Universitat Oberta de Catalunya

uoc.edu

Algoritmo de clasificación de lesiones en exámenes mamográficos

Joel Bustos
Máster en ciencia de datos
Ciencia de datos aplicada a la Salud

Sergi Martínez Maldonado
Àngels Rius Gavidia

02/01/2022

FICHA DEL TRABAJO FINAL

Título del trabajo:	Algoritmo de clasificación de lesiones en exámenes mamográficos.
Nombre del autor:	*Joel Bustos Pelegri*
Nombre del consultor/a:	*Sergi Martínez Maldonado*
Nombre del PRA:	*Àngels Rius Gavidia*
Fecha de entrega (mm/aaaa):	01/2022
Titulación:	*Máster en Ciencia de datos*
Área del Trabajo Final:	*Ciencia de datos aplicada a la salud*
Idioma del trabajo:	Español
Palabras clave	Deep Learning, Breast Cancer Diagnosis, Convolutional Networks

Resumen del Trabajo (máximo 250 palabras):.

El uso de técnicas de visión por computador aplicadas en el campo de la medicina, permiten acelerar el proceso de detección de cualquier tipo de enfermedad ayudando a los especialistas a la realización de diagnósticos y, por ende, reduciendo la tasa de mortalidad al detectar posibles sintomatologías durante etapas prematuras.

En concreto, las redes neuronales convolucionales forman parte del estado del arte en tareas de clasificación de imágenes gracias a que su arquitectura bidimensional se asemeja a la estructura de los datos de entrada.

En este trabajo, se han utilizado 4 arquitecturas de redes neuronales convolucionales para clasificar los distintos tipos de lesiones presentes en imágenes mamográficas como malignos o benignos.

Las decisiones tomadas por cada arquitectura han sido combinadas mediante un algoritmo *Random Forest* con el objetivo de emular el diagnóstico realizado por distintos especialistas a la hora de analizar un examen mamográfico.

La herramienta final generada a partir de la combinación secuencial de clasificadores, ha presentado métricas del 92 % para la clasificación de muestras benignas y malignas del set de datos *MIAS*.

Abstract (in English, 250 words or less):

The use of computer vision techniques applied in the field of medicine allows us to accelerate the process of detection of any type of disease, helping specialists to carry out diagnoses and reducing the mortality rate when detecting possible symptoms during premature stages.

Specifically, convolutional neural networks are part of the state-of-the-art in image classification tasks thanks to the fact that their two-dimensional architecture resembles the structure of the input data.

In this work, 4 convolutional neural network architectures have been used to classify the different types of lesions present in mammographic images, as malignant or benign.

The decisions made by each architecture have been combined using a Random Forest algorithm in order to emulate the diagnosis made by different specialists when analyzing a mammographic examination.

The final tool generated from the sequential combination of classifiers has presented metrics of 92% for the classification of benign and malignant samples from the MIAS data set.

Índice

Lista de figuras

Lista de tablas

Glosario

B

BIRADS: *Breast Imaging-Reporting and Data System* · 25
Bootstraping: Tecnica de muestreo con remplazo utilizada para realizar inferencia estadística. · 50

C

Capas de *pooling*: Capas presentes en una red neuronal convolucional cuyo objetivo consiste en reducir la dimensionalidad de los datos de entrada generando características más robustas y eficientes. Existen distintos tipos de *pooling* como por ejemplo, *max-pooling,* encargado de obtener el valor máximo de cada entrada. · 26
CBIS-DDSM: Curated Breast Imaging subset of DDSM · 22; *Curated Breast Imaging Subset of Digital Database for Screening Mammography* · 24
CNN: *Red Neuronal Convolucional* · 22

D

Data augmentation: Aumento sintético de datos. Se generan muestras de datos nuevas a partir de las muestras originales. · 16
Data mining: Minería de datos. Area científica que pretende extraer conocimiento útil a través de los datos. · 21
Deep Learning: Aprendizaje profundo. · 12
DICOM: *Digital Imaging and Communications in Medicine* · 25

F

FC: Fully-Connected. Capa de un modelo de deep learning caracterízado por que todas los parámetros presentes (neuronas) están interconnectados con todos los parámetros de la capa sucesora o predecesora. · 43
Fine tunning: Técnica deribada de la transferencia de aprendizaje dónde se realiza un ajuste fino de los parámetros de un modelo de *Deep Learning* al dominio de datos sobre el que se está aplicando. · 16

H

Hiperparámetros: Valores configurados durante la implementación de un modelo de minería de datos independientes de los datos utilizados. · 12

I

Idle: Integrated DeveLopment Environment para Python. · 12

L

Learning rate: Tasa de aprendizaje. Parámetro que determina la velocidad de actualización de los pesos de un modelo de *Deep Learning* durante el paso de *backpropagation.* · 48

M

MIAS: Mammographic Image Analysis Society · 21, 22, 23

O

Open source: Software de código abierto que permite su uso bajo una licencia de código abierto. · 12

P

Padding: Técnica en la cual se mantiene la dimensión de entrada de una observación durante las operaciones de pooling o de convolución presentes en una red neuronal convolucional. Para ello, se añaden píxeles con valor 0 en los margenes de cada entrada. · 26

Parameter tunning: Ajuste de los parámetros de un modelo a partir de los datos de entrenamiento utilizados. · 16

R

Relu: Rectiffied Linear Unit. Función de activación cuya salida es lineal para valores positivos y cero para valores negativos. · 26

ROI: *Region Of Interes. Zona de interés de una mamografía que contiene la lesión cancerígena.* · 21

S

Stride: Número de píxeles presentes entre operaciones de convolución consecutivas en las capas convolucionales de una arquitectura de red. · 26

SVM: Máquina de soporte vectorial · 20

T

Transfer learning: Transferencia de aprendizaje. Consiste en utilizar el conocimiento aprendido en un dominio de *deep learning* aplicados a otros dominio similar. · 12

1. Introducción

1.1 Contexto y justificación del Trabajo

El cáncer de seno, también conocido como cáncer de mama, destaca por ser el tipo de cáncer más común entre las personas. Según la Organización Mundial de la Salud, una de cada doce mujeres padece esta patología a lo largo de su vida [1]. Adicionalmente, el cáncer de seno es una de las principales causas de muerte producidas por cáncer. En 2012, fue el causante del 14.7 % del total de muertes entre mujeres, alcanzando el medio millón de defunciones [2]. En 2020, se diagnosticaron más de 2.2 millones de casos y un total de 685.000 defunciones. Siguiendo esta tendencia, si se detectasen anualmente 2.2 millones de casos nuevos, en el año 2030 habrán más de 23 millones de personas afectadas [3] y más de 4.6 millones de muertes [4]. Este escenario plantea la necesidad de hacer frente a esta enfermedad con el objetivo de reducir su alta tasa de mortalidad. Para ello, es de vital importancia poder diagnosticar y tratar esta patología durante sus etapas más tempranas, aumentando la probabilidad de supervivencia de las personas afectadas hasta tasas superiores al 90 %.

El proceso de detección y diagnóstico de cáncer de seno consta de distintas fases. En la primera, a partir de chequeos regulares o palpaciones, se busca cualquier tipo de anomalía nueva en la zona del seno como, por ejemplo, nódulos o masas que antes no estaban presentes. Adicionalmente, en esta primera etapa, también se realizan exámenes médicos como mamografías o ecografías con el objetivo de encontrar cualquier sintomatología no perceptible físicamente [5]. Recientemente, se están empezando a utilizar técnicas más novedosas como la tomosíntesis digital. Esta técnica, más costosa que una mamografía tradicional, consiste en realizar mamografías tridimensionales permitiendo aumentar las posibilidades de detección de cáncer de seno [5].

La segunda fase del diagnóstico empieza una vez detectada la posibilidad de padecer cáncer de seno. En esta etapa, se realiza una biopsia de tejido extirpando una muestra y tintándola con Hematoxilina y Eosina (H&E). Este proceso de coloreado facilita el análisis de las células del tejido al permitir diferenciar claramente sus estructuras; tintando el núcleo de color púrpura y, el citoplasma, de rosa [6].

Actualmente, el uso de mamografías es considerado el método más efectivo a la hora de detectar cáncer de seno durante sus etapas más tempranas [7]. Sin embargo, el análisis de este tipo de imágenes por parte de un especialista, supone una tarea costosa y propensa a errores debido al gran tamaño de las muestras a analizar y a la variabilidad de formas presentes en las patologías.

Las probabilidades de error por parte de un especialista durante el diagnóstico de esta enfermedad, pueden aumentar, además, por factores externos como la fatiga o la experiencia. Algunos estudios muestran que casi el 30 % de los casos

de cáncer diagnosticados podrían haberse encontrado en exámenes mamográficos clasificados previamente como negativos [8]. En concreto, la tasa de especificidad y de sensibilidad en la clasificación de cáncer de seno, puede variar entre el 89 % y el 97 % y el 77 % y el 87 %, respectivamente.

En vista de la variabilidad presente, es fundamental desarrollar sistemas de soporte automáticos que ayuden a los especialistas a la hora de determinar si una mamografía contiene una patología benigna o maligna, con el objetivo de reducir la tasa de error y de ganar agilidad durante el diagnóstico.

1.2 Objetivos del Trabajo

Ante la problemática planteada en la sección "*1.1 Contexto y justificación del Trabajo"*, uno de los principales objetivos del proyecto consistirá en desarrollar un sistema de ayuda automático que permita reducir la tasa de error a la hora clasificar cualquier patología presente en una mamografía. Este sistema automático será gratuito y sin ánimo de lucro, con el objetivo de ayudar en la lucha contra el cáncer de seno. Adicionalmente, el aplicativo no necesitará de dependencias y podrá ser ejecutado desde cualquier ordenador.

En este aspecto, la aplicación desarrollada podrá ser utilizada por los especialistas como herramienta de soporte, ayudando en el proceso de clasificación de patologías y reduciendo la carga de trabajo de cada uno. Cabe destacar que, debido al sesgo intrínseco de los datos utilizados, el aplicativo no persigue el objetivo de sustituir la decisión tomada por un especialista.

Por otra parte, este trabajo pretende implementar, comparar y analizar distintos algoritmos de inteligencia artificial que compongan el estado del arte en tareas de clasificación de imágenes médicas.

Finalmente, dada la importancia de la causa que persigue la finalidad del proyecto, se pretende que el código desarrollado durante el transcurso del mismo esté disponible públicamente para que otros investigadores puedan aprovecharlo.

1.3 Enfoque y método seguido

Para lograr los objetivos planteados en el presente documento, se ha seguido una metodología de trabajo ágil centrada en desarrollo de *software*. De este modo, el proyecto se ha planificado en un total de 9 *sprints* de 2 semanas de duración. En la sección "*1.4.3 Planificación inicial del proyecto"* se puede encontrar el detalle de los objetivos y las tareas a realizar en cada uno de los *sprints*.

Cabe destacar que un aspecto fundamental de las metodologías de trabajo ágiles es que son adaptativas en vez de reactivas, por lo que algunos de los objetivos prefijados podrían sufrir cambios y modificaciones en función de las necesidades emergentes en cada momento. Adicionalmente, dada la naturaleza de todo proyecto de minería de datos, existe una interacción bidireccional entre

las distintas fases que lo componen, hecho que refuerza el uso de una metodología ágil para el desarrollo de este proyecto.

En cuanto a las distintas etapas que compondrán el proyecto, estas han estado determinadas por la metodología CRISP-DM (*Cross Industry Standard Process for Data Mining)* tal y como se muestra en la Figura 1.

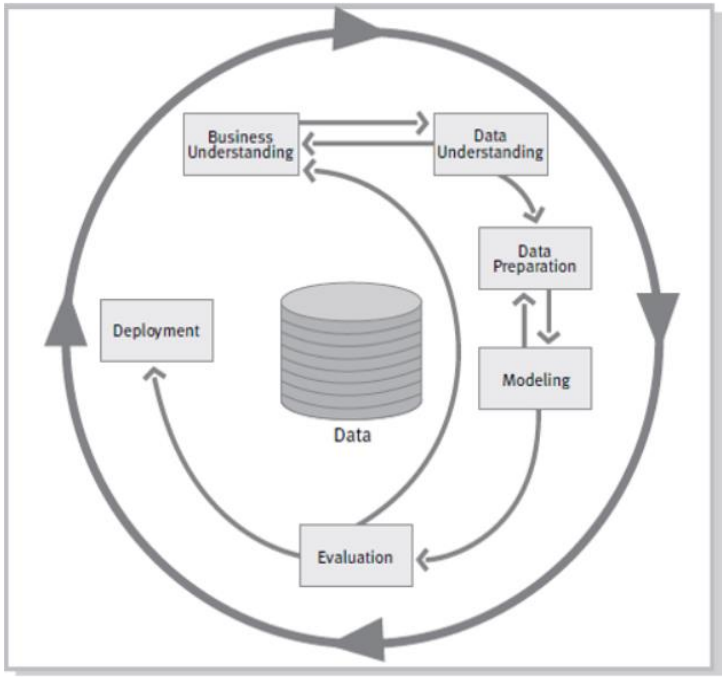


Figura 1*. Flujo de trabajo CRIPS-DM utilizado para la implementación del proyecto*

Por otra parte, para conseguir los objetivos planteados en el proyecto, se han utilizado implementaciones *open source* que componen el estado del arte en tareas de clasificación médica. Para ello, se han utilizado técnicas de *transfer learning* que permiten ajustar los hiperparámetros de cada red al conjunto de datos utilizado durante el proyecto.

En cuanto a las herramientas relacionadas con la implementación del proyecto, se ha utilizado el lenguaje de programación *Python* dada la gran variedad de librerías disponibles que permiten ganar versatilidad y agilidad a la hora de crear y modificar algoritmos de *Deep Learning* (librerías *Keras*, *Pytorch* o *Tensorflow)*; crear interfaces gráficas de usuario (librerías *PyQt5* o *Tkinter)* o crear aplicaciones independientes (librerías *Pyinstaller* o *Py2exe).*

Por otra parte, para la realización del versionado de código se ha utilizado *GitHub* persiguiendo el objetivo de que, bajo la licencia de este proyecto, otros autores puedan aprovechar el trabajo realizado con el fin de ayudar en la lucha contra el cáncer de seno. Finalmente, el *idle* de programación utilizado es *PyCharm v.2020.1.*

1.4 Planificación del Trabajo

1.4.1 Coste del proyecto

Como el objetivo principal de este proyecto persigue la construcción de una herramienta de *software* que permita realizar tareas de clasificación de imágenes médicas, quedando fuera del alcance la creación de cualquier tipo de *hardware* como prototipos o máquinas, el presupuesto del proyecto debe de cubrir, principalmente, los costes temporales utilizados por los intervinientes a la hora de implementar el algoritmo. De esta forma se pretenden financiar las horas de desarrollo de código, de toma de decisiones, de investigación, de reuniones, etc.

Por otra parte, las herramientas utilizadas para el desarrollo del proyecto son mayoritariamente *open source* a excepción del *idle* de programación que requiere de una licencia anual.

En este aspecto, teniendo en cuenta que la duración del proyecto es de unas 300 horas (equivalente a 12 créditos ECTS [9]), el presupuesto necesario se compone por:

- El salario de un *data scientist*, siendo este aproximadamente de 33.000€ anuales (2080 horas) [10], equivalente a 4.760€ según la duración del proyecto.

- El salario de un *supervisor del proyecto* o *Project manager,* siendo este aproximadamente de 44.000€ anuales (2080 horas) [11], equivalente a 6.346€ según la duración del proyecto.

- Licencia anual de *Pycharm* de 89€ [12].

En consecuencia, el coste del proyecto asciende a un total de 11.195 €.

1.4.2 Impacto medioambiental

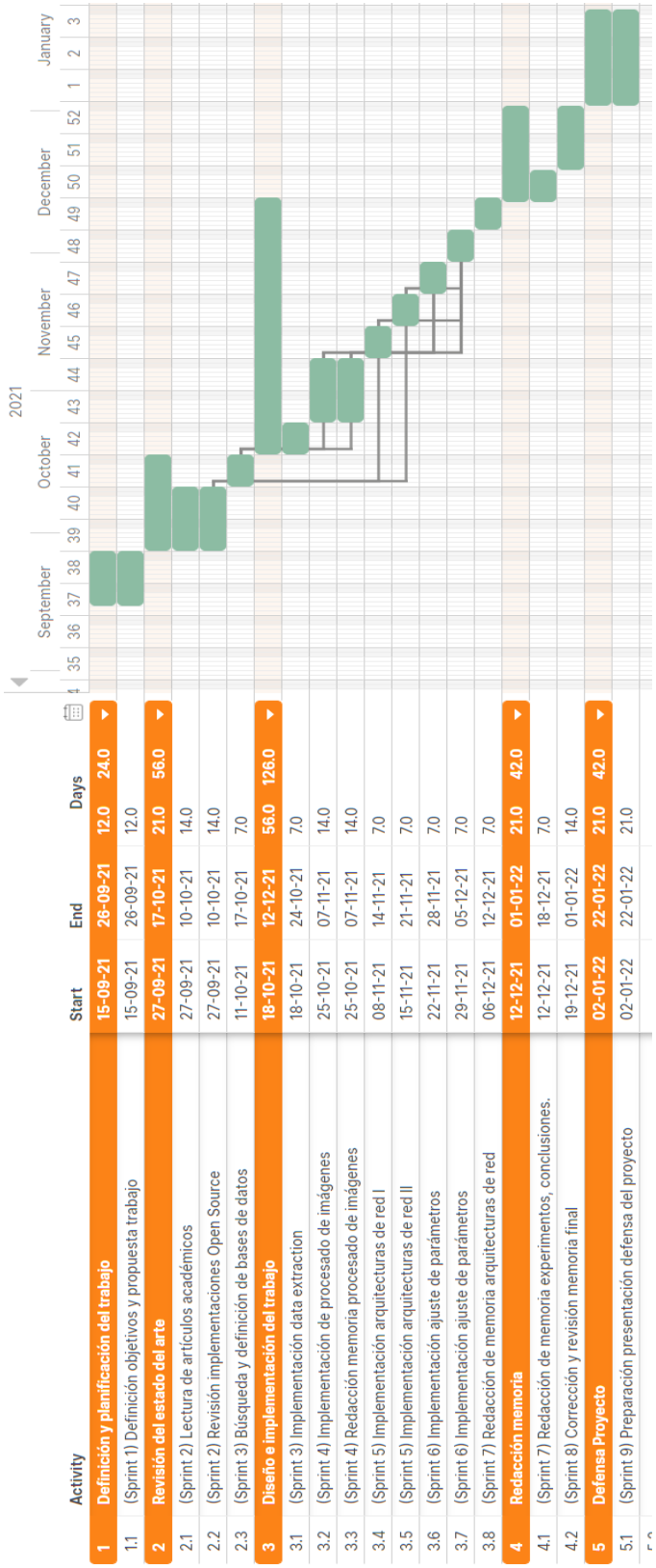
Para medir el impacto medioambiental que supondrá la realización de este proyecto, será necesario conocer la cantidad de CO_2 generado durante la creación de la herramienta de clasificación de lesiones de seno.

El consumo eléctrico del ordenador *MSI-GV62 7RD* con tarjeta gráfica *GeForce GTX1050* que será utilizado para el desarrollo de *software,* es de 0.15 KwH [13]. Adicionalmente, el proyecto tendrá una duración aproximada de 300 horas siendo necesarias 100 horas adicionales para la ejecución de los modelos de inteligencia artificial. En consecuencia, el consumo eléctrico total estimado asciende a 60 KwH.

En Cataluña, se considera que se producen un total de 321 gramos de CO_2 por kilovatio generado [14]. En consecuencia, la huella de carbono generada por este proyecto será de 19.26 kilogramos.

1.4.3 Planificación inicial del proyecto

La planificación inicial del proyecto se muestra en la Figura 2, mediante un diagrama de *Gantt* con los objetivos principales que englobarían los distintos *sprints,* así como sus entregables y tareas globales a realizar.



	Activity	Start	End	Days	
1	**Definición y planificación del trabajo**	**15-09-21**	**26-09-21**	**12.0**	**24.0**
1.1	(Sprint 1) Definición objetivos y propuesta trabajo	15-09-21	26-09-21	12.0	
2	**Revisión del estado del arte**	**27-09-21**	**17-10-21**	**21.0**	**56.0**
2.1	(Sprint 2) Lectura de artículos académicos	27-09-21	10-10-21	14.0	
2.2	(Sprint 2) Revisión implementaciones Open Source	27-09-21	10-10-21	14.0	
2.3	(Sprint 3) Búsqueda y definición de bases de datos	11-10-21	17-10-21	7.0	
3	**Diseño e implementación del trabajo**	**18-10-21**	**12-12-21**	**56.0**	**126.0**
3.1	(Sprint 3) Implementación data extraction	18-10-21	24-10-21	7.0	
3.2	(Sprint 4) Implementación de procesado de imágenes	25-10-21	07-11-21	14.0	
3.3	(Sprint 4) Redacción memoria procesado de imágenes	25-10-21	07-11-21	14.0	
3.4	(Sprint 5) Implementación arquitecturas de red I	08-11-21	14-11-21	7.0	
3.5	(Sprint 5) Implementación arquitecturas de red II	15-11-21	21-11-21	7.0	
3.6	(Sprint 6) Implementación ajuste de parámetros	22-11-21	28-11-21	7.0	
3.7	(Sprint 6) Implementación ajuste de parámetros	29-11-21	05-12-21	7.0	
3.8	(Sprint 7) Redacción de memoria arquitecturas de red	06-12-21	12-12-21	7.0	
4	**Redacción memoria**	**12-12-21**	**01-01-22**	**21.0**	**42.0**
4.1	(Sprint 7) Redacción de memoria experimentos, conclusiones.	12-12-21	18-12-21	7.0	
4.2	(Sprint 8) Corrección y revisión memoria final	19-12-21	01-01-22	14.0	
5	**Defensa Proyecto**	**02-01-22**	**22-01-22**	**21.0**	**42.0**
5.1	(Sprint 9) Preparación presentación defensa del proyecto	02-01-22	22-01-22	21.0	

Figura 2*. Diagrama de Gantt mostrando las fases del proyecto organizadas por sprints*

Finalmente, en las tablas mostradas a continuación se describen, de una manera más detallada, las entregas y tareas a realizar en cada sprint.

Sprint 1		**ID:** 1.1
Mayor constituyente: Documentación	**Inicio:** 15/09/2021	**Fin:** 26/09/2021
Descripción: Documentación de las bases del proyecto, objetivos, metodología y plan del proyecto.		
Tareas: 1. Descripción del proyecto y definición objetivos. 2. Plan de proyecto.		
Entregables: Algoritmo de clasificación de lesiones en mamografías.docx		

Tabla 1*. Planificación del sprint 1 detallando los entregables, la duración y los plazos.*

Sprint 2		**ID:** 2.1, 2.2
Mayor constituyente: Investigación	**Inicio:** 27/09/2021	**Fin:** 10/10/2021
Descripción: Revisión del estado del arte en tareas de clasificación de imágenes médicas con finalidad de diagnosticar cáncer de seno. Se pretende, además, revisar aquellas metodologías *open source.*		
Tareas: 1. Revisión artículos académicos 2. Búsqueda estructuras de red *open source* utilizadas.		
Entregables: Documento con las técnicas, metodologías y arquitecturas a utilizar basadas en el estado del arte actual.		

Tabla 2. *Planificación del sprint 2 detallando los entregables, la duración y los plazos*

Sprint 3		**ID:** 2.3, 3.1
Mayor constituyente: Código	**Inicio:** 11/10/2021	**Fin:** 24/10/2021
Descripción: Revisión de las bases de datos utilizadas en los artículos del estado del arte con la finalidad de obtener una base de datos para el proyecto.		
Tareas: 1. Búsqueda de bases de datos. 2. Implementación de la extracción de datos y unión de datos en caso de utilizar diversas fuentes de imágenes.		
Entregables: Código Python para la creación del conjunto de datos.		

Tabla 3. *Planificación del sprint 3 detallando los entregables, la duración y los plazos*

Sprint 4		**ID:** 3.2, 3.3
Mayor constituyente: Código y documentación.	**Inicio:** 25/10/2021	**Fin:** 08/11/2021
Descripción: Aplicación de técnicas de preprocesado de datos y de *data augmentation* para poder entrenar un algoritmo de *Deep Learning*.		
Tareas: 1. Implementación del procesado de datos. 2. Implementación de la *data augmentation*. 3. Redacción teórica del procesado de datos para la memoria final.		
Entregables: Código Python para el procesado del conjunto de datos.		

Tabla 4. *Planificación del sprint 4 detallando los entregables, la duración y los plazos*

Sprint 5		**ID:** 3.4, 3.5
Mayor constituyente: Código	**Inicio:** 08/11/2021	**Fin:** 21/11/2021
Descripción: Implementación de 4 arquitecturas de red y realizar pequeños experimentos sobre parte de los datos.		
Tareas: 1. Implementación arquitectura de red 1 2. Implementación arquitectura de red 2 3. Implementación arquitectura de red 3 4. Implementación arquitectura de red 4		
Entregables: Código Python con las arquitecturas implementadas.		

Tabla 5. *Planificación del sprint 5 detallando los entregables, la duración y los plazos*

Sprint 6		**ID:** 3.6, 3.7
Mayor constituyente: Código	**Inicio:** 22/11/2021	**Fin:** 05/12/2021
Descripción: Realización del ajuste de parámetros de las arquitecturas de red creadas en el *sprint* 5 para la implementación del algoritmo de combinación de modelos. Una vez generado el pipeline de clasificación completo, se debe de crear una aplicación independiente que permita obtener la clasificación de una imagen introducida.		
Tareas: 1. *Parameter tunning* y *fine tunning* de las redes. 2. Combinación de modelos. 3. Creación aplicación.		
Entregables: Aplicación para la clasificación de mamografías.		

