

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
SANTIAGO - CHILE



“DIAGNÓSTICO INTELIGENTE DE COVID-19 MEDIANTE DEEP LEARNING APLICADO A IMÁGENES DE TOMOGRFÍA COMPUTARIZADA”

PABLO ALBERTO AGUIRRE MOREAU

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN INFORMÁTICA

Profesor Guía: Mauricio Solar
Profesor Correferente: Mauricio Araya

Enero - 2023

DEDICATORIA

A todos los que avanzan la ciencia día a día.

RESUMEN

Resumen— El COVID-19 ha causado estragos en los sistemas de salud de los países afectados, debido a su naturaleza altamente contagiosa los centros médicos no cuentan con la capacidad suficiente para atender a los pacientes con síntomas por lo que se vuelve deseable el uso de una herramienta que permita analizar rápidamente las imágenes de tomografía computarizada del tórax en busca de un diagnóstico. En este trabajo se propone una solución basada en redes neuronales que permita clasificar imágenes CT en COVID-19 o No COVID-19 según las lesiones presentes en ellas. Para esto se hizo un modelo compuesto por una red neuronal base y un componente adicional para proveer información detallada de las imágenes. Los principales resultados obtenidos son AUC 0,67, *recall* 0,65 y *specificity* 0,69. Los resultados indican que el modelo tiene cierta capacidad de discriminación entre COVID-19 y No COVID-19 pero no se compara con las propuestas del estado del arte, sin embargo, el potencial de mejora que tiene la solución propuesta es grande.

Palabras Clave— COVID-19, clasificación, tomografía computarizada, aprendizaje profundo, imágenes médicas

ABSTRACT

Abstract— COVID-19 has wreaked havoc in the health systems of the affected countries. Due to its highly contagious nature, medical centers do not have sufficient capacity to care for patients with symptoms, making the use a tool that allows rapid analysis of CT images of the thorax in search of a diagnosis desirable. In this work, a neural network-based solution is proposed to classify CT images into COVID-19 or Non-COVID-19 according to the lesions present in them. For this purpose, a model composed of a base neural network and an additional component to provide detailed information of the images was developed. The main results obtained are AUC 0,67, *recall* 0,65 and *specificity* 0,69. The results indicate that the model has some discrimination capability between COVID-19 and Non COVID-19 but does not compare with the state of the art proposals, however, the potential for improvement that the proposed solution has is vast.

Keywords— COVID-19, classification, computerized tomography, deep learning, medical images

GLOSARIO

AUC: *Area Under the Receiver Operating Characteristic Curve.*

CNN: *Convolutional Neural Network.*

CT: *Computed Tomography.*

DICOM: *Digital Imaging and Communications in Medicine.*

FPN: *Feature Pyramid Network.*

FPR: *False Positive Rate.*

GGO: *Ground-glass opacity.*

HU: *Hounsfield Units.*

MERS-CoV: *Middle East Respiratory Syndrome Coronavirus.*

MSE: *Mean Squared Error.*

NIFTI: *Neuroimaging Informatics Technology Initiative.*

NPV: *Negative Predictive Value.*

PCR: *Polymerase Chain Reaction.*

PPV: *Positive Predictive Value.*

RESNET: *Residual Neural Network.*

ROC: *Receiver Operating Characteristic.*

ROI: *Region of Interest.*

SARS-CoV-1: *Severe Acute Respiratory Syndrome Coronavirus 1.*

TPR: *True Positive Rate.*

ÍNDICE DE CONTENIDOS

RESUMEN	III
ABSTRACT	III
GLOSARIO	IV
ÍNDICE DE FIGURAS	VII
ÍNDICE DE TABLAS	VIII
INTRODUCCIÓN	1
CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA	2
1.1 OBJETIVOS	5
1.1.1 OBJETIVO GENERAL	5
1.1.2 OBJETIVOS ESPECÍFICOS	5
CAPÍTULO 2: MARCO CONCEPTUAL	6
2.1 IMÁGENES DE TOMOGRAFÍA COMPUTARIZADA	6
2.2 USO DE CT EN EL DIAGNÓSTICO DE COVID-19	7
2.2.1 HALLAZGOS NORMALES	7
2.2.2 ANORMALIDADES CON INCIDENCIA ALTA	8
2.2.3 ANORMALIDADES CON INCIDENCIA INTERMEDIA	9
2.2.4 ANORMALIDADES CON INCIDENCIA BAJA	10
2.2.5 EVOLUCIÓN TEMPORAL DE ANORMALIDADES EN CT DEL TÓRAX	11
2.3 <i>DEEP LEARNING</i> COMO HERRAMIENTA DE DIAGNÓSTICO	12
2.3.1 REDES NEURONALES ARTIFICIALES	12
2.3.2 REDES NEURONALES CONVOLUCIONALES	13
2.3.3 MÉTRICAS	15
CAPÍTULO 3: ESTADO DEL ARTE	19
3.1 PREPROCESAMIENTO DE IMÁGENES	19
3.2 MODELOS DE <i>DEEP LEARNING</i>	21
3.3 RESULTADOS	24
CAPÍTULO 4: PROPUESTA DE SOLUCIÓN	26
4.1 DATASET	27
4.2 AUTOENCODER	28
4.3 RESNET	31
4.4 MODELO PROPUESTO	32
CAPÍTULO 5: RESULTADOS	34

CAPÍTULO 6: CONCLUSIONES	39
REFERENCIAS BIBLIOGRÁFICAS	43
ANEXO A:	46

ÍNDICE DE FIGURAS

1.1	Ejemplo de tomografía computarizada de tórax.	3
1.2	Tomografía computarizada de tórax perteneciente a hombre de 59 años con PCR positivo. Las flechas rojas marcan lesiones típicas de neumonía causada por COVID-19.	4
1.3	Representación de objeto DICOM.	4
2.1	Imagen CT con presencia de GGO bilateral y agrandamiento vascular en un hombre de 70 años con PCR positivo.	8
2.2	Imagen CT con presencia de opacidad curvilínea subpleural y opacidad de vidrio esmerilado en un hombre de 63 años con PCR positivo.	10
2.3	Imagen CT con presencia de patrón <i>crazy-paving</i> en un hombre de 66 años con PCR positivo.	10
2.4	Imágenes CT con presencia de lesiones cavitadas en ambos lóbulos inferiores.	11
2.5	Red neuronal con tres <i>inputs</i> , dos capas ocultas de cuatro nodos cada una y un <i>output</i>	12
2.6	Red convolucional utilizada para el reconocimiento de dígitos manuscritos.	14
2.7	Ejemplo de convolución usando un filtro de $3 \times 3 \times 1$ sobre una imagen de dimensiones $5 \times 5 \times 1$	14
2.8	Matriz de confusión.	16
2.9	Ejemplo de curva ROC.	17
2.10	Ejemplo de cálculo de AUC.	18
3.1	Arquitectura para la extracción de ROI y modelo de <i>deep learning</i>	21
3.2	Arquitectura para extracción de características y predicción a nivel de corte	22
3.3	Arquitectura para extracción de características y predicción a nivel de CTs	23
4.1	Ejemplo de <i>autoencoder</i> para codificación del dataset MNIST	28
4.2	Arquitectura de <i>autoencoder</i> convolucional con capas 3D	29
4.3	Tomografía computarizada de tórax	30

4.4	Reconstrucción de una tomografía computarizada de tórax	30
4.5	Ejemplo de bloque residual	31
4.6	Extracto del modelo propuesto, donde se detalla la concatenación de las dos redes neuronales	33
5.1	Matriz de confusión obtenida al evaluar la propuesta en el conjunto de <i>test</i> .	34
5.2	Curva ROC obtenida al evaluar la propuesta en el conjunto de <i>test</i>	35
5.3	Matriz de confusión obtenida al evaluar una <i>ResNet-50</i> en el conjunto de <i>test</i>	37
A.1	Ejemplo de funciones diferenciables.	46
A.2	Ejemplo de función convexa y no convexa.	46
A.3	Visualización del gradiente de la función $f(x, y) = 0,5x^2 + y^2$	47
A.4	Ejemplo de arquitectura de una <i>ResNet-50</i> para imágenes bidimensionales .	49

ÍNDICE DE TABLAS

3.1	Rendimiento del modelo de <i>deep learning</i> en la tarea de distinguir entre neumonía viral típica y COVID-19.	24
3.2	Rendimiento del modelo de <i>deep learning</i> en la tarea de distinguir entre COVID-19 y sano.	24
3.3	Rendimiento del modelo de <i>deep learning</i> en la tarea de distinguir entre COVID-19, neumonía bacteriana y sano.	25
3.4	Rendimiento del modelo de <i>deep learning</i> en la tarea de distinguir entre COVID-19 y sano.	25
5.1	Rendimiento del modelo de <i>deep learning</i> propuesto en la tarea de distinguir entre COVID-19 y No COVID-19.	35
5.2	Rendimiento de la <i>ResNet-50</i> en la tarea de distinguir entre COVID-19 y No COVID-19.	37
5.3	Comparación del rendimiento entre la propuesta y la <i>ResNet-50</i> sobre el conjunto de <i>test</i> en la tarea de distinguir entre COVID-19 y No COVID-19. . . .	38

INTRODUCCIÓN

Estos últimos años, el COVID-19 ha causado estragos en la población mundial, llegando a colapsar los sistemas de salud de los países afectados debido al gran volumen de personas contagiadas previo al desarrollo de una vacuna. Para aliviar esta situación, se plantea el uso de inteligencia artificial para hacer más expedito el proceso de diagnóstico y consecuente tratamiento de los pacientes que presentan síntomas relacionados a esta enfermedad.

La propuesta de solución presentada en este documento consiste en un modelo de *deep learning* capaz de identificar imágenes de tomografía computarizada afectadas por COVID-19 sin requerir un trabajo extensivamente complejo para hacerlo funcionar. Para esto se desarrolla una arquitectura modular que trabaja con datos en 3D para aprovechar la información espacial y temporal provista por estos antes de emitir un veredicto.

Para esto, se define con mayor detalle el problema abordado, junto con los objetivos que se persiguen en este trabajo.

Se continua con el marco conceptual, repasando los conceptos que hacen posible esta tarea; abarcando desde definiciones propias del dominio del problema, la medicina, hasta las herramientas informáticas que son utilizadas junto con la explicación de su funcionamiento y el por qué son relevantes para la resolución de este tipo de problemas.

Más adelante, se da a conocer en detalle algunas propuestas de solución, planteadas por investigadores alrededor del mundo, que han obtenido buenos resultados y son un referente en este momento para resolver el problema de diagnosticar COVID-19 en imágenes de tomografía computarizada del tórax. Se detalla el paso a paso seguido por estas soluciones, desde el trabajo previo aplicado a las imágenes hasta los ajustes realizados a los modelos de redes neuronales y las métricas obtenidas.

Después, se plantea la propuesta de solución en la que se basa este documento, la motivación detrás de ella, los datos utilizados y el detalle de la arquitectura de *deep learning*.

A continuación, se muestran los resultados obtenidos al evaluar el modelo propuesto, un breve análisis de estos y cómo se compara con aquellas propuestas del estado del arte.

Finalmente, se concluye el trabajo realizado, donde se da a conocer el aprendizaje obtenido durante esta experiencia y las recomendaciones a seguir en un trabajo futuro.

CAPÍTULO 1

DEFINICIÓN DEL PROBLEMA

Los coronavirus son una familia de virus que pueden causar afecciones que varían desde el resfriado común hasta problemas respiratorios severos. Se pueden contagiar entre animales y personas (transmisión zoonótica).

El COVID-19 es un nuevo tipo de coronavirus denominado SARS-CoV-2. La Organización Mundial de la Salud dio cuenta de este virus el 31 de diciembre de 2019, luego de un brote de neumonía viral en Wuhan, China; varios investigadores creen que se originó en murciélagos, pero todavía desconocen cómo se transmitió al ser humano [Mallapaty, 2020]. La propagación de este virus se da a través del contacto directo e indirecto con personas infectadas, mediante la saliva, la tos o el estornudo. Actualmente existen más de 80 millones de casos confirmados en el mundo y más de 1.8 millones de muertes [WHO, 2020], números que crecen día a día debido a la rápida transmisión del virus.

Los síntomas más comunes que presentan aquellos contagiados con el COVID-19 son:

- Fiebre.
- Tos.
- Fatiga.

También pueden presentarse otros síntomas como:

- Pérdida del olfato o el gusto.
- Congestión nasal.
- Dolor de cabeza.
- Dolor muscular.
- Nausea.
- Pérdida del apetito.
- Confusión.

El método más utilizado para diagnosticar el COVID-19 es el PCR o Polymerase Chain Reaction, obteniendo resultados entre 24 y 72 horas después de tomado el test [Redsalud, 2020]. Sin embargo, se ha reportado que la sensibilidad del PCR puede no ser lo suficientemente

alta para la detección y tratamiento temprano de los presuntos contagiados [Ai *et al.*, 2020, Fang *et al.*, 2020] . También se utilizan test rápidos como el Test Rápido de Antígenos, que entrega un resultado después de 30 min de haber sido tomado, pero con sensibilidad menor que la del PCR.

Como alternativa de diagnóstico, se ha explorado la idea de usar imágenes de tomografía computarizada. La tomografía computarizada (CT) es un examen que combina una serie de radiografías que se toman desde distintos ángulos alrededor del cuerpo y luego son procesadas para crear imágenes transversales de los huesos y tejidos blandos, entre otros (Figura 1.1).

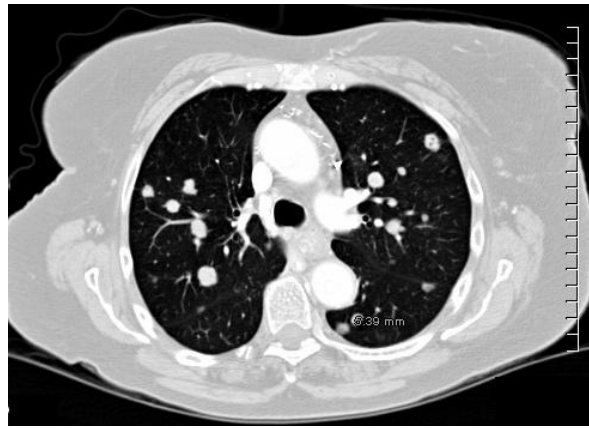


Figura 1.1: Ejemplo de tomografía computarizada de tórax.
Fuente: St. Elizabeth's Medical Center.

