

Universidad de La Habana  
Facultad de Matemática y Computación



# Aprendizaje automático orientado a la clasificación de cáncer de piel: un enfoque basado en EfficientNetB1

Autor:

**Deborah Famadas Rodríguez**

Tutores:

**Prof. Reinaldo Rodríguez Ramos\***

**Dr. Yudivian Almeida Cruz**

**Universidad de la Habana,**

**\*Universidade Federal Fluminense UFF  
Rio de Janeiro, Brasil**

Trabajo de Diploma  
presentado en opción al título de  
Licenciado en Ciencia de la Computación

Fecha

Diciembre 2023

[github.com/deborahfam/Thesis](https://github.com/deborahfam/Thesis)

A todas esas personitas hermosas que me ayudaron a llegar aquí. Este mérito es  
suyo.

# Agradecimientos

De esta forma concluyen 5 años de duro y constante trabajo y aprendizaje que no hubieran sido posible sin cada una de esas personas que estuvieron siempre apoyándome en cada paso que daba, que nunca me dejaron rendirme y que si hoy estoy donde estoy, es gracias a ellos. Esta sección es para agradecerles.

Bien saben mis padres que cuando pequeña no me despegaba un segundo de la computadora. Ese ímpetu por querer aprender cada vez más lo heredé de ellos. Por eso cuando tomé la decisión de coger una de las carreras más desafiantes en el mundo de la ciencia, no dudaron ni por un segundo en darme su apollo incondicional. No fue fácil la trayectoria, viniendo de una provincia no necesariamente céntrica, teniendo que estar becada en el 'Bahía', pasar noches sin dormir y alejada de la familia, cada día seguía recibiendo ese mensaje de mi mamá: 'Cuqui, cómo va todo?' que me sacaba del estrés momentáneo y volvía a experimentar la tranquilidad. A mi mamá, que siempre luchó por mí, incluso cuando ni me lo merecía, incluso ni cuando las fuerzas le daban, que me ha hecho la mujer que soy y la que seré, le regalo esto que hoy construí y mucho más.

De las abuelas se dice que son las que consienten a los nietos. Yo fui la primera de cuatro, su 'negrita bonita' (ni me pregunten por qué porque ni sé) y la más consentida. A mi 'Aya' le dedico esto. Por siempre decirme 'Pero tú eres inteligente, eso tú lo apruebas' lo mismo si le hablaba de Análisis matemático de 2 variables a si le hablaba de 'Educación física'. Ella fue la primera que vio en mí, lo que nadie, siempre sonriente, una sonrisa con algunas ventanas, pero con dirección al alma.

Esta tesis tiene un alo mágico, y es que un angelito me ha estado velando todo el tiempo. 'Eo', nombre que mi subconsciente de pequeña le asignó a mi abuelo, fue y será parte de mí siempre. Nunca seré capaz de olvidar su risa 'kkkk' mientras me jalaba las orejas y me decía 'Ere una caballaaaa'. Él también se merece todo.

A mi padre, que sabe amar a su forma única. El primero que me sentó delante de una computadora, que me cargaba a caballito y que al día de hoy sigue luchando por que yo salga adelante. A mis tíos, tías, primas, etc, con especial mención a mi tío 'Rene' que fue como un segundo padre para mí, y siempre estuvo cuando lo necesitábamos.

Y continuando el tema de los agradecimientos familiares. Sería incorrecto de mí

decir 'Le quiero agradecer a mis amigos', cuando son familia prácticamente.

Alguna vez se han fijado que somos muy poco agradecidos con el dedo pequeño del pie. Ese dedo que te ayuda a identificar obstáculos, incluso si tus ojos no pueden verlos (sobre todo si tus ojos no pueden verlos). Que te da estabilidad y soporte, que es gordito y aunque tiene 4 dedos más a los lados sigue siendo diferente. Yo tengo una persona así en mi vida, el 'Pocho' o como la gente lo llama 'Jean Pierre'. Es inconcluso de mi parte dedicarte un texto que no se empape en las lágrimas que derramaré si lo escribo, dado que solo tú y yo sabemos que significas. Te agradezco por tantas noches, tantas mañanas y tantas tardes, que no solo fuiste y eres amigo, mentor, el contacto más contactado y más importante de mi whatsapp, fuiste y eres TODO. Este mérito es completamente tuyo, esto y la libreta de discreta que tengo en la casa que a ver cuándo te la devuelvo jjj.

A mi otra familia, Omar y Andy, que me apadrinaron desde primer año, que me han visto crecer y yo a ellos, que me dieron el primer cocotazo de la carrera y que hacen muchas veces mejor mis días. A David, porque no me lo perdonaría nunca si no te dedico unas líneas, contigo también fueron noches, incansables e incesantes de puro estrés al recibir por cuarta vez de mi la frase 'Ay que no te entiendo' y tú, tan calmado y confiado como siempre volver a la carga, siempre creyéndome capaz, sino como un hombre tan ocupado como tú hubiera dedicado tanto a una causa perdida jjj. A tí que fue muy importante ese abrazo luego de esa prueba de Eda, para tí también va esto.

Comenta la gente que uno no se debe enamorar de sus mejores amigos, que rompe la amistad y que luego terminas arrepintiéndote. Pero una no puede cohibirse ante lo inevitable. Si bien besaste tú primero, no podría estar más agradecida al respecto. Y es que no hay otra voz más dulce que pueda calmar mi tormenta. Ay gaby, que más podría decirte que no haya dicho ya, que siempre estuviste ahí, a pesar que casa de lía te daba alergia, a pesar que solo tenías una muda de ropa, a pesar de todo, ibas, y solo el hecho de tenerte ahí era suficiente, es suficiente. Esto tampoco hubiera sido posible sin tí, y aunque lo dudes intensamente, te lo digo de corazón.

Y quiero agradecer a todos, a mis tutores y mis profesores por llevarme a ser la profesional que soy hoy en día, al Dr. Jose Ignacio por sus consideraciones y consejos sobre la tesis que fueron de gran ayuda. A mis amigos del pre, por decirme 'Si hija coge eso, si al final, loca ya estás', a los amigos de la carrera, a Lachi, Alejandra, Roxana, Ana y Toni por sacarme de mi algarabía y disfrutar conmigo en las fiestas, por reírse de mis boberías en las conferencias y por siempre estar igual de estresados que yo con los exámenes. A mis suegros, que tan solo 10 meses en mi vida, mira que si han causado hueco sentimental. A Y a todos aquellos que por falta de espacio (No me puedo pasar de 100 páginas en la tesis) o de memoria me ayudaron a lograr esto que hoy consigo.

# Opinión del tutor

El trabajo “Aprendizaje automático orientado a la clasificación de cáncer de piel: un enfoque basado en EfficientNetB1” presentado por la estudiante Deborah Famadas Rodríguez constituye un resultado notable en el campo de la biomedicina, bioinformática. En este trabajo de diploma se presenta una metodología para la clasificación automática del cáncer de piel mediante el uso de técnicas de aprendizaje profundo. La solución propuesta se basa en el uso de los pesos obtenidos de una red neuronal convolucional pre-entrenada, EfficientNetB1. Este proceso se beneficia del aprendizaje de grandes conjuntos de datos y experiencias previas para mejorar el reconocimiento de patrones en imágenes dermatológicas ya adquirido por la red convolucional. Luego se hace uso de un algoritmo propio de learning rate para ajustar el aprendizaje del modelo.

Para la implementación de este marco de trabajo, el diplomante debió asimilar un volumen considerable de información, de disímiles campos, entre ellos el referente al aprendizaje profundo establecido como una importante subdisciplina dentro del aprendizaje automático, especialmente en áreas relacionadas con la percepción visual humana. Esta metodología procesa datos a través de múltiples capas que incluyen estructuras complejas y transformaciones no lineales. Actualmente ha logrado avances significativos en áreas como la visión artificial, el reconocimiento de voz, el procesamiento del lenguaje natural, el reconocimiento de audio y la bioinformática. El aprendizaje profundo se ha reconocido como uno de los diez avances tecnológicos más significativos, dadas sus amplias aplicaciones potenciales en el análisis de datos. El enfoque del aprendizaje profundo abstrae los datos en distintos niveles, lo que permite su aplicación en tareas complejas como la detección de objetos y la clasificación. Su capacidad para reemplazar la extracción manual de características por algoritmos eficientes de aprendizaje ya sea no supervisada o semi-supervisada, ha revolucionado múltiples áreas. Esta revolución incluye el campo de la atención médica, donde la gestión y análisis de un volumen abrumador de datos médicos es un desafío crucial.

El trabajo escrito presenta una estructura clara y organizada que permite fácil comprensión de los contenidos incluidos. Además de presentar los resultados obtenidos en su trabajo, el diplomante presenta elementos técnicos acerca de aprendizaje automático, otros elementos referentes a la programación como un grupo de técnicas

y algoritmos tanto de las ramas de inteligencia artificial como del campo de la optimización, los cuales utiliza en el desarrollo de su trabajo, lo cual hacen del documento escrito un buen material de referencia para los futuros trabajos relacionados con el tema.

La diplomante ha trabajado con gran dedicación durante toda su trayectoria, lo caracterizan su constancia y dedicación al trabajo de investigación. Es una excelente estudiante con gran pasión por la investigación. Debe destacarse que tuvo que asimilar en un período muy corto todo un número de conceptos, definiciones, referentes a temas de la tesis, manejar una bibliografía compleja. Además, fue capaz de manera convincente y resuelta de enfrentar las dificultades con independencia, de manera creativa.

Por todo lo antes expuesto, le propongo al tribunal la evaluación del presente trabajo de excelente y que le sea concedida a Deborah Famadas Rodríguez el título de licenciado en Ciencia de la Computación.

**Tutores:** Dr. Reinaldo Rodríguez Ramos, Dr Yudivian Almeida Cruz.

# Resumen

En este trabajo de diploma se presenta una metodología para la clasificación automática del cáncer de piel mediante el uso de técnicas de aprendizaje profundo. La solución propuesta se basa en el uso de los pesos obtenidos de una red neuronal convolucional pre-entrenada, EfficientNetB1. Este proceso se beneficia del aprendizaje de grandes conjuntos de datos y experiencias previas para mejorar el reconocimiento de patrones en imágenes dermatológicas ya adquirido por la red convolucional. Luego se hace uso de un algoritmo propio de *learning rate* para ajustar el aprendizaje del modelo. Para reforzar la eficiencia, se incorporan capas de normalización, densas, de regularización y *dropout*, además de una capa de salida. Además, se realizan una serie de experimentos donde se evalúa el impacto de la división y normalización de datos con respecto a la eficiencia de clasificación del algoritmo.

# Abstract

In this diploma work, a methodology for automatic skin cancer classification using deep learning techniques is presented. The proposed solution is based on the use of weights obtained from a pre-trained convolutional neural network, EfficientNetB1. This process benefits from the learning of large datasets and previous experiences to improve the pattern recognition in dermatological images already acquired by the convolutional network. A proprietary learning rate algorithm is then used to adjust the model learning. To reinforce efficiency, normalization, dense, regularization and dropout layers are incorporated, as well as an output layer. In conjunction with this, a series of experiments are carried out, where the impact of the division and normalization with respect to the classification efficiency of the algorithm is evaluated.



# Índice general

|  |           |
|--|-----------|
| <b>Introducción</b>  | <b>1</b>  |
| <b>1. Estado del Arte</b>  | <b>5</b>  |
| 1.1. Teledermatología . . . . .  | 5         |
| 1.2. Técnicas tempranas de clasificación de imágenes . . . . .                         | 6         |
| 1.2.1. Clasificación y reconocimiento de patrones en imágenes médicas . . . . .        | 7         |
| 1.3. Introducción y desarrollo de <i>machine learning</i> en el campo médico . . . . . | 8         |
| 1.3.1. Análisis de imágenes médicas . . . . .  | 8         |
| 1.4. Datasets de cáncer de piel . . . . .  | 9         |
| 1.5. Redes neuronales convolucionales en la medicina . . . . .                         | 10        |
| 1.6. Trabajos basados en EfficientNet . . . . .  | 13        |
| 1.7. Conclusión del estado del arte . . . . .  | 13        |
| <b>2. Propuesta de solución</b>  | <b>15</b> |
| 2.1. Desarrollo del modelo y estrategias de optimización . . . . .                     | 15        |
| 2.1.1. Arquitectura del modelo EfficientNet . . . . .                                  | 16        |
| 2.2. Diseño y entrenamiento del modelo . . . . .                                       | 18        |
| 2.3. Ajuste dinámico del learning rate . . . . .                                       | 18        |
| <b>3. Detalles de implementación y experimentos</b>                                    | <b>21</b> |
| 3.1. Fuente de datos . . . . .   | 23        |
| 3.1.1. Transformación de datos . . . . .   | 24        |
| 3.1.2. Modelación y división del conjunto de datos . . . . .                           | 24        |
| 3.1.3. Generadores de datos y preprocesamiento . . . . .                               | 25        |
| 3.2. Capas adicionales y regularización . . . . .                                      | 27        |
| 3.2.1. Normalización por lotes . . . . .   | 27        |
| 3.2.2. Capa densa . . . . .  | 28        |
| 3.2.3. Dropout . . . . .   | 28        |
| 3.2.4. Capa de salida . . . . .  | 28        |
| 3.2.5. Optimización . . . . .  | 28        |
| 3.3. Experimentos . . . . .  | 29        |

|                        |   |           |
|------------------------|---|-----------|
| 3.3.1.                 | Experimento 1: Evaluación de la eficiencia de la división asimétrica de datos en la clasificación de imágenes de cáncer de piel . . . . . | 29        |
| 3.3.2.                 | Experimento 2: Análisis de la estratificación de datos en la clasificación de imágenes de cáncer de piel . . . . .                        | 33        |
| 3.4.                   | Discusión . . . . .   | 37        |
| 3.4.1.                 | Observaciones generales . . . . .   | 39        |
| 3.4.2.                 | Consideraciones finales . . . . .   | 40        |
| <b>Conclusiones</b>    |   | <b>43</b> |
| <b>Recomendaciones</b> |   | <b>45</b> |
| <b>Bibliografía</b>    |   | <b>47</b> |

# Índice de figuras

|   |    |
|---|----|
| 2.1. Estadísticas del rendimiento de los modelos de EfficientNet [34]. . . .                      | 17 |
| 2.2. Diagrama del modelo propuesto. . . . .   | 18 |
| 2.3. Diagrama del ajuste dinámico de la tasa de aprendizaje. . . . .                              | 19 |
| 3.1. Diagrama de división de datos. . . . .   | 25 |
| 3.2. Curva de aprendizaje a lo largo del proceso de entrenamiento del ex-<br>perimento 1. . . . . | 32 |
| 3.3. Matriz de confusión del experimento 1. . . . .   | 33 |
| 3.4. Gráfico de errores por clase en el conjunto de pruebas . . . . .                             | 33 |
| 3.5. Curva de aprendizaje a lo largo del proceso de entrenamiento del ex-<br>perimento 2. . . . . | 36 |
| 3.6. Matriz de confusión del experimento 2. . . . .   | 36 |
| 3.7. Gráfico de errores por clase en el conjunto de pruebas. . . . .                              | 37 |

# Índice de tablas

|   |    |
|---|----|
| 2.1. Descripción de términos clave en el entrenamiento de modelos de aprendizaje automático. . . . .                          | 20 |
| 3.1. Distribución de imágenes por categoría diagnóstica. . . . .  | 23 |
| 3.2. Datos transformados a formato categórico. . . . .  | 24 |
| 3.3. Parámetros de aumento de datos. . . . .  | 26 |
| 3.4. Distribución de los conjuntos de entrenamiento, validación y prueba. .   | 29 |
| 3.5. Experimento 1: Distribución de imágenes de cáncer de piel en los conjuntos de entrenamiento, test y validación. . . . .  | 30 |
| 3.6. Distribución de muestras por categoría después del sobre-muestreo. .   | 30 |
| 3.7. Distribución de muestras con peso asignado. . . . .  | 31 |
| 3.8. Estadísticas básicas del modelo del experimento 1. . . . .   | 31 |
| 3.9. Distribución de los conjuntos de entrenamiento, validación y prueba. .   | 34 |
| 3.10. Experimento 2: Distribución de imágenes de cáncer de piel en los conjuntos de entrenamiento, test y validación. . . . . | 34 |
| 3.11. Estadísticas básicas del modelo del experimento 2. . . . .  | 35 |
| 3.12. Informe de clasificación combinado para los Experimentos 1 y 2. . . .   | 39 |
| 3.13. Resultados del entrenamiento del experimento 1. . . . .   | 41 |
| 3.14. Resultados del entrenamiento del experimento 2. . . . .   | 42 |

# Introducción

Desde su surgimiento en 2006, el aprendizaje profundo se ha establecido como una importante subdisciplina dentro del aprendizaje automático, especialmente en áreas relacionadas con la percepción visual humana. Esta metodología procesa datos a través de múltiples capas que incluyen estructuras complejas y transformaciones no lineales [1]. Actualmente ha logrado avances significativos en áreas como la visión artificial, el reconocimiento de voz, el procesamiento del lenguaje natural, el reconocimiento de audio y la bioinformática [2]. Desde 2013, el aprendizaje profundo se ha reconocido como uno de los diez avances tecnológicos más significativos, dadas sus amplias aplicaciones potenciales en el análisis de datos [3].