Tabla 6. *Planificación del sprint 6 detallando los entregables, la duración y los plazos*

Sprint 7		**ID:** 3.8, 4.1
Mayor constituyente: Documentación	**Inicio:** 06/12/2021	**Fin:** 12/12/2021
Descripción: Finalización de la memoria del trabajo final a partir de la implementación realizada en el *sprint* 6 detallando los experimentos realizados, los resultados y sus conclusiones.		
Tareas: 1. Redacción memoria final.		
Entregables: Apartados de metodología y recursos de la memoria final.		

Tabla 7. *Planificación del sprint 7 detallando los entregables, la duración y los plazos*

Sprint 8		**ID:** 4.2
Mayor constituyente: Documentación	**Inicio:** 12/12/2021	**Fin:** 19/12/2021
Descripción: Revisión y corrección de los errores cometidos durante la generación del documento. Adicionalmente, este sprint servirá para recuperar hitos no conseguidos en alguno de los *sprints* de documentación previos.		
Tareas: 1. Redacción memoria final.		
Entregables: Algoritmo de clasificación de lesiones en mamografías.pdf		

Tabla 8. *Planificación del sprint 8 detallando los entregables, la duración y los plazos*

Sprint 9		**ID:** 4.3
Mayor constituyente: Documentación	**Inicio:** 03/01/2022	**Fin:** 21/01/2022
Descripción: Preparación de la defensa del trabajo y realización de la defensa.		
Tareas: 1. Creación del guion y presentación *power point* para la defensa del proyecto 2. Realización defensa.		
Entregables: Algoritmo de clasificación de lesiones en mamografías.ppt		

Tabla 9. *Planificación del sprint 9 detallando los entregables, la duración y los plazos*

1.5 Breve sumario de productos obtenidos

Al finalizar el proyecto, tal y como se ha descrito en la sección "*1.2 Objetivos del Trabajo*", se ha obtenido un *software* que, dada una imagen médica de un seno juntamente con anotaciones referentes a la zona de interés, es capaz de

determinar si se trata de una patología benigna o maligna juntamente con su probabilidad.

Esta implementación puede ser ejecutada desde cualquier ordenador sin la necesidad de tener nada instalado, siendo exclusivamente necesario el ejecutable. A continuación, en la Figura 3, se puede observar la interfaz gráfica de la herramienta generada.

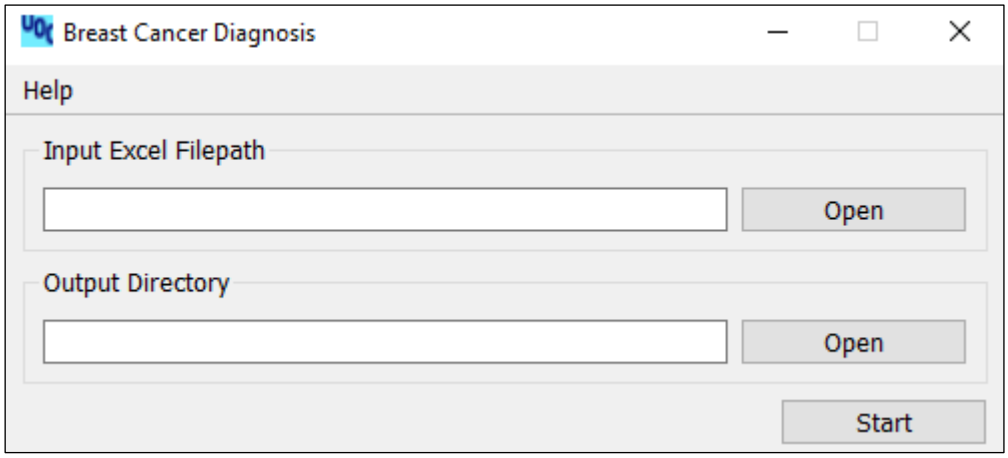


Figura 3. *Interfaz gráfica de la herramienta breast cancer diagnosis generada en el proyecto.*

Para ejecutar la aplicación, es necesario introducir un Excel (entrada *Input Excel Filepath* de la Figura 3) que contenga las siguientes columnas:

Campo	Descripción
ID	Identificador único de cada imagen
FILE_PATH	Ruta de la imagen a clasificar.
X_CORD	Coordenada x del centro de la patología expresada en píxeles.
Y_CORD	Coordenada y del centro de la patología expresada en píxeles.
RAD	Radio del círculo que contiene la patología en píxeles.

Tabla 10. *Campos requeridos en el Excel de entrada de la herramienta.*

Una vez introducido el fichero Excel, el programa generará en la carpeta especificada por el usuario (campo *Output Directory* de la Figura 3) un fichero Excel mostrando la clasificación de cada imanen (Figura 4).

B	F	G	H	I	J
FILE_PATH	X_CORD	Y_CORD	RAD	PATHOLOGY	MALIGNANT PROBABILITY
C:\Users\USUARIO\Desktop\Master Data Science\TFM\Code\data\00_RAW\MIAS\ALL\mdb315.pgm	516	447	93	BENIGN	18,46
C:\Users\USUARIO\Desktop\Master Data Science\TFM\Code\data\00_RAW\MIAS\ALL\mdb314.pgm	518	191	39	BENIGN	34,87
C:\Users\USUARIO\Desktop\Master Data Science\TFM\Code\data\00_RAW\MIAS\ALL\mdb312.pgm	240	263	20	BENIGN	5,99
C:\Users\USUARIO\Desktop\Master Data Science\TFM\Code\data\00_RAW\MIAS\ALL\mdb290.pgm	337	353	45	MALIGNANT	69,29
C:\Users\USUARIO\Desktop\Master Data Science\TFM\Code\data\00_RAW\MIAS\ALL\mdb274.pgm	127	505	123	MALIGNANT	98,36
C:\Users\USUARIO\Desktop\Master Data Science\TFM\Code\data\00_RAW\MIAS\ALL\mdb271.pgm	784	270	68	BENIGN	87,48

Figura 4. *Ejemplo del Excel de salida generado por el aplicativo. Las columnas en amarillo muestran las predicciones realizadas por el programa (Tipo de cáncer y probabilidad de cancer maligno).*

Por otra parte, juntamente con el Excel de salida, se generarán dos carpetas conteniendo las zonas de interés de cada mamografía clasificadas según su tipología (Figura 5).

Para poder tener trazabilidad de las ejecuciones realizadas, el programa genera un conjunto de archivos de texto que describen posibles errores durante las fases de procesado de las imágenes, así como el *log* de ejecuciones del aplicativo.

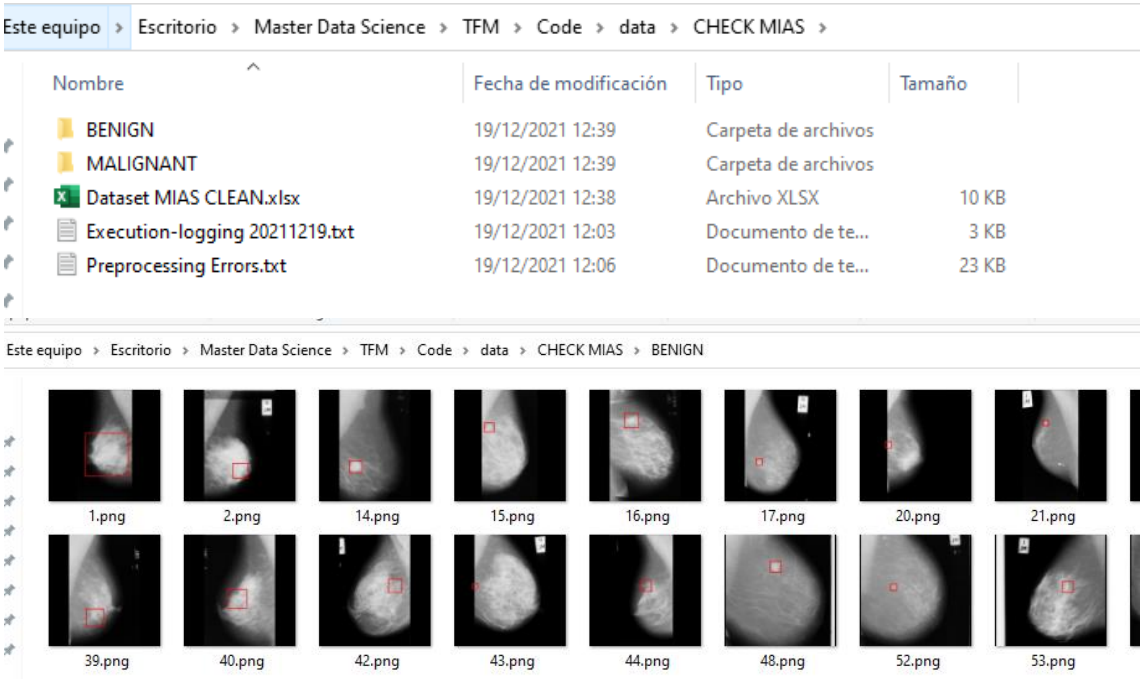


***Figura 5**: Ejemplo de los documentos de salida generados por el programa. Las carpetas BENIGN y MALIGNANT contiene las imágenes clasificadas, remarcando las zonas de interés (ROI) en rojo.*

Adicionalmente, junto con el software elaborado, se comparte el código desarrollado para la implementación de esta herramienta en el siguiente repositorio: https://github.com/jbustospelegri/breast_cancer_diagnosis.

Finalmente, como último producto generado, se ha escrito una memoria que recoge la metodología, las técnicas y arquitecturas desarrolladas durante la realización del proyecto.

1.6 Breve descripción de los otros capítulos de la memoria

El presente documento se estructura de la siguiente manera: En el apartado "*2. Estado del arte"* se expone el estado del arte presente en tareas de clasificación médica explicando qué técnicas han sido utilizadas a lo largo del tiempo para la clasificación de cáncer de seno. En el siguiente apartado, "*3. Materiales y Recursos",* se muestran las bases de datos utilizadas, así como las arquitecturas de red empleadas para la clasificación de patologías. Una vez finalizado este apartado, en la sección "*4. Metodología"* se expondrán las metodologías de procesado de datos y de entrenamiento de modelos realizadas durante el transcurso del proyecto. Finalmente, los dos últimos apartados contendrán los experimentos realizados juntamente con los resultados obtenidos ("*5. Experimentos y resultados")* y las conclusiones del trabajo ("*6. Conclusiones*").

2. Estado del arte

El análisis de exámenes mamográficos constituye una de las metodologías más recomendadas para poder diagnosticar la existencia de cáncer de seno durante las primeras etapas de la enfermedad. Su éxito recae principalmente en la posibilidad de identificar cualquier sintomatología mucho antes de que esta se exprese físicamente.

Para poder diagnosticar cualquier sintomatología a partir de una mamografía, los especialistas analizan distintas características de la imagen como, por ejemplo, la aparición de calcificaciones (depósitos de calcio dentro del tejido mamario que se identifican como pequeñas manchas blancas) o la presencia de masas (áreas más grandes y densas de lo normal, cuya forma discierne de la habitual) [15].

Debido a la dificultad y al alto número de factores a examinar, el análisis de mamografías es un proceso minucioso y extenso, cuyo éxito depende en gran parte de la experiencia del radiólogo. Adicionalmente, factores como la calidad de la mamografía o la densidad del seno pueden empeorar y limitar el análisis del tejido mamario al ocultar la presencia de zonas anormales. Por este motivo, el uso de distintos planos dentro de un mismo examen como, por ejemplo, el plano Oblicuo Medio Lateral (MLO) o el plano Cráneo Caudal (CC), podría incrementar hasta un 25 % la tasa de éxito del diagnóstico [16]. Aun así, el error promedio cometido por los radiólogos a la hora de identificar una patología se encuentra entorno al 30 % ([17], [18]).

En las últimas décadas se han empezado a utilizar sistemas de diagnóstico asistidos por ordenador (CAD) que pretenden ayudar a los especialistas durante el análisis de las mamografías. El objetivo de estos sistemas persigue agilizar el proceso de análisis y reducir la tasa de error a la hora de clasificar las posibles patologías presentes en los senos.

En primera instancia, los primeros sistemas CAD se basaban en técnicas de *machine learning* como árboles de decisión o máquinas de soporte vectorial (SVM). Fatih, M. et al. [19] utilizó una máquina de soporte vectorial juntamente con un algoritmo de selección de características para clasificar las muestras del set de datos de cáncer de mama de Wisconsin (*WBCD*) en benignas o malignas. El modelo presentado obtuvo una exactitud del 99.51 % utilizando únicamente 5 de las 9 características que componían el set de datos. Otros autores presentaron distintos trabajos sobre el mismo set de datos. Quinlan, J. et al. [20] utilizó un árbol de decisión C4.5 alcanzando una exactitud del 94.75 %. Abonyi, J. et al. [21], utilizó un algoritmo de segmentación confuso obteniendo una exactitud de 95.57 %. Lam, S. et al. [22], utilizó un algoritmo de *K-means* para obtener características similares entre los tumores benignos y malignos. La segmentación obtenida, la utilizó conjuntamente con el resto de atributos del set de datos para entrenar una máquina de soporte vectorial logrando una exactitud del 97.38 %. Un método muy similar fue utilizado por Hue-Ling, C. et al. [23] para clasificar las observaciones del set de datos en malignas o benignas. En este estudio, el autor utilizó una máquina de soporte vectorial *RS* para eliminar

aquellas características redundantes y, posteriormente, entrenar una máquina de soporte vectorial. El modelo resultante obtuvo una exactitud del 96.72 %.

Cabe remarcar que todos los sistemas de ayuda automático expuestos hasta el momento fueron desarrollados sobre un conjunto de datos formado por características extraídas directamente de las mamografías por especialistas. En concreto, atributos como la densidad del núcleo de las células, el perímetro o el área, formaban parte del set de datos [24]. En este sentido, los sistemas de ayuda implementados hasta la fecha, aunque conseguían decrementar la tasa de error cometida por los especialistas, no agilizaban el diagnóstico de patologías. Con el objetivo de agilizar dicho proceso, se empezaron a crear sistemas CAD cuyos algoritmos de *data mining* se entrenaban directamente sobre las imágenes mamográficas. El estado del arte para generar esta nueva implementación se dividía en dos fases: en la primera, un algoritmo de segmentación extraía información útil de las imágenes, eliminando datos irrelevantes y redundantes; en la segunda fase, las características resultantes eran utilizadas para entrenar un algoritmo de clasificación.

En primera instancia, Karssemeijer, N. et al. [25] creó un método para clasificar las imágenes de la base de datos *The Netherlands* en malignas y benignas. Su algoritmo estaba dividido en dos etapas. En la primera, un modelo de segmentación compuesto por *kernels gaussianos* extraía las zonas de interés (ROI) de las mamografías. Las imágenes resultantes eran combinadas, en la segunda fase, con características calculadas a partir de las imágenes originales y con información adicional del paciente, para alimentar una máquina de soporte vectorial. El modelo propuesto alcanzó un área bajo la curva del 91 %.

Por otra parte, Alhanahnah, M. et al. [26] procesaba las imágenes de entrada mediante una transformación discreta *Wavelet*. La salida de la imagen procesada era introducida en un regresor lineal obteniendo un área bajo la curva del 91.34 %. Adicionalmente, Chakraborty, D. et al. [27] utilizó técnicas de detección de bordes, métodos difusos, métodos de localización y métodos de *thresholding* para extraer características de las imágenes. Estas, fueron utilizadas en diversos modelos de *machine learning* como: *SVM, Random Forest o Nearest Neighbors* sin lograr alcanzar una exactitud superior al 90 %.

Finalmente, Mostafa, H. et al. [28] implementó un método de clasificación para el set de datos *Mammographic Image Analysis Society* (MIAS). Para ello, se propuso de nuevo un modelo basado en dos fases. En la primera etapa, se extraían características de las imágenes mediante la obtención de patrones locales binarios (*LBP*) y, posteriormente en la segunda fase, se alimentaba un algoritmo de clasificación *Random Forest.* El autor logró una exactitud del 97 % a la hora de clasificar las imágenes en malignas y benignas.

Cabe destacar que la efectividad de los sistemas CAD basados en técnicas de *machine learning* depende en gran parte del set de datos utilizado en el momento de entrenar los modelos. La colaboración de los especialistas, pues, compone un aspecto crucial a la hora de definir qué características serán útiles para realizar una correcta clasificación de patologías. En este sentido, existe la posibilidad de que los modelos generados contengan, de forma intrínseca, un

sesgo producido por el conocimiento de los especialistas a la hora de determinar qué características definen una determinada patología u otra [25].

En los últimos años, la aparición del aprendizaje profundo a través del uso de redes neuronales convolucionales (*Convolutional Neural Networks, CNN)* ha ido tomando especial relevancia en tareas de visión por computador dentro del campo de la medicina. Por ejemplo, en [29]–[31] se muestran un conjunto de sistemas CAD basados en aprendizaje profundo que han resultado ser exitosos en tareas de predicción y clasificación de patologías de seno.

La principal ventaja que muestran estos sistemas basados en redes neuronales convolucionales es la capacidad de aprender características directamente de los datos de entrada, aprovechando la estructura 2D presente en las mamografías a través de las capas convolucionales. Adicionalmente, este tipo de métodos componen sistemas *end-to-end* basados en los datos de entrada, permitiendo reducir el sesgo humano reflejado a partir de la definición de descriptores complejos y derivados del conocimiento de campo para poder detectar patologías.

El primer intento de implementar una *CNN* en tareas de clasificación de mamografías la realizó Sahinner, B. et al [32]. La estructura de red propuesta estaba constituida únicamente por dos capas ocultas, obteniendo unos resultados de clasificación no muy favorables. Más tarde, la aparición de redes neuronales convolucionales más profundas y complejas, así como la recopilación de grandes conjuntos de datos y el uso de métodos que permitían optimizar el proceso de entrenamiento, permitieron mejorar la actuación de los modelos de aprendizaje profundo en tareas de clasificación de imágenes médicas.

Li, H. et al. [33] propuso una arquitectura de red *DenseNet-II* para clasificar mamografías en benignas y malignas. Shen, L. et al. [34], utilizó las arquitecturas de red *VGG19* y *ResNet* para clasificar las bases de datos *Curated Breast-Imaging subset of DDSM* (*CBIS-DDSM*) y *Inbreast*, alcanzando un área bajo la curva del 91 % y 98 % respectivamente. S.A, Agnes. et al. [35], desarrolló un modelo de red *multiscale all convolucional neural network (MA-CNN)* en el cual se aplicaba un mayor *stride* en las capas convolucionales para retener la información de los píxeles vecinos. El modelo obtuvo una exactitud del 96.47 % en el set de datos *MIAS*. Moustafa, H. et al. [36] testeó distintas arquitecturas de red como *InceptionV3, DenseNet121, ResNet50, VGG16 y MobilenetV2* para clasificar las imágenes de las bases de datos *MIAS*, DDSM y CBIS-DDSM en malignas y benignas. Adicionalmente, el autor implementó un algoritmo de segmentación basado en una arquitectura *U-Net* para extraer las zonas de interés de las imágenes originales y utilizarlas para entrenar las arquitecturas de red mencionadas anteriormente. Por otra parte, para poder reducir la posible sobreespecialización de los modelos ante la escasa volumetría de datos existente, se utilizaron técnicas de *Data Augmentation* combinadas con técnicas de *transfer learning*. Finalmente, la arquitectura *InceptionV3* junto con la arquitectura *U-Net* lograron una exactitud del 98.87 % sobre el set de datos *DDSM*.

Enas, M.F. et al. [7] entrenó la misma arquitectura de red sobre dos conjuntos de datos distintos. El primer conjunto estaba compuesto por mamografías completas, mientras que el segundo, estaba formado exclusivamente por parches extraídos de las zonas de interés de la imagen. El modelo construido a partir de las zonas de interés presentó mejores resultados a la hora de clasificar las observaciones de los sets de datos *MIAS* y *Inbreast*, logrando una exactitud del 92.6 % y 96.49 % respectivamente. Por último, Chougrad H. et al. [37] realizó un análisis de cómo el uso de técnicas de *transfer learning* puede ser beneficioso a la hora de clasificar patologías. En su trabajo, la actuación de los modelos generados mediante transferencia de aprendizaje superaba en un 10 % la actuación de los modelos generados desde cero.

3. Materiales y Recursos

3.1 Bases de datos

Para el desarrollo de este proyecto y del sistema de ayuda automático propuesto, se han utilizado tres bases de datos distintas con imágenes mamográficas. Los sets de datos escogidos han sido aquellos que más presencia han tenido durante la revisión del estado del arte, así como para la construcción de herramientas similares a la propuesta.

En primera instancia, se encuentra la base de datos *Mammographic Image Analysis Society* (*MIAS*) [38] formada por un total de 322 imágenes en formato *"Portable Gray Map"* (PGM). Las mamografías presentes en este set tienen una resolución de 1024x1024 píxeles y contienen anotaciones realizadas por radiólogos. Entre estas, destaca información como la densidad del seno (graso, denso y semidenso) o el tipo de lesión presente (calcificaciones, masas o asimetrías, entre otras). Adicionalmente, se detalla la localización de las zonas de interés, indicando las coordenadas x e y (en píxeles) del centro de cada patología y el radio del círculo que la contiene. Finalmente, *MIAS* está formado por un total de 207 observaciones normales, 63 benignas y 52 malignas.

En segunda instancia, el conjunto de datos utilizado *Curated Breast Imaging Subset of Digital Database for Screening Mammography* (*CBIS-DDSM*) [39] es una versión del set de datos *Digital Database for Screening Mammography* (*DDSM*) estandarizada y actualizada, en la cual, cada mamografía se ha descomprimido y convertido a formato *Digital Imaging and Communications in Medicine* (DICOM). El set de datos está formado por un total de 3103 mamografías divididas en calcificaciones (1511 observaciones) y masas (1592 observaciones) que presentan tanto patologías benignas (1429 observaciones) como malignas (1457 observaciones). Cabe destacar que, en una misma observación, pueden coexistir varias patologías distintas. Adicionalmente, existen 682 observaciones en las que, analizando exclusivamente el examen mamográfico, no se puede asegurar la existencia de algún tipo de cáncer. Aun así, dado que los radiólogos han considerado la presencia de alguna característica de interés en ellas, han sido incluidas en la base de datos [40].

Por otra parte, *CBIS-DDSM,* también contiene información de las zonas de interés en las que se encuentra cada patología. De este modo, en la base de datos hay un total de 3568 zonas de interés representadas mediante máscaras (imágenes binarias dónde cada lesión se describe con el uso de píxeles blancos y, el resto de zonas, con píxeles negros, Figura 6). Finalmente, las imágenes de este set presentan resoluciones variables entre 500 y 5000 píxeles de anchura y 2300 y 6300 píxeles de altura.

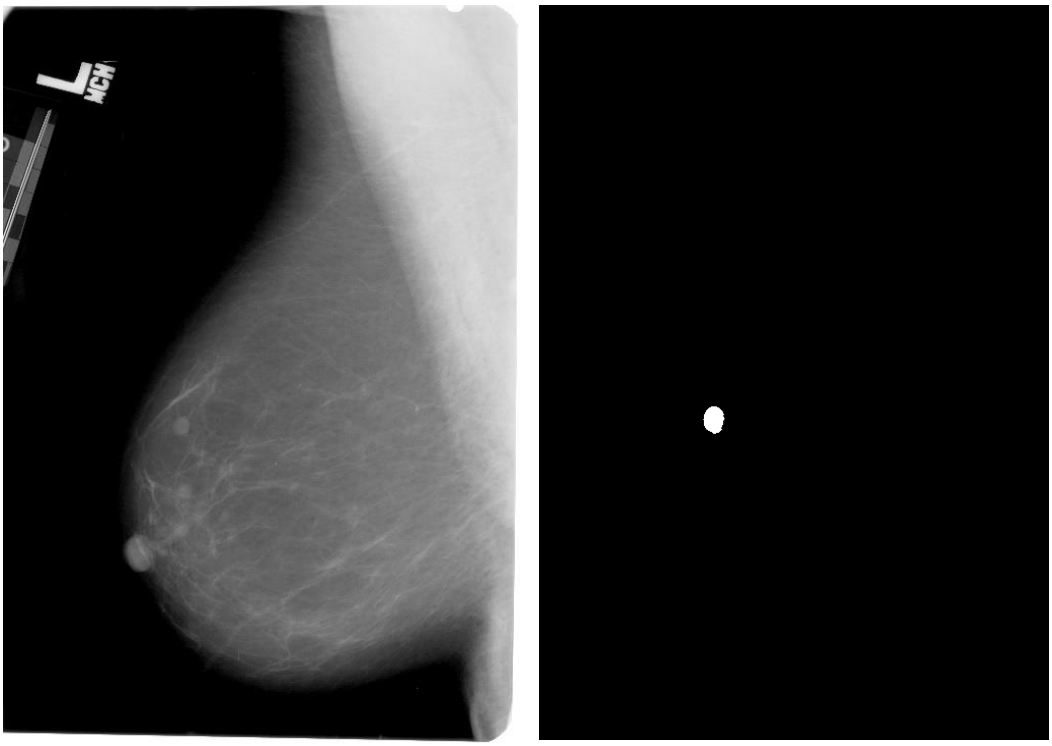


***Figura 6**. Ejemplo de la imagen Mass-Test_P_00099_LEFT_MLO con su correspondiente máscara indicando la zona de la patología.*

En última instancia, la base de datos *Inbreast* [41] contiene un total de 410 mamografías en formato *DICOM* procedentes de 115 pacientes. En este caso, cada patología está representada mediante el estandarte de anotación *Breast Imaging-Reporting and Data System* (BIRADS). A continuación, se muestra la distribución de las imágenes según su patología, así como la descripción correspondiente a cada anotación BIRADS (Tabla 11).

