

PROYECTO FIN DE GRADO

TÍTULO: Detección de melanoma en imágenes dermatoscópicas mediante el uso de información contextual y Redes Neuronales Convolucionales.

AUTORA: Brenda Fabiana Farinha Fernandes

TITULACIÓN: Grado en Ingeniería Electrónica de Comunicaciones

TUTORA: Juana María Gutiérrez Arriola

DEPARTAMENTO: Ingeniería Telemática y Electrónica

VºBº TUTOR/A

Miembros del Tribunal Calificador:

PRESIDENTE: Waldo Saúl Pérez Aguiar

TUTORA: Juana María Gutiérrez Arriola

SECRETARIO: Fernando Pescador del Oso

Fecha de lectura:

Calificación:

El Secretario/La Secretaria,



Agradecimientos

La presentación de este proyecto significa el final de una etapa en mi vida y no puedo dejar de expresar mi gratitud a todas aquellas personas que estuvieron presentes en ella. Fue un camino largo y difícil, y sin ustedes nada hubiese sido igual.

Soy muy afortunada de tenerlos en mi vida.

¡GRACIAS!

A mis padres,

A mis hermanos,

A mi madrina Jenny,

A Santiago de Compostela,

A mi tutora Juana Gutiérrez,

A la Universidad Simón Bolívar,

A la Universidad Politécnica de Madrid,

A Vanesa, José Arnaldo, Luisana, Leonardo, Kheyter, Nicolás, Manuel, Oqueber, Kelly, Sophia, Elisa, Adrián, David, Javier, Stalin, Diego, Katherine, Franz, Carlos Luis, Osmary, Rebeca, Alfonso y Jetsel; quienes vivieron junto a mi esta etapa de migración y cambios, sin duda alguna su amistad, apoyo y empatía han sido

importantes para alcanzar esta meta.

A María Gabriela y Manuel Alejandro; grandes amigos que no se encuentran tan cerca físicamente, pero no han dejado de estar presentes en cada etapa.

A mis amigos de la Universidad Simón Bolívar, los recuerdo con inmenso cariño:

Zadkiel, Jeckson, Leonardo, Jaime, Yura, Santiago, Patricia, Marco, Alejandro,

Nicola, Raynulfo, José, Eladio, Enzo, David, Noya, José Angel, Efrain, Isarai, Ananias, Asdrubal, Fernando, Anggie, Mario, Maryhec, César, Luz, Lisbeth, Román,

Daniel, Rafa, Verónica, Lisette, Mafer, Andrea, Oriana, Augusto y Miguel.

A mis compañeros y amigos del trabajo; Pablo, Iván, Raúl, Antonio y Nacho.

A todos, ¡GRACIAS INFINITAS!

Resumen

La detección temprana del melanoma, considerado como el tipo de cáncer de piel más agresivo, es de vital importancia para evitar su rápida propagación y disminuir las tasas de mortalidad en etapas avanzadas. Sin embargo, su diagnóstico supone importantes desafíos debido a la diversidad en las características de las lesiones y la poca accesibilidad a médicos expertos en el área de la dermatología.

El método de detección más empleado por médicos especializados es la dermatoscopia, un método no invasivo que permite visualizar imágenes aumentadas de las lesiones de piel en busca de potenciales patrones característicos de una lesión maligna. Sin embargo, este sistema puede derivar en falsas interpretaciones, además de no ser una herramienta accesible a todos los servicios médicos.

Ante este escenario, surge la necesidad de desarrollar nuevos métodos de diagnóstico efectivos y precisos, por lo que en el presente Proyecto Fin de Grado se desarrolla un sistema de clasificación de imágenes basado en aprendizaje profundo capaz de generar un diagnóstico de melanoma sobre imágenes dermatoscópicas, tomando en cuenta además, información contextual del paciente como la parte del cuerpo donde se aloja la lesión, el sexo y la edad; siendo estos parámetros tomados en cuenta por dermatólogos al momento de dar un diagnóstico.

El modelo presentado ha sido entrenado, validado y evaluado empleando los conjuntos de datos ofrecidos por la *Society for Imaging Informatics in Medicine* (SIIM) en colaboración con el *International Skin Imaging Collaboration* (ISIC) para su desafío propuesto para los años 2019 y 2020. Se ha desarrollado un sistema dividido en tres etapas o módulos: evaluando en el primer módulo las imágenes dermatoscópicas, en el segundo los datos contextuales del paciente y en el tercero el ensamblado de ambos módulos, donde se realiza una ponderación de predicciones por el método de *Soft Voting*. Con este sistema se consigue realizar un diagnóstico con una exhaustividad del 75,62% y una precisión del 56,75%.

Abstract

Early detection of melanoma, considered the most aggressive type of skin cancer, is of vital importance to prevent its rapid spread and reduce mortality rates in advanced stages. However, its diagnosis poses significant challenges due to the diversity in the characteristics of the lesions and the low accessibility to expert physicians in the field of dermatology.

The detection method most used by specialized doctors is dermoscopy, a non-invasive method that allows viewing magnified images of skin lesions in search of potential characteristic patterns of a malignant lesion. However, this system can lead to false interpretations, in addition to not being a tool accessible to all medical services.

Given this scenario, the need arises to develop new effective and precise diagnostic methods, so in this Final Degree Project an image classification system based on deep learning capable of generating a diagnosis of melanoma on dermoscopic images is developed. also taking into account the contextual information of the patient such as the part of the body where the lesion is located, the sex and the age of the patient; These parameters are taken into account by dermatologists when making a diagnosis.

The presented model has been trained, validated and evaluated using the data sets offered by the *Society for Imaging Informatics in Medicine* (SIIM) in collaboration with the *International Skin Imaging Collaboration* (ISIC) for their proposed challenge for the year 2019 and 2020. A system divided into three stages or module was developed, evaluating in the first module the dermoscopic images, in the second the contextual data of the patient and in the third the assembly of both modules, where a weighting of predictions is carried out after the method of `textitSoft Voting`. With this system it is possible to carry out a diagnosis with a completeness of 75.62% and a precision of 56.75%.

Índice general

Agradecimientos	I
Resumen	III
Abstract	V
Lista de acrónimos	XV
1. Introducción	1
1.1. Motivación	1
1.2. Objetivos	3
1.3. Estructura del documento	3
2. Marco tecnológico	5
2.1. Redes Neuronales Convolucionales	5
2.1.1. Capas convolucionales	7
2.1.2. Capas de agrupación (<i>pooling layers</i>)	8
2.1.3. Capa totalmente conectada (<i>fully connected layers</i>)	9
2.2. Función de activación	9
2.3. Métricas de evaluación del modelo	11
2.4. Arquitectura EfficientNet	13
3. Especificaciones y restricciones de diseño	17
3.1. Especificaciones de diseño	17

3.2. Restricciones del diseño	18
4. Descripción de la solución propuesta	19
4.1. Configuración del entorno de desarrollo	19
4.1.1. Google Colaboratory	19
4.1.2. Google Cloud Storage	20
4.1.3. Unidades de procesamiento tensorial (TPU)	20
4.1.4. TensorFlow	21
4.1.5. Formato TFRecord	22
4.2. Conjunto de datos	22
4.2.1. Análisis exploratorio de los datos	25
4.3. Modelo de la solución propuesta	31
4.3.1. Módulo de clasificación de melanoma en imágenes dermatoscópicas	31
4.3.2. Módulo de clasificación de melanoma con metadatos	38
4.3.3. Ensamblado final	40
5. Análisis de resultados	43
5.1. Conjunto de datos de validación	43
5.2. Conjunto de datos de evaluación	43
5.3. Métricas de evaluación	44
5.4. Evaluación del módulo de clasificación de imágenes	44
5.4.1. Evaluación del módulo de clasificación de metadatos	47
5.4.2. Evaluación del módulo final	47
6. Presupuesto	51
7. Impacto del proyecto	53
8. Conclusiones	55

<i>ÍNDICE GENERAL</i>	IX
-----------------------	----

8.0.1. Conclusiones generales	55
---	----

Bibliografía	57
---------------------	-----------

Índice de figuras

2.1. Esquema típico de una red neuronal profunda [7]	6
2.2. Esquema de una capa convolucional [8]	7
2.3. Esquema de una capa convolucional para una imagen de entrada a color [8]	8
2.4. Función de activación sigmoide [9]	10
2.5. Función de activación tangente hiperbólica (<i>tanh</i>) [9]	10
2.6. Función de activación ReLU [9]	11
2.7. Matriz de confusión	12
2.8. Rendimiento de arquitecturas de CNN [10]	14
2.9. Escala del modelo de CNN en la Arquitectura EfficientNet [10]	14
4.1. Imágenes dermatoscópicas con diagnóstico positivo de melanoma	24
4.2. Imágenes dermatoscópicas con diagnóstico negativo de melanoma	25
4.3. Diagrama general del sistema propuesto	31
4.4. Data augmentation: rotación horizontal	33
4.5. Data augmentation: rotación vertical	33
4.6. Data augmentation: ajuste del tono de las imágenes	34
4.7. Data augmentation: ajuste de la saturación de las imágenes	34
4.8. Data augmentation: ajuste de la saturación de las imágenes	34
4.9. Data augmentation: ajuste de la saturación de las imágenes	35
4.10. Modelo de CNN aplicado a la clasificación de imágenes	37
4.11. Modelo de Red Neuronal aplicado a la clasificación de metadatos	40

4.12. Diagrama general del sistema propuesto 41

Índice de tablas

4.1. Conjunto total de datos	23
4.2. Distribución real del conjunto de datos en formato TFRecord	24
4.3. Distribución de datos según su diagnóstico	25
4.4. Distribución del dato 'sex' en los conjuntos de datos	26
4.5. Distribución del dato 'age_approx' en los conjuntos de datos	27
4.6. Distribución del dato 'age_approx' en el conjunto de datos completo	28
4.7. Distribución del dato 'anatom_site_general_challenge' en el conjunto de datos	29
4.8. Distribución del dato 'anatom_site_general_challenge' en el conjunto de datos unificando las lesiones ubicadas en el torso	30
4.9. Distribución real de los datos según su diagnóstico en el conjunto de datos ofrecido en formato TFRecord	31
4.10. Distribución de los subconjuntos de datos según su diagnóstico	32
5.1. Conjunto de pruebas de entrenamiento para el módulo de clasificación de imágenes	45
5.2. Segundo conjunto de pruebas de entrenamiento para el módulo de clasificación de imágenes con la arquitectura EfficientNetB5	46
5.3. Evaluación del modelo final de clasificación ante el conjunto de datos de prueba	47
5.4. Evaluación del ensamble del modelo final de clasificación	49
6.1. Presupuesto general del proyecto	51
6.2. Amortización de los equipos informáticos	51

Lista de acrónimos

API Application Programming Interface

CNN Convolutional Neural Network

CPU Central Processing Unit

CSV Comma Separated Values

DL Deep Learning

GCS Google Cloud Storage

GPU Graphics Processing Unit

IA Inteligencia Artificial

ISIC International Skin Imaging Collaboration

ML Machine Learning

PFG Proyecto Fin de Grado

RGB Red-Green-Blue

SIIM Society for Imaging Informatics in Medicine

TPU Tensor Processing Unit

Capítulo 1

Introducción

En este capítulo se describe la motivación, alcance y objetivos principales a alcanzar tras la realización del Proyecto Fin de Grado (PFG), describiendo además la estructura del presente documento.

1.1. Motivación

El melanoma es un tipo de cáncer de piel ocasionado por la reproducción anormal de las células que producen el pigmento de la piel, llamadas melanocitos. La aparición o el cambio de la forma, color y tamaño de lunares, llagas, imperfecciones, marcas o protuberancias en cualquier área de la piel podrían ser una señal de este u otro tipo de cáncer de piel.

Desafortunadamente, tras el aumento de la exposición directa a la luz solar, el número de casos positivos de melanoma ha incrementado en las últimas décadas. Según datos publicados por la Sociedad Española de Oncología Médica [1] se estima que en el año 2023 se detecten cerca de 8.049 nuevos casos en España, posicionándose como el undécimo tipo de cáncer con mayor incidencia en el país, y siendo más frecuente su diagnóstico en hombres que en mujeres, y en personas mayores de 65 años de edad.

A nivel mundial, por su parte, se estima que en el año 2020 cerca de 324.635 personas fueron diagnosticadas de melanoma [2], representando el 1,5 % de todos los tumores detectados.