El enfoque del aprendizaje profundo abstrae los datos en distintos niveles, lo que permite su aplicación en tareas complejas como la detección de objetos y la clasificación. Su capacidad para reemplazar la extracción manual de características por algoritmos eficientes de aprendizaje, ya sea no supervisada o semi-supervisada, ha revolucionado múltiples áreas [4]. Esta revolución incluye el campo de la atención médica, donde la gestión y análisis de un volumen abrumador de datos médicos es un desafío crucial.

En el ámbito de la atención médica, especialmente en la dermatología, el aprendizaje profundo ha mostrado un potencial extraordinario. La dermatología, que se enfoca en el estudio y tratamiento de enfermedades de la piel, se enfrenta al desafío del cáncer de piel, el tipo de cáncer más frecuente a nivel mundial [5]. La detección precoz de este cáncer es vital [5], y aquí es donde el aprendizaje profundo, con su habilidad para analizar y clasificar imágenes médicas con precisión, juega un rol transformador. La integración de estas tecnologías en la práctica dermatológica no solo mejora la precisión diagnóstica, sino que también promete revolucionar el tratamiento y manejo de diversas afecciones cutáneas [6].

## Motivación

El uso de algoritmos de *machine learning* (ML) para el diagnóstico del cáncer de piel puede resultar una herramienta valiosa. A diferencia de otros tipos de cáncer, el

cáncer de piel se forma en la superficie de la piel y suele ser visible. Esto plantea una oportunidad única para la detección temprana y el tratamiento, lo cual es esencial, ya que la mayoría de los casos de cáncer de piel son tratables si se detectan a tiempo [5].

Estos algoritmos pueden identificar patrones complejos con una precisión y consistencia mayor que los métodos de diagnóstico humano, reduciendo así la posibilidad de diagnósticos incorrectos debido a la interpretación subjetiva y variable de los expertos [7]. Además, el ML puede procesar grandes volúmenes de datos rápidamente y su capacidad para aprender y adaptarse con el tiempo significa que la detección y clasificación del cáncer de piel puede mejorar continuamente [7].

Aunque ya existen algoritmos eficientes para la clasificación de melanomas, una forma de cáncer de piel, el desarrollo de un modelo capaz de clasificar varios tipos de cáncer de piel y generalizar entre ellos es todavía una problemática abierta. Esto podría ampliar el alcance de las imágenes dermatológicas procesables, mejorando potencialmente la precisión en la detección y el tratamiento de distintas formas de cáncer de piel.

## Antecedentes

El desarrollo de las Redes Neuronales Convolucionales (CNN) ha sido fundamental para la identificación de características en imágenes médicas, una base sobre la cual se construye la motivación actual para aplicar ML en la dermatología. Su uso en medicina ha demostrado ser eficaz para capturar patrones específicos en datos de imágenes con alta precisión [8].

Las universidades han desempeñado un papel crucial en la investigación y el desarrollo de tecnologías avanzadas en el campo de la medicina, especialmente en la detección y tratamiento del cáncer. Estos centros académicos no solo proporcionan una base sólida para la investigación teórica, sino que también fomentan la innovación práctica mediante el uso de tecnologías emergentes como el aprendizaje automático y la inteligencia artificial. En particular, nuestra universidad ha contribuido significativamente a este campo.

Un claro ejemplo de esta contribución es la tesis de Darien Viera Barredo titulada *Autómata celular estocástico en redes complejas para el estudio de la invasión, migración y metástasis del cancer* [9] proporciona un marco detallado sobre cómo se aborda el estudio del cáncer desde una perspectiva matemática y computacional avanzada. El modelo propuesto utiliza autómatas celulares estocásticos para simular el crecimiento avascular y vascular del tumor. En el se aborda la complejidad del ciclo vital tumoral, destacando la importancia de su comprensión tanto para la investigación del cáncer como para la salud pública. Tradicionalmente, la modelación matemática y computacional se ha centrado en las etapas tempranas del desarrollo

tumoral, donde la mortalidad es baja. Sin embargo, este estudio se enfoca en las fases avanzadas, que son críticas para la vida del paciente.

Complementando esta línea de investigación, la tesis reciente de Claudia Olavarrieta Martínez [10] propone un *ensemble* de redes neuronales para clasificar imágenes dermatoscópicas en cuatro categorías: melanoma, carcinoma basocelular, carcinoma espinocelular y otros utilizando la técnica de transferencia de conocimientos en redes como VGG16, ResNet50 y EfficientNet B0.

## Problemática

A pesar de los avances significativos en el campo del aprendizaje profundo y su aplicación en la detección y tratamiento del cáncer, como se evidencia en las investigaciones realizadas, aún persisten desafíos fundamentales. Uno de los principales retos es la necesidad de mejorar aún más la precisión y la eficiencia en el diagnóstico del cáncer de piel. Aunque los modelos actuales han mostrado resultados prometedores, la diferenciación precisa entre diversas formas de cáncer de piel sigue siendo compleja, especialmente en etapas tempranas o en casos atípicos. Además, la mayoría de los modelos existentes se centran en la clasificación de melanomas, dejando de lado otras formas de cáncer de piel. Entonces **¿es posible desarrollar un modelo avanzado de aprendizaje profundo que mejore significativamente la precisión y eficiencia en el diagnóstico de diversas formas de cáncer de piel, incluyendo tanto melanomas como otros tipos menos comunes, especialmente en etapas tempranas o en casos atípicos?** Se supone que, mediante la implementación de técnicas avanzadas de aprendizaje profundo y el análisis de un conjunto de datos más amplio y diversificado, se puede desarrollar un modelo que no solo clasifique con mayor precisión los melanomas, sino que también sea eficiente en la identificación de otras formas de cáncer de piel. Este modelo podría superar las limitaciones actuales y proporcionar un diagnóstico más preciso y temprano, lo que resultaría en un tratamiento más efectivo y una mejor tasa de supervivencia para los pacientes. Por lo tanto, el desarrollo de un modelo capaz de clasificar varios tipos de cáncer de piel y generalizar entre ellos es un objetivo crucial. Esto ampliaría el alcance de las imágenes dermatológicas procesables, mejorando potencialmente la precisión en la detección y el tratamiento.

## Objetivos

Este trabajo propone como objetivo fundamental el diseño y validación de un modelo predictivo basado en *deep learning* para el diagnóstico del cáncer de piel mediante la clasificación de imágenes dermatológicas. El modelo se enfocará en clasificar

imágenes, priorizando tanto la precisión de los resultados como la capacidad de generalización del modelo.

Entre los objetivos específicos del proyecto se encuentran:

1. Estudiar el estado del arte sobre las técnicas empleadas en el diagnóstico de imágenes dermatológicas y su efectividad.
2. Crear un modelo de *deep learning* que dado un conjunto de imágenes de cáncer de piel clasifique el tipo al que pertenecen.
3. Decidir, mediante una técnica de ML, la mejor distribución de datos para entrenamiento del modelo
4. Implementar mejoras potenciales al modelo y sus hiperparámetros para aumentar la precisión y generalización del mismo.
5. Implementar técnicas de validación para evaluar la precisión del modelo.

## Estructura de la tesis

El contenido de la tesis se organiza de la siguiente forma. En el capítulo 1 se exponen las principales alternativas presentes en la literatura que se han desarrollado para la clasificación de imágenes. En el capítulo 2 presenta el modelo propuesto para la implementación de un sistema de clasificación de imágenes de cáncer de piel. En el capítulo 3 se describen los algoritmos y técnicas utilizadas, los experimentos realizados y se exponen los resultados obtenidos y se analiza la efectividad de estos. Finalmente, se presentan las conclusiones de la tesis y las recomendaciones para investigaciones futuras.



# Capítulo 1

## Estado del Arte

En las últimas décadas, los avances en potencia computacional han permitido un progreso significativo en el análisis automatizado de imágenes. Se ha pasado del análisis básico de imágenes digitales a sofisticados algoritmos capaces de identificar patrones sutiles en las imágenes de lesiones cutáneas. Los progresos en el reconocimiento de melanomas a partir de imágenes dermatológicas, han demostrado que los sistemas automatizados pueden lograr un diagnóstico comparable al de los expertos humanos [11].

### 1.1. Teledermatología

La teledermatología, una rama emergente de la telemedicina, ha revolucionizado el campo de la atención dermatológica. Impulsada por el auge de las tecnologías digitales a finales del siglo XX, esta modalidad se ha consolidado como una herramienta vital en el diagnóstico y manejo de afecciones cutáneas. Desde la década de 2000, la teledermatología ha permitido consultas dermatológicas remotas, ampliando significativamente el alcance de los servicios de salud [12].

La investigación de Whited et al., en 2002, fue pionera en demostrar la efectividad de esta práctica [13]. El estudio resaltó una reducción notable en los tiempos de respuesta, con una mediana de 5 días para consultas teledermatológicas, en contraste con los 28 días de los métodos tradicionales. Además, se enfatizó su utilidad en casos urgentes o semi-urgentes.

Paralelamente, la teledermatopatología ha mostrado su potencial, ofreciendo una fiabilidad comparable a la evaluación histológica tradicional [12]. Uno de los usos importantes de esta técnica fue detectar cáncer de piel. Un estudio, Piccolo et al., [14] publicado en 2002, se centró en la concordancia entre los diagnósticos telepatológicos e histopatológicos convencionales, indicando contribuciones significativas en el campo de la teledermatopatología. Por término medio, el 78% de los telediagnósticos fueron

correctos (intervalo, 60%-95%), mientras que el 85% de los diagnósticos convencionales fueron correctos (intervalo, 60%-95%). Se obtuvo una concordancia diagnóstica perfecta en 7 (35%) de los 20 casos, y sólo se identificó una diferencia significativa en 1 caso.

Este avance en la teledermatología ha sentado las bases para la siguiente etapa en la medicina digital: el desarrollo de algoritmos computarizados que puedan detectar características en las imágenes, imitando el comportamiento humano, para luego a partir de estos obtener resultados.

## 1.2. Técnicas tempranas de clasificación de imágenes

Cuando se trata de la clasificación de imágenes, es esencial considerar que nuestro sistema visual humano (SVH) primero recibe las ondas electromagnéticas que pertenecen al espectro visible y luego las interpreta en el cerebro. Sin embargo, en el campo de la visión artificial, cuando introducimos una imagen en un ordenador, lo que se interpreta es una matriz de números generalmente en el rango de  $[0,255]$  y con tres dimensiones en caso de que sea una imagen a color (RGB). Como resultado, se puede notar una gran brecha entre el significado semántico de la clase asociada a una imagen y los valores de píxeles que la componen, lo que hace que la tarea de clasificación sea compleja para sistemas artificiales [8].

El reconocimiento de imágenes es una faceta de la inteligencia artificial que posibilita que los sistemas informáticos analicen e interpreten el contenido visual en imágenes. Esto se consigue detectando patrones y características distintivas que luego se emplean para clasificar y etiquetar objetos. Su propósito es automatizar el análisis visual, optimizando tiempo y recursos en diversas aplicaciones [15]. Esta tecnología se sustenta en algoritmos de aprendizaje automático, que entrenan a las máquinas para reconocer patrones visuales. Mediante el uso de bases de datos de imágenes etiquetadas para el entrenamiento, los algoritmos aprenden a identificar objetos y patrones. Una vez completado el entrenamiento, el modelo puede identificar automáticamente estos elementos en nuevas imágenes.

Desde finales del siglo XX, los ingenieros han dedicado esfuerzos significativos al desarrollo de técnicas y algoritmos para la categorización y reconocimiento de eventos a través de datos. Inicialmente, esto se centró en el procesamiento de texto para la clasificación automática de documentos, seguido por el tratamiento de sonidos e imágenes en diversos formatos.

Un hito importante fue la publicación de un artículo sobre reconocimiento de patrones en 1974 en *IEEE Transactions on Automatic Control* [16], evidenciando que ya en la década de 1970 se estaban implementando estas técnicas en el reconocimiento

de patrones. Los trabajos de este enfatizaron avances teóricos y experimentales significativos que impulsaron el progreso en el reconocimiento automático de patrones y el aprendizaje automático.

### 1.2.1. Clasificación y reconocimiento de patrones en imágenes médicas

En el ámbito de la medicina, se ha observado que la mayoría de los sistemas artificiales de diagnóstico, en un punto, toman decisiones que están cada vez menos relacionadas con la apariencia física de la imagen tal como la vería un radiólogo. En su lugar, estos sistemas se basan en los detalles del patrón matemático de las características físicas individuales de la imagen, que son extraídas por un sistema de visión artificial o un radiólogo, para tomar su decisión final. Estos patrones matemáticos han sido objeto de estudio durante décadas por científicos que han utilizado diversos métodos analíticos, incluyendo las redes neuronales [8].

Para abordar esta brecha entre la representación de la imagen y su significado, se han desarrollado varios tipos de algoritmos de clasificación. Algunos de estos algoritmos se basan en la detección de bordes, como el algoritmo de Canny [17]. Sin embargo, estos algoritmos son robustos cuando se trata de identificar una clase específica, pero si se desea clasificar una clase diferente, es necesario crear un nuevo modelo desde cero [18].

Esta serie de limitaciones restringieron su capacidad para abordar tareas complejas y desafiantes. La escasez de datos adecuados, la falta de recursos computacionales avanzados, arquitecturas simples, dificultades en el entrenamiento, generalización limitada, problemas de gradiente, falta de interpretabilidad y largos tiempos de entrenamiento fueron obstáculos clave en su desarrollo inicial.

Técnicas utilizadas en los inicios para la clasificación de patrones rompieron con algunas de las barreras de desarrollo: histograma de gradientes orientados (HOG) [19], Scale-Invariant Feature Transform (SIFT) [20], Binary Robust Independent Elementary Features (BRIEF) [21], Color Histograms [22], entre otras. En la bibliografía se encuentran métodos de bajo nivel como son segmentación de la imagen por niveles grises, bordes o formas, entre otros y métodos de alto nivel como clasificadores basados en redes neuronales, máquinas de soporte vectorial (SVM), árbol de decisiones, entre otros [23].

Se destaca, además, que las tres técnicas más usadas en la clasificación automática de imágenes son árboles de decisiones, redes neuronales y máquinas de vectores de soporte siendo las redes neuronales una de las más utilizadas en campo del aprendizaje profundo [23].

## 1.3. Introducción y desarrollo de *machine learning* en el campo médico

Las primeras aplicaciones de redes neuronales en imágenes médicas se orientaron hacia el análisis y clasificación de dichas imágenes para apoyar en diagnósticos y tratamientos.

### 1.3.1. Análisis de imágenes médicas

El análisis de imágenes médicas mediante redes neuronales se ha enfocado en campos de la medicina como resonancia magnética, medicina nuclear y radiología, permitiendo la identificación y clasificación de patologías o condiciones específicas. Además, estas tecnologías han encontrado aplicaciones en áreas como la oftalmología, para el diagnóstico de enfermedades oculares a partir de imágenes de retina, y en la cardiología, para la evaluación de imágenes de ecocardiogramas [8].

**Resonancia Magnética:** En el contexto de la esclerosis múltiple, se han explorado soluciones de segmentación basadas en redes neuronales convolucionales (CNNs) para segmentaciones rápidas y fiables de lesiones y estructuras de materia gris en imágenes de resonancia magnética multimodal [24].

**Medicina Nuclear:** En este campo un estudio demuestra cómo el aprendizaje profundo puede restaurar la calidad de imagen diagnóstica y mantener la precisión de la cuantificación de SUV para exploraciones PET con un conteo reducido, lo que podría aumentar la seguridad y reducir el costo de las imágenes PET [25].