BIRADS	Descripción	N.º de muestras
0	Estado insuficiente	0
1	Seno normal	67
2	Patología benigna	220
3	Sugestivo de benignidad <2 %.	23
4ª	Baja o moderada sospecha 2-10 %.	13
4b	Moderada sospecha 11-40 %.	8
4c	Moderada - alta sospecha 41-94 %.	22
5	Alta sospecha de malignidad >95 %.	49
6	Malignidad confirmada	8

***Tabla 11.** Conjunto de datos Inbreast. Distribución de las muestras en función del código BIRADS*

Por otra parte, el set de datos incluye distintos tipos de lesiones como masas, calcificaciones, asimetrías o distorsiones, localizadas dentro del seno mediante el uso de anotaciones. Estas anotaciones se informan mediante un conjunto de archivos *XML*, indicando el contorno de cada lesión a nivel de píxel. Existen un total de 7383 anotaciones para el set de datos *Inbreast*.

Finalmente, las imágenes de este set presentan resoluciones variables entre los 400 y 2600 píxeles de anchura y entre los 400 y 3800 píxeles de altura.

3.2 Redes Neuronales Convolucionales

Para clasificar correctamente el tipo de cáncer presente en los exámenes mamográficos, se han utilizado un conjunto de modelos de *Deep Learning* que componen el estado del arte en tareas de clasificación de imágenes médicas.

Cada arquitectura de red será representada utilizando la siguiente notación en función del tipo de capa presente:

- Capas convolucionales: *tamaño del filtro* (*anchura x altura x núm. de filtros) / stride (en caso de existir), padding (en caso de existir).* Por ejemplo, la notación "*3x3x32/2, padded*" corresponde a una capa convolucional con 32 filtros de tamaño 3x3, *stride* de 2 y con *padding*.

- Capas de *pooling*: *Función de pool, tamaño del filtro / stride (en caso de existir).* Por ejemplo, la notación "*avg-pooling, 3x3/1*" corresponde a una capa con función de *pool* promedio, filtro de tamaño 3x3 y *stride* de 1.

Finalmente, las capas densas situadas a la salida de cada arquitectura han sido suprimidas dado que no se han utilizado durante la realización de este proyecto.

3.2.1 VGG16

En el año 2014, Simonyan y Zisserman diseñaron una red neuronal convolucional que destaca por su simplicidad [42]. La arquitectura propuesta está formada por un total de 13 capas convolucionales con función de activación *ReLu* y caracterizadas por tener filtros de tamaño 3x3 y *stride* de 1.

Adicionalmente, el número de filtros utilizados en cada convolución se duplica, a medida que la red se hace más profunda, siendo 64, 128, 256 y 512 el número total de filtros utilizados en cada bloque de la red (véase Figura 7).

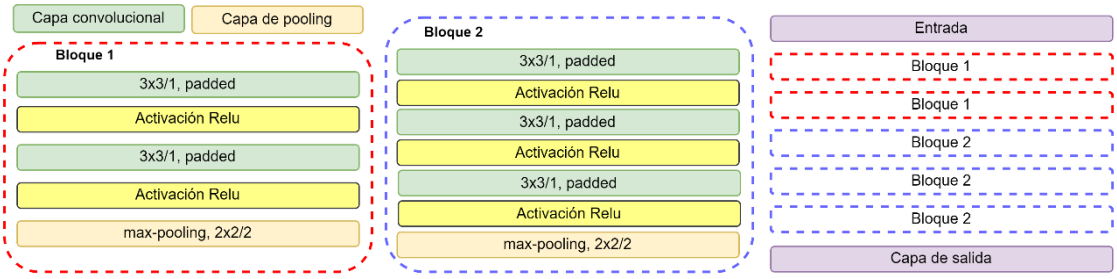


***Figura 7**. A la izquierda se muestran las dos posibles estructuras de bloque convolucional presentes en la arquitectura VGG16. A la derecha, la estructura global de la red.*

Por otra parte, la red presenta un conjunto de capas de *pooling* formadas por filtros de tamaño 2x2 y función de agrupamiento *max* para filtrar aquellas características irrelevantes presentes en cada entrada. La unión de capas convolucionales seguidas de una capa de *pooling* recibe el nombre de bloque. Hay un total de 5 bloques en la arquitectura *VGG16*.

Aunque la estructura de red *VGG16* destaca por su simplicidad, el número de parámetros a ajustar durante el proceso de entrenamiento supone una gran

desventaja al requerir de grandes cantidades de memoria y producir altos costes computacionales en el sistema.

3.2.2 ResNet50

Esta arquitectura fue propuesta en el año 2016 por un grupo de investigadores de Microsoft que perseguían la idea de que cualquier red profunda debería de tener, como mínimo, un comportamiento igual o superior que el de una red menos profunda [43]. Para ello, el modelo debería de ser capaz de aprender la función de identidad, es decir, la salida de una capa debería de ser igual que su entrada. Esto, sin embargo, no es posible, debido a diversos problemas como la explosión o la desaparición del gradiente [44], [45].

Para hacer frente a estas adversidades, los autores diseñaron la red de tal forma que pudiera ser capaz de aprender la función de identidad a partir de la diferencia o el residuo existente entre la entrada y la salida de algunas capas o subredes. Para ello, se realizaban interconexiones directas que unían cada entrada con su correspondiente salida tal y como se muestra en la Figura 8.

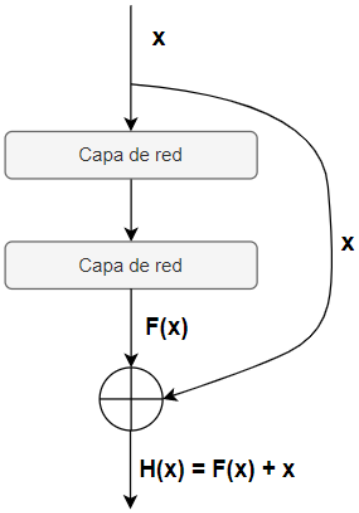


Figura 8. *Ejemplo de interconexión entre capas para reproducir la función identidad*

La función de salida, representada por *H(x),* depende, por una parte, de la salida de las capas de la red, representadas por F(x) y, por otra, por su entrada, representada por x. Así pues, la red sería capaz de omitir algunas subredes haciendo que F(x) sea 0. Adicionalmente, durante el paso de *backpropagation,* la red es capaz de enviar el gradiente directamente hacia la entrada, ignorando algunas capas convolucionales en caso de que estas no aportasen ningún tipo de información [46].

Por otra parte, la arquitectura *ResNet50* está formada por una capa convolucional con filtros de 7x7 y *stride* de 2, seguida de una capa de *max-pooling* con filtros de 3x3 y *stride* de 2. A continuación, le siguen un conjunto de bloques residuales formados por 3 capas convolucionales con filtros de tamaño 1x1, 3x3 y 1x1, respectivamente. Cada uno de estos bloques residuales se agrupa en 4 etapas distintas tal y como se muestra en la Figura 9. A la salida de la última etapa, una capa de *average-pooling* precede la salida del modelo.

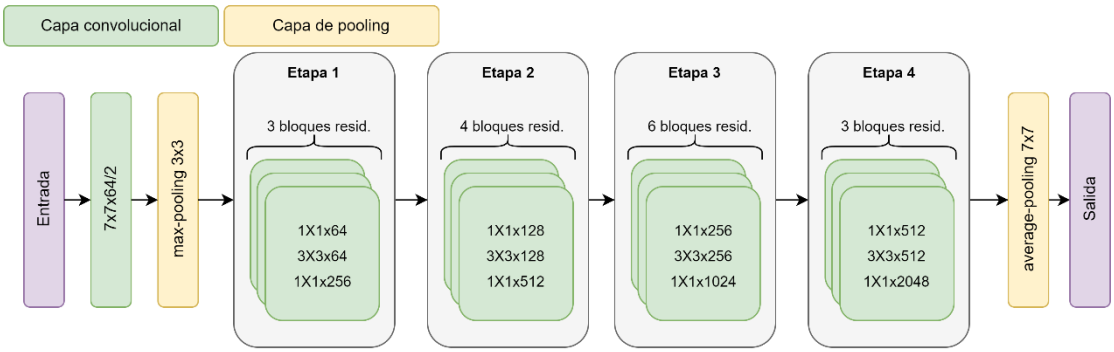


Figura 9. *Arquitectura ResNet50 formada por 50 capas.*

3.2.3 InceptionV3

En el año 2015, un grupo de investigadores de Google persiguió el objetivo de generar redes cuya computación fuera eficiente mediante el uso de convoluciones correctamente factorizadas y técnicas de regularización [47].

Factorizar correctamente una convolución significa utilizar un número menor de conexiones o parámetros, manteniendo los estándares de eficiencia de una red. Por ejemplo, una capa convolucional con filtros de tamaño 5x5 (5x5 = 25 parámetros) podría sustituirse por dos capas convolucionales de tamaño 3x3 (3x3 + 3x3 = 18 parámetros) decrementando el número de parámetros en un 28 %. Esta misma aproximación podría realizarse aplicando dos filtros consecutivos de tamaños 1x3 y 3x1 (6 parámetros) con el objetivo de reemplazar un filtro de tamaño 3x3 (9 parámetros), ahorrando un 33 % el número de parámetros necesarios.

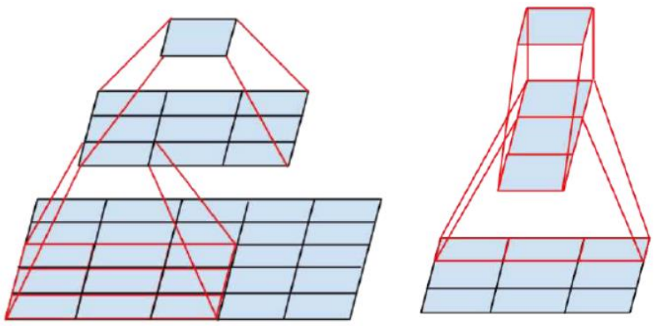


Figura 10*. A la izquierda, una convolución de tamaño 5x5 es realizada a partir de dos filtros de tamaño 3.3. A la derecha, una convolución de tamaño 3x3 es reemplazada por dos convoluciones de tamaño 1x3 y 3x1.*

Este tipo de configuraciones forman parte de los conocidos "*módulos Inception.*" Un *módulo Inception* consiste, principalmente, en un conjunto de capas convolucionales situadas de forma paralela y con distintos tamaños de filtro. Los mapas de características generados por cada capa convolucional se concatenarán produciendo una única salida que alimentará el siguiente módulo de la red.

El hecho de combinar distintos tamaños de convolución dentro de un mismo módulo permite que la arquitectura sea capaz de obtener tanto características de alto nivel mediante convoluciones más grandes, como características locales

a través de las convoluciones más pequeñas. Además, el uso de filtros de tamaño 1x1 al inicio de cada módulo permite reducir la dimensionalidad de los mapas de características decrementando en consecuencia, el coste computacional y temporal a la hora de entrenar la red [48].

Existen muchas variaciones posibles para el diseño de una arquitectura de red que contenga módulos *Inception.* En concreto, el modelo *InceptionV3* fue diseñado en el año 2016, introduciendo como novedad el uso de capas convolucionales con filtros de tamaño 3x3 en sustitución de los filtros de tamaño 5x5 (módulo *Inception* A, Figura 13) y capas convolucionales con filtros de tamaño 1x7 y 7x1 (módulo *Inception* B, Figura 14) en sustitución de filtros de tamaño 7x7.

Otra variación introducida en este modelo fueron los "*bloques de reducción" (*Figura 12*).* Estos pretenden sustituir la funcionalidad de las capas de *pooling* a la hora de reducir la dimensionalidad de los mapas de características, con el objetivo de evitar posibles cuellos de botella representacionales a la salida de los módulos *Inception*.

A continuación (Figura 11), se muestra la arquitectura de la red *InceptionV3,* así como los módulos que forman parte de ella.

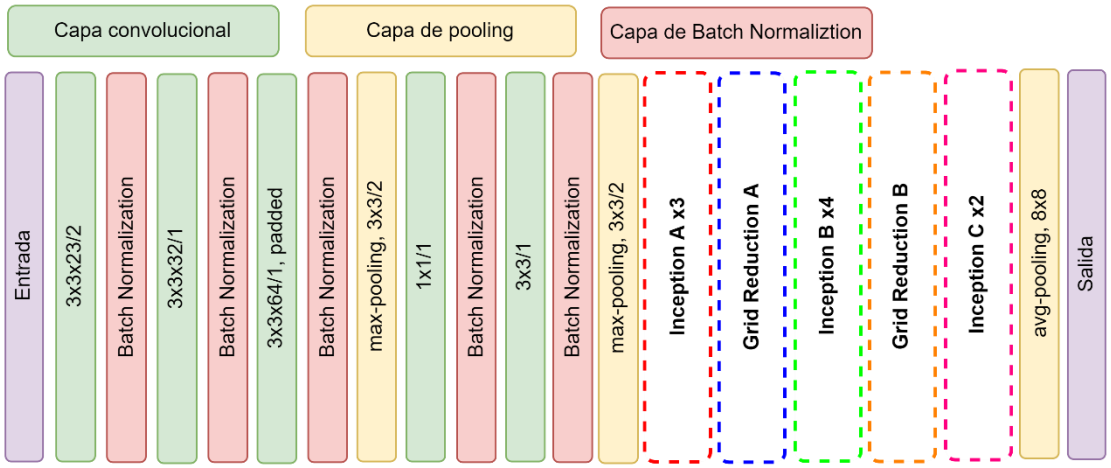


Figura 11. *Arquitectura Inception V3*

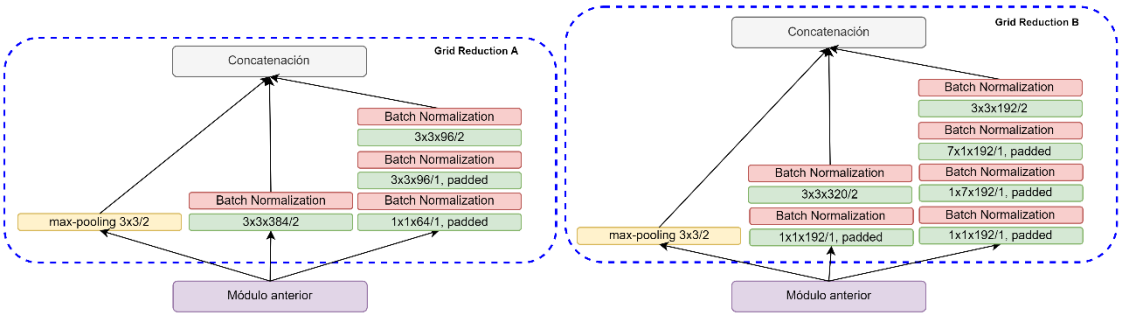


Figura 12. *Bloques de reducción espacial. Estos bloques pretenden reemplazar el comportamiento de una capa de pooling de forma que se evitan cuellos de botella representacionales.*

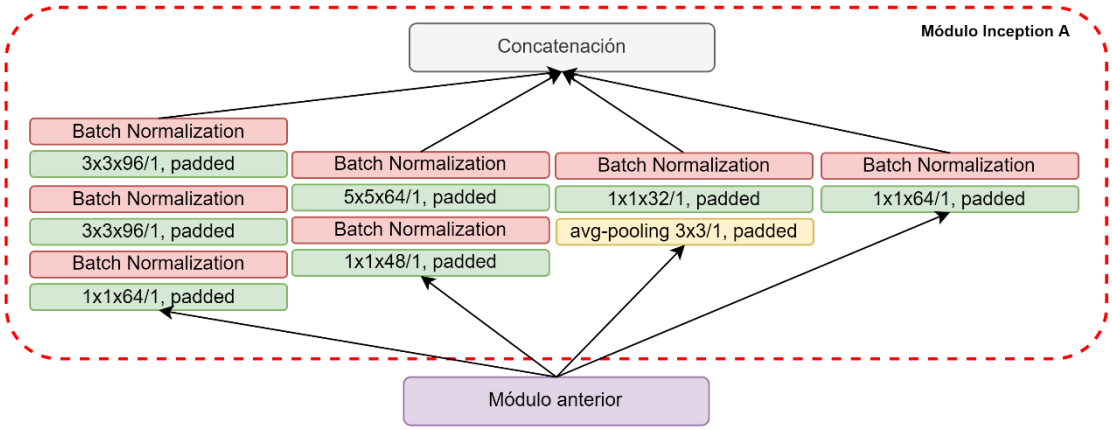


Figura 13. *Módulo Inception A. Los filtros situados a la izquierda sustituyen una convolución de tamaño 5x5.*

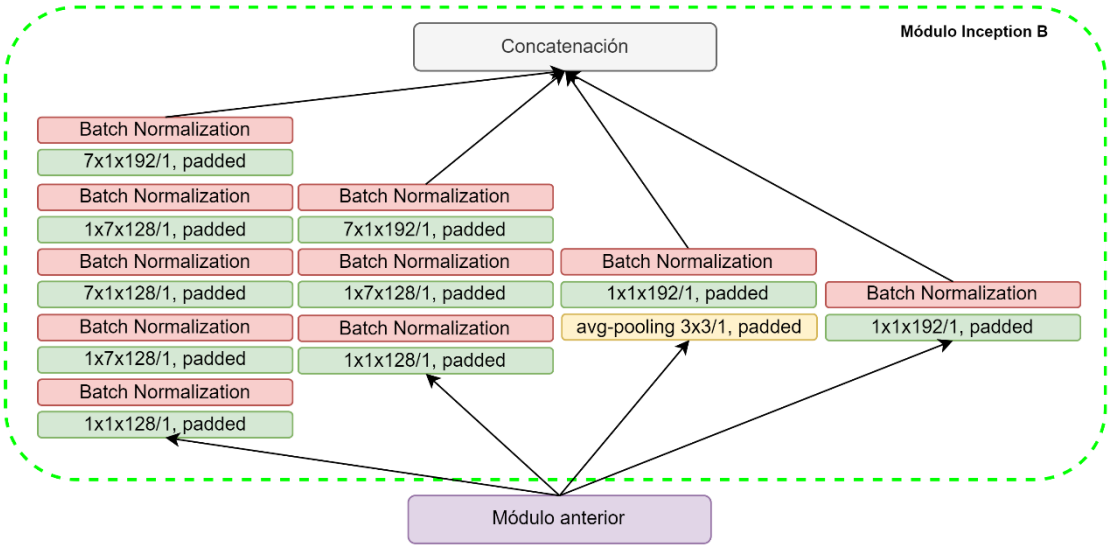


Figura 14. *Módulo Inception B. Los filtros de tamaño 7x1 y 1x7 sustituyen un filtro de tamaño 7x7.*

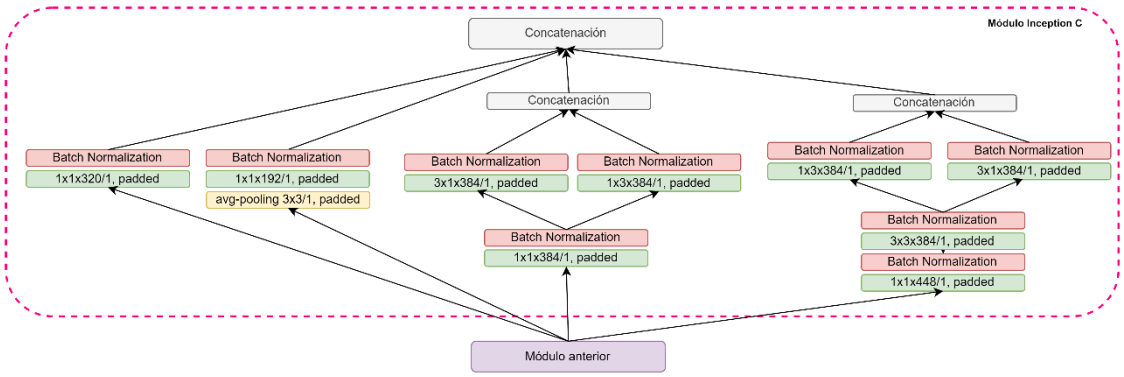


Figura 15. *Módulo Inception C. Los filtros de tamaños 1x3, 3x1 y 3x3 pretenden capturar información global de las entradas introduciendo características de alta dimensionalidad (filtros receptivos más grandes) a la red.*

3.2.4 DenseNet121

Gao Huang propuso, en el año 2018, un modelo de red en el que cada capa convolucional estaba conectada con el resto de capas convolucionales predecesoras y sucesoras.

Esta idea surgió de la base de que "*cualquier red convolucional puede ser más profunda, precisa y eficiente de entrenar si se construyen conexiones que unan las capas próximas a la entrada del modelo con aquellas capas cercanas a su salida."* [49]. De este modo, se diseñaron los conocidos "*bloques densos"* en los que cada capa convolucional utiliza, gracias al uso de conexiones directas, los mapas de características procedentes de todas las capas convolucionales predecesoras. Esta configuración, además, permite reducir el problema del desvanecimiento del gradiente y fortalece la propagación de características.

A continuación, en la Figura 16, se muestra la estructura de un bloque denso compuesto por 3 bloques de convolución. Tal y como se puede observar, cada bloque convolucional (formado por dos capas convolucionales y dos capas de *batch normalization*, Figura 17) está conectado con todos los bloques convolucionales posteriores.

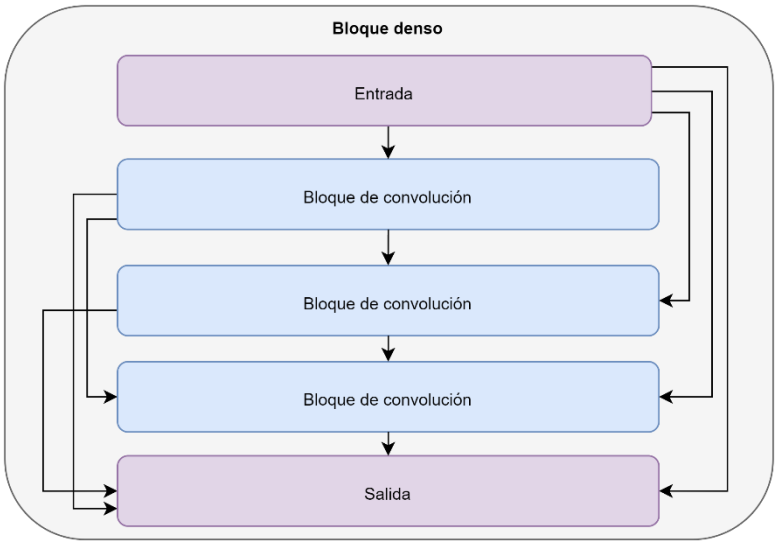


Figura 16. *Ejemplo de bloque denso compuesto por 3 bloques convolucionales*

Existen distintas arquitecturas de red que utilizan bloques densos en su configuración. Entre ellas, se encuentran *DenseNet121*, *DenseNet169*, *DenseNet201* o *DenseNet264*. Acorde con la volumetría de datos disponibles para la realización de este proyecto, se ha escogido utilizar la primera arquitectura de red dado que es la menos profunda. En la Figura 17, se muestra la estructura de *DenseNet121*.

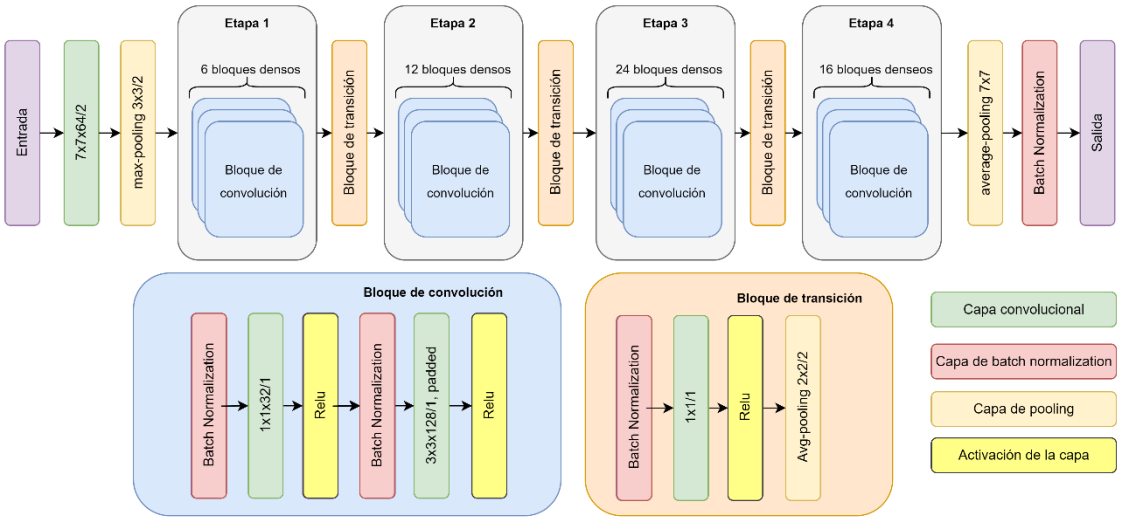


Figura 17. *Arquitectura DenseNet121. En la parte inferior se muestra la estructura que presenta un bloque de convolución y un bloque de transición. No se muestran el número de filtros en las capas convolucionales del bloque de transición, debido a que este es variable a medida que se avanza por la red. De izquierda a derecha, el número total de filtros es de 128, 256 y 512.*

3.3 Combinación de clasificadores

La intervención de varios especialistas en el análisis de exámenes mamográficos, puede incrementar la tasa de acierto entre un 8 % y un 16 % a la hora de diagnosticar lesiones de seno como cancerígenas [50]–[52]. Adicionalmente, la tasa de falsos positivos puede verse reducida entre un 10 % y un 27 % cuando la decisión de clasificar una patología como cancerígena se toma combinando el diagnóstico de múltiples profesionales, en vez de realizar pronósticos consecutivos espaciados en el tiempo.

Persiguiendo esta idea, dentro del ámbito de *data mining* existen distintas aproximaciones que permiten generar modelos más complejos a partir de la combinación de modelos simples. En este sentido, se pretende mitigar el error causado por cada modelo base, utilizando las decisiones tomadas por el resto. Existen dos opciones posibles para crear modelos combinados.

Por una parte, se encuentra la "*Combinación paralela de clasificadores base similares".* Esta consiste en generar clasificadores complejos paralelizando un conjunto de modelos simples y similares como, por ejemplo, árboles de decisión. Para generar cada uno de los clasificadores base, se entrena cada algoritmo utilizando muestras de datos distintas procedentes del conjunto de entrenamiento. La salida de cada algoritmo será utilizada de forma parcial para tomar la decisión final [53].