Las imágenes CT pueden evidenciar ciertas manifestaciones características en el pulmón asociadas al COVID-19 (Figura 1.2). Por lo tanto pueden servir como un método de diagnóstico temprano. Sin embargo, muchos de los estudios realizados están limitados por sesgos de selección y posibles confusiones debido a la presencia simultánea de otras enfermedades pulmonares; además, la prevalencia de anomalías relacionadas al COVID-19 depende de la etapa y la severidad en la que se encuentre la enfermedad [Kwee y Kwee, 2020].

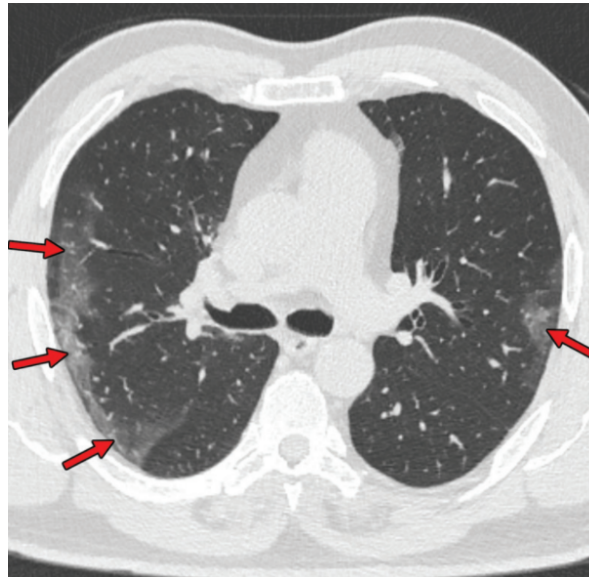


Figura 1.2: Tomografía computarizada de tórax perteneciente a hombre de 59 años con PCR positivo. Las flechas rojas marcan lesiones típicas de neumonía causada por COVID-19.

Fuente: Radiological Society of North America [Kwee y Kwee, 2020].

Recientemente, se han hecho avances en el área de las imágenes médicas aplicando técnicas de deep learning, por lo que se pueden explotar sus capacidades de generalización y extracción de características para obtener diagnósticos rápidos y confiables. Siendo esto último de suma importancia debido a la rápida propagación del virus.

Las imágenes CT se encuentran mayormente en formato DICOM. DICOM es un estándar internacional para transmitir, almacenar, recuperar, procesar y presentar información de imágenes médicas. Consiste en una serie de objetos con sus respectivos atributos [Pianyk, 2009]. Entre ellos se encuentran los parámetros de los equipos que se utilizaron para capturar las imágenes médicas, las imágenes tomadas y la información del paciente (Figura 1.3). Por lo tanto, se requiere un procesamiento previo para separar las imágenes de los diagnósticos.

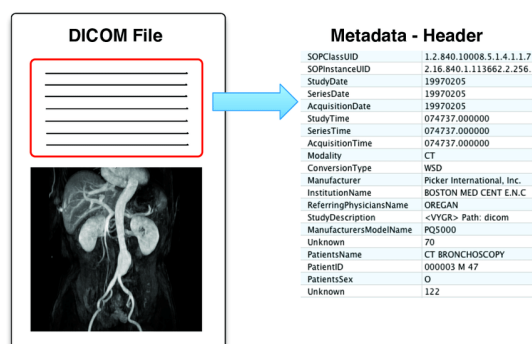


Figura 1.3: Representación de objeto DICOM.

Fuente: Researchgate [Bastião Silva, 2014].

1.1. OBJETIVOS

1.1.1. OBJETIVO GENERAL

Desarrollar un clasificador de COVID-19 basado en redes convolucionales para imágenes de tomografía computarizada del tórax.

1.1.2. OBJETIVOS ESPECÍFICOS

- Preparar un conjunto de imágenes adecuado para resolver el problema de detectar el COVID-19.
- Definir y entrenar un modelo capaz de detectar COVID-19 en imágenes de tomografía computarizada.
- Comparar el desempeño con otros métodos o propuestas similares.

CAPÍTULO 2

MARCO CONCEPTUAL

2.1. IMÁGENES DE TOMOGRAFÍA COMPUTARIZADA

La tomografía computarizada funciona de forma similar a los rayos X. Distintas partes del cuerpo absorben diferentes cantidades de radiación, por lo que se genera un contraste apreciable en las imágenes CT.

El procedimiento consiste en una serie de haces de rayos X y detectores electrónicos que rotan alrededor del paciente. Estos detectores miden la cantidad de radiación que es absorbida por todo el cuerpo. Luego, un software especial procesa el gran volumen de datos generado para crear imágenes bidimensionales transversales del cuerpo.

Avances en la tecnología han permitido a los escáneres CT obtener múltiples cortes del cuerpo en una sola rotación. Además, los cortes pueden ser más finos, resultando en mayor detalle y en capacidades adicionales de visualización.

Existen técnicas para reducir la cantidad de radiación a la que se expone el paciente durante el procedimiento [RadiologyInfo, 2018]. Estas consisten en:

- Ajustar la dosis de radiación de forma continua, de acuerdo al tamaño de la zona, a medida que el paciente se mueve por el escáner.
- Uso de software para filtrar el ruido en los datos.
- Colocar escudos hechos de bismuto sobre el paciente.
- Apagar el tubo de rayos X durante partes de la rotación.
- Reducir el voltaje máximo.

2.2. USO DE CT EN EL DIAGNÓSTICO DE COVID-19

De acuerdo con la *Fleischner Society*, las imágenes CT no están indicadas como pruebas de diagnóstico para el COVID-19 en pacientes asintomáticos o con síntomas respiratorios leves [Rubin *et al.*, 2020].

Sin embargo, recomienda el uso de imágenes médicas:

- Para establecer un estado pulmonar basal.
- Para identificar anomalías cardiopulmonares subyacentes que pueden facilitar el empeoramiento clínico en pacientes con síntomas leves de COVID-19.
- En pacientes con síntomas moderados a graves de COVID-19.

Últimamente se han publicado varios estudios que reportan hallazgos de COVID-19 en CT del tórax [Adams *et al.*, 2020]. Los hallazgos histológicos pulmonares de COVID-19 se caracterizan por daño alveolar difuso y se asemejan a aquellos observados en otros coronavirus, como el SARS-CoV-1 y el MERS-CoV [Schaller *et al.*, 2020].

La prevalencia de anomalías relacionadas al COVID-19 en imágenes CT depende de la etapa en la que se encuentre y la severidad, pudiendo separarse en las siguientes categorías [Kwee y Kwee, 2020].

2.2.1. HALLAZGOS NORMALES

La incidencia de hallazgos normales en pacientes sintomáticos es de 10,6% aproximadamente [Adams *et al.*, 2020]. Comúnmente se visualizan durante los primeros 4 a 5 días de aparecidos los síntomas, sin embargo, una cantidad no despreciable de pacientes presenta estos hallazgos en etapas tardías de la infección [Ding *et al.*, 2020, Bernheim *et al.*, 2020]. Para el caso de los pacientes asintomáticos, la incidencia de hallazgos normales es considerablemente alta (alrededor del 46% de los pacientes) [Inui *et al.*, 2020].

2.2.2. ANORMALIDADES CON INCIDENCIA ALTA

Según [Adams *et al.*, 2020], en más del 70 % de casos de COVID-19 con PCR positivo se han encontrado anomalías como:

- Opacidad de vidrio esmerilado (GGO) (Figura 2.1).
- Agrandamiento vascular (Figura 2.1).
- Anomalías bilaterales.
- Afectación del lóbulo inferior.

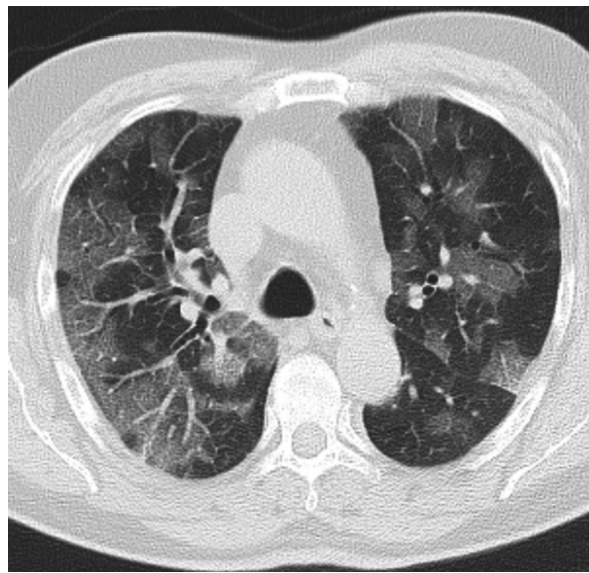


Figura 2.1: Imagen CT con presencia de GGO bilateral y agrandamiento vascular en un hombre de 70 años con PCR positivo.

Fuente: Radiological Society of North America [Kwee y Kwee, 2020].

La presencia de estos hallazgos debería ser suficiente para sospechar de un posible diagnóstico de COVID-19.

2.2.3. ANORMALIDADES CON INCIDENCIA INTERMEDIA

Entre el 10 % y el 70 % de los casos con COVID-19 positivos han presentado los siguientes hallazgos [Adams *et al.*, 2020]:

- Consolidación¹.
- Opacidad lineal (Figura 2.2).
- Engrosamiento del tabique y/o reticulación.
- Patrones del tipo *crazy-paving*² (Figura 2.3).
- Broncograma aéreo³.
- Engrosamiento pleural.
- Signo de halo⁴.
- Bronchiectasis⁵.
- Engrosamiento de la pared bronquial.
- Signo de halo invertido⁶.

También se ha reportado la siguiente distribución de lesiones: unilateral, multifocal, difusa, singular y/o focal, afectación del lóbulo medio o superior, ubicación periférica y ubicación central y periférica [Adams *et al.*, 2020].

¹Ocurre cuando el aire que normalmente llena las cavidades pequeñas de los pulmones es reemplazado por algo más, pudiendo ser pus, sangre, agua, restos de células o contenidos estomacales.

²Es un patrón lineal superpuesto en una opacidad de vidrio esmerilado. Se parece al patrón que forman los adoquines en las calles.

³Se refiere al fenómeno de bronquios llenos de aire rodeados por alvéolos llenos de pus u otros materiales

⁴Corresponde a una opacidad de vidrio esmerilado que rodea a un nódulo pulmonar.

⁵Es una enfermedad en la cual existe un agrandamiento permanente de partes de las vías respiratorias del pulmón

⁶Corresponde a una opacidad de vidrio esmerilado rodeada por un anillo de consolidación.

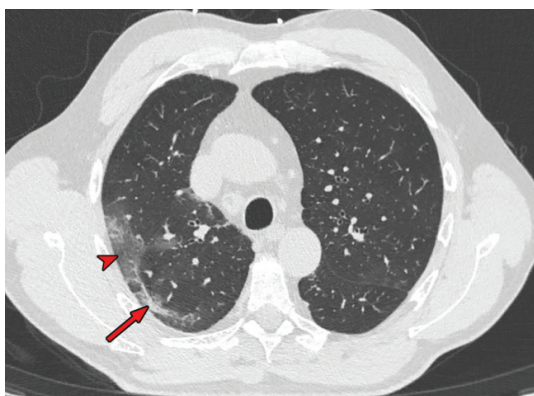


Figura 2.2: Imagen CT con presencia de opacidad curvilínea subpleural y opacidad de vidrio esmerilado en un hombre de 63 años con PCR positivo.

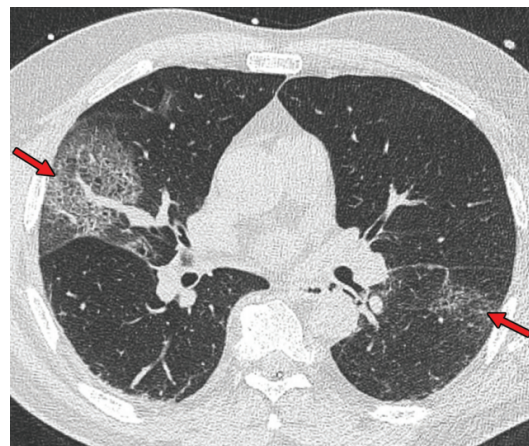


Figura 2.3: Imagen CT con presencia de patrón *crazy-paving* en un hombre de 66 años con PCR positivo.

Fuente: Radiological Society of North America [Kwee y Kwee, 2020]

2.2.4. ANORMALIDADES CON INCIDENCIA BAJA

Varios hallazgos han sido reportados como poco comunes en casos con PCR positivo [Adams *et al.*, 2020]. Estos incluyen:

- Derrame pleural.
- Linfadenopatía⁷.
- Signo *tree-in-bud*⁸.
- Distribución de lesión central.
- Derrame pericárdico.
- Lesiones pulmonares cavitadas (Figura 2.4).

La observación aislada de uno o más de estos hallazgos puede implicar un diagnóstico distinto al COVID-19, aunque no se puede descartar completamente [Adams *et al.*, 2020].

⁷Es una condición de los ganglios linfáticos en la que son anormales en tamaño o consistencia.

⁸Describe la aparición de múltiples áreas de nódulos centrolobulillares con un patrón de ramificación lineal.

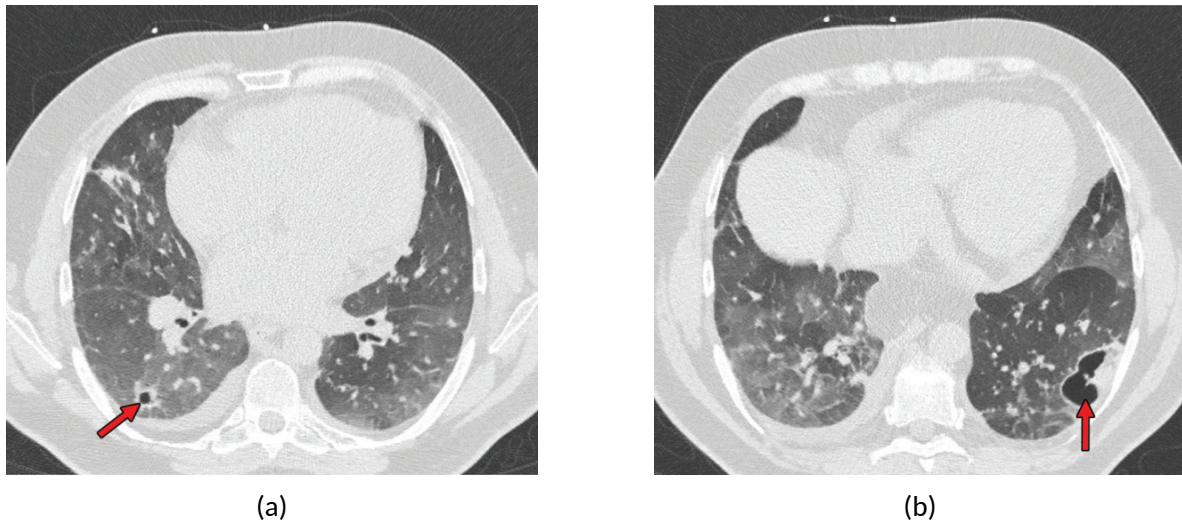


Figura 2.4: Imágenes CT con presencia de lesiones cavitadas en ambos lóbulos inferiores.
Fuente: Radiological Society of North America [Kwee y Kwee, 2020]

2.2.5. EVOLUCIÓN TEMPORAL DE ANORMALIDADES EN CT DEL TÓRAX

Conocer la forma en que evolucionan con el tiempo las anomalías es clave para determinar la etapa en la que se encuentra la infección y distinguirla de otras complicaciones potenciales.