**Radiología:** En radiología, la aplicación de la inteligencia artificial en el análisis de imágenes de cáncer, con un enfoque en radiomía y representaciones derivadas del aprendizaje profundo, y su uso para el soporte de decisiones en la gestión del cáncer [26].

Esto muestra que el análisis de imágenes médicas a través de redes neuronales representa un avance significativo en diversas áreas de la medicina. Las aplicaciones van desde la identificación de patologías en resonancia magnética, medicina nuclear y radiología, hasta el diagnóstico de enfermedades oculares y la evaluación cardíaca. Estas tecnologías no solo mejoran la precisión en la detección y clasificación de condiciones específicas, sino que también optimizan la eficiencia de los procesos diagnósticos y terapéuticos, destacando el papel crucial de la inteligencia artificial en el futuro de la medicina.

Teniendo en cuenta utilizando la inteligencia artificial se puede procesar grandes cantidades de datos, resulta una opción inteligente y acertada la recopilación de grandes volúmenes de datos (relacionados entre sí) para el procesamiento de los mismos con alguna herramienta de ML.

## 1.4. Datasets de cáncer de piel

En el campo de la dermatología, la generación de imágenes clínicas y dermatoscópicas es una práctica común para supervisar los cambios en las condiciones de la piel. Estas imágenes se han vuelto un recurso crucial para el avance de algoritmos de aprendizaje automático, especialmente en el desarrollo de Redes Neuronales Convolucionales (CNN). Los datasets son esenciales en ML porque proporcionan la base sobre la cual los algoritmos aprenden, se prueban, se comparan y evolucionan, permitiendo así la aplicación práctica y la innovación continua en este campo. Existen varios conjuntos de datos accesibles para la investigación en este ámbito.

Wu et al., recoge en su estudio un conjunto de los dataset de imágenes dermatológicas más utilizados en algoritmos de clasificación que se exponen a continuación:

- El PH2 Dataset, compuesto por 200 imágenes dermoscópicas de tres tipos de enfermedades de la piel es ampliamente utilizado para probar algoritmos de diagnóstico de enfermedades de la piel, mostrando altas tasas de precisión en varios estudios [27].
- MED-NODE Dataset: Contiene 170 imágenes digitales de melanoma y nevus. Diversos métodos han mostrado resultados significativos de clasificación en este dataset [27].
- HAM10000 Dataset: Recopilado por ISIC, incluye 10,015 imágenes dermoscópicas de siete enfermedades representativas de lesiones cutáneas pigmentadas. Se utiliza ampliamente en estudios de clasificación de cáncer de piel, superando a menudo el nivel de diagnóstico de los dermatólogos [27].
- Derm7pt Dataset: Contiene aproximadamente 2,000 imágenes clínicas y de dermoscopia de enfermedades de la piel. Se ha utilizado para probar diversas redes multitarea y métodos de clasificación [27].
- BCN20000 Dataset: Compuesto por 5,583 lesiones cutáneas y 19,424 imágenes dermoscópicas. Se emplea comúnmente en tareas de clasificación y segmentación de cáncer de piel [27].

- ISIC Dataset: Con más de 13,000 imágenes dermoscópicas, se enfoca en la clasificación y segmentación del cáncer de piel, siendo utilizado en numerosos estudios y competencias [27].

Mientras que, entre otros autores Das et al., [7] también menciona otros conjuntos de datos relevantes incluyen el Atlas Interactivo de Dermoscopia con 1000 ejemplos clínicos [7], la Biblioteca de Imágenes Dermofit con 1300 fotografías de alta resolución [7], el conjunto de datos de Asan [7], con 17,125 fotos clínicas, y el Hallym [7], con 125 fotos de casos de carcinoma basocelular. Además, los conjuntos de datos SD-198 y SD-260 [7] ofrecen una amplia gama de imágenes clínicas de diversas enfermedades de la piel. Dermnet NZ y Derm7pt [7] proporcionan colecciones extensas de fotografías clínicas, dermatoscópicas e histológicas, y The Cancer Genome Atlas [7] presenta una de las mayores colecciones de diapositivas de lesiones cutáneas patológicas.

Entre todos estos conjuntos de datos, el HAM10000 se destaca por su amplia utilización en la investigación del cáncer de piel. Este conjunto de datos es particularmente valioso debido a su extensa colección de imágenes de lesiones de piel, que incluye una variedad de tipos de cáncer de piel. Su uso generalizado en la comunidad científica y su relevancia en estudios recientes lo convierten en el dataset ideal para el desarrollo de esta tesis. La riqueza y diversidad de las imágenes en HAM10000 proporcionan una base sólida para entrenar y evaluar algoritmos de *machine learning*.

## 1.5. Redes neuronales convolucionales en la medicina

El avance en la clasificación y detección de enfermedades de la piel, impulsado por los conjuntos de datos como HAM10000, ha preparado el camino para aplicaciones más sofisticadas en el análisis de imágenes médicas. En este escenario, las Redes Neuronales Convolucionales (CNN) emergen como una herramienta esencial, aprovechando la profundidad y variedad de los datos disponibles. Estas redes son especialmente eficaces en tareas de reconocimiento y clasificación visual, gracias a su capacidad para procesar y aprender de grandes volúmenes de imágenes.

Una red neuronal convolucional está formada por diferentes capas, entre ellas las principales son las capas convolucionales, las capas de max-pooling, y las capas completamente conectadas. La capa convolucional tiene como objetivo realizar la convolución a la imagen de entrada, para extraer sus características. Realizar una convolución a una imagen, consiste en filtrar dicha imagen utilizando una máscara o ventana. La máscara se va desplazando por toda la imagen, multiplicándose de forma matricial [8].

En el campo de la medicina, estas facilitan la identificación de características relevantes en las imágenes médicas que pudieran ser indicativas de alguna condición

médica particular. Se aprovechan de estructuras específicas de datos, como imágenes, para capturar patrones con mayor precisión, reducir la carga computacional y mejorar la generalización y la interpretabilidad.

La evolución en la clasificación de lesiones cutáneas ha estado marcada por el uso innovador de redes neuronales convolucionales (CNNs). Una revisión sistemática en 2018 por Brinker et al., [28] analizó 13 artículos que implementaban CNNs para esta tarea, destacando su alto rendimiento. Los enfoques más comunes involucraron la reutilización de CNNs ya entrenadas con grandes conjuntos de datos, optimizadas posteriormente para la clasificación específica de lesiones cutáneas. Esta metodología, aunque efectiva, enfrentó retos como la dificultad de comparar distintos métodos debido a la variabilidad en los conjuntos de datos.

Posteriormente, las técnicas de aprendizaje automático, y en particular los modelos de aprendizaje profundo, emergieron como herramientas poderosas para el análisis de imágenes médicas. Un proyecto clave fue *A Deep Learning Approach to Skin Cancer Detection in Dermoscopy Images* [29], donde se utilizó un conjunto de 3400 imágenes dermatoscópicas del HAM10000, incluyendo lesiones melanoma y no melanoma. Se implementó una red neuronal convolucional profunda que procesaba imágenes directamente, identificando características valiosas sin necesidad de segmentación previa, lo cual representó un avance significativo en la simplificación y eficacia del proceso de clasificación.

En 2022, el estudio *Skin lesion classification of dermoscopic images using machine learning and convolutional neural network* [30], publicado en Nature, utilizó un subconjunto del HAM10000 para clasificar lesiones cutáneas mediante aprendizaje automático y CNN. Este enfoque ofreció resultados prometedores en la distinción entre lesiones malignas y benignas, logrando una precisión del 95,18% con el modelo CNN. La comparación con otros algoritmos de aprendizaje automático resaltó la superioridad de los modelos CNN en términos de precisión.

Tajerian , [31] presentó un enfoque metodológico para mejorar el diagnóstico de la lesiones cutáneas pigmentadas utilizando imágenes dermatoscópicas del conjunto de datos HAM10000. Este conjunto de datos se utilizó para analizar lesiones cutáneas pigmentadas. El modelo obtiene los mejores resultados en la detección de lesiones de nevos melanocíticos, con una puntuación F1 de 0,93.

Adegun et al., [32] recoge un conjunto de artículos que contribuyeron al desarrollo de algoritmos además de los anteriormente mencionados. Las referencias a los siguientes artículos están recogidas en el mismo [32]:

1. Majtner et al., usaron técnicas de CNN para la extracción de características y preprocesamiento, utilizando el conjunto de datos ISIC con 900 muestras de entrenamiento y 379 de prueba, clasificadas en benignas y malignas [32].
2. Vipin et al., implementaron un sistema de dos etapas, segmentación y clasifica-

ción, utilizando un conjunto de datos ISIC de 13,000 imágenes, reducido a 7,353 tras eliminar imágenes no utilizables [32].

3. Nasr-Esfahani et al., desarrollaron su propia CNN para preprocesar, extraer características y clasificar imágenes, con un conjunto de 170 imágenes no dermatoscópicas del UMCG, aumentado a 6,120 imágenes [32].
4. Attia et al., usaron una CNN completamente conectada con arquitecturas de autoencoder-decoder, alcanzando una precisión del 98% y una especificidad del 94% [32].
5. Mukherjee et al., desarrollaron una arquitectura CNN para la detección de lesiones malignas, logrando precisiones de 90.14% y 90.58% en los conjuntos de datos MEDNODE y Dermofit [32].
6. Sanketh et al., propusieron una CNN para la detección temprana de cáncer de piel, obteniendo un resultado óptimo del 98% [32].
7. Rahi et al., propusieron un modelo CNN con varias capas convolucionales y de agrupación máxima, logrando una precisión del 84.76% y una especificidad del 78.81% [32].
8. Gulati et al., emplearon redes preentrenadas como AlexNet y VGG16, obteniendo mejores resultados con VGG16 en modo de aprendizaje transferido [32].
9. Daghrir et al., combinaron una CNN con técnicas de aprendizaje automático clásicas, alcanzando una precisión individual del 85.5% con CNN [32].
10. Acosta et al., incorporaron técnicas de CNN basadas en máscaras y regiones con una estructura ResNet152 preentrenada, logrando una precisión del 90.4% y una especificidad del 92.5% [32].

El avance en la detección de cáncer mediante el uso de la inteligencia artificial (IA) y el aprendizaje automático (ML) ha sido significativo en las últimas décadas. El mismo ha demostrado un rendimiento excepcional en tareas de reconocimiento de imágenes, que es fundamental en la detección del cáncer de piel. Se llevó a cabo una revisión exhaustiva para evaluar el impacto de las técnicas de aprendizaje profundo en la detección precoz, en la que se analizaron diversos resultados de investigación y se presentaron mediante herramientas, gráficos, tablas y marcos para comprender mejor las técnicas predominantes en este campo [33].



## 1.6. Trabajos basados en EfficientNet

Uno de los modelos de redes convolucionales que ha tenido relevancia es EfficientNet. Este es una familia de modelos de red neuronal convolucional diseñada para la eficiencia, que significa que pueden lograr una mayor precisión con menos parámetros y computación. Es un novedoso método de escalado de modelos que utiliza un coeficiente compuesto sencillo pero muy eficaz para escalar las CNN de forma más estructurada [34].

Un estudio innovador relacionado con EfficientNet es el de Ali et al., [35]. Este desarrolló una cadena de procesamiento de imágenes previo al entrenamiento, que incluía la eliminación de cabellos en las imágenes, el aumento de datos y el redimensionamiento de las imágenes para cumplir con los requisitos de cada modelo de CNN. Utilizando transferencia de aprendizaje con pesos pre-entrenados de ImageNet y ajuste fino de las redes, se entrenaron variantes de EfficientNet (B0-B7) en el conjunto de datos HAM10000. El modelo más exitoso, EfficientNet B4, alcanzó una puntuación F1 y una precisión Top-1 del 87% y 87.91%, respectivamente, destacando que una complejidad intermedia del modelo puede ser óptima para este tipo de tareas. El modelo en cuestión relacionado con nuestro enfoque (EfficientNetB1) obtuvo una precisión del 86.5% y una precisión Top-1 del 86.5%.

Por otro lado Papiththira et al. [36] se centró específicamente en la detección de melanoma utilizando un enfoque basado en transferencia de aprendizaje profundo sin necesidad de preprocesamiento de imágenes. Este método se apoyó en el modelo EfficientNet pre-entrenado, complementado con un módulo de atención de canales para resaltar características específicas del melanoma en la clasificación. Evaluado en los conjuntos de datos UMGC y HAM10000, que incluyen imágenes clínicas y dermoscópicas, el enfoque propuesto superó los métodos del estado del arte con una precisión de clasificación del 84.12% y 96.32%, respectivamente.

## 1.7. Conclusión del estado del arte

Es notable un progreso en el análisis automatizado de imágenes, especialmente en el ámbito médico. Este avance ha sido impulsado significativamente por el desarrollo en la potencia computacional y la inteligencia artificial, lo que ha facilitado un diagnóstico más preciso y rápido de enfermedades. Paralelamente, el campo de la dermatología ha experimentado una revolución con la introducción de la teledermatología y teledermatopatología, mejorando la accesibilidad a servicios especializados y permitiendo diagnósticos rápidos y precisos a distancia, lo que es esencial para el tratamiento oportuno de enfermedades cutáneas.

Además, se ha destacado la importancia de los conjuntos de datos en el aprendizaje automático, como el HAM10000, que han sido cruciales en el desarrollo de

algoritmos en dermatología. Estos datasets proporcionan una base diversa y rica para el entrenamiento y evaluación de modelos, contribuyendo a avances significativos en la clasificación y detección de enfermedades de la piel. En términos de técnicas de clasificación de imágenes, se ha observado una evolución desde métodos iniciales como el procesamiento de bordes hasta el uso de algoritmos avanzados de aprendizaje automático y profundo. Las redes neuronales, en particular, han mostrado ser herramientas excepcionales para el análisis de imágenes médicas, aumentando la precisión y eficiencia en la identificación de patologías.

El papel de las redes neuronales convolucionales (CNN) en medicina ha sido fundamental, especialmente en dermatología, donde han posibilitado avances significativos en la clasificación de lesiones cutáneas. Los modelos basados en CNN han alcanzado una precisión comparable a la de especialistas humanos, lo que sugiere un futuro prometedor para la inteligencia artificial en el diagnóstico y tratamiento de enfermedades.

Sin embargo, a pesar de estos avances, todavía persisten desafíos como la variabilidad en los conjuntos de datos y la necesidad de mejorar la generalización de los modelos. Estos retos ofrecen oportunidades para futuras investigaciones y desarrollos, en particular en la mejora de algoritmos y en la integración de nuevas tecnologías en la práctica médica.

# Capítulo 2

## Propuesta de solución

Este capítulo introduce una propuesta para enfrentar el reto de clasificar de manera automática el cáncer de piel, empleando técnicas de aprendizaje profundo. La solución se centra en la utilización de una red neuronal profunda pre-entrenada, combinada con un algoritmo propio para el ajuste de precisión.

En el núcleo de la estrategia se encuentra el uso de una red neuronal convolucional pre-entrenada *EfficientNetB1*. Esta red, desarrollada a partir de extensos conjuntos de datos y experiencias previas, ofrece una base sólida y rica en características para nuestro modelo. Al aprovechar este pre-entrenamiento, se puede acelerar significativamente el proceso de aprendizaje del modelo, al tiempo que aumentamos su capacidad para generalizar y reconocer patrones complejos en las imágenes dermatoscópicas.

Para complementar el enfoque metodológico, se seleccionan un conjunto de herramientas tecnológicas avanzadas. Estas herramientas están diseñadas para optimizar el rendimiento del modelo, mejorar la precisión de la clasificación y garantizar una implementación efectiva. Entre ellas, destaca una capa de normalización, una capa densa, una de regularización, una de *dropout* y una de salida (capa densa con activación *softmax*). Estas técnicas le permiten al modelo afinar su capacidad de identificar con precisión las diferentes categorías de lesiones cutáneas.

### 2.1. Desarrollo del modelo y estrategias de optimización

Para abordar efectivamente el desafío de la detección de cáncer de piel, se seleccionó la arquitectura *EfficientNetB1* [37] como base del modelo. Esta red neuronal convolucional, parte de la familia EfficientNet, se caracteriza por su alta eficiencia y precisión. Utilizando un modelo pre-entrenado, se aprovecharon los pesos derivados de conjuntos de datos extensos, lo que facilitó la adaptación del modelo al conjunto

de datos específico [34].

