



UNIVERSIDAD
DE MÁLAGA



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA
GRADO EN INGENIERÍA DE LA SALUD

CLASIFICACIÓN DE IMÁGENES DE LESIONES CUTÁNEAS MEDIANTE
REDES NEURONALES CONVOLUCIONALES Y APRENDIZAJE
PROFUNDO

CLASSIFICATION OF SKIN LESION IMAGES WITH CONVOLUTIONAL
NEURAL NETWORKS AND DEEP LEARNING

Realizado por

Javier García Ciudad

Tutorizado por

Ezequiel López Rubio

Jorge García González

Departamento

Departamento de Lenguajes y Ciencias de la Computación

UNIVERSIDAD DE MÁLAGA

MÁLAGA, JUNIO 2020

Fecha de defensa: julio de 2020

Resumen

El cáncer de piel es el cuarto tipo de cáncer más común en el mundo, el primero en numerosos países y, además, se espera que su incidencia aumente en las próximas décadas. Gran parte de ellos presentan una mortalidad baja, pero una pequeña parte, los melanomas, tienen una mortalidad más elevada. Aún así, estos presentan una tasa de supervivencia cercana al 100 % cuando es detectado en las primeras etapas. No obstante, el diagnóstico de estas lesiones es complicado ya que estas patologías no son fácilmente identificables visualmente, por lo que un sistema de redes neuronales convolucionales capaz de clasificar correctamente este tipo de lesiones a partir de imágenes sería de gran utilidad.

En esta línea, se ha desarrollado un clasificador basado en una red convolucional Efficient-NetB0, variando distintos parámetros de entrenamiento y de la misma red hasta alcanzar un modelo final. Esta red es capaz de clasificar imágenes de 8 patologías principales, entre las que se encuentran los principales tipos de cáncer de piel entre otras enfermedades. Además consta de otra clase más que representa a todas las patologías no incluidas en las 8 anteriores, de forma que el modelo pueda reconocer cuándo una imagen es desconocida. En el trabajo nos hemos centrado en el comportamiento que presenta el modelo de cara a esta clase.

Los resultados, aunque se deben tener en cuenta sus limitaciones, son positivos, tanto en general como específicamente en la clase nombrada anteriormente: un 74 % de precisión general y una sensibilidad del 66 % para dicha clase. Pero sobre todo, demuestra que aunque haya margen de mejora es posible desarrollar un sistema de estas características, por lo que sería conveniente seguir investigando en esta dirección.

Palabras clave: redes neuronales convolucionales, inteligencia artificial, aprendizaje computacional, aprendizaje profundo, cáncer, cáncer de piel, melanoma, keras, tensorflow

Abstract

Skin cancer is the fourth most common type of cancer in the world, the first in many countries and, in addition, its incidence is expected to increase in the coming decades. Most of them have low mortality, but a small part, melanomas, have a higher mortality. Even so, these present a survival rate close to 100 % when it is detected in the early stages. However, the diagnosis of these injuries is complicated since these pathologies are not easily identifiable visually, so a system of convolutional neural networks capable of correctly classifying this type of injury from images would be very useful.

In this line, a classifier based on an Efficient-NetB0 convolutional network has been developed, varying different training and network parameters to reach a final model. This network is capable of classifying images of 8 main pathologies, among which are the main types of skin cancer and other diseases. In addition, it consists of another class that represents all the pathologies not included in the previous 8, so that the model can recognize when an image is unknown. In the work we have focused on the behavior of the model in relation to this class.

The results, although their limitations must be taken into account, are positive, both generally and specifically in the class named above: 74 % overall precision and a sensitivity of 66 % for the unknown class. But above all, it shows that although there is room for improvement, it is possible to develop a system with these characteristics, so it would be advisable to continue researching in this direction.

Keywords: convolutional neural networks, artificial intelligence, machine learning, deep learning, cancer, skin cancer, melanoma, keras, tensorflow

Índice

1. Introducción	1
2. Problema	5
3. Contexto	7
3.1. Inteligencia Artificial y Aprendizaje Profundo	7
3.2. Redes Neuronales Convolucionales	9
3.2.1. Capas Convolucionales	10
3.2.2. Capas Pooling	11
3.2.3. Clasificador	13
3.3. TensorFlow, Keras y Google Colab	14
3.4. Estado del arte: IA en medicina y dermatología	14
3.5. Enfermedades del problema	16
4. Datasets	19
5. Métodos	23
5.1. Entrada de datos, preprocesado y aumento	23
5.2. Arquitectura de la red	24
5.2.1. Transfer Learning	24
5.2.2. Arquitectura usada	25
5.2.3. Prevención del sobreajuste	26
5.3. Optimizador y función de pérdida	28
5.4. Evaluación del modelo	28
6. Resultados	31
6.1. Variación en tasa de aprendizaje	31
6.2. Aumento de capas entrenables	32
6.3. Introducción de regularización L2	33
6.4. Reducción de neuronas	34

6.5. Incorporación de datasets	36
7. Discusión	39
8. Conclusión	41
A. Especificaciones técnicas de Keras	43
B. Tabla completa de resultados	44
Referencias	47
Fuentes de las figuras	51
Fuentes del dataset <i>Unknown</i>	53

1. Introducción

El cáncer de piel se clasifica en dos tipos principales: melanomas, que representan un porcentaje muy pequeño de los casos de cáncer de piel (el 1% en EE.UU. [1]) pero la mayoría de las muertes, y no-melanomas, que, al contrario, son muy frecuentes pero presentan una mortalidad baja. A diferencia del melanoma y de todos los demás tipos de cáncer, el número exacto de casos de no-melanoma no se conoce con exactitud debido a que el reporte de casos es muy dispar entre los distintos países o incluso inexistente en algunos como Estados Unidos [2]. A pesar de ello, sí se conoce que tienen una gran incidencia, haciendo al cáncer de piel uno de los más frecuentes incluso estando probablemente subestimado debido a esta deficiencia en el seguimiento.

A nivel global, el cáncer de piel es el cuarto tipo de cáncer más común (figura 2) y, en países o poblaciones mayoritariamente de piel blanca como EE.UU., Reino Unido o Australia es, por mucho, el tipo de cáncer más frecuente. [3] [4] [5]

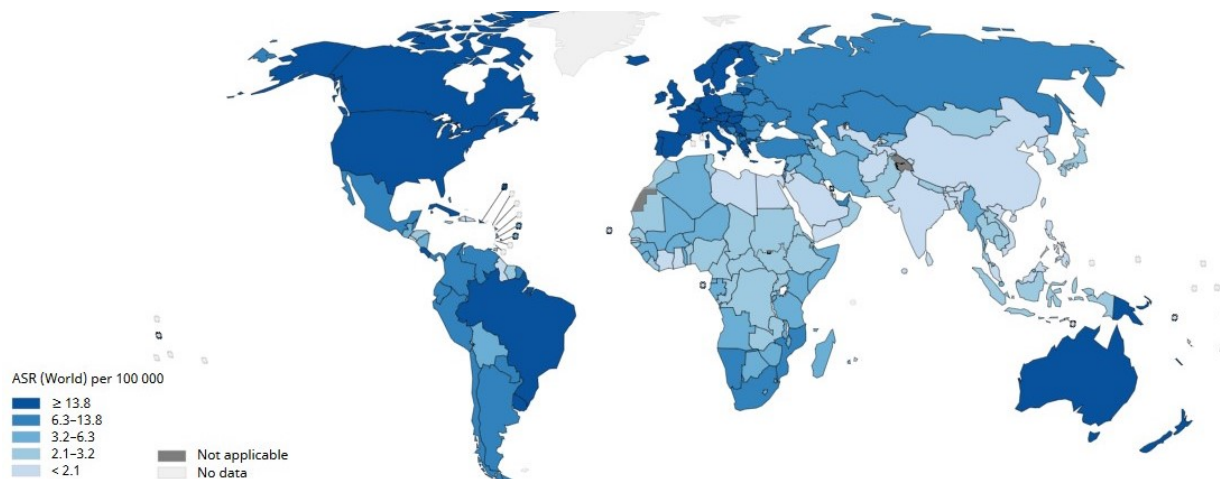


Figura 1: Casos de cáncer de piel por cada 100.000 habitantes [1]

En España, según la Asociación Española Contra el Cáncer (AECC), para 2020 se esperan 27.616 nuevos casos, de los cuales 22.172 corresponden a no melanomas y 5.443 a melanomas. Esto lo convierte en el quinto tipo de cáncer más común a nivel estatal (figura 3) [6]. En cuanto a mortalidad, este año se esperan 1.700 fallecidos por cáncer de piel (el número 16 en

cuanto a número de muertes) [7].