Se han definido cuatro etapas del COVID-19 en CT del tórax [Kwee y Kwee, 2020]:

- **Etapa temprana:** Entre 0 y 5 días desde la aparición de los síntomas. Se caracteriza por presentar hallazgos normales u opacidades de vidrio esmerilado.
- **Etapa progresiva:** Entre 5 y 8 días desde la aparición de los síntomas. Se caracteriza por el incremento de las opacidades de vidrio esmerilado y aparición de patrones *crazy-paving*.
- **Etapa *peak*:** Entre 9 y 13 días desde la aparición de los síntomas. Se caracteriza por la presencia de consolidación progresiva.
- **Etapa tardía:** Más de 14 días desde la aparición de los síntomas. Se caracteriza por la disminución gradual de la consolidación y las opacidades de vidrio esmerilado, mientras que pueden manifestarse signos de fibrosis.

2.3. DEEP LEARNING COMO HERRAMIENTA DE DIAGNÓSTICO

Debido al avance tecnológico de los dispositivos de medición, se ha hecho posible obtener un gran volumen de imágenes médicas de alta resolución. Este rápido crecimiento requiere de un mayor esfuerzo por parte de los profesionales de la salud al momento de analizar y realizar diagnósticos, los cuales resultan ser subjetivos, propensos a errores y pueden variar de un profesional a otro.

Como alternativa se ha planteado el uso de técnicas de *machine learning* para automatizar el proceso de diagnóstico, sin embargo, existen grandes variaciones entre pacientes y estas técnicas dependen en gran medida de características definidas por expertos, por lo que no son suficientes para tratar con problemas complejos. El siguiente paso consistió en dejar que los computadores sean los que aprendan a identificar las características que representan de mejor forma los datos para el problema a resolver. Este concepto es la base de los algoritmos de *deep learning*, los cuales son modelos compuestos por muchas capas que se pueden usar, por ejemplo, para realizar una transformación a un *input* (en este caso imágenes), obtener un *output* que indique presencia o ausencia de enfermedades y en el proceso, aprender características relevantes de los datos que ayuden a resolver el problema.

2.3.1. REDES NEURONALES ARTIFICIALES

Las redes neuronales, en su forma más simple, consisten de miles o millones de nodos de procesamiento que se encuentran densamente interconectados. Estos nodos o neuronas se encuentran organizados en capas unidireccionales, a través de las cuales se propagan los datos durante el entrenamiento y cada uno de ellos puede estar conectado con varios nodos ubicados en una capa anterior, de los cuales recibe los datos, y varios nodos en una capa posterior a los que les transmite los datos después de ser procesados (Figura 2.5).

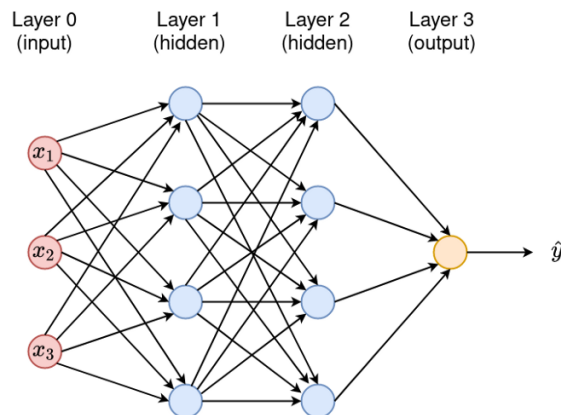


Figura 2.5: Red neuronal con tres *inputs*, dos capas ocultas de cuatro nodos cada una y un *output*.

Fuente: Towards Data Science [Koech, 2018].

Cada una de estas conexiones entre nodos tiene asociado un peso o *weight*. Durante el entrenamiento, los nodos reciben los datos en forma de números, los multiplican por estos pesos, suman todos los valores resultantes para obtener un único valor y si supera un umbral, lo transmiten a los nodos posteriores.

Para una determinada capa l , este proceso se resume en la ecuación 2.1

$$\begin{aligned} z^{(l)} &= W^{(l)} \cdot a^{(l-1)} + b^{(l)} \\ a^{(l)} &= \sigma(z^{(l)}) \end{aligned} \tag{2.1}$$

Donde $W^{(l)}$ corresponde a la matriz de pesos asociados a los nodos de la capa l , $a^{(l-1)}$ corresponde a las activaciones de los nodos de la capa anterior (o los datos de entrada en caso de ser la primera capa), $a^{(l)}$ corresponde a las activaciones de la capa actual, $b^{(l)}$ corresponde al sesgo o *bias* de los nodos y σ corresponde a la función de activación que actúa como el umbral que determina si se propagan los valores a las capas posteriores.

En una red neuronal, cambiar los pesos de cualquier conexión tiene un efecto en cascada sobre el resto de los nodos y sus activaciones en las capas subsecuentes. Por lo tanto, el proceso de aprendizaje que caracteriza a las redes neuronales ocurre, dado un conjunto de *inputs* y *outputs*, al obtener los pesos que minimizan una función de costo según un algoritmo de optimización, donde la función de costo corresponde a una aproximación de que tan equivocadas son las predicciones respecto a los valores esperados.

Este proceso de optimización de los pesos se conoce como *backpropagation*, el cual permite calcular el error atribuible a cada nodo y por ende, las derivadas parciales que conforman el gradiente que se utiliza en algoritmos de optimización como gradiente descendente (ver en anexos). Cada elemento del gradiente indica cuánto cambiaría la función de costo si es que se hace un cambio sobre ese parámetro.

2.3.2. REDES NEURONALES CONVOLUCIONALES

Las redes neuronales convolucionales (abreviadas como CNN por su nombre en inglés) son algoritmos de *deep learning* compuestos por capas convolucionales que toman como *input* una imagen, aprenden la relevancia de distintos aspectos u objetos dentro de la imagen y son capaces de diferenciar entre ellos.

Las imágenes no son más que matrices donde cada valor está asociado a un píxel, pudiendo llegar a tener millones de valores. Por lo tanto, utilizar redes neuronales con capas densas no es eficiente, ya que el número de conexiones entre nodos aumenta considerablemente, haciendo que el aprendizaje sea lento y costoso. Una red convolucional es capaz de capturar relaciones espaciales y temporales dentro de las imágenes de forma eficiente debido a

la reducción en el número de parámetros involucrados durante el aprendizaje y la reutilización de los pesos de los nodos. Las primeras capas convolucionales se encargan de extraer características globales de las imágenes, mientras que las capas profundas son capaces de obtener detalles finos (Figura 2.6).

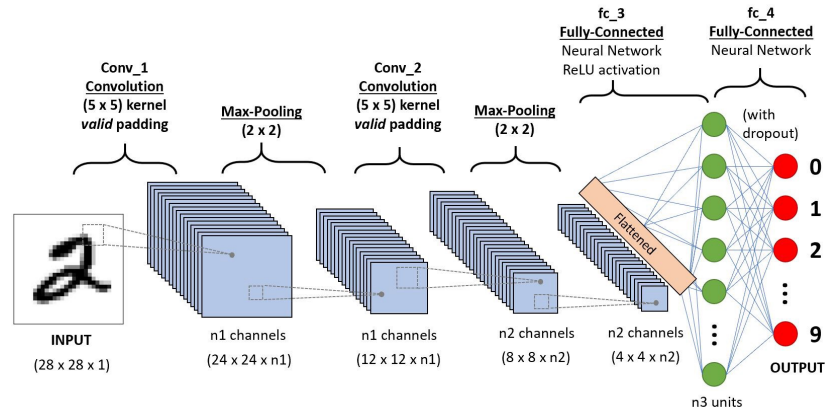


Figura 2.6: Red convolucional utilizada para el reconocimiento de dígitos manuscritos.
Fuente: Towards Data Science [Saha, 2022].

Las capas convolucionales extraen características de alto nivel de las imágenes mediante la aplicación de convoluciones por parte de elementos denominados filtros o *kernels*. Los filtros son matrices que realizan convoluciones sobre segmentos de las imágenes, desplazándose una cierta cantidad de píxeles en cada paso determinada por un parámetro *stride* hasta que recorre toda la imagen (Figura 2.7).

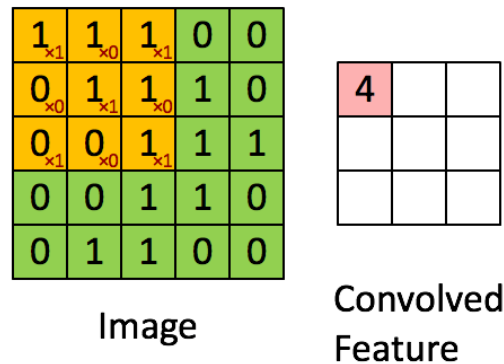


Figura 2.7: Ejemplo de convolución usando un filtro de $3 \times 3 \times 1$ sobre una imagen de dimensiones $5 \times 5 \times 1$.

Fuente: Towards Data Science [Saha, 2022].

Existen dos tipos de resultados después de realizar la convolución: en uno de ellos los datos resultantes tienen dimensión menor en comparación con el *input*, mientras que en el otro, la dimensión es igual o mayor. Esto se puede controlar mediante la aplicación de *padding*

sobre los datos antes de hacer las convoluciones. El *padding* consiste en rellenar con valores, generalmente ceros, los contornos de las imágenes para aumentar las dimensiones de estas. Existen dos tipos:

- **Valid Padding:** No se hace *padding* sobre la imagen, obteniendo *outputs* con dimensiones más pequeñas.
- **Same Padding:** Se hace *padding* de modo tal que el resultado de la convolución tenga las mismas dimensiones que el *input*.

2.3.3. MÉTRICAS

Las métricas son importantes para evaluar el desempeño de los modelos de predicción, ya que permiten entender qué tanto se acercan al comportamiento deseado y entregan un marco de comparación con otros modelos.

Existe una gran variedad de métricas, algunas son comunes para todos los problemas y entregan una visión general del rendimiento, mientras que otras se ajustan mejor a determinados problemas.

Para problemas de clasificación es habitual usar las siguientes métricas:

- **Accuracy**
- **Precision**
- **Recall o Sensitivity**
- **Specificity:**
- **Negative Predictive Value**
- **AUC**
- **F1-score**

Las métricas mencionadas anteriormente se pueden derivar de la matriz de confusión, siendo ésta una matriz que permite visualizar el rendimiento de un modelo predictivo (Figura 2.8).

		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

Figura 2.8: Matriz de confusión.
Fuente: Towards Data Science [Shung, 2018].

El **accuracy** representa la proporción entre las predicciones correctas y las predicciones totales. Puede no ser una buena métrica si el conjunto de datos no se encuentra balanceado, es decir, si la cantidad de ejemplos de cada clase es muy distinta. Sus valores se encuentran en el rango entre 0 y 1, pudiendo expresarse también como un porcentaje. Se calcula según se muestra en la ecuación 2.2

$$Accuracy = \frac{TN + TP}{TN + FP + TP + FN} \quad (2.2)$$

La **precision** representa la proporción de predicciones positivas que fueron clasificadas correctamente (por ejemplo, emails que fueron clasificados como spam y realmente son spam). Sus valores se encuentran en el rango entre 0 y 1, donde los valores cercanos a 1 indican que la cantidad de falsos positivos es muy pequeña en comparación con los verdaderos positivos. Se calcula según se muestra en la ecuación 2.3

$$Precision = \frac{TP}{TP + FP} \quad (2.3)$$

El **recall** mide la proporción de casos positivos que fueron clasificados correctamente (por ejemplo, emails que son spam y fueron clasificados como spam). Sus valores se encuentran en el rango entre 0 y 1. Valores cercanos a 1 indican que existen pocos casos positivos que fueron mal clasificados. Se calcula según se muestra en la ecuación 2.4

$$Recall = \frac{TP}{TP + FN} \quad (2.4)$$

La **specificity** mide la proporción de casos negativos que fueron clasificados correctamente (por ejemplo, emails que no son spam y fueron clasificados como no spam). Sus valores se encuentran en el rango entre 0 y 1. Valores cercanos a 1 indican que existen pocos casos

negativos que fueron clasificados erróneamente. Se calcula según se muestra en la ecuación 2.5

$$Specificity = \frac{TN}{TN + FP} \quad (2.5)$$

El **negative predictive value** representa la proporción de predicciones negativas que fueron clasificadas correctamente (por ejemplo, emails que fueron clasificados como no spam y realmente no son spam). Sus valores se encuentran en el rango entre 0 y 1, donde los valores cercanos a 1 indican que la cantidad de falsos negativos es muy pequeña en comparación con los verdaderos negativos. Se calcula según se muestra en la ecuación 2.6

$$NPV = \frac{TN}{TN + FN} \quad (2.6)$$

El **AUC** corresponde al valor del área que se encuentra debajo de la curva ROC. La curva **ROC** es un gráfico que muestra el rendimiento de un modelo de clasificación a través de la comparación entre la tasa de verdaderos positivos (TPR) y la tasa de falsos positivos (FPR) a lo largo de todos los umbrales de clasificación (Figura 2.9).

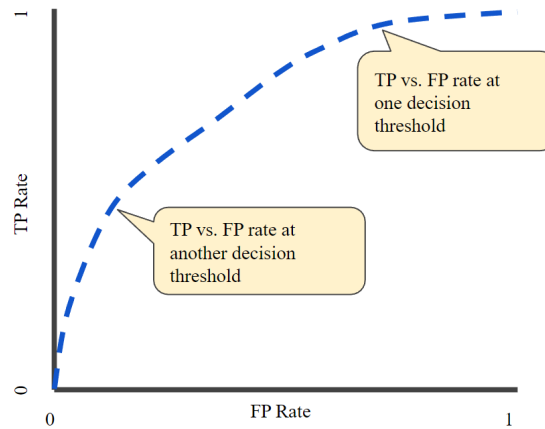


Figura 2.9: Ejemplo de curva ROC.
Fuente: Google [Google, 2022].