### 2.1.1. Arquitectura del modelo EfficientNet

La familia de arquitecturas EfficientNet [34], surgió con el objetivo de hallar un método adecuado para escalar las CNNs de manera que mejoraran tanto en precisión (i.e., rendimiento del modelo) como en eficiencia (es decir, en términos de parámetros del modelo y FLOPS).

Inicialmente, el enfoque se centraba en aumentar la profundidad o el ancho de la red. Sin embargo, este método de escalado único tenía limitaciones y no se comprendía completamente. La investigación de EfficientNet reveló que un escalado uniforme en todas las dimensiones de la red (profundidad, ancho y resolución) usando un conjunto de coeficientes de escalado fijos podría lograr un mejor equilibrio y rendimiento. Este enfoque innovador, denominado escalado compuesto, aborda el problema del escalado en ConvNets de manera integral. Los resultados empíricos mostraron que escalar cualquier dimensión de la red mejora la precisión, pero el beneficio disminuye en modelos más grandes. Por lo tanto, se propuso un método de escalado que coordina y equilibra estas dimensiones, en lugar de escalar una sola dimensión a la vez.

EfficientNet comenzó con la creación de un modelo base, EfficientNet-B0, utilizando una búsqueda de arquitectura neural multiobjetivo que optimizaba tanto la precisión como los FLOPS. Esta red base se caracterizaba por su bloque principal, el cuello de botella móvil invertido *MBConv*, y la optimización de compresión y excitación. El proceso de escalado de EfficientNet se realizó en dos etapas principales. Primero, se fijaron ciertos parámetros y luego se escaló la red base para obtener variantes más grandes, desde EfficientNet-B1 hasta B7, utilizando el método de escalado compuesto. Mientras que la arquitectura EfficientNet B0 tiene 5,3 millones de parámetros y acepta imágenes de entrada de  $224 \times 224$ , EfficientNet B7 cuenta con 66 millones de parámetros y acepta imágenes de  $600 \times 600$  [38].

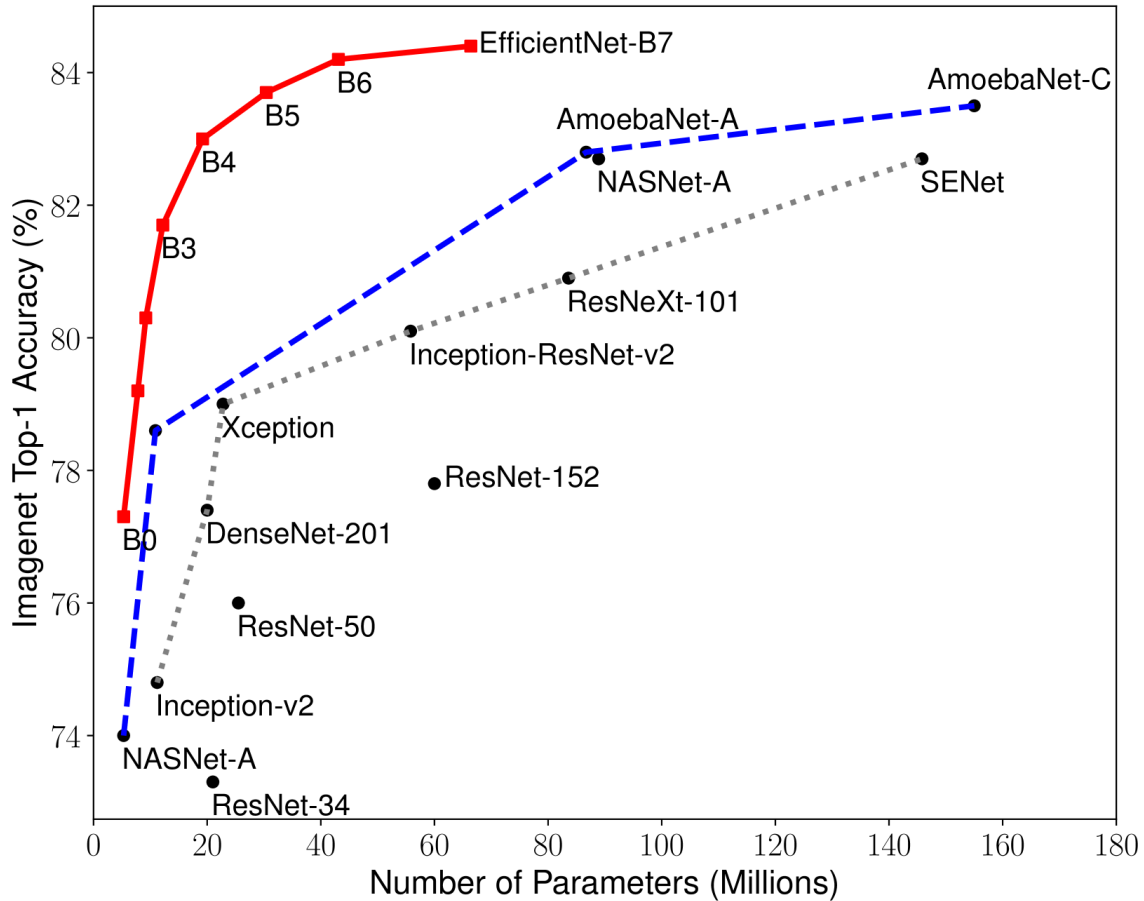


Figura 2.1: Estadísticas del rendimiento de los modelos de EfficientNet [34].

Los experimentos demostraron que el método de escalado compuesto mejoraba la precisión en modelos ya existentes como MobileNets y ResNet. Los modelos EfficientNet entrenados en ImageNet mostraron una precisión y eficiencia significativamente mayores en comparación con otros ConvNets, utilizando una cantidad mucho menor de parámetros y FLOPS. Especialmente, EfficientNet-B7 logró una precisión del 84.3% en top-1 en ImageNet, superando modelos anteriores y siendo considerablemente más pequeño y rápido.

La implementación del EfficientNetB1 puede ser particularmente beneficiosa para el análisis de datos. Este entre las variantes de la serie EfficientNet, se encuentra en un punto medio en términos de complejidad y tamaño, ofreciendo un equilibrio entre precisión y eficiencia computacional.

## 2.2. Diseño y entrenamiento del modelo

El modelo *EfficientNetB1* se carga pre-entrenado con pesos de *ImageNet* [39], una amplia base de datos de imágenes ampliamente utilizada para entrenamiento y *benchmarking* en visión por computadora. Luego a este se le omitió la capa superior para permitir la incorporación y personalización de capas adicionales.

Luego que *efficientnet* extrae las características relevantes se obtiene un vector de características. Este vector se normaliza mediante la capa de *Batch Normalization*, ajustando los valores a una escala manejable y estabilizando el entrenamiento. El vector normalizado se introduce en una capa densa. En esta capa cada neurona calcula una suma ponderada de sus entradas, aplica regularizadores si están activos, y luego utiliza la función de activación *ReLU*, que convierte los valores negativos a 0 y mantiene los positivos. Durante el entrenamiento, se aplica *dropout*, desactivando aleatoriamente una cantidad de neuronas en esta capa para prevenir el sobre-ajuste. La salida de la capa *Dropout* pasa a través de una capa densa final con 7 neuronas, una por cada clase para luego sus salidas sean convertidas en probabilidades. Luego se calcula el error comparando esta etiqueta con las probabilidades obtenidas con la salida del modelo. Ya por último el optimizador utilizado ajusta los pesos del modelo basándose en este error y con los pesos ajustados, la siguiente iteración comienza con una nueva imagen de entrada, repitiendo el proceso.

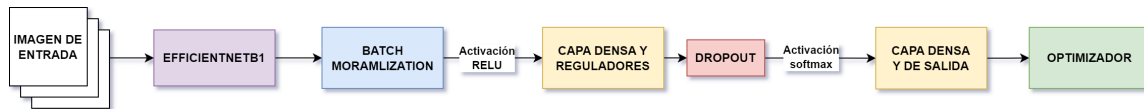


Figura 2.2: Diagrama del modelo propuesto.

## 2.3. Ajuste dinámico del learning rate

Un elemento innovador del entrenamiento fue el uso de un *callback* personalizado para el ajuste dinámico del learning rate [40]. Se implementa un mecanismo de ajuste de la tasa de aprendizaje (LRA) personalizado. Este, ajusta dinámicamente la tasa de aprendizaje durante el entrenamiento basado en la precisión y la pérdida de validación, con el objetivo de mejorar la eficiencia del entrenamiento y alcanzar una mejor convergencia. Los componentes clave del LRA son:

1. Modelo: La red neuronal sobre la que se aplicará el *callback*.
2. Paciencia: El número de épocas que se esperará sin mejora en la métrica de rendimiento antes de realizar un ajuste.

3. Umbral (*Threshold*): Un valor límite que define cuándo se considera que ha habido una mejora significativa en el rendimiento.
4. Factor de Ajuste: La magnitud por la cual se modificará el learning rate en caso de no observarse mejoras.
5. Dwell: Una opción que permite al modelo volver a un estado de pesos anterior si no hay mejoras tras el ajuste del learning rate.

La clase LRA se inicializa con varios parámetros, incluyendo la paciencia para ajustar y detener el entrenamiento, el umbral para cambiar entre monitorear la precisión o la pérdida, y factores para ajustar la tasa de aprendizaje. También guarda los pesos iniciales del modelo obtenidos de la red neuronal pre-entrenada. Durante el entrenamiento, el *callback* monitorea constantemente el rendimiento del modelo en términos de precisión y pérdida de validación. Este obtiene la precisión del entrenamiento, la precisión de validación, la pérdida de entrenamiento y la pérdida de validación y se va ajustando de la siguiente forma: si la precisión del entrenamiento es menor que el umbral, ajusta la tasa de aprendizaje basándose en la precisión del entrenamiento. De lo contrario, utiliza la pérdida de validación. Si hay mejoría (según la métrica monitoreada), guarda los mejores pesos y reinicia los contadores de paciencia. Si no hay mejoría durante un número de épocas igual a *patience*, ajusta la tasa de aprendizaje multiplicándola por un *factor*. Además el mismo comprueba si *dwell* es verdadero, de ser así restablece los pesos a los mejores pesos anteriores. Por cada iteración se imprime el estado actual del entrenamiento, incluyendo la tasa de aprendizaje y la métrica monitoreada. Si la tasa de aprendizaje se ha ajustado *stop\_patience* veces sin mejora, detiene el entrenamiento prematuramente evitando el sobre-ajuste.

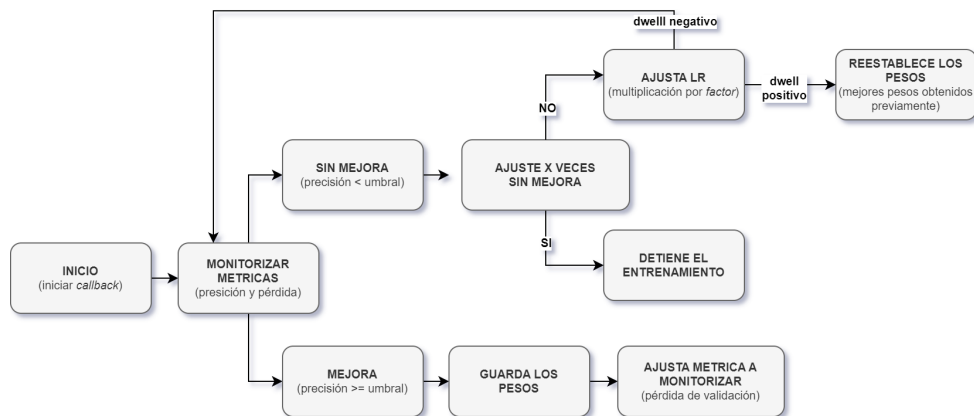


Figura 2.3: Diagrama del ajuste dinámico de la tasa de aprendizaje.

Las métricas utilizadas para la evaluación del modelo que se imprimen en cada iteración se describen en la siguiente tabla:

Tabla 2.1: Descripción de términos clave en el entrenamiento de modelos de aprendizaje automático.

| <b>Término</b>                  | <b>Descripción</b>  |
|---------------------------------|---|
| Epoch                           | Es una iteración completa sobre todo el conjunto de datos de entrenamiento.   |
| Loss (Pérdida)                  | Es una medida de cuán bien el modelo está realizando sus predicciones. Los valores decrecientes indican una mejora en el aprendizaje.   |
| Accuracy (Precisión)            | Muestra el porcentaje de etiquetas que el modelo predice correctamente para el conjunto de entrenamiento.   |
| V loss (Pérdida de Validación)  | Es similar a la pérdida, pero se calcula sobre un conjunto de datos que no se utiliza para el entrenamiento.  |
| V acc (Precisión de Validación) | Muestra el porcentaje de etiquetas que el modelo predice correctamente para el conjunto de datos de validación.   |
| LR (Learning Rate)              | La tasa de aprendizaje dicta cuánto se ajustan los pesos del modelo en cada actualización.  |
| Next LR (Próxima Learning Rate) | Indica la próxima tasa de aprendizaje planificada. La adaptación de la tasa de aprendizaje puede ayudar a evitar el estancamiento y mejorar la convergencia.  |
| Monitor                         | Muestra la métrica que se está utilizando para monitorizar el rendimiento del modelo. Cambia de <i>accuracy</i> a <i>val loss</i> , lo que probablemente indica que el cambio se hizo para evitar el sobreajuste. |
| Duration (Duración)             | Tiempo que tardó cada epoch en completarse. Importante para evaluar la eficiencia del entrenamiento.  |

Para la evaluación final del modelo se utilizan las métricas descritas anteriormente, matrices de confusión, gráficos sobre errores en clases, gráficos de precisión y pérdida e informes de clasificación.



## Capítulo 3

# Detalles de implementación y experimentos

Este capítulo presenta los métodos, técnicas, detalles e hiperparámetros utilizados en la implementación general del modelo y específica de cada experimento.

## Configuraciones generales

El código desarrollado en la implementación de este modelo en público y se puede descargar en github en el repositorio [github.com/deborahfam/Thesis](https://github.com/deborahfam/Thesis) en la carpeta llamada *pipelines*. Cada pipeline es un archivo *.ipynb*. Adelante se describen herramientas y tecnologías que fueron utilizadas en el proyecto:

## Herramientas y Tecnologías

### ■ Frameworks de Desarrollo:

- TensorFlow: Framework principal para la construcción y entrenamiento de modelos de *machine learning* y redes neuronales profundas.

### ■ APIs y bibliotecas de TensorFlow:

- Keras: *API* de alto nivel para la construcción y entrenamiento de modelos en TensorFlow.
- `tf.keras.layers`: Conjunto de capas predefinidas para la construcción de modelos.
- `tf.keras.optimizers`: Algoritmos de optimización como Adamax.

- `tf.keras.metrics`: Métricas de evaluación como *categorical\_crossentropy*.
  - `tf.keras.regularizers`: Técnicas de regularización como L1 y L2.
  - `tf.keras.preprocessing.image`: Herramientas de preprocesamiento y aumento de imágenes.
  - `tf.keras.models`: Herramientas para la creación y carga de modelos.
- **Manipulación de Datos:**
    - NumPy: Manejo de *arrays* y matrices de números n-dimensionales.
    - Pandas: Análisis y manipulación de datos estructurados.
- **Utilidades de Sistema y Archivos:**
    - Shutil: Operaciones de manejo de archivos.
    - OS: Interacción con el sistema operativo.
- **Visión por Computadora:**
    - OpenCV (cv2): Procesamiento de imágenes y visión por computadora.
- **Herramientas de Visualización de Datos:**
    - Matplotlib: Creación de gráficos y visualizaciones.
    - Seaborn: Visualización de datos estadísticos.
- **Evaluación de Modelos:**
    - Scikit-learn (sklearn): Herramientas para selección de modelos y evaluación.
- **Procesamiento de Imágenes:**
    - Pillow (PIL): Procesamiento de imágenes para aplicaciones Python.
- **Herramientas de Progreso y Medición de Tiempo:**
    - TQDM: Barras de progreso para loops.
    - Time: Medición del tiempo de ejecución de código.
- **Mejora de Interfaz y Presentación:**
    - IPython: Mejoras interactivas y de visualización para Python, especialmente en cuadernos Jupyter.