A pesar de ser uno de los tipos de cáncer menos común, es considerado como el

cáncer de piel más peligroso y agresivo dado que su propagación hacia otras partes del cuerpo puede desarrollarse rápidamente si no es detectado y tratado en una etapa temprana y oportuna.

Un estudio recientemente publicado por el *Journal of the American Academy of Dermatology* [3] resalta la importancia de una detección precoz, y revela que la tasa de supervivencia de los pacientes diagnosticados en etapa I y tratados de forma inmediata es significativamente mayor a los tratados en los siguientes 30 a 59 días desde la obtención del diagnóstico.

En este sentido, la detección temprana resulta fundamental por lo que en términos de investigación, diagnóstico y tratamiento, el campo del melanoma ha presentado avances significativos durante los últimos años.

Los métodos de detección de esta enfermedad incluyen la auto-exploración de la piel y exámenes dermatológicos acompañados en algunos casos de una biopsia de la lesión. Entre estos exámenes dermatológicos se encuentra la dermatoscopia, una técnica no invasiva que proporciona una imagen ampliada y nítida de las lesiones cutáneas mediante el uso de un instrumento óptico denominado dermatoscopio. Este método permite visualizar estructuras más profundas e identificar características propias de las lesiones que no son perceptibles a simple vista, lo que ha representado un avance importante en la práctica clínica dermatológica. Sin embargo, la interpretación de estas imágenes dermatoscópicas es un proceso manual que requiere ser realizado por dermatólogos calificados, y además de requerir tiempo, es posible que una lesión maligna pase desapercibida.

Ante este escenario, y tras el constante avance de la tecnología en el ámbito de la medicina y la visión artificial, surge la necesidad de desarrollar nuevas herramientas capaces de diagnosticar lesiones cutáneas con mayor precisión y en un periodo corto de tiempo.

Entre estas herramientas destaca la inteligencia artificial basada en modelos de aprendizaje profundo, conocido comúnmente como *deep learning* por su nombre en inglés. En particular destacan las redes neuronales convolucionales (CNN por sus siglas en inglés de *Convolutional Neural Networks*) inspiradas en la estructura y funcionamiento del cerebro humano, siendo capaces de analizar y clasificar grandes cantidades de imágenes y datos.

De esta forma, la capacidad analítica de dicha técnica en el análisis de las lesiones,

combinado con la experiencia clínica de médicos especializados, puede representar una mejora significativa en la calidad del diagnóstico precoz y la atención médica necesaria.

1.2. Objetivos

El objetivo principal de este Proyecto Fin de Grado es desarrollar un sistema de clasificación de imágenes basado en aprendizaje profundo capaz de generar un diagnóstico de melanoma sobre imágenes dermatoscópicas junto con información contextual del paciente.

Se desea analizar si parámetros como la edad, el sexo y la localización de la lesión en el cuerpo, pueden ofrecer un diagnóstico con mayor precisión, en comparación a un análisis único de las imágenes.

1.3. Estructura del documento

El presente documento está organizado en 8 capítulos. El primero de ellos presenta la introducción al proyecto describiendo los objetivos a alcanzar. En el capítulo 2 se describen los aspectos teóricos que engloban las tecnologías empleadas en el desarrollo del proyecto para la clasificación de imágenes. En el capítulo 3 se exponen las especificaciones y restricciones planteadas; mientras que en el capítulo 4 se describe de forma detallada la solución propuesta al planteamiento del proyecto, realizando, además, un análisis exhaustivo de los datos de entrada al sistema. Seguidamente, se realiza un análisis de resultados en el capítulo 5. En los siguientes dos capítulos se detallan los costes asociados al desarrollo del proyecto y el impacto que puede ofrecer a la sociedad. Finalmente, en el capítulo 8 se presentan las principales conclusiones tras la finalización del proyecto y se proponen diferentes líneas de trabajo futuras.

Capítulo 2

Marco tecnológico

En este capítulo se describen los aspectos teóricos que engloban las tecnologías estudiadas en el presente Proyecto Fin de Grado, analizando principalmente las redes neuronales convolucionales y la clasificación de imágenes mediante su implementación.

2.1. Redes Neuronales Convolucionales

Las redes neuronales convolucionales (CNN por sus siglas en inglés de *Convolutional Neural Networks*) son un tipo de red neuronal artificial inspiradas en la organización de la corteza visual del cerebro humano, que ofrecen una alta tasa de efectividad en el campo del aprendizaje profundo.

Debido a su capacidad para identificar patrones y características relevantes en imágenes, las CNN se utilizan principalmente para tareas de visión artificial, como la clasificación de imágenes, detección de objetos y la segmentación de imágenes.

La historia de las CNN se remonta a la década de 1980, cuando Yann LeCun desarrolló la primera red neuronal convolucional para identificar y reconocer caracteres escritos a mano [4]. En 1988 fue utilizada para el reconocimiento de objetos en imágenes en 3D [5]. Sin embargo, no fue hasta el año 2012 cuando se desarrolló una CNN de múltiples capas, llamada AlexNet, que revolucionó significativamente los enfoques anteriores [6].

Desde entonces, y con el aumento de la capacidad computacional y la disponibilidad de grandes conjuntos de datos, las CNN se han convertido en una tecnología

clave en la visión por computadora y se han empleado en una amplia variedad de aplicaciones, como la traducción automática, el reconocimiento facial y la detección de enfermedades a partir de imágenes médicas.

Las CNN se utilizan predominantemente para tareas basadas en el reconocimiento de imágenes, donde las imágenes de entrada deben ser consistentes en términos de resolución, formato y escala. Además, deben ser lo suficientemente grandes y con una calidad adecuada para poder ser procesadas por la red y poder aprender de ellas de manera efectiva.

En líneas generales, este tipo de redes neuronales incluye la capa de entrada, una o más capas ocultas, y una capa de salida tal y como se muestra en la Figura 2.1.

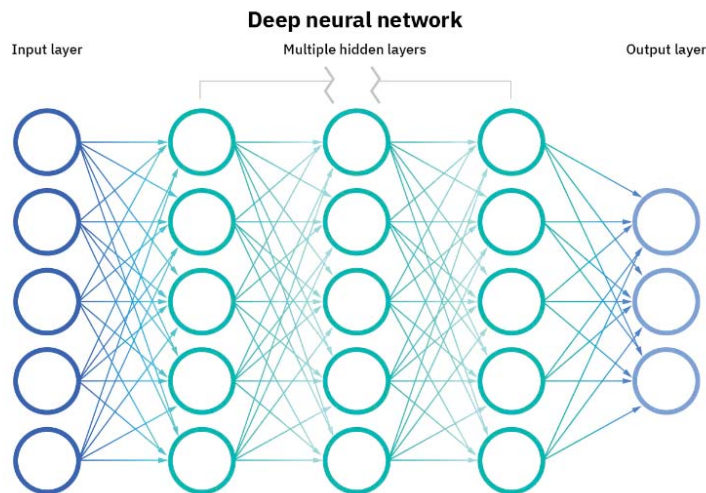


Figura 2.1: Esquema típico de una red neuronal profunda [7]

Entre las capas ocultas, las CNN se caracterizan principalmente por contener las siguientes tres:

- Capas convolucionales
- Capas de agrupación (*pooling layers*)
- Capa totalmente conectada (*fully connected layers*)

Cada una de ellas tiene un propósito distinto enfocada en diferentes aspectos del procesamiento de los datos. Con cada capa, la CNN aumenta su complejidad, identificando mayores partes de la imagen a medida que los datos avanzan a través de las capas.

2.1.1. Capas convolucionales

En el contexto de las CNN, la convolución es una operación matemática donde se aplican filtros o *kernel* a cada píxel de una imagen de entrada para producir una matriz de salida denominada *mapa de características* (Figura 2.2).

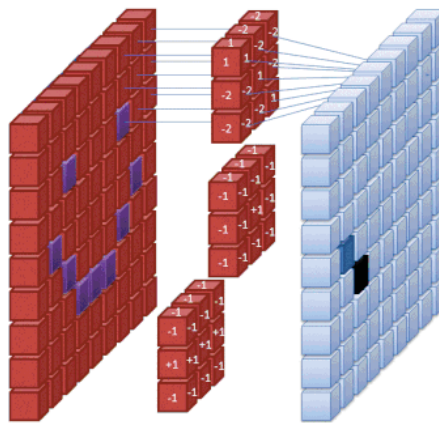


Figura 2.2: Esquema de una capa convolucional [8]

El filtro o *kernel* es una pequeña matriz numérica que se desplaza a través de la imagen de entrada. En cada posición del desplazamiento, se multiplica el valor de los píxeles de la imagen de entrada por los valores correspondientes del filtro y se suman los productos resultantes, generando mapas de características convolucionales que almacenan información relevante, como texturas, bordes o patrones. Los elementos del filtro serán los pesos entrenables de cada capa que se ajustan durante el entrenamiento de la red.

En imágenes a color, estas se representan como un tensor de tres dimensiones, donde la primera dimensión es el ancho de la imagen, la segunda es la altura de la imagen y la tercera es el número de canales de color. Durante el proceso de convolución, se aplica un filtro con una profundidad igual al número de canales de color de la imagen, generalmente tres en las representaciones de color estándar.

Una vez que se completa la convolución en cada uno de estos canales, se combinan los resultados mediante una operación de suma ponderada para obtener una única salida de la capa de convolución tal y como se observa en la Figura 2.3.

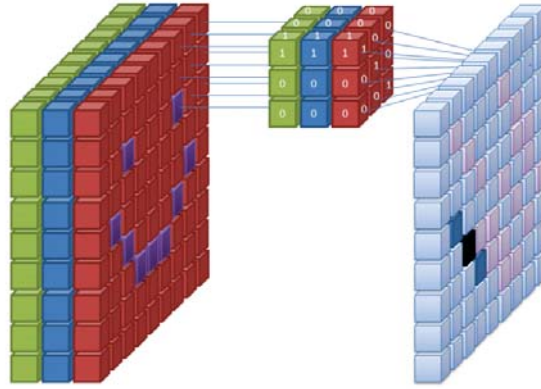


Figura 2.3: Esquema de una capa convolucional para una imagen de entrada a color [8]

Después de aplicar la operación de convolución descrita, se suele aplicar una función de activación para obtener la salida de la capa.

2.1.2. Capas de agrupación (*pooling layers*)

La capa de agrupación es responsable de reducir el tamaño espacial de la salida de la capa de convolución y reducir la cantidad de parámetros entrenables en la red, al mismo tiempo que se mantienen las características importantes de los datos de entrada. Así, su objetivo es disminuir la complejidad computacional de la red requerida para procesar los datos y acelerar a su vez el proceso de entrenamiento.

Similar a la capa convolucional, la operación de agrupación barre un filtro a través de la capa de entrada aplicando una función de agrupación. Hay dos tipos principales de agrupación:

Max pooling

A medida que el filtro se mueve a través de la entrada, calcula el valor máximo dentro del campo receptivo para enviarlo a la matriz de salida.

Average pooling

A medida que el filtro se mueve a través de la entrada, calcula el valor promedio dentro del campo receptivo para enviarlo a la matriz de salida.

Dependiendo de la complejidad de las imágenes, el número de las capas de convolución y de agrupación pueden aumentarse para capturar aún más detalles de bajo nivel. Sin embargo, esto implicaría una mayor potencia computacional.

2.1.3. Capa totalmente conectada (*fully connected layers*)

La capa totalmente conectada, llamada también *fully connected layer*, es utilizada para tomar las características extraídas por las capas convolucionales y de agrupación, y procesarlas para la clasificación empleando una función de activación. Estas salidas se combinan para formar un vector de características que se utiliza para clasificar la imagen en una o varias categorías.

2.2. Función de activación

Las funciones de activación son funciones matemáticas aplicadas a la salida de una red neuronal o capa en una red neuronal, modificando el valor resultante o imponiendo un umbral que se debe sobrepasar para poder realizar una predicción o para poder proseguir a la siguiente capa. Dado que el objetivo es que la red neuronal sea capaz de resolver problemas cada vez más complejos, las funciones de activación generalmente harán que los modelos sean no lineales.

Su elección depende del tipo de problema de aprendizaje supervisado a resolver: clasificación, regresión, segmentación, entre otros. Las más empleadas en modelos de clasificación binaria, objetivo del presente proyecto, son las siguientes:

Función de activación *sigmoide*

Función matemática definida por la Ecuación 2.1. Es usada como capa final en un problema de clasificación binaria generando valores en el rango de 0 a 1. Puede interpretarse como la probabilidad de que la entrada pertenezca a la clase positiva.