La tasa de verdaderos positivos se conoce como **recall**, mientras que la tasa de falsos positivos se calcula según se muestra en la ecuación 2.7

$$FPR = \frac{FP}{FP + TN} \quad (2.7)$$

Por lo tanto, el **AUC** se puede interpretar como la capacidad del modelo predictivo de diferenciar correctamente entre un caso positivo y un caso negativo. Sus valores se encuentran entre 0 y 1, donde aquellos cercanos a 1 indican una mayor capacidad de diferenciación (Figura 2.10).

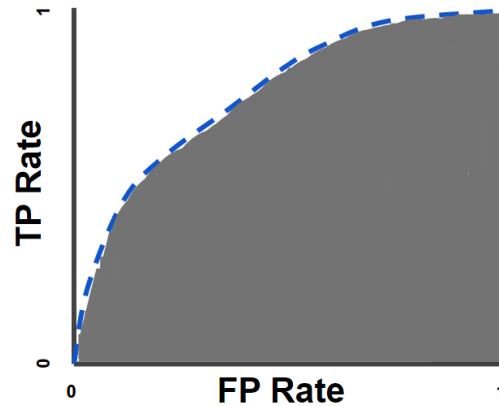


Figura 2.10: Ejemplo de cálculo de AUC.
Fuente: Google [Google, 2022].

El **F1-score** corresponde a la media armónica entre la precisión y el *recall*. Sus valores se encuentran en el rango entre 0 y 1, donde los valores cercanos a 1 indican que la precisión y el *recall* son muy buenos. Se calcula según se muestra en la ecuación 2.8

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (2.8)$$

CAPÍTULO 3

ESTADO DEL ARTE

Debido a la rápida y constante investigación sobre el tema, el estado de arte ha cambiado considerablemente. En general, existen 2 formas de abordar el problema del diagnóstico de COVID-19 usando CTs.

El primero consiste en trabajar con cortes de las imágenes CT debido a que se requiere menos capacidad de cómputo y tiempo para procesarlos. Destacan las propuestas de [Wang *et al.*, 2021] y [Song *et al.*, 2021]. Por otra parte, se toma el enfoque de trabajar con las imágenes CT completas, es decir, trabajar con datos volumétricos. Una ventaja de esto es que se añade una dimensión extra de información, sin embargo, la magnitud de los datos es considerablemente mayor que trabajar solo con los cortes por lo que se requiere de un procesamiento extra. Destaca la propuesta de [Subramaniyan *et al.*, 2021].

3.1. PREPROCESAMIENTO DE IMÁGENES

Para la propuesta de [Wang *et al.*, 2021], se recolectan 1065 imágenes CT correspondientes a 259 pacientes, de los cuales, 180 son casos de neumonía viral típica y 79 son casos confirmados de COVID-19. A estas imágenes se les delinean manualmente las regiones de interés para obtener los datos con los que se alimenta el modelo de *deep learning*.

Las ROI pueden ser sombras irregulares y cambios intersticiales en etapas tempranas, GGO e infiltraciones en ambos pulmones en etapa de progresión, pseudo cavitaciones, ganglios linfáticos agrandados y GGO multifocal en casos de neumonía viral típica. Estas son delineadas según los siguientes pasos:

- Convertir la imagen a escala de grises.
- Binarización de la escala de grises. Se selecciona como umbral de binarización la mínima frecuencia de todos los histogramas de frecuencia de píxeles de color gris.
- Relleno del área de fondo. Se usa el método *flood fill*⁹ para expandir la imagen en 1 píxel negro y los píxeles negros cerca del borde fueron convertidos a blancos.
- Se invierten los colores y se determina que los 2 contornos más grandes corresponden a los pulmones.
- El rectángulo delimitador más pequeño del área del pulmón se considera el marco de la ROI y la imagen original es recortada para obtener las imágenes finales.

⁹Algoritmo que determina y altera el área conectada a un nodo dado en un arreglo multidimensional con algún atributo coincidente.

Para mejorar la fiabilidad del modelo, las ROI son redimensionadas a un tamaño fijo de $299 \times 299 \times 3$ píxeles, copiando la misma imagen a lo largo de cada canal RGB, creando así una imagen RGB virtual. Son separadas en 3 conjuntos:

- **Entrenamiento:** 320 ROI extraídas de CTs pertenecientes al *Xi'an Jiaotong University First Affiliated Hospital*, de las cuales, 160 corresponden a neumonía viral típica y 160 corresponden a COVID-19.
- **Validación interna:** 455 ROI extraídas de CTs pertenecientes al *Xi'an Jiaotong University First Affiliated Hospital*, de las cuales, 360 corresponden a neumonía viral típica y 95 corresponden a COVID-19.
- **Validación externa:** 290 ROI extraídas de CTs pertenecientes al *Nanchang University First Hospital* y *Xi'an No.8 Hospital of Xi'an Medical College*, de las cuales, 220 corresponden a neumonía viral típica y 70 corresponden a COVID-19.

Por otro lado, para la propuesta de [Song et al., 2021], se recolectan CTs pertenecientes a 274 pacientes; de los cuales, 88 son casos de COVID-19 positivo, 100 son casos de neumonía bacteriana y 86 son pacientes sanos.

Al igual que el método descrito anteriormente, se siguen una serie de pasos para obtener los datos que son utilizados para el entrenamiento del modelo de *deep learning*, los que se detallan a continuación:

- Las imágenes CT pueden contener más de 200 cortes, por lo que los cortes cercanos entre sí tienden a ser similares. Para mitigar esto, solo se seleccionan 15 cortes representativos y equidistantes. Esto aumenta la velocidad de cómputo y reduce el impacto que produce la diferente cantidad de cortes en CTs de distintos hospitales
- Algunos cortes contienen pulmones incompletos, se utiliza la librería OpenCV para detectar estos casos y se descartan aquellos con regiones pulmonares que ocupan menos del 50 % del total de la imagen.
- Los contornos de los pulmones pueden variar sustancialmente persona a persona, pudiendo ocurrir que el modelo de *deep learning* se sobre ajuste a las características de dichos contornos, por lo tanto, se rellenan las áreas vacías con secciones de pulmón.

Finalmente, se obtienen 777 cortes de COVID-19 positivo pertenecientes a CTs del *Renmin Hospital of Wuhan University* y *Third Affiliated Hospital*; 505 cortes de neumonía bacteriana y 708 cortes de pacientes sanos, ambos pertenecientes a CTs del *Renmin Hospital of Wuhan University* y *Sun Yat-Sen Memorial Hospital*.

Se crean conjuntos de entrenamiento, validación y *test* siguiendo una separación del 60 %\30 %\10 % respectivamente.

La propuesta de [Subramaniyan *et al.*, 2021] no detalla el dataset utilizado, sin embargo es interesante lo que plantean y sirve como punto de comparación con los métodos anteriores.

Dado que se trabaja con datos volumétricos, se sigue un proceso especial para preparar las imágenes para el entrenamiento del modelo. Los pasos se detallan a continuación:

- Para empezar, se hace un remuestreo, redimensionado y normalización de las CTs.
- Las CTs resultantes son muestreadas, segmentadas usando *fuzzy c means*¹⁰ y se extraen características con el método *gray-level co-occurrence matrix*¹¹ para obtener regiones del pulmón con anomalías. Para evitar cambios morfológicos debido al muestreo, se usa el mismo factor de escalado en todas las dimensiones y se hace un *padding*¹² de ceros para obtener las dimensiones finales ($138 \times 256 \times 256$).
- Se seleccionan ventanas dentro de las CTs, las cuales son escaladas para aumentar el contraste y se extraen para obtener sub-volúmenes con las dimensiones de la ventana.
- Se normalizan los volúmenes resultantes.

3.2. MODELOS DE DEEP LEARNING

La propuesta de [Wang *et al.*, 2021] consiste en una arquitectura compuesta por 3 procesos principales: preprocesamiento de CTs; *feature extraction* de las ROI y entrenamiento; clasificación con dos capas densas y predicción usando clasificadores binarios (Figura 3.1).

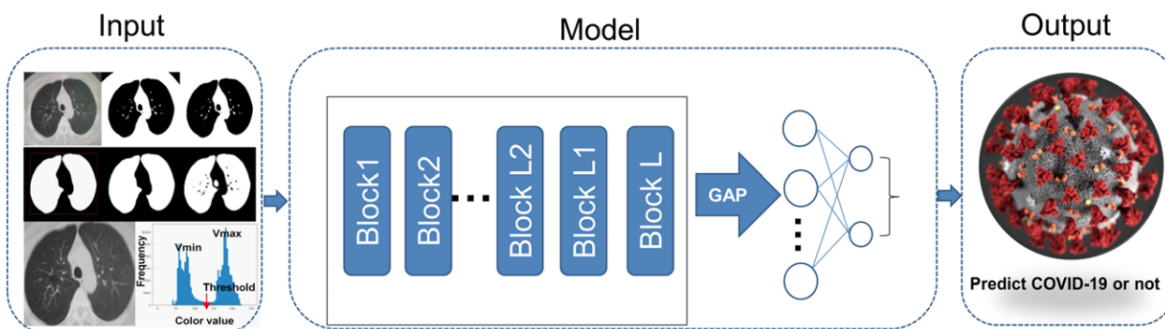


Figura 3.1: Arquitectura para la extracción de ROI y modelo de *deep learning*

Fuente: Springer [Wang *et al.*, 2021].

Se realiza *transfer learning* utilizando la CNN *Inception v3*, la que se entrena previamente con 1,2 millones de imágenes del dataset *ImageNet*. La red neuronal resultante se divide

¹⁰Forma de *clustering* en la que cada dato puede pertenecer a más de un *cluster*.

¹¹Matriz que se define sobre una imagen para ser la distribución de valores de píxeles concurrentes (valores de escala de grises o colores) en un *offset* dado.

¹²Consiste en agregar píxeles extras alrededor de una imagen.

en dos partes: la primera parte consiste en una red *Inception* modificada (*M-Inception*) y pre-entrenada para convertir las imágenes en vectores unidimensionales de características, mientras que la segunda parte consiste en una red neuronal densa para la clasificación.

La diferencia entre la *Inception* tradicional y la *M-Inception* yace en la última capa densa, la cual se modifica para reducir las dimensiones de los *features*.

Por otro lado, la propuesta de [Song et al., 2021] consiste en una arquitectura compuesta por 3 pasos: preprocesamiento de CTs; obtención de predicciones a nivel de corte usando la red *Details Relation Extraction* (*DRENet*) y agregación de las predicciones para obtener diagnósticos a nivel de paciente (Figura 3.2).

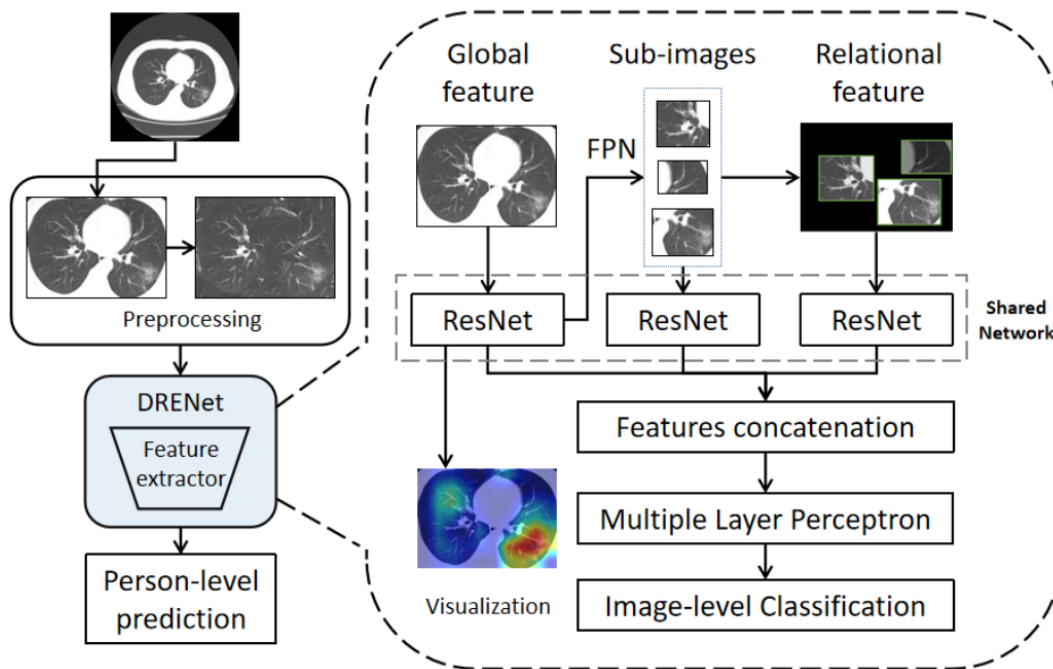


Figura 3.2: Arquitectura para extracción de características y predicción a nivel de corte
Fuente: IEEE [Song et al., 2021].

La *DRENet* está construida en base a la *ResNet50* con la adición de una *Feature Pyramid Network* (FPN) para extraer los detalles de cada imagen. La FPN se modifica para identificar las regiones con lesiones de diferentes tamaños y se le acopla un módulo de atención para aprender la importancia de las características.

El flujo que sigue la *DRENet* es el siguiente:

- Cada corte se introduce en una *ResNet50* para extraer el *feature map*, enfatizando en regiones pequeñas, ya que las lesiones son generalmente pequeñas y el uso de *feature maps* grandes solo introduciría ruido al modelo.

- El *feature map* pasa por una capa de *pooling* y una capa densa para obtener las *features* globales. La FPN utiliza estas últimas para identificar las regiones importantes dentro de la imagen y les asigna una puntuación.
- De acuerdo a estas regiones, las top K sub-imágenes son recortadas y al mismo tiempo, se genera una nueva imagen a partir de la original multiplicando las regiones de las top K sub-imágenes con sus correspondientes puntuaciones. Las regiones no presentes en las top K sub-imágenes, se ponen en cero.
- Se alimenta la *ResNet50* con la nueva imagen y las sub-imágenes anteriores para extraer *features* locales dentro de estas últimas y *features* relacionales entre ellas.
- Tanto las *features* locales como las relacionales se concatenan con las *features* globales y se introducen en un perceptrón multicapa para obtener la predicción a nivel de corte.

Para cada paciente, se hacen predicciones a nivel de corte y se utiliza el *mean pooling* para agregar los valores resultantes, obteniendo la predicción a nivel de paciente.