### 3.1. Fuente de datos

El conjunto de datos utilizado para el entrenamiento y validación del algoritmo es el HAM10000. Este es acrónimo de *Human Against Machine with 10000 training images* (Humano Contra Máquina con 10000 imágenes de entrenamiento), y se presenta como una solución al problema de la falta de diversidad y tamaño reducido en los conjuntos de datos disponibles para el diagnóstico automatizado de lesiones cutáneas pigmentadas. Este conjunto de datos es notable por su extenso alcance y diversidad, abarcando una amplia gama de lesiones cutáneas pigmentadas comunes [41].

#### HAM10000

Las 10015 imágenes dermatoscópicas del conjunto de datos HAM10000 se recopilieron a lo largo de 20 años desde dos ubicaciones diferentes: el Departamento de Dermatología de la Universidad Médica de Viena, Austria, y la práctica de cáncer de piel de Cliff Rosendahl en Queensland, Australia [41]. En comparación con otros conjuntos de datos, HAM10000 ofrece un conjunto más diverso y completo de imágenes dermatoscópicas para la investigación del aprendizaje automático. Las imágenes y los metadatos del HAM10000 tienen la siguiente distribución.

Tabla 3.1: Distribución de imágenes por categoría diagnóstica.

| Categoría Diagnóstica                                   | Cantidad | Porcentaje |
|---|----------|------------|
| Melanocytic nevi (NV)                                   | 6705     | 66.95%     |
| Melanoma (MEL)  | 1113     | 11.11%     |
| Benign keratosis-like lesions (BKL)                     | 1099     | 10.97%     |
| Basal cell carcinoma (BCC)                              | 514      | 5.13%      |
| Actinic Keratosis and Intraepithelial Carcinoma (AKIEC) | 327      | 3.27%      |
| Vascular lesions (VASC)                                 | 142      | 1.42%      |
| Dermatofibroma (DF)                                     | 115      | 1.15%      |

Las imágenes almacenadas, originalmente como diapositivas, fueron digitalizadas usando un escáner *Nikon Coolscan 5000 ED*. Posteriormente, se ajustaron manualmente para centrar las lesiones y se aplicaron correcciones al histograma para mejorar el contraste visual y la reproducción del color. Para separar eficientemente las imágenes dermatoscópicas de otros tipos de imágenes (como primeros planos y vistas generales), se utilizó un método automatizado que clasificaba más de 30,000 imágenes. Se empleó una arquitectura *InceptionV3*, entrenada con un conjunto de imágenes etiquetadas manualmente, para categorizar las imágenes. Las imágenes mal clasificadas por este método fueron revisadas y corregidas manualmente. Se realizó una revisión manual final para excluir imágenes con ciertos atributos no deseados, como contenido

potencialmente identificable, imágenes fuera de enfoque o con artefactos perturbadores como prendas, y lesiones completamente no pigmentadas. Las imágenes restantes fueron revisadas para asegurar una reproducción de color y luminosidad adecuadas, aplicando correcciones manuales si era necesario [41].

Los datos utilizados como medio de aprendizaje para este proyecto son imágenes y metadatos. El dataset HAM10000 contiene imágenes y un archivo de metadatos que contienen información relacionada con cada imagen en formato *one hot encoding* [42].

### 3.1.1. Transformación de datos

Lo primero realizado fue la importación y transformación los datos para el modelo. En el proceso de importación, para la carga de imágenes utilizamos `plt.imread` de la biblioteca Matplotlib y `pd.read_csv` de la biblioteca Pandas para cargar los metadatos. Para la transformación de datos los metadatos asociados a la clasificación, etiquetados con el método mencionado (*one hot encoding*) fueron convertidos a un formato *categorico* [43] para su procesamiento. En este proceso a cada elemento se le asignó una etiqueta basada en las categoría de la lesión cutánea, como 'MEL' (Melanoma), 'NV' (Nevus Melanocítico), entre otras. Se elimina del *dataframe*, además, cualquier columna innecesaria, dejando solo las etiquetas y nombres de imágenes relevantes. De tal forma que los datos quedan distribuidos por clase.

Tabla 3.2: Datos transformados a formato categorico.

| Index | Imágenes                | Etiqueta |
|-------|-------------------------|----------|
| 0     | <i>ISIC_0024306.jpg</i> | NV       |
| 1     | <i>ISIC_0024307.jpg</i> | NV       |
| 2     | <i>ISIC_0024308.jpg</i> | NV       |
| 3     | <i>ISIC_0024309.jpg</i> | NV       |
| 4     | <i>ISIC_0024310.jpg</i> | MEL      |

### 3.1.2. Modelación y división del conjunto de datos

Los datos se modelan a partir de un *dataframe* de Pandas [44]. Estos son divididos en 3 conjuntos: Entrenamiento, Validación y Prueba, utilizando un enfoque simple de división de datos en porcentaje.

Inicialmente, el conjunto de datos, fue dividido en dos subconjuntos: *train*, que se destinó para el entrenamiento, y *dummy*, que fue utilizado como una combinación temporal para los conjuntos de validación y prueba. Luego el segundo conjunto fue separado en *valid*, destinado a la validación y *test*, utilizado para las pruebas.

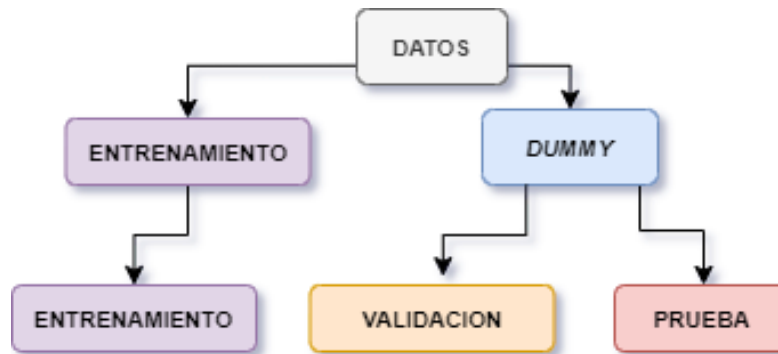


Figura 3.1: Diagrama de división de datos.

Para la división del conjunto de datos en entrenamiento, validación y prueba se utilizó la función `train_test_split` de la biblioteca Scikit-Learn con las variables `random_state=123` y `shuffle=True`. Estas variables aseguran aleatoriedad en la división mezclando los datos antes de dividirlos y que la división sea reproducible. La división, es distinta en cada experimento y se especifican los parámetros utilizados luego en el capítulo.

### 3.1.3. Generadores de datos y preprocesamiento

Uno de los principales problemas del *dataset*, como se puede observar en la tabla 3.1, es el desequilibrio en la representación de las clases, un problema habitual en los conjuntos de datos médicos en los que algunas enfermedades son más raras que otras. Para solucionar este problema, el conjunto de datos se equilibra limitando el número máximo de muestras por clase. Esto garantiza que el modelo no esté sesgado hacia las clases más comunes y pueda generalizar mejor entre varios tipos de lesiones cutáneas.

La cantidad de muestras por clase se ajustó de manera selectiva. Incluso con estas transformaciones algunas clases seguían desbalanceadas, lo que llevó a la implementación de la técnica de *class weighting* [45]. Con este método se asignó pesos diferenciados a cada clase durante el entrenamiento del modelo, reforzando así la señal de aprendizaje para las clases menos representadas.

Además, a los conjuntos segmentados en entrenamiento, validación y prueba, se les aplicó *data augmentation*. Esta técnica aplicó varias transformaciones a las imágenes, como rotación, desplazamiento y zoom, durante su carga, generando lotes de imágenes optimizados para el entrenamiento y evaluación del modelo [46]. Para la carga y procesamiento de imágenes se utilizó la clase `ImageDataGenerator` de Keras [47].

Las transformaciones aplicadas fueron las siguientes

Tabla 3.3: Parámetros de aumento de datos.

| Parámetro          | Descripción  | Valor     |
|--------------------|--|-----------|
| rotation range     | Rango de rotación                                  | 20 grados |
| width shift range  | Rango de desplazamiento horizontal                 | 20 %      |
| height shift range | Rango de desplazamiento vertical                   | 20 %      |
| shear range        | Rango de corte                                     | 20 %      |
| zoom range         | Rango de zoom                                      | 20 %      |
| horizontal flip    | Activación de volteo horizontal                    | verdadero |
| fill mode          | Modo de relleno para manejar los píxeles faltantes | nearest   |

## Preparación del modelo

El fragmento de código siguiente detalla la creación y compilación de el modelo de aprendizaje. El mismo se basa en EfficientNetB1. Para la compilación de este se hace uso de `model.compile`, con un callback personalizado para ajustar la tasa de aprendizaje y para entrenar utilizamos el `model.fit`.

```

1  model_name='EfficientNetB1'
2  base_model=tf.keras.applications.EfficientNetB1(include_top=False
3  , weights="imagenet",input_shape=img_shape , pooling='max')
4
5  x=base_model.output
6  x=keras.layers.BatchNormalization(axis=-1, momentum=0.99, epsilon
7  =0.001 )(x)
8  x = Dense(256, kernel_regularizer = regularizers.l2(l = 0.016),
9  activity_regularizer=regularizers.l1(0.006),
10         bias_regularizer=regularizers.l1(0.006) ,
11         activation='relu')(x)
12 x=Dropout(rate=.45, seed=123)(x)
13
14 output=Dense(class_count, activation='softmax')(x)
15 model=Model(inputs=base_model.input, outputs=output)
16 model.compile(Adamax(learning_rate=.001), loss='
17 categorical_crossentropy', metrics=['accuracy'])

```

## Detalles de Hiperparámetros y Configuración

- **Arquitectura del Modelo:** Se utiliza EfficientNetB1 preentrenado con pesos de ImageNet como base.
- **Tasa de Aprendizaje Inicial:** Establecida en 0,001.

- **Regularización:** Aplicación de regularización L2 con  $\lambda = 0,016$  y regularización L1 con  $\lambda = 0,006$  para los pesos y sesgos.
- **Dropout:** Se implementa una tasa de dropout del 45% para mitigar el sobreajuste.
- **Dimensiones de la Imagen:**  $224 \times 224 \times 3$ .
- **Optimizador:** Adamax.
- **Métrica de Evaluación:** La precisión (*accuracy*) se utiliza como la métrica de evaluación principal.
- **Callbacks:** Se incorpora un ajuste de tasa de aprendizaje (LRA) con los siguientes parámetros
  - **Epochs:** 40.
  - **Patience:** 1.
  - **Stop Patience:** 3.
  - **Threshold:** 0.9.
  - **Factor:** 0.5.
  - **Dwell:** True
  - **Freeze:** False
  - **Batches:** `train_steps = len(train_gen.labels)/batch_size`

## 3.2. Capas adicionales y regularización

Para ajustar y optimizar el modelo de EfficientNetB1, se añaden capas adicionales:

### 3.2.1. Normalización por lotes

Esta capa se define con un *eje de normalización* establecido en  $-1$ , lo que indica que la normalización se aplica a lo largo del último eje en el tensor de entrada. Además, se configura un *momentum* de 0,99 y un valor de *epsilon* de 0,001. El alto valor de *momentum* ayuda a mantener la estabilidad de las medias y varianzas móviles a lo largo del entrenamiento, mientras que el pequeño valor de epsilon evita divisiones por cero, asegurando así cálculos numéricos estables [48].

### 3.2.2. Capa densa

Se integra una capa densa con 256 neuronas, que juega un papel clave en la síntesis de las características aprendidas por el modelo. Se utiliza un regularizador L2 con un  $\lambda$  de 0,016 para los pesos de la capa, lo que ayuda a penalizar y controlar el tamaño de los pesos, reduciendo así el riesgo de sobre-ajuste. Además, tanto el regularizador de actividad como el regularizador de *bias* se configuran con un regularizador L1 con un  $\lambda$  de 0,006. Este enfoque impone una penalización en los pesos y los sesgos, promoviendo un modelo más simple y disperso. La función de activación utilizada es *ReLU*, conocida por su eficacia en la introducción de no linealidad en el modelo, lo que permite aprender relaciones complejas entre las características [49].

### 3.2.3. Dropout

En la arquitectura del modelo, se integra una capa de *dropout* para aumentar la robustez y prevenir el sobre-ajuste. Esta capa se configura con una tasa de desactivación del 45%, lo que significa que, durante el entrenamiento, el 45% de las neuronas se desactivarán aleatoriamente en cada paso. Para asegurar la reproducibilidad, se establece una semilla(seed) 123. Esta introducción de aleatoriedad ayuda a que el modelo no dependa excesivamente de ninguna característica o neurona específica [50].

### 3.2.4. Capa de salida

La capa de salida utiliza una activación *softmax* para transformar las salidas del modelo en probabilidades de pertenencia a cada clase. Esto clasifica las entradas en categorías distintas, proporcionando probabilidades para cada clase, lo cual es esencial en la clasificación multi-clase.

### 3.2.5. Optimización

Para el proceso de entrenamiento se utiliza el optimizador *Adamax* [51]. Este es una variante del conocido optimizador Adam, que combina las ventajas de los métodos adaptativos de tasa de aprendizaje con una implementación más robusta en entornos con gradientes dispersos, lo cual es común en imágenes médicas.

La función de pérdida elegida es la *categorical crossentropy* [43], idónea para problemas de clasificación multi-clase. Se configura el modelo para minimizarla y se rastrea la precisión como métrica principal.



### 3.3. Experimentos

Como se había expresado previamente, se llevaron a cabo una serie de experimentos de los cuales analizaremos los siguientes para la distribución y normalización de datos. La metodología general para ambos experimentos coincide, en ambos se utiliza una red convolucional, EfficientNetB1 y capas adicionales.

#### 3.3.1. Experimento 1: Evaluación de la eficiencia de la división asimétrica de datos en la clasificación de imágenes de cáncer de piel

Este experimento se realiza con la intención de evaluar el comportamiento del modelo utilizando una técnica clásica de división de datos, teniendo en cuenta que la distribución de datos es importante en cuestión de efectividad de los algoritmos. El mismo se centra en una división altamente asimétrica, con un enfoque predominante en el conjunto de entrenamiento. A esta división se le aplica luego un determinado peso para que el algoritmo preste especial atención a las clases menos representadas.

#### Distribución de datos

Las herramientas que se utiliza para hacer la división se hace uso del *dummy split* para mantener la proporción deseada entre validación y prueba. Se escoge la siguiente distribución de datos para realizar una prueba de datos en la que existiera desproporción y se entrenara el modelo para ver fallas e índices de sobre-ajuste y en otras versiones mejorar la distribución de datos del modelo. La distribución fue la siguiente:

Tabla 3.4: Distribución de los conjuntos de entrenamiento, validación y prueba.

| Diagnostic category | Porcentaje | Cantidad de datos |
|---------------------|------------|-------------------|
| Entrenamiento       | 95 %       | 9514              |
| Validación          | 2.5 %      | 251               |
| Prueba              | 2.5 %      | 251               |

Por lo que la division de datos por clase quedaría de la siguiente forma:

Tabla 3.5: Experimento 1: Distribución de imágenes de cáncer de piel en los conjuntos de entrenamiento, test y validación.

| Diagnostic category | Entrenamiento | Validación | Prueba |
|---------------------|---------------|------------|--------|
| NV                  | 300           | 158        | 163    |
| MEL                 | 300           | 25         | 35     |
| BKL                 | 300           | 34         | 30     |
| DF                  | 115           | 5          | 3      |
| AKIEC               | 300           | 11         | 7      |
| BCC                 | 300           | 16         | 10     |
| VASC                | 142           | 1          | 3      |

### Generadores de datos y preprocesamiento

Se establece para este un tamaño objetivo de muestras por clase de 300 , y se utiliza un bucle para iterar a través de cada clase única.

Tabla 3.6: Distribución de muestras por categoría después del sobre-muestreo.

| Diagnostic category | Sampling |
|---------------------|----------|
| AKIEC               | 300      |
| BCC                 | 300      |
| BKL                 | 300      |
| DF                  | 115      |
| MEL                 | 300      |
| NV                  | 300      |
| VASC                | 142      |

Como se evidencia anteriormente, al separar la data en clases el conjunto se mantiene desbalanceado. Se hace necesario aplicar el método llamado *Class Weighting* para compensar este desbalanceo. Este método consiste en asignar un peso a cada clase inversamente proporcional a su frecuencia. De esta manera, las clases con menor representación tendrán un mayor peso y las clases con mayor representación tendrán un menor peso. Esto permite que el modelo se entrene de manera más equilibrada y que no se sesgue hacia las clases con mayor representación.

