

Grado en Ciencia de Datos



VNIVERSITAT
DE VALÈNCIA

Trabajo Fin de Grado

Detección de enfermedades usando técnicas de
aprendizaje profundo

Autor: Francesc Domenech Villalba

Tutores:

Joan Vila-Francés, PhD
José Rafael Magdalena Benedicto, PhD

Declaración de autoría:

Yo, Francesc Domenech Villalba, declaro la autoría del Trabajo Fin de Grado titulado “Detección de enfermedades usando técnicas de aprendizaje profundo” y que el citado trabajo no infringe las leyes en vigor sobre propiedad intelectual. El material no original que figura en este trabajo ha sido atribuido a sus legítimos autores.

Valencia, 19 de junio de 2024

Fdo: Francesc Domenech Villalba

Resumen:

El presente Trabajo de Fin de Grado (TFG) se centra en la aplicación de técnicas de aprendizaje profundo para la detección de enfermedades a partir de radiografías de tórax. Más allá de simplemente entrenar modelos, este estudio evalúa diferentes arquitecturas de redes neuronales mediante la técnica de aprendizaje por transferencia, analizando su comportamiento, métricas de evaluación y tiempo de ejecución ante diferentes codificaciones de los datos de entrada. Se entrena dos modelos. Por un lado, la ResNet, preentrenada de forma genérica sobre el conjunto de datos de ImageNet, y por otro lado CXR Foundation, que incluye una etapa adicional de preentrenamiento específico sobre imágenes de radiografías de tórax. Se demuestra la importancia de este paso extra de preentrenamiento usando datos similares al problema a resolver para mejorar el flujo tradicional del aprendizaje por transferencia en un problema tan complejo como es la detección de enfermedades a partir de imágenes de radiografías. Las métricas utilizadas son el área bajo la curva, matriz de confusión, F1-Score, precisión y recall.

Para evaluar su eficacia

Terminología clave: aprendizaje profundo, detección de enfermedades, radiografías de tórax, ResNet, Google Health, aprendizaje por transferencia, clasificación binaria, clasificación multiclas, métricas de evaluación.

Agradecimientos:

Quiero agradecer a todos los que han hecho posible este Trabajo Fin de Grado. En primer lugar, agradecer a mis compañeros de carrera, quienes han sido esenciales en este trayecto; sin ellos esta experiencia en la universidad no habría sido lo mismo. En segundo lugar, agradecer a todos los profesores con los que he coincidido a largo de estos años, por su dedicación en todo momento a nosotros.

Reconocimiento especial a mis tutores, Joan Vila-Francés y Rafael Magdalena Benedicto, por darme la oportunidad de realizar este proyecto y por su orientación decisiva en todo momento. Gracias por su paciencia y por guiarme en cada paso del camino.

ÍNDICE

1	INTRODUCCIÓN.....	1
1.1	MOTIVACIÓN	1
1.2	OBJETIVOS	1
1.3	APRENDIZAJE AUTOMÁTICO	2
1.4	INTRODUCCIÓN A LAS REDES NEURONALES	3
1.4.1	<i>MULTILAYER PERCEPTRON (MLP)</i>	4
1.5	REDES NEURONALES CONVOLUCIONALES	6
1.5.1	<i>PASO 1: EXTRACCIÓN DE CARACTERÍSTICAS</i>	6
1.5.2	<i>PASO 2: CLASIFICACIÓN</i>	7
1.6	VISIÓN POR COMPUTADORA	8
2	MARCO TEÓRICO.....	9
2.1	TIPOS DE PROBLEMAS DE CLASIFICACIÓN	9
2.1.1	<i>One-vs-All (OvA)</i>	10
2.1.2	<i>One-vs-One (OvO)</i>	10
2.2	TRANSFER LEARNING	10
2.2.1	<i>RESNET</i>	11
2.2.2	<i>CXR FOUNDATION</i>	13
3	CONJUNTO DE DATOS UTILIZADO.....	15
3.1	ADAPTACIÓN DEL CONJUNTO DE DATOS AL TRABAJO.....	15
4	IMPLEMENTACIÓN.....	19
4.1	ARQUITECTURAS Y CONFIGURACIÓN DE HIPERPARÁMETROS.....	19
4.1.1	<i>RESNET</i>	19
4.1.2	<i>CXR FOUNDATION</i>	20
4.2	ENFOQUES	22
4.2.1	<i>ENFOQUE 1: NO ENFERMO O ENFERMO</i>	22
4.2.2	<i>ENFOQUE 2: CLASIFICACIÓN MULTICLASE</i>	22
4.2.3	<i>ENFOQUE 3: CLASIFICACIÓN BINARIA TRIPLE</i>	22
4.2.4	<i>MÉTRICAS</i>	23
4.2.5	<i>ELECCIÓN ÓPTIMA DE HIPERPARÁMETROS</i>	26
5	RESULTADOS	27
5.1	ENFOQUE 1: NO ENFERMO O ENFERMO.....	27
5.1.1	<i>CXR FOUNDATION</i>	27
5.1.2	<i>ResNet</i>	29
5.1.3	<i>RESUMEN DE CLASIFICACIÓN DEL ENFOQUE 1</i>	31
5.2	ENFOQUE 2: CLASIFICACIÓN MULTICLASE	32
5.2.1	<i>CXR FOUNDATION</i>	32
5.2.2	<i>ResNet</i>	34
5.2.3	<i>RESUMEN DE CLASIFICACIÓN DEL ENFOQUE 2</i>	36
5.3	ENFOQUE 3	37
5.3.1	<i>CXR FOUNDATION</i>	37
5.3.2	<i>ResNet</i>	42
5.3.3	<i>RESUMEN DE CLASIFICACIÓN DEL ENFOQUE 3</i>	47
5.4	TIEMPOS DE EJECUCIÓN	47

6	DISCUSIÓN.....	48
6.1	ANÁLISIS DEL ENTRENAMIENTO	48
7	CONCLUSIONES.....	52
7.1	TRABAJO FUTURO.....	52
8	BIBLIOGRAFÍA.....	53
9	ANEXO	55

ÍNDICE DE FIGURAS

Figura 1.1: Diagrama de una neurona biológica [8].....	3
Figura 1.2: Esquema de una red neuronal artificial [9]	4
Figura 1.3: Esquema de un Multilayer Perceptron (MLP)	5
Figura 1.4: Esquema general de una red convolucional [11]	6
Figura 1.5: Proceso de convolución [11].....	7
Figura 1.6: Ejemplo de <i>Max Pooling</i> [11]	7
Figura 2.1: Gráfico de la mejora del rendimiento usando aprendizaje por transferencia [4]	10
Figura 2.2: Representación de un bloque residual en una red convolucional	11
Figura 2.3 : Comparación entre la arquitectura VGG19 y ResNet50 con y sin bloques residuales. [19].....	12
Figura 2.4: Clasificación usando el modelo CXR Foundation según relevancia y volumen [1].....	14
Figura 3.1: Imagen de una radiografía clasificada como normal.	16
Figura 3.2: Radiografía mostrando signos de EPOC.....	16
Figura 3.3: Radiografía mostrando características de neumonía.....	17
Figura 3.4: Radiografía indicativa de falla cardiaca.....	17
Figura 3.5: Gráfico de la frecuencia de aparición de cada clase	18
Figura 4.1: Configuración del MLP aplicado al modelo ResNet	20
Figura 4.2: Configuración del MLP aplicado al modelo CXR Foundation.....	21
Figura 4.3: Imagen de una curva ROC	24
Figura 4.4: Esquema de una matriz de confusión.....	24
Figura 5.1: Monitorización de la función de pérdidas para el modelo CXR en el enfoque 1	27
Figura 5.2: Monitorización de la curva ROC para el modelo CXR en el enfoque 1.....	28
Figura 5.3: Curva ROC para el modelo CXR en el enfoque 1	28
Figura 5.4: Matriz de confusión para el modelo CXR Foundation en el enfoque 1.....	29
Figura 5.5: Monitorización de la función de pérdidas para el modelo ResNet en el enfoque 1	29
Figura 5.6: Monitorización de la curva ROC para el modelo ResNet en el enfoque 1 ..	30
Figura 5.7: Curva ROC para el modelo ResNet en el enfoque 1.....	30
Figura 5.8: Matriz de confusión para el modelo ResNet en el enfoque 1	31
Figura 5.9: Monitorización de la función de pérdidas para el modelo CXR en el enfoque 2	32
Figura 5.10: Monitorización de la curva ROC para el modelo CXR en el enfoque 2....	32
Figura 5.11: Curva ROC para el modelo CXR en el enfoque 2 (multiclas)	33
Figura 5.12: Matriz de confusión para el modelo CXR Foundation en el enfoque 2 (multiclas)	33
Figura 5.13: Monitorización de la función de pérdidas para el modelo ResNet en el enfoque 2	34
Figura 5.14: Monitorización de la curva ROC para el modelo ResNet en el enfoque 2	34

Figura 5.15: Curva ROC para el modelo ResNet en el enfoque 2.....	35
Figura 5.16: Matriz de confusión para el modelo ResNet en el enfoque 2 (multiclas)	35
Figura 5.17: Monitorización de la función de pérdidas para el modelo CXR en el enfoque 3 (clase COPD o diferente).....	37
Figura 5.18: Monitorización de la curva ROC para el modelo CXR en el enfoque 3 (clase COPD o diferente)	37
Figura 5.19 : Monitorización de la función de pérdidas para el modelo CXR en el enfoque 3 (clase INSUFICIENCIA CARDIACA o diferente).....	38
Figura 5.20: Monitorización de la curva ROC para el modelo CXR en el enfoque 3 (clase INSUFICIENCIA CARDIACA o diferente)	38
Figura 5.21: Monitorización de la función de pérdidas para el modelo CXR en el enfoque 3 (clase NEUMONÍA o diferente).....	39
Figura 5.22: Monitorización de la curva ROC para el modelo CXR en el enfoque 3 (clase NEUMONÍA o diferente)	39
Figura 5.23: Curva ROC de cada clase para el modelo CXR en el enfoque 3	40
Figura 5.24: Matriz de confusión del ensemble en el enfoque 3 (multiclas) usando predicciones de CXR Foundation.....	40
Figura 5.25: Matriz de confusión para el modelo CXR Foundation en el enfoque 3 (clase signos EPOC o diferente)	41
Figura 5.26: Matriz de confusión para el modelo CXR Foundation en el enfoque 3 (clase INSUFICIENCIA CARDIACA o diferente)	41
Figura 5.27: Matriz de confusión para el modelo CXR Foundation en el enfoque 3 (clase NEUMONÍA o diferente)	41
Figura 6.1: Gráfico que muestra la mejora en el rendimiento en el diagnóstico conforme aumenta el número de imágenes [1]	51

Capítulo 1

INTRODUCCIÓN

1.1 MOTIVACIÓN

La radiografía de tórax es una de las pruebas más frecuentes y accesibles para detectar, diagnosticar y manejar enfermedades pulmonares. Anualmente se generan entorno a 837 millones de radiografías [1] acompañados de sus correspondientes informes en hospitales de todo el mundo. En el paradigma actual, la inteligencia artificial, y más en concreto las redes neuronales, han surgido como una herramienta de ayuda, aunque reconocer y localizar los patrones de enfermedades para estos sistemas automatizados sigue siendo una tarea realmente compleja.

Diagnosticar de forma precisa puede suponer la diferencia entre una intervención exitosa y el agravamiento de la situación del paciente. El uso de estos métodos puede suponer una herramienta de ayuda realmente útil, pero nunca sustituirá el trabajo de un humano.

1.2 OBJETIVOS

Se ha invertido gran esfuerzo en desarrollar técnicas de aprendizaje máquina capaces detectar anomalías en las radiografías de tórax [2] [3]. Desarrollar estas técnicas para imágenes en rayos X requiere gran cantidad de datos y recursos computacionales.

El aprendizaje por transferencia es una técnica que consiste en reutilizar un modelo entrenado en una tarea diferente pero relacionada, para poder reducir la necesidad de grandes conjuntos de datos y la carga computacional [4].

El objetivo principal de este trabajo es evaluar diferentes arquitecturas usando la técnica de aprendizaje por transferencia, y ver si aquellas que incluyen una etapa extra de preentrenamiento específico sobre el problema a resolver mejoran el resultado final. Este trabajo va más allá de simplemente entrenar modelos; busca realizar un análisis de su rendimiento y comportamiento frente a diferentes codificaciones de los datos de entrada. Para ello, se emplearán métricas como el área bajo la curva, matrices de confusión, recall y precisión. Además, se examinarán los tiempos de ejecución.

1.3 APRENDIZAJE AUTOMÁTICO

El Aprendizaje Automático (*Machine Learning*, en inglés) ha emergido como una de las disciplinas más innovadoras y transformadoras dentro del campo de la inteligencia artificial y las ciencias de la computación. Esta tecnología se basa en la capacidad de los sistemas informáticos para aprender y mejorar a partir de datos. El avance del aprendizaje automático ha sido impulsado por el crecimiento exponencial de los datos disponibles, debido a la mejora de los sistemas de almacenamiento, el aumento de la velocidad de procesamiento de los datos, las mejoras en la confiabilidad de la transmisión de datos y la aparición de sistemas de administración de bases de datos más potentes [5].

El volumen total de datos digitales en 2020 se estimó en 44 zettabytes y se espera que aumente a 175 zettabytes en 2025. Esta gran cantidad de información presenta oportunidades para el análisis y la obtención de conocimientos, aunque también plantea desafíos considerables, como el almacenamiento, procesamiento y manejo efectivo de conjuntos de datos tan grandes. En el ámbito del aprendizaje automático, se pueden resolver varios tipos de problemas:

- *Clasificación*: Asignar elementos a categorías predefinidas, como clasificar correos electrónicos como spam o no spam.
- *Regresión*: Predecir valores numéricos continuos, como las ventas futuras de un producto.
- *Agrupamiento (Clustering)*: Agrupar elementos con características similares en conjuntos, como segmentar clientes en grupos según sus hábitos de compra.
- *Detección de anomalías*: Identificar datos atípicos o inusuales, como detectar fraudes en transacciones financieras.

El objetivo es examinar datos para formular hipótesis que posteriormente serán validadas con más datos. El primer paso de este proceso implica la recopilación de los conjuntos de datos necesarios. A continuación, se analizan las propiedades estadísticas de estos datos para comprender su comportamiento y características. Seguidamente, se selecciona y aplica la técnica adecuada para el análisis de los datos. Esto resulta en la creación de un modelo. El último paso consiste en interpretar los resultados obtenidos, evaluando el modelo para determinar su eficacia y decidir si debe ser aprobado o descartado.

Las técnicas usadas en el aprendizaje automático, que permiten hacer predicciones o tomar decisiones informadas basadas en grandes cantidades de datos están basadas principalmente en la estadística. Entre ellas, cabe destacar por un lado las redes neuronales artificiales (basadas en el cerebro humano), y por otro lado algoritmos como son la regresión lineal, árboles de decisión, máquinas vectoriales de soporte, etc.

1.4 INTRODUCCIÓN A LAS REDES NEURONALES

Las Redes Neuronales Artificiales (RNA) están inspiradas en el comportamiento biológico de las neuronas y en la estructura del cerebro [6], buscando emular ciertas características humanas, como la capacidad de memorizar y asociar hechos, siendo de esta forma sistemas inteligentes que funcionan de manera distinta a los ordenadores tradicionales. Problemas que parecen sencillos, como la identificación de objetos en imágenes, resultan muy complejos y requieren mucho tiempo de procesamiento para los ordenadores actuales, mientras que el cerebro humano realiza estas tareas en fracciones de segundo gracias al número de neuronas y a la alta interconexión que hay entre estas.

Una neurona, a diferencia de otras células vivas, tiene la capacidad de comunicación. Es la unidad básica del cerebro, y está compuesta por varios elementos [7]:

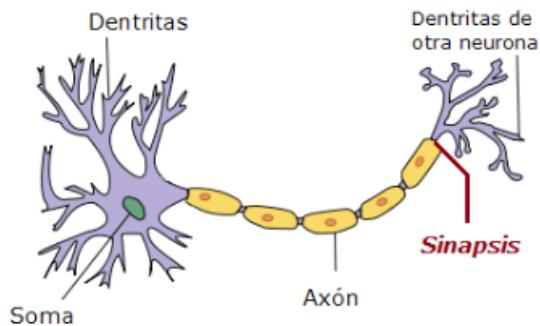


Figura 1.1: Diagrama de una neurona biológica [8]

- *Dendritas*: extensiones ramificadas que reciben señales de otras neuronas y las transmiten al soma. Actúan como las antenas de la neurona.
- *Soma*: contiene el núcleo de la célula y es el centro metabólico de la neurona.
- *Axón*: prolongación que transmite señales eléctricas desde el soma hacia otras neuronas, músculos o glándulas.
- *Sinapsis*: punto de comunicación entre dos neuronas. La señal se transmite desde el axón de una neurona a las dendritas de otra mediante la liberación de neurotransmisores.

La principal función de las neuronas es transmitir los impulsos nerviosos; estos son recibidos a través de las dendritas, llegando al axón, donde pasan a una nueva neurona a través de la conexión sináptica.

Una neurona usa funciones matemáticas para emular su funcionamiento. En todo modelo de neurona artificial hay cinco elementos básicos:

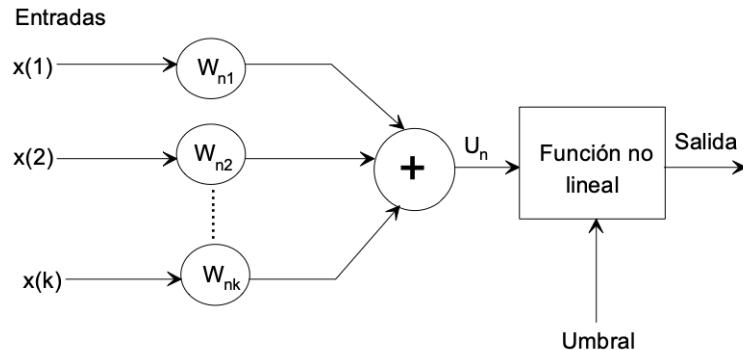


Figura 1.2: Esquema de una red neuronal artificial [9]

- *Datos de entrada*
- *Pesos sinápticos*: son los parámetros del sistema. El aprendizaje consiste en hallar el valor óptimo de estos pesos. Es una situación parecida a la que se da en los axones de las neuronas biológicas [10]
- *Sumador*: se encarga de sumar todas las entradas multiplicadas por sus respectivos pesos.
- *Función de activación no lineal*: se aplica a la salida del sumador.
- *Umbral exterior*: valor que determina cuándo la neurona se activa.

1.4.1 MULTILAYER PERCEPTRON (MLP)

El Perceptrón Multicapa (*MLP, Multilayer Perceptron*) se desarrolló para superar las limitaciones de las redes neuronales anteriores, que solo podían clasificar datos en regiones linealmente separables [9]. La clave fue añadir capas de neuronas entre las capas de entrada y salida, denominadas capas ocultas. No obstante, esto planteó el reto de determinar un "error" en el proceso de aprendizaje de estas neuronas ocultas.

Es por ello que, un *Multilayer Perceptron* consiste en varias capas de neuronas conectadas entre sí, formando una estructura en cascada. Cada neurona recibe entradas, realiza una suma ponderada de estas entradas utilizando pesos específicos, y luego aplica una función de activación no lineal para generar una salida.