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (2.1)$$

Gráficamente se representa como la Figura 2.4.

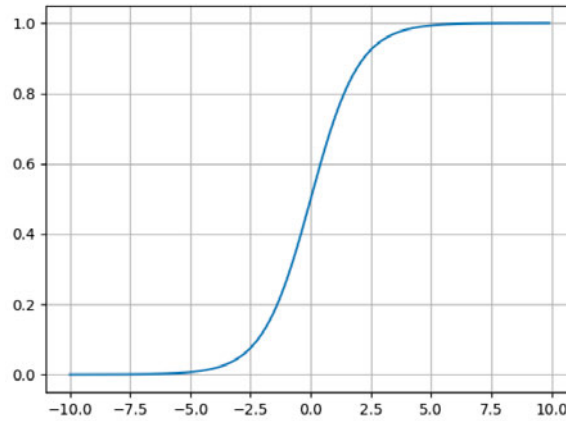


Figura 2.4: Función de activación sigmoide [9]

Función de activación tangente hiperbólica (*tanh*)

Función similar a la *sigmoide* con la diferencia de que los valores de salida están en el rango de -1 a 1 (Ecuación 2.2 y Figura 2.5).

$$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \quad (2.2)$$

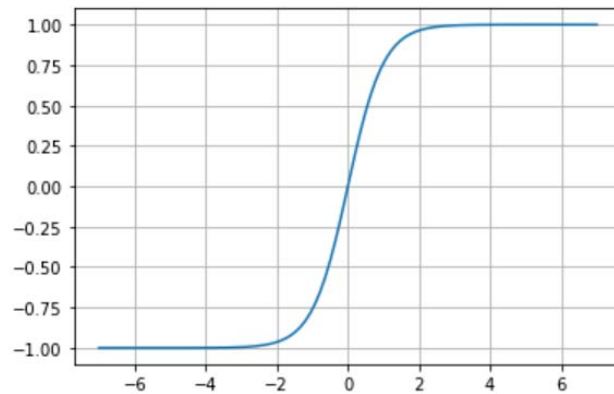


Figura 2.5: Función de activación tangente hiperbólica (*tanh*) [9]

Función de activación ReLU (*Rectified Linear Unit*)

La función de activación ReLU es una función no lineal que produce una salida de cero para todas las entradas negativas y una salida lineal para todas las entradas

positivas. Matemáticamente se define con la Ecuación 2.3 y su curva se representa en la Figura 2.6.

$$\text{ReLU}(z) = \max(0, x) \quad (2.3)$$

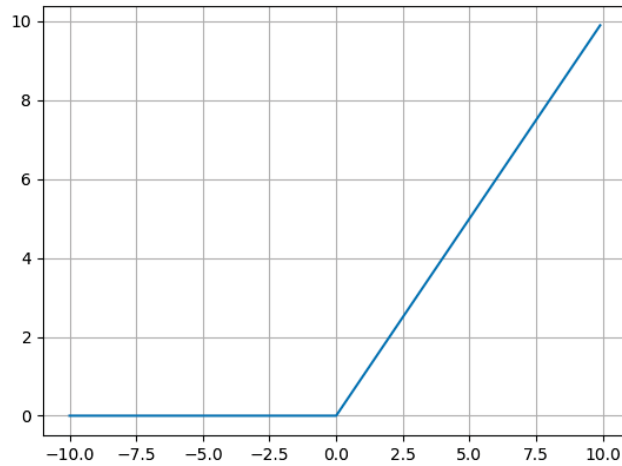


Figura 2.6: Función de activación ReLU [9]

2.3. Métricas de evaluación del modelo

Para valorar el rendimiento de un clasificador se definen las siguientes métricas:

Matriz de confusión

Una matriz de confusión es una técnica que resume el rendimiento de un modelo de clasificación y determina dónde se producen errores en la clasificación.

Cada celda de la matriz indica la cantidad de veces que el modelo clasifica de forma correcta o incorrecta cada clase. Siendo que cada fila representa las clases reales, y las columnas las clases predichas por el modelo.

En la Figura 2.7 se ilustra una matriz de confusión, donde cada métrica permite evaluar el comportamiento del modelo entrenado:

- Verdaderos Positivos (VP): El valor real es positivo, y el modelo predice un positivo.

- Verdaderos Negativos (VN): El valor real es negativo, y el modelo predice un negativo.
- Falsos Positivos (FP): El valor real es negativo, y el modelo predice un positivo.
- Falsos Negativos (FN): El valor real es positivo, y el modelo predice un negativo.

VALORES PREDICCIÓN	VP VERDADEROS POSITIVOS	FP FALSOS POSITIVOS
	FALSOS NEGATIVOS FN	VERDADEROS NEGATIVOS VN
	VALORES REALES	

Figura 2.7: Matriz de confusión

De esta forma, los verdaderos negativos y los verdaderos positivos indican una clasificación correcta. Mientras que los falsos negativos y falsos positivos indican una clasificación incorrecta.

Precisión

La precisión mide la calidad del modelo de aprendizaje automático en tareas de clasificación, respondiendo a la siguiente pregunta: ¿Qué porcentaje de diagnósticos positivos son correctos?

Matemáticamente se representa como la Ecuación 2.4.

$$precision = \frac{VP}{VP + FP} \quad (2.4)$$

Exhaustividad o *recall*

La exhaustividad, conocida comúnmente como *recall* por su nombre en inglés, mide la capacidad del modelo para identificar correctamente los casos positivos dentro de

una clase. Responde a la pregunta ¿qué porcentaje de diagnósticos positivos es capaz de detectar el modelo?

Un alto porcentaje de exhaustividad es fundamental para garantizar que no se pasen por alto casos positivos.

Matemáticamente se representa como la Ecuación 2.5.

$$recall = \frac{VP}{VP + FN} \quad (2.5)$$

F1-Score

Esta métrica combina las medidas de precisión y exhaustividad. Matemáticamente se representa como la Ecuación 2.6.

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \quad (2.6)$$

2.4. Arquitectura EfficientNet

En los últimos años ha habido un crecimiento significativo en el desarrollo de modelos de redes neuronales convolucionales que han ofrecido sistemas con mejores rendimientos y precisión en la clasificación de imágenes.

En 2019, Google Research propuso una nueva arquitectura de redes neuronales convolucionales, llamada EfficientNet [10], que logra mejorar la eficiencia de los modelos reduciendo el número de parámetros y la cantidad de operaciones por segundo que requiere.

Su rendimiento se puede apreciar en la Figura 2.8, donde se compara con respecto a otras arquitecturas, evaluando su desempeño tras ser entrenada con la base de datos ImageNet; base de datos de más de 14 millones de imágenes de objetos etiquetadas en más de 20,000 categorías diferentes, que se utiliza para entrenar y evaluar modelos de redes neuronales convolucionales.

Esta arquitectura destaca por utilizar un algoritmo de búsqueda para encontrar la mejor combinación de valores de las tres dimensiones que componen una CNN: ancho

(*width*), profundidad (*depth*) y resolución de entrada (*input resolution*). Este enfoque se basa en la idea de que el aumento simultáneo de las tres dimensiones de la red puede mejorar el rendimiento del modelo sin necesidad de aumentar la complejidad computacional, tal como se observa en la Figura 2.9

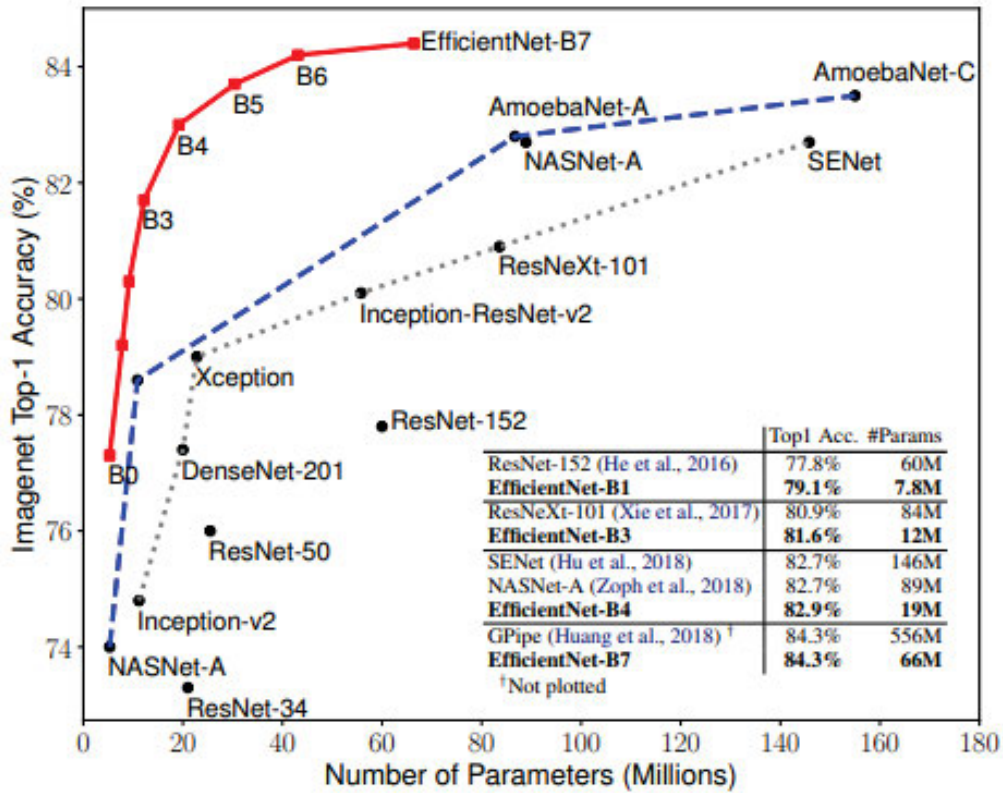
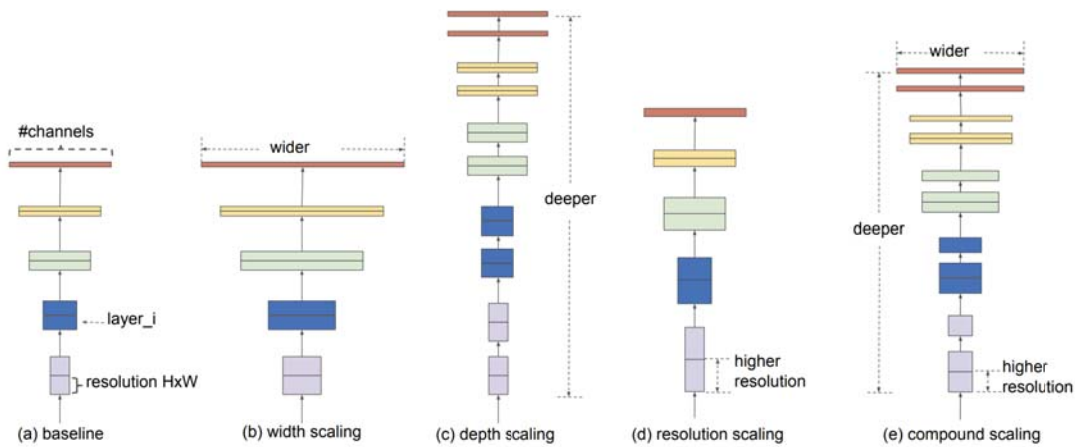


Figura 2.8: Rendimiento de arquitecturas de CNN [10]



Model Scaling. (a) is a baseline network example; (b)-(d) are conventional scaling that only increases one dimension of network width, depth, or resolution. (e) is our proposed compound scaling method that uniformly scales all three dimensions with a fixed ratio.

Figura 2.9: Escala del modelo de CNN en la Arquitectura EfficientNet [10]

Otras arquitecturas aumentan la profundidad de la red, lo que conlleva más capas de neuronas en la red y por ende mayor uso de recursos computacionales; mientras que otras aumentan la resolución de entrada que permite detectar mayor número de patrones al inicio del entrenamiento.

Capítulo 3

Especificaciones y restricciones de diseño

En este capítulo se listan las especificaciones y restricciones que definen y limitan el sistema diseñado.