Por otra parte, se encuentra la "*Combinación secuencial de clasificadores*". Esta consiste en construir un clasificador complejo a partir de clasificadores simples y diferentes como, por ejemplo, una red neuronal y una máquina de soporte vectorial. Existen dos métodos distintos de combinar secuencialmente clasificadores simples: el método de *stacking* y el de *cascading* [54]. Ambos se basan en la idea de producir un clasificador final que utilice como datos de entrada las predicciones parciales generadas por el conjunto de clasificadores

base, en lugar de los datos originales. De esta forma, es posible aprovechar el conocimiento específico de cada modelo simple a la hora de generar la clasificación final. Cabe destacar que, en el método de *cascading* es posible utilizar los datos originales de entrada, juntamente con las predicciones realizadas por los clasificadores simples, para obtener la decisión final.

Dado que en este proyecto se han implementado distintas arquitecturas de redes neuronales convolucionales, se combinará el conocimiento recogido por cada una de ellas para poder clasificar las lesiones presentes en los exámenes mamográficos. En este sentido, se pretende emular el comportamiento que tendrían varios radiólogos a la hora de consensuar cual es el pronóstico final de cada lesión.

Para combinar las decisiones individuales de los clasificadores simples, se ha implementado un algoritmo de *stacking* tal y como se muestra en la Figura 18. De esta forma, las predicciones realizadas por cada red servirán de entrada a un clasificador *Random Forest,* que tendrá el objetivo de realizar la clasificación final de cada muestra.

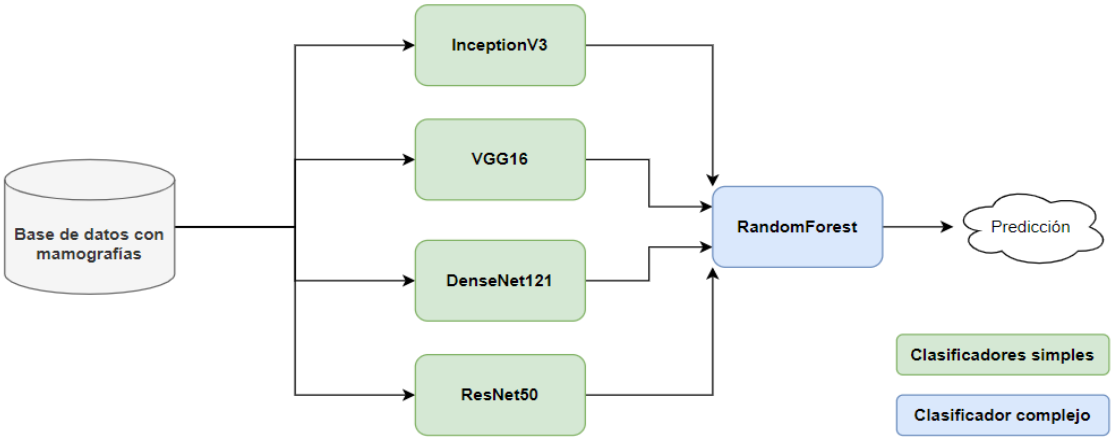


Figura 18. *Combinación secuencial de clasificadores mediante stacking. Las predicciones parciales de las redes neuronales nutren un algoritmo de Random Forest para obtener la predicción final.*

3.3.1 Random Forest

Random Forest es un algoritmo que combina un conjunto de árboles de decisión entrenados de forma aleatoria y paralela. Para generar dicha aleatoriedad, se obtienen diferentes versiones del conjunto de entrenamiento usando distintas combinaciones de variables y escogiendo sus observaciones a partir de un proceso de muestreo con reemplazo. La decisión final se obtiene promediando las predicciones parciales de cada uno de los árboles de decisión que componen el modelo [55].

Dado que cada árbol de decisión es entrenado a partir de una muestra aleatoria de variables, es posible medir el peso relativo que tiene cada una de ellas en la salida del modelo. En este aspecto, el uso de este algoritmo como ensamblador permitirá observar la importancia de cada red neuronal convolucional a la hora de tomar la decisión definitiva.

4. Metodología

4.1 Construcción del set de datos

Para la implementación de este proyecto, se han realizado dos aproximaciones distintas, tal y como se detallará en la sección "*5. Experimentos y resultados".* La primera de ellas se basa en el uso de imágenes completas para la aplicación de algoritmos de *Deep Learning* y la segunda, en el uso exclusivo de las zonas de interés (zonas que contienen la lesión). Así pues, esta sección pretende exponer, de forma clara y sencilla, cómo ha sido la construcción del conjunto de datos utilizado en ambos experimentos.

4.1.1 Creación del conjunto de datos

El conjunto de datos final se ha creado a partir de la unión de todas las bases de datos mencionadas en la sección "*3.1 Bases de datos".* Previamente a la unificación de estas, se han realizado un conjunto de modificaciones y descartes con el objetivo de obtener muestras homogéneas y balanceadas.

En primera instancia, en el set de datos *CBIS-DDSM* se han eliminado todas aquellas observaciones cuya lesión estaba etiquetada como *Benign without callback.* Tal y como se ha comentado en la sección *"*"*3.1 Bases de datos"*, esta clasificación no indica con certeza la existencia de lesiones benignas o malignas, por lo que su introducción en el set de datos final podría generar confusión en el algoritmo.

Por otra parte, un total de 5 observaciones presentes en el set contienen tanto lesiones benignas como malignas. Esta casuística ha sido descartada para la realización del experimento con imágenes completas, dado que se pretende resolver un problema de clasificación binaria en el cual cada observación puede pertenecer exclusivamente a una única clase. No obstante, este descarte no aplica para el experimento a nivel de zonas de interés, ya que se cumple el requisito en el que una observación pertenece exclusivamente a una única clase.

En segunda instancia, para el set de datos *Inbreast,* se han suprimido aquellas observaciones cuyas clasificaciones *BIRADS* son 1 (*Seno Normal*), 3 (*Sugestivo de benignidad < 2 %*) y 4a (*Bajo o moderada sospecha*). De nuevo, este descarte se ha realizado dado que no es posible asegurar con certeza si las lesiones son claramente benignas o malignas. Adicionalmente, 2 imágenes son descartadas del set de datos, debido a que no tienen correctamente localizada la zona de la lesión en los ficheros XML.

En tercera instancia, para el set de datos *MIAS,* se han suprimido aquellas observaciones que no presentaban ningún tipo de patología. Adicionalmente, al igual que en el caso de *CBIS-DDSM,* se ha suprimido 1 observación que contenía tanto patologías benignas como malignas. Finalmente, otra observación cuya lesión no estaba localizada ha sido descartada, tanto para la aproximación con imágenes completas como para la aproximación con imágenes de las zonas de interés.

Por último, la heterogeneidad de resoluciones presente en todas las bases de datos juntamente con la dispersión y el tamaño característico de las calcificaciones han impedido generar recortes suficientemente homogéneos a la hora de unificar los conjuntos de datos. Además, la clasificación de calcificaciones mediante el uso de mamografías resulta una tarea compleja, debido a la poca representación espacial que tienen este tipo de lesiones con respecto a la imagen completa[1]. En este sentido, se han descartado todas aquellas calcificaciones presentes en los conjuntos de datos.

A continuación, se muestran de forma gráfica las exclusiones realizadas en todos los conjuntos de datos contenidos en el proyecto (de la Figura 20 a la Figura 24).

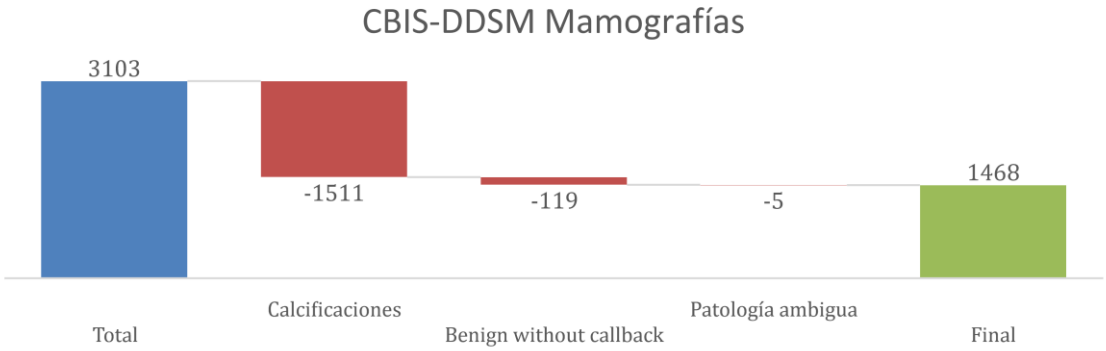


***Figura 20**. Exclusiones conjunto de datos CBIS-DDSM para el experimento con imágenes completas*

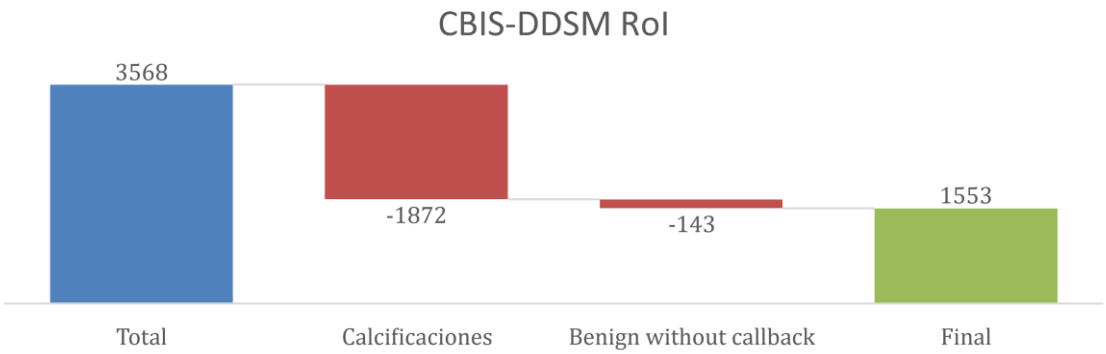


***Figura 19.** Exclusiones conjunto de datos CBIS-DDSM para el experimento con zonas de interés*

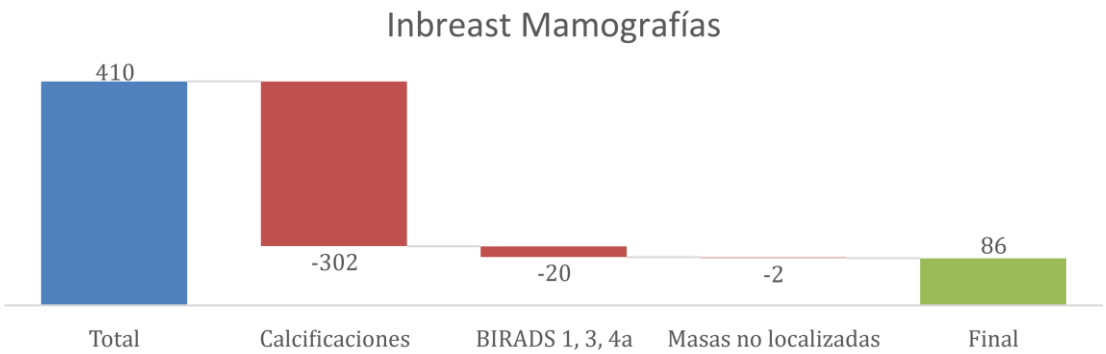


***Figura 21**. Exclusiones conjunto de datos Inbreast para el experimento con imágenes completas.*

[1] Una mamografía digital presenta aproximadamente una resolución de 0.5 milímetros por píxel y el tamaño medio de una calcificación es de 0.5 milímetros ([81], [82]). En este sentido, una calcificación queda representada mediante un único píxel en imágenes con resoluciones de entorno los 3000 – 5000 píxeles.

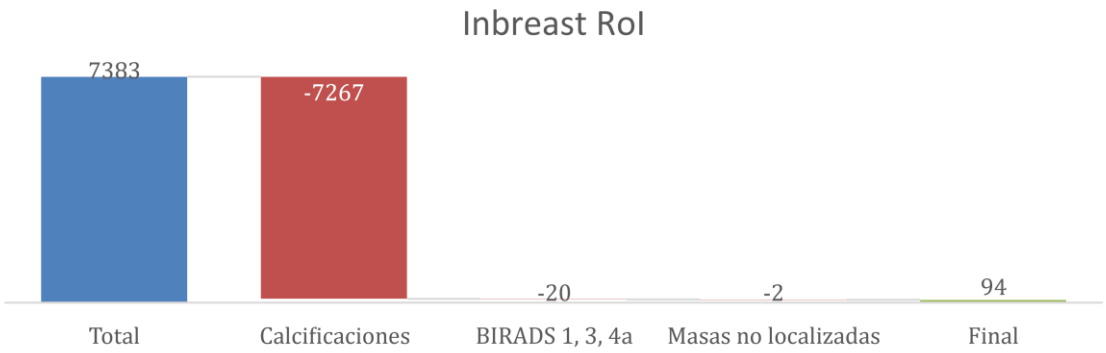


Figura 22. *Exclusiones conjunto de datos Inbreast para el experimento con zonas de interés*

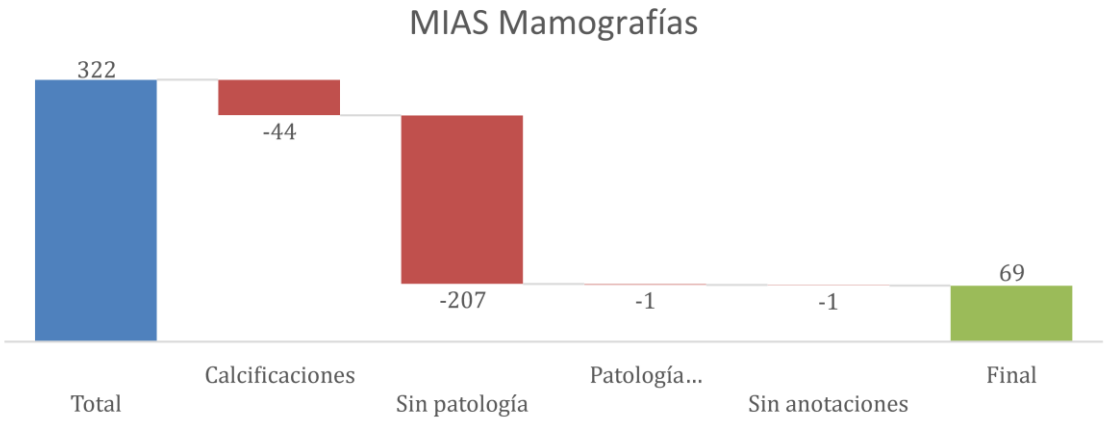


Figura 23*. Exclusiones conjunto de datos MIAS para el experimento con imágenes completas*

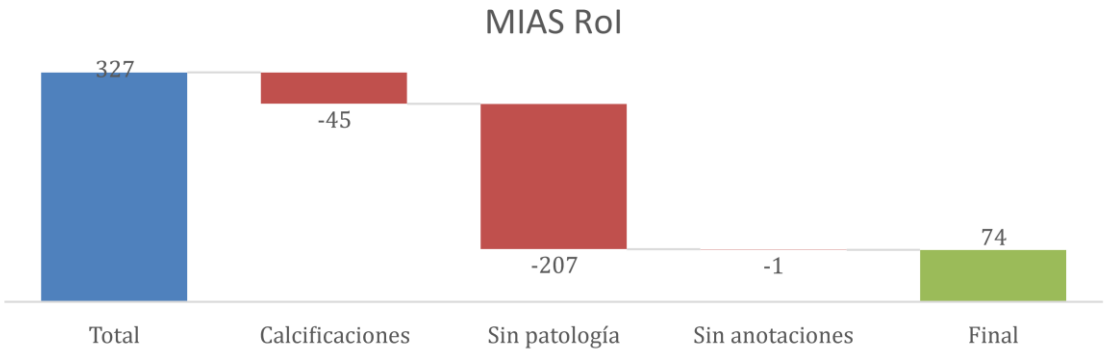


Figura 24. *Exclusiones conjunto de datos MIAS para el experimento con zonas de interés.*

Una vez finalizadas las modificaciones de cada base de datos, se han unido las observaciones resultantes creando un único conjunto formado por un total de 1623 muestras para el experimento con imágenes completas y 1721 muestras para el experimento con recortes de las zonas de interés. Adicionalmente, los conjuntos de datos resultantes muestran una proporción de clases balanceada, tal y como se puede observar en la Figura 25.

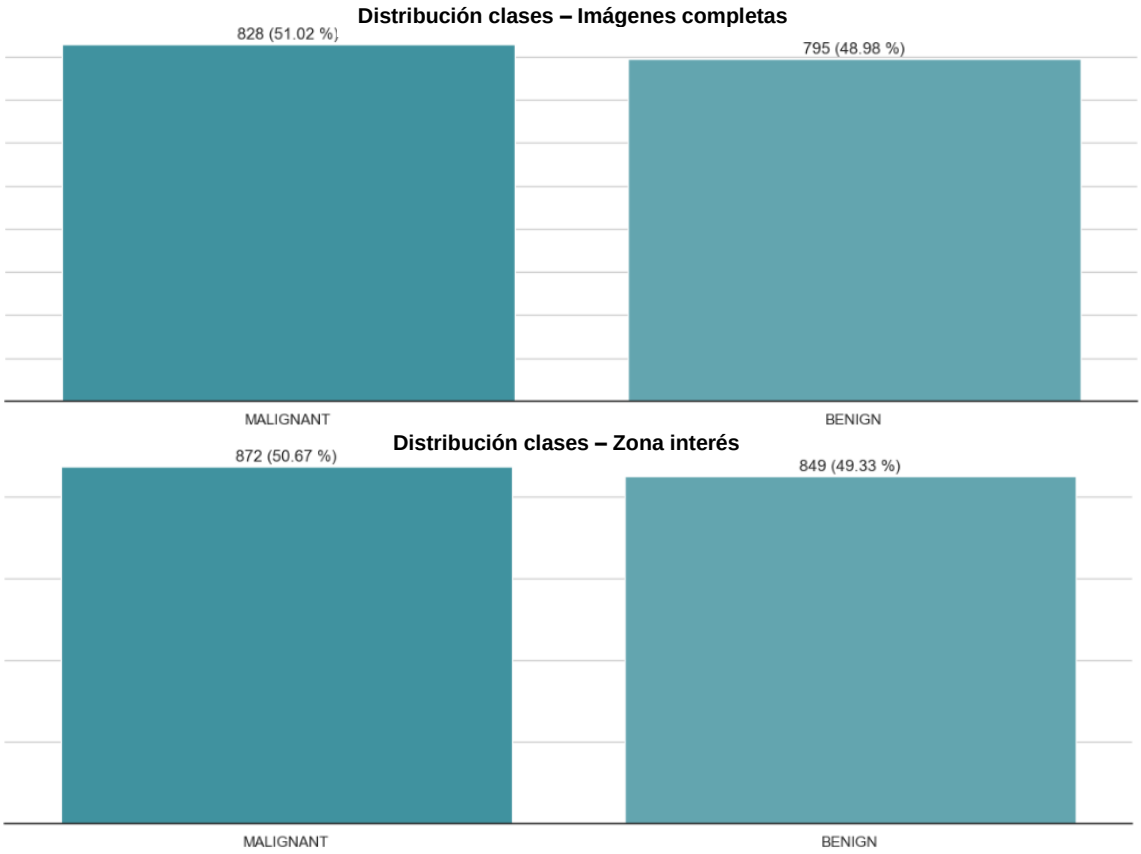


***Figura 25**. Distribución de clases para la realización del experimento con imágenes completas (arriba) y con zonas de interés (abajo).*

Para poder crear una herramienta generalizable que permita clasificar correctamente la clase de cada lesión, se ha dividido el set de datos en tres subconjuntos disjuntos, cada uno con una finalidad específica.

Por una parte, se ha creado un set de datos de entrenamiento que servirá para ajustar los parámetros de los modelos utilizados. Por otra, se ha creado un set de datos de validación cuyo objetivo es seleccionar aquellos parámetros que permitan una mayor generalización del modelo evitando de esta forma, el sobreajuste. Finalmente, el último conjunto de datos generado servirá para evaluar la capacidad clasificadora del modelo final resultante. Este subconjunto recibe el nombre de conjunto de test.

Dado que la herramienta resultante deberá de ser utilizada con bases de datos nunca vistas durante la implementación del algoritmo, se ha decidido utilizar todo el set de datos *MIAS* modificado como conjunto de test. De esta forma, será posible evaluar el comportamiento de la herramienta final, comparándola con las implementaciones presentes en el estado del arte.

La creación de los subconjuntos de entrenamiento y validación se ha hecho partir de las instancias pertenecientes a los sets de datos *Inbreast* y CBIS-DDSM, atribuyendo un 70 % del total de las observaciones al conjunto de entrenamiento y, el 30 % restante, al conjunto de validación.

El número final de muestras pertenecientes a cada set de datos es de 1152, 495 y 74 para los subconjuntos de entrenamiento, validación y test, respectivamente,

para el experimento con recortes de las zonas de interés; y de 1087, 467 y 69 muestras para los subconjuntos de entrenamiento, validación y test, respectivamente, para el experimento con imágenes completas. A continuación, se muestra la distribución de clases para cada subconjunto.

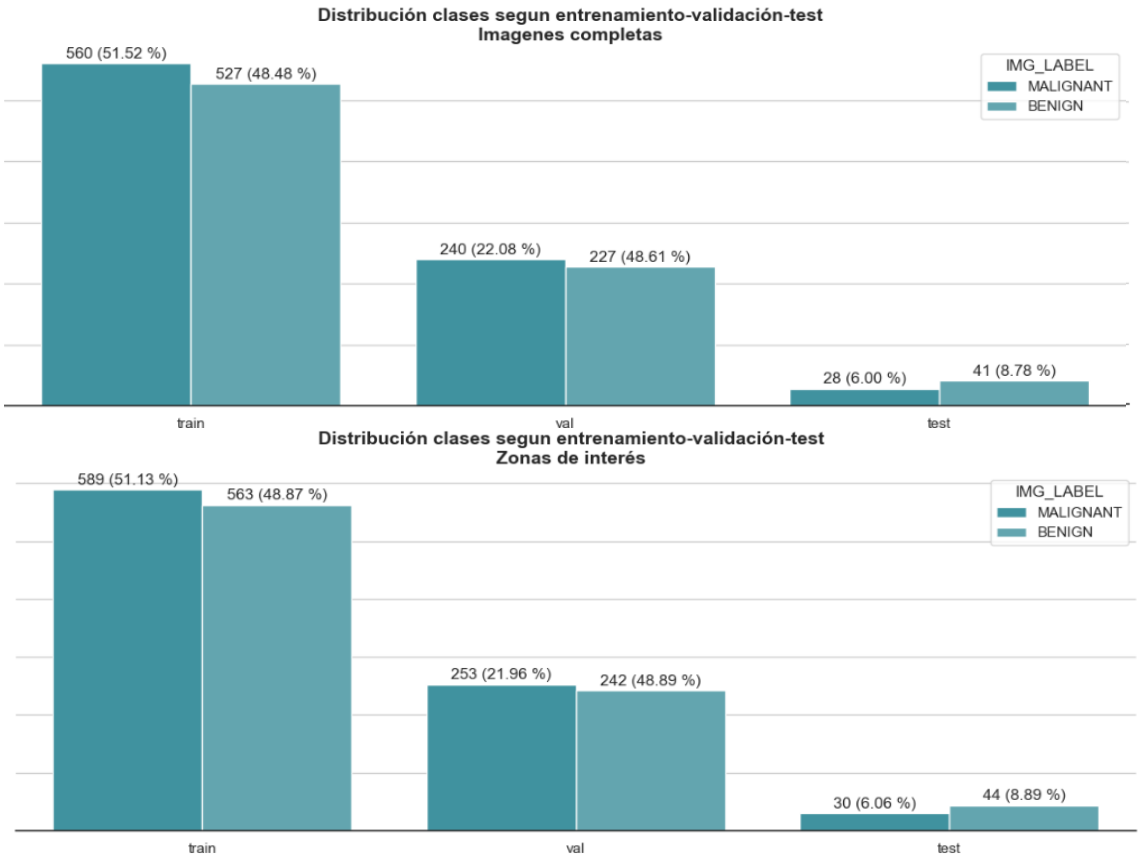


Figura 26. *Distribución de clases para los conjuntos de entrenamiento, validación y test.*

4.1.2 Preprocesado de imágenes

Los niveles de contraste utilizados para representar cada mamografía, así como la presencia de anotaciones y de ruido, impiden detectar con claridad aquellas características que determinan si una lesión es benigna o maligna. Además, el entrenamiento de modelos de *Deep Learning* mediante el uso de imágenes con resoluciones elevadas, conlleva un coste computacional y de memoria muy elevado. Para hacer frente a estas adversidades, se han procesado las imágenes mamográficas en una etapa previa al entrenamiento de las arquitecturas de red.

En primer lugar, se han convertido todas las imágenes a formato *png* y se han normalizado para almacenarlas utilizando 8 bits de memoria. Cabe recordar que, dada la heterogeneidad de las bases de datos utilizadas, cada una presenta un formato (*pgm o dicom)* y un tamaño de imagen (*8-bit, 16-bit*) distinto.

Una vez estandarizado el formato, se han recortado los márgenes de cada mamografía con el objetivo de suprimir la presencia de bordes blancos que podrían crear mapas de características aleatorios en los filtros de las primeras capas de cada modelo. A continuación, se ha suavizado cada imagen eliminando el ruido granular mediante el uso de un filtro medio de tamaño 3x3.