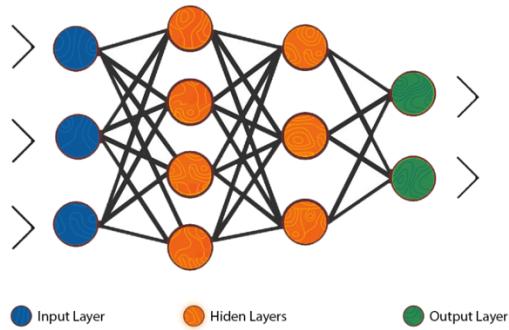


Figura 1.3: Esquema de un Multilayer Perceptron (MLP)

De esta forma se obtiene una predicción

$$\hat{y} = f \left(\sum_{j=1}^n x_j w_j + w_0 \right) \quad (1.1)$$

donde $f(*)$ es la función de activación y w_0 un parámetro llamado *bias*. Durante el entrenamiento, los parámetros w_i se ajustan utilizando el algoritmo *backpropagation*. El MLP se forma al combinar múltiples de estas neuronas, organizadas en capas interconectadas, como se ilustra en la figura 1.3. Cada nodo representa una neurona, y la salida de una capa se convierte en la entrada de la siguiente para generar el vector de predicción final de salida \hat{y} . El proceso de *backpropagation* consta de tres etapas: el paso hacia adelante, donde se hace una inferencia inicial con el vector de entrada x ; el cálculo del error, evaluando qué tan cerca está \hat{y} del objetivo y utilizando una función de coste L ; y la retropropagación del error, que ajusta los pesos de la red para minimizar este error durante el entrenamiento.

En la fase de retropropagación del error, se utiliza la regla de la cadena para calcular cómo cada peso w_i contribuye al error total. Posteriormente, los pesos se ajustan en la dirección contraria al gradiente de la función de coste, lo que reduce el error. Esta actualización de pesos se describe mediante la fórmula:

$$w_i(t+1) = w_i(t) - \alpha \frac{\partial L}{\partial w_i} \quad (1.2)$$

donde t es la época de entrenamiento y α la tasa de aprendizaje, un parámetro que determina la velocidad de aprendizaje. Estos pasos se repiten hasta que la función de coste L converja, indicando que el proceso de entrenamiento ha finalizado.

1.5 REDES NEURONALES CONVOLUCIONALES

Las redes neuronales convolucionales (*CNN*, por sus siglas en inglés) son un tipo de redes neuronales frecuentemente empleadas para procesar imágenes, debido a su habilidad para identificar patrones complejos en grandes volúmenes de datos visuales.

Las CNN utilizan filtros (conocidos como *kernels*) que realizan convoluciones sobre las entradas visuales, permitiendo que cada filtro se especialice en detectar un tipo específico de características, tales como bordes, texturas o patrones. Surgen como una alternativa al perceptrón multicapa para procesar datos visuales de una forma más eficiente.

A continuación, se muestra el esquema general de las redes neuronales convolucionales, que será descrito posteriormente.

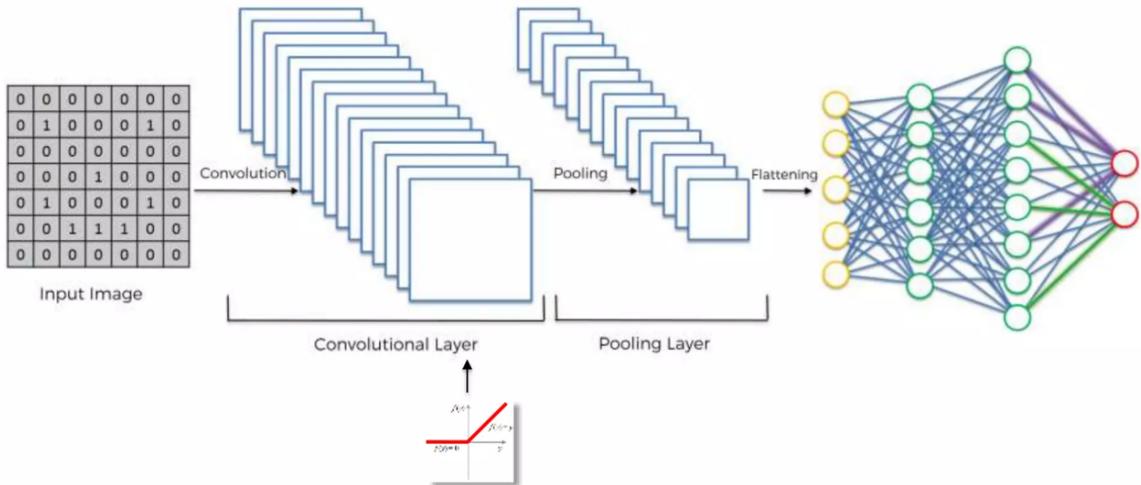


Figura 1.4: Esquema general de una red convolucional [11]

1.5.1 PASO 1: EXTRACCIÓN DE CARACTERÍSTICAS

1.5.1.1 CONVOLUCIÓN

En el contexto de procesamiento de imágenes, que utiliza elementos discretos, definimos la operación de convolución entre una imagen f de tamaño $M \times N$ y un *kernel* de tamaño $P \times Q$ como:

$$(f * w)[i, j] = \sum_{k=-\frac{P}{2}}^{\frac{P}{2}} \sum_{l=-\frac{Q}{2}}^{\frac{Q}{2}} f[i - k, j - l]w[k, l] \quad (2.2)$$

Esta operación calcula la correlación entre la imagen de entrada y el *kernel*, y sirve para ver el parecido entre ambos. El resultado se conoce como mapa de características. Esta operación se realiza múltiples veces con diferentes *kernels*, cuyos valores (pesos) se aprenden durante el proceso de entrenamiento de la red.

Posteriormente, al mapa de características resultante se le suma un sesgo y se le aplica una función de activación.

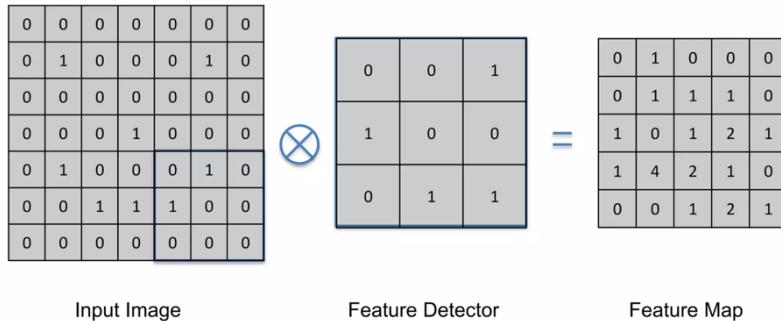


Figura 1.5: Proceso de convolución [11]

1.5.1.2 POOLING

Esta operación es opcional, consiste en aplicar una operación sobre una parte de la matriz, siendo el máximo o la media las más comunes. Se hace entre etapas convolucionales para reducir el tamaño de la imagen resultante, reduciendo el coste computacional.

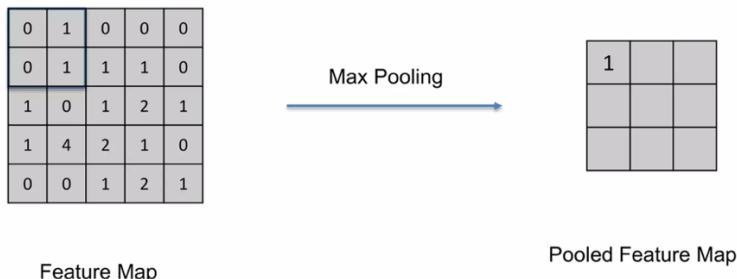


Figura 1.6: Ejemplo de *Max Pooling* [11]

1.5.2 PASO 2: CLASIFICACIÓN

Después de la extracción de características, se utiliza un conjunto de capas densas para realizar la predicción final. En primer lugar, se estira la matriz resultante del paso previo en un vector unidimensional, que será la entrada a la red neuronal. Finalmente, para realizar la clasificación final se propone una red neuronal totalmente conectada (perceptrón multicapa) encargada de realizar la predicción.

1.6 VISIÓN POR COMPUTADORA

La visión por computadora es un campo multidisciplinario dentro de la inteligencia artificial que se enfoca en el desarrollo de algoritmos y técnicas que permiten a las máquinas interpretar y comprender el mundo visual. Este campo se ha vuelto fundamental en la era digital actual, con aplicaciones que abarcan desde la detección de objetos y el reconocimiento facial hasta la conducción autónoma y el análisis de imágenes médicas.

La evolución de la visión por computadora ha sido impulsada principalmente por los avances en el aprendizaje automático y las redes neuronales profundas. En particular, los modelos de aprendizaje profundo, como las redes convolucionales (*CNN*, por sus siglas en inglés), han demostrado un gran desempeño en tareas complejas incluso superando las habilidades humanas en algunos casos.

Capítulo 2

MARCO TEÓRICO

En este capítulo se presentan los conceptos teóricos necesarios para entender el proyecto. Comenzando con los distintos tipos de clasificación que existen en el aprendizaje automático, destacando los enfoques que se pueden adoptar según las necesidades de la tarea a resolver. Posteriormente se detallan las arquitecturas utilizadas en la resolución del problema planteado: CXR Foundation y ResNet.

2.1 TIPOS DE PROBLEMAS DE CLASIFICACIÓN

En el campo del aprendizaje máquina, la clasificación es una tarea supervisada que implica asignar una etiqueta o categoría a cada dato de entrada, basada en sus características. Hay varios tipos de clasificación, en función del tipo de problema que se quiera resolver:

- **Clasificación Binaria:** es la forma más simple de clasificación donde las instancias se etiquetan en una de dos categorías.
 - **Ejemplo:** Clasificar pacientes como "enfermos" o "sanos".
- **Clasificación Multiclasa:** en este caso, las instancias pueden ser clasificadas en una de tres o más categorías.
 - **Ejemplo:** Clasificar imágenes de dígitos escritos a mano en los números del 0 al 9.
- **Clasificación Multietiqueta:** Cada instancia puede pertenecer a múltiples categorías simultáneamente.
 - **Ejemplo:** Clasificación de imágenes en redes sociales, donde una sola foto puede tener varias etiquetas como “paisaje”, “naturaleza” o “animales”.

Además, es posible adaptar problemas multiclasa o multietiqueta como un conjunto de problemas binarios, dependiendo de las necesidades específicas de la tarea a resolver. Esto implica descomponer un problema complejo, donde se requiere clasificar entre varias categorías o etiquetas, en múltiples problemas de decisión binaria.

2.1.1 One-vs-All (OvA)

Se entrena un clasificador binario para cada clase, donde cada clasificador aprende a distinguir una clase particular de todas las demás clases combinadas. Es un enfoque más simple, sin embargo, dificulta la interpretabilidad del modelo y puede ser menos preciso si hay clases similares entre sí.

2.1.2 One-vs-One (OvO)

Se entrena un clasificador binario para cada par de clases posibles. Esto significa que para N clases, se entrenan $\frac{N \times (N-1)}{2}$ clasificadores binarios. Suele ser más preciso ya que cada clasificador se enfoca en distinguir solo entre dos clases y mejora la interpretabilidad del modelo, sin embargo, es más costoso computacionalmente.

2.2 TRANSFER LEARNING

El aprendizaje por transferencia (*transfer learning*, en inglés) es un método en el aprendizaje automático que consiste en aprovechar una arquitectura utilizada en un problema similar, usando como punto de partida los pesos obtenidos en dicha tarea. Su objetivo es mejorar el rendimiento inicial en la tarea objetivo utilizando el conocimiento transferido a partir de una tarea similar acelerando el proceso de aprendizaje, alcanzando un rendimiento final superior en la nueva tarea [4].

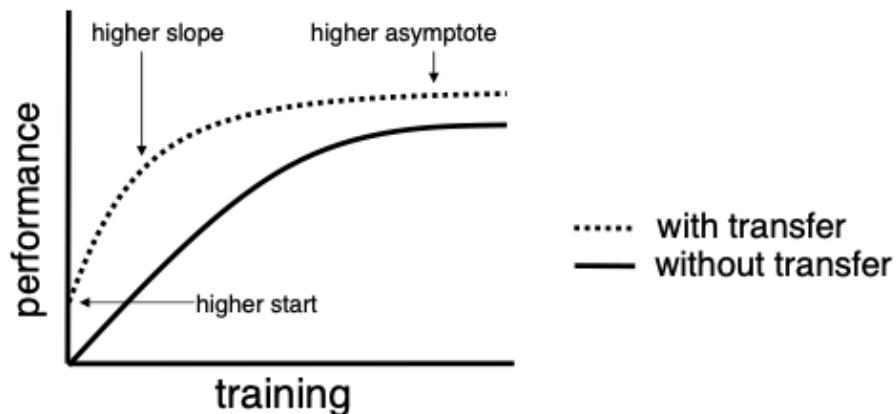


Figura 2.1: Gráfico de la mejora del rendimiento usando aprendizaje por transferencia [4]

Este enfoque cada vez está ganando mayor terreno, y será el nuevo impulsor del éxito en el campo del aprendizaje automático [12]. Sin embargo, el principal problema se da cuando las tareas no están altamente relacionadas y la transferencia de conocimientos hace que pueda disminuir el rendimiento, provocando comportamientos no deseados o difíciles de interpretar, lo que se conoce como transferencia negativa.

2.2.1 RESNET

En 2009 se creó la base de datos ImageNet [13] por la Universidad de Stanford. Fue diseñada para su uso en la investigación de software y reconocimiento de imágenes. Su objetivo era etiquetar y categorizar cerca de 22.000 imágenes. A partir de este conjunto de datos original, se creó una versión reducida en la que se agruparon ciertas categorías, resultando en un total de 1000. Este conjunto de entrenamiento contenía 1.2 millones de imágenes etiquetadas, mientras que el conjunto de prueba constaba de 50,000 imágenes no etiquetadas. Este conjunto de datos se usó en el desafío conocido como ImageNet Large Scale Visual Recognition Challenge (ILSVRC), que tuvo lugar entre los años 2012 y 2017, donde diferentes grupos de investigadores propusieron sus arquitecturas, con diferentes enfoques, entre las que cabe destacar VGGNet entre otras.

VGGNet [14] es una arquitectura propuesta por la Universidad de Oxford en 2014. Fue una de las primeras redes neuronales profundas propuestas, destacando por su simplicidad y eficacia. La arquitectura utiliza pequeños filtros de tamaño 3×3 y capas de *max pooling* para reducir las dimensiones espaciales, permitiendo un diseño más manejable y eficiente en términos de cálculo.

En 2015 un equipo de investigadores de *Microsoft Research* propuso la arquitectura ResNet [15], que abordaba desafíos encontrados en estas redes neuronales profundas. En las arquitecturas mencionadas anteriormente se había observado que aumentar la profundidad de la red mejoraba el rendimiento, pero también introducía problemas como el sobreajuste (*overfitting*) y el desvanecimiento de gradiente (*vanishing gradients*) [16]. Para combatir el sobreajuste, se aplicaron técnicas de regularización, como el *dropout* [17] o regularizaciones de tipo L₁ o L₂. Para abordar el problema del desvanecimiento del gradiente, introdujeron conexiones residuales, que añaden conexiones directas que saltan capas, evitando que al actualizar los pesos por el algoritmo de retropropagación en arquitecturas con muchas capas, los valores tiendan a infinito. Fue una solución innovadora que permitió entrenar redes más profundas de manera efectiva.

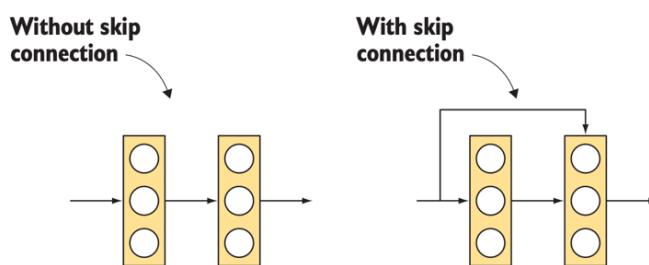


Figura 2.2: Representación de un bloque residual en una red convolucional¹

En los bloques residuales se omite el uso de capas de *pooling*, pero se incorpora normalización por lotes (*batch normalization* [18]) para controlar la explosión de los gradientes. También se introducen capas de convolución 1x1 (*bottleneck layers*) para reducir la complejidad computacional de las operaciones realizadas. Se propusieron

¹ FUENTE: https://www.researchgate.net/figure/Illustration-of-the-skip-connection-gated-module-SGate-The-SGate-gathers-features-of_fig4_361454502

arquitecturas con diferente número de capas según la complejidad del problema a resolver, siguiendo esta metodología. Entre ellas, destaca la ResNet50, que fue la utilizada para este proyecto. En la siguiente figura, se muestra un esquema comparativo entre la arquitectura VGG19 y ResNet50 con y sin bloques residuales.

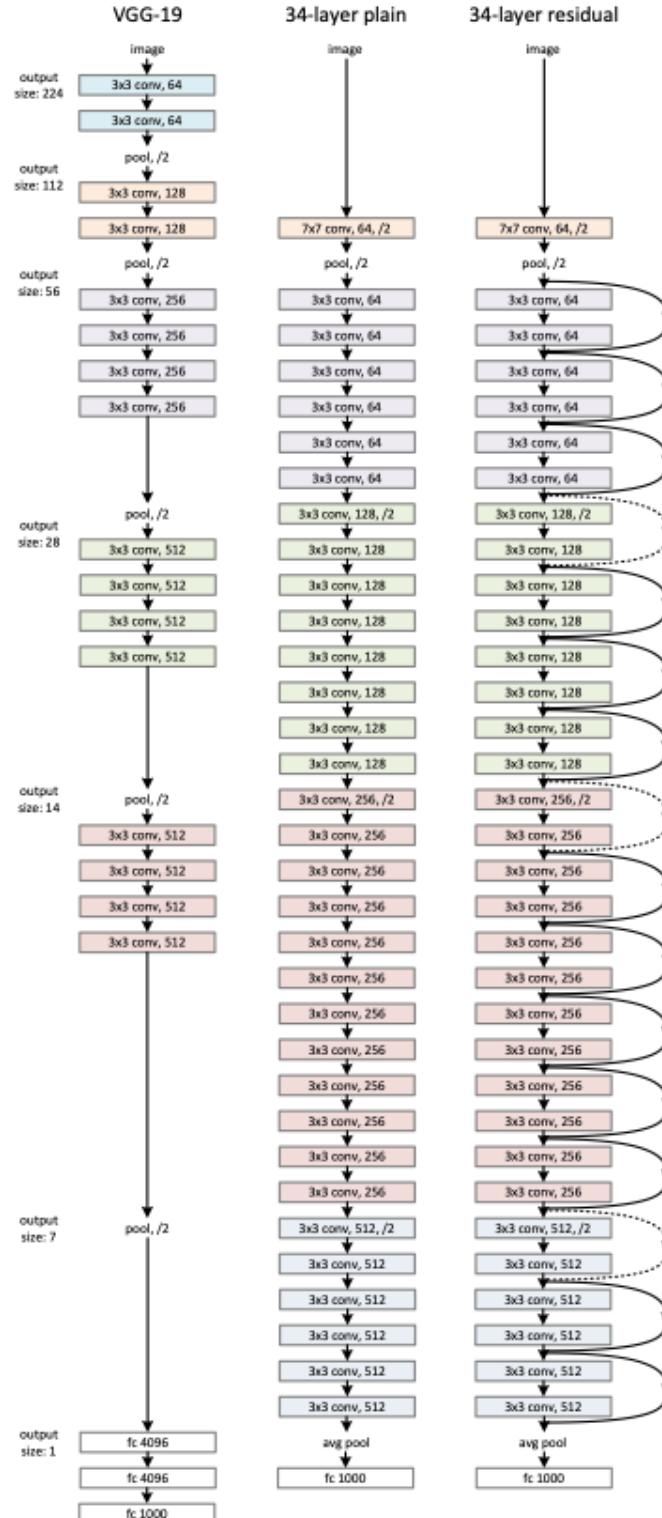


Figura 2.3 : Comparación entre la arquitectura VGG19 y ResNet50 con y sin bloques residuales. [19]

2.2.2 CXR FOUNDATION

CXR FOUNDATION [1] es una herramienta proporcionada por el equipo de Google Health, que sirve para la clasificación de enfermedades en imágenes de radiografías.