3.1. Especificaciones de diseño

- El sistema asignará un diagnóstico (benigno o maligno) a cada imagen y datos de entrada.
- Las imágenes y la información contextual del paciente se tomarán de la base de datos ofrecidos por la *International Skin Imaging Collaboration* (ISIC) y *Society for Imaging Informatics in Medicine* (SIIM).
- Los datos contextuales del paciente empleados son: edad, sexo y lugar anatómico donde se aloja la lesión.
- El sistema se basará en la teoría de Redes Neuronales Convolucionales para la clasificación de las imágenes.
- El sistema se desarrollará en lenguaje de programación Python.
- Se empleará la herramienta de desarrollo Google Collaboratory.
- El sistema no examinará imágenes que no contengan una lesión de piel.

3.2. Restricciones del diseño

- Cantidad de datos de entrenamiento limitado.
- Desbalance significativo del conjunto de datos, donde las imágenes con melano-
ma representa a penas una décima parte del total de imágenes.
- Capacidad de computo y memoria RAM de las TPU ofrecidas por Google Co-
laboratory.

Capítulo 4

Descripción de la solución propuesta

En este capítulo se describe la solución implementada partiendo de la configuración del entorno de desarrollo y la exploración del conjunto de datos, siguiendo de la descripción de las distintas fases de preprocesamiento y procesamiento de los datos que componen el sistema desarrollado.

4.1. Configuración del entorno de desarrollo

Para el diseño de este proyecto se emplea la versión gratuita de Google Colaboratory, la Unidad de Procesamiento Tensorial (TPU) como recurso informático ofrecido por Google, y se almacena el conjunto de datos en formato *TensorFlow Record* (TFRecord) en Google Cloud Storage (GCS). Por otro lado, se hace uso del lenguaje Python y las librerías TensorFlow y Keras disponibles.

A continuación se brindan los aspectos más destacables de cada uno de estos recursos.

4.1.1. Google Colaboratory

Google Colaboratory, conocido también como Google Colab, es un servicio gratuito en la nube creado por Google Research que proporciona un entorno de desarrollo Jupyter Notebook compatible con el lenguaje de programación Python. Permite a

los usuarios desarrollar tareas basadas en aprendizaje automático y análisis de datos desde un navegador web sin necesidad de configurar un entorno local de desarrollo.

Ofrece a sus usuarios acceso gratuito, pero limitado, a Unidades de Procesamiento Gráfico (GPU) y Unidades de Procesamiento Tensorial (TPU). Cuenta además con una amplia gama de bibliotecas y herramientas útiles para el desarrollo de sistemas basados en aprendizaje automático tales como Keras, TensorFlow, PyTorch, Pandas y Numpy.

La versión de pago de Google Colab ofrece mayor capacidad de memoria y tiempos de ejecución en cada sesión, además de mayor disponibilidad de los recursos de procesamiento, como GPU y TPU [11].

Por otra parte, los datos pueden almacenarse en Google Drive, donde la versión gratuita tiene una limitación de 15 GB, o en Google Cloud Storage, un servicio para almacenar objetos de hasta 5 TB.

4.1.2. Google Cloud Storage

Google Cloud Storage (GCS) es un servicio de almacenamiento de datos no estructurados en la nube. Estos datos son almacenados en contenedores llamados *buckets*, desde el que se puede acceder mediante Google Colaboratory. Este servicio se caracteriza por su rendimiento, latencia y durabilidad cuando se trata de grandes conjuntos de datos y operaciones complejas de procesamiento.

GCS ofrece diferentes clases de almacenamiento, entre ellas el Standard Storage el cuál es usado en el presente proyecto y proporciona el nivel más alto de disponibilidad y rendimiento, además de ser ideal para datos a los que se accede con frecuencia y en proyectos a corto plazo.

El coste de este tipo de almacenamiento es de 0.018 € por GB al mes [12]. Sin embargo, GCS ofrece una prueba gratuita de 90 días de la que se ha hecho uso para el desarrollo del proyecto.

4.1.3. Unidades de procesamiento tensorial (TPU)

Las Unidades de Procesamiento Tensorial (TPU del inglés *Tensor Processing Unit*) [13], creadas por Google Research, no son más que circuitos integrados personalizados

para acelerar tareas relacionadas al aprendizaje automático que requieren cálculos complejos con grandes volúmenes de datos.

Han sido desarrolladas como una alternativa eficiente y de alto rendimiento a las conocidas Unidades de Procesamiento Gráfico (GPU) y a las CPU convencionales, ofreciendo mayor velocidad de procesamiento, mejor ancho de banda de memoria y menos consumo de energía. Pueden ser empleadas con diferentes frameworks de aprendizaje automático, tales como Pytorch o TensorFlow.

La arquitectura de las TPU consta de múltiples núcleos de procesamiento tensorial que permiten realizar en paralelo complejos cálculos vectoriales y matriciales a alta velocidad. Cada núcleo cuenta con su propia memoria local de alta velocidad empleada para almacenar los datos y parámetros necesarios para realizar los cálculos, lo cual genera una gran optimización de los tiempos de acceso.

Este recurso se encuentra disponible de forma gratuita a través de Google Colaboratory que ofrece una TPU de 8 núcleos y está configurada para poder leer datos desde Google Cloud Storage.

4.1.4. TensorFlow

TensorFlow [14] es una biblioteca de código abierto diseñada específicamente para el desarrollo e implementación de modelos de aprendizaje automático. Fue diseñada por Google y está optimizada para obtener el máximo rendimiento de las TPU descritas anteriormente.

Permite leer y procesar grandes volúmenes de datos almacenados en GCS de una manera eficiente, garantizando un flujo de datos óptimos desde el almacenamiento en la nube hasta las TPU. TensorFlow proporciona una gama amplia de capas, optimizadores, funciones de activación y métricas diseñadas para el entrenamiento de modelos de aprendizaje profundo.

Google brinda además una API de alto nivel de TensorFlow, llamada Keras [15], que permite de una forma intuitiva la construcción y entrenamiento de modelos de aprendizaje profundo.

4.1.5. Formato TFRecord

TFRecord es un formato de archivo binario propio de TensorFlow para almacenar y procesar grandes volúmenes de datos de manera rápida y eficiente.

Los datos en este formato ocupan menos espacio en disco siendo comprimidos y transformados en formato binario. Admite además múltiples hilos de lectura y escritura de manera simultánea, lo cual permite un alto rendimiento en el procesamiento de datos en sistemas distribuidos.

Un archivo en formato TFRecord contiene una secuencia de registros, donde cada registro consta de una o más características con sus valores respectivos. Estas características pueden ser de diferentes tipos (enteros, flotantes o cadenas de textos).

4.2. Conjunto de datos

Desde el 2016, el *International Skin Imaging Collaboration* (ISIC) ha patrocinado anualmente desafíos sobre visión artificial basados en la detección de melanoma, ofreciendo un conjunto amplio de imágenes con diagnósticos verificados mediante histopatología¹ y bajo el acuerdo de médicos expertos de los siguientes institutos: Hospital Clínic de Barcelona, Medical University of Vienna, Memorial Sloan Kettering Cancer Center, Melanoma Institute Australia, The University of Queensland, y University of Athens Medical School.

En el año 2018, los resultados superaron sistemáticamente a médicos especializados, pudiendo ofrecer en cuestión de segundos y con alta precisión, un diagnóstico sobre imágenes de lesiones cutáneas.

Posteriormente, ISIC propuso en sus desafíos del 2019 y 2020 evaluar el impacto que podría ofrecer el contexto clínico de los pacientes. De esta forma, en el desafío del año 2019 [16] se buscó clasificar nueve distintos tipos de lesiones de piel sobre imágenes dermatoscópicas incluyendo la información contextual del paciente en el sistema de detección. Este conjunto cuenta con 25.331 imágenes y un fichero tipo *Comma-Separated Values* (CSV) con la información médica del paciente.

A su vez, para el año 2020, ISIC en asociación con *Society for Imaging Informatics*

¹**Hispatología:** la rama de la Patología que trata el diagnóstico de enfermedades a través del estudio de los tejidos.

in Medicine (SIIM), solicitó a los participantes diseñar un algoritmo que fuese capaz de identificar la presencia de melanoma en imágenes dermatoscópicas junto con la información contextual del paciente. Para este desafío se presentó un conjunto de datos de 33.126 imágenes y el correspondiente fichero CSV con los datos del paciente.

En este sentido, para el desarrollo del presente proyecto se emplean ambos conjuntos de datos (*ISIC Challenge 2019* [17, 18, 19] y el *Melanoma Classification Challenge 2020* [20]), contando con un total de 58.457 imágenes tal como se refleja en la Tabla 4.1.

Tabla 4.1: Conjunto total de datos

Conjunto de datos	Número de datos
ISIC Challenge 2019	25.331
ISIC Challenge 2020	33.126
Total	58.457

Cada imagen es asociada a un paciente mediante un identificador único. La información clínica de cada paciente se proporciona en un fichero tipo CSV con la siguiente información para cada una de las imágenes:

- `image_name` - Identificador único de la imagen
- `patien_id` - Identificador único del paciente
- `sex` - Sexo del paciente
- `age_approx` - Edad aproximada del paciente
- `anatom_site_general_challenge` - Ubicación de la lesión en el cuerpo
- `diagnosis` - Información detallada del diagnóstico
- `benign_malignant` - Identificador de malignidad o no de la lesión
- `target` - Identificador binario de malignidad de la lesión (1 maligno, 0 benigno)

Los organizadores del desafío proporcionan las imágenes tanto en formato JPEG como en *Digital Imaging and Communications in Medicine* (DICOM).

Chris Deotte, participante de la competencia de SIIM-ISIC Melanoma 2020, transformó los conjuntos de datos del año 2019 y 2020 a formato *TensorFlow Record* (TFRecord) [21].

Estos conjuntos de datos se encuentran equilibrados donde cada fichero TFRecord cuenta con un 1,8% de imágenes dermatoscópicas y datos de pacientes con diagnóstico de lesiones malignas, y están disponibles en diferentes tamaños (256x256, 384x384, 512x512 y 768x768 píxeles). Para la realización de este proyecto se ha optado por emplear el formato de 256x256 píxeles [22] [23].

Es importante destacar que tras cargar el conjunto de datos en formato TFRecord, se observa una pérdida de datos. En la Tabla 4.2 se observa el total de datos de entrada disponibles y la cantidad de datos perdidos. El número de datos perdidos representa apenas un 0,84% del conjunto total empleado, por lo que la ausencia de estos no afectará significativamente el rendimiento del sistema.

Tabla 4.2: Distribución real del conjunto de datos en formato TFRecord

Conjunto de datos	Datos disponibles	Datos esperados	Datos perdidos
ISIC Challenge 2019	25.272	25.331	59
ISIC Challenge 2020	32.692	33.126	434
Total	57.964	58.457	493

En la Figura 4.1 se muestra un conjunto aleatorio de imágenes con diagnóstico positivo de melanoma, y en la Figura 4.2 un conjunto de imágenes con diagnóstico negativo.