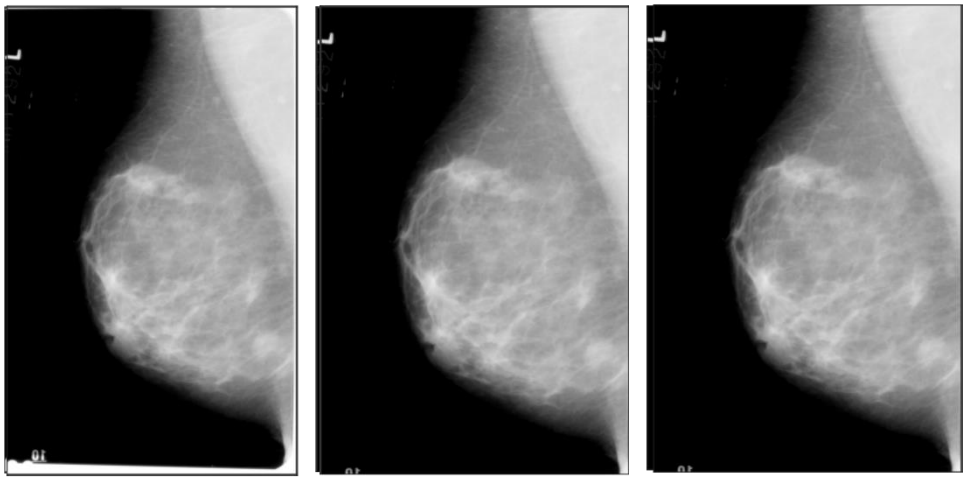


Figura 27. *Eliminación del ruido y de los bordes de una imagen. (a) Imagen original Mass-Test_ - P_00405_LEFT_MLO. (b) Imagen con los márgenes recortados. (c) Imagen sin ruido granular.*

El siguiente paso del procesado ha consistido en eliminar cualquier anotación realizada por un radiólogo. Para ello, en primera instancia, se ha binarizado cada imagen a partir de un valor de *threshold* (umbral) constante de 30, de esta forma, se asigna el valor 1 a cualquier píxel superior o igual al *threshold* y 0, a cualquier valor inferior. Posteriormente, la máscara resultante ha sido modificada mediante filtros morfológicos de apertura[2] y de dilatación[3] para suavizar los bordes. Finalmente, dado que las anotaciones presentes en las mamografías contienen áreas inferiores a la del seno, se ha filtrado cada máscara obteniendo exclusivamente aquella zona de mayor tamaño. El resultado final es utilizado para eliminar los artefactos contenidos en las imágenes originales y obtener exclusivamente la zona del seno.

Figura 28. *Eliminación de anotaciones. (a) imagen original sin ruido y recortada. (b) máscara binaria con valor de threshold constante de 30 (c) máscara modificada con filtros morfológicos (d) selección del área de mayor tamaño (e) imagen sin anotaciones.*

Para mejorar la calidad y el contraste de las imágenes favoreciendo el proceso de detección y clasificación de anormalidades, se ha aplicado una normalización *min-max* y se ha utilizado una técnica de ecualización del histograma adaptativo limitada por contraste (*CLAHE).* Este método de ecualización genera, para cada región de la imagen, un conjunto de histogramas que serán utilizados para redistribuir la luminosidad a lo largo de todos los píxeles de la imagen.

[2] Transformación morfológica compuesta por una erosión y una dilatación.
[3] Transformación morfológica utilizada para incrementar el tamaño de una región.

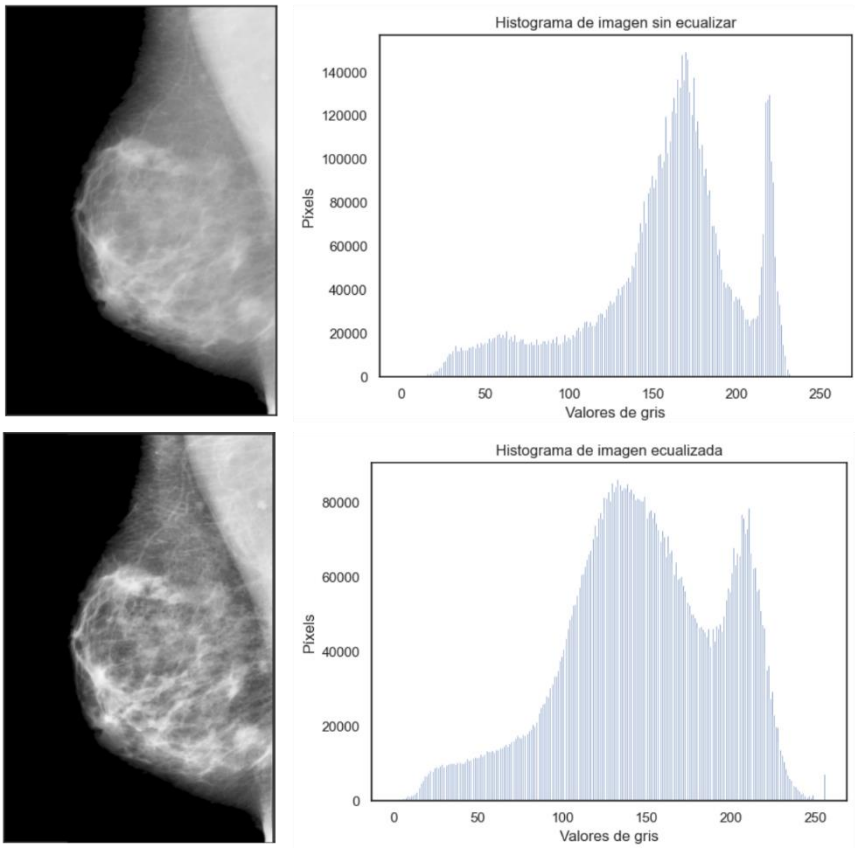


Figura 29. *Transformación CLAHE y normalización. Arriba, la imagen original sin ecualizar ni normalizar juntamente con su histograma. Abajo, la imagen ecualizada y normalizada juntamente con su histograma. Se han suprimido los píxeles con valor 0 en la construcción de cada histograma para representar mejor el efecto producido por la ecualización.*

Una vez alcanzado este punto, se han utilizado las anotaciones presentes en cada set de datos, para extraer las zonas de interés de cada mamografía. Para ello, se han creado máscaras binarias en las que se han asignado valores de píxel de 1 a aquellas zonas marcadas por cada radiólogo y, valores de píxel de 0 al resto. A continuación, se ha creado un rectángulo concéntrico que envuelve cada zona de interés, dejando un margen del 20 % con respecto al borde de cada lesión. Este margen, pretende capturar cualquier tipo de información presente alrededor de cada patología que pueda ser de utilidad a la hora de determinar si una lesión es benigna o maligna ([34], [35]).

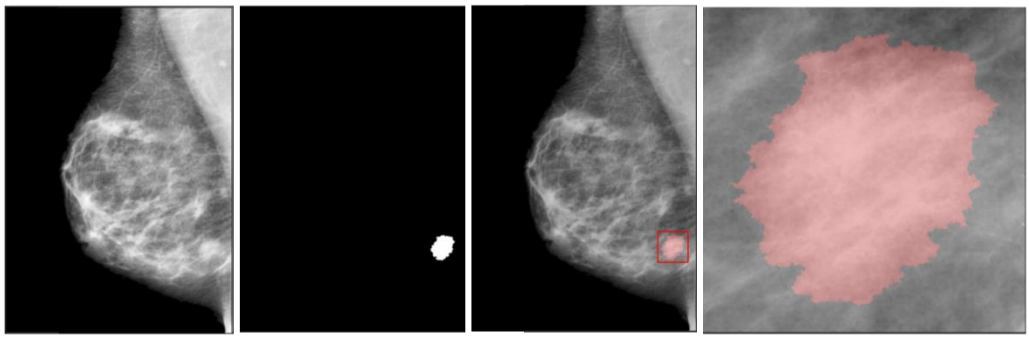


Figura 30. *Recorte de las zonas de interés. (a) Imagen original (b) Máscara generada a partir de las anotaciones. (c) Generación del rectángulo concéntrico a la zona de interés. El área del rectángulo contiene un margen del 20% respecto a la zona de lesión. (d) Zona de interés recortada*

A continuación, con el objetivo de reducir el coste computacional y la capacidad de memoria necesarias para entrenar cada algoritmo, se han escalado todas imágenes a una resolución de 224x224 píxeles para las redes *ResNet50*, *DenseNet121* y *VGG16;* y 299x299 píxeles para la red *InceptionV3*. La resolución escogida ha sido tomada en base al tamaño de las imágenes originales con las que se entrenó cada modelo en la competición *ILSRVC (ImageNet Large Scale Visual Recognition Competition)* [42], [44], [47], [49]

Para finalizar, se han normalizado los valores de píxel de cada imagen con el objetivo de aumentar la velocidad de convergencia de los modelos y evitar la saturación de neuronas. Este proceso, de nuevo, depende del modelo de clasificación seleccionado, de forma que para *InceptionV3*, los píxeles de cada imagen se han normalizado entre los valores -1 y 1; para las redes *ResNet50* y *VGG16*, se ha aplicado una conversión de canales transformando las imágenes de RGB a BGR y se ha estandarizado cada canal utilizando la media y la desviación estándar obtenida a partir de las imágenes del set de datos *ImageNet.* Finalmente, en el caso de *DenseNet121*, los píxeles de cada imagen se han escalado entre los valores 0 y 1, y cada canal se ha normalizado utilizando, nuevamente, la media y la desviación estándar obtenida de *ImageNet*.

A continuación se muestra el Seudocódigo 1 utilizado para realizar el procesado de imágenes.

ALGORÍTMO: PREPROCESADO DE MAMOGRAFÍAS

Entrada: Imágenes mamográficas de los distintos conjuntos de datos
Para cada mamografía
1. Convertir a formato PNG y 8-bit
2. Recortar los bordes
3. Eliminar ruido granular
4. Eliminar anotaciones
5. Normalización min-max
6. Aplicar ecualización del histograma adaptativo limitada por contraste
7. **Si (**experimento zonas de interés)
   Crear máscara de la imagen
   Generar rectangulo concéntrico
   Realizar recorte
8. **Si** (modelo Inception v3)
   Escalado de resolución 299 x 299
   Else
   Escalado de resolución a 224x224
9. **Caso** InceptionV3
   Normalización píxeles entre valores -1 y 1.
   Caso (ResNet50, VGG16)
   Conversión canales RGB a BGR
   Estandarización de canal con media y desviación estándar de Imagenet
   Caso (DenseNet121)
   Normalización píxeles entre valores 0 y 1
   Estandarización de canal con media y desviación estandar de Imagenet

Salida: Imagen preprocesada

Seudocódigo 1. *Pasos realizados en el procesado de mamografías*

4.1.3 Expansión artificial del conjunto de datos

La escasez de datos disponibles supone una de las limitaciones más importantes a la hora de desarrollar algoritmos de detección y clasificación de imágenes médicas. Por una parte, existen muy pocos sets de datos públicos y, por otra, la volumetría de trabajo necesaria para realizar anotaciones suficientemente detalladas y correctas es muy grande [56].

Adicionalmente, se debe de tener en cuenta que cuanto mayor es el tamaño del modelo de *Deep learning* utilizado, mayor debe de ser el conjunto de datos con el que se entrena. En un modelo de *Deep learning,* las capas iniciales captan aquellas características de más alto nivel como, por ejemplo, bordes o aristas, mientras que las capas finales capturan información más detallada y específica que sirve para discriminar entre las posibles salidas del modelo. En este aspecto, los problemas de clasificación de imágenes requieren de un gran número de parámetros y, por ende, el número de datos con los que ajustar dichos parámetros debe de ser suficientemente grande como para evitar un sobreajuste del modelo a los datos de entrada [57].

Para hacer frente a esta situación, existe un conjunto de técnicas de "*expansión artificial del conjunto de datos",* también conocido como "*data augmentation"* que permite aumentar artificialmente el número de muestras del conjunto de entrenamiento generando nuevas instancias a partir de las ya existentes. Para ello, dado que las redes neuronales convolucionales son invariantes a las traslaciones, al punto de vista, al tamaño y a la iluminación [57], se realiza un conjunto de transformaciones y alteraciones geométricas como, por ejemplo, rotaciones o volteos.

Para que las técnicas de *data augmentation* sean eficientes, es necesario que las muestras sintéticas generadas sean una representación de la realidad. En este aspecto, dado que una lesión puede aparecer con cualquier rotación o tamaño, se han creado muestras sintéticas a partir de volteos, traslaciones y deformaciones, tanto en el eje vertical como en el horizontal; rotaciones fijas de 90º, 180º y 270º; zooms del 10 % y rotaciones libres entre -15º y 15º.

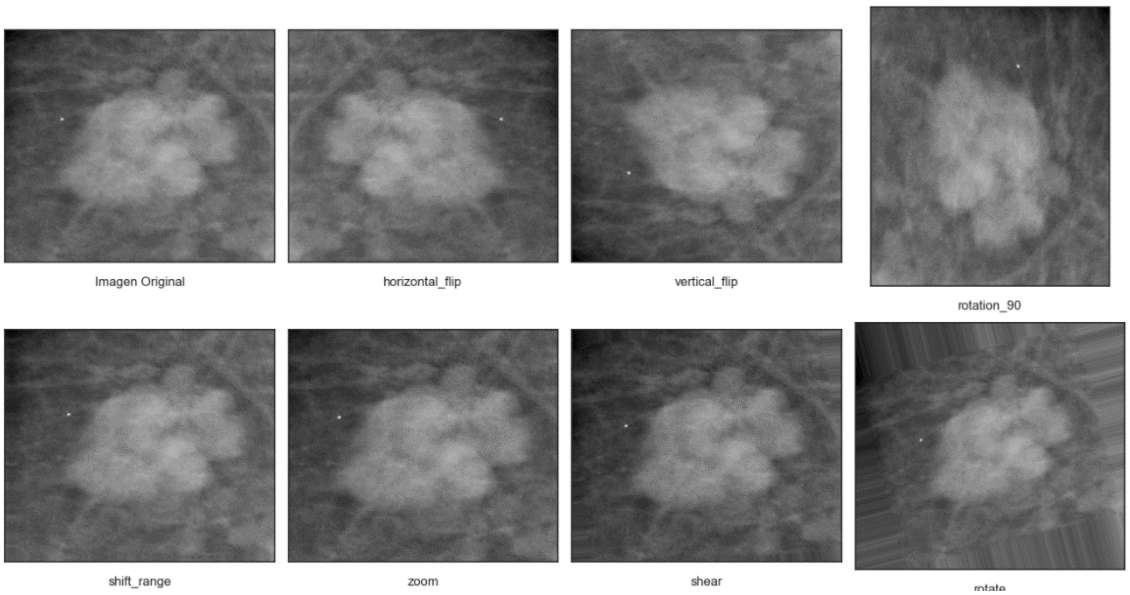


Figura 31. *Técnicas de data augmentation aplicadas.*

Las transformaciones generadas permiten aumentar el conjunto de datos de entrenamiento en un factor de 7, obteniendo un total de 8064 muestras con respecto las 1152 originales, para el experimento con zonas de interés y, 7609 muestras con respecto las 1087 originales, para el experimento con imágenes completas. Destacar que el aumento de datos se aplica exclusivamente al set de datos de entrenamiento excluyendo, pues, los conjuntos de test y validación.

Por último, se ha utilizado una técnica de *online augmentation* mediante el uso de la librería *Albumentations* [58] para generar las transformaciones sintéticas de las imágenes. Esta técnica persigue el objetivo de generar las transformaciones a nivel de *batch* antes de alimentar el modelo [59].

4.2 Modelos de clasificación.

4.2.1 Redes neuronales convolucionales

Los modelos de *Deep Learning* utilizados durante la implementación de este proyecto se han construido aprovechado las arquitecturas de red *VGG16*, *DenseNet121*, *ResNet50* e *InceptionV3,* detalladas en la sección *"3.2 Redes Neuronales Convolucionales".*

Para poder reutilizar dichas arquitecturas, se han sustituido las capas superiores encargadas de clasificar las observaciones del conjunto de datos *Imagenet*, por una arquitectura adaptada al problema de clasificación tratado en este estudio. En consecuencia, se han diseñado dos infraestructuras distintas, tal y como se puede observar en la Figura 32.

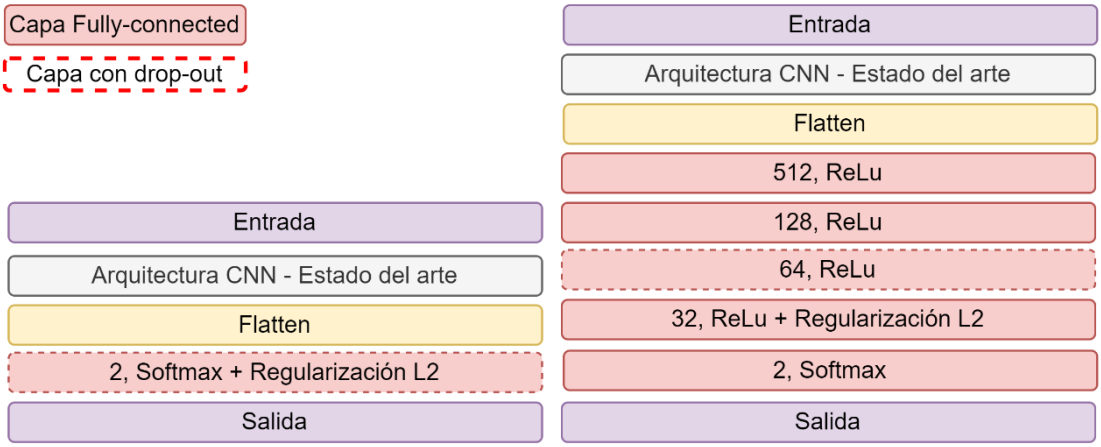


Figura 32. *Arquitecturas de red propuestas para la clasificación de cáncer de seno. La notación para cada capa es: Número de neuronas, activación + regularización (si aplica).*

La primera arquitectura de red, denominada "*arquitectura simple"* de aquí en adelante*,* está diseñada exclusivamente por una única capa *Fully-Connected (FC)* situada a la salida de cada modelo. Esta capa está formada por dos neuronas con función de activación *softmax* permitiendo interpretar las salidas del modelo, como una distribución de probabilidades [60]. En este sentido, saber qué probabilidad existe de que una lesión sea maligna o benigna puede resultar de gran utilidad a la hora de dar soporte a las decisiones tomadas por un especialista.

La segunda arquitectura de red, denominada "*arquitectura compleja"* de aquí en adelante*,* está formada por un total de 5 capas *FC* apiladas a la salida de cada modelo. En este caso, las capas previas a la capa de salida, utilizan la función de activación *Rectified Linear units (ReLu)* con el objetivo de introducir no-linealidad en el sistema incrementando la velocidad de convergencia, sin perjudicar la decisión final del algoritmo [61]. Por último, el número de neuronas utilizadas en cada capa *FC* es de 512, 128, 64, 32 y 2, respectivamente.

Para medir el error producido a la hora de clasificar cada observación se utilizará la función de pérdidas *categorical cross-entropy.* Esta permite cuantificar la diferencia entre las distribuciones de probabilidad generadas por las neuronas con activaciones *softmax* de la capa de salida. A continuación, se define la expresión para esta función de pérdidas (Ec. 1).

$$L = -\frac{1}{N}\sum_{i=1}^{N}[c_i\, ln(y_i) + (1 - c_i)\, ln(1 - y_i)] \qquad \textit{(Ec. 1)}$$

donde N es el número total de muestras de entrenamiento, c_i es la salida verdadera para la instancia i-ésima e y_i es la salida del modelo para la instancia i-ésima.

A partir de la ecuación 1 (Ec. 1) se deduce que las pérdidas generadas por el modelo serán más pequeñas a medida que la función *softmax* asigne una probabilidad mayor a la clase verdadera de cada instancia y una probabilidad menor al resto; por el contrario, el error del modelo incrementaría gradualmente.

Adicionalmente, se han utilizado técnicas de regularización y de *dropout* durante el diseño de ambas infraestructuras de red para asegurar la creación de modelos generalizables.

En este caso, las técnicas de regularización pretenden reducir la influencia de los pesos más grandes de la red durante el cálculo del gradiente descendente en la fase de *backpropagation*. Para ello, se añade un término adicional a la función de coste del modelo, tal y como se muestra en la ecuación 2:

$$C = L + \lambda R(\boldsymbol{W}) \qquad \textit{(Ec. 2)}$$

donde L es la función de coste o de pérdidas original del modelo, λ es un hiperparámetro que mide la importancia relativa de la componente de regularización y R(**W**) es el término de regularización añadido que depende de la matriz de pesos de la red.

Si el valor de la función de coste del modelo original es muy próximo a cero, pero el valor del término de regularización es alto, se estará produciendo un sobreajuste de los parámetros a partir de la actualización de los pesos más grandes del modelo. Por el contrario, si el valor de la función de coste del modelo original es elevado y el valor del término de regularización es bajo, se estará produciendo un infra ajuste de los parámetros del modelo. De esta forma, el término de regularización persigue encontrar aquellos pesos más pequeños, a la vez que se minimiza la función de coste ([62], [63]).

Entre las técnicas de regularización más conocidas, destacan: la técnica de regularización *Lasso;* la técnica de regularización *Ridge,* y la técnica de regularización *ElasticNet.* En este proyecto, se ha utilizado la técnica de *Regularización Ridge* calculada a partir de la norma L2 del vector de parámetros del modelo (ecuación 3), asignando un valor de 0.25 al parámetro λ.

$$R(\boldsymbol{W}) = \sum_{i}\sum_{j} w_{i,j}^{2} \qquad \textit{(Ec. 3)}$$

De esta forma, el coste final de cada modelo se definirá como:

$$C = -\frac{1}{N}\sum_{i=1}^{N}[c_i\, ln(y_i) + (1-c_i)\, ln(1-y_i)] + 0.25\sum_{i}\sum_{j} w_{i,j}^{2} \qquad \textit{(Ec. 4)}$$

donde N es el número total muestras de entrenamiento, c_i es la salida verdadera para la instancia *i-ésima*, y_i es la salida de del modelo para la instancia i-ésima y $w_{i,j}$ es el valor de *i,j-ésimo* de la matriz de pesos de la red.

Finalmente, la técnica de *dropout* [64] consiste en inhabilitar aleatoriamente la participación de un conjunto de neuronas durante la fase de entrenamiento de un modelo. El nivel de aleatoriedad estará determinado por la *tasa de dropout,* la cual establece la probabilidad con la que cada neurona puede quedar *inactiva* durante la fase de entrenamiento [65]. En el diseño propuesto, ambas arquitecturas de red contienen una única capa cuya *tasa de dropout* es del 25 %.

4.2.1.1 Transferencia de aprendizaje y ajuste fino de parámetros.

La profundidad y el número de parámetros de las redes neuronales convolucionales creadas en este proyecto plantean distintas adversidades a hacer frente durante el desarrollo del mismo.

En primer lugar, el tiempo de entrenamiento necesario para minimizar la función de coste de cada modelo puede ser relativamente alto si no se configuran adecuadamente parámetros, como la velocidad de aprendizaje o la inicialización de los pesos. Adicionalmente, la planificación del proyecto estimada, en 300 horas, limita la búsqueda exhaustiva de aquellos hiperparámetros que permitan optimizar el rendimiento del modelo a la hora de clasificar las lesiones de seno.

En segundo lugar, aunque se utilicen técnicas de *data augmentation* combinadas con técnicas de regularización o *dropout,* la escasez de observaciones contenidas en los conjuntos de datos puede dificultar la fase de aprendizaje de cada modelo, generando resultados poco generalizables.

Para hacer frente a las dificultades expuestas anteriormente, se han utilizado técnicas de transferencia de aprendizaje ([37], [66], [67]). Estas se basan en *"transferir el conocimiento"* que tiene un modelo entrenado sobre un gran conjunto de datos a otro dominio distinto. El conocimiento de cada algoritmo está reflejado en los parámetros de las capas que lo componen y es importante remarcar que esta técnica solo es aplicable cuando los parámetros aprendidos

por un modelo durante la primera etapa son suficientemente generales y útiles para aplicarlos en otros entornos similares [68].

En este proyecto, para realizar la clasificación de imágenes médicas, se han utilizado los parámetros aprendidos por las redes al entrenarse sobre el set de datos de *Imagenet.* En este caso, dado que el conocimiento extraído pertenece a un dominio distinto al de la medicina, se han adoptado distintas estrategias de *tunning* para ajustar los parámetros de la red al problema específico planteado.

En primera instancia, se ha definido una estrategia denominada *"0FT"* en la que cada arquitectura de red actuará como extractor de características. En este sentido, se entrenarán exclusivamente los pesos de las capas *FC* pertenecientes a la arquitectura *simple* y *compleja,* situadas a la salida de cada modelo de *Deep Learning*.

A continuación, considerando que las capas convolucionales más próximas a la entrada de un modelo aprenden características de alto nivel, aplicables a casi todo tipo de tareas de clasificación, y que las capas más profundas aprenden características de bajo nivel, específicas de cada ámbito [69], las estrategias "*1FT", "2FT", "3FT"* y "4FT" buscan optimizar el número de parámetros que cada red necesita ajustar para clasificar correctamente imágenes mamográficas. De este modo, en función de cada arquitectura de red, se ha definido un conjunto determinado de capas a entrenar para cada estrategia. Para la arquitectura *VGG16,* el total de parámetros a entrenar estará determinado por la composición de un *bloque convolucional;* para *ResNet50,* por la composición de un *bloque residual;* para *DenseNet121,* por la composición de un *bloque denso* y, finalmente, para *InceptionV3,* por la composición de un *módulo Inception.* En la Tabla 12 se muestra el número total de capas ajustables durante la fase de entrenamiento, en función de la estrategia de *tunning* utilizada para cada algoritmo.

En la última estrategia, denominada *"ALL",* se ajustan completamente todos los parámetros presentes en las arquitecturas de red de forma que únicamente se aprovecha el conocimiento previamente adquirido en la base de datos *Imagenet* para inicializar los pesos de cada arquitectura. Para comprobar la efectividad de esta estrategia, también se han entrenado las arquitecturas de red inicializándolas con pesos completamente aleatorios.