Surge con el objetivo de reducir el tamaño del conjunto de datos para modelos de aprendizaje profundo haciendo uso de un enfoque avanzado conocido como aprendizaje supervisado contrastivo [SupCon] [20], ya que el principal reto para resolver con éxito este tipo de problemas tan específicos, es que se necesitan conjuntos de datos muy extensos.

Como se ha comentado en el apartado anterior, el aprendizaje por transferencia es un enfoque que reutiliza un modelo preentrenado en una tarea diferente pero relacionada. Un flujo común en el aprendizaje por transferencia sería, en primer lugar, entrenar un modelo de aprendizaje profundo en una tarea genérica (usando grandes conjuntos de datos no médicos) y luego hacer un ajuste fino en una tarea médica específica. Este enfoque sería más efectivo cuando la tarea fuente y la tarea objetivo son similares (ambas médicas), pero requeriría decenas de miles de imágenes médicas etiquetadas. Por ello surge el aprendizaje supervisado contrastivo, que consiste en 3 pasos: preentrenamiento de imágenes genéricas (usando los pesos del conjunto de datos de ImageNet) preentrenamiento específico de radiografías de tórax y entrenamiento específico de la tarea a resolver. Donde el primer paso usa grandes conjuntos de datos de imágenes no médicas para el preentrenamiento y el segundo paso usa conjuntos de datos de radiografías de tórax con etiquetas ruidosas obtenidas a partir de los informes de radiología.

Además, cabe destacar que el preentrenamiento de la red de radiografía de tórax utiliza etiquetas ruidosas derivadas de informes de radiología, mientras que el entrenamiento y las evaluaciones específicas de la tarea médica utilizan etiquetas limpias de revisiones de imágenes por radiólogos, pruebas moleculares o resultados clínicos. Esta red de radiografías de tórax convierte las imágenes en vectores numéricos, conocidos como *embeddings*, que contienen toda la información relevante de la imagen.

A continuación se muestra un esquema comparativo del proceso que se acaba de describir. El primer caso hace referencia al clásico flujo de aprendizaje por transferencia, mientras que el segundo caso hace referencia al aprendizaje supervisado contrastivo, donde, posteriormente, se plantean 3 enfoques de clasificación (lineal, no lineal y ajuste fino de hiperparámetros de toda la red) en función de la cantidad de datos disponible y su relevancia.

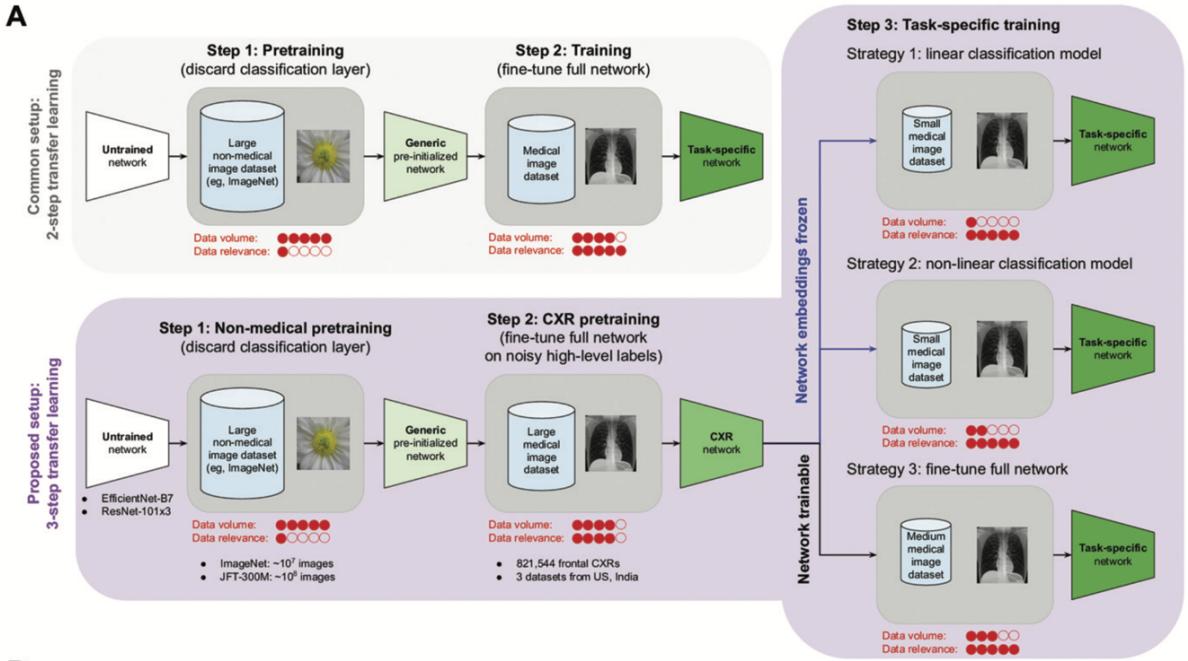


Figura 2.4: Clasificación usando el modelo CXR Foundation según relevancia y volumen [1]

Acerca de la arquitectura de la red preentrenada para imágenes de radiografías, el equipo de Google no ha publicado nada debido a la protección de datos de las personas cuyas radiografías se han usado para este experimento. El único dato conocido es que el número de parámetros de la red es entorno a 25 veces superior al de las conocidas redes convolucionales mencionadas anteriormente y que se usan mecanismos convolucionales. Además, se usan imágenes de varios hospitales en el proceso de entrenamiento, para evitar posibles sesgos durante la toma de radiografías.

Capítulo 3

CONJUNTO DE DATOS UTILIZADO

PadChest [21] es un conjunto de datos etiquetados de radiografías de tórax de alta resolución para la exploración de imágenes médicas junto con sus correspondientes informes.

Incluye más de 160 000 imágenes obtenidas de 67 000 pacientes interpretadas por radiólogos del Hospital de San Juan (Alicante) desde 2009 hasta 2017. En los informes, hay hasta 174 hallazgos radiográficos y 19 diagnósticos. Un hallazgo radiográfico es cualquier descripción objetiva que se detecta en una imagen radiográfica, sin interpretar su significado clínico, mientras que un diagnóstico es la identificación de una enfermedad basada en un hallazgo radiográfico haciendo una interpretación de su significado clínico.

Las imágenes están tomadas con diferentes vistas: anteroposterior, posteroanterior, lateral y oblicua. En la mayoría de casos no está la etiqueta disponible. Lo ideal sería usar aquellas tomadas con vista anteroposterior o posteroanterior, sin embargo, se evaluará el comportamiento en diferentes casos.

Del total de los informes, el 27% fue anotado manualmente por radiólogos especializados y el resto fue etiquetado a partir del informe radiológico mediante técnicas de procesado del lenguaje.

En la actualidad, es una de las mayores bases de datos accesibles de radiografías de tórax.

3.1 ADAPTACIÓN DEL CONJUNTO DE DATOS AL TRABAJO

El conjunto de datos original ocupa alrededor de 1TB, y está repartido en 55 carpetas. Por razones obvias de falta de recursos computacionales, el problema se va a adaptar. A continuación, se describe el proceso para la selección de imágenes y adaptación de objetivos.

El objetivo de este proyecto es clasificar los 4 diagnósticos más frecuentes, que en este caso son: normal, signos de EPOC, falla cardíaca y neumonía. A continuación, se muestra una descripción para comprenderlos mejor.

- NORMAL: no hay ningún diagnóstico

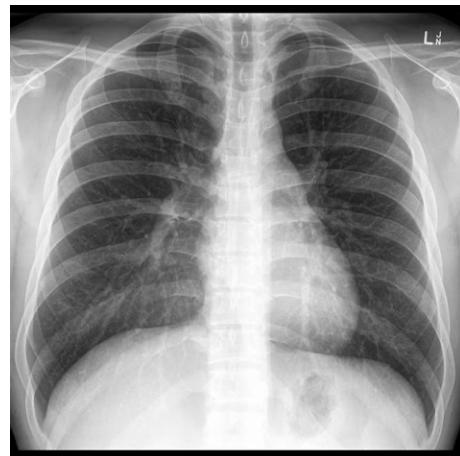


Figura 3.1: Imagen de una radiografía clasificada como normal.²

- SIGNOS DE EPOC (ENFERMEDAD PULMONAR OBSTRUCTIVA CRÓNICA) (*COPD SIGNS*, en inglés): provocan una limitación persistente del flujo de aire debido a la obstrucción de las vías respiratorias. Los signos radiográficos que muestran la obstrucción crónica de los pulmones son el aplanamiento del diafragma, que se puede ver en la parte inferior de la imagen, y un aumento del espacio intercostal, que se puede ver en la parte derecha de la imagen.

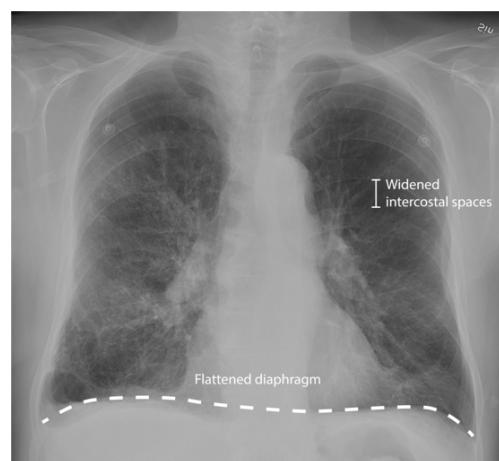


Figura 3.2: Radiografía mostrando signos de EPOC.³

² FUENTE: <http://radiopaedia.org>

³ FUENTE: <http://radiopaedia.org>

- NEUMONÍA: es una infección que inflama los sacos aéreos de uno o ambos pulmones, que pueden llenarse de líquido o pus. Los signos radiográficos que muestran la neumonía son las opacidades, como se puede apreciar en la imagen.

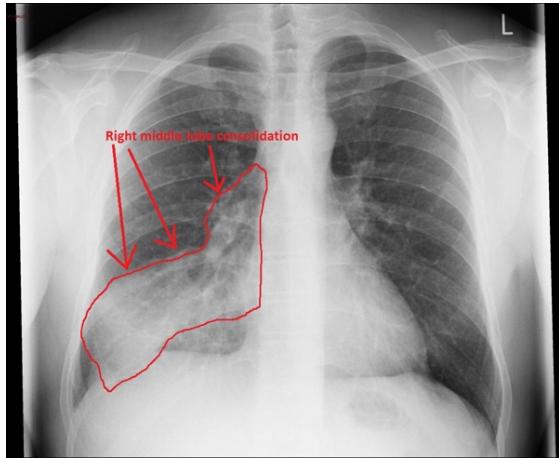


Figura 3.3: Radiografía mostrando características de neumonía.⁴

- FALLA CARDIACA (HEART INSUFFICIENCY): es una situación en la que el corazón no puede bombear sangre de manera eficiente. Los signos radiográficos que muestran la falla cardiaca son las *líneas de Kerley* (líneas finas en los pulmones debido a la congestión linfática), marcadas en la imagen con la flecha de la parte izquierda; y un agrandamiento del corazón, marcado en la imagen con la flecha de la parte derecha.

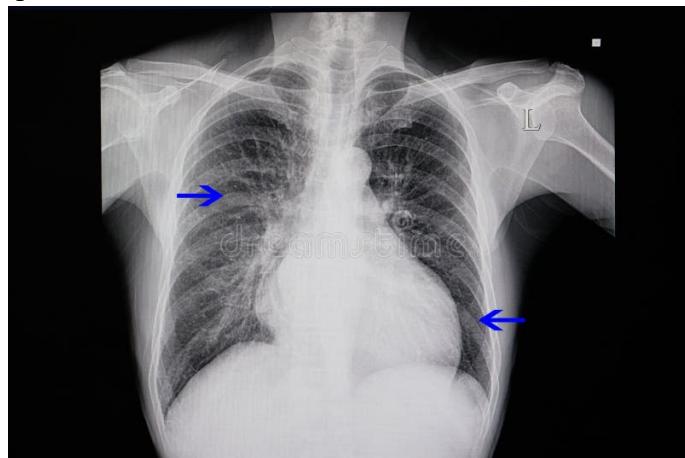


Figura 3.4: Radiografía indicativa de falla cardiaca.⁵

Para obtener los datos, en primer lugar, se filtraron las imágenes quedando solo aquellas que se han etiquetado manualmente por un radiólogo. De este subconjunto, se seleccionaron las imágenes con alguno de los diagnósticos mencionados anteriormente, y se descargan las 5 carpetas con más imágenes cumpliendo todos estos requisitos, que en este caso son la 2, 4, 10 y 54.

⁴ FUENTE: <http://radiopaedia.org>

⁵ FUENTE: <http://radiopaedia.org>

En el conjunto de datos resultante, hay una pequeña minoría de reportes de imágenes que presentan 2 o más diagnósticos, que serán eliminadas para que no generen confusión en el modelo. Es decir, inicialmente se trataba de un problema multietiqueta, donde solo un 2,5% de los reportes indicaban más de un diagnóstico, y por ello se va a adaptar a un problema de clasificación multiclasa. Finalmente, todas las imágenes son redimensionadas a 512×512 .

A continuación, se muestra un diagrama de frecuencias de aparición de cada clase, observando cierto desbalanceo.

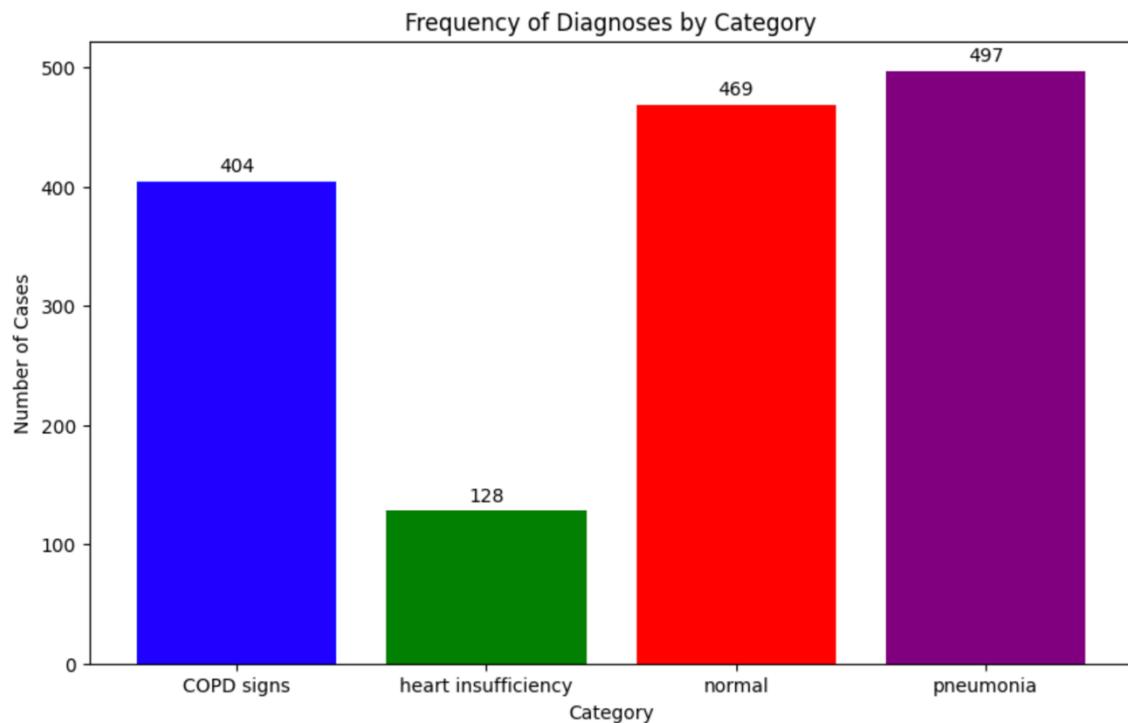


Figura 3.5: Gráfico de la frecuencia de aparición de cada clase

Capítulo 4

IMPLEMENTACIÓN

En este capítulo se presentan los detalles de la implementación y configuración de los modelos así como los diferentes enfoques utilizados para ver cómo se comportaban las redes en cada caso. También se definen las métricas utilizadas para comparar los modelos.

4.1 ARQUITECTURAS Y CONFIGURACIÓN DE HIPERPARÁMETROS

4.1.1 RESNET

En cuanto a la arquitectura, se escogió el modelo ResNet50, utilizando los pesos de ImageNet, aplicando el flujo tradicional de aprendizaje por transferencia, utilizando un entrenamiento en 2 pasos.

El tamaño de las imágenes de entrada se redujo de 512×512 a 224×224 debido a la gran carga computacional, posteriormente eran reescaladas al rango $[0,1]$. Todas las capas de la red fueron descongeladas. La última capa de la ResNet fue reemplazada, para ello, a la salida se añadió una capa de *Pooling*, y posteriormente se incluyeron 3 capas densas con 512, 256 y 128 neuronas respectivamente. Entre capas, se añadió *Dropout* con una probabilidad del 20% y *BatchNormalization*, para evitar problemas de sobreajuste y desvanecimiento de los gradientes. Finalmente se añadió una última capa densa, cuyos parámetros variaban en función del enfoque a resolver, en el caso del enfoque binario, estaba formada por 2 neuronas y función de activación sigmoide, mientras que en el caso del enfoque multiclasa, estaba formada por 4 neuronas y función de activación *softmax*.

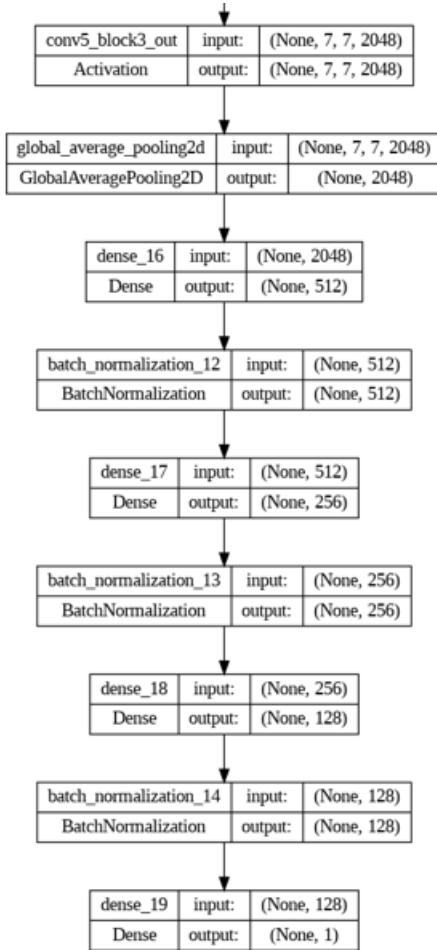


Figura 4.1: Configuración del MLP aplicado al modelo ResNet

Con respecto a los hiperparámetros, se utilizó el optimizador Adam con una tasa de aprendizaje de 0.001 y un decaimiento de 0.01. El número de épocas fue de 50 en todos los casos y el tamaño del batch size de 128.