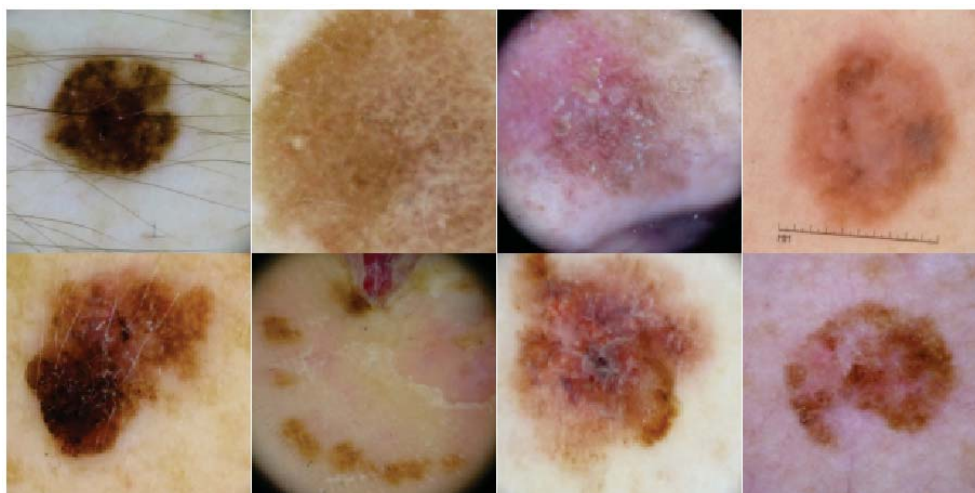


Figura 4.1: Imágenes dermatoscópicas con diagnóstico positivo de melanoma

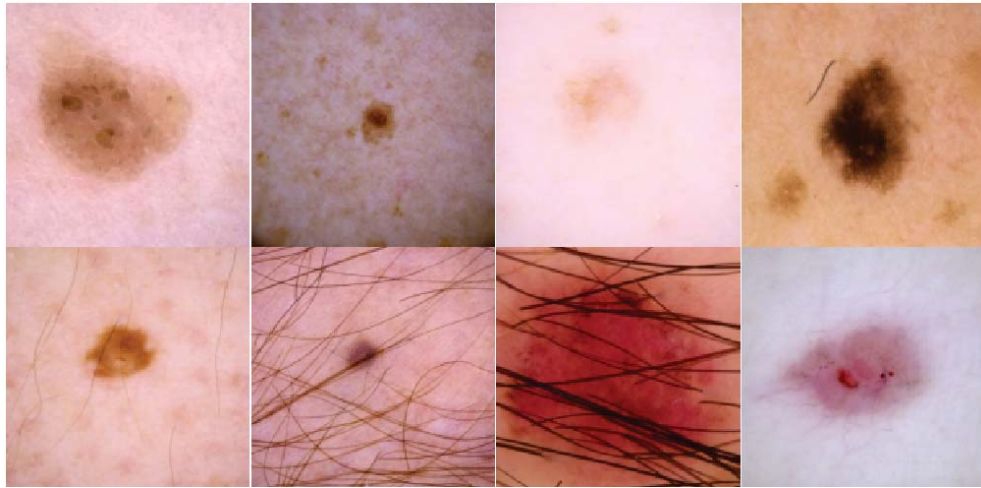


Figura 4.2: Imágenes dermatoscópicas con diagnóstico negativo de melanoma

4.2.1. Análisis exploratorio de los datos

En el presente proyecto se hace uso de los siguientes metadatos: `target`, `sex`, `age_approx` y `anatom_site_general_challenge`.

Esta información contextual del paciente será tomada desde los ficheros CSV proporcionados por SIIM-ISIC para los desafíos, y se distribuye de la siguiente manera según su clasificación de diagnóstico benigno o maligno:

target: Identificador del diagnóstico de melanoma

En la Tabla 4.3 se observa la distribución de los datos según su diagnóstico. Existe un desbalanceo significativo entre las dos clases, aspecto importante a tomar en cuenta en el diseño del modelo que se explicará en apartados posteriores.

Tabla 4.3: Distribución de datos según su diagnóstico

Conjunto de datos	Diagnóstico benigno	Diagnóstico maligno	Total
ISIC Challenge 2019	20.809	4.522	25.331
ISIC Challenge 2020	32.542	584	33.126
Total	53.351	5.106	58.457

sex: Sexo del paciente

Tal y como se observa en la Tabla 4.4, en ambos conjuntos de datos no se recopiló el sexo en algunas tomas. Estas se catalogan como desconocido y se le otorga al sistema de esta forma. Se refleja que la incidencia es mayor en personas del sexo masculino.

Tabla 4.4: Distribución del dato 'sex' en los conjuntos de datos

Conjunto de datos ISIC Challenge 2019			
Sexo	Diagnóstico benigno	Diagnóstico maligno	Total
Masculino	10.825	2.461	13.286
Femenino	9.681	1.980	11.661
Desconocido	303	81	384
Total	20.809	4.522	25.331
Conjunto de datos ISIC Challenge 2020			
Sexo	Diagnóstico benigno	Diagnóstico maligno	Total
Masculino	16.716	364	17.080
Femenino	15.761	220	15.981
Desconocido	65	0	65
Total	32.542	584	33.126
Conjunto de datos total			
Sexo	Diagnóstico benigno	Diagnóstico maligno	Total
Masculino	27.541	2.825	30.366
Femenino	25.442	2.200	27.642
Desconocido	368	81	449
Total	53.351	5.106	58.457

age_approx: Edad del paciente

En la Tabla 4.5 se observa la distribución de las edades en los conjuntos de datos del 2019 y 2020, mientras que en la Tabla 4.6 la distribución total de ambos conjuntos de datos. Estas edades son aproximadas.

El melanoma tiene una mayor incidencia en personas con edades comprendidas entre los 50 y 70 años de edad.

Tabla 4.5: Distribución del dato 'age_approx' en los conjuntos de datos

ISIC Challenge 2019				ISIC Challenge 2020		
Edad	Diagnóstico benigno	Diagnóstico maligno	Total	Diagnóstico benigno	Diagnóstico maligno	Total
0	54	0	54	2	0	2
5	112	1	113	0	0	0
10	141	1	142	17	0	17
15	370	5	375	130	2	132
20	370	18	388	649	6	655
25	632	45	388	1.528	16	1.544
30	1.097	102	1.199	2.334	24	2.358
35	1.475	176	1.651	2.825	25	2.850
40	1.988	258	2.246	3.552	24	3.576
45	2.212	373	2.585	4.412	54	4.466
50	2.025	464	2.489	4.217	53	4.270
55	1.769	401	2.170	3.760	64	3.824
60	1.626	410	2.036	3.175	65	3.240
65	1.597	478	2.075	2.457	70	2.527
70	1.550	570	2.120	1.910	58	1.968
75	1.413	383	1.796	919	62	981
80	1.068	391	1.459	383	36	419
85	958	361	1.319	140	9	149
90	0	0	0	64	16	80
desc.	352	85	437	68	0	68
Total	20.809	4.522	25.331	32.542	584	33.126

Tabla 4.6: Distribución del dato 'age_approx' en el conjunto de datos completo

Conjunto de datos total			
Edad	Diagnóstico benigno	Diagnóstico maligno	Total
0	56	0	56
5	112	1	113
10	158	1	159
15	500	7	507
20	1.019	24	1.043
25	2.160	61	2.221
30	3.431	126	3.557
35	4.300	201	4.501
40	5.540	282	5.822
45	6.624	427	7.051
50	6.242	517	6.759
55	5.529	465	5.994
60	4.801	475	5.276
65	4.054	548	4.602
70	3.460	628	4.088
75	2.332	445	2.777
80	1.451	427	1.878
85	1.098	370	1.468
90	64	16	80
desc.	420	85	505
Total	53.351	5.106	58.457

anatom_site_general_challenge: Localización de la lesión

Como se observa en la Tabla 4.7, en el conjunto de datos del 2019 las imágenes dermatoscópicas tomadas del torso fueron catalogadas según la parte anterior, posterior y lateral del mismo; mientras que en el 2020 se catalogaron de forma general como imágenes tomadas del torso.

Tal y como se observa, la incidencia del melanoma es mayor en las zonas expuestas al sol, como lo son la cabeza, el cuello, torso (pecho y espalda), piernas y brazos.

Tabla 4.7: Distribución del dato 'anatom_site_general_challenge' en el conjunto de datos

ISIC Challenge 2019			
Parte del cuerpo	Diagnóstico benigno	Diagnóstico maligno	Total
tórax/abdomen	5.584	1.331	6.915
espalda	2.357	430	2.787
torso lateral	40	14	54
extremidad baja	4.194	796	4.990
extremidad superior	2.186	724	2.910
cabeza/cuello	3.707	880	4.587
palmas/suelas	197	201	398
oral/genital	40	19	59
desconocido	2.504	127	2.631
Total	20.809	4.522	25.331
ISIC Challenge 2020			
Parte del cuerpo	Diagnóstico benigno	Diagnóstico maligno	Total
torso	16.588	257	16.845
extremidad baja	8.293	124	8.417
extremidad superior	4.872	111	4.983
cabeza/cuello	1.781	74	1.855
palmas/suelas	370	5	375
oral/genital	120	4	124
desconocido	518	9	527
Total	32.542	584	33.126
Conjunto de datos total			
Parte del cuerpo	Diagnóstico benigno	Diagnóstico maligno	Total
tórax/abdomen	5.584	1.331	6.915
espalda	2.357	430	2.787
torso lateral	40	14	54
torso	16.588	257	16.845
extremidad baja	12.487	920	13.407
extremidad superior	7.058	835	7.893
cabeza/cuello	5.488	954	6.442
palmas/suelas	567	206	773
oral/genital	160	23	183
desconocido	3.022	136	3.158
Total	53.351	5.106	58.457

En el conjunto de datos empleado para el desarrollo del sistema se unificaron las lesiones de piel del torso. Esta distribución final se observa en la Tabla 4.8.

Tabla 4.8: Distribución del dato 'anatom_site_general_challenge' en el conjunto de datos unificando las lesiones ubicadas en el torso

ISIC Challenge 2019			
Parte del cuerpo	Diagnóstico benigno	Diagnóstico maligno	Total
torso	7.981	1.775	9.756
extremidad baja	4.194	796	4.990
extremidad superior	2.186	724	2.910
cabeza/cuello	3.707	880	4.587
palmas/suelas	197	201	398
oral/genital	40	19	59
desconocido	2.504	127	2.631
Total	20.809	4.522	25.331
ISIC Challenge 2020			
Parte del cuerpo	Diagnóstico benigno	Diagnóstico maligno	Total
torso	16.588	257	16.845
extremidad baja	8.293	124	8.417
extremidad superior	4.872	111	4.983
cabeza/cuello	1.781	74	1.855
palmas/suelas	370	5	375
oral/genital	120	4	124
desconocido	518	9	527
Total	32.542	584	33.126
Conjunto de datos total			
Parte del cuerpo	Diagnóstico benigno	Diagnóstico maligno	Total
torso	24.569	2.032	26.601
extremidad baja	12.487	920	13.407
extremidad superior	7.058	835	7.893
cabeza/cuello	5.488	954	6.442
palmas/suelas	567	206	773
oral/genital	160	23	183
desconocido	3.022	136	3.158
Total	53.351	5.106	58.457

4.3. Modelo de la solución propuesta

Una vez configurado el entorno de trabajo y estudiado la estructura del conjunto de datos a emplear, se plantea como solución general el diagrama ilustrado en la Figura 4.3.

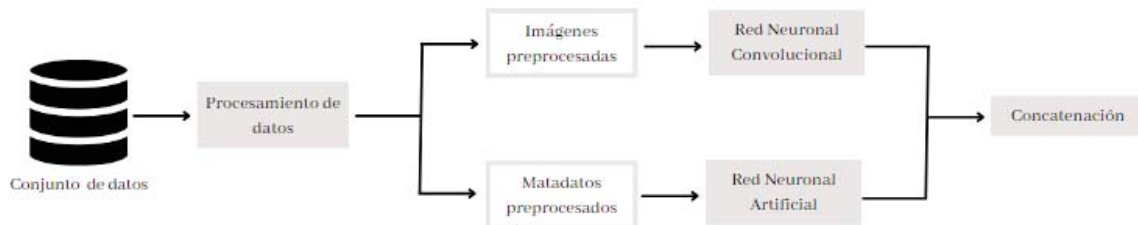


Figura 4.3: Diagrama general del sistema propuesto

A continuación se describe por separado los módulos de clasificación de melanoma en imágenes dermatoscópicas y clasificación de melanoma a partir de la información contextual del paciente. Finalmente se describirá la solución final tras combinar el resultado de ambos módulos.

4.3.1. Módulo de clasificación de melanoma en imágenes dermatoscópicas

En primer lugar, al cargar el conjunto de datos disponible en formato TFRecord se observa una pérdida del 0,84% del total de datos ofrecidos por los organizadores del desafío. Se asume dicha pérdida al representar un bajo porcentaje.

En este sentido, la distribución de las imágenes con las que se desarrolla el sistema según su diagnóstico se refleja en la Tabla 4.9:

Tabla 4.9: Distribución real de los datos según su diagnóstico en el conjunto de datos ofrecido en formato TFRecord

Conjunto de datos	Diagnóstico benigno	Diagnóstico maligno	Total
ISIC Challenge 2019	20.763	4.509	25.272
ISIC Challenge 2020	32.111	581	32.692
Total	52.874	5.090	57.964

Por su parte, con el objetivo de garantizar una evaluación precisa y mejorar el rendimiento del modelo de forma objetiva, se realiza una división del conjunto de datos de forma estratificada para el entrenamiento y comprobación del modelo. Se divide el conjunto de datos en tres grandes subconjuntos de datos:

- Conjunto de entrenamiento.
- Conjunto de validación, para ajustar los parámetros del modelo.
- Conjunto de test, para medir la capacidad del modelo al clasificar las imágenes.