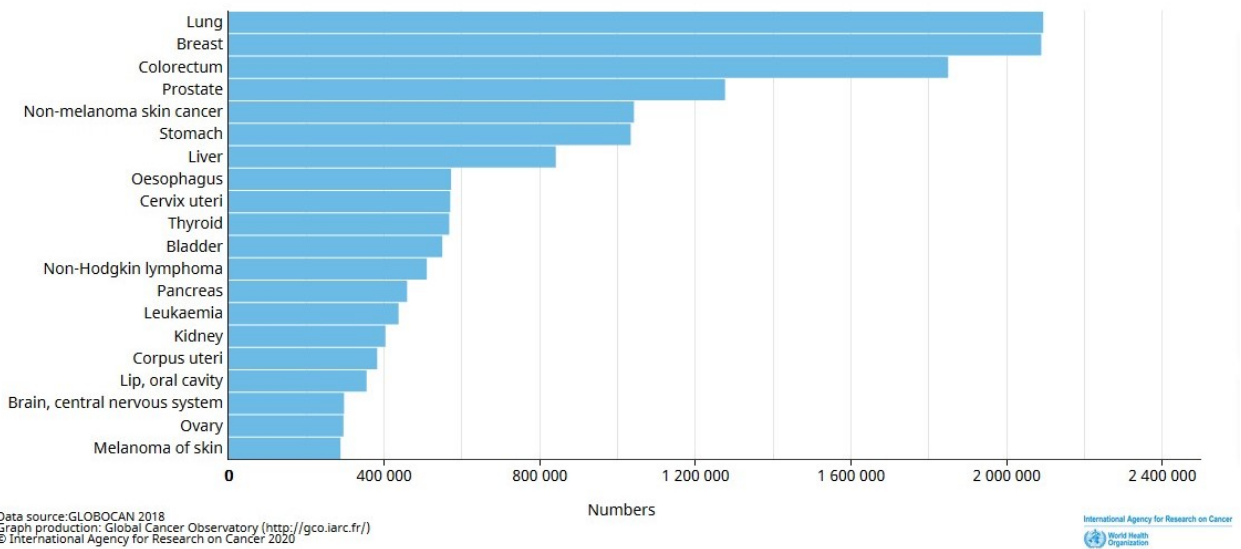


Figura 2: Número estimado de nuevos casos de cáncer en 2020 en el mundo [1]. El cáncer de piel es el cuarto más común (sumando melanoma y no-melanoma).

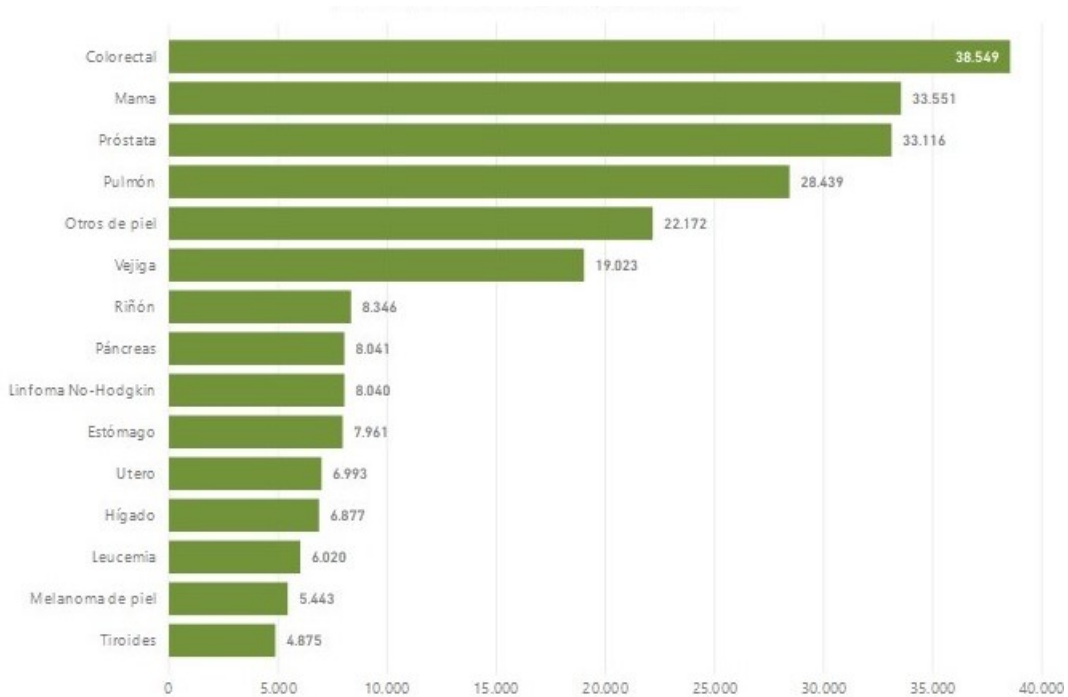


Figura 3: Nuevos casos detectados al año en España por tipo de cáncer [2]

A pesar de los esfuerzos por concienciar a la población sobre los riesgos de la exposición

solar, la mayor parte aún toma el sol entre las 12 y las 16 horas, se quema la piel cada año o utiliza crema de FPS 25 o menor, que son algunas de las principales acciones a evitar [8]. Esto, sumado al envejecimiento de la población, hace que la incidencia del cáncer de piel aumente un 10 % cada año y que se estime que en 2040 se llegará a la cifra de 40.900 nuevos casos, 34.439 no melanomas y 6.461 melanomas, convirtiéndose así en el tercer tipo de cáncer más común en España [6]. También se observa esta tendencia ascendente a nivel mundial, ya que en 2040 habrá un 84 % más de casos que en 2018 [9].

Precisamente debido a esta gran incidencia y al aumento que se producirá en los próximos años, es necesario mejorar los métodos de detección y diagnóstico del cáncer de piel ya que, en realidad, la probabilidad de sobrevivir a un cáncer de piel cuando se detecta en una fase temprana es bastante alta. Para el melanoma en particular, la tasa de supervivencia para un tumor que aún se encuentra localizado es del 99 %, del 65 % en aquellos que ya han empezado a expandirse hacia debajo de la piel y del 25 % en los que ya se ha expandido a otras partes del cuerpo [10].

Sin embargo, la detección temprana de estas enfermedades no es sencilla, ya que en primera instancia se basa en una inspección visual y las características morfológicas que presentan los distintos cánceres de piel pueden ser muy similares entre sí, así como similares a otras lesiones benignas. En este contexto, el uso de algoritmos de reconocimiento de patrones basados en técnicas de inteligencia artificial cobra una especial relevancia, ya que permitiría a los médicos detectar el cáncer de piel en fases aún más tempranas de lo que hoy día son capaces, así como a agilizar el proceso ante el aumento de pacientes que se prevé en el futuro.

2. Problema

El problema a resolver en este trabajo es un problema de clasificación de imágenes cutáneas en 9 categorías a partir de redes neuronales convolucionales.

Estas categorías representan las enfermedades cutáneas cancerosas y precancerosas más comunes y otras que, por su similitud, a menudo son confundidas con estas. Además, se incluye una última categoría *outlier*, que abarcaría todas aquellas patologías que no entren en ninguna de las anteriores. Las categorías, con las siglas con las que nos referiremos a ellas, son las siguientes: melanoma (MEL), nevus melanocítico (NV), carcinoma de células basales (CCB), queratosis actínica (QA), queratosis benigna (QB), dermatofibroma (DF), lesión vascular (VASC), carcinoma de células escamosas (CCE) y la última categoría, en donde entrarían las imágenes que no entran dentro de las 8 enfermedades principales, a la que denominamos categoría *unknown* (UNK). Cada imagen pertenece a (y será clasificada en) solo una de estas 9 categorías.

En particular, en este trabajo nos hemos centrado en la capacidad que tienen los modelos probados para detectar correctamente las imágenes de la categoría *unknown* (UNK). Esto tiene una dificultad añadida ya que, en contraste con las demás categorías, la variabilidad de las imágenes que pertenecen a esta clase es muy alta por no pertenecer a ninguna patología concreta, lo que dificultará a la red neuronal la adquisición de unos patrones que le sirvan de identificación.

La categoría *unknown* hace que el problema sea más parecido a un caso clínico real en el que el modelo desarrollado fuese usado para detectar el cáncer. En dicho contexto, habrá ocasiones en las que se evalúe una lesión que en realidad no pertenece a ninguna de las patologías principales. Si en esta situación no existiera la categoría *unknown*, el modelo daría una predicción errónea o en todo caso una predicción difusa entre las categorías, por lo que lo más correcto sería predecir que es una patología distinta de las demás.

3. Contexto

3.1. Inteligencia Artificial y Aprendizaje Profundo

Podríamos definir la Inteligencia Artificial (IA) como aquellas técnicas y sistemas informáticos capaces de realizar tareas que normalmente requieren de inteligencia humana, como la percepción visual, la toma de decisiones o el reconocimiento del habla.

Al comienzo de su desarrollo, estaba basada en técnicas que no usaban el aprendizaje, sino que seguían un conjunto de reglas establecidas para una determinada tarea, como por ejemplo jugar al ajedrez. Sin embargo este enfoque se quedaba corto para tareas más complejas y surgió el subcampo que hoy conocemos como Aprendizaje Computacional (o Machine Learning), en el que un sistema es capaz de determinar por sí solo las reglas necesarias para llevar a cabo un problema sin estar específicamente programado para esa tarea.

Dentro del Aprendizaje Computacional podemos distinguir entre aprendizaje supervisado, no supervisado y aprendizaje por refuerzo. En el supervisado se le suministran al sistema los datos y las salidas deseadas a dichos datos, como por ejemplo las imágenes y la categoría a la que pertenecen; en el no supervisado el sistema extrae por sí solos las categorías existentes en el problema a partir de las diferencias que encuentra entre los datos; y en el aprendizaje por refuerzo esto es aprendido en base a la retroalimentación que recibe del entorno.

Además, dentro del Aprendizaje Computacional se distingue otro subcampo, conocido como Aprendizaje Profundo (o Deep Learning), una serie de algoritmos basados en redes neuronales artificiales, las cuales son capaces de extraer patrones de datos y aprender a realizar tareas a partir de ellos. En este trabajo usaremos Aprendizaje Profundo supervisado por medio de redes neuronales convolucionales. A continuación se define lo que es una red neuronal artificial, mientras que las redes neuronales convolucionales se definirán en el siguiente apartado.