4.1.2 CXR FOUNDATION

En el diseño de la red, este modelo introduce un paso adicional en el proceso de entrenamiento para mejorar el rendimiento en tareas específicas de radiografía de tórax. El primer paso consiste en un preentrenamiento genérico, igual que en el apartado anterior, utilizando los pesos de ImageNet, seguido de un preentrenamiento específico usando un conjunto de datos específico de radiografías de tórax.

Finalmente, se ajustó un modelo de clasificación no lineal. Para ello, igual que en el caso previo, a la salida del paso anterior se añadió una capa de *Pooling*, y posteriormente se incluyeron 3 capas densas con 512, 256 y 128 neuronas respectivamente. Entre capas, se añadió *Dropout* con una probabilidad del 20% y *BatchNormalization*, para evitar problemas de sobreajuste y desvanecimiento del gradiente. Finalmente se añadió una última capa densa, cuyos parámetros variaban en función del enfoque a resolver; en el caso del enfoque binario, estaba formada por 2 neuronas y función de activación

sigmoide, mientras que en el caso del enfoque multiclas, estaba formada por 4 neuronas y función de activación *softmax*.

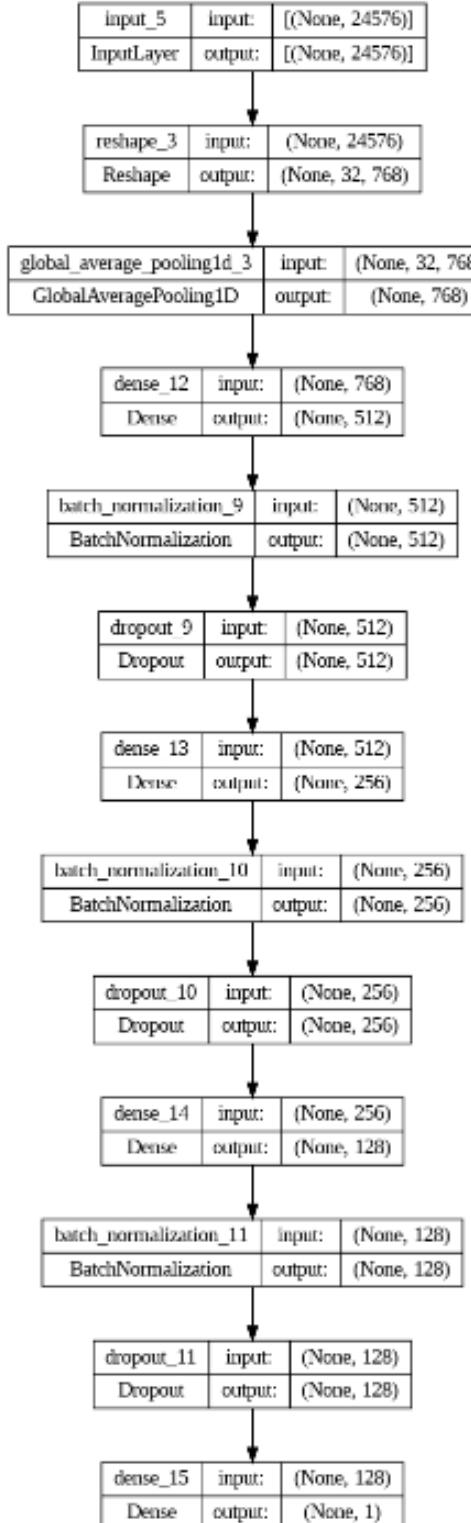


Figura 4.2: Configuración del MLP aplicado al modelo CXR Foundation

Con respecto a los hiperparámetros, se utilizó el optimizador LARS [22], con una tasa de aprendizaje de 0.02 y sin decaimiento. El número de épocas fue de 500 en todos los casos y el tamaño del batch size de 512.

4.2 ENFOQUES

4.2.1 ENFOQUE 1: NO ENFERMO O ENFERMO

Para este primer caso, se codificaron las tres clases de enfermedades dentro de una única clase, de manera que se trataba de resolver un problema binario donde había que clasificar cada imagen como enferma (0) o no enferma (1). A pesar de ser un problema binario, el principal inconveniente de esta aproximación es que todas las enfermedades se codifican con la misma etiqueta, aunque cada enfermedad está identificada por diferentes síntomas. Esto dificulta al modelo establecer patrones claros para identificar cada clase correctamente. Destacar que este enfoque permite asignar pesos en función de la frecuencia de aparición de cada clase y seleccionar el número de muestras deseado para evitar el desbalanceo.

4.2.2 ENFOQUE 2: CLASIFICACIÓN MULTICLASE

En este segundo enfoque se plantea resolver un problema de clasificación multiclasa, en el que hay que identificar la clase correcta a partir de la imagen de entrada.

A diferencia del caso anterior, cada clase está identificada por un conjunto definido de síntomas. Sin embargo, surge el problema del desbalanceo de clases, donde algunas clases tienen significativamente más ejemplos que otras. Esto puede provocar un mal rendimiento en las clases minoritarias o en aquellas con una amplia variedad de síntomas, especialmente cuando algunos de estos síntomas aparecen con poca frecuencia.

4.2.3 ENFOQUE 3: CLASIFICACIÓN BINARIA TRIPLE

En este último caso, se plantean tres problemas binarios, donde, para cada imagen, se intenta identificar si cada una de las enfermedades está presente o no. Posteriormente, se construye un ensemble que recibe las probabilidades de pertenencia a cada clase. En caso de que ninguna enfermedad sea predicha, es decir, todas las predicciones se encuentran por debajo del umbral de clasificación, se asigna la clase "no enfermo", y si se predicen varias clases, el ensemble decide cuál es la correcta, escogiendo aquella cuya diferencia entre el umbral y la predicción sea mayor.

Aunque entrenar tres modelos es computacionalmente más costoso, este enfoque es el más robusto. Esto se debe a que permite balancear las clases manualmente y asignar pesos en función de la frecuencia de aparición de cada clase, como en el primero enfoque. Además, cada modelo se centra en detectar patrones específicos para identificar cada enfermedad, que, junto al ensemble, hace que sea más fiable la predicción.

4.3 MÉTRICAS

A continuación, se definen las métricas de rendimiento para evaluar los modelos.

4.3.1 ENTROPÍA CRUZADA

La entropía cruzada es una medida utilizada para evaluar el rendimiento de los modelos de clasificación. Se define como la cantidad de información adicional necesaria para describir la verdadera distribución de las etiquetas usando la distribución de probabilidades predicha por el modelo, es decir, mide la discrepancia entre las distribuciones de probabilidad real y predicha.

Matemáticamente, la entropía cruzada para un solo ejemplo se expresa como:

$$H(p, q) = - \sum_x p(x) * \log q(x) \quad (4.1)$$

Donde:

- n es el número de clases en el problema de clasificación.
- p es el vector de distribución de probabilidad de las etiquetas reales
- q es el vector de distribución de probabilidad predicha por el modelo.

4.3.2 ÁREA BAJO LA CURVA

El área bajo la curva (AUC, por sus siglas en inglés de "*Area Under the Curve*") es una métrica que presenta la relación entre la tasa de verdaderos positivos (Sensibilidad) y la tasa de falsos positivos (1 - Especificidad) de un modelo, a diferentes umbrales de clasificación. Cada punto, representa un par de valores de sensibilidad y especificidad correspondientes a un umbral específico. Es muy robusta al desbalanceo de clases.

Un AUC de 1 indica que el modelo tiene una capacidad perfecta para clasificar correctamente todos los positivos y negativos, mientras que un AUC de 0.5 indica que el modelo realiza una asignación aleatoria.

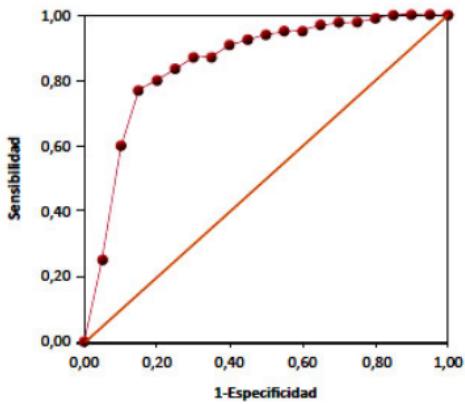


Figura 4.3: Imagen de una curva ROC⁶

4.3.3 MATRIZ DE CONFUSIÓN

La matriz de confusión es una tabla que permite visualizar la precisión de un algoritmo de clasificación, comparando los valores predichos por el modelo con los valores reales observados. Permite analizar los errores cometidos entre clases.

		Predicted 0	Predicted 1
Actual 0	TN	FP	
	FN	TP	

Figura 4.4: Esquema de una matriz de confusión⁷

- **Verdaderos Positivos (TP):** Casos en los que el modelo predice correctamente la clase positiva.
- **Verdaderos Negativos (TN):** Casos en los que el modelo predice correctamente la clase negativa.
- **Falsos Positivos (FP):** Casos en los que el modelo predice incorrectamente un resultado positivo.
- **Falsos Negativos (FN):** Casos en los que el modelo predice incorrectamente un resultado negativo.

⁶ FUENTE: <https://towardsdatascience.com/a-simple-explanation-of-the-roc-curve-and-auc-64db32d75541>

⁷ FUENTE: https://miro.medium.com/v2/resize:fit:1780/1*LQ1YMKBlbDhH9K6Ujz8QTw.jpeg

A partir de la matriz de confusión, se calculan las siguientes métricas:

4.3.4 ACCURACY

El *accuracy* se define como la proporción de predicciones correctas (tanto verdaderos positivos como verdaderos negativos) entre el total de casos examinados. Aunque es la métrica más común, es muy poco adecuada en problemas de clasificación desbalanceada.

Matemáticamente se define como:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.2)$$

4.3.5 PRECISIÓN

La precisión es la proporción de predicciones positivas que fueron realmente correctas.

Matemáticamente se define como:

$$Precision = \frac{TP}{TP + FP} \quad (4.3)$$

4.3.6 RECALL

El *recall* indica cuántos de los casos realmente positivos fueron capturados por el modelo mediante predicciones positivas.

Matemáticamente se define como:

$$Recall = \frac{TP}{TP + FN} \quad (4.4)$$

4.3.7 F1 SCORE

Esencialmente, el *F1 Score* es la media armónica de la precisión y el *recall*, lo que proporciona un equilibrio entre estas dos métricas. Se utiliza ampliamente en situaciones donde no se desea comprometer ni la precisión ni el recall en favor del otro.

Matemáticamente se define como:

$$F_1 = 2 \times \left(\frac{precision \times recall}{precision + recall} \right) \quad (4.5)$$

Un valor F1 de 1 indica que el modelo es perfecto, es decir, tiene tanto una Precisión como un Recall del 100%, mientras que un valor F1 de 0 indica que al menos uno entre precisión o recall es 0.

4.4 ELECCIÓN ÓPTIMA DE HIPERPARÁMETROS

La elección de hiperparámetros es una tarea compleja que generalmente se realiza por prueba y error. Al inicio de este capítulo, se presentó la configuración de hiperparámetros que resultó ser la mejor para entrenar los modelos considerados. Esto se logró mediante una búsqueda exhaustiva, probando todas las posibles combinaciones y eligiendo la que mejora las métricas de rendimiento.

A continuación, se muestra una tabla con los parámetros considerados y óptimos para cada modelo, para obtenerlos se consideraron 100 épocas para el modelo de CXR Foundation y 10 para ResNet. Aunque idealmente es mejor entrenar con más épocas, por falta de recursos computacionales es la decisión considerada.

Tabla 1

Configuración de hiperparámetros para el modelo CXR Foundation

PARÁMETRO	VALORES	ÓPTIMO
OPTIMIZADOR	Adam, Lars, SGD	Lars
BATCH SIZE	64, 128, 256, 512, 1024	512
TASA APRENDIZAJE	0.1, 0.02, 0.003, 0.0004	0.02
TASA DECAIMIENTO	0, 0.01, 0.03	0.03
DROPOUT	0, 0.2, 0.4	0.2
Nº DE CAPAS	1,2,3	3
Nº NEURONAS POR CAPA	512, 256, 128	512,256,128

Tabla 2

Configuración de hiperparámetros para el modelo ResNet

PARÁMETRO	VALORES	ÓPTIMO
OPTIMIZADOR	Adam, Lars, SGD	ADAM
BATCH SIZE	64, 128, 256, 512, 1024	128
TASA APRENDIZAJE	0.1, 0.02, 0.003, 0.0004	0.003
TASA DECAIMIENTO	0, 0.01, 0.03	0.01
DROPOUT	0, 0.2, 0.4	0.2
Nº DE CAPAS	1,2,3	3
Nº NEURONAS POR CAPA	512, 256, 128	512, 256, 128

Capítulo 5

RESULTADOS

En este capítulo, se presentan y analizan los resultados de los experimentos descritos en el capítulo anterior. El procedimiento es el siguiente: para cada enfoque, se muestra el resultado obtenido de los dos modelos. Se analiza: la evolución de la función de pérdidas y el área bajo la curva durante el entrenamiento para el conjunto de entrenamiento y validación. También se muestra el área bajo la curva, la matriz de confusión, la precisión, el *recall*, el *accuracy* y F1-score para el conjunto de prueba.

5.1 ENFOQUE 1: NO ENFERMO O ENFERMO

Recordar que el primer enfoque se trataba de un problema binario en el que todas las enfermedades se codificaban con la misma etiqueta, de forma que el problema se reducía a clasificar cada imagen como no enferma (1) o enferma (0).

5.1.1 CXR FOUNDATION

5.1.1.1 MONITORIZACIÓN DE LA FUNCIÓN DE PÉRDIDAS Y CURVA ROC

Las figuras 5.1 y 5.2 muestran el valor de la función de pérdidas y el área bajo la curva para el conjunto de entrenamiento y validación con el paso de las épocas para el modelo CXR Foundation.

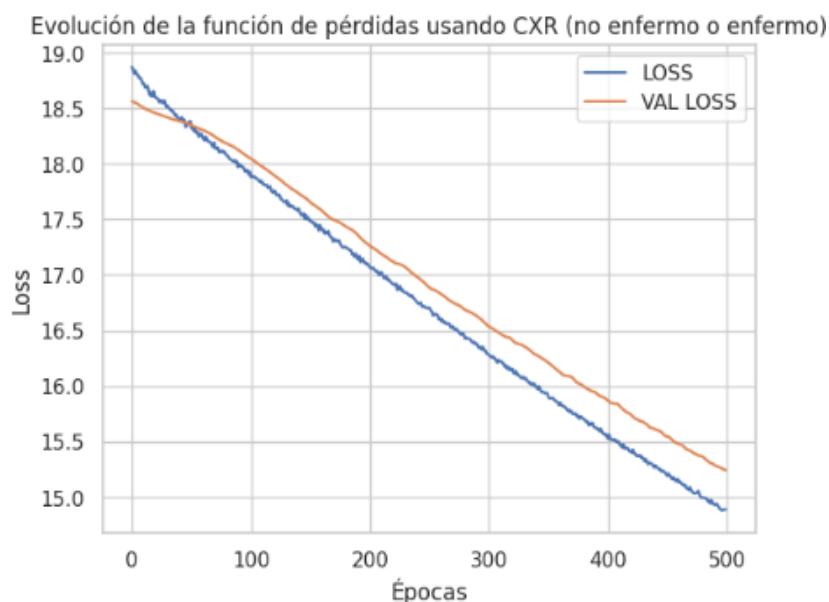


Figura 5.1: Monitorización de la función de pérdidas para el modelo CXR en el enfoque 1

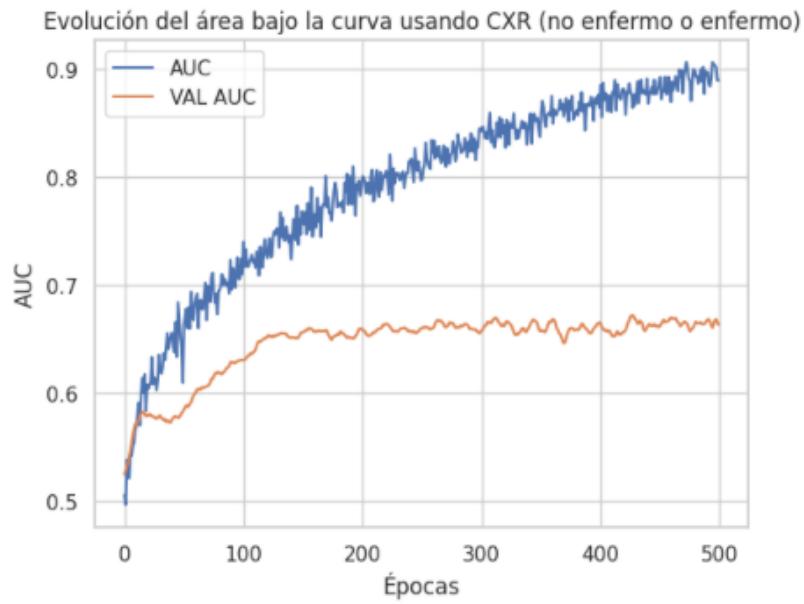


Figura 5.2: Monitorización de la curva ROC para el modelo CXR en el enfoque 1

Las figuras 5.3 y 5.4 muestran el área bajo la curva y matriz de confusión para el conjunto de prueba, respectivamente.

5.1.1.2 CURVA ROC

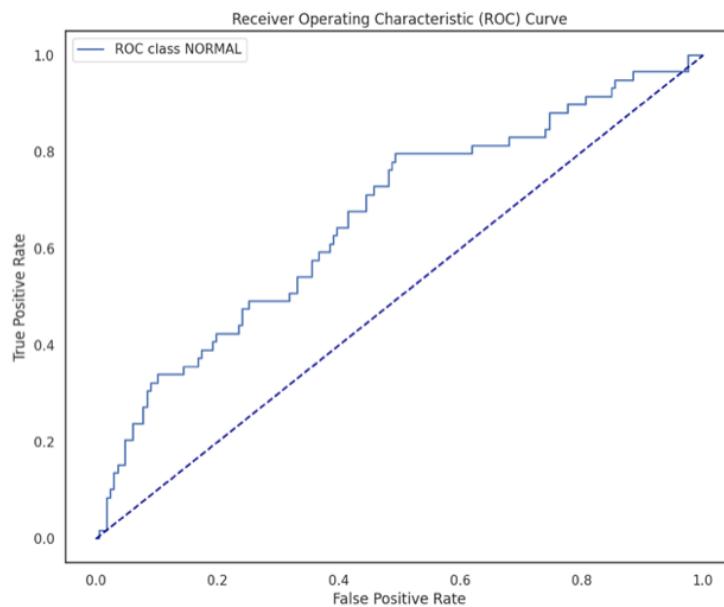


Figura 5.3: Curva ROC para el modelo CXR en el enfoque 1

5.1.1.3 MATRIZ DE CONFUSIÓN

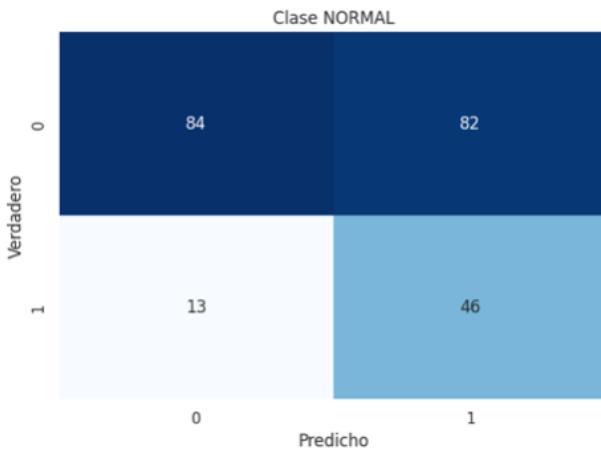


Figura 5.4: Matriz de confusión para el modelo CXR Foundation en el enfoque 1

5.1.2 ResNet

Las figuras 5.5 y 5.6 muestran el valor de la función de pérdidas y el área bajo la curva para el conjunto de entrenamiento y validación con el paso de las épocas para el modelo ResNet del primer enfoque.