La distribución de estos subconjuntos de datos según el diagnóstico se refleja en la Tabla 4.10.

Tabla 4.10: Distribución de los subconjuntos de datos según su diagnóstico

Subconjunto de datos	Diagnóstico benigno	Diagnóstico maligno	Total
Entrenamiento	34.490	3.282	37.772
Validación	6.190	648	6.838
Test	12.194	1160	13.354
Total	52.874	5.090	57.964

Ahora bien, la tasa de diagnósticos malignos representa a penas un 8,78 % del total de imágenes disponibles en todo el conjunto. Numerosos estudios han demostrado que un alto grado de desbalanceo puede generar efectos negativos en el rendimiento del sistema [24], pudiendo presentarse una tendencia a predecir de forma incorrecta la clase minoritaria, por lo que se estudian diferentes métodos para mitigar dicho problema.

Entre los más empleados en el entrenamiento de redes neuronales artificiales destacan los siguientes:

- Sobre-muestreo: Repetición de datos de la clase minoritaria.
- Sub-muestreo: Eliminación de datos de la clase mayoritaria.
- *Data augmentation*: Generación de nuevos datos a través de transformaciones de los datos existentes.
- Ajuste de hiperparámetros de la Red entrenada.

- Ponderación de clases: Indicar al modelo el peso que debe tomar para cada clase.

En este proyecto se realizarán pruebas con los métodos de *data augmentation*, ponderación de clases, y ajuste de hiperparámetros, con el fin de conseguir los mejores resultados de rendimiento.

A continuación se describe el desarrollo e implementación de las mismas.

Data augmentation

El *data augmentation* o el aumento de datos es una técnica para incrementar los datos de entrenamiento a partir de datos disponibles, mediante una serie de transformaciones aleatorias. En el caso de las imágenes se pueden realizar transformaciones geométricas y del espacio de color, tales como:

- Rotación horizontal (Figura 4.4):



Figura 4.4: Data augmentation: rotación horizontal

- Rotación vertical (Figura 4.5):



Figura 4.5: Data augmentation: rotación vertical

- Ajuste del tono de las imágenes (Figura 4.6):



Figura 4.6: Data augmentation: ajuste del tono de las imágenes

- Ajuste de la saturación de las imágenes (Figura 4.7):



Figura 4.7: Data augmentation: ajuste de la saturación de las imágenes

- Ajuste del contraste de las imágenes (Figura 4.8):



Figura 4.8: Data augmentation: ajuste de la saturación de las imágenes

- Ajuste del brillo de las imágenes (Figura 4.9):

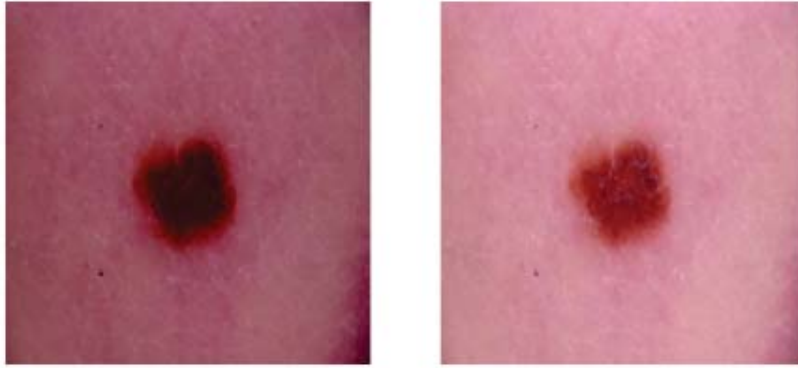


Figura 4.9: Data augmentation: ajuste de la saturación de las imágenes

Ponderaciones de clases

Durante el proceso de entrenamiento del modelo es posible abordar el desequilibrio de las clases ajustando la contribución de cada una de ellas en el modelo. Se asignan pesos más altos a las clases minoritarias permitiendo que el modelo preste más atención y dedique más esfuerzo a aprender a identificarlas.

El cálculo de las ponderaciones de clases, o *class_weight* como se describen en algunas bibliografías, puede representarse como la proporción inversamente proporcional a la frecuencia de cada clase, asignando un peso menor a las clases más frecuentes y un peso mayor a las menos frecuentes (Ecuación 4.1)

$$class_weight = \frac{total\ de\ datos\ de\ entrada}{número\ de\ clases \cdot total\ de\ datos\ de\ la\ clase} \quad (4.1)$$

Este valor de ponderación se calcula para cada una de las clases y se le otorga al sistema como parámetro en la función de entrenamiento.

En el caso del sistema desarrollado en este Proyecto la clase 0 (diagnóstico benigno y clase mayoritaria) tiene un peso de 0.56 y la clase 1 (diagnóstico maligno y clase minoritaria) tiene un peso de 4.86.

Ajuste de hiperparámetros

Durante la definición de la arquitectura y el entrenamiento del modelo se establecen diferentes parámetros que permiten optimizar el rendimiento del modelo tras el desequilibrio de las clases, tales como:

- **Función de pérdida BinaryCrossEntropy:**

Comúnmente empleada en problemas de clasificación binaria para cuantificar la discrepancia entre los valores reales y las predicciones del modelo. Cuanto mayor sea dicha discrepancia, mayor será el valor de la pérdida.

En términos generales su objetivo es ajustar los parámetros del modelo durante el entrenamiento para disminuir de esta forma las discrepancias antes descritas.

- **Optimizador Adam:**

Conocido como *Adaptive Moment Estimation* es un algoritmo que permite actualizar los pesos de un modelo durante el proceso de entrenamiento, combinando la adaptación de tasas de aprendizaje individualizadas, estimaciones de momentos de gradientes y decaimiento exponencial para ofrecer un rendimiento efectivo y eficiente.

- **Epochs:**

Se refiere al número de veces que el algoritmo de entrenamiento recorre el conjunto de datos de entrenamiento por completo. Durante cada *epoch* el algoritmo de entrenamiento realiza las siguientes tareas sobre cada una de los datos de entrada al modelo de entrenamiento:

1. Analiza cada dato de entrada y obtiene predicciones sobre ella.
2. Calcula la función de pérdida para evaluar la calidad de la predicción hecha.
3. Emplea la función de pérdida para calcular el gradiente de los pesos del modelo.
4. Actualiza los pesos del modelo empleando un algoritmo de optimización.

Como puede resultar evidente, durante más *epochs* de entrenamiento el modelo mejora su capacidad de generalización. Sin embargo, un número alto puede conllevar al sobreajuste, u *overfitting*, y tener dificultades para generalizar en nuevos datos perdiendo la capacidad de hacer predicciones precisas en datos no vistos previamente.

Una vez definido y normalizado el conjunto de datos de entrada, y establecidos los parámetros a tener en cuenta, se define el modelo de Redes Neuronales Convolucionales para su entrenamiento. Se describe, a continuación, las distintas capas que la componen.

Modelo de CNN para la clasificación de imágenes

En la Figura 4.10 se observa en forma de gráfico el modelo aplicado.

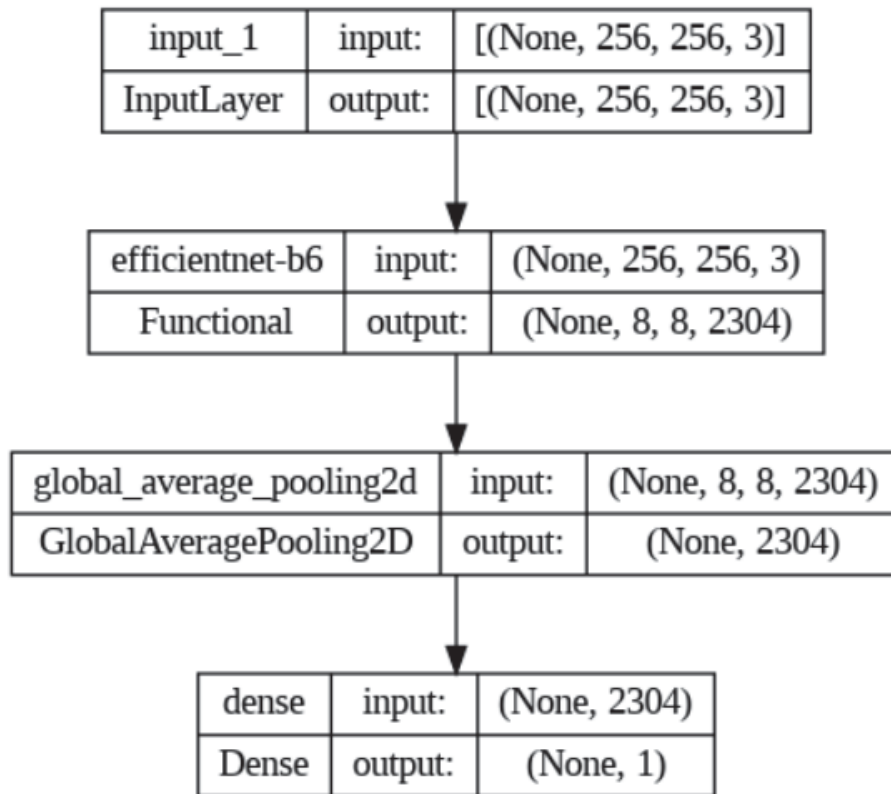


Figura 4.10: Modelo de CNN aplicado a la clasificación de imágenes

Donde:

- **Capa de entrada:**

El modelo recibe como entrada una imagen en formato de arreglo tridimensional, definido por sus tres dimensiones (alto, ancho y canal). Para el entrenamiento del modelo se emplea el conjunto de datos en tamaño de 256x256 píxeles.

- **Capa de convolución:**

Se define con el modelo de arquitectura de EfficientNet, realizando diferentes pruebas para sus dos modelos EfficientNetB5 y EfficientNetB6, siendo estos dos de los modelos de CNN con mayor tasa de rendimiento [10].

Para la implementación del sistema se emplea, además, la técnica de *transfer learning*, o aprendizaje por transferencia, permitiendo adaptar y emplear los conocimientos previos y características aprendidas por EfficientNet en tareas de clasificación de imágenes aplicadas en la gran base de datos *ImageNet*.

- **Capa de agrupación *GlobalAveragePooling2D***

Empleada para reducir las dimensiones espaciales de los mapas de características a la salida de la capa de convolución realizando una operación de promedio sobre todas las posiciones espaciales de cada canal de la imagen.

- **Capa totalmente conectada (Dense)**

Se agrega la capa Dense como capa totalmente conectada para realizar la clasificación final, que combina y transforma las características de las capas previas mediante una combinación lineal ponderada y la función de activación *sigmoide*.

En esta capa, se define además, el parámetro inicializador de sesgo o *bias_initializer*. Permite ajustar el punto de partida de la función de activación proporcionando una capacidad adicional de aprendizaje a la red neuronal.

El sesgo ayuda a modelar la relación entre las características de entrada y las salidas objetivo, permitiendo que la red aprenda desplazamientos o desviaciones sistemáticas en los datos.

Se establece un valor constante para todos los sesgos de esta capa, establecido por el cálculo del logaritmo natural del cociente del número de ocurrencias positivas entre el número de ocurrencias negativas (Ecuación 4.2)

$$bias_initializer = \log\left(\frac{total\ de\ datos\ positivos}{total\ de\ datos\ negativos}\right) \quad (4.2)$$

4.3.2. Módulo de clasificación de melanoma con metadatos

Se toman como datos de entrada relevantes la edad del paciente, el sexo del paciente y la localización de la lesión en el cuerpo. Estos datos son tomados desde el fichero TFRecord de donde también se obtuvieron las imágenes.

La división del conjunto de datos para entrenamiento, validación y test para este modulo es la misma, por lo que las predicciones finales se harán sobre el mismo conjunto de datos de test.

Los datos del paciente se obtienen desde el conjunto de datos en formato TFRecord codificados de la siguiente forma:

- **Edad (*age_approx*):** Número entero de 64bits desde 0 a 90. La edad en los pacientes desconocidos se establece como la media de las edades.