Una red neuronal artificial (figura 4) es un modelo computacional que se asemeja a las

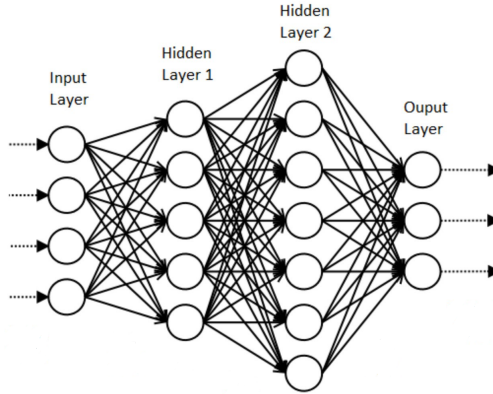


Figura 4: Red neuronal [3]

sinapsis neuronales reales y que consiste en un conjunto de neuronas conectadas entre sí. Estas neuronas están organizadas en capas, en donde cada neurona recibe información de neuronas de capas anteriores, la procesa y la propaga a neuronas de la siguiente capa. Cada neurona asigna un peso a cada una de sus entradas, que son las salidas que le llegan de la capa de neuronas anterior. Como vemos en la figura 5, este peso (w_i) multiplica esa información que le llega de cada neurona (x_i) y después se suman todas las entradas más un valor de sesgo o *bias* (b). Hasta aquí, las neuronas manejan la información de una manera lineal, pero en última instancia se introduce una función no lineal llamada función de activación (f), que suele ser una función ReLU (figura 6) o una función sigmoide (figura 7). Este elemento no lineal es muy importante para la red neuronal ya que le permite encontrar correlaciones complejas entre los datos, lo que le da la capacidad de realizar las difíciles tareas para las que son usadas o extraer patrones que los humanos no somos capaces de ver.

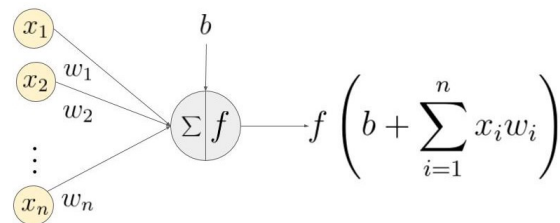


Figura 5: Salida de una neurona [4]

El mecanismo que les permite aprender a las redes neuronales de aprendizaje supervisado

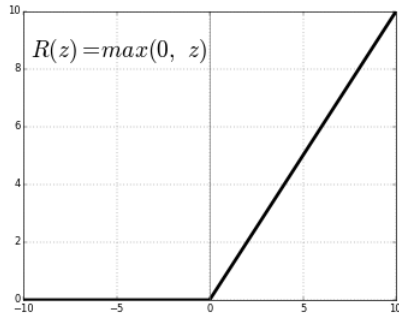


Figura 6: Función ReLU [5]

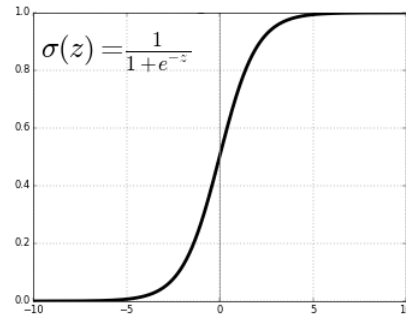


Figura 7: Función sigmoide [5]

es el conocido como algoritmo de retropropagación. Cada red neuronal tendrá una forma de medir cómo de preciso es su funcionamiento, para lo que se definirá una función de error, llamada función de pérdida. Esta función de pérdida determina el error entre los resultados deseados (que se le han suministrado a la red) y la predicción que la red ha hecho para esos mismos datos. Además, la red consta de otra función denominada función gradiente u optimizador. Este optimizador calcula el gradiente de la pérdida en función de los pesos de la red. El sentido en el que calculan la pérdida en función de los pesos va desde las capas superiores, que son las que dan el resultado final de la red, hasta las inferiores, que son por donde entran los datos; es decir, en sentido contrario al que la red es recorrida por los datos.

Por tanto, el proceso seguido durante el entrenamiento es el siguiente (figura 8): la red hace una predicción de los datos que le son suministrados, la función de pérdida calcula el error entre estas predicciones y los resultados reales, el optimizador determina cómo se puede disminuir el error en función de los pesos y, por último, se corrigen los pesos aumentándolos o disminuyéndolos, en función de lo que el optimizador haya determinado para cada uno. Finalmente, este proceso se repite hasta que la red alcance el mínimo error posible, es decir, cuando ya no muestra mejoría.

3.2. Redes Neuronales Convolucionales

Las redes neuronales que vamos a usar son las llamadas Redes Neuronales Convolucionales (CNN, del inglés Convolutional Neural Networks), un tipo de red que se utiliza mayoritaria-

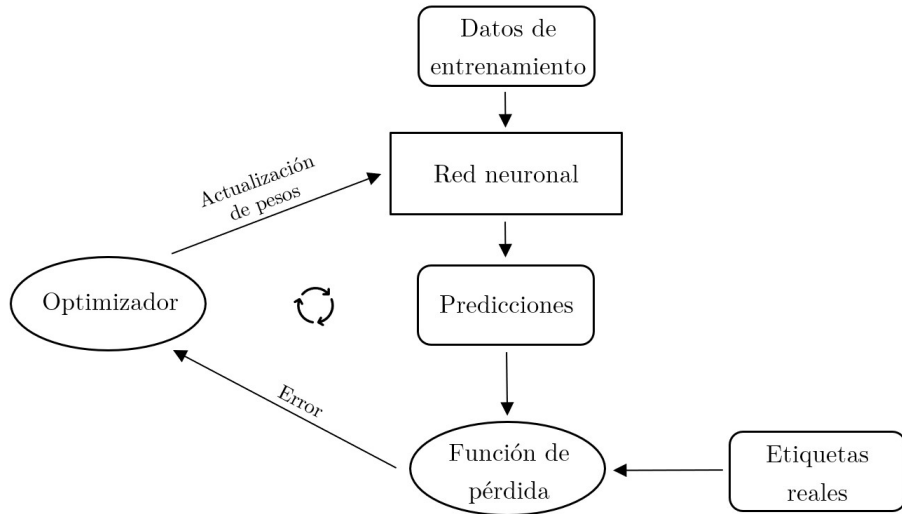


Figura 8: Proceso de entrenamiento de la red neuronal

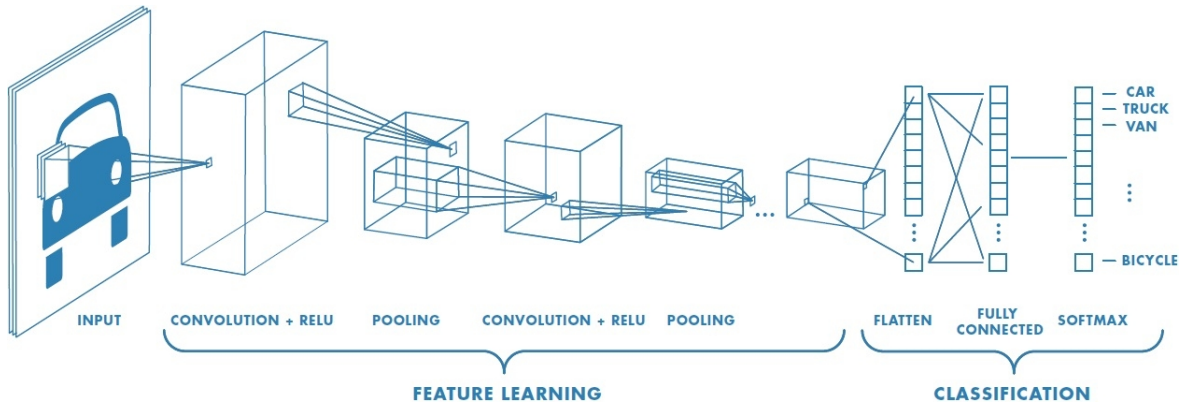


Figura 9: Red neuronal convolucional [6]

mente para tareas de análisis de imagen y de señal. La razón principal por el que se usan este tipo de redes en dichas tareas es que permiten observar patrones locales en una imagen, de forma que si lo que buscamos es un objeto la red lo detectará independientemente de en qué parte de la imagen se encuentre. En contraste, si usáramos una red densa, que son las explicadas anteriormente, solo podríamos encontrar patrones generales en la imagen, es decir, cuando el objeto que estamos buscando estuviese en un sitio determinado de la imagen.

3.2.1. Capas Convolucionales

Estas son las capas principales que extraen los patrones de interés. Cada capa consta de uno o varios filtros de imagen (los tres pequeños *kernels* de la figura 10), de manera que

cada uno de los valores matriciales de estos filtros son los pesos a entrenar. De esta forma, cuando estos filtros hacen una convolución a través de toda la imagen, la neurona encargada de analizar un determinado píxel no solo analiza dicho píxel, sino que también recibe información de los píxeles de alrededor, que es lo que facilita encontrar los patrones locales comentados anteriormente. Lo que hace cada uno de estos filtros es resaltar una característica determinada de la imagen, de forma que cada capa cada vez buscará unas características más complejas basadas en las que ha inspeccionado la capa anterior (figura 11).