Tabla 3.7: Distribución de muestras con peso asignado.

| Diagnostic category | Sampling | Weighting |
|---------------------|----------|-----------|
| AKIEC               | 300      | 1.00      |
| BCC                 | 300      | 1.00      |
| BKL                 | 300      | 1.00      |
| DF                  | 115      | 2.60      |
| MEL                 | 300      | 1.00      |
| NV                  | 300      | 1.00      |
| VASC                | 142      | 2.11      |

Ya luego de esta modelación de datos se empieza a entrenar y evaluar el algoritmo hasta obtener los resultados.

## Resultados

En este experimento se observa que la precisión de entrenamiento aumenta con cada epoch, la pérdida de entrenamiento disminuye consistentemente y la tasa de aprendizaje permanece constante al principio y luego disminuye para afinar el entrenamiento a medida que el modelo comienza a converger, lo cual indica que el modelo esta entrenando de forma correcta. Sin embargo, la división asimétrica de datos en este experimento influye en el aprendizaje del modelo. El uso de división asimétrica puede estar causando que el modelo esté sesgado hacia clases con más muestras, afectando la precisión general y la capacidad de generalizar. Esto se evidencia dado que, a pesar de la disminución de la pérdida de validación y el aumento de la precisión de validación, la notable diferencia entre la precisión de entrenamiento y la precisión de validación podría indicar un potencial sobre-ajuste.

Tabla 3.8: Estadísticas básicas del modelo del experimento 1.

| E   | Loss  | Acc    | V loss  | V acc  | LR        | M        | Batch |
|-----|-------|--------|---------|--------|-----------|----------|-------|
| 1   | 9.587 | 40.581 | 8.95658 | 56.800 | $10^{-2}$ | acc      | 85.25 |
| 2   | 7.798 | 67.615 | 7.67235 | 66.800 | $10^{-2}$ | acc      | 21.72 |
| 3   | 6.884 | 79.340 | 6.96014 | 69.600 | $10^{-2}$ | acc      | 22.56 |
| 4   | 6.214 | 87.365 | 6.35865 | 71.200 | $10^{-2}$ | acc      | 25.81 |
| 5   | 5.646 | 91.690 | 5.94812 | 75.200 | $10^{-2}$ | vloss    | 23.08 |
| 6   | 5.172 | 92.999 | 5.44954 | 76.800 | $10^{-2}$ | vloss    | 23.23 |
| 7   | 4.735 | 94.479 | 5.06016 | 76.400 | $10^{-2}$ | vloss    | 23.19 |
| ... | ...   | ...    | ...     | ...    | ...       | ...      | ...   |
| 34  | 0.627 | 99.886 | 1.24470 | 76.800 | 0.00013   | val_loss | 23.30 |

La curva de aprendizaje refleja los resultados mencionados anteriormente. Se presentan dos gráficos: uno para la pérdida y otro para la precisión, a lo largo de los epochs de entrenamiento y validación. En ellos, la pérdida de entrenamiento (línea roja) y la pérdida de validación (línea verde) decrecen a lo largo del tiempo, evidenciando el aprendizaje del modelo. La epoch 31, marcada con un punto azul, se identifica como la más óptima en términos de pérdida de validación. La precisión de entrenamiento (línea roja) se acerca a un 100%, un indicativo de posible sobre-ajuste. Por otro lado, la precisión de validación (línea verde), aunque mejora, muestra variabilidad significativa, alcanzando su máximo en la epoch 29, igualmente señalada con un punto azul. La discrepancia entre las precisiones de entrenamiento y validación sugiere una deficiente generalización del modelo.

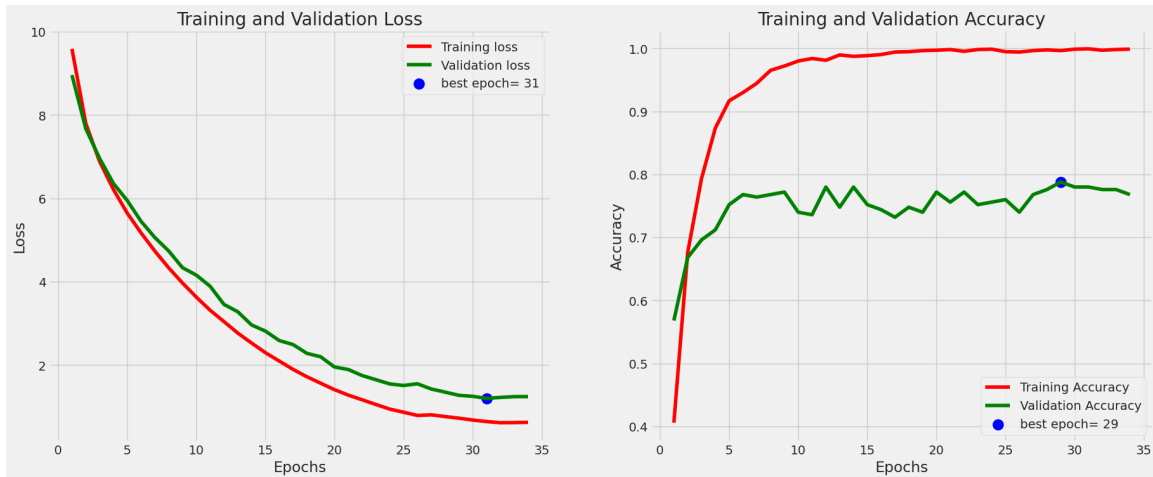


Figura 3.2: Curva de aprendizaje a lo largo del proceso de entrenamiento del experimento 1.

El modelo tardó en entrenarse aproximadamente 2h y tuvo una efectividad de aproximadamente del 81%.

Se implementó para la evaluación del rendimiento del modelo una matriz de confusión, de la que se concluye que:

|        |       | Confusion Matrix |     |     |    |     |     |      |
|--------|-------|------------------|-----|-----|----|-----|-----|------|
| Actual | AKIEC | 7                | 0   | 0   | 0  | 0   | 0   | 0    |
|        | BCC   | 0                | 10  | 0   | 0  | 0   | 0   | 0    |
|        | BKL   | 2                | 2   | 19  | 0  | 5   | 1   | 1    |
|        | DF    | 0                | 0   | 0   | 3  | 0   | 0   | 0    |
|        | MEL   | 2                | 3   | 1   | 0  | 28  | 1   | 0    |
|        | NV    | 4                | 2   | 9   | 6  | 7   | 134 | 1    |
|        | VASC  | 0                | 0   | 0   | 0  | 0   | 0   | 3    |
|        |       | AKIEC            | BCC | BKL | DF | MEL | NV  | VASC |
|        |       | Predicted        |     |     |    |     |     |      |

Figura 3.3: Matriz de confusión del experimento 1.

- **AKIEC:** 7 correctamente identificadas, 0 errores.
- **BCC:** 10 correctamente identificadas, 0 errores.
- **BKL:** 19 correctamente identificadas, 2 clasificadas como AKIEC, 2 como BCC, 5 como NV, 1 como MEL, 1 como VASC.
- **DF:** 0 correctamente identificadas, 3 clasificadas como NV.
- **MEL:** 28 correctamente identificadas, 2 clasificadas como AKIEC, 3 como BCC, 1 como NV.
- **NV:** 134 correctamente identificadas, errores con varias clases.
- **VASC:** 0 correctamente identificadas, 3 clasificadas como NV.

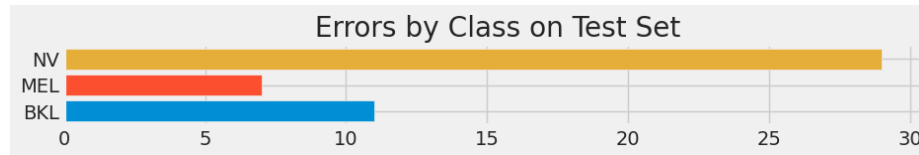


Figura 3.4: Gráfico de errores por clase en el conjunto de pruebas

### 3.3.2. Experimento 2: Análisis de la estratificación de datos en la clasificación de imágenes de cáncer de piel

Este experimento explora la estratificación de datos para mantener una distribución uniforme de etiquetas en cada conjunto de datos. Dado que la distribución anterior muestra un sobre-ajuste, se opta por una proporción de los conjuntos de datos es más equilibrada en comparación con el Experimento 1, lo que ofrece *insights* sobre la importancia de la distribución equitativa de datos en el entrenamiento y evaluación de modelos.

### Distribución de datos

Se utiliza *stratify* en ambas divisiones para mantener la distribución de etiquetas *label* en cada conjunto. Se calcula la proporción del conjunto de prueba sobre la suma del conjunto de *test* y el de validación. La distribución de datos fue la siguiente:

Tabla 3.9: Distribución de los conjuntos de entrenamiento, validación y prueba.

| Diagnostic category | Porcentaje | Cantidad de datos |
|---------------------|------------|-------------------|
| Entrenamiento       | 70%        | 7010              |
| Validación          | 15%        | 1502              |
| Prueba              | 15%        | 1503              |

Luego este quedaría distribuido en datos de entrenamiento, test y validación por clase de la siguiente forma:

Tabla 3.10: Experimento 2: Distribución de imágenes de cáncer de piel en los conjuntos de entrenamiento, test y validación.

| Diagnostic category | Training | Validation | Testing |
|---------------------|----------|------------|---------|
| NV                  | 500      | 1006       | 1006    |
| MEL                 | 500      | 167        | 167     |
| BKL                 | 500      | 165        | 165     |
| DF                  | 500      | 17         | 17      |
| AKIEC               | 500      | 49         | 49      |
| BCC                 | 500      | 77         | 77      |
| VASC                | 500      | 21         | 22      |

### Generadores de datos y preprocesamiento

Se establece también un tamaño objetivo de muestras por clase (500). Se itera sobre cada clase, y se realiza un re-muestreo con reemplazo para grupos menores al tamaño deseado y sin reemplazo para los grupos que ya alcanzan o superan el tamaño deseado. En este experimento, a diferencia del anterior, en cada clase se alcanza la misma cantidad de muestras, por lo que no es necesario aplicar *Class Weighting*.

### Resultados

Similar al Experimento 1, en este experimento también se observa una disminución constante en la pérdida de entrenamiento (Loss) con cada epoch pasando de 8.418 a 0.406. Esto indica un aprendizaje efectivo y una mejora continua en la capacidad del modelo para predecir con precisión las clases. La precisión (Acc), que comienza

en 48.371% y alcanza el 98.057%, corrobora esta mejora constante. A diferencia del Experimento 1, la pérdida de validación (V loss) en este experimento, aunque también muestra una tendencia descendente, tiene una alineación más estrecha entre la precisión de entrenamiento y la precisión de validación. La pérdida de validación disminuye de manera más consistente y la precisión de validación es comparativamente más alta que en el Experimento 1, lo que indica una mejor capacidad de generalización.

Tabla 3.11: Estadísticas básicas del modelo del experimento 2.

| E   | Loss  | Acc    | V loss  | V acc  | LR        | M        | Batch  |
|-----|-------|--------|---------|--------|-----------|----------|--------|
| 1   | 8.418 | 48.371 | 7.41700 | 66.911 | $10^{-2}$ | accuracy | 184.55 |
| 2   | 6.362 | 69.943 | 5.81767 | 69.907 | $10^{-2}$ | accuracy | 107.94 |
| 3   | 5.110 | 77.686 | 4.76153 | 72.969 | $10^{-2}$ | accuracy | 106.30 |
| 4   | 4.181 | 81.371 | 3.86405 | 78.362 | $10^{-2}$ | accuracy | 104.60 |
| 5   | 3.417 | 84.771 | 3.25588 | 77.097 | $10^{-2}$ | accuracy | 101.90 |
| 6   | 2.777 | 87.314 | 2.70253 | 78.495 | $10^{-2}$ | accuracy | 101.85 |
| 7   | 2.247 | 90.143 | 2.24392 | 80.360 | $10^{-2}$ | val_loss | 101.98 |
| 8   | 1.819 | 91.143 | 1.98878 | 77.364 | $10^{-2}$ | val_loss | 101.74 |
| 9   | 1.464 | 92.543 | 1.70207 | 77.896 | $10^{-2}$ | val_loss | 101.27 |
| 10  | 1.197 | 94.000 | 1.43860 | 81.225 | $10^{-2}$ | val_loss | 101.71 |
| ... | ...   | ...    | ...     | ...    | ...       | ...      | ...    |
| 22  | 0.406 | 98.057 | 0.81854 | 83.955 | 0.00006   | val_loss | 101.18 |

Similar al primer gráfico, en el segundo la pérdida de entrenamiento y validación disminuye, lo que es positivo. La mejor epoch basada en la pérdida de validación es la 19, que ocurre antes que en el primer gráfico, lo que indica una convergencia más rápida. La precisión de entrenamiento también es alta, pero no tan cercana al 100% como en el primer gráfico, lo que sugiere un menor riesgo de sobre-ajuste. La precisión de validación muestra menos variabilidad y una alineación más cercana con la precisión de entrenamiento, lo que es un indicador de mejor generalización. La diferencia entre la precisión de entrenamiento y validación es menor en comparación con el primer gráfico, lo que sugiere que el modelo podría estar generalizando mejor.



Figura 3.5: Curva de aprendizaje a lo largo del proceso de entrenamiento del experimento 2.

El modelo tardó en entrenarse aproximadamente 1h y tuvo una efectividad de aproximadamente del 87%.

La matriz de confusión de este experimento muestra los siguientes resultados.

|        |       | Confusion Matrix |     |     |    |     |     |      |
|--------|-------|------------------|-----|-----|----|-----|-----|------|
| Actual | AKIEC | 47               | 1   | 0   | 0  | 1   | 0   | 0    |
|        | BCC   | 0                | 77  | 0   | 0  | 0   | 0   | 0    |
|        | BKL   | 6                | 2   | 137 | 1  | 12  | 7   | 0    |
|        | DF    | 0                | 0   | 0   | 17 | 0   | 0   | 0    |
|        | MEL   | 2                | 0   | 8   | 1  | 142 | 14  | 0    |
|        | NV    | 2                | 15  | 44  | 5  | 69  | 862 | 9    |
|        | VASC  | 0                | 0   | 0   | 0  | 0   | 0   | 22   |
|        |       | AKIEC            | BCC | BKL | DF | MEL | NV  | VASC |
|        |       | Predicted        |     |     |    |     |     |      |

Figura 3.6: Matriz de confusión del experimento 2.

- **AKIEC:** 47 correctamente identificadas, 1 confundida con BCC, 1 con MEL.
- **BCC:** 77 correctamente identificadas, ninguna confundida.
- **BKL:** 137 correctamente identificadas, confundida con otras clases en 28 ocasiones.
- **DF:** 17 correctamente identificadas, sin confusiones.
- **MEL:** 142 correctamente identificadas, confundida con otras clases en 25 ocasiones.
- **NV:** 862 correctamente identificadas, confundida con otras clases en 144 ocasiones.
- **VASC:** 22 correctamente identificadas, sin confusiones.



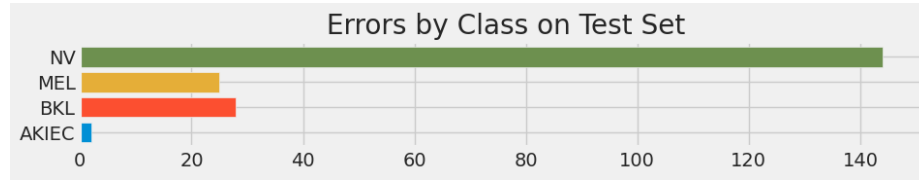


Figura 3.7: Gráfico de errores por clase en el conjunto de pruebas.

### 3.4. Discusión

En esta sección se realiza un análisis comparativo para evaluar la efectividad de los experimentos realizados. Para ello se generan un informe de clasificación que incluye precisión, recuperación (recall), puntuación F1 y F2 y soporte para cada clase y además se analizan los resultados de clasificación de ambos modelos en cuanto a la matrices de confusión y los gráficos de errores por clase generados.