Por último, la arquitectura propuesta por [Subramaniyan *et al.*, 2021] sigue el flujo mostrado en la Figura 3.3

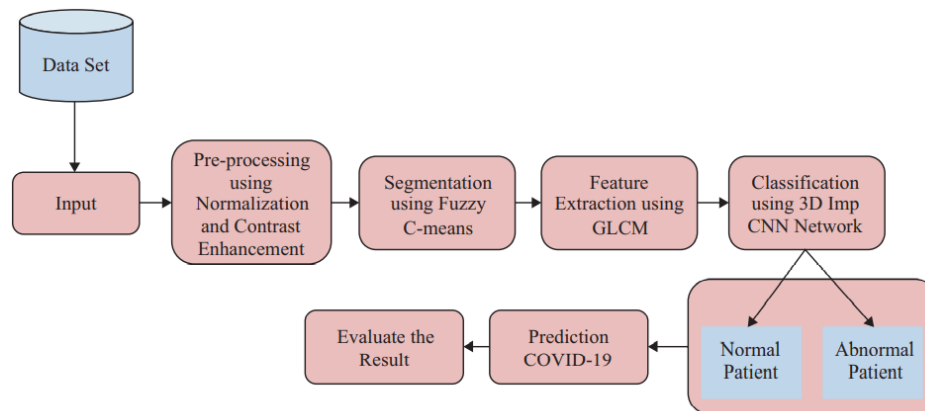


Figura 3.3: Arquitectura para extracción de características y predicción a nivel de CTs
Fuente: Springer [Subramaniyan *et al.*, 2021].

El modelo de *deep learning* se denomina *3D ImpCNN*. No entran en detalles sobre la estructura, sin embargo, destacan algunos aspectos importantes.

- Los pesos de la capa densa se propagan hacia los *feature maps* de las capas convolucionales para generar mapas de atención.
- La red almacena la información en volúmenes mediante la aplicación de *3D max pooling*.
- La función *loss* corresponde a *binary cross entropy*.

3.3. RESULTADOS

El modelo de *deep learning* planteado por [Wang *et al.*, 2021] se entrena durante 15000 *epochs* usando un *learning rate* inicial de 0,01, el que es ajustado automáticamente durante el entrenamiento. Solo se entrena la parte modificada de la *M-Inception* y se utiliza *Adaptive Moment Estimation Gradient Descent* como optimizador.

Para mejorar el *accuracy* de clasificación, se utiliza un *ensemble*¹³ de clasificadores. Se obtienen los resultados mostrados en la Tabla 3.1

Métrica	Val. Interna	Val. Externa
AUC (95 % CI)	0.93 (0.90 to 0.96)	0.81 (0.71 to 0.84)
Accuracy, %	89.5	79.3
Sensitivity	0.88	0.83
Specificity	0.87	0.67
PPV	0.71	0.55
NPV	0.95	0.90
Youden Index	0.75	0.48
F1 score	0.77	0.63

Tabla 3.1: Rendimiento del modelo de *deep learning* en la tarea de distinguir entre neumonía viral típica y COVID-19.

Por el lado de [Song *et al.*, 2021], fijan la cantidad de sub-imágenes que se extraen de los *inputs* en $K = 3$. Se entrena el modelo en dos tareas distintas: discriminar entre pacientes sanos y pacientes con COVID-19; distinguir entre pacientes con COVID-19, pacientes con neumonía bacteriana y pacientes sanos.

Los resultados que se obtienen a nivel de paciente para la primera tarea se detallan en la Tabla 3.2

Datos	AUC	Accuracy, %	Precision	Recall	Specificity	F1 score
Val. Interna	0.93	-	-	-	-	-
Val. Externa	0.95	86	0.79	0.96	0.77	0.87

Tabla 3.2: Rendimiento del modelo de *deep learning* en la tarea de distinguir entre COVID-19 y sano.

¹³Consiste en entrenar más de una red neuronal en el mismo conjunto de datos, luego usar cada uno de los modelos entrenados para hacer una predicción antes de combinar las predicciones para obtener una predicción final. Esto ayuda a reducir la varianza de las predicciones.

Para la segunda tarea, las métricas a nivel de paciente al evaluar en el conjunto de validación externa se detallan en la Tabla 3.3

Datos	Accuracy, %	Precision	Recall	Specificity	F1 score
Val. Externa	93	0.86	0.93	0.93	0.93

Tabla 3.3: Rendimiento del modelo de *deep learning* en la tarea de distinguir entre COVID-19, neumonía bacteriana y sano.

Finalmente, los resultado que obtienen [Subramaniyan *et al.*, 2021] en la tarea de discriminar entre pacientes con COVID-19 positivo y pacientes sanos se resumen en la Tabla 3.4

Datos	Accuracy, %	Precision, %	Recall, %	Specificity, %	F1 score
Validación	96.5	96.1	95.2	96.2	95.6

Tabla 3.4: Rendimiento del modelo de *deep learning* en la tarea de distinguir entre COVID-19 y sano.

CAPÍTULO 4

PROPUESTA DE SOLUCIÓN

La mayoría de los modelos propuestos para el problema de clasificación de COVID-19 en imágenes de tomografía computarizada del tórax se basan en el uso de cortes transversales debido a limitaciones tales como el tiempo que tarda el entrenamiento, la complejidad computacional inherente al uso de datos volumétricos y los recursos de hardware limitados. La ventaja de trabajar con datos volumétricos es que se cuenta con tres dimensiones de información, las cuales se condensan en información espacial y temporal.

El uso de cortes transversales implica que se pierde la dimensión temporal o en este caso, la altura a la que se encuentran las lesiones en los pulmones. Esto puede ser clave a la hora de identificar anomalías, ya que una misma lesión podría ser síntoma de distintas enfermedades dependiendo si se encuentra en la parte superior del pulmón o en la parte inferior. Para suplir esta pérdida, se han usado técnicas como la segmentación de las imágenes para proveer a las redes neuronales de información adicional sobre las imágenes, además de emplear metodologías tales como analizar por separado los cortes de una tomografía y ponderar los resultados para obtener una visión global del diagnóstico.

El objetivo de esta propuesta es desarrollar un modelo de redes neuronales que tome como entrada los datos volumétricos de las tomografías y sea capaz de identificar las anomalías sin tener que recurrir a técnicas sofisticadas de preprocesamiento de los datos. Para aprovechar toda la información de las tomografías, se propone usar una red convolucional compuesta por capas 3D capaces de aplicar operaciones de convolución y *pooling* sobre las imágenes completas. La desventaja de este acercamiento es que la cantidad de datos disponibles para el entrenamiento es menor y se deben tener en cuenta las restricciones mencionadas al comienzo de este inciso, es por esto que, de forma adicional, se plantea el uso de un *autoencoder* que sea capaz de extraer información clave de las imágenes volumétricas y la condense en una matriz de características que se usa junto con la red convolucional principal para aumentar la fiabilidad de la clasificación.

4.1. DATASET

El dataset utilizado para el entrenamiento de las redes neuronales propuestas corresponde a tomografías computarizadas de tórax provistas por hospitales municipales de Moscú como parte del programa de investigación ***Molecular Sciences for Medicine***[Morozov *et al.*, 2020]. Consta de 1100 estudios en formato **NiftI**, de los cuales, 856 contienen lesiones atribuidas al COVID-19 y 254 son de personas sanas o con lesiones debido a otras enfermedades. Por otra parte, las tomografías computarizadas almacenan la intensidad de los *voxels* en unidades *Hounsfield (HU)*; para este dataset, las intensidades oscilan en el rango entre -1000 a 2000 *HU*.

La distribución de las personas que participaron de los estudios es:

- 42 % hombres, 56 % mujeres y 2 % no reportaron su género.
- La edad se encuentra entre 18 y 97 años con mediana de 47 años.

Para utilizar el dataset como parte del entrenamiento de las redes neuronales, es necesario realizar un proceso previo. Este consiste en tres pasos:

- Rotar en 90° los volúmenes para dejarlos en una orientación fija.
- Normalizar la intensidad de los *voxels* para que los valores se encuentren entre 0 y 1. Para esto, se usa -1000 *HU* como cota inferior y 400 *HU* como cota superior, esta última debido a que sobre 400 *HU* se encuentran huesos con diferentes radio intensidades que no son relevantes para la aplicación que se plantea en este documento.
- Redimensionar los volúmenes, dejándolos con dimensiones $128 \times 128 \times 64$ para reducir los recursos de hardware necesarios.

4.2. AUTOENCODER

Un autencoder es una red neuronal artificial capaz de aprender a comprimir y codificar datos de forma eficiente, además de poder usar dicha codificación para reconstruir una representación lo más fiel posible de los datos originales (Figura 4.1).

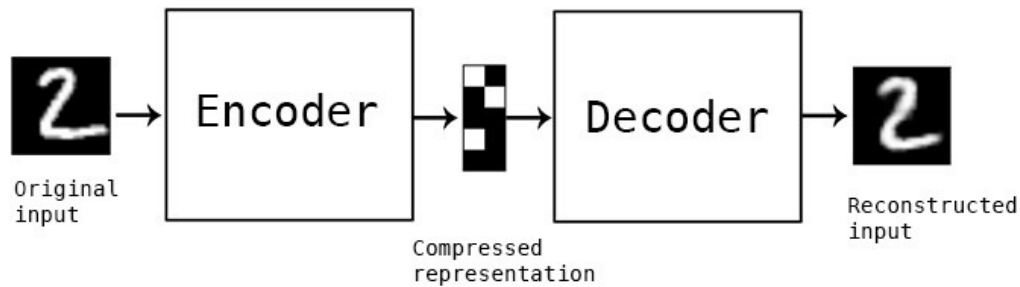


Figura 4.1: Ejemplo de *autoencoder* para codificación del dataset MNIST
Fuente: Towards Data Science [Badr, 2019].

La estructura básica de un *autoencoder* se encuentra compuesta de cuatro partes:

- **Encoder:** Componente que se encarga de la reducción de dimensiones de los datos y la compresión de los datos para obtener una representación codificada.
- **Bottleneck:** Capa que contiene la representación codificada de los datos (vectores de características).
- **Decoder:** Componente que se encarga de decodificar la representación generada por el encoder para obtener un resultado lo más cercano al original.
- **Reconstruction Loss:** Función de pérdida que se usa para medir el rendimiento del decoder y que tanto se parece el resultado con los datos originales.

El encoder y el decoder pueden estar conformados por capas densas, convolucionales o LSTM según la naturaleza de los datos y la aplicación. Para la propuesta de solución se plantea el uso de un *autoencoder* con capas convolucionales 3D, función de pérdida **Mean Squared Error** y optimizador **Adam** (Figura 4.2). El *autoencoder* tiene como entrada imágenes de dimensiones $128 \times 128 \times 64$ y genera una matriz de dimensiones $16 \times 16 \times 8 \times 32$ extraída desde el *encoder*.

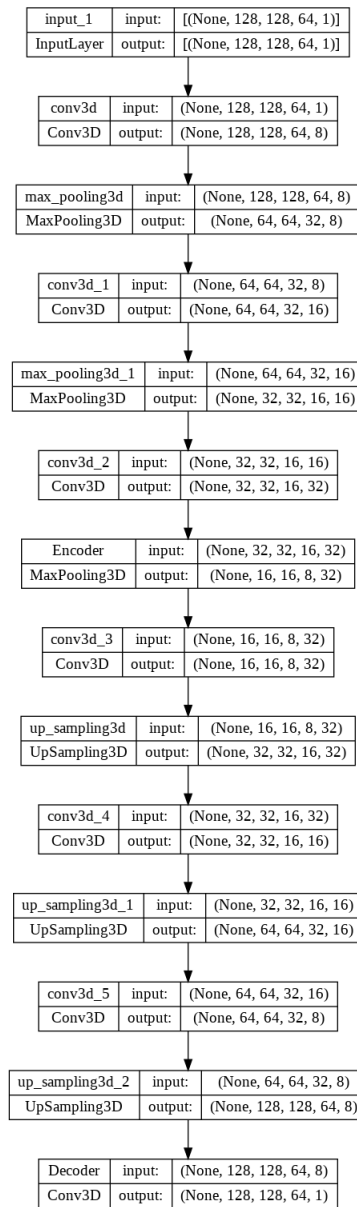


Figura 4.2: Arquitectura de *autoencoder* convolucional con capas 3D
Fuente: Propia.

Se entrena durante 100 *epochs*, con un *batch* de 16 imágenes y una separación de 75 %/25 % para el conjunto de entrenamiento y validación respectivamente.

Para visualizar su desempeño se toma como ejemplo la tomografía con diagnóstico COVID-19 mostrada en la Figura 4.3

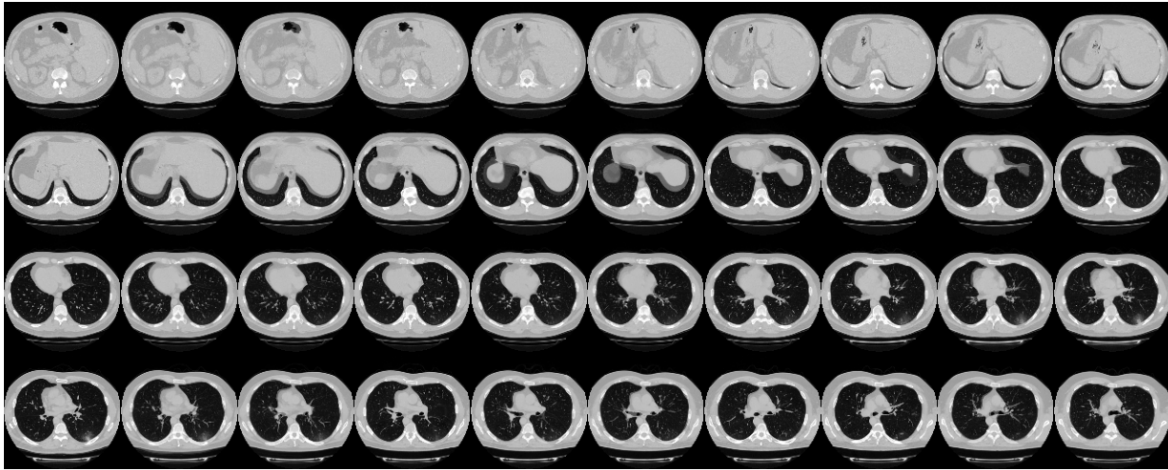


Figura 4.3: Tomografía computarizada de tórax
Fuente: [Morozov et al., 2020].