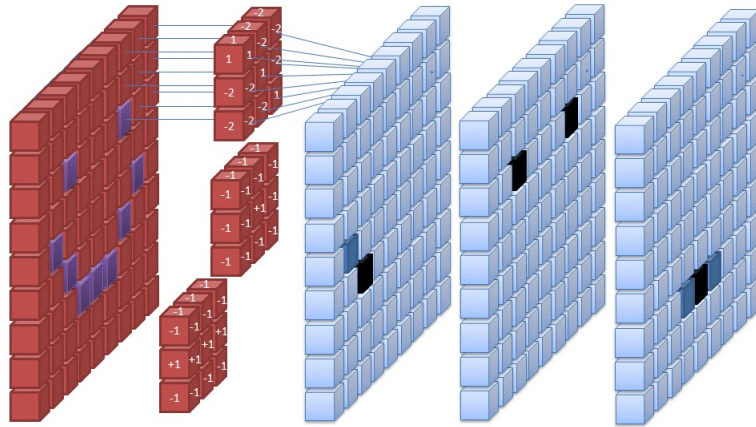


Figura 10: Capa convolucional con solo 3 filtros [7]

Así, la salida de una capa convolucional será un tensor con las diversas versiones de la imagen resultantes de haber aplicado cada filtro, que en la figura 10 estaría compuesto por las tres matrices azules. Estas versiones son lo que conocemos como mapas de características o *feature maps*.

3.2.2. Capas Pooling

El objetivo de estas capas es reducir la información de la imagen y lo hacen reduciendo su tamaño. Al igual que las capas convolucionales, lo hacen también por medio de una convolución, de modo que recorren la imagen píxel a píxel y escogen o bien el valor medio de

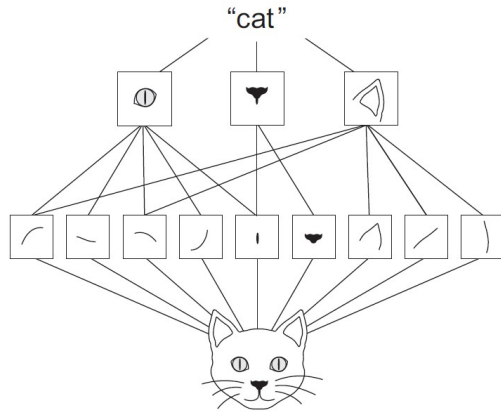


Figura 11: Las capas más profundas buscan características más complejas a partir de las de capas anteriores. [8]

entre dicho píxel y los de su alrededor, o el valor máximo (figura 12).

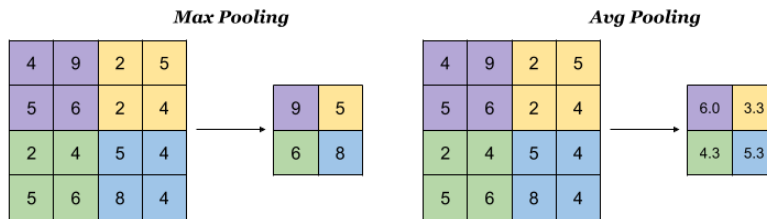


Figura 12: Capa pooling [9]

Por su parte, las capas convolucionales también reducen las dimensiones de la imagen en cada capa (a menos que, precisamente por evitarlo, se aplique un *padding* [figura 13]). Por tanto, la acción conjunta de las capas convolucionales y las *pooling* hace que las dimensiones de la imagen se vayan reduciendo progresivamente, de manera que la información que no es relevante se va desechando para quedarse solo con aquellas características que sí son determinantes, tal y como se puede observar en la figura 14. Esta capacidad de abstracción para seleccionar solo los datos realmente relevantes es lo que hace a las CNN una gran herramienta para el análisis de imágenes.

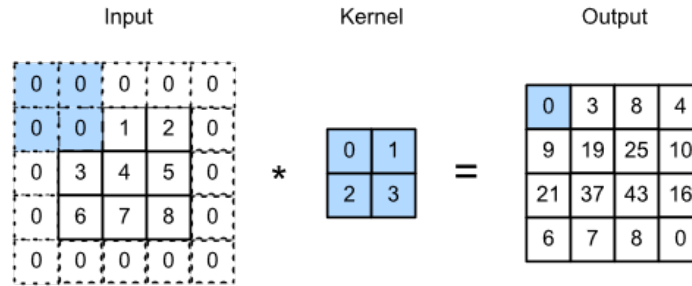


Figura 13: El *padding* evita que las dimensiones se reduzcan tras una convolución. [10]

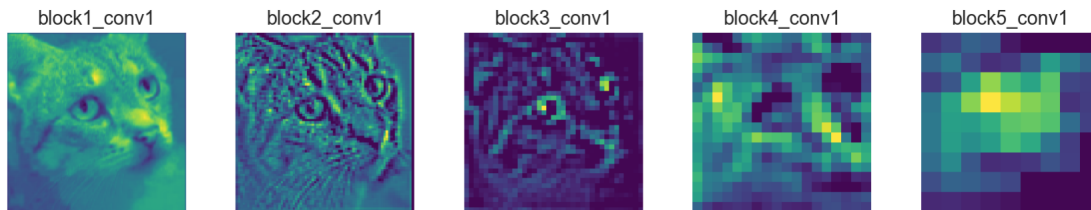


Figura 14: Mapa de características de una imagen conforme va penetrando en una CNN: sus dimensiones se reducen y solo se mantienen las características más importantes. [11]

3.2.3. Clasificador

Una vez ha terminado la parte convolucional de la red, que ha permitido extraer todas las características de la imagen, se aplana la imagen o el tensor resultante por medio de una capa denominada *flatten*. Esto se hace para poder introducir la imagen en las últimas capas, que son capas densas como las que se explicaron en un primer momento. Estas capas tienen como finalidad clasificar finalmente la imagen a partir de las características encontradas por las capas convolucionales.

Por último, la última capa consta de una función de activación *softmax*. Esta función (ecuación 1) transforma los valores de salida de las nueve últimas neuronas (x_i en la ecuación) en un vector de probabilidades de forma que las probabilidades de las nueve clases suman 1. Esto permite tener un resultado mucho más interpretable, pues la imagen será diagnosticada con la clase que presente mayor probabilidad. Por el contrario, si usáramos una función sigmoide no podríamos interpretar las salidas de la red como probabilidades.

$$softmax(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}} \quad (1)$$

3.3. TensorFlow, Keras y Google Colab

Este trabajo se ha desarrollado en Python con las librerías TensorFlow y Keras. Keras es una API de alto nivel en código abierto para el desarrollo de redes neuronales. Esta funciona sobre diversos *backends* encargados de realizar todo el cálculo diferencial y de tensores, que en nuestro caso será TensorFlow, una librería de computación numérica para Aprendizaje Computacional, también en código abierto y desarrollada por Google.

Además, a falta de un ordenador con GPU (necesaria dada la gran cantidad de cálculos que se realizan al entrenar los modelos), este trabajo se ha implementado en el entorno Google Colab, el cual permite usar GPUs en la nube.

3.4. Estado del arte: IA en medicina y dermatología

Durante los últimos años, la aplicación de la IA en diversas áreas han hecho que surjan tecnologías que hasta hace poco nos parecían imposibles, como los coches autónomos o el reconocimiento facial. Por su parte, aunque a un ritmo más lento por motivos como la poca disponibilidad de datos o la estricta regulación a la que están sujetos los productos sanitarios, también en la medicina se están dando pasos para incorporar esta tecnología.

Algunas de las áreas médicas en las que la influencia de la IA está aumentando son el diagnóstico, la monitorización de pacientes o el descubrimiento de nuevos fármacos. Dentro del área de diagnóstico, en donde se enmarca este trabajo, las disciplinas en las se ha conseguido mayor avance son aquellas que tienen una gran relación con las imágenes, como son la radiología, dermatología, oftalmología y patología. De hecho, en estas áreas ya se han desarrollado sistemas de diagnóstico con mayor precisión que la de doctores humanos [11] [12] [13].

En particular, en dermatología destaca el trabajo de *A. Esteva et al. (2017)* [14], en el que se entrenó una CNN con 129.450 imágenes clínicas de 2.032 enfermedades cutáneas diferentes, el dataset de imágenes dermatológicas más grande usado, al menos hasta ese momento. Este

sistema logró tener una precisión similar a la de 21 dermatólogos en el diagnóstico del cáncer de piel. Algunos otros, como *Y. Fujisawa et al. (2019)* [15] han logrado también buenos resultados con un dataset pequeño (4867 imágenes), obteniendo una precisión incluso superior a la de un grupo de 13 dermatólogos (92,4% y 85,3%, respectivamente). En cambio, en *A. Hekler et al. (2019)* [16], el mejor resultado en el diagnóstico del cáncer de piel se obtuvo cuando se usaron conjuntamente una CNN y un dermatólogo: la precisión fue de 82,95%, un 1,36% más que el mejor de los dos métodos por separado (el 81.59% conseguido por la CNN). Por último, en *Y. Liu et al. (2020)* [17] obtuvieron una precisión en el diagnóstico de 26 enfermedades cutáneas similar a la de dermatólogos (66% y 63%) y mayor que la de médicos de atención primaria y enfermeros (44% y 40%), recalcando las implicaciones que esto podría tener en áreas con menos recursos, donde la mayoría de los casos recaen sobre estos últimos.