#### Estadísticas de eficacia

Ambas matrices de confusión muestran en común que ambos modelos se desarrollaron bien y que tienen gran efectividad en la clasificación. En estas, la clase con el mayor número de verdaderos positivos es "NV", en la primera con 134 casos predichos correctamente y en la segunda con 862 casos. Sin embargo en otras se nota una mejora de clasificación clara. En el caso de 'AKIEC', la tasa de  $TP$  mejoró significativamente, pasando de 7 de 300 a 47 de 500. Incluso teniendo en cuenta el aumento del tamaño del conjunto de datos, se trata de una clara mejora. Se observan mejoras similares en otras clases, como 'BCC', 'BKL', 'DF', 'MEL' y 'NV'.

La tasa de  $TP$  (*true positive* o verdaderos positivos) ha aumentado no sólo en términos absolutos, sino también proporcionalmente si se tiene en cuenta el mayor tamaño del conjunto de datos. 'VASC' es un caso especial; mientras que la primera matriz no muestra ningún  $TP$  y tiene 3  $FN$  (false negative o falsos negativos), la segunda matriz, a pesar de tener un gran número de  $FP$  (false positive o falsos positivos) (22), muestra que el modelo ha empezado a reconocer esta clase, cosa que antes no hacía.

Con respecto a los errores expresados de cada experimento en la clasificación, en el caso de 'NV', el número de errores del gráfico de barras del experimento 1 se relaciona numéricamente con el conjunto de pruebas, por lo que el modelo tiene casi las mismas posibilidades de hacer una predicción correcta que de dar un falso positivo. En el segundo experimento, a pesar del aumento de errores, el modelo tiene la misma proporción de verdaderos positivos que de falsos positivos, lo que sugiere que el rendi-

miento del modelo se ha mantenido constante en relación con el tamaño del conjunto de datos. 'MEL' y 'BKL' muestran un aumento de los errores en el segundo gráfico de barras, pero también es proporcional al aumento del tamaño del conjunto de datos. La proporción de verdaderos positivos frente a falsos positivos sigue siendo la misma. 'AKIEC' presenta un número relativamente pequeño de errores en el experimento 2, lo que puede deberse a que el conjunto de datos es más grande.

## Informe de clasificación

La tabla siguiente proporcionan una visión cuantitativa de la precisión, el recall (sensibilidad), las métricas F1 y F2 y el soporte (número de muestras verdaderas) (NM) para cada categoría diagnóstica evaluada. Estos indicadores de rendimiento son esenciales para comprender la capacidad del modelo para identificar correctamente cada condición, así como su confiabilidad general en un conjunto de datos diverso. Se añadieron a la tabla los campos *Accuracy*, *MacroAvg*(M Avg) y *WeightedAvg*(W Avg) para un mejor análisis de los resultados:

- **Accuracy:** Esta métrica mide la proporción de predicciones correctas realizadas por el modelo en relación con el total de predicciones. Se calcula como el número de predicciones correctas dividido por el número total de predicciones.
- **Promedio Macro (Macro Average):** Este enfoque calcula la métrica (como la precisión, el recall, o el F1-score) de manera independiente para cada clase y luego toma el promedio de estas métricas. No tiene en cuenta el desequilibrio de las clases, es decir, trata a todas las clases por igual, independientemente de su frecuencia en el conjunto de datos.
- **Promedio Ponderado (Weighted Average):** Similar al promedio macro, pero en este caso, el promedio se pondera. Esto significa que la métrica para cada clase se multiplica por el número de instancias en esa clase (el soporte) antes de calcular el promedio. De esta manera, las clases con más instancias tienen más peso en el promedio final.

Se calcula además  $F_\beta$  o F2. El cálculo del puntaje F2 es una forma de considerar la precisión y la recuperación (*recall*) en un único número, ponderando uno de estos aspectos más que el otro. En nuestro caso específico le asignamos al recobrado prioridad con respecto a la precisión, dado que en la práctica sería mas importante evitar los falsos positivos. Asignamos  $\beta = 2$ .

Tabla 3.12: Informe de clasificación combinado para los Experimentos 1 y 2.

| Categoría    | Experimento 1 |        |      |       |     | Experimento 2 |        |      |       |      |
|--------------|---------------|--------|------|-------|-----|---------------|--------|------|-------|------|
|              | Acc           | Recall | F1   | F2    | NM  | Acc           | Recall | F1   | F2    | NM   |
| AKIEC        | 0.47          | 1.00   | 0.64 | 0.816 | 7   | 0.82          | 0.96   | 0.89 | 0.928 | 49   |
| BCC          | 0.59          | 1.00   | 0.74 | 0.878 | 10  | 0.81          | 1.00   | 0.90 | 0.955 | 77   |
| BKL          | 0.66          | 0.63   | 0.64 | 0.636 | 30  | 0.72          | 0.83   | 0.77 | 0.805 | 165  |
| DF           | 0.33          | 1.00   | 0.50 | 0.711 | 3   | 0.71          | 1.00   | 0.83 | 0.924 | 17   |
| MEL          | 0.70          | 0.80   | 0.75 | 0.778 | 35  | 0.63          | 0.85   | 0.73 | 0.795 | 167  |
| NV           | 0.99          | 0.82   | 0.90 | 0.849 | 163 | 0.98          | 0.86   | 0.91 | 0.882 | 1006 |
| VASC         | 0.60          | 1.00   | 0.75 | 0.882 | 3   | 0.71          | 1.00   | 0.83 | 0.924 | 22   |
| <b>Acc</b>   |               |        | 0.81 |       | 251 |               |        | 0.87 |       | 1503 |
| <b>M Avg</b> | 0.62          | 0.89   | 0.70 |       | 251 | 0.77          | 0.93   | 0.84 |       | 1503 |
| <b>W Avg</b> | 0.86          | 0.81   | 0.83 |       | 251 | 0.89          | 0.87   | 0.87 |       | 1503 |

El campo *accuracy* muestra que la precisión mejora de 0,81 a 0,87, reflejando un aumento en la capacidad general del modelo para hacer predicciones correctas. El Macro Avg (Promedio Macro) aumento de 0,62 a 0,77 en precisión y de 0,70 a 0,84 en la puntuación F1, indicando una mejora en el rendimiento medio del modelo a través de todas las categorías. El Weighted Avg (Promedio Ponderado) muestra un incremento de 0,86 a 0,89 en precisión y de 0,83 a 0,87 en la puntuación F1, mostrando que, teniendo en cuenta el número de muestras (soporte), el rendimiento general del modelo ha mejorado.

En todas las categorías, el puntaje F2 es mayor en el Experimento 2 que en el Experimento 1. Esto indica que este fue más exitoso en equilibrar la precisión y el recall, especialmente con un enfoque en minimizar los falsos negativos. La mejora general en el Experimento 2 confirma un mejor equilibrio en el conjunto de datos y la efectividad de esto para nuestro propósito.

### 3.4.1. Observaciones generales

En general los experimentos demuestran resultados interesantes en la clasificación. El Experimento 2 muestra mejoras generalizadas en precisión, *recall* y puntuaciones F1 para la mayoría de las categorías. El soporte aumenta significativamente, lo que indica que se evaluó al modelo con más muestras, proporcionando una base más robusta para la evaluación del rendimiento. A pesar de que el soporte es mayor, lo que generalmente hace más desafiante mantener altas métricas, el Experimento 2 demuestra mejoras en las métricas en general. Este, a diferencia del 1, maneja mejor el desequilibrio de clases, lo que se refleja en una mejor diferenciación entre ciertas categorías diagnósticas, lo que es indicativo de un modelo más preciso y confiable.

### 3.4.2. Consideraciones finales

Los resultados obtenidos son prometedores y sugieren que los modelos de aprendizaje profundo tienen un potencial considerable para mejorar la precisión y la eficiencia del diagnóstico del cáncer de piel. Tomando como referencia los resultados del experimento 2, se obtuvo una eficacia de clasificación de 87 %, lo cual es bajo con respecto a métodos más robustos mencionados en el estado del arte pero prometedor teniendo en cuenta que los algoritmos desarrollados utilizando EfficientNet obtienen resultados entre 84 % y 86.5 % de eficacia [35].

Es notable además la importancia para el dataset específico utilizado (HAM10000) de un balance proporcional en el conjunto de datos, dado que se evidencia en los resultados que la desproporción entre los mismos lleva a errores de sobre-ajuste y a un peor rendimiento del modelo.

Tabla 3.13: Resultados del entrenamiento del experimento 1.

| Epoch  | Loss  | Acc    | V Loss  | V Acc  | LR      | Next LR | Monitor      | Duration |
|--------|-------|--------|---------|--------|---------|---------|--------------|----------|
| 1 /40  | 9.587 | 40.581 | 8.95658 | 56.800 | 0.00100 | 0.00100 | precisión    | 85.25    |
| 2 /40  | 7.798 | 67.615 | 7.67235 | 66.800 | 0.00100 | 0.00100 | precisión    | 21.72    |
| 3 /40  | 6.884 | 79.340 | 6.96014 | 69.600 | 0.00100 | 0.00100 | precisión    | 22.56    |
| 4 /40  | 6.214 | 87.365 | 6.35865 | 71.200 | 0.00100 | 0.00100 | precisión    | 25.81    |
| 5 /40  | 5.646 | 91.690 | 5.94812 | 75.200 | 0.00100 | 0.00100 | pérdida val. | 23.08    |
| 6 /40  | 5.172 | 92.999 | 5.44954 | 76.800 | 0.00100 | 0.00100 | pérdida val. | 23.23    |
| 7 /40  | 4.735 | 94.479 | 5.06016 | 76.400 | 0.00100 | 0.00100 | pérdida val. | 23.19    |
| 8 /40  | 4.334 | 96.528 | 4.73837 | 76.800 | 0.00100 | 0.00100 | pérdida val. | 22.90    |
| 9 /40  | 3.969 | 97.211 | 4.33689 | 77.200 | 0.00100 | 0.00100 | pérdida val. | 22.74    |
| 10 /40 | 3.631 | 98.008 | 4.15826 | 74.000 | 0.00100 | 0.00100 | pérdida val. | 22.56    |
| 11 /40 | 3.315 | 98.406 | 3.89153 | 73.600 | 0.00100 | 0.00100 | pérdida val. | 23.11    |
| 12 /40 | 3.044 | 98.122 | 3.45597 | 78.000 | 0.00100 | 0.00100 | pérdida val. | 22.90    |
| 13 /40 | 2.768 | 98.976 | 3.27968 | 74.800 | 0.00100 | 0.00100 | pérdida val. | 23.01    |
| 14 /40 | 2.530 | 98.748 | 2.96314 | 78.000 | 0.00100 | 0.00100 | pérdida val. | 22.32    |
| 15 /40 | 2.298 | 98.862 | 2.81532 | 75.200 | 0.00100 | 0.00100 | pérdida val. | 22.32    |
| 16 /40 | 2.097 | 99.032 | 2.58953 | 74.400 | 0.00100 | 0.00100 | pérdida val. | 25.73    |
| 17 /40 | 1.898 | 99.431 | 2.49290 | 73.200 | 0.00100 | 0.00100 | pérdida val. | 23.25    |
| 18 /40 | 1.721 | 99.488 | 2.28217 | 74.800 | 0.00100 | 0.00100 | pérdida val. | 22.98    |
| 19 /40 | 1.565 | 99.659 | 2.19958 | 74.000 | 0.00100 | 0.00100 | pérdida val. | 22.36    |
| 20 /40 | 1.411 | 99.715 | 1.95434 | 77.200 | 0.00100 | 0.00100 | pérdida val. | 22.71    |
| 21 /40 | 1.280 | 99.829 | 1.89412 | 75.600 | 0.00100 | 0.00100 | pérdida val. | 25.82    |
| 22 /40 | 1.169 | 99.545 | 1.74949 | 77.200 | 0.00100 | 0.00100 | pérdida val. | 23.20    |
| 23 /40 | 1.055 | 99.829 | 1.64937 | 75.200 | 0.00100 | 0.00100 | pérdida val. | 23.06    |
| 24 /40 | 0.944 | 99.886 | 1.54814 | 75.600 | 0.00100 | 0.00100 | pérdida val. | 22.80    |
| 25 /40 | 0.870 | 99.488 | 1.51110 | 76.000 | 0.00100 | 0.00100 | pérdida val. | 22.87    |
| 26 /40 | 0.793 | 99.431 | 1.55223 | 74.000 | 0.00100 | 0.00050 | pérdida val. | 22.38    |
| 27 /40 | 0.807 | 99.659 | 1.42688 | 76.800 | 0.00050 | 0.00050 | pérdida val. | 25.70    |
| 28 /40 | 0.765 | 99.772 | 1.35257 | 77.600 | 0.00050 | 0.00050 | pérdida val. | 23.38    |
| 29 /40 | 0.727 | 99.659 | 1.27547 | 78.800 | 0.00050 | 0.00050 | pérdida val. | 22.97    |
| 30 /40 | 0.682 | 99.886 | 1.25111 | 78.000 | 0.00050 | 0.00050 | pérdida val. | 22.80    |
| 31 /40 | 0.646 | 99.943 | 1.20186 | 78.000 | 0.00050 | 0.00050 | pérdida val. | 23.36    |
| 32 /40 | 0.619 | 99.715 | 1.22486 | 77.600 | 0.00050 | 0.00025 | pérdida val. | 23.07    |
| 33 /40 | 0.621 | 99.829 | 1.24364 | 77.600 | 0.00025 | 0.00013 | pérdida val. | 23.57    |
| 34 /40 | 0.627 | 99.886 | 1.24470 | 76.800 | 0.00013 | 0.00006 | pérdida val. | 23.30    |

Tabla 3.14: Resultados del entrenamiento del experimento 2.

| <b>Epoch</b> | <b>Loss</b> | <b>Acc</b> | <b>V Loss</b> | <b>V Acc</b> | <b>LR</b> | <b>Next LR</b> | <b>Monitor</b> | <b>Duration</b> |
|--------------|-------------|------------|---------------|--------------|-----------|----------------|----------------|-----------------|
| 1 /40        | 8.418       | 48.371     | 7.41700       | 66.911       | 0.00100   | 0.00100        | precisión      | 184.55          |
| 2 /40        | 6.362       | 69.943     | 5.81767       | 69.907       | 0.00100   | 0.00100        | precisión      | 107.94          |
| 3 /40        | 5.110       | 77.686     | 4.76153       | 72.969       | 0.00100   | 0.00100        | precisión      | 106.30          |
| 4 /40        | 4.181       | 81.371     | 3.86405       | 78.362       | 0.00100   | 0.00100        | precisión      | 104.60          |
| 5 /40        | 3.417       | 84.771     | 3.25588       | 77.097       | 0.00100   | 0.00100        | precisión      | 101.90          |
| 6 /40        | 2.777       | 87.314     | 2.70253       | 78.495       | 0.00100   | 0.00100        | precisión      | 101.85          |
| 7 /40        | 2.247       | 90.143     | 2.24392       | 80.360       | 0.00100   | 0.00100        | pérdida val.   | 101.98          |
| 8 /40        | 1.819       | 91.143     | 1.98878       | 77.364       | 0.00100   | 0.00100        | pérdida val.   | 101.74          |
| 9 /40        | 1.464       | 92.543     | 1.70207       | 77.896       | 0.00100   | 0.00100        | pérdida val.   | 101.27          |
| 10 /40       | 1.197       | 94.000     | 1.43860       | 81.225       | 0.00100   | 0.00100        | pérdida val.   | 101.71          |
| 11 /40       | 0.992       | 94.000     | 1.24007       | 81.358       | 0.00100   | 0.00100        | pérdida val.   | 101.23          |
| 12 /40       | 0.826       | 94.800     | 1.10713       | 81.625       | 0.00100   | 0.00100        | pérdida val.   | 100.90          |
| 13 /40       | 0.699       | 95.086     | 1.08394       | 79.228       | 0.00100   | 0.00100        | pérdida val.   | 101.31          |
| 14 /40       | 0.580       | 96.571     | 0.86793       | 84.021       | 0.00100   | 0.00100        | pérdida val.   | 101.53          |
| 15 /40       | 0.498       | 96.886     | 0.85960       | 83.688       | 0.00100   | 0.00100        | pérdida val.   | 100.92          |
| 16 /40       | 0.453       | 96.914     | 0.88978       | 81.025       | 0.00100   | 0.00050        | pérdida val.   | 116.17          |
| 17 /40       | 0.436       | 97.857     | 0.93461       | 79.561       | 0.00050   | 0.00025        | pérdida val.   | 100.73          |
| 18 /40       | 0.449       | 97.543     | 0.80205       | 83.755       | 0.00025   | 0.00025        | pérdida val.   | 101.01          |
| 19 /40       | 0.420       | 97.971     | 0.79843       | 84.887       | 0.00025   | 0.00025        | pérdida val.   | 115.73          |
| 20 /40       | 0.399       | 98.429     | 0.82976       | 83.888       | 0.00025   | 0.00013        | pérdida val.   | 100.45          |
| 21 /40       | 0.395       | 98.571     | 0.83847       | 83.688       | 0.00013   | 0.00006        | pérdida val.   | 101.25          |
| 22 /40       | 0.406       | 98.057     | 0.81854       | 83.955       | 0.00006   | 0.00003        | pérdida val.   | 101.18          |

# Conclusiones

En el presente trabajo se dio solución al problema de clasificación de tumores de cáncer de piel mediante un algoritmo de machine learning. La metodología propuesta se basó en la utilización de una red neuronal convolucional pre-entrenada, EfficientNetB1, un algoritmo de ajuste de aprendizaje y varias capas de optimización adicionales. Para el entrenamiento y la validación del modelo, se utilizó el conjunto de datos HAM10000, que destaca por su diversidad y extenso alcance, abarcando una amplia gama de lesiones cutáneas pigmentadas.