5.1.2.1 MONITORIZACIÓN DE LA FUNCIÓN DE PÉRDIDAS Y CURVA ROC

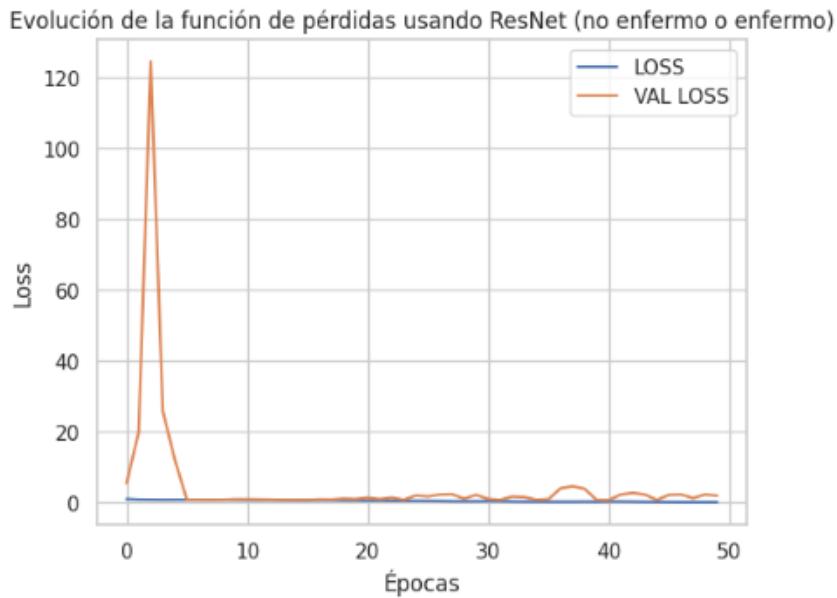


Figura 5.5: Monitorización de la función de pérdidas para el modelo ResNet en el enfoque 1

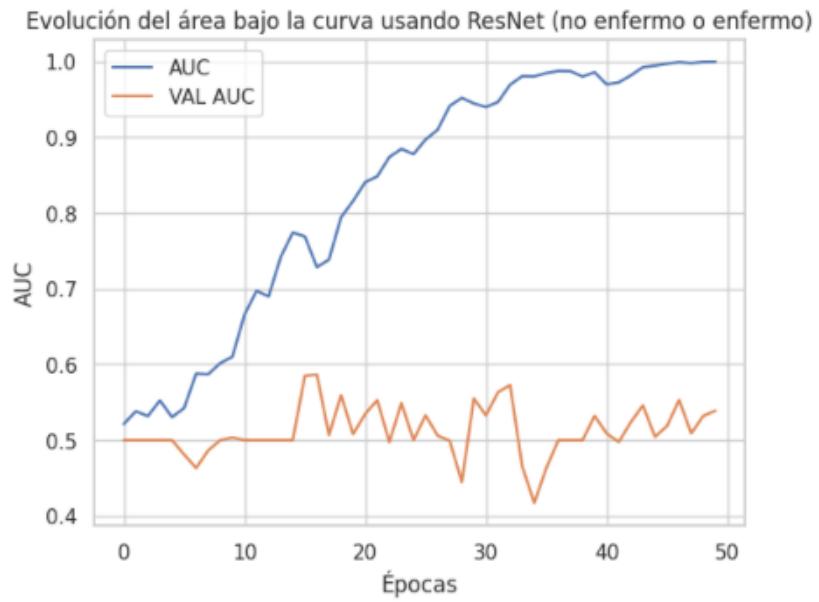


Figura 5.6: Monitorización de la curva ROC para el modelo ResNet en el enfoque 1

A continuación, las figuras 5.7 y 5.8 muestran el área bajo la curva y matriz de confusión del modelo ResNet para el conjunto de prueba, respectivamente. El área bajo la curva permite diferentes valores de sensibilidad y especificidad asociados a diferentes umbrales, y la matriz de confusión permite ver las predicciones correctas e incorrectas entre clases.

5.1.2.2 CURVA ROC

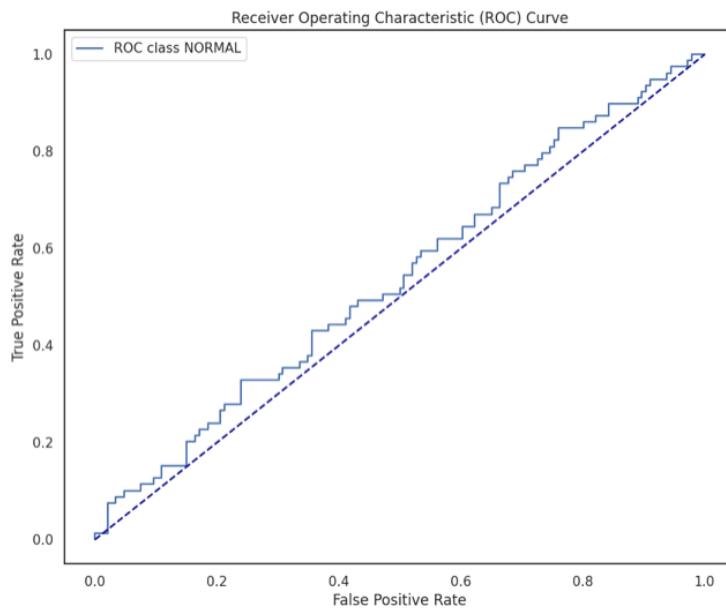


Figura 5.7: Curva ROC para el modelo ResNet en el enfoque 1

5.1.2.3 MATRIZ DE CONFUSIÓN

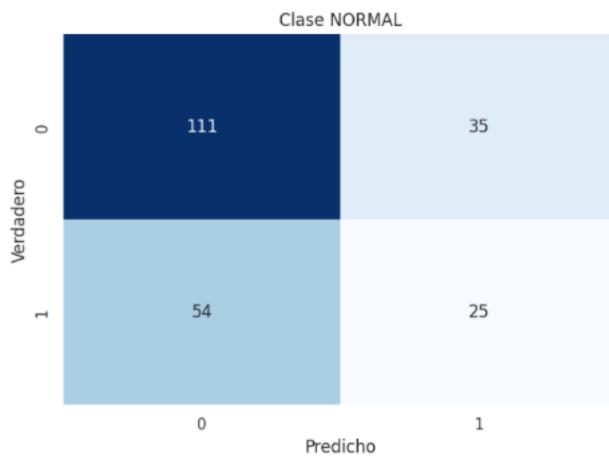


Figura 5.8: Matriz de confusión para el modelo ResNet en el enfoque 1

5.1.3 RESUMEN DE CLASIFICACIÓN DEL ENFOQUE 1

Tabla 3

Resumen de clasificación del enfoque 1

MÉTRICA	CXR FOUNDATION	RESNET
ÁREA BAJO LA CURVA	0.66	0.54
PRECISIÓN	0.36	0.42
RECALL	0.78	0.32
F1-SCORE	0.49	0.36
ACCURACY	0.58	0.60

Se puede observar que el comportamiento de la ResNet es prácticamente aleatorio, ya que la curva ROC es muy próxima a 0.5 para el conjunto de prueba, estos resultados corroboran los valores obtenidos en la monitorización de la función de pérdidas y el área bajo la curva durante el entrenamiento.

Por otro lado, el modelo CXR Foundation, muestra un rendimiento superior. Su mayor carencia son la cantidad de falsos positivos, por ello, el *recall* es ampliamente superior a la precisión.

5.2 ENFOQUE 2: CLASIFICACIÓN MULTICLASE

El segundo enfoque trataba de un problema multiclase, en el que hay que identificar la clase correcta a partir de la imagen de entrada.

5.2.1 CXR FOUNDATION

Las figuras 5.9 y 5.10 muestran el valor de la función de pérdidas y el área bajo la curva para el conjunto de entrenamiento y validación con el paso de las épocas para el modelo CXR Foundation.

5.2.1.1 MONITORIZACIÓN DE LA FUNCIÓN DE PÉRDIDAS Y CURVA ROC

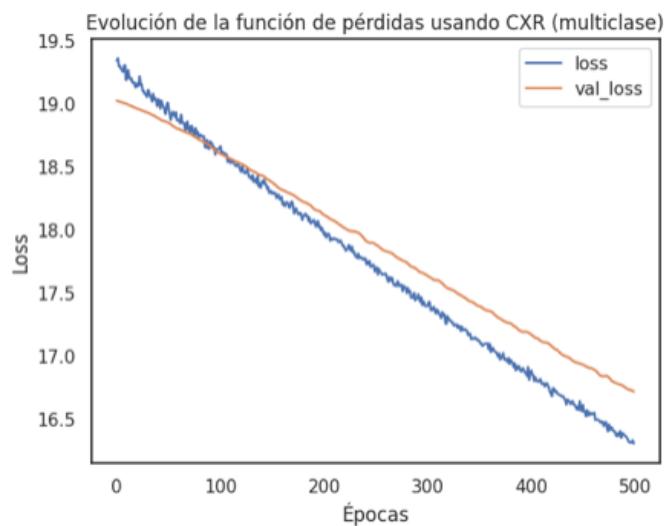


Figura 5.9: Monitorización de la función de pérdidas para el modelo CXR en el enfoque 2

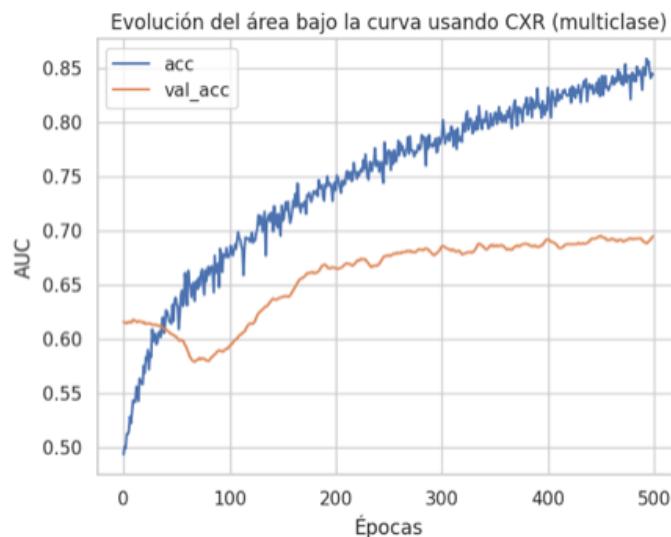


Figura 5.10: Monitorización de la curva ROC para el modelo CXR en el enfoque 2

Seguidamente, las figuras 5.11 y 5.12 muestran el área bajo la curva y matriz de confusión del modelo CXR Foundation para el conjunto de prueba, respectivamente.

5.2.1.2 CURVA ROC

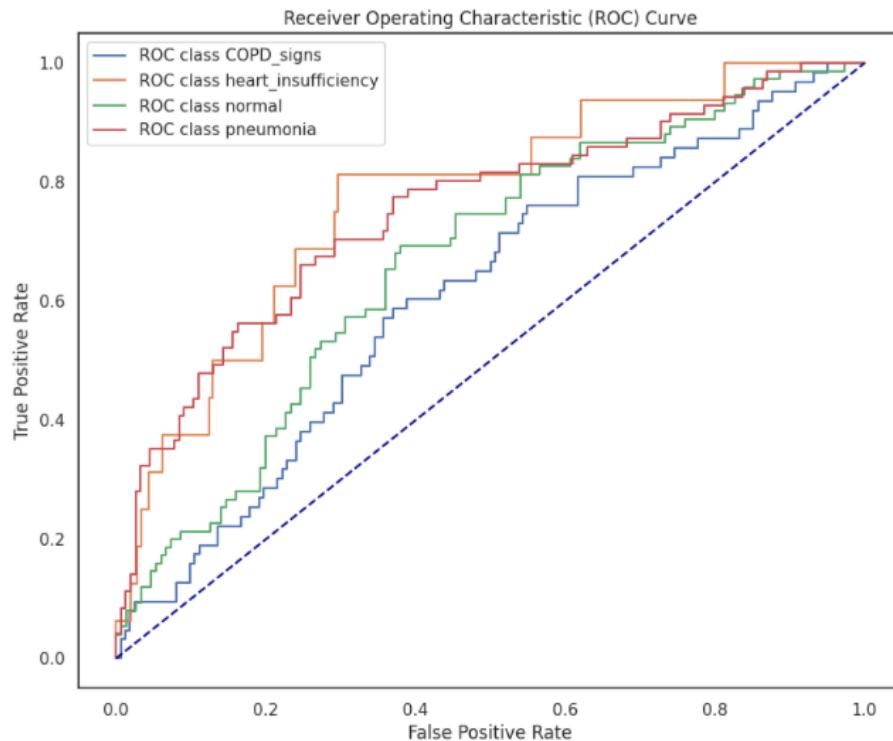


Figura 5.11: Curva ROC para el modelo CXR en el enfoque 2 (multiclas)

5.2.1.3 MATRIZ DE CONFUSIÓN

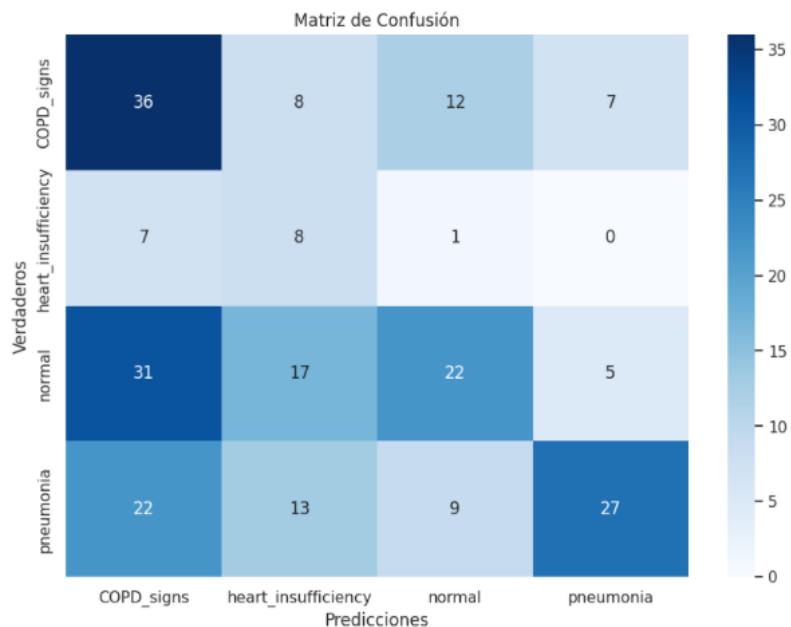


Figura 5.12: Matriz de confusión para el modelo CXR Foundation en el enfoque 2 (multiclas)

5.2.2 ResNet

Las figuras 5.13 y 5.14 muestran el valor de la función de pérdidas y el área bajo la curva para el conjunto de entrenamiento y validación con el paso de las épocas para el modelo ResNet.

5.2.2.1 MONITORIZACIÓN DE LA FUNCIÓN DE PÉRDIDAS Y CURVA ROC

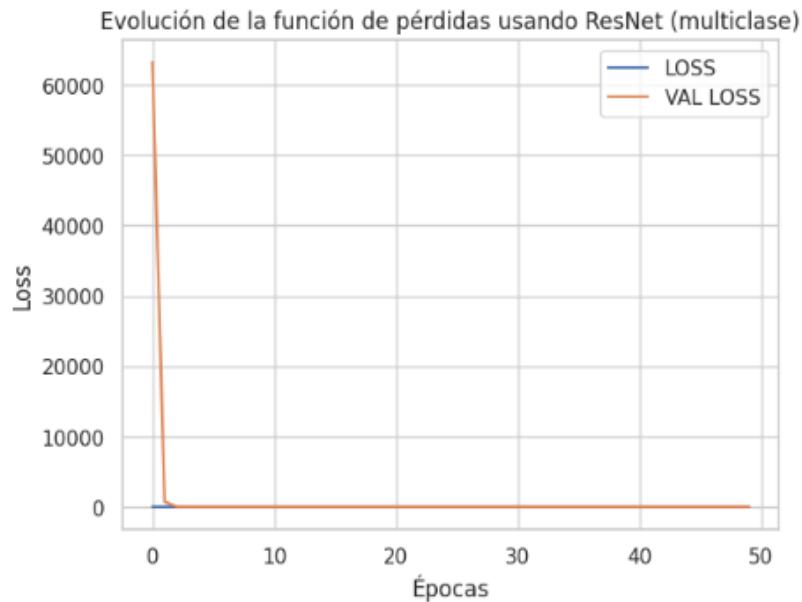


Figura 5.13: Monitorización de la función de pérdidas para el modelo ResNet en el enfoque 2

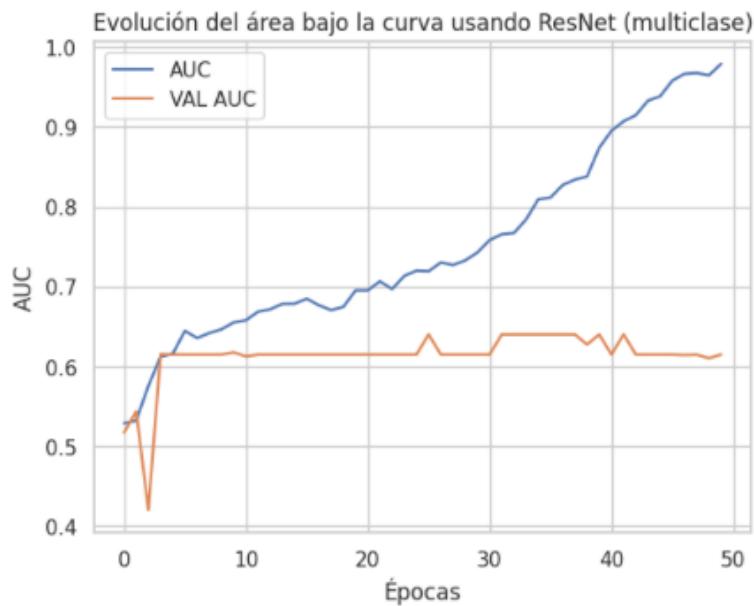


Figura 5.14: Monitorización de la curva ROC para el modelo ResNet en el enfoque 2

A continuación, las figuras 5.15 y 5.16 muestran el área bajo la curva y matriz de confusión del modelo ResNet para el conjunto de prueba, respectivamente. El área bajo la curva permite diferentes valores de sensibilidad y especificidad asociados a diferentes umbrales, y la matriz de confusión permite ver las predicciones correctas e incorrectas entre clases.