Debido a esta gran precisión de las CNN en el diagnóstico de imágenes, algunos incluso ven factible que la IA termine reemplazando a algunos médicos, especialmente a los radiólogos. Sin embargo, esto es complicado ya que en realidad los médicos hacen otras tareas aparte de interpretar imágenes, como por ejemplo consultar tratamientos con otros especialistas o interpretar las imágenes teniendo en cuenta el historial clínico del paciente, por no hablar del factor humano en ámbitos como la comunicación o la empatía con los pacientes [18]. Además, no está claro que los resultados de estos estudios sean extrapolables a unas condiciones clínicas reales [19], así como tampoco muestran una buena generalización entre distintos datasets o entre grupos de población distintos. También se deben resolver algunos aspectos éticos, como quién sería el responsable de un diagnóstico erróneo, ya que los sistemas de IA suelen funcionar como una caja negra en la que no es posible saber en qué se ha basado la CNN para dar uno u otro diagnóstico [20]. Por estas razones, una respuesta bastante común a este debate es que serán los médicos que usen IA los que reemplazarán a los que no.

En cualquier caso, la IA se perfila ya como una gran herramienta para la medicina clínica, pues aparte de aumentar la precisión, aumentará la eficiencia de los ya sobrecargados sistemas de salud, facilitando también el acceso a la atención médica en sitios donde cuenten con pocos

profesionales.

3.5. Enfermedades del problema

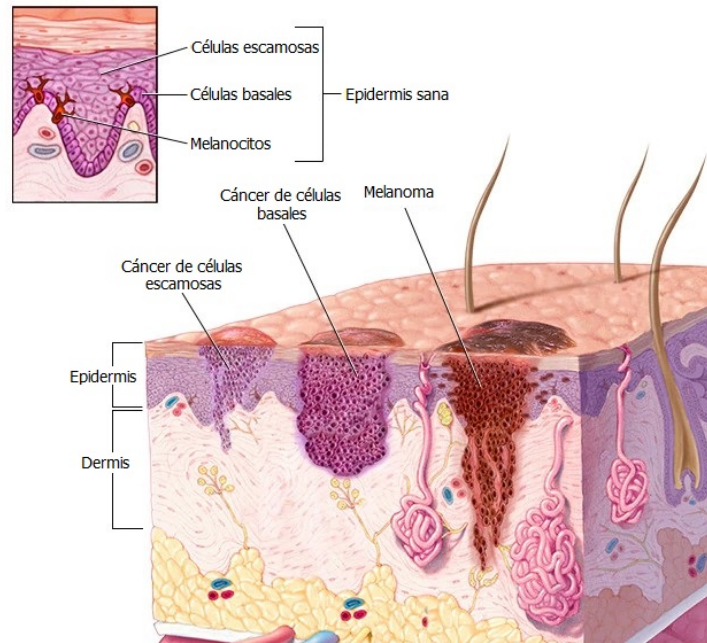


Figura 15: Vista por capas de los tres tipos de cáncer y las células que los producen [12]

El melanoma (MEL) es un tipo de cáncer que se desarrolla a partir de un crecimiento incontrolado de las células melanocíticas, que son un tipo de célula presente en la epidermis (figura 15) encargada de producir melanina, la sustancia que broncea la piel y la protege de las radiaciones ultravioleta. Estas células tienen un crecimiento bastante más rápido que el de las demás células potencialmente cancerosas de la epidermis, por lo que son más propensas a invadir los tejidos de alrededor y finalmente propagarse a otras partes del cuerpo, motivo por el cual el melanoma es el cáncer de piel con mayor mortalidad. Aunque la exposición al sol aumenta el riesgo, también se debe a causas genéticas entre otras, por lo que también puede surgir en zonas no expuestas. Como se ha comentado en la introducción, este es el menos común de los cánceres de piel. Sin embargo, aún siendo muy minoritario en comparación con los demás (en torno al 1% en EE.UU. [1], el 16% en España [6]), su alta mortalidad hace que el número absoluto de muertes debido al melanoma sea mucho mayor que el de cánceres no melanoma (en España, 1.003 fallecidos al año frente a 698 [7]). Esto hace que sea el de

mayor interés a la hora de perfeccionar su diagnóstico.

El nevus melanocítico (NV) es lo que conocemos como lunares, es decir, una concentración benigna de células melanocíticas. Por lo general, se dice que la principal diferencia entre un lunar y un melanoma es que en este último las características morfológicas (como los bordes, el color o la variación en el tiempo) son muy irregulares. Sin embargo, esto no siempre se cumple y además hay una enorme variedad de tipos nevus, así que las características de los lunares también pueden ser muy variables. Además, los nevus pueden volverse malignos con el tiempo, lo que complica aún más la tarea de identificar un melanoma. Por esta razón, es muy importante saber distinguir nevus y melanomas correctamente.

Después, nos encontramos con el cáncer de células basales (CCB). De cada 10 casos de cáncer no melanoma, 8 corresponden a este tipo, por lo que es el más común de los cánceres de piel [2]. Estos tienen bastante relación con la exposición al sol y por tanto suelen aparecer en áreas expuestas, como la cara, cabeza y cuello. Las células basales son las que están justo debajo de la epidermis (figura 15) y cuando son cancerosas su crecimiento es lento, por lo que es raro que invadan otras partes del cuerpo, de ahí su baja mortalidad.

En cuanto al cáncer de células escamosas (CCE), también está muy relacionado con la exposición al sol por lo que aparece en zonas similares al de células basales además de en otras zonas como heridas recurrentes o cicatrices. Aunque las células escamosas se encuentran justo encima de las basales (figura 15), y por tanto en una zona un poco más externa, son un poco más propensas a crecer hacia dentro y propagarse debido a que su crecimiento es más rápido. Por esta razón, su mortalidad es algo más elevada que el de células basales.

Por otro lado tenemos la queratosis actínica (QA), una lesión precancerosa también provocada por un exceso de exposición al sol. Forma parte de la ruta natural de un cáncer de células escamosas, cuando las células están empezando a replicarse, por lo que es difícil saber si desembocará finalmente en un cáncer o si la lesión no irá a más. Por tanto, es importante saber diferenciar cuanto antes un cáncer de células escamosas de un caso de queratosis actínica.

Respecto a la queratosis benigna (QB), esta puede ser resultado de diversas patologías, como la queratosis seborreica o el lentigo solar. Estas son proliferaciones de diversas células de la epidermis como las basales o las melanocíticas, pero que no derivan en cáncer.

Por último, tenemos el dermatofibroma (DF) y la lesión vascular (VASC). El primero es un tipo de tumor benigno de crecimiento lento, formado en la dermis a partir de células fibrosas, mientras que la lesión vascular engloba diversas patologías que tienden a formar un tumor vascular benigno formado a partir de pequeños vasos sanguíneos, lo que le da un color rojizo.

Por lo general, en cualquiera de estas patologías se pueden encontrar casos que se asemejen a cualquiera de las demás e incluso a otras que no están aquí. Esto hace patente la gran utilidad que podría tener una herramienta de análisis de imagen que permita diferenciar estas patologías incluso en casos que puedan dar lugar a confusión, y en especial a identificar el melanoma por ser la de mayor mortalidad.

4. Datasets

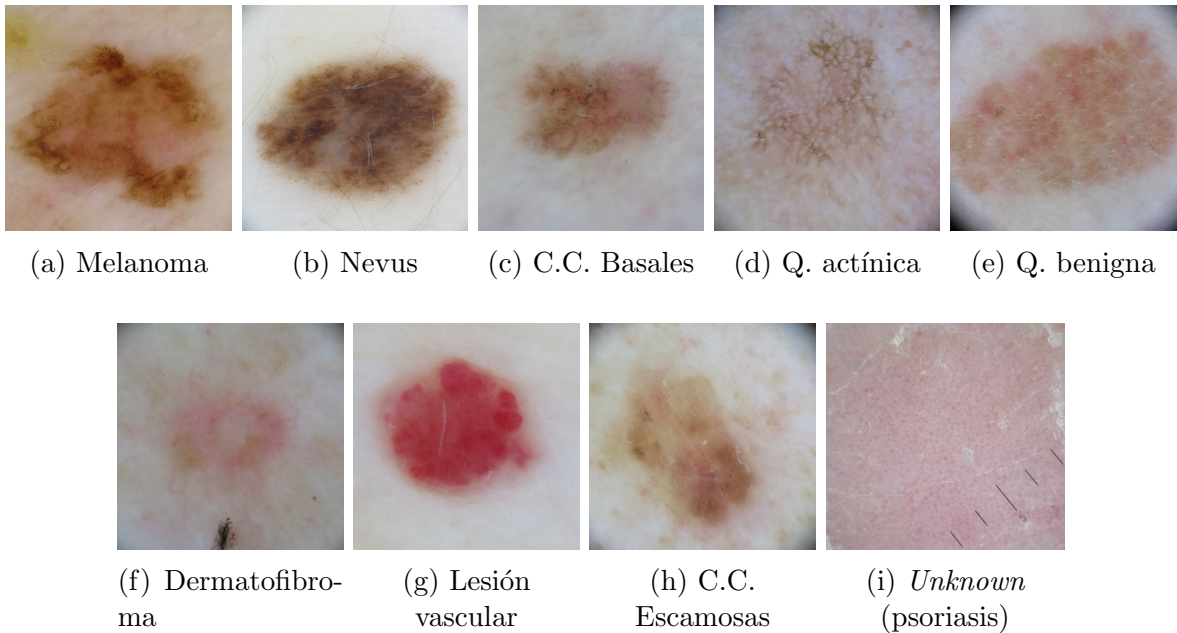


Figura 16: Muestra de las imágenes del problema

En este trabajo usaremos imágenes de 5 datasets distintos, los cuales se describen a continuación. No obstante, la utilización de estos no será simultánea, sino que se empezarán usando solo algunos, para después ir añadiendo los demás y ver si su utilización beneficia o perjudica a los modelos computacionales usados.