Arquitectura	Capas totales	1 FT	2 FT	3 FT	4 FT
VGG16	16	3	6	9	11
ResNet50	50	9	27	39	48
InceptionV3	86	9	18	24	34
DenseNet121	121	33	82	107	119

Tabla 12. *Número de capas entrenables en cada estrategia de ajuste fino utilizada. Dado que las capas de pooling no contienen parámetros entrenables y que las capas de batch normalization están en modo de inferencia, no se contabilizan en la tabla. Se excluye en el recuento, las capas FC diseñadas en el proyecto.*

A continuación, se muestran gráficamente las estrategias de ajuste fino planteadas en el proyecto.

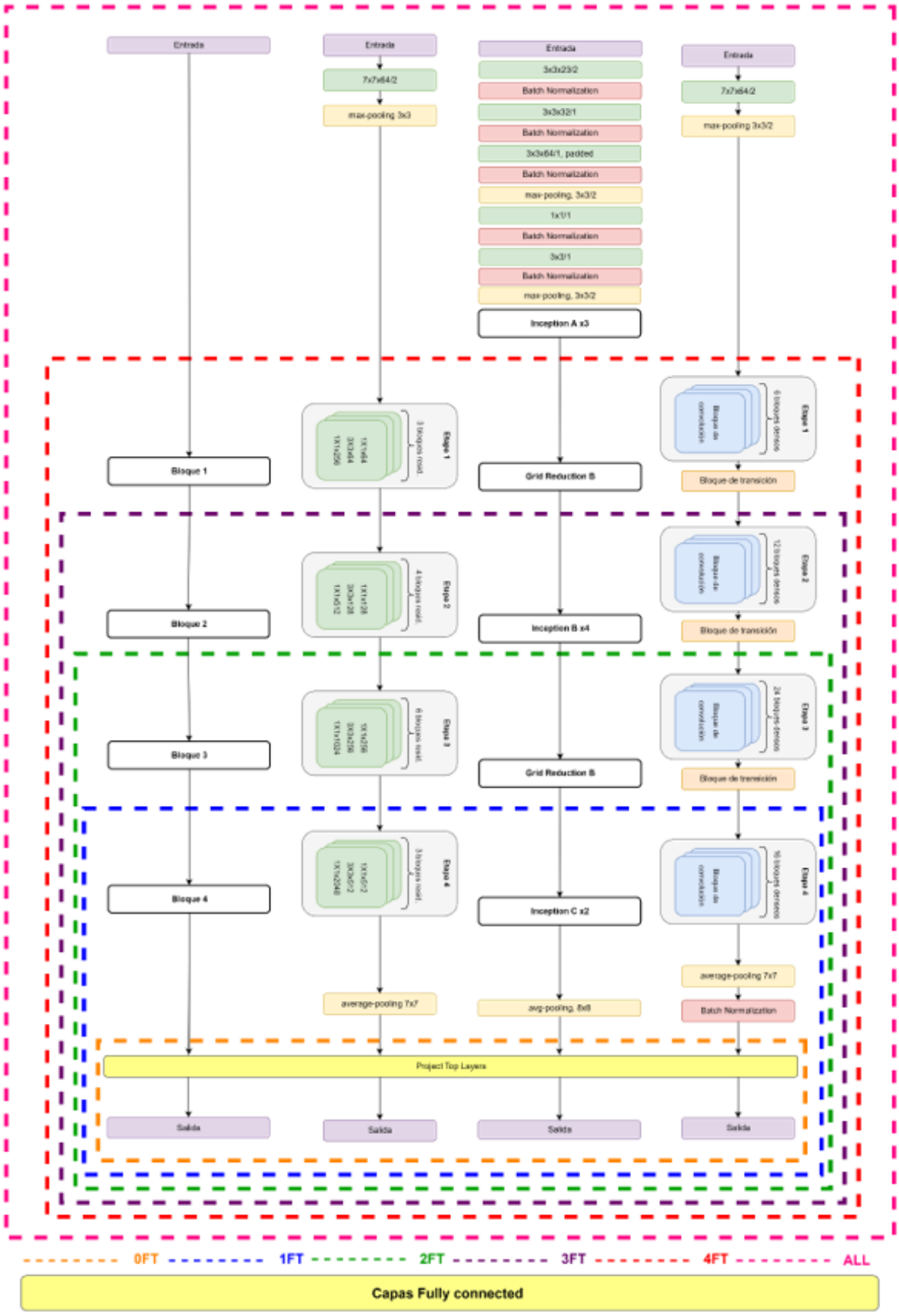


Figura 33. *Esquema con los modelos de red y las estrategias de ajuste fino de parámetros implementadas.*

4.2.1.2 Entrenamiento de los modelos

Las arquitecturas de red implementadas en este proyecto se entrenarán utilizando un total de 150 épocas. De nuevo, con la finalidad de garantizar una correcta generalización de los modelos, se ha adoptado una estrategia de *Early Stopping* para parar el entrenamiento una vez el error de validación empiece a incrementarse [70]. Dada la inestabilidad de la función de coste producida por el cálculo del gradiente descendiente, se ha dejado un periodo de paciencia de 20 épocas de modo que, si una vez finalizado este periodo no se ha producido un decremento del error de validación, se parará el entrenamiento recuperando aquellos parámetros que generalicen mejor.

Por otra parte, para encontrar los parámetros que minimizan la función de coste de cada modelo, se ha utilizado un optimizador *Adam (Adaptative Moment Estimation),* ya que adapta la actualización de los pesos de la red teniendo en cuenta, por una parte, el momento de los gradientes de primer y segundo orden (principio de optimización *Momentum* [71]) y, por otra, la importancia relativa de las características asociadas a cada parámetro, es decir, parámetros relacionados con características más frecuentes producirán actualizaciones más pequeñas, mientras que parámetros asociados a características menos frecuentes producirán actualizaciones más grandes (principio de optimización *AdaGrad* [72]*)* [73], [74]. El *learning rate* utilizado para el optimizador *Adam* es de 1e-*3*.

Cabe destacar que, para las estrategias "*1FT", "2FT", "3FT"* y "4FT", se debe de tener en cuenta que la inicialización aleatoria de los parámetros de las capas *FC* situadas a la salida de cada arquitectura, podría producir cambios bruscos en los parámetros ya aprendidos por cada modelo entrenado en *Imagenet*. En este sentido, se ha dividido el entrenamiento de estas estrategias en dos fases distintas en: la primera, se entrenan exclusivamente los parámetros pertenecientes a las capas *FC,* utilizando *learning rate* de $1e^{-3}$ durante un total de 5 épocas; en la segunda, se descongelan los pesos de las capas definidas en cada estrategia para ser entrenados durante un total de 150 épocas utilizando un *learning rate* de $1e^{-4}$. En ambas fases, el optimizador utilizado es *Adam.*

Finalmente, la actualización de los parámetros de la red se realizará utilizando una técnica de *mini-batch gradient descent (MBGD).* En esta, se dividirá el conjunto de datos de entrenamiento en un conjunto de lotes pequeños o *batches,* que serán utilizados para calcular el coste generado por el modelo. De esta forma, se pretende encontrar un equilibrio entre el coste computacional generado por el cálculo del gradiente descendente y la eficiencia del mismo [75]. El tamaño de cada lote, dependerá del set de datos y de la arquitectura de red utilizada en cada experimento.

4.2.2 Combinación secuencial de modelos y frontera de decisión

La construcción de un único modelo a partir de la combinación secuencial de clasificadores se ha realizado seleccionando aquellas arquitecturas de red que presentan, por una parte, una mayor tasa de acierto a la hora de clasificar correctamente el tipo de cáncer presente en una mamografía y, por otra, una mayor capacidad de generalización a la hora de clasificar instancias nuevas.

Durante el diagnóstico de cáncer de seno, es importante clasificar correctamente tanto las muestras benignas como las malignas. En primer lugar, determinar como maligno un cáncer benigno implica la realización de biopsias innecesarias ocasionando, muchas veces, ansiedad, disgusto y enfado por parte de las personas afectadas [76]. Además, la realización de este tipo de tratamientos supone un impacto económico y temporal elevado. En concreto, únicamente entre el 15 % y el 30 % de las biopsias han sido realizadas en cánceres malignos [37], [77]. Al mismo tiempo, diagnosticar como benigno un cáncer maligno implica graves problemas de salud, pudiendo ocasionar la muerte en aquellos casos en los que las células cancerígenas se han expandido a otras zonas del cuerpo sin realizarse previamente ningún tipo de intervención (cáncer metastásico) [78].

En consecuencia, la métrica utilizada para seleccionar cada una de las arquitecturas de red que formarán parte de la combinación secuencial de modelos, deberá de tener en cuenta tanto la tasa de verdaderos positivos (muestras malignas clasificadas correctamente), como la tasa de verdaderos negativos (muestras benignas clasificadas correctamente). En este sentido, la métrica seleccionada ha sido el área bajo la curva (*AUC, de Area Under the Curve)* [79].

Adicionalmente, con el objetivo de combinar aquellos modelos que presenten una mayor capacidad de generalización, se ha aplicado una técnica de *Bootstrap* sobre el conjunto de datos de validación para calcular, con un nivel de significación del 5%, el valor del área bajo la curva de cada arquitectura. En la Tabla 13 se muestran las estrategias de entrenamiento seleccionadas para cada modelo.

	VGG16	***ResNet50***	***DenseNet121***	***InceptionV3***
Estrategia	2FT	2FT	ALL	4FT

Tabla 13. *Arquitecturas de red seleccionadas para la combinación secuencial de modelos.*

Llegados a este punto, las predicciones realizadas por cada una de las arquitecturas de red servirán para alimentar un regresor *Random Forest* encargado de indicar la probabilidad que tiene una muestra de ser maligna. Para seleccionar la profundidad y el número de árboles de decisión que formarán parte de este algoritmo, se ha utilizado una estrategia de *GridSearch* optimizando el valor del área bajo la curva obtenido para el conjunto de datos de validación. El árbol resultante está formado por 3 niveles y 450 estimadores.

Finalmente, para traducir cada probabilidad generada por el modelo final en una clasificación, se ha seleccionado aquel *threshold* que maximiza tanto la tasa de verdaderos positivos como la tasa de verdaderos negativos en el set de datos de validación. Para ello, se ha utilizado el estadístico *J de Youlden* [80]:

$$J = Sensibilidad + Especificidad - 1 = TPR + TNR - 1 \qquad \textit{(Ec. 5)}$$

donde TPR es la tasa de verdaderos positivos y TNR es la tasa de verdaderos negativos.

El valor del *threshold* óptimo que ha maximizado dicha estadística ha sido del 63.14 %. De este modo, toda muestra cuya probabilidad sea superior o igual al *threshold* será clasificada como maligna y, en caso contrario, como benigna.

5. Experimentos y resultados

En esta sección se mostrarán los experimentos realizados durante el transcurso del proyecto para construir la herramienta de clasificación de lesiones de seno.

Juntamente con cada ensayo, aparte de mostrar las matrices de confusión obtenidas, se calcularán métricas como el área bajo la curva, la precisión, la sensibilidad, la exactitud y el valor *F1* para medir la actuación de cada clasificador.

A continuación, se definen las métricas mencionadas anteriormente:

$$Precisión\ (\%) = \frac{TP}{TP + FP} \qquad \textit{(Ec. 6)}$$

$$Sensibilidad\ (\%) = \frac{TP}{TP + FN} \qquad \textit{(Ec. 7)}$$

$$Exactitud\ (\%) = \frac{TN}{TN + FP} \qquad \textit{(Ec. 8)}$$

$$F1 = 2 \cdot \frac{Precisión\ x\ Sensibilidad}{Precisión + Sensibilidad} \qquad \textit{(Ec. 9)}$$

donde TP son los verdaderos positivos (muestras malignas clasificadas correctamente); TN son los de verdaderos negativos (muestras benignas clasificadas correctamente); FP son los falsos positivos (muestras benignas clasificadas como malignas) y FN son los falsos negativos (muestras malignas clasificadas como benignas).

Cabe destacar que todas las métricas de evaluación han sido calculadas para los conjuntos de datos de entrenamiento, validación y test aplicando técnicas de *bootstraping* utilizando 1000 iteraciones y una muestra compuesta por el 75 % de los datos disponibles en cada subconjunto. De esta forma, cada métrica será representa con un nivel de significancia del 5%.

Por otra parte, tal y como se ha expuesto en la sección "*4.2.1 Redes neuronales convolucionales",* se han diseñado dos modelos de red distintos diferenciados por el número de capas densas utilizadas a la salida de cada red neuronal convolucional. En vista que la capacidad de almacenaje del sistema utilizado es limitada, se han definido dos tamaños de *batch* distintos para cada uno de estos diseños. Para la "*arquitectura simple",* el tamaño de *batch* ha sido de 18 instancias, y para la *"arquitectura compleja",* de 14 instancias. Adicionalmente, destacar que para poder realizar experimentos homogéneos, todas las arquitecturas de red neuronal convolucional han utilizado el mismo tamaño de *batch* independientemente del número de parámetros o de la profundidad presente en cada modelo.

Finalmente, todos los ensayos han sido implementados utilizando una tarjeta gráfica *NVIDIA Geforce GTX 1050* con 4Gb de memoria.

5.1 Experimentos con imágenes completas

En esta sección se describirán los experimentos realizados utilizando el conjunto de datos formado por imágenes mamográficas completas.

Tal y como se puede observar en la Figura 34, el tiempo necesario para entrenar cada modelo aumenta a medida que el número de parámetros a aprender es mayor. Así pues, las estrategias "*0FT"* de cada modelo han necesitado un tiempo de entrenamiento mucho menor que el resto de estrategias. Por otra parte, parece que la inicialización de los pesos no ha tenido impacto en el tiempo de convergencia necesario para entrenar cada arquitectura con la estrategia *"ALL"*. Este hecho podría estar relacionado con la volumetría de datos utilizada para entrenar cada arquitectura.

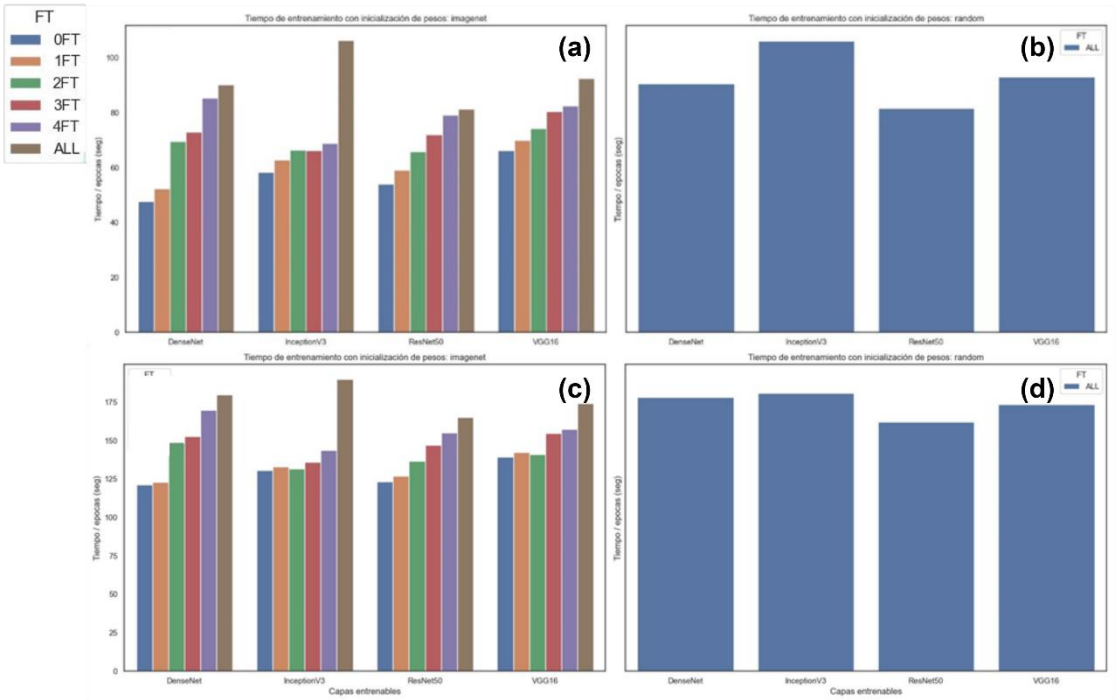


Figura 34. *Tiempo de ejecución. (a) 0FT, 1FT, 2FT, 3FT, 4FT y ALL con pesos inicializados en Imagenet y arquitectura simple. (b) tiempo de ejecución de la estrategia ALL con pesos aleatorios y arquitectura simple. (c) 0FT, 1FT, 2FT, 3FT, 4FT y ALL con pesos inicializados en Imagenet y arquitectura compleja. (d) tiempo de ejecución de la estrategia ALL con pesos aleatorios y arquitectura compleja.*

En la Figura 35, se muestran las gráficas de pérdidas generadas por cada estrategia de ajuste fino utilizada. En general, la convergencia de la arquitectura *VGG16* (color rojo) parece haber sido mucho más lenta que en el resto de redes. El principal causante de este comportamiento podría ser el hecho de que esta red no contiene conexiones residuales o convoluciones factorizadas que permitan agilizar la actualización del gradiente a partir del paso de *backpropagation.*

Por otra parte, el número de épocas necesarias para entrenar cada arquitectura de red ha sido distinto. Además, el sobreajuste de datos ha producido que en muchas ocasiones, los modelos paralizasen su entrenamiento antes de llegar a las 150 épocas.

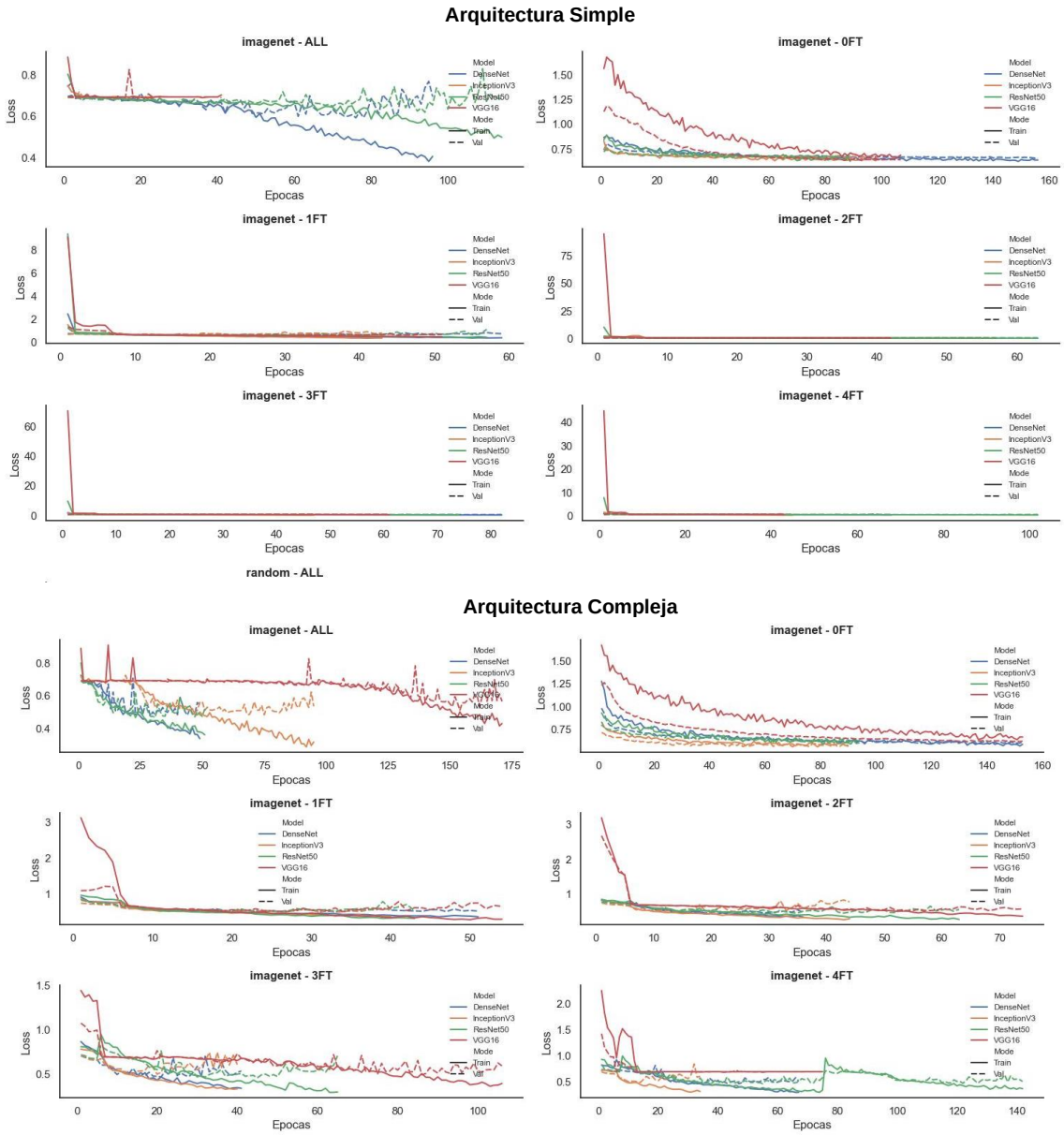


Figura 35. *Gráfica de pérdidas. Se muestra el valor de la función de pérdidas para las distintas estrategias de entrenamiento utilizadas.*

La efectividad de cada arquitectura puede variar en función de la estrategia de *fine tunning* utilizada. La Figura 36 muestra como el valor *AUC* incrementa a medida que el número de parámetros entrenables de cada red va aumentando, no obstante, llega un punto en que deja de ser óptimo y la capacidad clasificadora de cada algoritmo empieza a verse afectada. Adicionalmente, la *arquitectura compleja* presenta mejores métricas que la *arquitectura simple.*

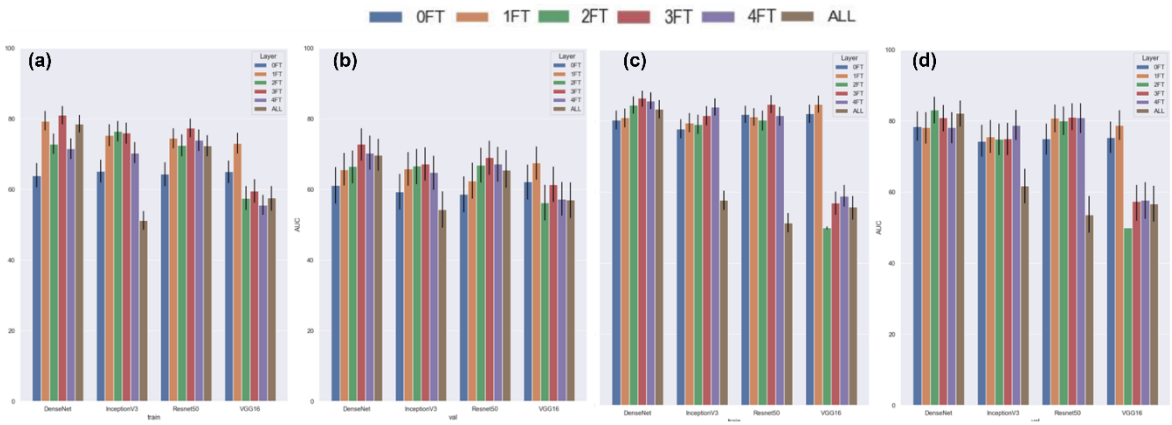


***Figura 36**. Áreas bajo la curva en función de la estrategia de entrenamiento utilizada a partir de los pesos de Imagenet. (a) Métricas AUC para el conjunto de datos de entrenamiento utilizando la arquitectura simple. (b) Métricas AUC para el conjunto de datos de validación utilizando la arquitectura simple. (c) Métricas AUC para el conjunto de datos de entrenamiento utilizando la arquitectura compleja. (d) Métricas AUC para el conjunto de datos de validación utilizando la arquitectura compleja.*

A continuación, las figuras (de la Figura 37 a la Figura 41) muestran las métricas de precisión, exactitud, sensibilidad, f1 y *AUC* obtenidas para los distintos conjuntos de datos mediante las arquitecturas de red *simple* y *compleja.*

Ante los resultados obtenidos, se puede observar como la combinación secuencial de clasificadores mediante el uso de un algoritmo de *Random Forest* ha incrementado considerablemente la actuación del modelo diseñado a la hora de clasificar las imágenes mamográficas. En algunas ocasiones, el incremento producido ha sido casi del 10 %.