La reconstrucción obtenida por el *autoencoder* se muestra en la Figura 4.4:

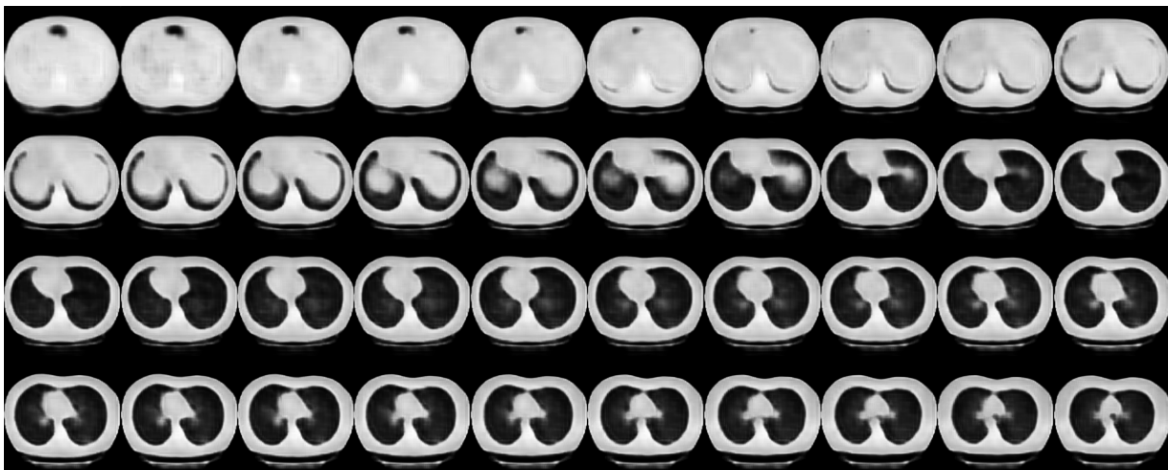


Figura 4.4: Reconstrucción de una tomografía computarizada de tórax
Fuente: Propia.

Como se observa, la reconstrucción pierde los detalles finos de la imagen pero conserva la información de la estructura general de los pulmones. Esto puede ser relevante, ya que uno de los problemas de trabajar con tomografías de tórax es que las redes neuronales tienden a enfocarse en los bordes del cuerpo en vez del interior de los pulmones, requiriendo un tratamiento previo de las imágenes para mitigar este comportamiento.

4.3. RESNET

Las *ResNets* son redes neuronales muy profundas que emplean un tipo de conexión entre capas para combatir los problemas que conlleva el utilizar muchas capas (por ejemplo, los gradientes desvanecientes). Este tipo de conexiones especiales se denominan *skip connections* y su función es proporcionarle a una capa el *output* de una capa anterior sin pasar por medio de otras capas intermedias. A un conjunto de capas con *skip connections* se le conoce como bloque residual y es la base de las *ResNets* (Figura 4.5)

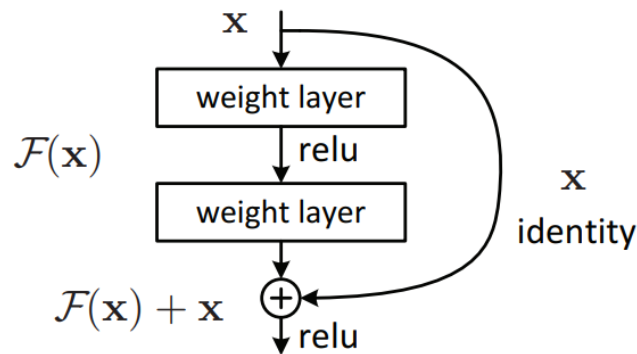


Figura 4.5: Ejemplo de bloque residual
Fuente: [He et al., 2016].

Para la propuesta de solución se plantea el uso de una *ResNet* con 50 capas, entre las que se encuentran capas convolucionales 3D. A diferencia de una *ResNet* usada para clasificación, el *output* de esta red es un vector de 512 *features* que es utilizado en conjunto con el vector generado por el *autoencoder* para realizar la clasificación de las imágenes CT.

La arquitectura de esta red se puede dividir en 6 partes:

- Procesamiento previo del *input*.
- Bloque compuesto por 1 segmento convolucional y 2 segmentos de identidad.
- Bloque compuesto por 1 segmento convolucional y 3 segmentos de identidad.
- Bloque compuesto por 1 segmento convolucional y 5 segmentos de identidad.
- Bloque compuesto por 1 segmento convolucional y 2 segmentos de identidad.
- Bloque compuesto por capas densas.

Para más detalle, ver Figura A.4 en los anexos .

4.4. MODELO PROPUESTO

El modelo propuesto está compuesto por la *ResNet* mencionada anteriormente y una red convolucional simple encargada de interpretar la matriz de características generada por el *autoencoder*. Los *outputs* de ambas redes se concatenan y se pasan a través de capas densas para obtener la clasificación de las imágenes.

La motivación detrás de esta propuesta proviene del supuesto que el uso de imágenes 3D, si bien proporciona más información que puede ser capturada por las redes neuronales, conlleva un peor desempeño debido a los recursos de hardware necesarios para realizar el entrenamiento de las redes neuronales (se deben utilizar *batches* más pequeños durante el entrenamiento, las imágenes tienen dimensiones más pequeñas, entre otros). Para solucionar este problema, durante el entrenamiento de la red principal, se añade la información extra provista por la matriz generada por el *autoencoder*, la cual contiene la características más relevantes de las imágenes de forma condensada.

De esta forma se obtiene un modelo compuesto por dos redes neuronales concatenadas, con función de pérdida **Binary Cross Entropy** y optimizador **Adam** con decaimiento exponencial en la tasa de aprendizaje. El *input* es un arreglo compuesto por las imágenes de dimensiones $128 \times 128 \times 64$ y la matriz de características generada por el *autoencoder* de dimensiones $16 \times 16 \times 8 \times 32$.

Para evitar tener un conjunto de datos desbalanceado, se seleccionan 254 imágenes correspondientes a COVID-19 y 254 imágenes correspondientes a estudios No COVID-19, el cual se separa en 60 %/20 %/20 % para el conjunto de entrenamiento, validación y test respectivamente usando un muestreo estratificado, de esta forma se asegura que todos los conjuntos tienen la misma proporción de clases. Se entrena durante 85 *epochs*, con un *batch* de 8 imágenes y se toma un *checkpoint* de las iteraciones con mejor *accuracy* sobre el conjunto de validación.

La Figura 4.6 muestra un extracto de la arquitectura completa del modelo donde se muestra la concatenación con el componente adicional encargado de procesar la matriz de características generada por el *autoencoder*:

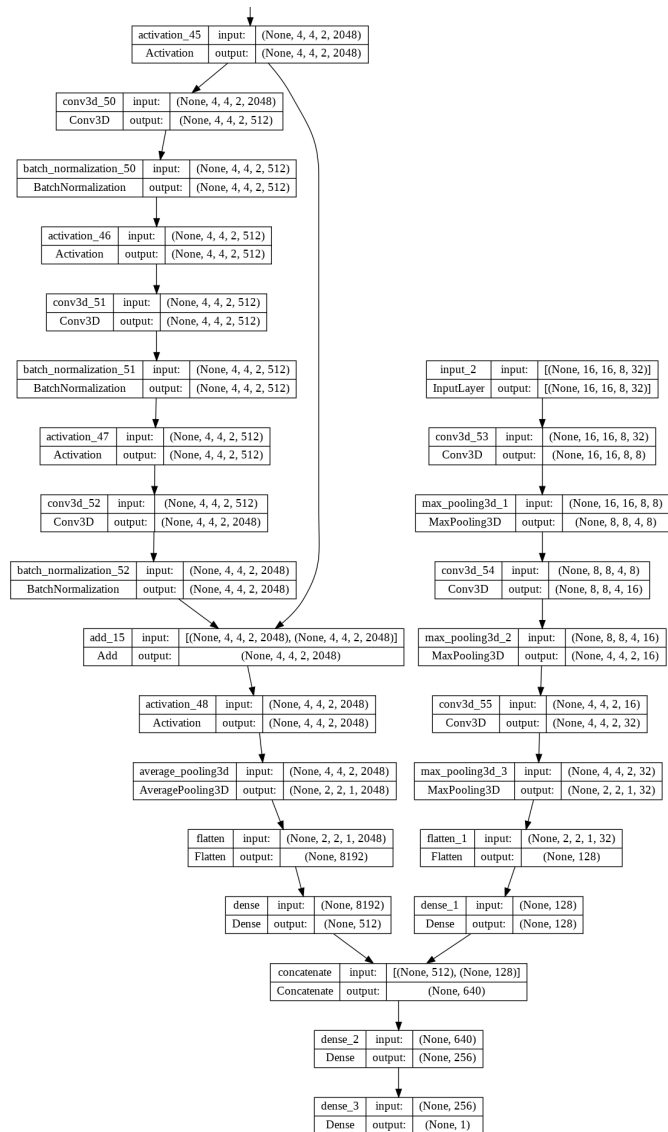


Figura 4.6: Extracto del modelo propuesto, donde se detalla la concatenación de las dos redes neuronales

Fuente: Propia.

CAPÍTULO 5

RESULTADOS

Para evaluar el rendimiento de la solución propuesta se utiliza el conjunto de *test*, el cual está compuesto por 102 imágenes correspondientes a las siguientes clases: 51 COVID-19 y 51 No COVID-19. El modelo no se expone a este conjunto durante el entrenamiento, esto es importante para obtener métricas libres de sesgo que permitan visualizar el rendimiento de la propuesta.

Los modelos de *deep learning* arrojan probabilidades al momento de hacer una predicción, por lo tanto, para categorizar dichas probabilidades entre las distintas clases es necesario determinar un umbral a partir del cual una predicción se considera de una clase u otra. En otras palabras, las probabilidades que entregan los modelos son una medida de la confianza que tiene el modelo de que el *input* pertenece a la clase positiva (COVID-19 en este caso) o no. Para esto se selecciona como umbral 0,6 ya que es un valor que no es muy estricto y deja menos margen para que la red deje al azar la clasificación.

Una vez que se categorizan las predicciones se genera la matriz de confusión, la que se muestra en la Figura 5.1

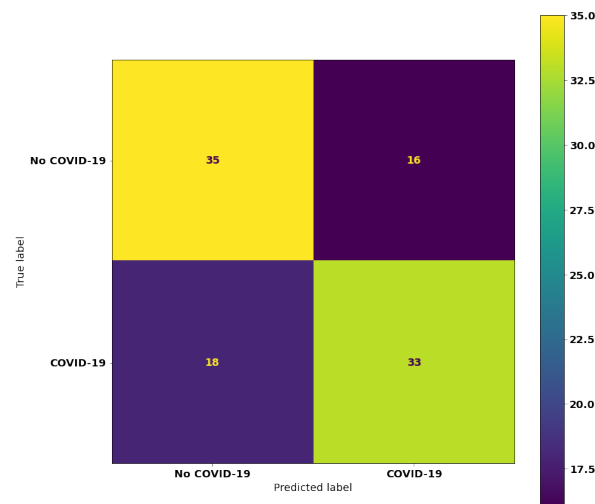


Figura 5.1: Matriz de confusión obtenida al evaluar la propuesta en el conjunto de *test*
Fuente: Propia.

Como se puede ver, las diagonales de la matriz de confusión indican una coincidencia entre las predicciones y los valores reales, por lo tanto, a pesar de que se observa una leve confusión entre las clases, en general el modelo propuesto tiene la capacidad de discriminar entre los diagnósticos. Cabe destacar que la cantidad de cada diagnóstico que fue clasificado correctamente es similar en ambos casos, por lo que se puede decir que el modelo no tiene una preferencia o sesgo hacia uno u otro.

Para profundizar en el desempeño, se obtienen las métricas detalladas en la Tabla 5.1 a partir de la matriz de confusión usando las ecuaciones definidas en el marco conceptual

Datos	AUC	Accuracy, %	Precision	NPV	Recall	Specificity	F1-score
Validación	0.70	70	0.73	0.76	0.63	0.76	0.67
Test	0.67	67	0.67	0.69	0.65	0.69	0.66

Tabla 5.1: Rendimiento del modelo de *deep learning* propuesto en la tarea de distinguir entre COVID-19 y No COVID-19.

El AUC representa la capacidad del modelo de diferenciar entre las distintas clases, por lo tanto valores cercanos a 1 indican que el modelo puede clasificar correctamente los datos en sus correspondientes clases. El modelo propuesto obtuvo 0,67 en el conjunto de *test* por lo que tiene cierta capacidad de diferenciar correctamente los diagnósticos.

De igual forma esto se puede ver en la gráfica de la curva ROC (Figura 5.2), la cual muestra una inclinación hacia el eje de la tasa de verdaderos positivos

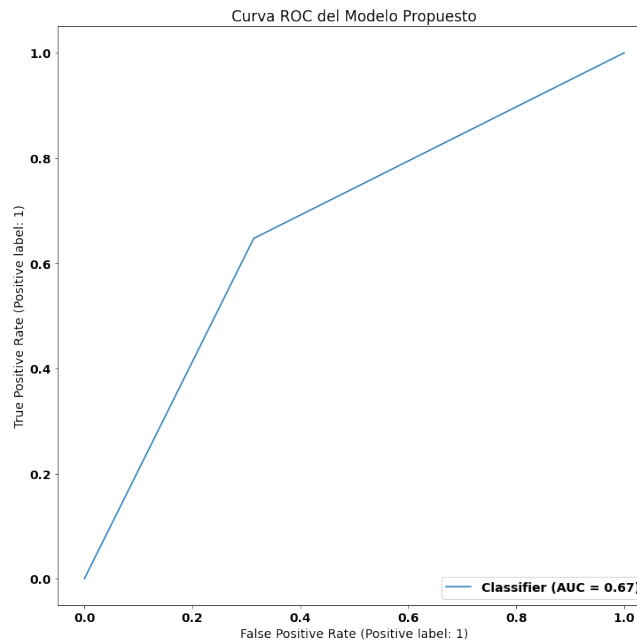


Figura 5.2: Curva ROC obtenida al evaluar la propuesta en el conjunto de *test*
Fuente: Propia.

Por otra parte, el *accuracy* es una medida de la razón entre las predicciones correctas y las predicciones totales, suele no ser confiable en problemas con conjuntos de datos desbalanceados pero en el caso de la propuesta, se cuenta con la misma cantidad de datos para ambas clases. Valores cercanos a 1 son mejores y modelo propuesto obtuvo un *accuracy* de 0,67 en el conjunto independiente o de *test*, por lo que se puede decir que el 67 % de las predicciones corresponden al diagnóstico real.