Cada vez que se entrene un modelo, el 15% de los datos de entrenamiento se usará como conjunto de validación. Este conjunto de validación servirá para evaluar la precisión del modelo a medida que transcurre el entrenamiento y, a su vez, compararlo con la respectiva evolución en los datos de entrenamiento.

Casi todas las imágenes son imágenes dermatoscópicas, es decir, tomadas con un dermatoscopio, un instrumento usado por los dermatólogos similar a una lupa, que además elimina la luz que refleja la piel, permitiendo visualizar capas más profundas de la lesión y muchas más características que a simple vista.

Dataset	MEL	NV	CCB	QA	QB	DF	VASC	CCE	UNK
ISIC 2019	4.522	12.875	3.323	867	2.624	239	253	628	0
Unknown	0	0	0	0	0	0	0	0	355
7 point	252	575	42	0	85	20	29	0	9
SD	267	0	1.407	570	872	139	0	0	11.382
Skin L2	48	131	53	0	57	14	41	0	49

Tabla 1: Distribución de los datasets usados entre las 9 clases

■ ISIC

Precisamente ante el gran potencial que tiene el uso de la inteligencia artificial en el diagnóstico de enfermedades cutáneas y del melanoma en particular, surge el Proyecto ISIC [21], el cual cada año organiza una competición de clasificación automática de imágenes cutáneas para la detección del melanoma. El archivo de imágenes de esta organización es, actualmente, el mayor banco de imágenes de este tipo, por lo que se ha convertido en el principal referente de datos para la investigación de nuevos métodos automáticos de detección de este tipo de enfermedades. Por esta razón, la base de imágenes de este trabajo será el dataset publicado en la competición ISIC 2019 [22].

Este está compuesto por imágenes cutáneas clasificadas en 8 de las categorías en las que basamos nuestro problema: melanoma, nevus melanocítico, carcinoma de células basales, queratosis actínica, queratosis benigna, dermatofibroma, lesión vascular y carcinoma de células escamosas.

Está formado por 25.331 imágenes. En la Tabla 1, se muestra el desglose de imágenes de los distintos datasets en las 9 categorías.

Como se puede advertir, algunas clases están muy desbalanceadas respecto a otras, lo que en parte complica el problema. Esto se intentará resolver gracias a la suma de otros datasets y a partir de *image augmentation*.

- **Unknown**

Este dataset agrupa imágenes únicamente de la última categoría de nuestro problema. Las imágenes han sido recopiladas por el autor de este trabajo de diversas páginas web especializadas en dermatología y dermatoscopia, así como de artículos científicos. Está compuesto de 355 imágenes dermatoscópicas de diversas patologías cutáneas muy comunes, como dermatitis, psoriasis, lupus, verrugas, molusco contagioso o sarna entre algunas otras. Todas las fuentes de origen de este conjunto de datos pueden ser consultadas en el apartado correspondiente de la bibliografía.

Junto al anterior dataset, constituye la base de este trabajo, a la que se irán añadiendo los datasets que se explican a continuación, tal y como se ha explicado al principio de este apartado. Además, se reservará un 15% tanto de este dataset como del anterior (ISIC) para conformar el conjunto de test, con el que evaluaremos todos los modelos, mientras que los datasets que vienen a continuación se usarán solo como datos de entrenamiento (y validación).

- **7 point** [23] [24]

Este conjunto de datos agrupa 1.011 imágenes pertenecientes a 17 categorías que, a efectos de aplicación en nuestro problema, se pueden distribuir según la tabla 1.

- **Skin L2** [25] [26]

Este dataset está formado por 389 imágenes pertenecientes a 8 categorías. El correspondiente reparto entre nuestras categorías se puede ver en la tabla 1.

- **SD** [27] [28]

A diferencia de los demás datasets, este contiene imágenes clínicas en vez de dermatoscópicas, es decir, hechas con cámara en vez de un dermatoscopio. Consiste en dos

versiones: la primera, el SD-198, contiene 6.584 imágenes clínicas de 198 enfermedades cutáneas distintas, y el segundo, el SD-260, consiste en 12.582 imágenes de 260 enfermedades cutáneas. Al juntar ambas versiones, se han eliminado imágenes duplicadas y también aquellas enfermedades que no se podían adaptar a nuestro problema o cuyo diagnóstico era poco concreto. También, como consecuencia del gran número de imágenes de nevus (categoría NV), se ha decidido no incorporar las imágenes de dicha clase para no desbalancear aún más los datos del problema. Como resultado, tenemos 14.637 imágenes repartidas en nuestras categorías según la tabla 1. Además, todas las imágenes tenían una marca de agua en la parte inferior que podía interferir en el modelo, por lo que se implementó un preprocesado en MatLab que recortara una pequeña franja en la parte inferior de todas las imágenes.

5. Métodos

5.1. Entrada de datos, preprocesado y aumento

Al suministrar los datos de entrenamiento a los modelos se aplica un aumento de datos. Esto ayuda a evitar el sobreajuste aumentando la variabilidad de los datos por medio de diversas transformaciones de imagen, de manera que cada vez que se suministre una imagen al modelo se le aplicará una transformación aleatoria dentro de los márgenes que nosotros le suministramos. En nuestro caso, las transformaciones y rangos que se ha aplicado a las imágenes son los siguientes:

- Rotación de hasta 40°
- Desplazamiento lateral de hasta 0,2 del ancho de la imagen
- Desplazamiento vertical de hasta 0,2 del alto de la imagen
- Zoom de hasta 0,2 del tamaño de la imagen
- Volteo vertical
- Volteo horizontal

Además, cada arquitectura consta de su propia función de preprocesado, que normalmente consiste en rescalar la imagen entre otras acciones, por lo que también se le aplica a cada imagen antes de que entre en el modelo.

Al conjunto de test únicamente se le aplicará la función de preprocesado de la arquitectura correspondiente, ya que al ser el conjunto con el que evaluaremos todos los modelos las imágenes deben ser siempre iguales.

En el apéndice A se explica más detalladamente cómo se ha implementado la entrada de datos con Keras.

5.2. Arquitectura de la red

5.2.1. Transfer Learning

En este trabajo no se ha desarrollado una CNN propia, sino que se han usado otras que ya están desarrolladas gracias a lo que se conoce como *transfer-learning*.

Esta estrategia consiste en usar una CNN que ya ha sido entrenada con un enorme conjunto de imágenes, normalmente con numerosas clases de objetos cotidianos como animales, vehículos, ropa, etc. Por ejemplo, uno de los datasets más usados con este propósito es el ImageNet, que consta de 1,4 millones de imágenes de 1000 clases. De esta manera, las características visuales aprendidas por esta red, como bordes o formas desde las más simples a las más complejas, se pueden considerar un conjunto genérico de características que se pueden encontrar en todas las imágenes y que, por lo tanto, se pueden usar para casi cualquier problema. Esta parte del proceso se conoce como *feature extraction* o extracción de características y nos ahorra la parte que tiene mayor coste computacional, que es entrenar las capas convolucionales de la CNN.

Sin embargo, la red entrenada previamente no se carga íntegramente, sino que solo se usa la parte convolucional (figura 17). Como se explicó en la sección 3.2, las características encontradas por la parte convolucional pasan después a la parte que conocemos como clasificador, que es la responsable de clasificar finalmente la imagen en función de dichas características. Por esta razón, el clasificador de la red se sustituye por uno nuevo que sí será entrenado en base a nuestro problema. Esto se debe a que, como se ha comentado anteriormente, las características visuales de la imagen sí son más genéricas y pueden resultar útiles para detectar patrones en base a esas características más básicas. Sin embargo, las asociaciones aprendidas en el clasificador sobre qué características corresponden a cada clase sí son más específicas, así que es necesario entrenar la parte densa de la red en el problema específico a resolver [29].

Además, también se ha probado a ver cómo influye la técnica denominada como *fine-tuning*. Esto consiste en descongelar (es decir, someter al entrenamiento) algunas de las

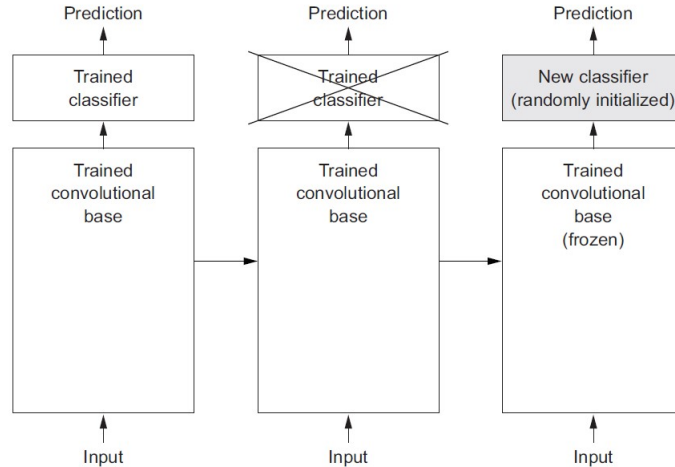


Figura 17: *Transfer learning* [8]

últimas capas convolucionales de la red cargada en el *transfer-learning*. Esto se hace cuando entrenando solo el clasificador, los resultados no son del todo buenos, por lo que es necesario adaptar también al problema las características visuales que buscan las últimas capas convolucionales. La razón de esto es que las características de estas últimas capas son algo más complejas y específicas, por lo que no siempre son generalizables para cualquier problema.