Por otra parte, la *arquitectura compleja* ha obtenido unas métricas de clasificación muy superiores a las obtenidas por la *arquitectura simple.*

mode	metric	DenseNet	InceptionV3	Resnet50	VGG16	RandomForest
train	AUC	0.72 [0.69, 0.75]	0.76 [0.73, 0.79]	0.77 [0.75, 0.80]	0.73 [0.70, 0.76]	0.82 [0.80, 0.85]
train	accuracy	0.72 [0.69, 0.75]	0.76 [0.73, 0.79]	0.77 [0.74, 0.80]	0.73 [0.70, 0.76]	0.82 [0.80, 0.85]
train	precision	0.72 [0.69, 0.75]	0.76 [0.73, 0.79]	0.79 [0.77, 0.82]	0.75 [0.72, 0.77]	0.84 [0.81, 0.86]
train	recall	0.72 [0.69, 0.75]	0.76 [0.73, 0.79]	0.77 [0.74, 0.80]	0.73 [0.70, 0.76]	0.82 [0.80, 0.85]
train	f1	0.72 [0.68, 0.75]	0.76 [0.73, 0.79]	0.77 [0.74, 0.79]	0.72 [0.69, 0.75]	0.82 [0.79, 0.85]
val	AUC	0.70 [0.66, 0.75]	0.67 [0.63, 0.72]	0.69 [0.64, 0.73]	0.68 [0.63, 0.72]	0.76 [0.72, 0.81]
val	accuracy	0.70 [0.66, 0.75]	0.67 [0.63, 0.73]	0.69 [0.64, 0.74]	0.67 [0.62, 0.73]	0.76 [0.72, 0.81]
val	precision	0.71 [0.66, 0.75]	0.68 [0.63, 0.72]	0.70 [0.65, 0.74]	0.69 [0.64, 0.74]	0.76 [0.72, 0.81]
val	recall	0.70 [0.65, 0.75]	0.67 [0.63, 0.72]	0.69 [0.64, 0.74]	0.67 [0.63, 0.73]	0.76 [0.72, 0.81]
val	f1	0.70 [0.65, 0.75]	0.67 [0.63, 0.73]	0.69 [0.64, 0.74]	0.67 [0.62, 0.72]	0.76 [0.72, 0.81]

***Figura 37.** Métricas para los conjuntos de entrenamiento y validación en función del modelo utilizado para la arquitectura simple. Para cada métrica se muestra el intervalo de confianza con un nivel de significación del 5%.*

mode	metric	DenseNet	InceptionV3	Resnet50	VGG16	RandomForest
test	AUC	0.73	0.71	0.64	0.70	0.74
test	accuracy	0.72	0.71	0.64	0.70	0.74
test	precision	0.74	0.72	0.66	0.72	0.76
test	recall	0.72	0.71	0.64	0.70	0.74
test	f1	0.72	0.71	0.63	0.69	0.74

***Figura 38**. Métricas el conjunto de test en función de cada modelo utilizado en la arquitectura simple.*

mode	metric	DenseNet	InceptionV3	Resnet50	VGG16	RandomForest
train	AUC	0.86 [0.83, 0.88]	0.84 [0.81, 0.86]	0.85 [0.83, 0.87]	0.85 [0.83, 0.87]	0.90 [0.88, 0.92]
train	accuracy	0.86 [0.83, 0.88]	0.84 [0.81, 0.86]	0.85 [0.82, 0.87]	0.85 [0.82, 0.87]	0.90 [0.88, 0.92]
train	precision	0.86 [0.83, 0.88]	0.84 [0.82, 0.86]	0.85 [0.82, 0.87]	0.85 [0.82, 0.87]	0.90 [0.88, 0.92]
train	recall	0.86 [0.83, 0.88]	0.84 [0.81, 0.86]	0.85 [0.82, 0.87]	0.85 [0.83, 0.87]	0.90 [0.88, 0.92]
train	f1	0.86 [0.83, 0.88]	0.84 [0.81, 0.86]	0.85 [0.82, 0.87]	0.85 [0.82, 0.87]	0.90 [0.88, 0.92]
val	AUC	0.78 [0.74, 0.82]	0.79 [0.75, 0.82]	0.81 [0.77, 0.85]	0.79 [0.74, 0.83]	0.85 [0.81, 0.88]
val	accuracy	0.78 [0.74, 0.82]	0.79 [0.74, 0.82]	0.81 [0.77, 0.85]	0.79 [0.74, 0.83]	0.85 [0.81, 0.88]
val	precision	0.78 [0.74, 0.82]	0.79 [0.75, 0.83]	0.81 [0.78, 0.85]	0.79 [0.75, 0.83]	0.85 [0.81, 0.88]
val	recall	0.78 [0.74, 0.82]	0.79 [0.74, 0.83]	0.81 [0.77, 0.85]	0.79 [0.75, 0.83]	0.85 [0.81, 0.88]
val	f1	0.78 [0.74, 0.82]	0.79 [0.75, 0.83]	0.81 [0.77, 0.85]	0.79 [0.75, 0.83]	0.85 [0.81, 0.88]

Figura 39. *Métricas para los conjuntos de entrenamiento y validación en función del modelo utilizado para la arquitectura compleja. Para cada métrica, se muestra el intervalo de confianza con un nivel de significación del 5%.*

mode	metric	DenseNet	InceptionV3	Resnet50	VGG16	RandomForest
test	AUC	0.85	0.85	0.83	0.89	0.92
test	accuracy	0.84	0.84	0.83	0.88	0.91
test	precision	0.87	0.86	0.85	0.89	0.92
test	recall	0.84	0.84	0.83	0.88	0.91
test	f1	0.84	0.84	0.83	0.88	0.91

Figura 40*. Métricas el conjunto de test en función de cada modelo utilizado en la arquitectura compleja.*

Finalmente, la Figura 41 y la Figura 42, muestran las matrices de confusión obtenidas por la combinación secuencial de clasificadores para las *arquitecturas simple* y *compleja,* respectivamente.

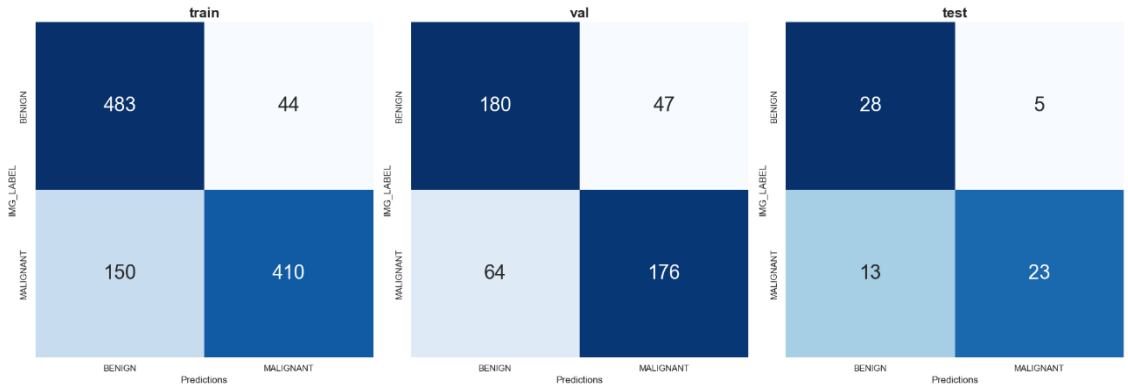


Figura 41. *Matrices de confusión generadas por el Random Forest para los conjuntos de entrenamiento, validación y test.*

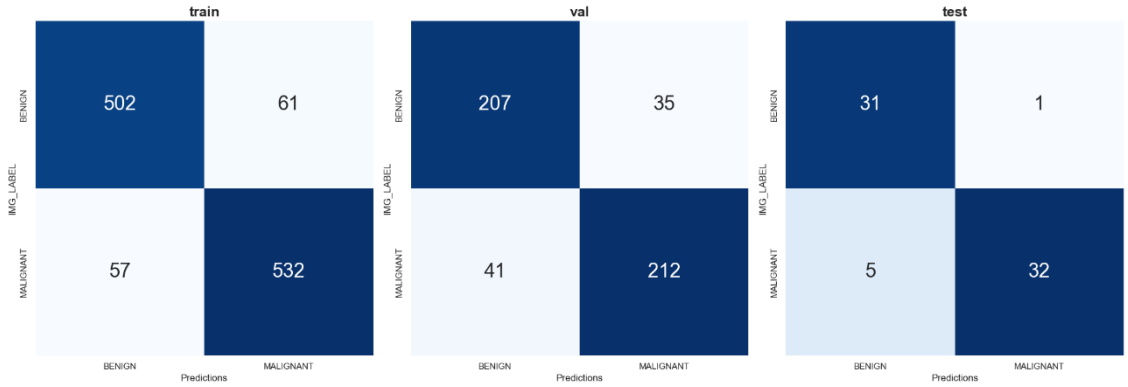


Figura 42. *Matrices de confusión generadas por el Random Forest para los conjuntos de entrenamiento, validación y test.*

5.2 Experimentos con recortes de las zonas de interés

En esta sección se describirán los experimentos realizados utilizando el conjunto de datos formado por los recortes de las zonas de interés.

Tal y como se puede observar en la Figura 43 y en la Figura 44, tanto el tiempo necesario para entrenar las arquitecturas de red, como el valor de la función de coste de cada modelo, presentan comportamientos muy similares a los obtenidos en el experimento con imágenes completas.

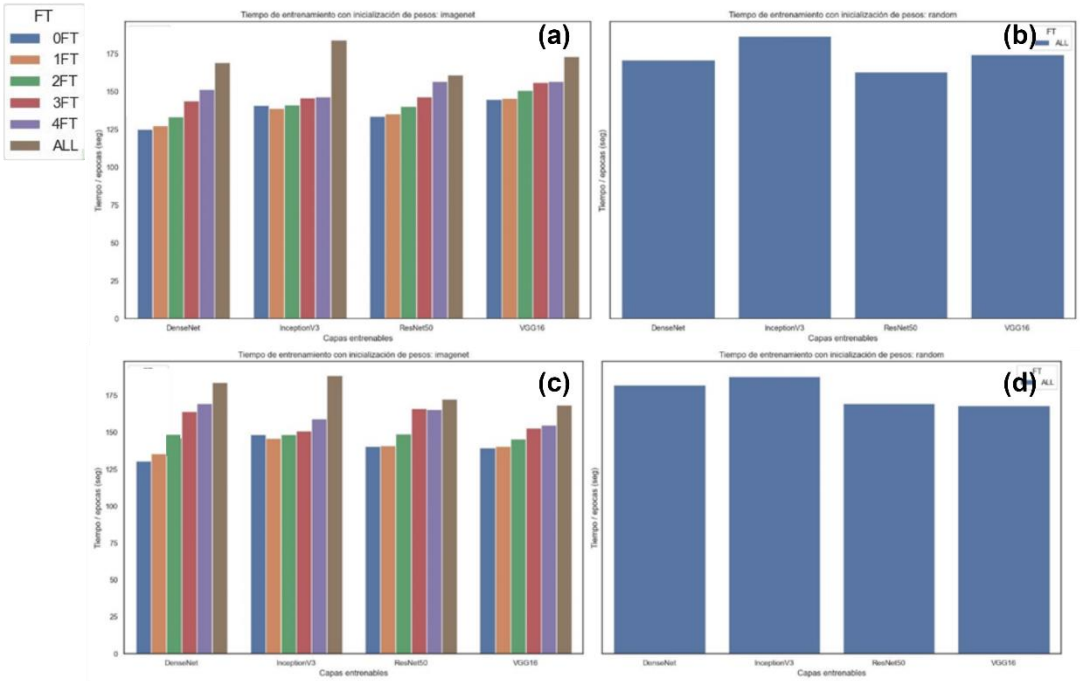


Figura 43.*Tiempo de ejecución. (a) 0FT, 1FT, 2FT, 3FT, 4FT y ALL con pesos inicializados en Imagenet y arquitectura simple. (b) tiempo de ejecución de la estrategia ALL con pesos aleatorios y arquitectura simple. (c) 0FT, 1FT, 2FT, 3FT, 4FT y ALL con pesos inicializados en Imagenet y arquitectura compleja. (d) tiempo de ejecución de la estrategia ALL con pesos aleatorios y arquitectura compleja.*

Adicionalmente, en la Figura 45 se puede ver de nuevo como la efectividad de cada arquitectura varia en función de la estrategia de *fine tunning* utilizada. Además, el uso de una arquitectura más profunda incrementa considerablemente la capacidad clasificadora del modelo, siendo las redes *ResNet50* o *DenseNet121* las mejores. En este mismo sentido, el diseño de la *arquitectura compleja* compuesta por 5 capas *FC* al final de cada módulo convolucional ha obtenido mejores valores de *AUC* para todas las estrategias de *fine tunning* definidas en comparación de la *arquitectura simple*.

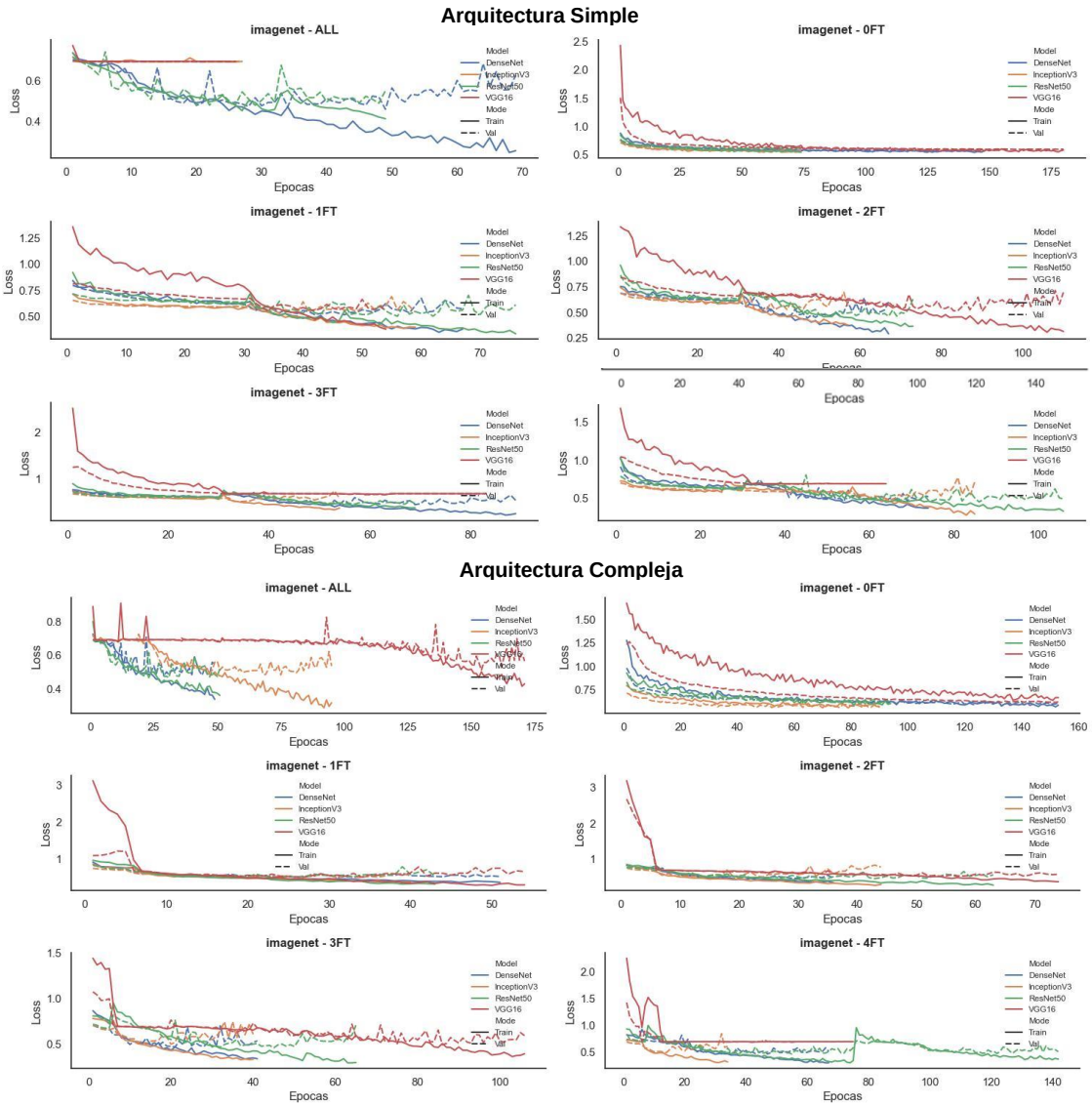


Figura 44. *Gráfica de pérdidas. Se muestra el valor de la función de pérdidas para las distintas estrategias de entrenamiento utilizadas.*

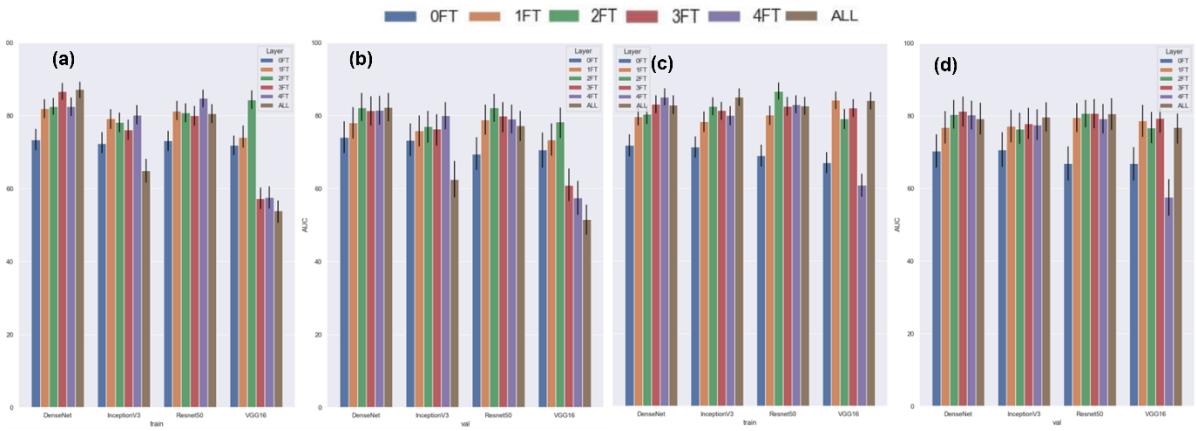


Figura 45. *Áreas bajo la curva en función de la estrategia de entrenamiento utilizada a partir de los pesos de Imagenet. (a) Métricas AUC para el conjunto de datos de entrenamiento utilizando la arquitectura simple. (b) Métricas AUC para el conjunto de datos de validación utilizando la arquitectura simple. (c) Métricas AUC para el conjunto de datos de entrenamiento utilizando la arquitectura compleja. (d) Métricas AUC para el conjunto de datos de validación utilizando la arquitectura compleja.*

En las siguientes figuras (de la Figura 46 a la Figura 41) se muestran las métricas de precisión, exactitud, sensibilidad, F1 y *AUC* obtenidas para los distintos conjuntos de datos mediante las arquitecturas de red *simple* y *compleja.*

Ante los resultados obtenidos, se puede observar cómo las métricas obtenidas sobre las zonas de interés son muy superiores a las métricas obtenidas sobre las imágenes completas. En este sentido, la poca representación espacial de las patologías, combinada con el escalado realizado durante el procesado de imágenes, producen efectos adversos a la hora de realizar clasificaciones sobre imágenes completas.

Asimismo, el error cometido por la *arquitectura compleja* a la hora de clasificar correctamente instancias de la clase maligna (sensibilidad entre el 86 % y el 92 %) es inferior que el que comete un radiólogo (sensibilidad entre el 77 % y el 87 %). Nuevamente, la *arquitectura compleja* ha obtenido unas métricas de clasificación muy superiores a las obtenidas por la *arquitectura simple.*

mode	metric	DenseNet	InceptionV3	Resnet50	VGG16	RandomForest
train	AUC	0.87 [0.85, 0.89]	0.80 [0.77, 0.83]	0.81 [0.78, 0.83]	0.84 [0.82, 0.87]	0.90 [0.87, 0.92]
train	accuracy	0.87 [0.85, 0.89]	0.80 [0.77, 0.83]	0.81 [0.78, 0.83]	0.84 [0.82, 0.87]	0.89 [0.87, 0.91]
train	precision	0.87 [0.85, 0.90]	0.80 [0.77, 0.83]	0.81 [0.78, 0.83]	0.84 [0.82, 0.87]	0.90 [0.88, 0.92]
train	recall	0.87 [0.85, 0.89]	0.80 [0.77, 0.83]	0.81 [0.78, 0.83]	0.85 [0.82, 0.87]	0.89 [0.87, 0.92]
train	f1	0.87 [0.85, 0.89]	0.80 [0.77, 0.83]	0.81 [0.78, 0.83]	0.84 [0.82, 0.87]	0.89 [0.87, 0.91]
val	AUC	0.82 [0.78, 0.86]	0.80 [0.76, 0.84]	0.82 [0.78, 0.86]	0.78 [0.74, 0.82]	0.84 [0.80, 0.87]
val	accuracy	0.82 [0.78, 0.86]	0.80 [0.76, 0.84]	0.82 [0.78, 0.86]	0.78 [0.74, 0.82]	0.84 [0.80, 0.88]
val	precision	0.82 [0.78, 0.86]	0.80 [0.76, 0.84]	0.82 [0.78, 0.86]	0.79 [0.74, 0.82]	0.84 [0.81, 0.88]
val	recall	0.82 [0.78, 0.86]	0.80 [0.76, 0.84]	0.82 [0.78, 0.86]	0.78 [0.74, 0.82]	0.84 [0.80, 0.87]
val	f1	0.82 [0.78, 0.86]	0.80 [0.76, 0.84]	0.82 [0.78, 0.86]	0.78 [0.74, 0.83]	0.84 [0.80, 0.87]

Figura 46. *Métricas para los conjuntos de entrenamiento y validación en función del modelo utilizado. Para cada métrica, se muestra el intervalo de confianza con un nivel de significación del 5%.*

mode	metric	DenseNet	InceptionV3	Resnet50	VGG16	RandomForest
test	AUC	0.82	0.82	0.84	0.79	0.84
test	accuracy	0.81	0.82	0.85	0.80	0.82
test	precision	0.83	0.83	0.85	0.80	0.85
test	recall	0.81	0.82	0.85	0.80	0.82
test	f1	0.81	0.83	0.85	0.80	0.83

Figura 47*. Métricas el conjunto de test en función de cada modelo utilizado.*

mode	metric	DenseNet	InceptionV3	Resnet50	VGG16	RandomForest
train	AUC	0.85 [0.83, 0.88]	0.85 [0.83, 0.88]	0.83 [0.80, 0.85]	0.82 [0.80, 0.85]	0.95 [0.94, 0.97]
train	accuracy	0.85 [0.83, 0.87]	0.85 [0.83, 0.88]	0.83 [0.80, 0.85]	0.82 [0.80, 0.85]	0.95 [0.94, 0.97]
train	precision	0.85 [0.83, 0.88]	0.85 [0.83, 0.88]	0.83 [0.81, 0.86]	0.82 [0.80, 0.85]	0.95 [0.94, 0.97]
train	recall	0.85 [0.83, 0.88]	0.85 [0.83, 0.87]	0.83 [0.80, 0.85]	0.82 [0.80, 0.85]	0.95 [0.94, 0.96]
train	f1	0.85 [0.83, 0.87]	0.85 [0.83, 0.87]	0.83 [0.80, 0.85]	0.82 [0.80, 0.85]	0.95 [0.94, 0.97]
val	AUC	0.80 [0.76, 0.84]	0.80 [0.76, 0.84]	0.81 [0.77, 0.85]	0.79 [0.75, 0.84]	0.89 [0.85, 0.92]
val	accuracy	0.80 [0.76, 0.84]	0.80 [0.76, 0.84]	0.81 [0.77, 0.85]	0.79 [0.75, 0.84]	0.89 [0.85, 0.92]
val	precision	0.80 [0.76, 0.85]	0.80 [0.76, 0.84]	0.81 [0.77, 0.85]	0.79 [0.75, 0.84]	0.89 [0.86, 0.92]
val	recall	0.80 [0.76, 0.84]	0.80 [0.75, 0.84]	0.81 [0.77, 0.85]	0.79 [0.75, 0.83]	0.89 [0.86, 0.92]
val	f1	0.80 [0.76, 0.84]	0.80 [0.76, 0.84]	0.81 [0.77, 0.84]	0.79 [0.76, 0.84]	0.89 [0.85, 0.92]

Figura 48. *Métricas para los conjuntos de entrenamiento y validación en función del modelo utilizado. Para cada métrica se muestra el intervalo de confianza con un nivel de significación del 5%.*

mode	metric	DenseNet	InceptionV3	Resnet50	VGG16	RandomForest
test	AUC	0.85	0.80	0.84	0.80	0.92
test	accuracy	0.85	0.80	0.84	0.80	0.92
test	precision	0.85	0.80	0.84	0.80	0.92
test	recall	0.85	0.80	0.84	0.80	0.92
test	f1	0.85	0.80	0.84	0.80	0.92

***Figura 49**. Métricas el conjunto de test en función de cada modelo utilizado.*

Finalmente, la Figura 41 y la Figura 42 muestran las matrices de confusión obtenidas por la combinación secuencial de clasificadores para las *arquitecturas simple* y *compleja,* respectivamente.

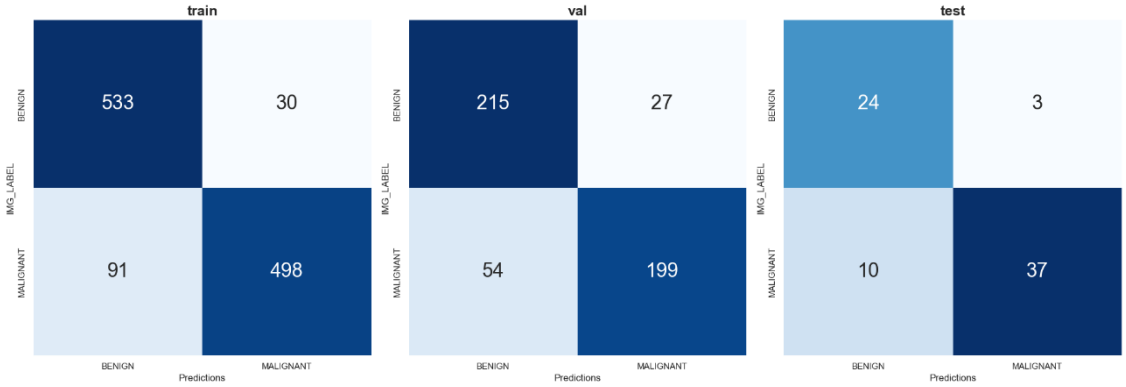


***Figura 50.** Matrices de confusión generadas por el Random Forest para los conjuntos de entrenamiento, validación y test*

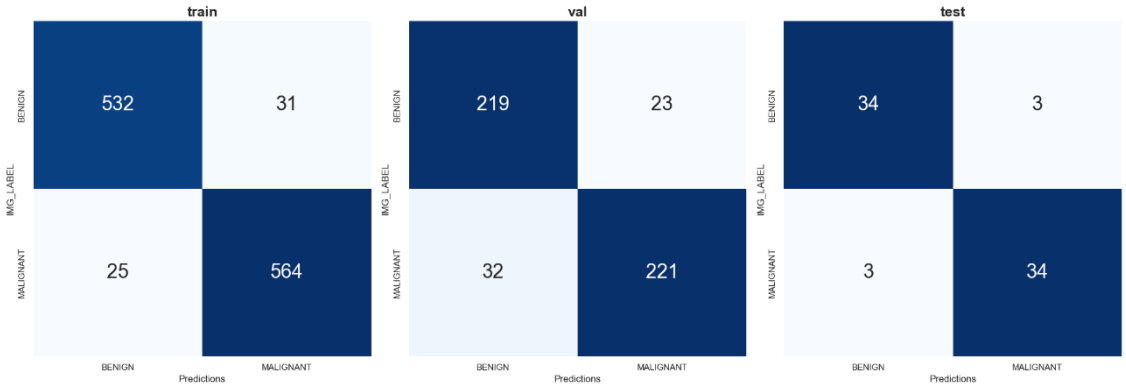


***Figura 51.** Matrices de confusión generadas por el Random Forest para los conjuntos de entrenamiento, validación y test.*

6. Conclusiones

Al finalizar el proyecto, se ha desarrollado con éxito la implementación de una herramienta automática que permite clasificar las lesiones presentes en imágenes mamográficas como malignas o benignas. Ante esta situación, cualquier especialista podría utilizarla como herramienta de soporte durante la realización de diagnósticos. Adicionalmente, el *software* producido no requiere de ningún tipo de dependencia para ser instalado, pudiéndose ejecutar desde cualquier ordenador y facilitando que su distribución sea relativamente sencilla.