5.2.2.2 CURVA ROC

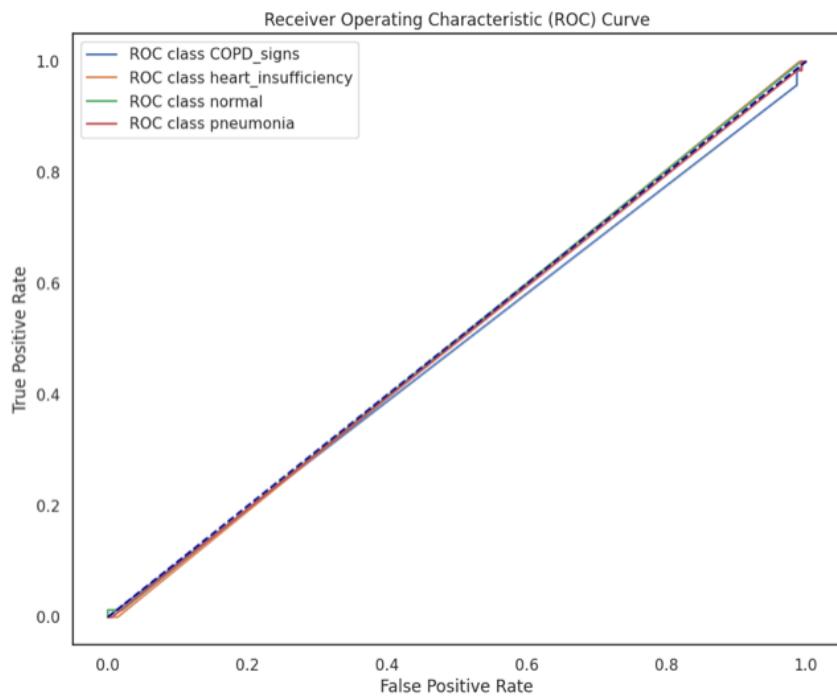


Figura 5.15: Curva ROC para el modelo ResNet en el enfoque 2

5.2.2.3 MATRIZ DE CONFUSIÓN

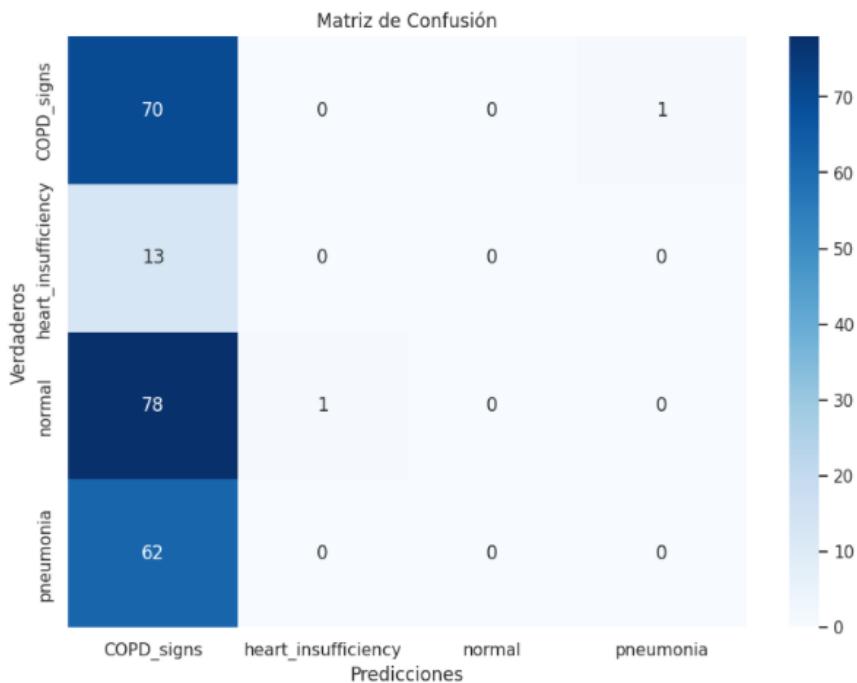


Figura 5.16: Matriz de confusión para el modelo ResNet en el enfoque 2 (multiclas)

5.2.3 RESUMEN DE CLASIFICACIÓN DEL ENFOQUE 2

Tabla 4

Resumen de clasificación del enfoque 2

MÉTRICA	CXR				RESNET			
	EPOC	INSUF.	NORMAL	NEUM.	EPOC	INSUF.	NORMAL	NEUM.
AUC	0.61	0.77	0.67	0.75	0.49	0.50	0.50	0.50
PRECISION	0.38	0.16	0.47	0.55	0.31	0.00	0.00	0.27
RECALL	0.57	0.75	0.68	0.65	0.99	0.00	0.00	0.98
F1-SCORE	0.45	0.27	0.56	0.59	0.48	0.00	0.00	0.43
ACCURACY	0.61	0.71	0.64	0.72	0.32	0.93	0.65	0.28

En este enfoque se muestra también que el comportamiento de la ResNet es prácticamente aleatorio, la curva ROC es muy próxima a 0.5 para el conjunto de prueba en las 4 clases y se obtienen valores de precisión, *recall* y F1 de 0 para las clases “Insuficiencia cardíaca” y “Normal”. Esto ocurre porque el modelo asigna todas las muestras a la misma clase y no se puede calcular el resultado. Para las clases “Signos EPOC” y “Neumonía” muestra un valor próximo a 1 de *recall* y de 0.3 aproximadamente de precisión, debido a la cantidad de falsos positivos.

En el caso del modelo CXR Foundation, muestra mejor rendimiento, siendo la clase “Insuficiencia cardíaca” la más compleja de detectar, en su caso también se observa un valor de precisión muy inferior al de recall, por la cantidad de falsos positivos. En el resto de enfermedades el comportamiento es el mismo, pero sin tanta diferencia entre ambos valores.

5.3 ENFOQUE 3

Recordar que en este enfoque se entran tres clasificadores diferentes que se combinan posteriormente usando un ensemble para poder comparar con el modelo anterior.

5.3.1 CXR FOUNDATION

5.3.1.1 MONITORIZACIÓN DE LA FUNCIÓN DE PÉRDIDAS Y CURVA ROC

Las figuras 5.17, 5.18, 5.19, 5.20, 5.21, 5.22 muestran las curvas de aprendizaje de cada problema binario planteado para el modelo CXR Foundation.

CLASE SIGNOS EPOC

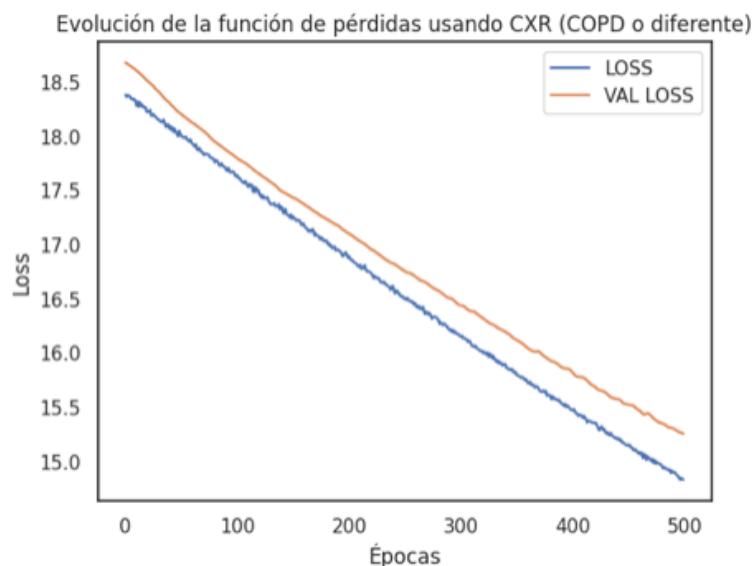


Figura 5.17: Monitorización de la función de pérdidas para el modelo CXR en el enfoque 3 (clase COPD o diferente)

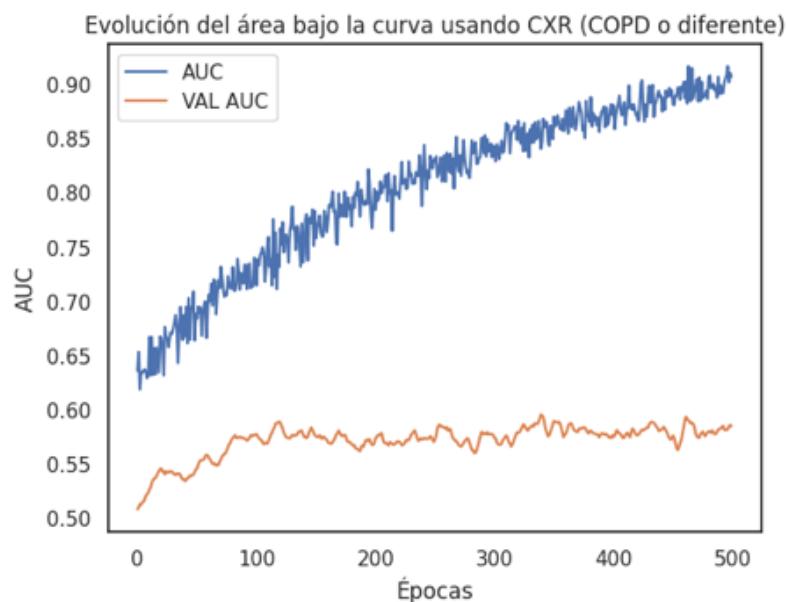


Figura 5.18: Monitorización de la curva ROC para el modelo CXR en el enfoque 3 (clase COPD o diferente)

CLASE INSUFICIENCIA CARDIACA

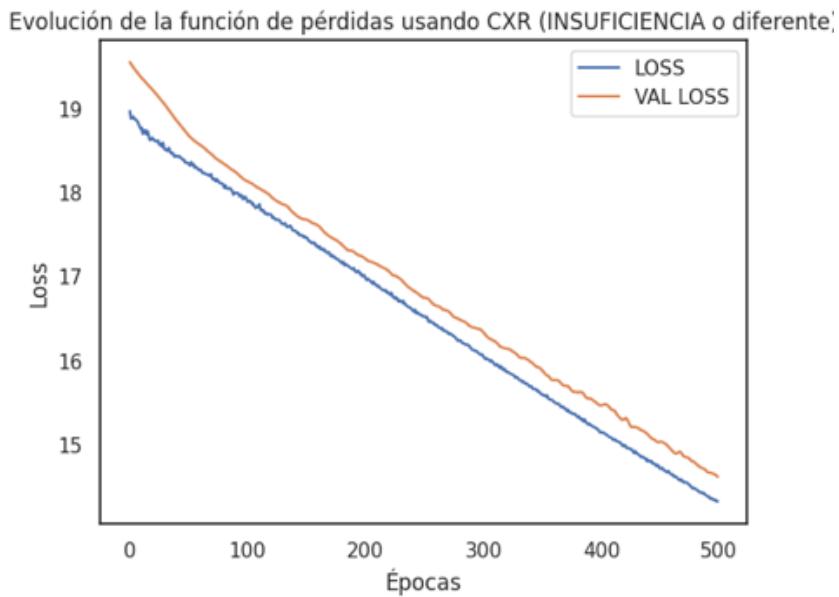


Figura 5.19 : Monitorización de la función de pérdidas para el modelo CXR en el enfoque 3 (clase INSUFICIENCIA CARDIACA o diferente)

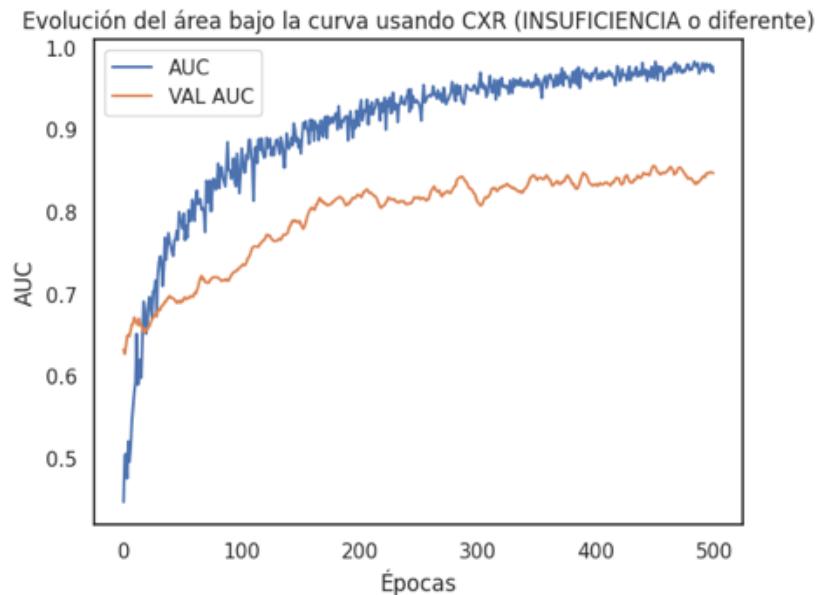


Figura 5.20: Monitorización de la curva ROC para el modelo CXR en el enfoque 3 (clase INSUFICIENCIA CARDIACA o diferente)

CLASE NEUMONÍA

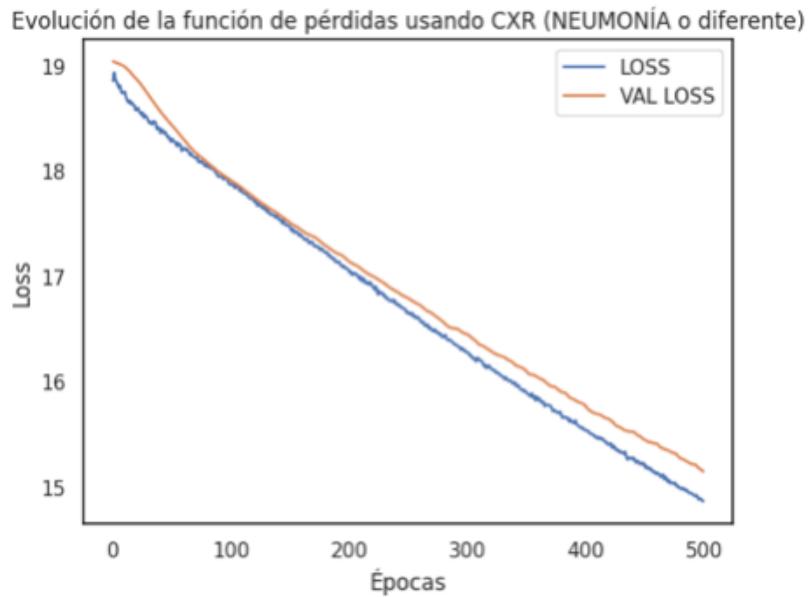


Figura 5.21: Monitorización de la función de pérdidas para el modelo CXR en el enfoque 3 (clase NEUMONÍA o diferente)

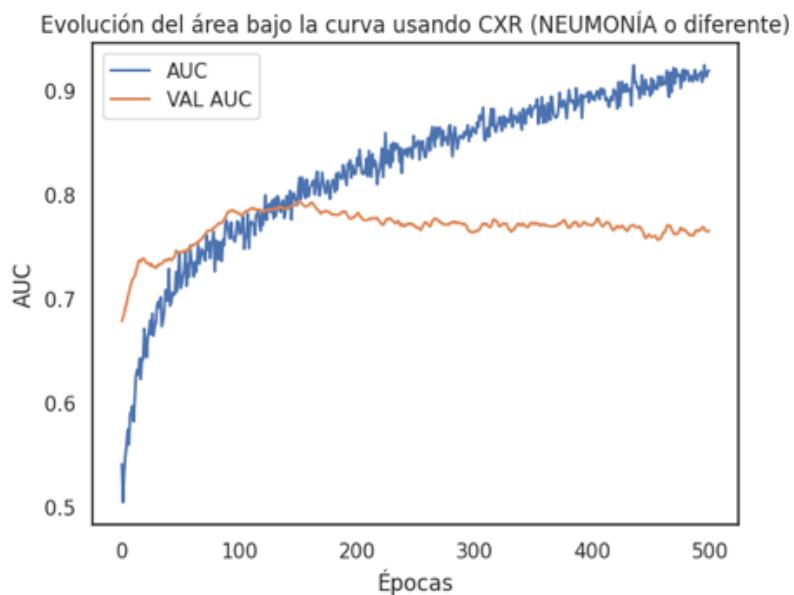


Figura 5.22: Monitorización de la curva ROC para el modelo CXR en el enfoque 3 (clase NEUMONÍA o diferente)

La figura 5.23 muestra el área bajo la curva para cada uno de los problemas binarios considerados en dicho enfoque para el modelo CXR Foundation.

5.3.1.2 CURVA ROC

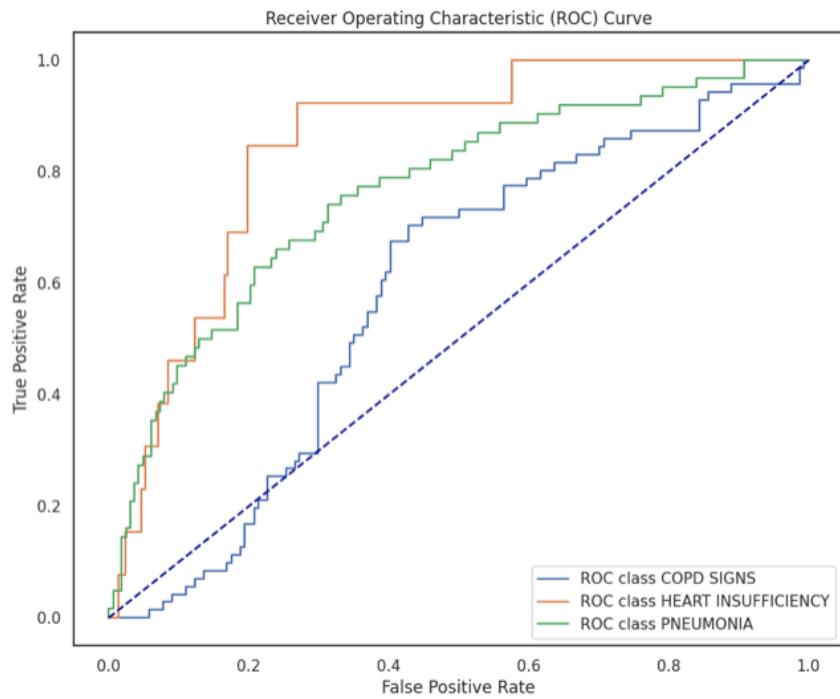


Figura 5.23: Curva ROC de cada clase para el modelo CXR en el enfoque 3

Seguidamente, la figura 5.24 muestra la matriz de confusión de la salida del ensemble, como si fuera un problema multiclase, para el modelo CXR Foundation.

