae en los últimos ciclos. A partir de la gráfica de pérdida, se puede ver que el modelo tiene una buena ratio de aprendizaje para los dos conjuntos de datos.

Otro dato significativo para el análisis del modelo son los resultados de la especificidad y la sensibilidad.

La sensibilidad es la proporción de casos positivos que fueron correctamente identificadas por el algoritmo.

$$\text{Sensibilidad} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Negativos}}$$

La especificidad es la proporción de casos negativos que fueron correctamente identificadas por el algoritmo.

$$\text{Especificidad} = \frac{\text{Verdaderos Negativos}}{\text{Verdaderos Negativos} + \text{Falsos Positivos}}$$

<Lorena Pelegero Alonso>

A continuación se representa gráficamente la especificidad y la sensibilidad respecto al número de ciclos:

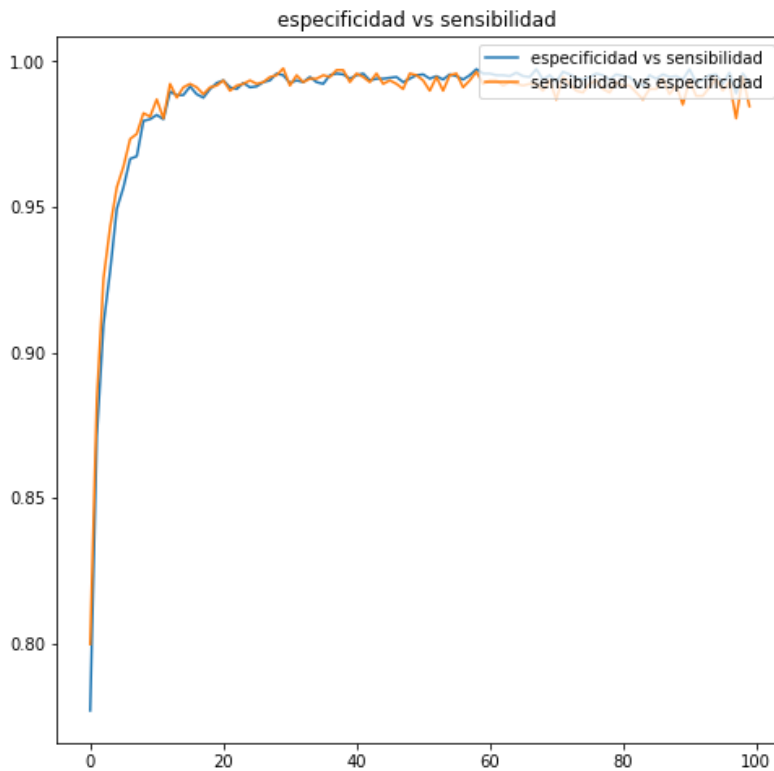


Ilustración 19: representación gráfica de la sensibilidad y especificidad vs los ciclos del modelo mejorado

A partir del gráfico se observa que se ha obtenido una sensibilidad y una especificidad muy elevada lo cual es un indicador de que el modelo se ajusta correctamente a los datos. Para poder evaluar los valores exactos de sensibilidad y especificidad se construye la siguiente matriz de confusión con los datos de validación:

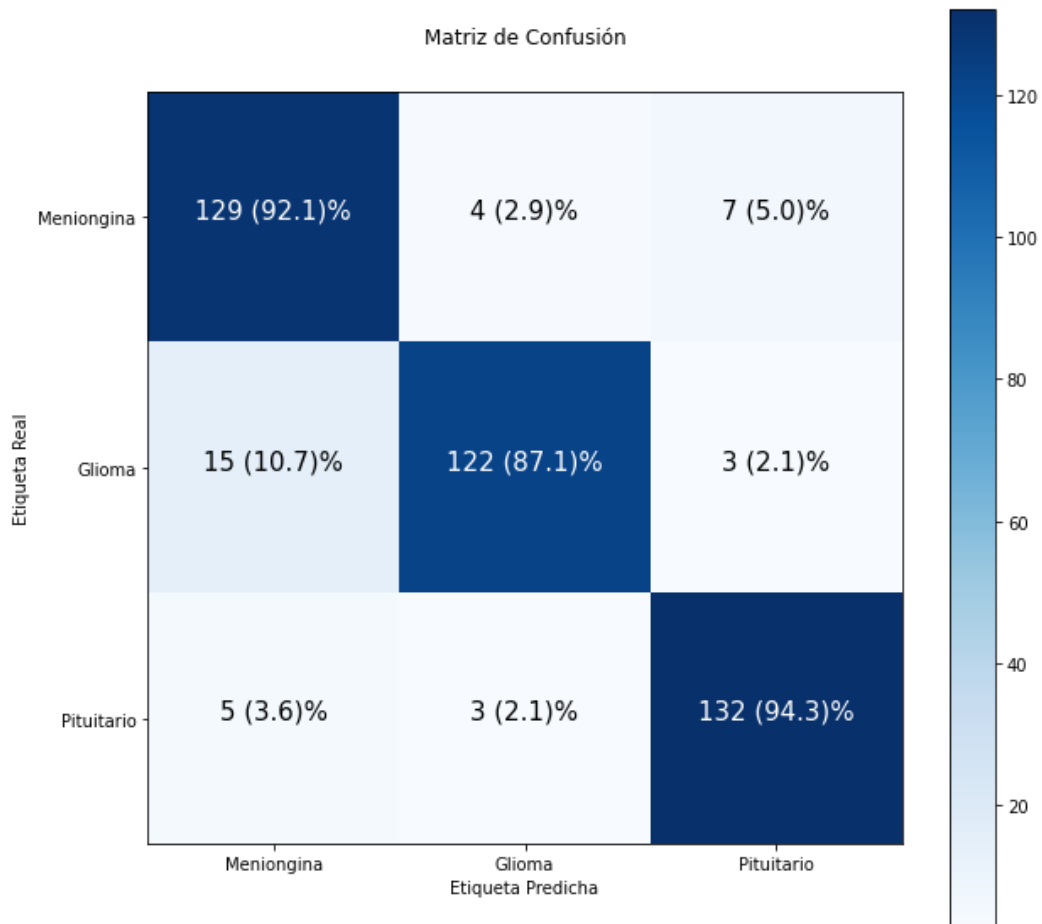


Ilustración 20: matriz de Confusión con los resultados del modelo

La matriz de confusión es una herramienta que permite analizar los resultados de cómo trabaja un algoritmo de aprendizaje supervisado. Tiene formato de tabla donde se representan en las columnas las predicciones de cada clase mientras que en las filas se representa la clase real. A partir de la comparación de las clases predichas respecto a las clases reales se puede calcular el grado de precisión, exactitud, sensibilidad y especificidad del modelo.

Para la clase meniingina, se obtienen los siguientes parámetros:

TP	FN	FN
FP	TN	TN
FP	TN	TN

- **TP(verdaderos positivos)=129**
- **FP(Falsos positivos):15+5=20**
- **FN(Falsos negativos):4+7=11**
- **TN(Verdaderos negativos):
122+3+3+132=260**

Ilustración 21: representación gráfica de los TP, FN, FP y TN

$$\text{Exactitud} = \frac{\text{Verdaderos Positivos} + \text{Verdaderos Negativos}}{\text{Total}} = \frac{129+260}{420} = 0.926$$

$$\text{Precisión} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Positivos}} = \frac{129}{129+20} = 0.866$$

$$\text{Sensibilidad} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Negativos}} = \frac{129}{129+11} = 0.921$$

$$\text{Especificidad} = \frac{\text{Verdaderos Negativos}}{\text{Verdaderos Negativos} + \text{Falsos Positivos}} = \frac{260}{260+20} = 0.929$$

Se construye la siguiente tabla con todos los resultados de cada clase:

Clase	Métrica	Exactitud	Precisión	Sensibilidad	Especificidad
Meniנגina		0.926	0.866	0.921	0.929
Glioma		0.940	0.946	0.871	0.975
Pituitario		0.957	0.930	0.943	0.964

Tabla 4: resultados de la exactitud, precisión, sensibilidad y especificidad para cada clase.