A continuación, se presentan las conclusiones del estudio, organizadas por ítems clave:

- **Potencial del Aprendizaje Profundo:** Los resultados obtenidos demuestran un potencial considerable de los modelos de aprendizaje profundo para mejorar la precisión y eficiencia en el diagnóstico de cáncer de piel.
- **Eficacia de la Clasificación:** El experimento 2 alcanzó una eficacia del 87%, indicando una capacidad prometedora de clasificación, ligeramente inferior a los métodos más robustos del estado del arte pero considerando que los algoritmos desarrollados utilizando EfficientNet generalmente alcanzan eficacias entre el 84% y el 86.5%.
- **Comparación entre Experimentos:** Se observaron mejoras significativas en el segundo experimento en términos de precisión, recall, y puntuaciones F1, lo que sugiere una mayor precisión y confiabilidad en la clasificación.
- **Evolución de la Pérdida y Precisión:** Se registró una mejora constante en la precisión y una disminución en la pérdida a lo largo de las iteraciones en ambos experimentos, aunque con señales de sobre-ajuste en el primer experimento.
- **Análisis de Curvas de Aprendizaje:** Las curvas de aprendizaje de ambos experimentos mostraron disminuciones en la pérdida y mejoras en la precisión, con una convergencia más rápida y mejor generalización en el segundo experimento.

- **Estadísticas de Eficacia y Matrices de Confusión:** Las matrices de confusión y estadísticas de eficacia revelaron una clasificación precisa en varias clases de cáncer de piel, especialmente en el segundo experimento.
- **Importancia de la Calidad y Composición del Conjunto de Datos:** La calidad y composición del conjunto de datos HAM10000 se identifican como aspectos críticos para el éxito de los modelos de aprendizaje profundo en la clasificación de imágenes dermatológicas. Se observó que la desproporción en este conjunto de datos lleva a errores de sobre-ajuste y a un rendimiento inferior del modelo. Esto indica que la calidad y la composición del conjunto de datos son aspectos críticos para el éxito de los modelos de aprendizaje profundo en aplicaciones de clasificación de imágenes dermatológicas.



# Recomendaciones

La sección de recomendaciones de esta tesis ofrece opciones para mejorar y refinar el proceso de clasificación automática del cáncer de piel utilizando las técnicas de aprendizaje profundo propuestas en el presente trabajo. Estas recomendaciones están divididas en dos categorías principales: una enfocada en el procesamiento y estrategias de normalización de datos, y la otra centrada en la optimización y ajuste de hiperparámetros del modelo. Cada categoría contiene una lista de pasos específicos y sugerencias prácticas para abordar desafíos comunes y mejorar la precisión y eficiencia del modelo. A continuación, se presentan estas recomendaciones en detalle:

## **Datos: procesamiento y estrategias de normalización**

1. **Enriquecimiento de Datos:** Complementar el conjunto de datos con imágenes adicionales de fuentes confiables para mejorar la robustez del modelo.
2. **Aumentar la Diversidad de Datos:** Utilizar técnicas de aumento de datos para generar variantes adicionales de las imágenes existentes, como rotaciones, zoom, o cambios en el brillo, para mejorar la capacidad del modelo de generalizar a partir de nuevos datos.
3. **Balance de Clases mediante Aumento:** Para categorías con menos ejemplos, aplicar técnicas de aumento de datos de manera selectiva para equilibrar la distribución de clases y reducir el sesgo del modelo.
4. **Estratificación Mejorada:** Asegurarse de que la estratificación de datos se aplique de manera que todas las particiones (entrenamiento, validación, prueba) tengan una distribución representativa de cada clase.

## **Modelo: optimización y ajuste de hiperparámetros**

1. **Regularización y Arquitectura de Red:** Explorar diferentes parámetros de regularización, para dropout y L1/L2 para encontrar el mejor equilibrio entre el rendimiento y la prevención del sobre-ajuste.

2. Presición y sesgos: Implementar modelos más complejos de clasificación para las clases menos representadas.
3. Optimizadores: Utilizar otros optimizadores como RMSprop o SGD para mejorar la velocidad de convergencia y el rendimiento del modelo.

# Bibliografía

- [1] Yann LeCun, Yoshua Bengio y Geoffrey Hinton. «Deep learning». En: *nature* 521.7553 (2015), págs. 436-444 (vid. pág. 1).
- [2] Li Deng, Dong Yu y col. «Deep learning: methods and applications». En: *Foundations and trends® in signal processing* 7.3-4 (2014), págs. 197-387 (vid. pág. 1).
- [3] Lei Cai, Jingyang Gao y Di Zhao. «A review of the application of deep learning in medical image classification and segmentation». En: *Annals of translational medicine* 8.11 (2020) (vid. pág. 1).
- [4] Hyun Ah Song y Soo-Young Lee. «Hierarchical representation using NMF». En: *Neural Information Processing: 20th International Conference, ICONIP 2013, Daegu, Korea, November 3-7, 2013. Proceedings, Part I* 20. Springer. 2013, págs. 466-473 (vid. pág. 1).
- [5] *Estadísticas importantes sobre el cáncer de piel tipo melanoma*. Último acceso: 2023-11-27. American Cancer Society, 2023. URL: <https://www.cancer.org/es/cancer/tipos/cancer-de-piel-tipo-melanoma/acerca/estadisticas-clave.html> (vid. págs. 1, 2).
- [6] Fundación Piel Sana - Academia Española de Dermatología y Venereología. *¿Qué es la Dermatología?* <https://aedv.fundacionpielsana.es/piel-sana/que-es-la-dermatologia/>. Último acceso: 19 de noviembre de 2023 (vid. pág. 1).
- [7] Kinnor Das y col. «Machine learning and its application in skin cancer». En: *International Journal of Environmental Research and Public Health* 18.24 (2021), pág. 13409 (vid. págs. 2, 10).
- [8] Departamento de Ingeniería Eléctrica. Universidad Nacional de Colombia Alberto Delgado PhD. Profesor Asociado. *Aplicación de las Redes Neuronales en Medicina*. Online; accessed 22-October-2023. 1999. URL: <https://repositorio.unal.edu.co/bitstream/handle/unal/32711/19460-64062-1-PB.pdf> (vid. págs. 2, 6-8, 10).

- [9] Darien Viera Barredo. «Autómata celular estocástico en redes complejas para el estudio de la invasión, migración y metástasis del cancer». Tesis de Licenciatura. Habana, Cuba: Universidad de la Habana, 2019 (vid. pág. 2).
- [10] Claudia Olavarrieta Martínez. «Ensemble de redes convolucionales para la clasificación de lesiones de cáncer de piel». Tesis de Licenciatura. Habana, Cuba: Universidad de la Habana, 2022 (vid. pág. 3).
- [11] X. Du-Harpur y col. «What is AI? Applications of artificial intelligence to dermatology». En: *Frontiers in Medicine* 7 (2020). DOI: 10.3389/fmed.2020.00100. URL: <https://www.frontiersin.org/articles/10.3389/fmed.2020.00100/full> (vid. pág. 5).
- [12] G Romero y col. «Practice models in teledermatology in Spain: longitudinal study, 2009-2014». En: *Actas Dermo-Sifiliográficas (English Edition)* 109.7 (2018), págs. 624-630 (vid. pág. 5).
- [13] John D Whited y col. «Teledermatology's impact on time to intervention among referrals to a dermatology consult service». En: *Telemedicine Journal and e-Health* 8.3 (2002), págs. 313-321 (vid. pág. 5).
- [14] D1 Piccolo y col. «Concordance between telepathologic diagnosis and conventional histopathologic diagnosis: a multiobserver store-and-forward study on 20 skin specimens». En: *Archives of dermatology* 138.1 (2002), págs. 53-58 (vid. pág. 5).
- [15] Qindel. *Reconocimiento de imágenes: Qué es y cómo funciona*. Último acceso: 2023-12-01. 2023. URL: <https://www.qindel.com/que-es-y-como-funciona-el-reconocimiento-de-imagenes/> (vid. pág. 6).
- [16] L. Gerhardt. «Pattern recognition and machine learning». En: *IEEE Transactions on Automatic Control* 19.4 (1974), págs. 461-462. DOI: 10.1109/TAC.1974.1100578 (vid. pág. 6).
- [17] Datamount. *What Canny Edge Detection Algorithm is All About*. Último acceso: 2023-12-01. 2023. URL: <https://medium.com/@datamount/what-canny-edge-detection-algorithm-is-all-about-103d94553d21> (vid. pág. 7).
- [18] Weibin Rong y col. «An improved CANNY edge detection algorithm». En: *2014 IEEE international conference on mechatronics and automation*. IEEE. 2014, págs. 577-582 (vid. pág. 7).
- [19] *Cómo Extraer Features de una Imagen con HOG en Scikit-Image*. Último acceso: 2023-11-27. DataSmarts Español, 2020. URL: <https://datasmarts.net/es/como-extraer-features-de-una-imagen-con-hog-en-scikit-image/> (vid. pág. 7).
- [20] Tony Lindeberg. «Scale invariant feature transform». En: (2012) (vid. pág. 7).

- [21] Michael Calonder y col. «Brief: Binary robust independent elementary features». En: *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, Part IV 11*. Springer. 2010, págs. 778–792 (vid. pág. 7).
- [22] *Color Histograms in Image Retrieval*. Disponible en: <https://www.pinecone.io/learn/series/image-search/color-histograms/>. 2023. URL: <https://www.pinecone.io/learn/series/image-search/color-histograms/> (vid. pág. 7).
- [23] Isidoro Gil Leiva, Pedro Díaz Ortuño y José Vicente Rodríguez Muñoz. *Técnicas y usos en la clasificación automática de imágenes*. 2019 (vid. pág. 7).
- [24] Kang-Woo Lee y col. «Analysis of facial ultrasonography images based on deep learning». En: *Scientific reports* 12.1 (2022), pág. 16480 (vid. pág. 8).
- [25] Akshay S Chaudhari y col. «Low-count whole-body PET with deep learning in a multicenter and externally validated study». En: *NPJ digital medicine* 4.1 (2021), pág. 127 (vid. pág. 8).
- [26] Kaustav Bera y col. «Predicting cancer outcomes with radiomics and artificial intelligence in radiology». En: *Nature Reviews Clinical Oncology* 19.2 (2022), págs. 132–146 (vid. pág. 8).
- [27] Yinhao Wu y col. «Skin cancer classification with deep learning: a systematic review». En: *Frontiers in Oncology* 12 (2022), pág. 893972 (vid. págs. 9, 10).
- [28] Titus Josef Brinker y col. «Skin cancer classification using convolutional neural networks: systematic review». En: *Journal of medical Internet research* 20.10 (2018), e11936 (vid. pág. 11).
- [29] A Ameri. «A deep learning approach to skin cancer detection in dermoscopy images». En: *Journal of biomedical physics & engineering* 10.6 (2020), pág. 801 (vid. pág. 11).
- [30] Bhuvaneshwari Shetty y col. «Skin lesion classification of dermoscopic images using machine learning and convolutional neural network». En: *Scientific Reports* 12.1 (2022), pág. 18134 (vid. pág. 11).
- [31] Amin Tajerian y col. «Design and validation of a new machine-learning-based diagnostic tool for the differentiation of dermatoscopic skin cancer images». En: *Plos one* 18.4 (2023), e0284437 (vid. pág. 11).
- [32] Adekanmi Adegun y Serestina Viriri. «Deep learning techniques for skin lesion analysis and melanoma cancer detection: a survey of state-of-the-art». En: *Artificial Intelligence Review* 54 (2021), págs. 811–841 (vid. págs. 11, 12).

- [33] Mehwish Dildar y col. «Skin cancer detection: a review using deep learning techniques». En: *International journal of environmental research and public health* 18.10 (2021), pág. 5479 (vid. pág. 12).
- [34] Mingxing Tan y Quoc Le. *EfficientNet: Improving Accuracy and Efficiency through AutoML and Model Scaling*. Accessed: [24 de noviembre de 2023]. 2019. URL: <https://blog.research.google/2019/05/efficientnet-improving-accuracy-and.html> (vid. págs. 13, 16, 17).
- [35] Karar Ali y col. «Multiclass skin cancer classification using EfficientNets—a first step towards preventing skin cancer». En: *Neuroscience Informatics* 2.4 (2022), pág. 100034 (vid. págs. 13, 40).
- [36] S Papiththira y T Kokul. «Melanoma Skin Cancer Detection Using EfficientNet and Channel Attention Module». En: *2021 IEEE 16th International Conference on Industrial and Information Systems (ICIIS)*. IEEE. 2021, págs. 227-232 (vid. pág. 13).
- [37] *EfficientNetB1*. <https://towardsdatascience.com/complete-architectural-details-of-efficientnets-7e50bc115fe0> (vid. pág. 15).
- [38] Mingxing Tan y Quoc V. Le. «EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks». En: *ar5iv* (2019). URL: <https://ar5iv.labs.arxiv.org/html/1905.11946> (vid. pág. 16).
- [39] Pinecone. *ImageNet - The Dataset that Transformed Image Classification*. 2021. URL: <https://www.pinecone.io/learn/series/image-search/imagenet/> (vid. pág. 18).
- [40] *Tasa de aprendizaje*. <https://www.jeremyjordan.me/nn-learning-rate/>. 2023 (vid. pág. 18).
- [41] Philipp Tschandl, Cliff Rosendahl y Harald Kittler. «The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions». En: *Scientific data* 5.1 (2018), págs. 1-9 (vid. págs. 23, 24).
- [42] *One-hot encoding*. <https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/>. 2023 (vid. pág. 24).
- [43] Vijay Singh. *Keras - Categorical Cross Entropy Loss Function*. Accessed: [25 de noviembre de 2023]. 2021. URL: <https://vitalflux.com/keras-categorical-cross-entropy-loss-function/> (vid. págs. 24, 28).
- [44] *Dataframes de Pandas*. [https://pandas.pydata.org/docs/getting\\_started/intro\\_tutorials/01\\_table\\_oriented.html](https://pandas.pydata.org/docs/getting_started/intro_tutorials/01_table_oriented.html). 2023 (vid. pág. 24).

- [45] Analytics Vidhya. *How to Improve Class Imbalance using Class Weights in ML?* Online; accessed 16-November-2023. Oct. de 2020. URL: <https://www.analyticsvidhya.com/blog/2020/10/improve-class-imbalance-class-weights/> (vid. pág. 25).
- [46] *Aumentación de datos*. <https://machinelearningmastery.com/image-augmentation-deep-learning-keras/>. 2023 (vid. pág. 25).
- [47] *ImageDataGenerator*. <https://keras.io/api/preprocessing/image/>. 2023 (vid. pág. 25).
- [48] *Regularización*. <https://machinelearningmastery.com/how-to-reduce-overfitting-in-deep-neural-networks-with-weight-regularization-in-keras/>. 2023 (vid. pág. 27).
- [49] *Capas densas*. <https://towardsdatascience.com/activated-dense-layer-intuition-and-implementation-93e090cad34>. 2023 (vid. pág. 28).
- [50] *Dropout*. <https://machinelearningmastery.com/dropout-regularization-deep-learning-models-keras/>. 2023 (vid. pág. 28).
- [51] *Adamax*. <https://keras.io/api/optimizers/adamax/>. 2023 (vid. pág. 28).