5.3.1.3 MATRIZ DE CONFUSIÓN

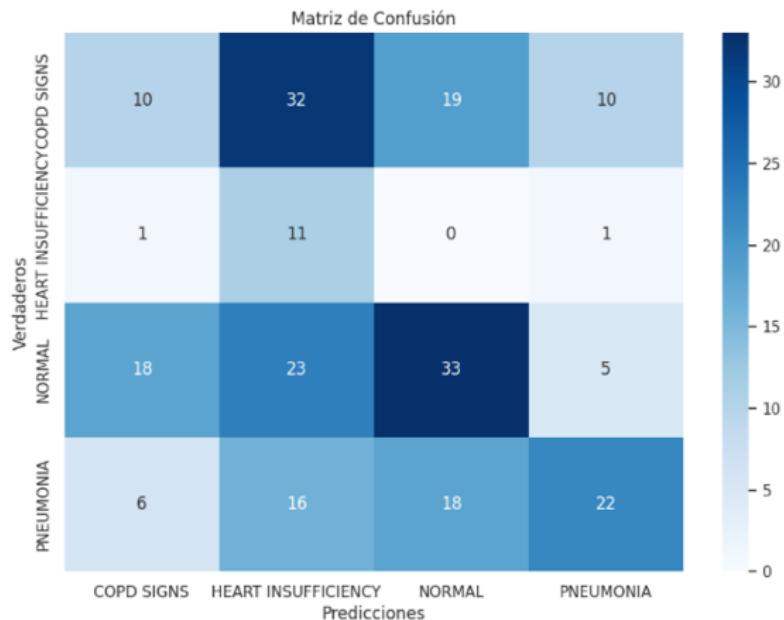


Figura 5.24: Matriz de confusión del ensemble en el enfoque 3 (multiclas) usando predicciones de CXR Foundation

Las figuras 5.25, 5.26 y 5.27 muestran las matrices de confusión de cada uno de los problemas binarios en el tercer enfoque para el modelo CXR Foundation.

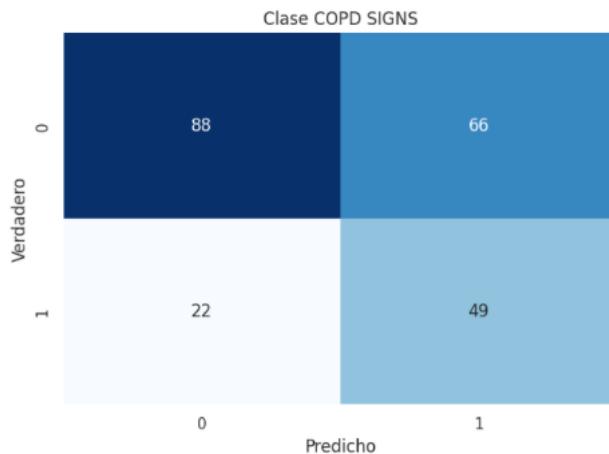


Figura 5.25: Matriz de confusión para el modelo CXR Foundation en el enfoque 3 (clase signos EPOC o diferente)

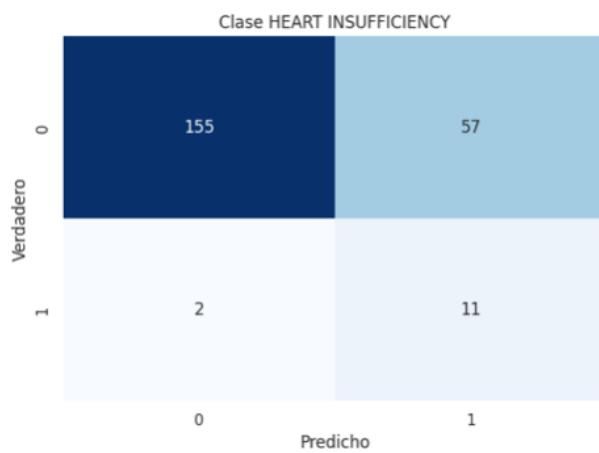


Figura 5.26: Matriz de confusión para el modelo CXR Foundation en el enfoque 3 (clase INSUFICIENCIA CARDIACA o diferente)

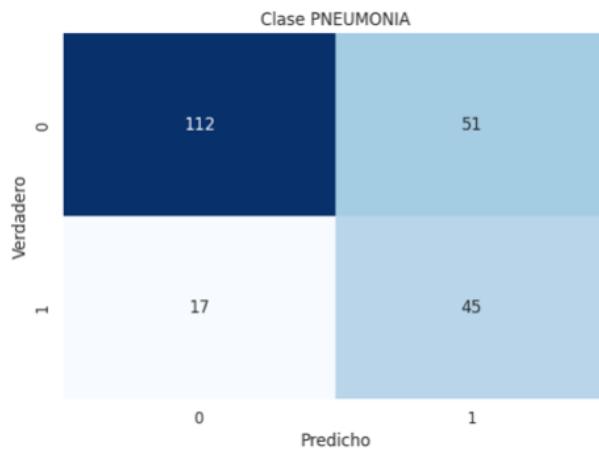


Figura 5.27: Matriz de confusión para el modelo CXR Foundation en el enfoque 3 (clase NEUMONÍA o diferente)

5.3.2 ResNet

Las figuras 5.28, 5.29, 5.30, 5.31, 5.32, 5.33 muestran las curvas de aprendizaje de cada problema binario planteado para el modelo ResNet.

5.3.2.1 MONITORIZACIÓN DE LA FUNCIÓN DE PÉRDIDAS Y CURVA ROC

CLASE SIGNOS EPOCH

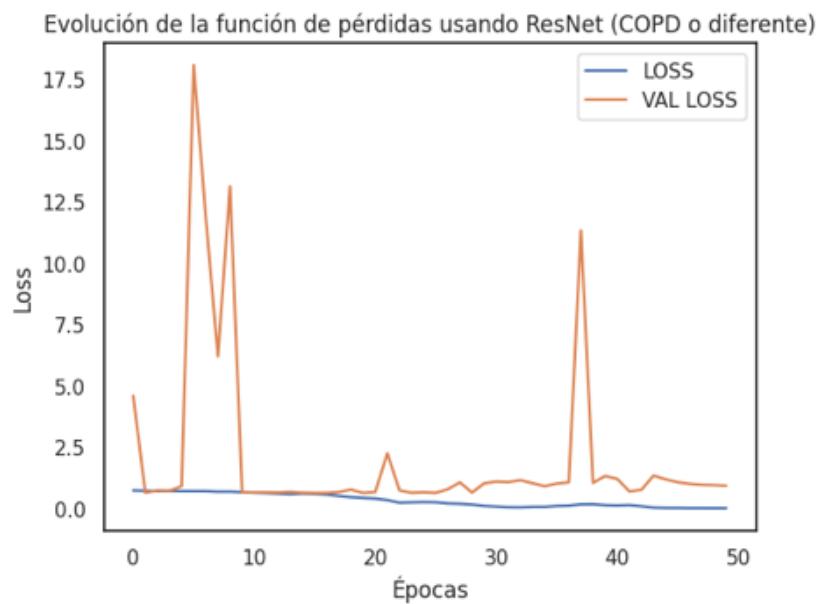


Figura 5.28: Monitorización de la función de pérdidas para el modelo ResNet en el enfoque 3 (clase Signos EPOCH o diferente)

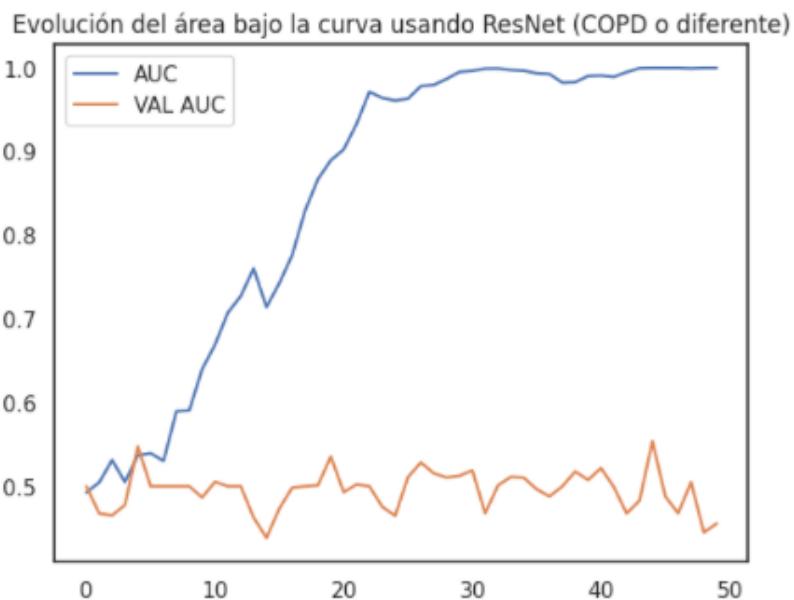


Figura 5.29: Monitorización de la curva ROC para el modelo ResNet en el enfoque 3 (clase Signos EPOCH o diferente)

Evolución de la función de pérdidas usando ResNet (INSUFICIENCIA o diferente)

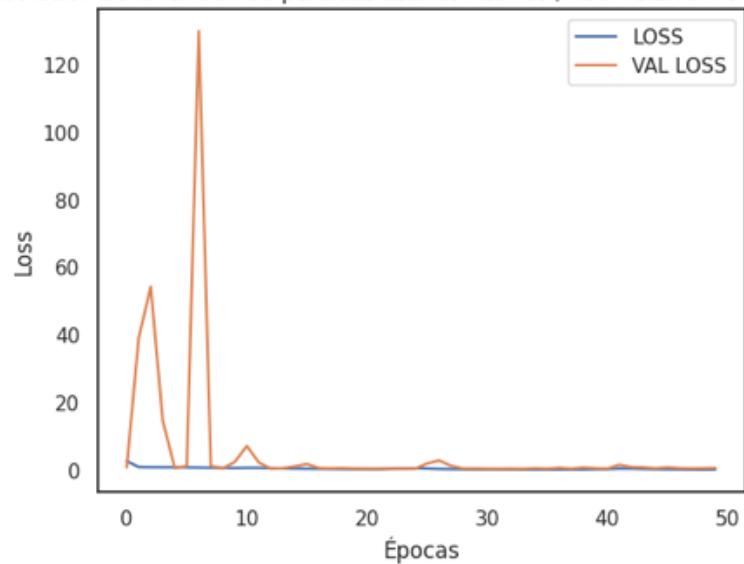


Figura 5.30: Monitorización de la función de pérdidas para el modelo ResNet en el enfoque 3 (clase INSUFICIENCIA CARDIACA o diferente)

Evolución del área bajo la curva usando ResNet (INSUFICIENCIA o diferente)

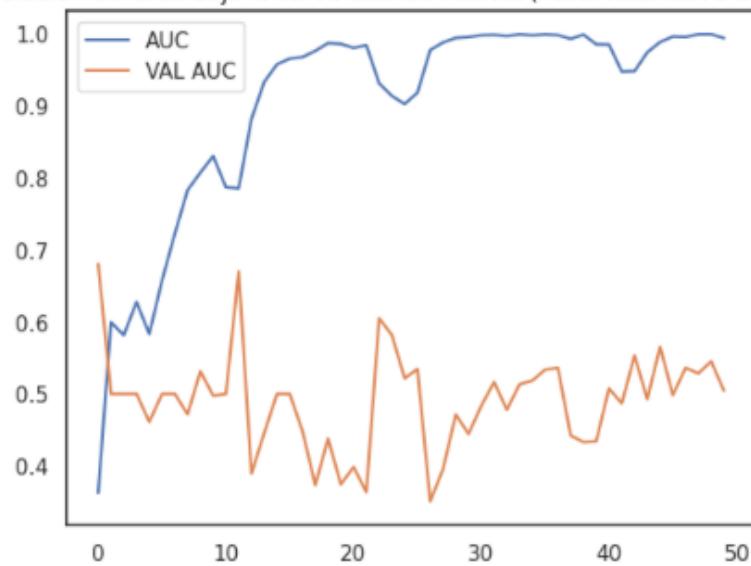


Figura 5.31: Monitorización de la Curva ROC para el modelo ResNet en el enfoque 3 (clase INSUFICIENCIA CARDIACA o diferente)

Evolución de la función de pérdidas usando ResNet (NEUMONÍA o diferente)

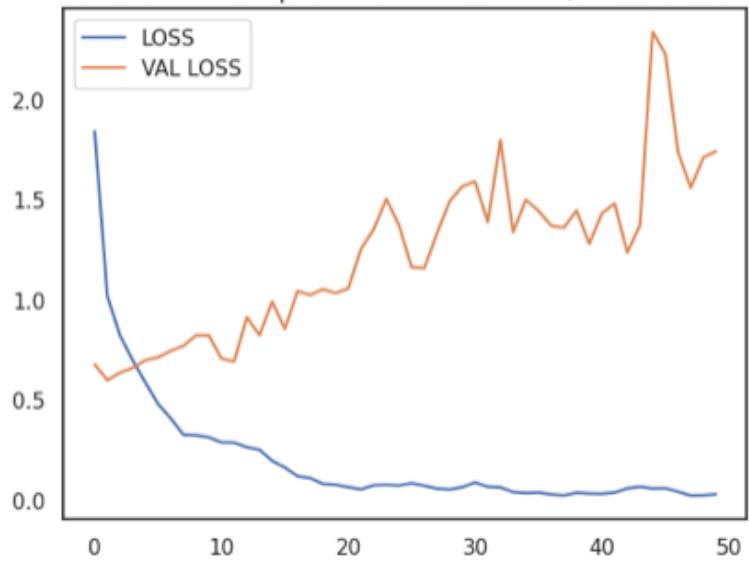


Figura 5.32: Monitorización de la función de pérdidas para el modelo ResNet en el enfoque 3 (clase NEUMONÍA o diferente)

Evolución del área bajo la curva usando ResNet (NEUMONÍA o diferente)

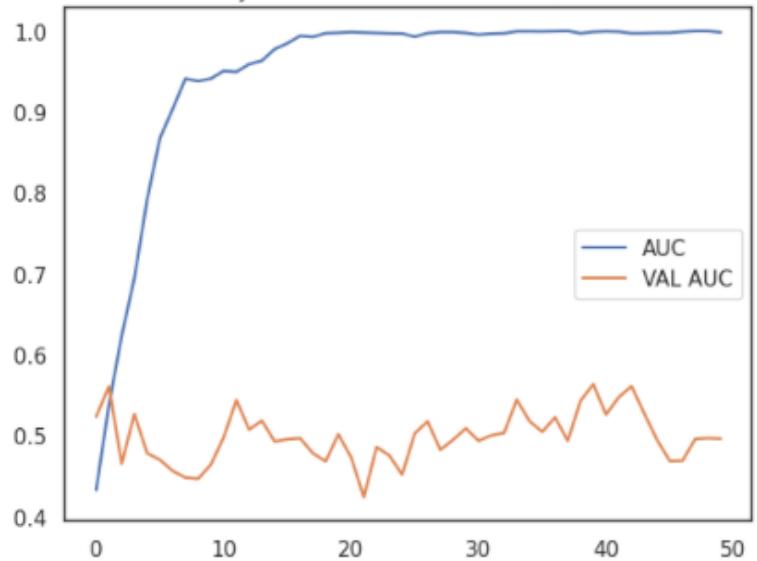


Figura 5.33: Monitorización de la curva ROC para el modelo ResNet en el enfoque 3 (clase NEUMONÍA o diferente)

La figura 5.23 muestra el área bajo la curva para cada uno de los problemas binarios considerados en dicho enfoque del modelo CXR Foundation.

5.3.2.2 CURVA ROC

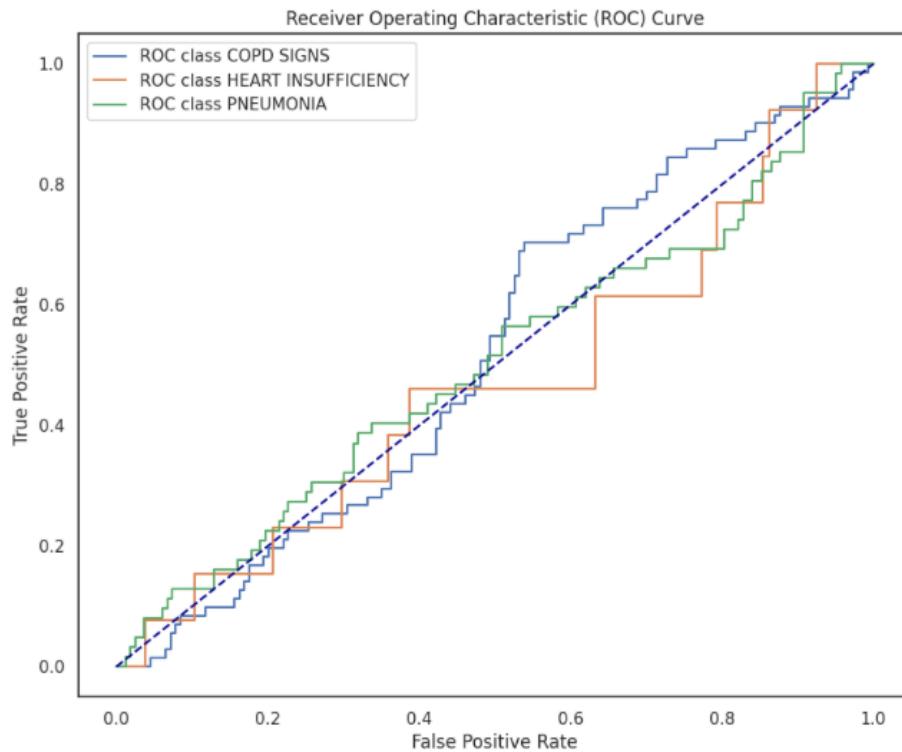


Figura 5.34: Curva ROC de cada clase para el modelo ResNet en el enfoque 3

La figura 5.35 muestra la matriz de confusión de la salida del ensemble, como si fuera un problema multiclase para el modelo ResNet

5.3.2.3 MATRIZ DE CONFUSIÓN

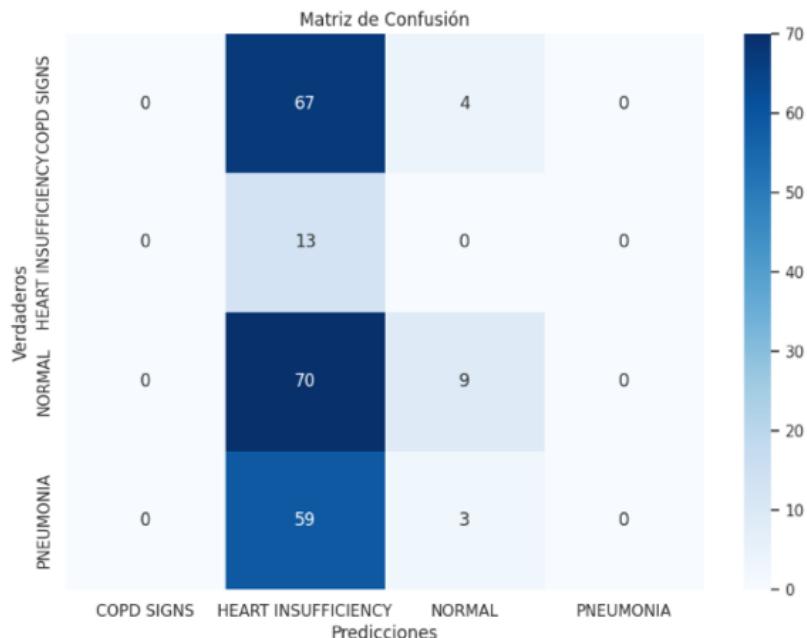


Figura 5.35: Matriz de confusión del ensemble en el enfoque 3 (multiclase) usando predicciones de ResNet

Las figuras 5.36, 5.37 y 5.38 muestran las matrices de confusión de cada uno de los problemas binarios en el tercer enfoque para el modelo CXR Foundation.