5.2.2. Arquitectura usada

En este trabajo se ha usado la red EfficientNetB0. Dentro de las CNN más usadas para *transfer-learning*, la familia de redes EfficientNet es una de las más recientes. Estas redes tienen dos características principalmente reseñables: la primera es que es una de las redes más precisas hasta el momento, y la segunda es que logran esta gran precisión con un número de parámetros mucho menor comparado con las otras redes frecuentemente usadas, como por ejemplo las ResNet [30] (figura 18). Por lo tanto, como nuestros recursos computacionales son limitados, hemos escogido esta familia EfficientNet y, en particular, la red B0 ya que es la que menos parámetros entrenables tiene.

Por otra parte, como se explicó anteriormente, tendremos que añadir un clasificador a esta red. La base de este serán cuatro capas: una capa *flatten*, una capa densa de 128 neu-

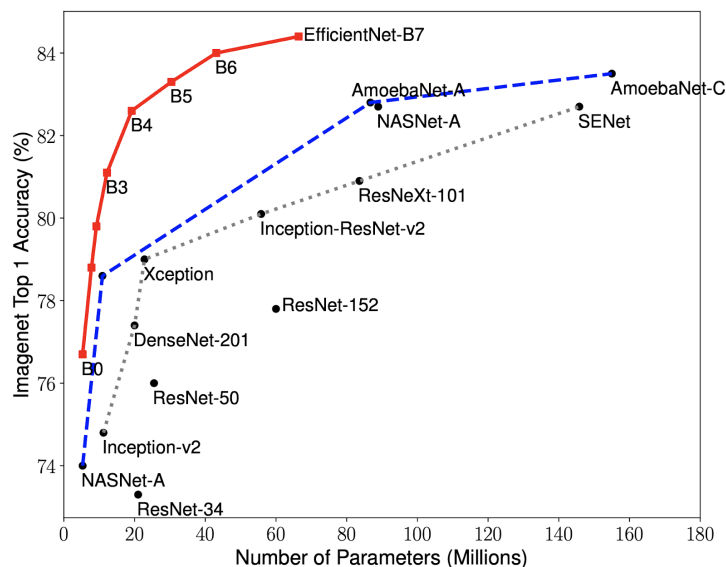


Figura 18: Comparación de la precisión de diversas CNN respecto al número de parámetros [13]

ronas, una capa *dropout* con una tasa de 0.5, y por último una capa densa de 9 neuronas con la función *softmax* que nos devolverá la probabilidad de cada clase. Sin embargo, en el transcurso del trabajo iremos introduciendo diversas modificaciones para ver si mejoran el modelo, como por ejemplo disminuir las neuronas de la primera capa densa, introducir regularización L2 en esta misma capa o aumentar las capas entrenables.

5.2.3. Prevención del sobreajuste

Estas modificaciones que se acaban de nombrar están principalmente destinadas a prevenir el sobreajuste u *overfitting*. Esto se produce cuando el modelo aprende demasiado sobre las imágenes de entrenamiento, de forma que identifica patrones de más. Estos patrones que aprende de más son características muy particulares de las imágenes con las que se está entrenando, pero que realmente no tienen ninguna relación con el objeto que se intenta identificar. Por ejemplo, en nuestro caso se podría deber al dispositivo con el que se obtuvieron las imágenes, por lo que si nuestro modelo aprende los patrones característicos de dicho dispositivo, al intentar aplicárselos a una nueva imagen tomada con un dispositivo distinto obtendrá un peor resultado. Es decir, el modelo no logra generalizar para cualquier imagen. Esto se hace evi-

dente en la curva de entrenamiento, en donde la diferencia entre la precisión de entrenamiento y la de validación aumenta cada vez más, ya que mientras más se especializa el modelo, mejores resultados obtiene en los datos de entrenamiento pero peor en los de validación (figura 19).

De este modo, las capas *dropout* ayudan a prevenir el sobreajuste ya que, durante el entrenamiento, desactivan algunas neuronas en la proporción que se le asigne (figura 20). Así, se evita que todas las neuronas de una capa se fijen en las mismas características de la capa anterior, de modo que la red se verá forzada a no depender solo de unas pocas características.

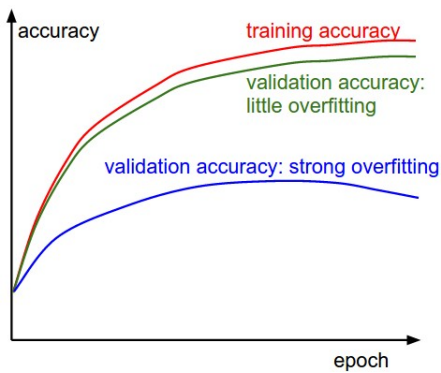


Figura 19: Sobreajuste [14]

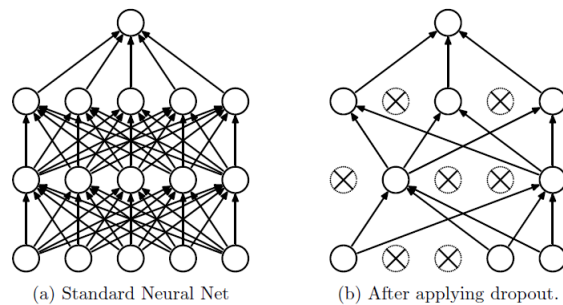


Figura 20: Dropout [15]

Por otro lado, la regularización L2 hace algo parecido. Su funcionamiento consiste en penalizar el error de aquellos pesos que son demasiado grandes, sumando el valor que nosotros le indiquemos. Así, se dificulta de nuevo la excesiva dependencia de algunas características.

Por último, reducir las neuronas reduce el número de patrones que la red es capaz de aprender y, por tanto, le hace ser más selectiva a la hora de seleccionarlos. Esto le obliga a seleccionar la información más relevante y dejar fuera la que no lo es, pero por otra parte también puede ser una desventaja, ya que puede que tenga que dejar fuera algunas características que también serían de utilidad.

5.3. Optimizador y función de pérdida

Como optimizador de la red se ha escogido el optimizador Adam. Los optimizadores constan de una tasa de aprendizaje, la cual hay que ajustar. La tasa de aprendizaje es el tamaño del paso o *step size* con el que el optimizador se mueve por la función de error para actualizar los pesos. Una tasa de aprendizaje mayor hará que el aprendizaje sea más rápido pero también menos preciso, ya que al pasar por el mínimo de la función de error puede que se “salte” el mínimo, haciendo incluso que el modelo no converja nunca si esta tasa es demasiado alta. Por otro lado, una tasa de aprendizaje menor será más precisa, pero también hará que el entrenamiento más lento (figura 21). En nuestro caso, se prueban tasas de aprendizaje de $1e-3$, $1e-4$ y $1e-5$ para ver cuál es la más adecuada.

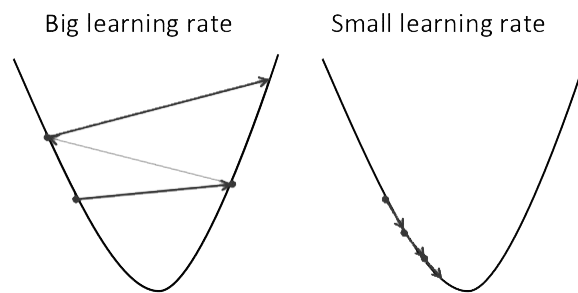


Figura 21: Comparación de tasa de aprendizaje alta y baja [16]

Respecto a la función de pérdida, escogeremos la función *categorical crossentropy*. Esta es la más adecuada para nuestro problema ya que mide el error para problemas de clasificación con múltiples clases.

5.4. Evaluación del modelo

Se tienen diversos indicadores para evaluar los modelos y el entrenamiento. En primer lugar tenemos las gráficas de entrenamiento. Estas muestran la evolución de la precisión y el error del modelo después de cada época del entrenamiento, tanto en los datos del conjunto de entrenamiento como en los datos del conjunto de evaluación, por lo que nos permite saber si hay sobreajuste o prever si el modelo ya no mejorará más aunque se siga entrenando.

Además, como se introdujo en la sección 4 (Datasets), tenemos un conjunto de test para evaluar todos los modelos. De cada uno, obtendremos la predicción que hace para estos datos y la compararemos con las etiquetas reales. A partir de esto, obtendremos la precisión, la pérdida y la matriz de confusión. También obtendremos otros parámetros estadísticos para cada clase como la sensibilidad, especificidad, PPV y NPV, así como la media entre todas las clases de estos mismos valores. Estos parámetros nos permiten saber qué clases se están clasificando mejor y cuáles peor.

Como ya se ha comentado, en este trabajo se hace especial hincapié en la clase UNK, por lo que para comparar los modelos nos hemos fijado, además de en la precisión general del modelo en el conjunto de test, en la sensibilidad de la clase UNK, pues así sabemos en qué medida los modelos son capaces de identificar imágenes de esta clase.