Para la realización de esta herramienta, se han analizado distintas arquitecturas de red que componen el estado del arte en tareas de clasificación de imágenes médicas, como son *VGG16, DenseNet121, Resnet50* e *InceptionV3.* A partir de los experimentos realizados, se puede observar cómo la estructura, juntamente con la profundidad y el número de parámetros que utiliza cada red, afectan a la hora de clasificar las observaciones. Aquellas arquitecturas que presentan una mayor profundidad y contienen conexiones residuales en su diseño han demostrado tener una mejor actuación que el resto. En concreto, *DenseNet121* ha obtenido las mejores métricas de clasificación a lo largo de todos los experimentos. Ante esta situación, la conectividad entre las capas iniciales y las capas finales de cada red parece incrementar la eficiencia de esta sin comprometer factores como el tiempo de entrenamiento o el coste computacional necesario.

Asimismo, para poder aplicar las arquitecturas de red mencionadas anteriormente al problema de clasificación planteado en este proyecto, se han diseñado dos bloques compuestos por capas *Fully-Connected* a la salida de cada modelo. Ambos diseños contenían un número de capas distinto. De nuevo, el bloque con mayor profundidad ha demostrado tener un poder clasificatorio mayor en los experimentos realizados.

Los conjuntos de datos *Curated Breast Imaging Subset of Digital Database for Screening Mammography (CBIS-DDSM)* y *Inbreast* han permitido realizar el entrenamiento de las arquitecturas de red implementadas en el proyecto. Para evitar la creación de algoritmos poco generalizables producida por la escasez de datos disponibles, se han implementado técnicas de regularización, de *dropout y* de d*ata augmentation.*

Adicionalmente, se han utilizado distintas estrategias de *transfer learning* y de *fine tunning* modificando el número de parámetros y de capas a ajustar durante el entrenamiento de cada algoritmo. De los experimentos, se puede concluir que entrenar aquellas capas más próximas a la salida de la red puede ser beneficioso, dado que el modelo está aprendiendo características propias del campo de estudio utilizado. Sin embargo, existe un punto en que este procedimiento deja de ser óptimo y la capacidad clasificadora de cada dominio empieza a decaer. Un posible motivo podría ser la poca volumetría de observaciones presente en el set de datos utilizado. En este aspecto, el modelo no tiene capacidad suficiente para aprender características de alto nivel que sean suficientemente generalizables, produciendo un sobreajuste de los parámetros y reduciendo su eficiencia. Por lo tanto, el uso de técnicas de *transfer learning*

juntamente con estrategias de *fine tunning* puede resultar beneficioso cuando el conjunto de datos utilizado es pequeño.

Con el objetivo de incrementar el poder predictivo de cada red neuronal convolucional, se ha implementado una combinación secuencial de clasificadores utilizando un algoritmo de *Random Forest.* Este modelo es el encargado de realizar la predicción final de cada muestra a partir de las clasificaciones individuales realizadas por cada arquitectura de red. Esta configuración permite incrementar la tasa de sensibilidad en un 4 % con respecto al resto de redes, asemejándose al incremento producido cuando se combinan múltiples opiniones de especialistas durante el análisis de cáncer de seno.

Finalmente, para poder medir la eficacia de la herramienta en un escenario real, se han recortado 74 muestras a partir de las anotaciones presentes en el set de datos *Mammographic Image Analysis Society (MIAS)*. La actuación del aplicativo supera a la de un especialista mostrando tasas de sensibilidad entre el 86 % y el 92 % versus el 77 % y el 87 %, respectivamente.

Pese a que el aplicativo desarrollado es capaz de clasificar correctamente instancias nuevas, está limitado por la necesidad de tener las zonas de interés de cada mamografía anotadas. En este aspecto, una posible línea de mejora y futuro desarrollo sería la implementación de un sistema *end-to-end* que permitiera detectar las zonas anómalas de una mamografía. Esta implementación podría realizarse como una línea de trabajo paralela a este proyecto, desarrollando una arquitectura de segmentación que permitiera detectar aquellas zonas de interés y utilizarlas como entrada al algoritmo propuesto. Asimismo, podrían aplicarse otro tipo de técnicas como *Multiple Instance Learning* o técnicas de detección de objetos como *You Only Look Once (YOLO).*

6.1 Planificación final del proyecto

Al finalizar el proyecto, se han realizado un conjunto de modificaciones con respecto a la planificación inicial de este. En primer lugar, la búsqueda de información del estado del arte a partir de la lectura de artículos académicos ha estado presente durante todo el proyecto. En este aspecto, esta tarea perteneciente al *sprint* 2 se ha repetido durante los *sprints* posteriores.

En segundo lugar, tanto el procesado como la obtención de una única base de datos con mamografías, ha resultado ser más complicada de lo que se esperaba, produciendo retrasos en su implementación. Este hecho ha producido que la tarea 3.3 se demorase hasta el 19 de diciembre.

Finalmente, limitaciones técnicas como la capacidad de memoria del ordenador o la tarjeta gráfica utilizada para entrenar los algoritmos de *Deep Learning* propuestos, han incrementado el tiempo necesario para la realización de las tareas 3.6 y 3.7. Este hecho, de nuevo, ha producido que las tareas 3.8, 4.1 y 4.2 se retrasaran produciendo, además, una reducción de los plazos necesarios para la realización de las tareas 4.1 y 4.2.A continuación, se detalla el diagrama de Gantt del proyecto modificado.

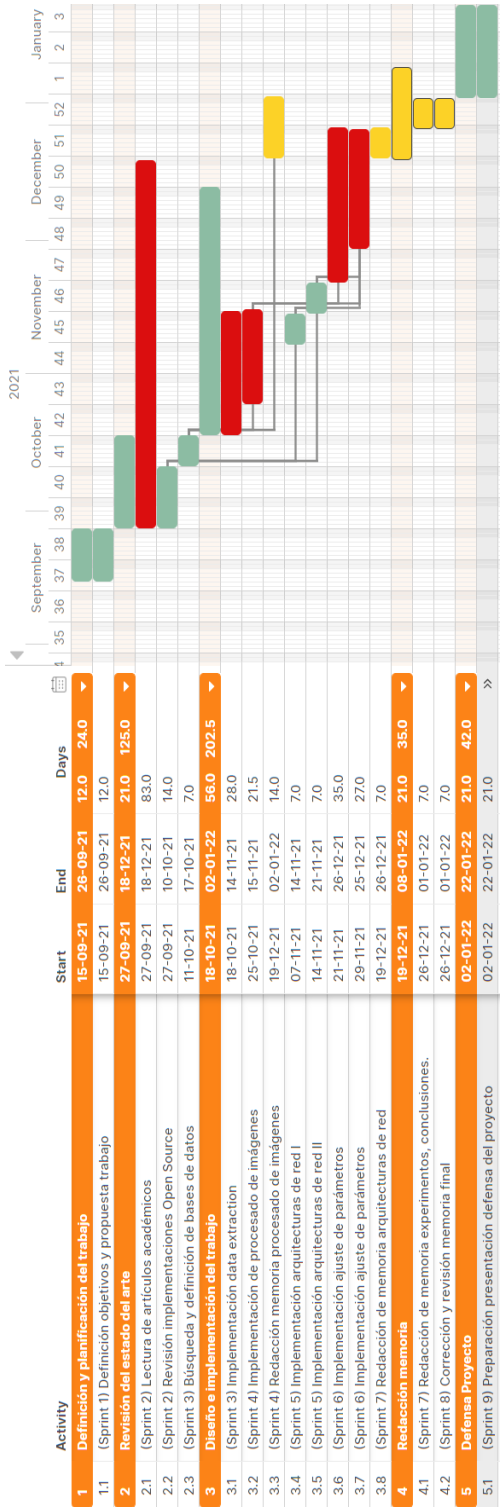


Figura 52. *Diagrama de Gantt con las fases del proyecto modificadas durante la realización del mismo. Las zonas en rojo indican demoras en la realización de las tareas. Las zonas en amarillo indican tareas iniciadas con retraso respeto la planificación inicial.*

7. Bibliografía

[1] "Cáncer de mama," Mar. 26, 2021. https://www.who.int/es/news-room/fact-sheets/detail/breast-cancer (accessed Sep. 23, 2021).

[2] C. P. Stewart, B. W., Wild, "World Cancer Report 2014 - WHO - OMS -," *IARC Nonserial Publication*, pp. 16–54, 2014.

[3] P. Boyle and B. Levin, "World CanCer report 2008," *Cancer Control*, vol. 199, 2008, doi: 10.1016/j.cma.2010.02.010.

[4] C. Fitzmaurice *et al.*, "Global, regional, and national cancer incidence, mortality, years of life lost, years lived with disability, and disability-adjusted life-years for 32 cancer groups, 1990 to 2015: a systematic analysis for the global burden of disease study," *JAMA oncology*, vol. 3, no. 4, pp. 524–548, 2017.

[5] The American Cancer Society, "American Cancer Society Recommendations for the Early Detection of Breast Cancer," Apr. 22, 2021. https://www.cancer.org/cancer/breast-cancer/screening-tests-and-early-detection/american-cancer-society-recommendations-for-the-early-detection-of-breast-cancer.html (accessed Sep. 23, 2021).

[6] National Breast Cancer Foundation, "Breast Cancer Diagnosis." http://www.nationalbreastcancer.org/breast-cancer-diagnosis (accessed Oct. 17, 2021).

[7] E. M. F. el Houby and N. I. R. Yassin, "Malignant and nonmalignant classification of breast lesions in mammograms using convolutional neural networks," *Biomedical Signal Processing and Control*, vol. 70, 2021, doi: 10.1016/j.bspc.2021.102954.

[8] D. Ribli, A. Horváth, Z. Unger, P. Pollner, and I. Csabai, "Detecting and classifying lesions in mammograms with Deep Learning OPEN", doi: 10.1038/s41598-018-22437-z.

[9] Educación y fomento, "El sistema universitario espanyol. Il sistema universitario spagnolo." https://www.educacionyfp.gob.es/italia/dam/jcr:b53864d2-65a3-4526-abf4-61ef02f5be34/el-sistema-universitario-espa-ol2.pdf (accessed Sep. 23, 2021).

[10] Indeed, "Desarrolla tu carrera professional." https://es.indeed.com/career/data-scientist/salaries (accessed Sep. 23, 2021).

[11] EALDE, "¿Cuál es el salario de un Project Manager en 2021?," Jan. 07, 2021. https://www.ealde.es/salario-project-manager-2021/ (accessed Dec. 19, 2021).

[12] Jetbrains, "Subscription option & pricing." https://www.jetbrains.com/pycharm/buy/#personal?billing=yearly (accessed Sep. 23, 2021).

[13] "MSI-Specifications." https://es.msi.com/Laptop/GV62-7RD/Specification (accessed Dec. 29, 2021).

[14] Gencat, "Guies per al càlcul d'emissions de GEH." https://canviclimatic.gencat.cat/ca/actua/guia_de_calcul_demissions_de_co2/ (accessed Dec. 29, 2021).

[15] American Cancer society, "What does the doctor look for on a Mammogram?," Oct. 03, 2019. https://www.cancer.org/es/cancer/cancer-de-seno/pruebas-de-deteccion-y-deteccion-temprana-del-cancer-de-

seno/mamogramas/que-busca-el-medico-en-un-mamograma.html (accessed Sep. 23, 2021).
[16] S. Sasikala, M. Bharathi, M. Ezhilarasi, M. Ramasubba Reddy, and S. Arunkumar, “Fusion of MLO and CC View Binary Patterns to Improve the Performance of Breast Cancer Diagnosis,” *Current Medical Imaging Reviews*, vol. 14, no. 4, 2018, doi: 10.2174/1573405614666180104162408.
[17] K. Kerlikowske *et al.*, “Performance of screening mammography among women with and without a first-degree relative with breast cancer,” *Annals of Internal Medicine*, vol. 133, no. 11, 2000, doi: 10.7326/0003-4819-133-11-200012050-00009.
[18] L. Berlin, “Radiologic errors, past, present and future,” *Diagnosis*, vol. 1, no. 1, 2014, doi: 10.1515/dx-2013-0012.
[19] M. F. Akay, “Support vector machines combined with feature selection for breast cancer diagnosis,” *Expert Systems with Applications*, vol. 36, no. 2 PART 2, 2009, doi: 10.1016/j.eswa.2008.01.009.
[20] J. R. Quinlan, “Improved use of continuous attributes in C4.5,” *Journal of Artificial Intelligence Research*, vol. 4, 1996, doi: 10.1613/jair.279.
[21] J. Abonyi and F. Szeifert, “Supervised fuzzy clustering for the identification of fuzzy classifiers,” *Pattern Recognition Letters*, vol. 24, no. 14, 2003, doi: 10.1016/S0167-8655(03)00047-3.
[22] B. Zheng, S. W. Yoon, and S. S. Lam, “Breast cancer diagnosis based on feature extraction using a hybrid of K-means and support vector machine algorithms,” *Expert Systems with Applications*, vol. 41, no. 4 PART 1, 2014, doi: 10.1016/j.eswa.2013.08.044.
[23] H. L. Chen, B. Yang, J. Liu, and D. Y. Liu, “A support vector machine classifier with rough set-based feature selection for breast cancer diagnosis,” *Expert Systems with Applications*, vol. 38, no. 7, 2011, doi: 10.1016/j.eswa.2011.01.120.
[24] D. Dua and C. Graff, “UCI Machine Learning Repository.” 2017. [Online]. Available: http://archive.ics.uci.edu/ml
[25] T. Kooi *et al.*, “Large scale deep learning for computer aided detection of mammographic lesions,” *Medical Image Analysis*, vol. 35, 2017, doi: 10.1016/j.media.2016.07.007.
[26] M. R. Al-Hadidi, A. Alarabeyyat, and M. Alhanahnah, “Breast Cancer Detection Using K-Nearest Neighbor Machine Learning Algorithm,” 2017. doi: 10.1109/DeSE.2016.8.
[27] D. Dutta and D. Chakraborty, “A deep convolutional neural network based framework for breast cancer detection,” 2020. doi: 10.1109/WIECON-ECE52138.2020.9398008.
[28] M. S. Darweesh *et al.*, “Early breast cancer diagnostics based on hierarchical machine learning classification for mammography images,” *Cogent Engineering*, vol. 8, no. 1, p. 1968324, 2021.
[29] J. Arevalo, F. A. González, R. Ramos-Pollán, J. L. Oliveira, and M. A. Guevara Lopez, “Representation learning for mammography mass lesion classification with convolutional neural networks,” *Computer Methods and Programs in Biomedicine*, vol. 127, 2016, doi: 10.1016/j.cmpb.2015.12.014.
[30] G. Carneiro, J. Nascimento, and A. P. Bradley, “Unregistered multiview mammogram analysis with pre-trained deep learning models,” in *Lecture*

Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2015, vol. 9351. doi: 10.1007/978-3-319-24574-4_78.

[31] W. Peng, R. v. Mayorga, and E. M. A. Hussein, “An automated confirmatory system for analysis of mammograms,” *Computer Methods and Programs in Biomedicine*, vol. 125, 2016, doi: 10.1016/j.cmpb.2015.09.019.

[32] B. Sahiner *et al.*, “Classification of mass and normal breast tissue: A convolution neural network classifier with spatial domain and texture images,” *IEEE Transactions on Medical Imaging*, vol. 15, no. 5, 1996, doi: 10.1109/42.538937.

[33] H. Li, S. Zhuang, D. ao Li, J. Zhao, and Y. Ma, “Benign and malignant classification of mammogram images based on deep learning,” *Biomedical Signal Processing and Control*, vol. 51, 2019, doi: 10.1016/j.bspc.2019.02.017.

[34] L. Shen, L. R. Margolies, J. H. Rothstein, E. Fluder, R. McBride, and W. Sieh, “Deep Learning to Improve Breast Cancer Detection on Screening Mammography,” *Scientific Reports*, vol. 9, no. 1, 2019, doi: 10.1038/s41598-019-48995-4.

[35] S. A. Agnes, J. Anitha, S. I. A. Pandian, and J. D. Peter, “Classification of Mammogram Images Using Multiscale all Convolutional Neural Network (MA-CNN),” *Journal of Medical Systems*, vol. 44, no. 1, 2020, doi: 10.1007/s10916-019-1494-z.

[36] W. M. Salama and M. H. Aly, “Deep learning in mammography images segmentation and classification: Automated CNN approach,” *Alexandria Engineering Journal*, vol. 60, no. 5, 2021, doi: 10.1016/j.aej.2021.03.048.

[37] H. Chougrad, H. Zouaki, and O. Alheyane, “Deep Convolutional Neural Networks for breast cancer screening,” *Computer Methods and Programs in Biomedicine*, vol. 157, 2018, doi: 10.1016/j.cmpb.2018.01.011.

[38] J Suckling, *he Mammographic Image Analysis Society Digital Mammogram Database*, vol. 1069. International Congress Series, 1994.

[39] R. S. Lee, F. Gimenez, A. Hoogi, K. K. Miyake, M. Gorovoy, and D. L. Rubin, “A curated mammography data set for use in computer-aided detection and diagnosis research,” *Scientific Data*, vol. 4, no. 1, p. 170177, Dec. 2017, doi: 10.1038/sdata.2017.177.

[40] “Digital Database for Screening Mammography. Overview of Volume: benign_without_callback_01.” http://www.eng.usf.edu/cvprg/mammography/DDSM/thumbnails/benign_without_callbacks/benign_without_callback_01/overview.html (accessed Dec. 24, 2021).

[41] I. C. Moreira, I. Amaral, I. Domingues, A. Cardoso, M. J. Cardoso, and J. S. Cardoso, “INbreast,” *Academic Radiology*, vol. 19, no. 2, pp. 236–248, Feb. 2012, doi: 10.1016/j.acra.2011.09.014.

[42] K. Simonyan and A. Zisserman, “VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION,” 2015. [Online]. Available: http://www.robots.ox.ac.uk/

[43] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition.” [Online]. Available: http://image-net.org/challenges/LSVRC/2015/

[44] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, “Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning.”

[45] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 2261–2269. doi: 10.1109/CVPR.2017.243.
[46] A. Bosch, J. Casas, and T. Lozano, “Residual networks (ResNet),” in *Deep Learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 150–151.
[47] C. Szegedy, V. Vanhoucke, S. Ioffe, and J. Shlens, “Rethinking the Inception Architecture for Computer Vision.”
[48] A. Bosch, J. Casas, and T. Lozano, “Inception,” in *Deep learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 152–156.
[49] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks.” [Online]. Available: https://github.com/liuzhuang13/DenseNet.
[50] L. E. M. Duijm, J. H. Groenewoud, J. Fracheboud, and H. J. de Koning, “Additional Double Reading of Screening Mammograms by Radiologic Technologists: Impact on Screening Performance Parameters,” *JNCI: Journal of the National Cancer Institute*, vol. 99, no. 15, pp. 1162–1170, Aug. 2007, doi: 10.1093/jnci/djm050.
[51] N. Karssemeijer, J. D. Otten, A. A. J. Roelofs, S. van Woudenberg, and J. H. C. L. Hendriks, “Effect of independent multiple reading of mammograms on detection performance,” May 2004, p. 82. doi: 10.1117/12.535225.
[52] S. Taylor-Phillips and C. Stinton, “Double reading in breast cancer screening: considerations for policy-making,” 2020.
[53] J. Gironés, J. Casas, J. Minguillón, and R. Casihuelas, “Combinación de clasificadores,” in *Minería de datos: Modelos y algoritmos*, 1ª., dâctilos, Ed. Barcelona: Oberta UOC Publishing, SL, 2017, pp. 251–252.
[54] J. Gironés, J. Casas, J. Minguillón, and R. Casihuelas, “Combinación secuencial de clasificadores base diferentes,” in *Minería de datos: Modelos y algoritmos*, 1ª., dâctilos, Ed. Barcelona: Oberta UOC Publishing, SL, 2017, pp. 258–261.
[55] J. Gironés, J. Casas, J. Minguillón, and R. Casihuelas, “Random forests,” in *Minería de datos: Modelos y algoritmos*, 1ª., dâctilos, Ed. Barcelona: Oberta UOC Publishing, SL, 2017, pp. 255–256.
[56] T. Mahmood, J. Li, Y. Pei, and F. Akhtar, “An Automated In-Depth Feature Learning Algorithm for Breast Abnormality Prognosis and Robust Characterization from Mammography Images Using Deep Transfer Learning,” *Biology*, vol. 10, no. 9, p. 859, Sep. 2021, doi: 10.3390/biology10090859.
[57] A. Bosch, J. Casas, and T. Lozano, “Data Augmentation,” in *Deep learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 164–169.
[58] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, “Albumentations: Fast and Flexible Image Augmentations,” *Information*, vol. 11, no. 2, p. 125, Feb. 2020, doi: 10.3390/info11020125.
[59] J. Gironés, J. Casas, J. Minguillón, and R. Casihuelas, “Data augmentation,” *Minería de datos: Modelos y algoritmos*. pp. 164–169, 2017.

[60] A. Bosch, J. Casas, and T. Lozano, “Softmax,” in *Deep learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 84–85.
[61] A. Bosch, J. Casas, and T. Lozano, “Capa ReLU (rectified linear units),” in *Deep learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 134–135.
[62] Christian Versloot, “What are L1, L2 and Elastic Net Regularization in neural networks?,” *Machine Curve*, Jan. 21, 2020. https://www.machinecurve.com/index.php/2020/01/21/what-are-l1-l2-and-elastic-net-regularization-in-neural-networks/ (accessed Dec. 29, 2021).
[63] A. Bosch, J. Casas, and T. Lozano, “Regularización L2,” in *Deep learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 93–95.
[64] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, Dec. 2014.
[65] A. Bosch, J. Casas, and T. Lozano, “Dropout,” in *Deep learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 96–97.
[66] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, “Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1717–1724. doi: 10.1109/CVPR.2014.222.
[67] H.-C. Shin *et al.*, “Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning,” *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, May 2016, doi: 10.1109/TMI.2016.2528162.
[68] A. Bosch, J. Casas, and T. Lozano, “Transfer Learning,” in *Deep learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 160–164.
[69] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How transferable are features in deep neural networks?” 2014.
[70] A. Bosch, J. Casas, and T. Lozano, “Early stopping,” in *Deep learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 95–96.
[71] N. Qian, “On the momentum term in gradient descent learning algorithms,” *Neural Networks*, vol. 12, no. 1, pp. 145–151, Jan. 1999, doi: 10.1016/S0893-6080(98)00116-6.
[72] J. Duchi, E. Hazan, and Y. Singer, “Adaptive Subgradient Methods for Online Learning and Stochastic Optimization.” pp. 2121–2159, 2010.
[73] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” Dec. 2014.
[74] A. Bosch, J. Casas, and T. Lozano, “Algoritmos de entrenamiento,” in *Deep learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 85–87.
[75] A. Bosch, J. Casas, and T. Lozano, “Épocas, iteraciones y batch,” in *Deep learning. Principios y fundamentos*, 1ª., Editorial UOC and S. L. Reverté-Aguilar, Eds. Barcelona, 2019, pp. 82–84.

[76] J. Brodersen and V. D. Siersma, “Long-Term Psychosocial Consequences of False-Positive Screening Mammography,” *The Annals of Family Medicine*, vol. 11, no. 2, pp. 106–115, Mar. 2013, doi: 10.1370/afm.1466.
[77] H.-C. Shin *et al.*, “Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning,” *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, May 2016, doi: 10.1109/TMI.2016.2528162.
[78] American cancer society, “Treatment of Stage IV (Metastatic) Breast Cancer,” Oct. 27, 2021. https://www.cancer.org/cancer/breast-cancer/treatment/treatment-of-breast-cancer-by-stage/treatment-of-stage-iv-advanced-breast-cancer.html#written_by (accessed Dec. 30, 2021).
[79] C. M. Florkowski, “Sensitivity, specificity, receiver-operating characteristic (ROC) curves and likelihood ratios: communicating the performance of diagnostic tests.,” *The Clinical biochemist. Reviews*, vol. 29 Suppl 1, pp. S83-7, Aug. 2008.
[80] D. Berrar, “Performance Measures for Binary Classification,” in *Encyclopedia of Bioinformatics and Computational Biology*, Elsevier, 2019, pp. 546–560. doi: 10.1016/B978-0-12-809633-8.20351-8.
[81] R. M. Rangayyan, T. M. Nguyen, F. J. Ayres, and A. K. Nandi, “Effect of Pixel Resolution on Texture Features of Breast Masses in Mammograms”, doi: 10.1007/s10278-009-9238-0.
[82] “Understanding Breast Calcifications,” Oct. 30, 2020. https://www.breastcancer.org/symptoms/testing/types/mammograms/mamm_show/calcifications (accessed Dec. 28, 2021).