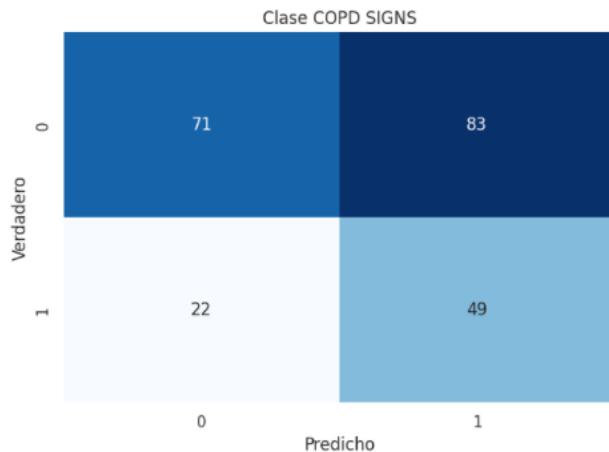


Figura 5.36: Matriz de confusión para el modelo ResNet en el enfoque 3 (clase COPD o diferente)

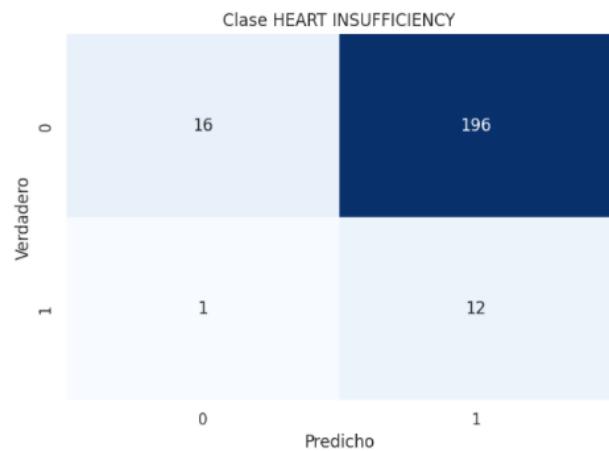


Figura 5.37: Matriz de confusión para el modelo ResNet en el enfoque 3 (clase INSUFICIENCIA CARDIACA o diferente)

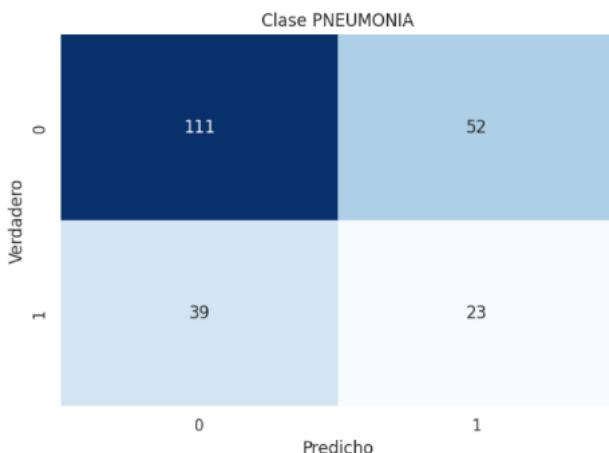


Figura 5.38: Matriz de confusión para el modelo ResNet en el enfoque 3 (clase NEUMONÍA o diferente)

5.3.3 RESUMEN DE CLASIFICACIÓN DEL ENFOQUE 3

Tabla 5

Resumen de clasificación del enfoque 3

MÉTRICA	CXR			RESNET		
	EPOC	INSUF.	NEUM.	EPOC	INSUF	NEUM.
AUC	0.58	0.85	0.77	0.52	0.47	0.50
PRECISION	0.43	0.16	0.47	0.37	0.06	0.31
RECALL	0.69	0.85	0.73	0.69	0.92	0.37
F1-SCORE	0.53	0.27	0.57	0.48	0.11	0.34
ACCURACY	0.61	0.74	0.70	0.53	0.12	0.60

Como se viene observando, en este caso el comportamiento de la ResNet también es aleatorio, mostrando resultados en el área bajo la curva de 0.5 aproximadamente en todos los casos, aunque sin ser tan extremo como en el enfoque anterior que clasificaba todas las muestras con la misma etiqueta.

Por otro lado, el modelo CXR Foundation muestra resultados ligeramente superiores al apartado anterior para las clases “Insuficiencia cardiaca” y “Neumonía” en el área bajo la curva, y ligeramente inferior para la clase “Signos EPOC”. El *recall* es superior en todos los casos respecto al enfoque anterior, y la precisión también lo es, excepto en la clase “Insuficiencia cardiaca” que se mantiene y “Neumonía”, donde disminuye.

5.4 TIEMPOS DE EJECUCIÓN

Para la configuración de hiperparámetros descrita en el capítulo 3 y 1498 imágenes totales, los tiempos de ejecución durante el entrenamiento son de 70 minutos para el modelo ResNet y 5 minutos para el modelo CXR Foundation. Cabe destacar que el modelo CXR Foundation requiere una etapa previa al entrenamiento que consiste en generar los embeddings de las imágenes, que son la entrada al perceptrón multicapa. Este paso dura alrededor de 5 horas.

Capítulo 6

DISCUSIÓN

En este capítulo se analizan los resultados obtenidos en el capítulo anterior. Se interpretan las gráficas que monitorizan el proceso de aprendizaje durante el entrenamiento y se plantean hipótesis acerca de su comportamiento.

6.1 ANÁLISIS DEL ENTRENAMIENTO

ENFOQUE 1

Por lo que respecta al modelo CXR Foundation, durante el proceso de entrenamiento se observa un indicio de sobreajuste tal y como se puede observar en las figuras 5.1 y 5.2. Aunque la función de pérdidas disminuye constantemente tanto en el conjunto de entrenamiento como en el de validación, el valor del área bajo la curva en validación se estabiliza alrededor de la época 120, mientras que en el conjunto de entrenamiento siempre aumenta. Esto indica que el modelo no mejora su capacidad de predicción, lo que puede indicar un límite en la generalización. También cabe destacar que los valores obtenidos de la función de pérdidas son muy elevados para los valores obtenidos del área bajo la curva en validación. Son comportamientos que pueden ocurrir al usar redes preentrenadas en problemas complejos. En la ResNet, las figuras 5.5 y 5.6 permiten también evidenciar sobreajuste. En las primeras épocas se observa un valor en la función de pérdidas en el conjunto de validación de entorno a 120, y hace que no se pueda monitorizar correctamente el comportamiento durante el entrenamiento. El área bajo la curva aumenta siempre con el paso de las épocas para el conjunto de entrenamiento, mientras que en validación siempre se obtiene un resultado entre 0.5 y 0.6, lo que indica que la asignación en las predicciones es prácticamente aleatoria.

ENFOQUE 2

En el modelo CXR Foundation de este segundo enfoque, el comportamiento observado es similar al analizado en las figuras 5.1 y 5.2, con señales de sobreajuste. En la figura 5.9 la función de pérdidas disminuye en los conjuntos de entrenamiento y validación, pero el área bajo la curva de la figura 5.20 muestra que para el conjunto de validación se estabiliza alrededor de la época 300, mientras que en entrenamiento sigue aumentando. Esto muestra que el modelo no está mejorando su capacidad de generalización. También se observa que los valores obtenidos de la función de pérdidas son muy elevados para los valores obtenidos de la curva ROC en validación. Por otro lado, la ResNet muestra un comportamiento similar también al enfoque anterior, en las figuras 5.13 y 5.14 se observa cierto sobreajuste. Un valor de 60000 en la función de pérdidas para el conjunto de validación en la primera época impide que no se puede apreciar el comportamiento

durante el entrenamiento. El área bajo la curva para el conjunto de entrenamiento aumenta siempre con el paso de las épocas, mientras que en validación siempre se obtiene un resultado de entorno a 0.6, por lo que se puede decir que el modelo no consigue adquirir capacidad de generalización.

ENFOQUE 3

Finalmente, en este tercer enfoque que plantea tres problemas binarios, para el modelo CXR Foundation se observa el mismo comportamiento en los tres modelos que en los dos enfoques anteriores. Durante la monitorización del comportamiento de la red se observan valores altos de la función de pérdidas para el conjunto de entrenamiento y validación que siempre disminuyen con el paso de las épocas. Por otro lado, el área bajo la curva se estabiliza entorno a la época 100 para el primer y tercer modelo, y entorno a la época 300 para el segundo, mientras que en el conjunto de entrenamiento siempre aumenta este valor en los tres casos. Finalmente, para la ResNet ocurre también algo parecido que para los dos enfoques anteriores. En los modelos 1 y 2 la gráfica de la función de pérdidas muestra algunos valores atípicos que impiden interpretar el aprendizaje. Siempre disminuye con el paso de las épocas en el conjunto de entrenamiento, pero no ocurre lo mismo en el conjunto de validación. Se observan algunos picos atípicos del valor de la función de pérdidas, pero no tan drásticos como en los otros casos. Por otro lado, el valor del área bajo la curva siempre aumenta con el paso de las épocas en el caso del entrenamiento, pero también muestra un comportamiento prácticamente aleatorio en el caso de validación.

RESUMEN

Los análisis previamente comentados permiten extraer patrones de comportamiento de las arquitecturas consideradas sobre el conjunto de datos de PadChest. El comportamiento ideal es que la función de pérdidas disminuya con el paso de las épocas tanto en el conjunto de entrenamiento como en el de validación mostrando valores próximos a 0, y que la métrica seleccionada aumente también en el conjunto de entrenamiento y validación con el paso de las épocas mostrando valores similares entre ambos conjuntos. Este comportamiento ideal indica capacidad de generalización, de forma que, al predecir nuevos datos, la predicción sea correcta y representativa del comportamiento observado durante el entrenamiento.

Sin embargo, como se ha podido ver, no es lo que ha ocurrido en este caso. Y es que, se ha demostrado que utilizar redes preentrenadas sobre conjuntos de datos diferentes puede provocar comportamientos no deseados y difíciles de controlar e interpretar.

En el caso del modelo CXR Foundation, se observa que la función de pérdidas disminuye siempre con el paso de las épocas tanto en entrenamiento como en validación, sin embargo, los valores son muy lejanos a 0 en todos los enfoques, probablemente esto

ocurre debido a la inicialización de los pesos. Por otro lado, el área bajo la curva en el conjunto de entrenamiento siempre aumenta con el paso de las épocas mientras que en el conjunto de validación se termina estabilizando. El valor en el que se estabiliza depende la complejidad de la tarea a resolver. Se obtienen buenos resultados, pero nunca óptimos.

Aunque lo ideal es que converjan ambas gráficas, cuando se da este caso, en el que la función de pérdidas continúa disminuyendo, pero la métrica se estabiliza, prima la métrica sobre el valor de la función de pérdidas, ya que es nuestro objetivo a maximizar.

Por otro lado, la ResNet muestra comportamientos prácticamente aleatorios en todos los casos. En la mayoría de veces, se observan valores atípicos durante la monitorización de la función de pérdidas en el entrenamiento que impiden interpretar los resultados. También es un comportamiento totalmente indeseado que ocurre al utilizar este tipo de redes. Una posible explicación es que, como el comportamiento del área bajo la curva es prácticamente aleatorio, se inicialicen mal los pesos o los valores obtenidos de la época previa no sea representativos de la actual. En el caso del área bajo la curva, se observa como siempre aumenta en el conjunto de entrenamiento durante el paso de las épocas, pero siempre se mantiene próximo a 0.5 en el caso de validación, indicando un comportamiento prácticamente aleatorio. Esto implica una falta de capacidad de generalización total para este modelo sobre el conjunto de datos de PadChest en la situación actual de capacidad de procesamiento y almacenamiento de datos.

También se ha observado que la calidad del rendimiento se ve afectada por la complejidad del problema. Para un individuo no especializado, clasificar estas imágenes a simple vista es extremadamente difícil, por lo que, para un sistema informático, con los recursos disponibles utilizados lo es todavía más. Además, se ha observado que la calidad de los resultados varía en función de los síntomas que definen a cada enfermedad. Una enfermedad puede estar caracterizada por muchos síntomas, apareciendo algunos de ellos con poca frecuencia. También puede haber síntomas comunes entre diferentes enfermedades, lo que aumenta todavía más la complejidad del problema.

Por lo tanto, es importante destacar que tal y como muestra A. Sellergen en su artículo [1], el rendimiento de los modelos está fuertemente influenciado por la cantidad de imágenes utilizadas en el entrenamiento y a pesar de no obtenerse resultados óptimos, se encuentran dentro del rango esperado [19].

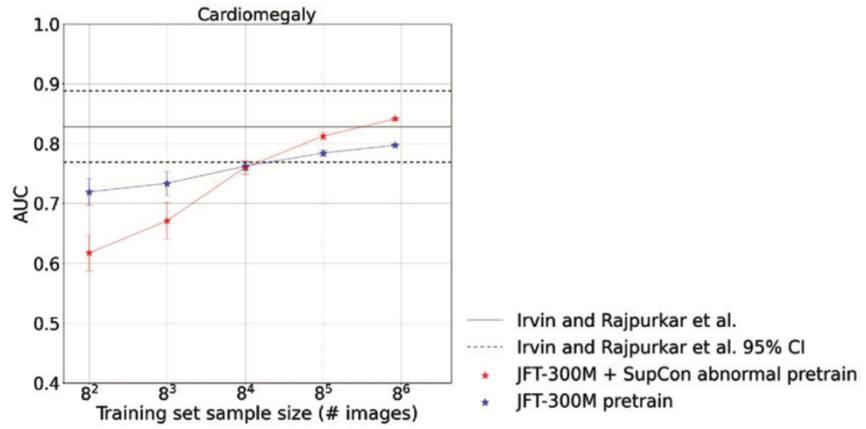


Figura 6.1: Gráfico que muestra la mejora en el rendimiento en el diagnóstico conforme aumenta el número de imágenes [1]

Capítulo 7

CONCLUSIONES

El objetivo de este proyecto era comparar diferentes arquitecturas utilizando la técnica de aprendizaje por transferencia y ver si aquella que incluye una etapa adicional de preentrenamiento específico sobre la tarea que se iba a resolver mejoraba el resultado final. Se ha demostrado que añadir este paso extra en la clasificación de enfermedades en imágenes de radiografías es crucial para obtener mejores resultados, mostrando el modelo CXR Foundation un rendimiento claramente superior a ResNet en todas las métricas consideradas.

Por otra parte, se ha experimentado que la selección y ajuste óptima de hiperparámetros es una de las partes más importantes y costosas para un buen rendimiento, pero para hacer más visible su efecto sobre los resultados es necesario contar con gran cantidad de imágenes para poder entrenar modelos robustos.

Finalmente, se ha visto que generar un modelo binario por cada clase y combinar los resultados con un ensemble que se encargue de realizar la predicción final en lugar de un modelo multiclas, mejora la precisión final de los resultados, aunque es computacionalmente más costoso.

7.1 TRABAJO FUTURO

Como se ha visto a lo largo del proyecto, el principal problema es la falta de recursos computacionales para poder almacenar y procesar más imágenes. Para resolver este problema, planeamos solicitar acceso a más recursos computacionales del Centro Nacional de Supercomputación. Esto nos permitiría trabajar con más imágenes y agregar más enfermedades a nuestro estudio, lo cual es crucial para mejorar la calidad de nuestros resultados. Con más capacidad computacional, podríamos entrenar modelos más efectivos y realizar comparaciones más detalladas y precisas.

Capítulo 8

Bibliografía

- [1] A. B. S. e. al., «Simplified Transfer Learning for Chest Radiography Models Using Less Data,» *Radiology*, 2022.
- [2] P. Rajpurkar, «Detection on Chest X-Rays with Deep Learning,» *arXiv [cs.CV]*, 2017.
- [3] X. Wang, «ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases,» *arXiv*, 2017.
- [4] J. S. Lisa Torrey, «Transfer Learning,» de *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, IGI global, 2010, pp. 242-264.
- [5] L. C. M. Félix, «Data mining: torturando a los datos hasta [*] que confiesen,» 2002. [En línea]. Available: <https://www.uoc.edu/pdf/web/esp/art/uoc/molina1102/molina1102.pdf>. (accedido 2 de junio de 2024)
- [6] C. S. G. L. Chin-Teng Lin, Neural Fuzzy Systems. A Neuro-Fuzzy Synergism to Intelligent Systems, Prentice-Hall, 1996.
- [7] X. B. Olabe, REDES NEURONALES ARTIFICIALES Y SUS APLICACIONES, 1998.
- [8] M. Fabien, «github,» [En línea]. Available: <https://maelfabien.github.io/deeplearning/Perceptron/#.> (accedido 5 de junio de 2024)
- [9] A. S. E. a. M. J. Serrano, Redes Neuronales Artificiales, 2009.
- [10] R. Lahoz-Beltrá, Bioinformática. Simulación, Vida Artificial e Inteligencia Artificial, Díaz de Santos SA, 2004.
- [11] SuperDataScience Team , «superdatascience.com,» 2018. [En línea]. Available: <https://www.superdatascience.com/blogs/the-ultimate-guide-to-convolutional-neural-networks-cnn.> (accedido 12 de junio de 2024)
- [12] A. Ng, *NIPS*, 2016.
- [13] Stanford Vision Lab, Stanford University, Princeton University , «image-net,» [En línea]. Available: <https://image-net.org/index.php>. (accedido 15 de junio de 2024)

- [14] A. Z. Karen Simonyan, «Very Deep Convolutional Networks for Large-Scale Image Recognition,» *arXiv:1409.1556 [cs.CV]*, 2014.
- [15] X. Z. S. R. J. S. Kaiming He, «Deep Residual Learning for Image Recognition,» *arXiv:1512.03385 [cs.CV]*, 2015.
- [16] P. S. a. P. F. Y. Bengio, «Learning long-term dependencies with gradient descent is difficult,» de *IEEE Transactions on Neural Networks*, 1994.
- [17] N. S. A. K. I. S. a. R. R. S. G. E. Hinton, «Improving neural networks by preventing coadaptation of feature detectors,» *arXiv:1207.0580*, 2012.
- [18] S. I. a. C. Szegedy, «Batch normalization: Accelerating deep network training by reducing internal covariate shift,» *ICML*, 2015.
- [19] C. Chen, «A deep learning-based algorithm for pulmonary tuberculosis detection in chest radiography,» *scientificreports*, 2024.
- [20] P. Khosla, «Supervised Contrastive Learning,» *arXiv:2004.11362 [cs.LG]*, 2020.
- [21] A. P. J.-M. S. M. d. I. I.-V. Aurelia Bustos, «PADCHEST: A LARGE CHEST X-RAY IMAGE DATASET WITH MULTI-LABEL ANNOTATED REPORTS,» *arXiv:1901.07441v1*, 2019.
- [22] I. G. B. G. Yang You, «Large Batch Training of Convolutional Networks,» *arXiv:1708.03888 [cs.CV]*, 2017.
- [23] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 1998.
- [24] J. Proakis, *Tratamiento Digital De Señales*, Prentice-Hall, 1997.
- [25] B. Kumar, «The Brief History of Convolutional Neural Networks,» 2021. [En línea]. Available: <https://medium.com/appyhigh-technology-blog/convolutional-neural-networks-a-brief-history-of-their-evolution-ee3405568597>. (accedido 12 de junio de 2024)

Capítulo 9

ANEXO

El código se puede encontrar en el siguiente enlace:

<https://github.com/cescdovi/ChestXRay>