La *precision* mide la proporción de predicciones positivas que se clasifican correctamente, es decir, predicciones que son clasificadas como COVID-19 y que realmente corresponden a imágenes con dicho diagnóstico. También sirve para evaluar la cantidad de falsos positivos, por lo que valores cercanos a 1 indican que el modelo es capaz de identificar correctamente las clases positivas y se tienen pocos casos de falsos positivos en comparación. El modelo propuesto obtuvo 0,67 en el conjunto de *test*, por lo tanto se tiene cierta cantidad de falsos positivos o diagnósticos No COVID-19 clasificados como COVID-19.

El *negative predictive value* es lo inverso a la *precision*, por lo que mide la proporción de predicciones negativas que se clasifican correctamente, por ejemplo, imágenes CT que son clasificadas como No COVID-19 y que realmente tienen ese diagnóstico. Valores cercanos a 1 son mejores e indican que el modelo es capaz de identificar correctamente la clase negativa, además de presentar una baja cantidad de falsos negativos, lo que es realmente importante ya que puede poner en riesgo la vida de los pacientes. El modelo propuesto obtuvo 0,69 en el conjunto de *test*.

El *recall* es una medida de la proporción de casos positivos que son clasificados correctamente, es decir, imágenes que tienen como diagnóstico COVID-19 y que el modelo es capaz de clasificar como tal. A diferencia de la *precision* y los falsos positivos, el *recall* toma en cuenta la cantidad de falsos negativos, por lo que valores cercanos a 1 indican una menor cantidad de estos. El modelo propuesto obtuvo 0,65 en el conjunto de *test*, por lo que existe cierta cantidad de imágenes correspondientes a COVID-19 clasificadas como No COVID-19, pero no llega a ser considerable.

La *specificity* es lo opuesto al *recall*, por lo tanto es una medida de la proporción de los casos negativos que son clasificados correctamente, en el dominio de este documento correspondería a la proporción de diagnósticos No COVID-19 que son clasificados como tal. Valores cercanos a 1 son mejores, el modelo propuesto obtuvo 0,69 en el conjunto de *test*, por lo que se tiene una cantidad de diagnósticos No COVID-19 clasificados como COVID-19 pero no llega a ser igual a la cantidad clasificada correctamente.

El *F1-score* es la media armónica entre la *precision* y el *recall*, por lo que entrega un resumen de qué tan buenas son estas métricas. Valores cercanos a 1 son mejores y el modelo propuesto obtuvo 0,66 en el conjunto de *test*, valor que se refleja en las métricas individuales señaladas anteriormente.

A modo de ejercicio, se construye un modelo conformado por una *ResNet-50* clásica, bajo las mismas condiciones de entrenamiento pero sin la información adicional provista por la matriz de características generada por el *autoencoder*. Se obtiene la matriz de confusión mostrada en la Figura 5.3

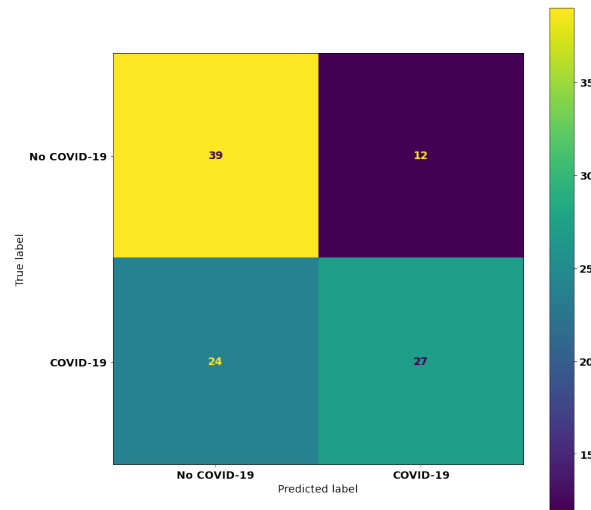


Figura 5.3: Matriz de confusión obtenida al evaluar una *ResNet-50* en el conjunto de *test*
Fuente: Propia.

Como se puede apreciar, la *ResNet-50* presenta cantidades similares para diagnósticos correspondientes a COVID-19 que son clasificados como tal y que también son clasificados como No COVID-19. Por lo tanto, se puede decir que tiene un mayor sesgo hacia la clase negativa, es mejor identificando las imágenes con diagnóstico No COVID-19.

De igual forma, se recopilan las métricas derivadas de la matriz de confusión para obtener mayor detalle del rendimiento del modelo (Tabla 5.2)

Datos	AUC	Accuracy, %	Precision	NPV	Recall	Specificity	F1-score
Validación	0.67	67	0.69	0.73	0.61	0.73	0.65
Test	0.65	65	0.69	0.76	0.53	0.76	0.60

Tabla 5.2: Rendimiento de la *ResNet-50* en la tarea de distinguir entre COVID-19 y No COVID-19.

En general las métricas obtenidas por la *ResNet-50* son similares a las del modelo propuesto, encontrándose la mayor diferencia en el *recall*. Esto se refleja correctamente en la matriz de confusión y lo que se menciona anteriormente sobre la confusión que tiene la *ResNet-50* al clasificar COVID-19. De igual forma, la *specificity* de la *ResNet-50* es mayor ya que es mejor identificando los casos No COVID-19.

En la Tabla 5.3 se incluye la comparación de las métricas de ambos modelos sobre el conjunto de *test*

Modelos	AUC	Accuracy, %	Precision	NPV	Recall	Specificity	F1-score
Propuesta	0.67	67	0.67	0.69	0.65	0.69	0.66
ResNet-50	0.65	65	0.69	0.76	0.53	0.76	0.60

Tabla 5.3: Comparación del rendimiento entre la propuesta y la *ResNet-50* sobre el conjunto de *test* en la tarea de distinguir entre COVID-19 y No COVID-19.

A partir de lo observado, se puede asumir que la información generada por el *autoencoder* sirve para mejorar el rendimiento a la hora de clasificar COVID-19. Esto queda en evidencia al comparar las matrices de confusión, donde el modelo propuesto tiene un comportamiento más homogéneo, sin mostrar preferencia por un diagnóstico en particular.

CAPÍTULO 6

CONCLUSIONES

La propuesta de solución tiene como alcance construir un modelo de *deep learning* capaz de tomar imágenes de tomografía computarizada y clasificarlas en COVID-19 o No COVID-19. El factor diferenciador de la propuesta es que trabaja con las imágenes 3D y no con cortes transversales como es común ver en otras propuestas del estado del arte, además las imágenes no necesitan ser sometidas a un procesamiento especialmente complejo previo al análisis. En base a esto, se logró desarrollar un modelo con un rendimiento inferior a lo que se observa en propuestas más sofisticadas pero que contiene un gran potencial de mejora. Este potencial reside en la naturaleza modular de la solución, ya que consta de una arquitectura base a la cual se le pueden adicionar componentes que aporten información adicional sobre las imágenes. En este caso, se utilizó un *autoencoder* como medio de extracción de características relevantes que sirven de apoyo a la arquitectura principal para comprender los detalles más sutiles de las imágenes y por lo tanto, poder distinguir entre COVID-19 y No COVID-19 de forma más fidedigna; análogo a lo que sería un manual de instrucciones durante el armado de un mueble.

Actualmente la solución propuesta solo puede clasificar COVID-19 y No COVID-19, sin embargo esto podría modificarse para permitir el diagnóstico de distintas enfermedades pulmonares y solo se encuentra limitado por el conjunto de datos utilizado, no obstante, las enfermedades con lesiones u otros indicadores similares podrían causar confusión en el modelo por lo que se debe tener en cuenta. También existe una limitación en las dimensiones de las imágenes de tomografía computarizada utilizadas por el modelo propuesto debido a los recursos de hardware disponibles, estas deben ser de $128 \times 128 \times 64$, por lo que añade una restricción adicional al tipo de estudios que se pueden analizar, sobretodo si estos cuentan con menos de 64 cortes.

En general, las métricas obtenidas al evaluar el modelo propuesto son prometedoras, siendo las más importantes el AUC con un valor de 0,67, el *recall* con 0,65 y la *specificity* con 0,69. Estas métricas señalan que el modelo tiene cierta capacidad para distinguir entre una imagen con COVID-19 y una sin COVID-19, además de que puede clasificar correctamente cada clase gran parte de las veces y no muestra tendencia alguna a elegir una sobre la otra.

En base a los resultados, se puede concluir que los objetivos se cumplieron de forma satisfactoria, siendo el principal desarrollar un clasificador de COVID-19 a partir de imágenes de tomografía computarizada. Para esto, se trabajó con un conjunto de 1100 estudios o tomografías en formato *NiftI* provenientes del programa de investigación **Molecular Sciences for Medicine**[Morozov *et al.*, 2020], de las cuales 856 corresponden a COVID-19 y 254 cuentan con diagnóstico saludable o de otras enfermedades no atribuidas al COVID-19. Las imágenes fueron preparadas mediante un proceso simple que consiste en normalizar el rango de los valores y ajustar las dimensiones a un tamaño que pueda ser usado por el modelo.

Para comprobar la validez de los objetivos se definió un modelo basado en una arquitectura clásica (*ResNet-50*) y que carece del componente de información adicional que diferencia al modelo propuesto, el entrenamiento se realizó bajo las mismas condiciones en ambos casos; los resultados obtenidos entregan una base de comparación y además permiten cuantificar el aporte del componente adicional. Se observa que proveer información adicional sobre las imágenes tiene un impacto positivo, ya que mejora la confianza del modelo al momento de identificar aquellas que presentan COVID-19 y disminuye el sesgo que se evidencia en la *ResNet-50* (véase la *specificity* en comparación con el *recall* en la Tabla 5.3), por lo que la hipótesis planteada toma fuerza.

La principal contribución del trabajo realizado es desarrollar una arquitectura modular con un margen de expansión considerable, a partir de la cual poder diagnosticar COVID-19 en imágenes de tomografía computarizada y que se puede aplicar en otros dominios con complejidades similares, es decir, que requieran de una atención detallada para poder discriminar entre las distintas posibilidades de decisión.

Dado que se utilizan imágenes 3D, el modelo tiene acceso a una dimensión de información adicional que le provee el contexto necesario para comprender la forma en que se relacionan entre sí las imágenes, sobretodo porque la altura en la que se encuentran las lesiones dentro de los pulmones es de importancia para establecer un diagnóstico adecuado. Esto simplifica el trabajo previo que se debe realizar para obtener buenos resultados, por lo que la barrera de entrada para solucionar problemas de este tipo se ve reducida.

Gracias a lo anterior, la propuesta puede ser aplicada no solo como herramienta de diagnóstico de COVID-19, si no que puede extenderse a otras patologías.

El impacto que puede derivarse de la propuesta puede no ser grande, ya que el área de estudio está en constante evolución y es posible encontrar distintas alternativas de arquitecturas con rendimientos superiores y que utilizan técnicas del estado del arte que han demostrado ser confiables, sin embargo, la mayoría requiere de procesos complejos para preparar el conjunto de datos, ya sea extrayendo regiones de interés u obteniendo máscaras de segmentación y son altamente especializadas. Por lo tanto, uno de los aportes de la propuesta yace en la flexibilidad inherente del diseño planteado, pudiendo ser aplicada en múltiples problemas relacionados con imágenes si se mantiene la arquitectura base o alguna similar a esta o ser expandido a otros tipos de problemas sin grandes modificaciones.

Pero sin duda, el mayor aporte de este trabajo y otros similares en el área es que permiten avanzar la tecnología al servicio de las personas, mientras más investigaciones se hagan al respecto, aumenta el entendimiento de la naturaleza del problema y se pueden lograr mejores desarrollos y soluciones que pueden ser extrapolados a otros problemas.

La principal desventaja de una propuesta basada en datos volumétricos es que se cuenta con recursos de hardware limitados, esta situación podría mejorar a medida que aumenta la disponibilidad de componentes con mayores capacidades y los procesos se hacen más eficientes, pero actualmente se debe tener especial consideración con esto. Es importante

saber aprovechar los recursos disponibles de la mejor forma para que no se pierda el balance entre la información adicional proveniente de usar datos con mayor cantidad de dimensiones y el consumo de recursos que los acompaña.

Las soluciones al problema planteado en este documento no se limitan a trabajar solo con redes neuronales convolucionales, mediante una manipulación del conjunto de datos se puede trabajar con redes más simples como los perceptrones multicapa o redes más complejas como las redes neuronales recurrentes.

Debido a la complejidad inherente de la problemática, la mayor parte de las soluciones propuestas involucran el uso de modelos muy profundos para poder extraer todas las características posibles de los datos, por más sutiles que sean; sin embargo, esto trae consigo una serie de inconvenientes relacionados con la forma en que se diluye la información durante el proceso de aprendizaje de las redes neuronales, como lo es el problema de los gradientes desvanecientes. Para el modelo propuesto se utilizó como base una *ResNet-50* ya que está diseñada especialmente para evitar este problema mediante el uso de conexiones entre capas conocidas como *skip connections* y que actúan como forma de preservación de la información de las capas anteriores a medida que se propaga a las capas más profundas. Por lo tanto, se debe tener en cuenta esta situación si se quieren utilizar otros tipos de arquitecturas.

Como trabajo futuro, se pueden atacar distintos frentes para mejorar el desempeño de la propuesta. Uno de estos podría ser explorar el uso de métodos más sofisticados para la extracción de características de las imágenes, ya que el *autoencoder* utilizado en este trabajo produce imágenes con baja resolución y suele perder los detalles finos. Otra alternativa podría involucrar el uso de distintos componentes que aporten con información de los datos a la arquitectura base, ya sea complementando el actual o reemplazándolo completamente. Finalmente, se puede modificar la arquitectura base para adaptarla mejor al problema (ya sea ajustando los hiper parámetros y/o modificando la estructura como tal) o usar algo completamente distinto según lo que se quiera lograr, un ejemplo de esto último podría ser utilizar una red neuronal recurrente para aprovechar la naturaleza secuencial de los datos.

Lo mencionado anteriormente se enfoca en la arquitectura de la propuesta, pero otro acercamiento igualmente válido puede ser explorar técnicas de procesamiento de las imágenes para que el entrenamiento del modelo sea mucho más robusto. Se podrían tratar las tomografías para que no incluyan el contorno del cuerpo, ya que es algo que varía mucho entre cada imagen y solo le aporta ruido al modelo, de igual forma se pueden utilizar máscaras de las zonas de interés mediante segmentación de las imágenes para dirigir la atención del modelo hacia donde se encuentran las lesiones.