- **Sexo (*sex*):** Codificada como:
 - -1: Desconocido
 - 0: Masculino
 - 1: Femenino

- **Localización de la lesión en el cuerpo (*anatom_site_general_challenge*):** Codificada según su ubicación como:
 - -1 Desconocido
 - 0: cabeza/cuello
 - 1: extremidad superior
 - 2: extremidad inferior
 - 3: torso
 - 4: palmas/suelas
 - 5: oral/genital

Las variables sexo y localización de la lesión en el cuerpo han sido codificadas con el método de OneHotEncoder. De esta forma se cuenta con 11 entradas para la Red Neuronal a desarrollar para el módulo descrito.

Una vez normalizados los datos de entrada, se establece el modelo de Red Neuronal a implementar.

Modelo de Red Neuronal para la clasificación de metadatos

En la Figura 4.11 se observa en forma de gráfico el modelo aplicado.

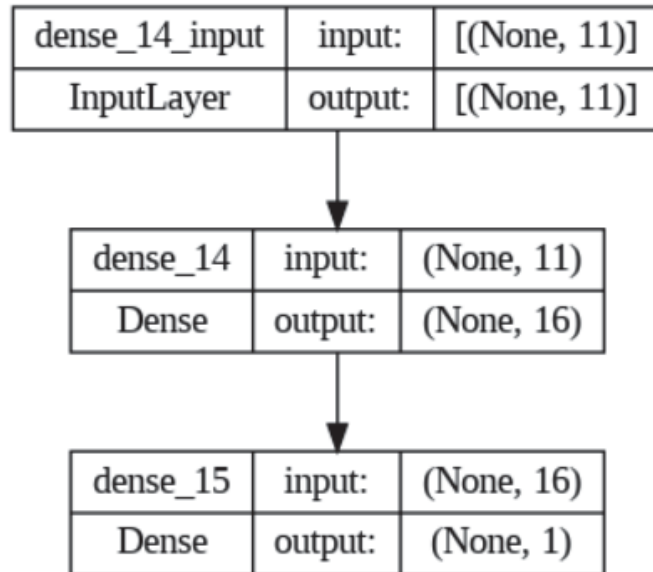


Figura 4.11: Modelo de Red Neuronal aplicado a la clasificación de metadatos

Se define un modelo secuencial, donde:

- **Capa Dense:**

Capa densa de 16 neuronas de salida, una función de activación ReLu y una regularización L2 con un factor de penalización de 0.01, y una entrada de 11 dimensiones.

- **Capa Dense:**

Capa densa de 1 neurona de salida y función de activación *sigmoid* para la clasificación binaria.

Por otro lado, se aplica el método de ponderación de clases, anteriormente descrito, y se establece una tasa de aprendizaje de 0.001 con un optimizador de tipo Adam.

4.3.3. Ensamblado final

Como salida del proceso se realiza una predicción final del sistema mediante la combinación de la predicción de ambos módulos, aplicando el método de *Soft Voting*. En líneas generales, el esquema del modelo se representa como la Figura 4.12

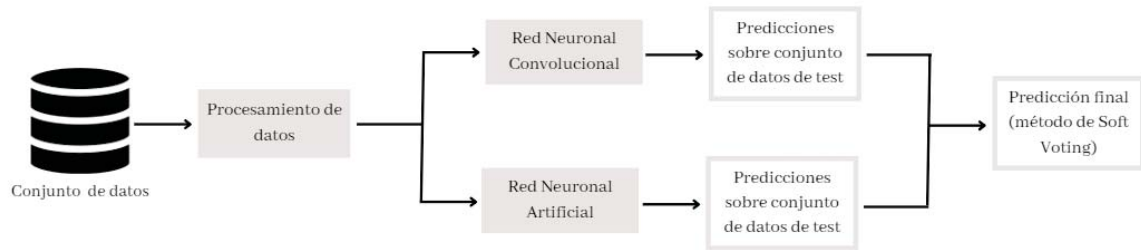


Figura 4.12: Diagrama general del sistema propuesto

El *Soft Voting* es una técnica de ensamble empleada para combinar las predicciones de varios modelos individuales, obteniendo una predicción final con mayor precisión, aprovechando la diversidad de los modelos individuales y combinando sus enfoques de clasificación disminuyendo la sensibilidad a decisiones individuales incorrectas.

En este sentido, se realiza una serie de pruebas sobre los módulos de clasificación de imágenes y de metadatos antes descritos con el objetivo de tomar el mejor modelo desarrollado, se toman predicciones de ambos módulos, y se calcula el promedio ponderado de las probabilidades de clasificación de ambos modelos.

Capítulo 5

Análisis de resultados

En este Capítulo se representan los resultados obtenidos durante las pruebas realizadas en las diferentes etapas del Proyecto. Se evalúa progresivamente cada módulo de entrenamiento por separado con el fin de obtener el modelo con mayor rendimiento. Finalmente, se evalúa la unión de ambos módulos obteniendo las predicciones finales al sistema desarrollado.

5.1. Conjunto de datos de validación

Durante el proceso de entrenamiento se emplea el conjunto de datos de validación para ajustar los hiperparámetros del modelo, seleccionar la mejor configuración del mismo y monitorear el rendimiento del modelo.

Este conjunto de datos cuenta con un total de 6.383 datos de entrada, de entre los cuales 648 cuentan con un diagnóstico positivo de melanoma.

5.2. Conjunto de datos de evaluación

La evaluación del modelo final se realiza con un subconjunto de datos dedicado para ello. En este sentido, durante la fase de entrenamiento y validación, el sistema no ha tomado en cuenta los datos proporcionados en dicho subconjunto.

Este conjunto cuenta con un total de 13.354 datos de entrada, de entre los cuales 1.160 cuentan con un diagnóstico de melanoma.

5.3. Métricas de evaluación

Dado que el objetivo principal del sistema es predecir las muestras con diagnóstico de melanoma, resulta importante evaluar las métricas de precisión y exhaustividad, o *recall*.

Se considera como modelo ideal el que ofrezca mayores porcentajes de predicción y exhaustividad. Un modelo con una predicción baja y una exhaustividad alta será capaz de generar un alto porcentaje de diagnósticos malignos correctos; sin embargo, ofrecerá un porcentaje también alto de falsos diagnósticos malignos. Aunque esto tenga un coste, se considera sumamente importante prevalecer el diagnóstico de verdaderos casos positivos de melanoma.

5.4. Evaluación del módulo de clasificación de imágenes

Tal como se describe en el Capítulo anterior, se realizan diferentes pruebas de entrenamiento.

En la Tabla 5.1 se ilustran los valores obtenidos tras la validación del modelo variando los siguientes parámetros:

- Arquitectura de la CNN: EfficientNetB5 y EfficientNetB6.
- Asignación de tasa de aprendizaje variable entre 0.0005, 0.005 y 0.001.

Destaca el modelo desarrollado con la arquitectura EfficientNetB5 y la asignación de una tasa de aprendizaje de 0,001 para el optimizador Adam.

Tomando como base dicho modelo, se aplica a continuación las siguientes técnicas en pro de mejorar el rendimiento del modelo de clasificación tras el desbalance de los datos de entrada. Los resultados se observan en la Tabla 5.2:

- Asignación del parámetro *bias_initializer* a la capa totalmente conectada.
- Asignación de ponderación de clases.

Tabla 5.1: Conjunto de pruebas de entrenamiento para el módulo de clasificación de imágenes

Modelo	Arquitectura CNN	Tasa de aprendizaje	Exactitud	Precisión	Exhaustividad	F1-score
1	EfficientNetB5	0,001	94,59 %	79,01 %	71,45 %	75,04 %
2	EfficientNetB6	0,001	94,61 %	81,42 %	65,59 %	72,65 %
3	EfficientNetB5	0,005	94,64 %	81,71 %	66,20 %	73,14 %
4	EfficientNetB6	0,005	94,76 %	78,75 %	66,36 %	72,03 %
5	EfficientNetB5	0,0005	94,81 %	81,73 %	62,81 %	71,03 %
6	EfficientNetB6	0,0005	94,59 %	84,50 %	67,28 %	74,91 %

Tabla 5.2: Segundo conjunto de pruebas de entrenamiento para el módulo de clasificación de imágenes con la arquitectura EfficientNetB5

Modelo	Arquitectura CNN	Tasa de aprendizaje	<i>bias_initializer</i>	Ponderación de clases	Exactitud	Precisión	Exhaustividad	F1-score
1	EfficientNetB5	0,001	No	No	94,59 %	79,01 %	71,45 %	75,04 %
2	EfficientNetB5	0,001	Yes	No	94,31 %	80,56 %	66,51 %	72,86 %
3	EfficientNetB5	0,001	No	Yes	94,86 %	52,23 %	83,18 %	64,17 %
4	EfficientNetB5	0,001	Yes	Yes	95,34 %	53,31 %	82,41 %	64,74 %

A pesar de que se observa una ligera diferencia entre las métricas obtenidas en los modelos 3 y 4, se toma el número 3 como mejor modelo, donde se establece una tasa de aprendizaje de 0.001, y se aplica la ponderación de clases; siendo este parámetro destacable para la optimización de modelos de aprendizaje profundo con conjuntos de datos desbalanceados, tal y como es el caso.

Ahora bien, a pesar de que el porcentaje de precisión es inferior al modelo de detección sin definir la ponderación de clases, el porcentaje de exhaustividad es significativamente mayor, por lo que se toma como mejor modelo. Este ofrece un 83,18% de probabilidad de detectar que un paciente padece de cancer de piel de tipo melanoma. A pesar de esto, el porcentaje de baja precisión indica que un 52,23% de los pacientes detectados con melanoma realmente padecen de melanoma.

Una vez definido el mejor modelo, se evalúa el rendimiento del modelo ante el conjunto de prueba que contiene datos no vistos por la red durante el entrenamiento. Esta evaluación proporciona una medida objetiva del rendimiento del modelo en datos no vistos previamente (Tabla 5.3)

Tabla 5.3: Evaluación del modelo final de clasificación ante el conjunto de datos de prueba

Exactitud	Precisión	Exhaustividad	F1-score
93,69 %	57,79 %	74,98 %	65,27 %

5.4.1. Evaluación del módulo de clasificación de metadatos

Tras el entrenamiento del módulo de clasificación de metadatos se obtuvo que el modelo es preciso en un 69,90% y cuenta con una exhaustividad de 27,4% y una exactitud del 92,13%.

5.4.2. Evaluación del módulo final

Una vez realizadas las predicciones sobre cada módulo, se evalúa el desempeño que pueda tener el sistema tras realizar una ponderación de la predicciones realizadas, tal y como se explicó en el Capítulo 4.

En la Tabla 5.4 se observa el resultado de ensamblar ambos modelos tras el método

de Soft Voting.

Tal y como se observa, la toma en cuenta de los datos del paciente aumenta en un 0,64 % la exhaustividad del modelo para generar un diagnóstico maligno al ponderar las predicciones en un 90 % del módulo de imágenes y 10 % del módulo de metadatos.

Con esto, el sistema es capaz de ofrecer un 75,62 % de probabilidad de detectar que un paciente con melanoma lo padece. En contraposición, el sistema no posee una alta precisión, siendo que el 56,75 % de los pacientes detectados con melanoma en el conjunto de prueba realmente sufren de la enfermedad.

Esto último implica un coste médico elevado, puesto que el sistema, llevado a la práctica médica, puede conllevar a realizar biopsias no necesarias. Sin embargo, como se ha comentado anteriormente resulta de vital importancia que el sistema sea capaz de detectar con alta probabilidad las lesiones malignas en pacientes afectados con la enfermedad.

Tabla 5.4: Evaluación del ensamble del modelo final de clasificación

Modelo	Peso del módulo de clasificación de imágenes	Peso del módulo de clasificación de metadatos	Exactitud	Precisión	Exhaustividad	F1-score
1	1	0	93,69 %	57,79 %	74,98 %	65,27 %
2	0.9	0.1	93,74 %	56,75 %	75,62 %	64,84 %
3	0.8	0.2	93,63 %	48,25 %	68,32 %	56,56 %
4	0.75	0.25	93,65 %	47,86 %	67,91 %	56,15 %

Capítulo 6

Presupuesto

En este capítulo se proporciona una descripción detallada de los costes asociados que implicaría el desarrollo de este Proyecto Fin de Grado en un ámbito comercial. Es importante destacar que para la elaboración de este proyecto se han empleado servicios de computación en la nube gratuitos y con suscripciones gratuitas de prueba.

En la Tabla 6.1 se refleja el detalle de cada uno de los recursos empleados y sus costos.