6 Discusión

Analizando los resultados obtenidos en la tabla anterior se observa que la exactitud para la clase 1 (Meniongina) es la más baja, es posible que se deba a que la selección de imágenes de entrenamiento no ha sido totalmente heterogénea y difieran de las de validación, esto provoca que haya algunas imágenes de validación que no logra asociar con las de entrenamiento. Igualmente observando los valores de exactitud podemos ver que la cercanía entre los datos predichos y los datos reales es muy elevada. En la precisión nos indica que la proporción de número de predicciones correctas con el total de predicciones es muy elevada, dándonos el resultado más bajo para la clase 1. Enfocándonos en la sensibilidad se puede confirmar que la capacidad del algoritmo de detectar los casos positivos es muy elevada con poco margen de error, lo mismo pasa con la especificidad que nos indica la relación entre la cantidad de casos negativos detectados con la cantidad de casos negativos totales.

7 Valoración económica

En esta sección se hace un análisis de costes de este proyecto.

Para poder aumentar la velocidad de cómputo y la cantidad de memoria de RAM se han invertido 10€/mes en la suscripción de Google Colab.

Si tenemos en cuenta la mano de obra para realizar el proyecto y teniendo en cuenta que estaría alrededor de 20€/hora, el gasto total se refleja en la siguiente tabla:

Inversiones	Tiempo	Precio	Total
Creación modelo	120h	20€/h	2400€
Redacción Memoria	50h	20€/h	1000€
Google Colab	3 meses	10€/mes	30€
Total	-	-	3430€

Tabla 5: coste para la realización del proyecto.

El coste total de la realización de este proyecto es de 3430€ con un tiempo total de 170h.

8 Conclusiones

De todos los resultados obtenidos en este trabajo se pueden obtener las siguientes conclusiones:

- 3060 imágenes no son suficientes para poder conseguir un modelo con suficiente exactitud. Por ello, se ha tenido que utilizar la función de Data Generator para aumentar los datos de entrenamiento.
- Las CNN son la mejor manera de implementar modelos de clasificación a partir de imágenes debido a que son muy efectivas para reducir el número de parámetros sin perder la calidad de los modelos.
- 93% de exactitud es un porcentaje elevado pero no suficiente si hablamos de fines médicos, por tanto, el uso del modelo debe ser para orientar y no para decidir.

Se han establecido 7 objetivos para este trabajo, de los cuales se han cumplido 6, es decir, el 86%. Los objetivos cumplidos son:

- Determinar el tipo de tumor cerebral presente en diferentes imágenes a través de un modelo de red neuronal convolucional. Debido a que hemos conseguido superar la exactitud establecida, podemos dar este objetivo por superado.
- Estudiar las diferencias fisiológicas entre los 3 tipos de tumores. Este objetivo se ha realizado en el apartado de “Estado del arte”
- Utilizar una Red Neuronal Convencional para la predicción del tumor. Se ha realizado un modelo de CNN explicado detalladamente en el apartado de “Metodología”.
- Conseguir una predicción del tipo de tumor con una precisión de mínimo 80%. Hemos conseguido superar el 90% de precisión después de haber mejorado el modelo.
- Investigar la posible optimización del modelo escogido. Debido a que no se ha obtenido el rendimiento deseado en el primer intento, se ha estudiado la manera de mejorarlo en el apartado de “Metodología”
- Investigar la relevancia del estudio y las posibles aplicaciones en la comunidad. Este objetivo se ha cumplimentado en el apartado de “Introducción” y “Estado del arte”

El objetivo que queda pendiente para líneas futuras es el siguiente:

- Desarrollar una interfaz web para implementar el modelo donde se visualicen los resultados de las predicciones. Este objetivo no se ha podido realizar debido a que la formación necesaria para implementar la página web con Flask no ha sido suficiente y debido al tiempo limitado no es posible finalizarlo.

<Lorena Pelegero Alonso>

8.1 Líneas de futuro

Si se realiza la continuación de este proyecto se debería enfocar en aumentar la precisión del modelo, debido, a que un 93% sea una cifra elevada, si estamos hablando de finalidades médicas este porcentaje es considerado insuficiente ya que está en riesgo la vida de personas. Para poder conseguir el 99% habría que poner a prueba el transfer learning ya que debido al limitante número de imágenes con el data augmentation no hemos tenido suficiente. Otro punto para desarrollar en este proyecto sería finalizar una aplicación web para que a un usuario le sea más fácil acceder al uso del modelo.