También se podría aumentar el tamaño del conjunto de datos mediante rotaciones u otras transformaciones aplicadas a las imágenes, teniendo especial cuidado ya que transformaciones muy extremas como rotaciones de 180 grados o reflejar las imágenes puede causar problemas al momento de realizar las predicciones, debido a que las posiciones de los órganos cambiarían completamente y la significancia de las lesiones provocadas por enfermedades es relativa a estas posiciones.

Finalmente, trabajar con imágenes CT de mayor resolución puede ser beneficioso para mejorar la capacidad de discriminación de la propuesta, por lo tanto, si se desea utilizar imágenes con distintas dimensiones a las usadas en este trabajo, se deben modificar las capas de entrada tanto de la red neuronal principal como la del *autoencoder*; también es probable que se deba hacer un ajuste en las capas internas y en los hiperparámetros.

La gracia de esta propuesta es que los caminos a seguir pueden surgir a partir de cualquier combinación de las alternativas mencionadas anteriormente y solo se encuentran limitados por la creatividad.

REFERENCIAS BIBLIOGRÁFICAS

- [Adams *et al.*, 2020] Adams, H. J., Kwee, T. C., Yakar, D., Hope, M. D., y Kwee, R. M. (2020). Chest ct imaging signature of coronavirus disease 2019 infection: In pursuit of the scientific evidence. *Chest*, 158(5):1885–1895.
- [Ai *et al.*, 2020] Ai, T., Yang, Z., Hou, H., Zhan, C., Chen, C., Lv, W., Tao, Q., Sun, Z., y Xia, L. (2020). Correlation of chest ct and rt-pcr testing in coronavirus disease 2019 (covid-19) in china: a report of 1014 cases. *Radiology*, p. 200642.
- [Badr, 2019] Badr, W. (2019). Auto-encoder: What is it? and what is it used for. <https://towardsdatascience.com/auto-encoder-what-is-it-and-what-is-it-used-for-part-1-3e5c6f017726>. Accessed: 08-10-2022.
- [Bastião Silva, 2014] Bastião Silva, L. (2014). Medical imaging services supported on cloud. https://www.researchgate.net/publication/260374716_Medical_imaging_services_supported_on_cloud. Accessed: 10-12-2020.
- [Bernheim *et al.*, 2020] Bernheim, Adam and Mei, Xueyan and Huang, Mingqian and Yang, Yang and Fayad, Zahi A and Zhang, Ning and Diao, Kaiyue and Lin, Bin and Zhu, Xiqi and Li, Kunwei and others (2020). Chest ct findings in coronavirus disease-19 (covid-19): relationship to duration of infection. *Radiology*.
- [Ding *et al.*, 2020] Ding, X., Xu, J., Zhou, J., y Long, Q. (2020). Chest ct findings of covid-19 pneumonia by duration of symptoms. *European Journal of Radiology*, p. 109009.
- [Fang *et al.*, 2020] Fang, Y., Zhang, H., Xie, J., Lin, M., Ying, L., Pang, P., y Ji, W. (2020). Sensitivity of chest ct for covid-19: comparison to rt-pcr. *Radiology*, p. 200432.
- [Google, 2022] Google (2022). Classification: Roc curve and auc. <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>. Accessed: 02-11-2022.
- [He *et al.*, 2016] He, K., Zhang, X., Ren, S., y Sun, J. (2016). Deep residual learning for image recognition. En *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.
- [Inui *et al.*, 2020] Inui, S., Fujikawa, A., Jitsu, M., Kunishima, N., Watanabe, S., Suzuki, Y., Umeda, S., y Uwabe, Y. (2020). Chest ct findings in cases from the cruise ship “diamond princess” with coronavirus disease 2019 (covid-19). *Radiology. Cardiothoracic Imaging*, 2(2).
- [Koech, 2018] Koech, K. (2018). The basics of neural networks (neural network series) — part 1. <https://towardsdatascience.com/the-basics-of-neural-networks-neural-network-series-part-1-4419e343b2b>. Accessed: 10-04-2021.

- [Kwee y Kwee, 2020] Kwee, T. C. y Kwee, R. M. (2020). Chest ct in covid-19: What the radiologist needs to know. *RadioGraphics*, 40(7):1848–1865.
- [Kwiatkowski, 2021] Kwiatkowski, R. (2021). Gradient descent algorithm — a deep dive. <https://towardsdatascience.com/gradient-descent-algorithm-a-deep-dive-cf04e8115f21>. Accessed: 28-11-2022.
- [Mallapaty, 2020] Mallapaty, S. (2020). Where did covid come from? who investigation begins but faces challenges. <https://www.nature.com/articles/d41586-020-03165-9>. Accessed: 26-11-2020.
- [Morozov et al., 2020] Morozov, S. P., Andreychenko, A., Pavlov, N., Vladzymyrskyy, A., Ledikhova, N., Gomboleviskiy, V., Blokhin, I. A., Gelezhe, P., Gonchar, A., y Chernina, V. Y. (2020). Mosmeddata: Chest ct scans with covid-19 related findings dataset. *arXiv preprint arXiv:2005.06465*.
- [Mukherjee, 2022] Mukherjee, S. (2022). The annotated resnet-50. <https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758>. Accessed: 25-11-2022.
- [Piankyh, 2009] Piankyh, O. S. (2009). *Digital imaging and communications in medicine (DICOM): a practical introduction and survival guide*. Springer Science & Business Media.
- [RadiologyInfo, 2018] RadiologyInfo (2018). Computed tomography (ct) - chest. <https://www.radiologyinfo.org/en/info.cfm?pg=chestct>. Accessed: 21-12-2020.
- [Redsalud, 2020] Redsalud (2020). Exámenes covid-19. <https://www.redsalud.cl/nuestra-red/campanas/examenes-covid-19-disponibles>. Accessed: 26-11-2020.
- [Rubin et al., 2020] Rubin, Geoffrey D and Ryerson, Christopher J and Haramati, Linda B and Sverzellati, Nicola and Kanne, Jeffrey P and Raoof, Suhail and Schluger, Neil W and Volpi, Annalisa and Yim, Jae-Joon and Martin, Ian BK and others (2020). The role of chest imaging in patient management during the covid-19 pandemic: a multinational consensus statement from the fleischner society. *Chest*.
- [Saha, 2022] Saha, S. (2022). A comprehensive guide to convolutional neural networks - the eli5 way. <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>. Accessed: 17-10-2022.
- [Schaller et al., 2020] Schaller, T., Hirschbühl, K., Burkhardt, K., Braun, G., Trepel, M., Märkl, B., y Claus, R. (2020). Postmortem examination of patients with covid-19. *JAMA*.
- [Shung, 2018] Shung, K. P. (2018). Accuracy, precision, recall or f1? <https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9>. Accessed: 02-11-2022.

- [Song *et al.*, 2021] Song, Ying and Zheng, Shuangjia and Li, Liang and Zhang, Xiang and Zhang, Xiaodong and Huang, Ziwan and Chen, Jianwen and Wang, Ruixuan and Zhao, Huiying and Zha, Yunfei and others (2021). Deep learning enables accurate diagnosis of novel coronavirus (covid-19) with ct images. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.
- [Subramaniyan *et al.*, 2021] Subramaniyan, M., Sampathkumar, A., Jain, D. K., Ramachandran, M., Patan, R., y Kumar, A. (2021). Deep learning approach using 3d-impenn classification for coronavirus disease. *Artificial Intelligence and Machine Learning for COVID-19*, pp. 141–152.
- [Wang *et al.*, 2021] Wang, Shuai and Kang, Bo and Ma, Jinlu and Zeng, Xianjun and Xiao, Mingming and Guo, Jia and Cai, Mengjiao and Yang, Jingyi and Li, Yaodong and Meng, Xiangfei and others (2021). A deep learning algorithm using ct images to screen for coronavirus disease (covid-19). *European radiology*, pp. 1–9.
- [WHO, 2020] WHO (2020). Coronavirus disease (covid-19) pandemic. <https://www.who.int/emergencies/diseases/novel-coronavirus-2019>. Accessed: 26-11-2020.

ANEXO A

Gradiente Descendente

El gradiente descendente es un algoritmo de optimización iterativo utilizado para encontrar mínimos o máximos locales de una función. Es usado comúnmente en *machine learning* y *deep learning* para minimizar una función de costo.

Para su aplicación, las funciones deben ser:

- **Diferenciables:** una función es diferenciable si cuenta con una derivada en cada punto de su dominio.
- **Convexas:** una función es convexa si la línea que une a dos puntos se encuentra sobre su curva. Otra forma de comprobarlo es si la derivada de segundo orden de la función es mayor a 0. Si la función tiene mas de una variable es recomendable usar la matriz Hessiana para determinar si es convexa.

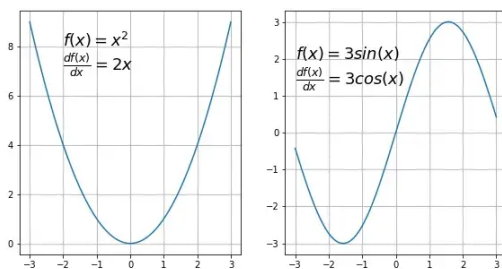


Figura A.1: Ejemplo de funciones diferenciables.

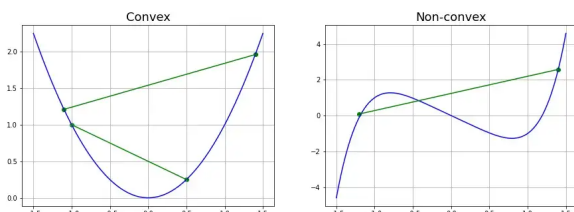


Figura A.2: Ejemplo de función convexa y no convexa.

Fuente: Towards Data Science [Kwiatkowski, 2021]

Un gradiente se define como la derivada en un punto de la función, en caso que tenga una sola variable, o un vector de derivadas donde cada componente indica la dirección en los ejes de las variables. Para el gradiente descendente se necesita la pendiente a lo largo de un eje por lo que se usan derivadas parciales.

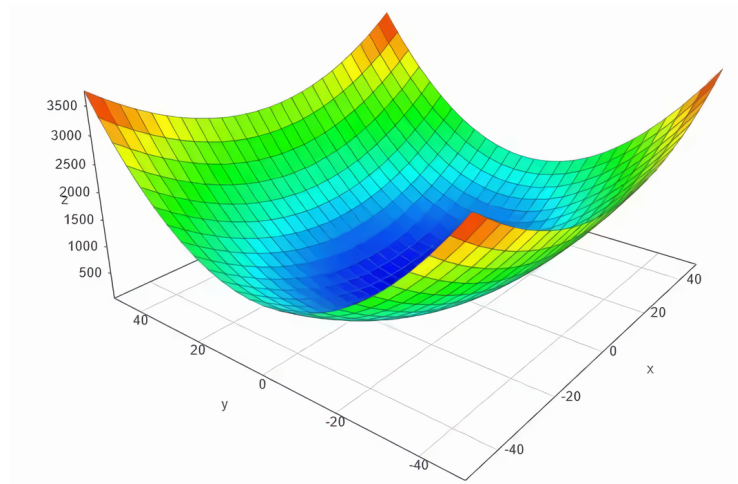


Figura A.3: Visualización del gradiente de la función $f(x, y) = 0,5x^2 + y^2$
Fuente: [Kwiatkowski, 2021].

Por lo tanto, el gradiente descendente calcula iterativamente el siguiente punto usando el gradiente en la posición actual escalado mediante un factor llamado tasa de aprendizaje α y restándoselo a la posición actual. Se hace una resta debido a que se quiere minimizar la función. Este proceso se resume en la siguiente ecuación:

$$\theta_{(t+1)} = \theta_t - \alpha \frac{\partial J(\theta_t)}{\partial \theta_t} \quad (\text{A.1})$$

Donde α corresponde a la tasa de aprendizaje y θ corresponde a uno de los pesos del modelo que se está optimizando.

La tasa de aprendizaje es relevante ya que controla el tamaño de los saltos que realiza el algoritmo y tiene un efecto importante en el rendimiento:

- Una tasa de aprendizaje pequeña conlleva a que el gradiente descendente tarde más tiempo en converger a un mínimo, incluso pudiendo agotar la cantidad de iteraciones antes de llegar a un punto óptimo.
- Una tasa de aprendizaje grande puede provocar que el gradiente descendente no converja al punto óptimo o incluso podría divergir.

Optimizador Adam

Adam es un algoritmo de optimización que extiende al gradiente descendente y se usa en la optimización de los modelos de *machine learning* y *deep learning*. Su nombre es un acrónimo de *adaptive moment estimation* y como su nombre lo indica, usa una combinación de *momentum* y tasa de aprendizaje adaptativa para converger de forma más rápida y eficiente a un punto óptimo.

El *momentum* corresponde a una estrategia para acelerar la velocidad de convergencia del gradiente descendente en escenarios de alta curvatura de la función objetivo o en presencia de gradientes inestables. Se incorpora en la ecuación del gradiente descendente de la siguiente forma:

$$\begin{aligned}v_{(t+1)} &= \gamma v_t + \alpha \frac{\partial J(\theta_t)}{\partial \theta_t} \\ \theta_{(t+1)} &= \theta_t - v_{(t+1)}\end{aligned}\tag{A.2}$$

Donde γ corresponde al parámetro del *momentum*, v corresponde a la velocidad y θ corresponde a uno de los pesos del modelo que se está optimizando.

Por otra parte, la tasa de aprendizaje adaptativa se refiere al proceso de actualizar el valor de la tasa de aprendizaje durante el entrenamiento del modelo.

Ejemplo de *ResNet-50*

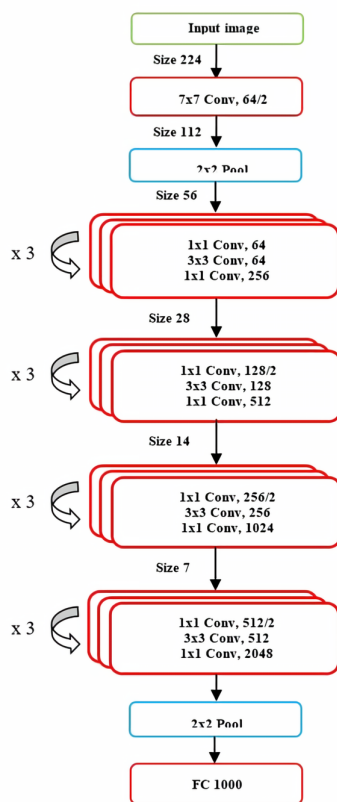


Figura A.4: Ejemplo de arquitectura de una *ResNet-50* para imágenes bidimensionales
Fuente: [Mukherjee, 2022].