Tabla 6.1: Presupuesto general del proyecto

Concepto	Costo unidad	Unidades	Total (€)
Portátil ASUS Vivobook F1402Z	750 (€)	1	50
Suscripción de Google Colab Pro+	51,12 (€/mes)	4 meses	204,48
Almacenamiento estándar en Google Cloud Storage	0.018 (€/GB/mes)	10 GB x 4 meses	40
Salario ingeniero	25 (€/hora)	320 horas	8.000
Total			8294,48

Tabla 6.2: Amortización de los equipos informáticos

Equipo	Costo del equipo	Vida útil (años)	Amortización anual
Portátil ASUS Vivobook F1402Z	750 €	5	50 €

Capítulo 7

Impacto del proyecto

El diagnóstico temprano de enfermedades cancerígenas es fundamental para incrementar las tasas de supervivencia de los pacientes, generar menos efectos secundarios y menos costes de tratamiento. Por lo tanto, es crucial contar con métodos de diagnóstico precisos y accesibles. En el caso del cáncer de piel, el método de detección más empleado es la dermatoscopia, donde médicos especializados visualizan, a través de un dermatoscopio, una imagen ampliada de la lesión en busca de potenciales patrones característicos de lesiones malignas.

Si bien este método ha resultado ser un avance importante en el área de la dermatología, también presenta una serie de limitaciones y desafíos. Su uso correcto y efectivo requiere de la experiencia y la habilidad de médicos experimentados siendo la interpretación de las lesiones de piel una tarea compleja, y una falsa interpretación podría dar paso a omitir lesiones potencialmente malignas.

Por otra parte, el acceso a médicos dermatólogos especializados supone una tarea difícil en entornos donde la atención médica cuenta con recursos limitados, lo que retrasa el diagnóstico temprano de la enfermedad.

Ante esta problemática, los sistemas de Aprendizaje Profundo, tales como el desarrollado en el presente Proyecto, han surgido como herramientas potencialmente prometedoras en el campo de la detección de melanoma. Su impacto se extiende a las siguientes grandes áreas:

- Salud: Detección temprana y efectiva de lesiones malignas disminuyendo la tasa de mortalidad de los pacientes afectados. Además de ser una herramienta de

soporte de apoyo para médicos dermatólogos.

- **Economía:** Disminución significativa en los costes de atención médica, tanto para el paciente como para los diferentes sistemas de salud.
- **Social:** Ofrecen la posibilidad de ampliar el acceso a una detección temprana de melanoma en entornos desfavorecidos o remotos, reduciendo las desigualdades en la atención médica.
- **Tecnología:** Estas herramientas pueden integrarse a tecnologías de análisis de imágenes en tiempo real, mejorando aún más la accesibilidad y la calidad de la atención médica. Adicionalmente, su desarrollo fomenta la investigación y el avance tecnológico en pro de la realización de mejoras continuas en los algoritmos de detección y en la comprensión de los factores de riesgo asociados.

Por otro lado, dentro del ámbito de los Objetivos de Desarrollo Sostenible (ODS) definidos por la Organización de las Naciones Unidas (ONU), el sistema desarrollado con los siguientes objetivos:

- Garantizar una vida sana y promover el bienestar para todos en todas las edades.
- Reducción de las desigualdades.

Capítulo 8

Conclusiones

En este capítulo se presentan las conclusiones generales obtenidas tras la realización del presente Proyecto Fin de Grado.

8.0.1. Conclusiones generales

El objetivo principal del presente Proyecto Fin de Grado es el desarrollo de un sistema capaz de detectar la presencia de melanoma en imágenes dermatoscópicas asociadas a datos clínicos de pacientes que son tomados en cuenta por los médicos expertos al realizar una toma de decisiones sobre la lesión, tales como la edad, el sexo y la localización de la lesión en el cuerpo.

Para la realización de este proyecto se tomó el conjunto de datos ofrecido por las asociaciones SIIM e ISIC en sus desafíos anuales del 2019 y 2020, contando con un conjunto de datos total de 57.964 datos de entrada, de los cuales solo un 8,78 % (5.090) son datos asociados a pacientes diagnosticados con cáncer de piel de tipo melanoma.

Este desbalance de casos en el conjunto de datos representa un gran desafío siendo que los sistemas de aprendizaje profundo requieren de una senda cantidad de datos para poder generalizar las características de los datos de entrada y realizar una clasificación con gran presión y exhaustividad.

Para contrarrestar dicho desbalance se aplicaron distintos métodos de aumento de datos y ajustes de parámetros del modelo, evitando así un sobre ajuste de la clase mayoritaria. Tras aplicar dichos métodos se observó que las métricas de evaluación resultaba superior.

A su vez, otra de las limitaciones al momento de desarrollar sistemas de esta índole, es el uso de los recursos computacionales. El modelo fue desarrollado con la herramienta en línea de Google Colaboratory, la cual ofrece recursos computacionales limitados en cuanto a número de TPU disponibles, capacidad de memoria y tiempo de uso. Esto representa un reto importante, siendo qué, en varias ocasiones el estudiante no contaba con recursos hardware disponibles para el entrenamiento del modelo.

Por su parte, el sistema se dividió en tres módulos distintos. En el primer módulo se entrenó una red neuronal convolucional con el conjunto de imágenes para su diagnóstico. En el segundo módulo se clasificaron los datos del paciente y la lesión en una red densa. Y finalmente, se aplicó el método de *Soft Voting* para obtener predicciones finales en base a la ponderación de cada módulo antes descrito.

Tras aplicar el método descrito se observó que la exhaustividad del sistema era mayor en comparación al sistema desarrollado tomando en cuenta únicamente las imágenes. En concreto, el sistema ofrece un 75,62 % de probabilidad de diagnosticar una lesión maligna en un paciente realmente afectado.

Si bien es cierto que este no representa un alto porcentaje de predicción, se debe tener en cuenta que estos sistemas desarrollados no sustituyen a médicos expertos, pero si resulta una excelente herramienta de soporte al momento de la toma de decisiones en el diagnóstico por dermatoscopia, método en el que se visualiza una imagen ampliada de la lesión de piel en busca de potenciales patrones de malignidad.

Por otro lado, en comparación con los resultados obtenidos por los ganadores del desafío propuesto por SIIM-ISIC para el año 2020 [25], además de contar con recursos computacionales superiores, mayor número de datos de entrada empleando conjuntos de datos publicados en años anteriores, se ensamblaron 18 diversos modelos variando el tamaño de las imágenes a la entrada de cada modelo.

De esta forma la solución ganadora del concurso obtuvo un modelo final con 94,90 % de exactitud, mientras que métricas como la exhaustividad y la precisión no fueron otorgadas por los ganadores al no ser un requisito del concurso.

Reportaron además, una duración de 45 horas de entrenamiento del modelo final.

Bibliografía

- [1] Sociedad Española de Oncología Médica (SEOM), “Las cifras del cáncer en España 2023”, 2023. [En línea]. Disponible en: https://seom.org/images/Las_cifras_del_Cancer_en_Espana_2023.pdf [Acceso: 07 de febrero 2023]
- [2] Cancer.Net, “Melanoma: Estadísticas”, 2023. [En línea]. Disponible en: <https://www.cancer.net/es/tipos-de-cancer/melanoma/estadisticas> [Acceso: 04 de abril 2023]
- [3] Ruzica Z. Conic, Claudia I. Cabrera, Alok A. Khorana, Brian R. Gastman, “Determination of the impact of melanoma surgical timing on survival using the National Cancer Database”, *Journal of the American Academy of Dermatology*, Volume 78, Issue 1, Pages 40-46.e7., 2018
- [4] Y. LeCun et al., “Backpropagation Applied to Handwritten Zip Code Recognition”, *Neural Computation*, Volume 1, no. 4, pp. 541-551, Dec. 1989, doi: 10.1162/neco.1989.1.4.541.
- [5] Lowe, D. G. (1999). Object recognition from local scale-invariant features. Proceedings of the seventh IEEE international conference on computer vision, 2, 1150-1157.
- [6] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25, 1097-1105.
- [7] IBM Cloud, “AI vs. Machine Learning vs. Deep Learning vs. Neural Networks: What’s the Difference?”, 2022. [En línea]. Disponible en: <https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks> [Acceso: 7 marzo 2023]

- [8] Wikipedia, “Redes neuronales convolucionales basadas en regiones”, 2022. [En línea]. Disponible en: https://ca.m.wikipedia.org/wiki/Xarxes_neuronals_convolucionals_basades_en_regions [Acceso: 7 febrero 2023]
- [9] Jahaziel Ponce blog personal sobre Inteligencia Artificial, “Conoce qué son las funciones de activación y cómo puedes crear tu función de activación usando python, R y tensorflow”, 2020. [En línea]. Disponible en: <https://jahazielponce.com/funciones-de-activacion-y-como-puedes-crear-la-tuya-usando-python-r-y-tensorflow/> [Acceso: 10 febrero 2023]
- [10] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In Proceedings of the 36th International Conference on Machine Learning (ICML), Long Beach, California, 2019.
- [11] Google Colaboratory, “Elige el plan de Colab adecuado para ti”, 2022. [En línea]. Disponible en: <https://colab.research.google.com/signup> [Acceso: 04 de abril 2023]
- [12] Google Cloud, “Precios de Cloud Storage”, 2022. [En línea]. Disponible en: [PreciosdeCloudStorage](#) [Acceso: 04 de abril 2023]
- [13] Google Cloud, “Cloud Tensor Processing Unit (TPU)”, 2022. [En línea]. Disponible en: <https://cloud.google.com/tpu/docs/tpus?hl=es-419> [Acceso: 04 de abril 2023]
- [14] TensorFlow, “TensorFlow API”, 2023. [En línea]. Disponible en: https://www.tensorflow.org/api_docs/python/tf [Acceso: 04 de abril 2023]
- [15] Keras, “Keras API reference”, 2023. [En línea]. Disponible en: <https://keras.io/api/> [Acceso: 04 de abril 2023]
- [16] ISIC Challenge, “ISIC 2019 Challenge”, 2019. [En línea]. Disponible en: <https://challenge.isic-archive.com/landing/2019/> [Acceso: 06 de abril 2023]
- [17] Tschandl P., Rosendahl C. & Kittler H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Sci. Data* 5, 180161 doi.10.1038/sdata.2018.161 (2018)
- [18] Noel C. F. Codella, David Gutman, M. Emre Celebi, Brian Helba, Michael A. Marchetti, Stephen W. Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, Allan Halpern: ”Skin Lesion Analysis Toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging

- (ISBI), Hosted by the International Skin Imaging Collaboration (ISIC)", 2017; arXiv:1710.05006.
- [19] Marc Combalia, Noel C. F. Codella, Veronica Rotemberg, Brian Helba, Veronica Vilaplana, Ofer Reiter, Allan C. Halpern, Susana Puig, Josep Malvehy: "BCN20000: Dermoscopic Lesions in the Wild", 2019; arXiv:1908.02288
- [20] International Skin Imaging Collaboration, "SIIM-ISIC 2020 Challenge Dataset", 2020. [En línea]. Disponible en: <https://doi.org/10.34970/2020-ds01> [Acceso: 06 de abril 2023]
- [21] Kaggle, "How to create TFRecord by Chris Deotte", 2020. [En línea]. Disponible en: <https://www.kaggle.com/code/cdeotte/how-to-create-tfrecords/notebook> [Acceso: 06 de abril 2023]
- [22] Kaggle, "ISIC 2019 TFRecords 256x256", 2020. [En línea]. Disponible en: <https://www.kaggle.com/cdeotte/isic2019-256x256> [Acceso: 06 de abril 2023]
- [23] Kaggle, "Melanoma TFRecords 256x256", 2020. [En línea]. Disponible en: <https://www.kaggle.com/datasets/cdeotte/melanoma-256x256> [Acceso: 06 de abril 2023]
- [24] Francisco J. Pulgar, Antonio J. Rivera, Francisco Charte, María J. del Jesus. "Análisis del impacto de datos desbalanceados en el rendimiento predictivo de redes neuronales convolucionales", Instituto Andaluz Interuniversitario en Data Science and Computational Intelligence (DaSCI) de la Universidad de Jaén, 2018.
- [25] Qishen Ha, Bo Liu, Fuxu Liu. "Identifying Melanoma Images using EfficientNet Ensemble: Winning Solution to the SIIM-ISIC Melanoma Classification Challenge", Cornell University, 2020.