8.2 Seguimiento de la planificación

En líneas generales se ha seguido una buena planificación menos por el tiempo invertido en mejorar el modelo que ha sido mayor del previsto y esto ha provocado que el planning se haya retrasado a tal punto que no se ha podido invertir el tiempo suficiente a la realización de la aplicación web de Flask.

9 Glosario

IMR: Imagen por resonancia magnética (Magnetic Resonance Imaging)
TP: Verdaderos positivos (True Positive)
FP: Falsos positivos (False Positive)
FN: Falsos negativos (False Negative)
TN: Verdaderos negativos (True Negative)
IA: Inteligencia artificial (Intelligence Artificial)
CNN: red neuronal convolucional (Convolutional neural network)
ANN: red neuronal artificial (Artificial neural network)
RNN: red neuronal recurrente (Recurrent neural network)
ADN: Ácido desoxirribonucleico
SVM: Máquinas de vectores de soporte (Support Vector Machine)
KNN: k vecinos más próximos (k-nearest neighbors)
PCA: análisis de componentes principales (Principal component análisis)
LDA: mínimo ancestro común (Lowest common ancestor)
RMSP: Raíz del error cuadrático medio (Root Mean Squared Propagation)

10 Bibliografía

1. McNeill, K. A. (2016). Epidemiology of Brain Tumors. *Neurologic Clinics*, 34(4), 981– 998.
2. Kuhl, C. K. (2000). MRI of breast tumors. *European Radiology*, 10(1), 46–58.
3. Omondiagbe, D. A., Veeramani, S., & Sidhu, A. S. (2019). Machine Learning Classification Techniques for Breast Cancer Diagnosis. *IOP Conference Series: Materials Science and Engineering*, 495, 012033.
4. Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V., & Fotiadis, D. I. (2015). Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 13, 8–17.
5. Mohammed, M., Mwambi, H., Mboya, I. B., Elbashir, M. K., & Omolo, B. (2021). A stacking ensemble deep learning approach to cancer type classification based on TCGA data. *Scientific Reports*, 11(1).
6. Seetha, J., & Raja, S. S. (2018). Brain Tumor Classification Using Convolutional Neural Networks. *Biomedical and Pharmacology Journal*, 11(3), 1457–1461.D.
7. Cheng, Jun, et al. "Enhanced Performance of Brain Tumor Classification via Tumor Region Augmentation and Partition." *PloS one* 10.10 (2015).
8. Cheng, Jun, et al. "Retrieval of Brain Tumors by Adaptive Spatial Pooling and Fisher Vector Representation." *PloS one* 11.6 (2016). Matlab source codes are available on github
9. Benvenuti, L., Kloss, C., & Pirker, S. (2016). Identification of DEM simulation parameters by Artificial Neural Networks and bulk experiments. *Powder Technology*, 291, 456–465.
10. Clark, D., 2021. *Top 16 Open Source Deep Learning Libraries and Platforms - KDnuggets*. [online] KDnuggets. Available at: <<https://www.kdnuggets.com/2018/04/top-16-open-source-deep-learning-libraries.html>>.
11. 2021. [online] Available at: <<https://www.elperceptron.com/las-15-mejores-librerias-de-python-para-machine-learning-y-deep-learning/>>.
12. Codecademy News. 2021. *Flask vs. Django: Which Framework Should You Choose?*. [online] Available at: <<https://www.codecademy.com/resources/blog/flask-vs-django/>>.
13. BarD Software s.r.o. (2021). GanttProject: free project management tool for windows, macOS and Linux. GanttProject. <<https://www.ganttproject.biz/download/free/>>
14. Nazir, M., Shakil, S. and Khurshid, K., 2021. Role of deep learning in brain tumor detection and classification (2015 to 2020): A review. *Computerized Medical Imaging and Graphics*, 91, p.101940.
15. Aans.org. 2021. *Brain Tumors - Classifications, Symptoms, Diagnosis and Treatments*. [online] Available at: <<https://www.aans.org/en/Patients/Neurosurgical-Conditions-and-Treatments/Brain-Tumors>>.
16. Kopec, M., Błaszczuk, M., Radek, M. and Abramczyk, H., 2021. Raman imaging and statistical methods for analysis various type of human brain tumors and breast cancers. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 262, p.120091.
17. Diagnóstico Journal. 2021. *El modelo Deep Learning clasifica los tumores cerebrales con una sola RM*. [online] Available at:

- <<https://diagnosticojournal.com/el-modelo-de-deep-learning-clasifica-los-tumores-cerebrales-con-una-sola-rm/>>.
18. Sathya, R. and Abraham, A., 2013. Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification. *International Journal of Advanced Research in Artificial Intelligence*, 2(2).
 19. GeeksforGeeks. 2021. *Supervised and Unsupervised learning - GeeksforGeeks*. [online] Available at: <<https://www.geeksforgeeks.org/supervised-unsupervised-learning/>>.
 20. GreatLearning Blog: Free Resources what Matters to shape your Career!. 2021. *Types of Neural Networks and Definition of Neural Network*. [online] Available at: <<https://www.mygreatlearning.com/blog/types-of-neural-networks/>>.
 21. Medium. 2021. *A Quick Guide to Activation Functions In Deep Learning*. [online] Available at: <<https://towardsdatascience.com/a-quick-guide-to-activation-functions-in-deep-learning-4042e7add5b?source=rss-8bf0420ce77b-----2>>.
 22. Analytics Vidhya. 2021. *How Does the Gradient Descent Algorithm Work in Machine Learning?*. [online] Available at: <<https://www.analyticsvidhya.com/blog/2020/10/how-does-the-gradient-descent-algorithm-work-in-machine-learning/>>.
 23. GitHub. 2021. *GitHub - lpelegero/Image-Classification-for-Brain-Cancer: TFM*. [online] Available at: <<https://github.com/lpelegero/Image-Classification-for-Brain-Cancer>>.

Anexos

1. Acceso al código en Github:

<https://github.com/lpelegero/Image-Classification-for-Brain-Cancer>

2. Cálculos de la exactitud, precisión, sensibilidad y especificidad para cada clase:

-Clase 1 (Meniongina):

- **TP(verdaderos positivos)=129**
- **FP(Falsos positivos):15+5=20**
- **FN(Falsos negativos):4+7=11**
- **TN(Verdaderos negativos): 122+3+3+132=260**

$$\text{Exactitud} = \frac{\text{Verdaderos Positivos} + \text{Verdaderos Negativos}}{\text{Total}} = \frac{129+260}{420} = 0.926$$

$$\text{Precisión} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Positivos}} = \frac{129}{129+20} = 0.866$$

$$\text{Sensibilidad} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Negativos}} = \frac{129}{129+11} = 0.921$$

$$\text{Especificidad} = \frac{\text{Verdaderos Negativos}}{\text{Verdaderos Negativos} + \text{Falsos Positivos}} = \frac{260}{260+20} = 0.929$$

-Clase 2 (Glioma):

- **TP(verdaderos positivos)=122**
- **FP(Falsos positivos):4+3=7**
- **FN(Falsos negativos):15+3=18**
- **TN(Verdaderos negativos): 132+129+7+5=273**

$$\text{Exactitud} = \frac{\text{Verdaderos Positivos} + \text{Verdaderos Negativos}}{\text{Total}} = \frac{122+273}{420} = 0.940$$

$$\text{Precisión} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Positivos}} = \frac{122}{122+7} = 0.946$$

$$\text{Sensibilidad} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Negativos}} = \frac{122}{122+18} = 0.871$$

<Lorena Pelegero Alonso>

$$\text{Especificidad} = \frac{\text{Verdaderos Negativos}}{\text{Verdaderos Negativos} + \text{Falsos Positivos}} = \frac{273}{273+7} = 0.975$$

-Clase 3 (Pituitario):

- **TP(verdaderos positivos)=132**
- **FP(Falsos positivos):7+3=10**
- **FN(Falsos negativos):5+3=8**
- **TN(Verdaderos negativos): 122+129+4+15=270**

$$\text{Exactitud} = \frac{\text{Verdaderos Positivos} + \text{Verdaderos Negativos}}{\text{Total}} = \frac{132+270}{420} = 0.957$$

$$\text{Precisión} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Positivos}} = \frac{132}{132+10} = 0.930$$

$$\text{Sensibilidad} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Negativos}} = \frac{132}{132+8} = 0.943$$

$$\text{Especificidad} = \frac{\text{Verdaderos Negativos}}{\text{Verdaderos Negativos} + \text{Falsos Positivos}} = \frac{270}{270+10} = 0.964$$