6. Resultados

En esta sección se analiza cómo han ido influyendo los cambios a la red neuronal de base explicada anteriormente en la sección 5.2.2 hasta finalmente alcanzar el mejor modelo. Se comenta en función de los principales parámetros que se han usado para evaluar los modelos (comentados en la sección 5.4), mientras que la tabla completa de resultados se muestra en el apéndice B.

6.1. Variación en tasa de aprendizaje

En primer lugar, se probaron tres tasas de aprendizaje distintas para ver cuál era la más adecuada: $1e-3$, $1e-4$ y $1e-5$. El entrenamiento con cada una de ellas se ejecutó por 20 épocas. Esto dio lugar a los modelos del A al C (tabla 2).

Como se puede observar en la tabla 2, la sensibilidad de la clase UNK es bastante peor en la tasa de aprendizaje más grande, $1e-3$, por lo que esta quedó descartada desde este punto. En cuanto a las otras dos, la de $1e-4$ mostró alguna ventaja sobre la de $1e-5$, pero se optó por no elegir todavía ninguna de las dos para ir probando ambas también junto a las demás modificaciones.

Modelo	TA	Precisión Test	Sens. UNK
A	$1e-3$	0,630	0,056
B	$1e-4$	0,681	0,333
C	$1e-5$	0,684	0,278

Tabla 2: Resultados de tres tasas de aprendizaje distintas

En las matrices de confusión se puede observar el comportamiento en todas las categorías. En la figura 22a se observa cómo el modelo A no aprende nada excepto en las tres primeras categorías, ya que la mayoría de imágenes son clasificadas en la clase NV o en la CCB. En contraste, la del modelo B en la figura 22b (muy similar a la que se obtuvo del modelo C) sí muestra aprendizaje en más clases. No obstante, aunque sea algo mejor este aprendizaje

resulta insuficiente aún, pues la mayoría siguen siendo clasificadas como NV o CCB.

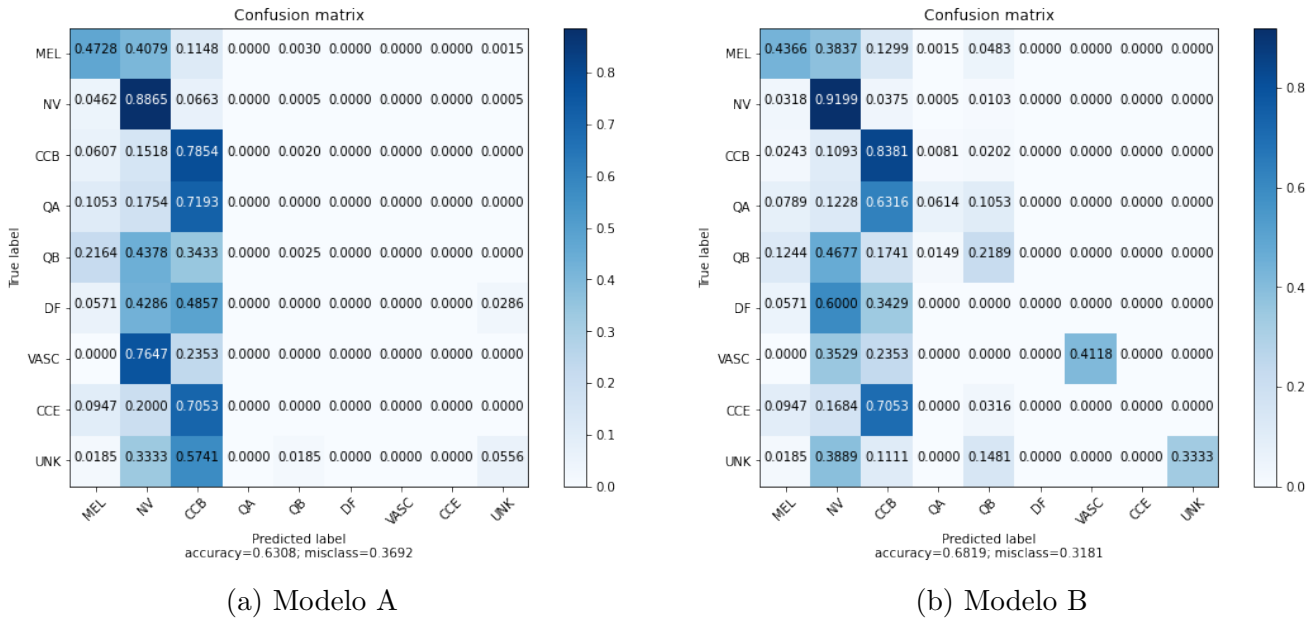


Figura 22: Matrices de confusión de (a) modelo A y (b) modelo B, mostrando la diferencia entre las tasas de aprendizaje de $1e-3$ y $1e-4$ respectivamente

Esto lo podemos corroborar de nuevo con las curvas de entrenamiento, en donde el modelo A (figura 23a) presenta un comportamiento algo inestable, mientras que el modelo B (figura 23b) es más estable pero aún así su curva es bastante horizontal, lo que demuestra que no estaba aprendiendo mucho.

6.2. Aumento de capas entrenables

Después se aumentó el número de capas entrenables de 3 (es decir, solo el clasificador) a 9, incluyendo así las últimas capas de la red convolucional. Esto dio lugar a los modelos D y E (tabla 3). Aquí se observa una notable mejora de la sensibilidad con ambas tasas de aprendizaje respecto a los modelos anteriores, de nuevo con la de $1e-4$ (modelo D) obteniendo un mejor resultado. Aún así, se siguieron manteniendo ambas tasas de aprendizaje y, en vista de los resultados, se adoptó el número de 9 capas entrenables en vez de 3 para las futuras pruebas.

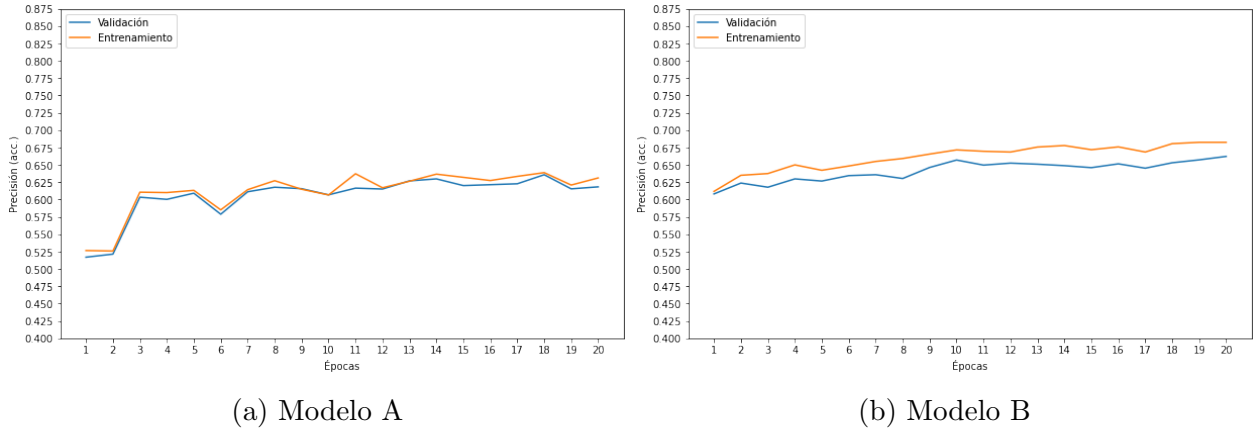


Figura 23: Curvas de entrenamiento de (a) modelo A y (b) modelo B, mostrando la diferencia entre las tasas de aprendizaje de $1e-3$ y $1e-4$ respectivamente

Modelo	Capas descongeladas	TA	Precisión Test	Sens. UNK
B	3	$1e-4$	0,681	0,333
D	9	$1e-4$	0,735	0,556
C	3	$1e-5$	0,684	0,278
E	9	$1e-5$	0,710	0,463

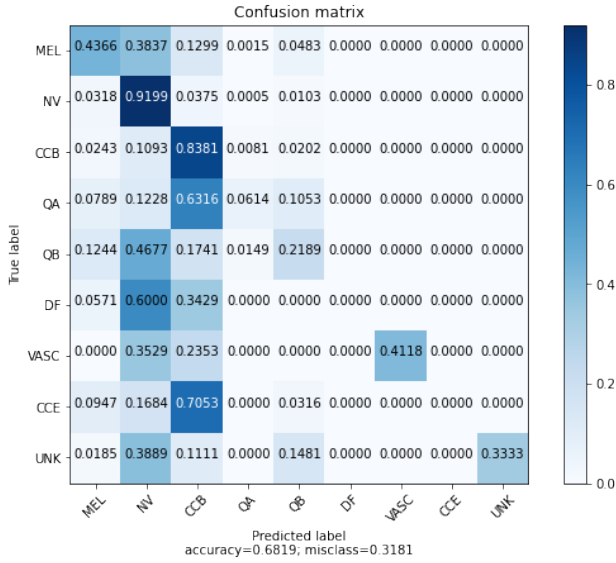
Tabla 3: Comparación de resultados entre modelos con 3 y 9 capas entrenables

A modo general, en las matrices de confusión (figura 24) se puede observar mejoría para aquellas clases que no se estaban clasificando bien, pues aunque muchas siguen siendo clasificadas como NV o CCB, el porcentaje de imágenes que son clasificadas en dichas clases ha disminuido, especialmente en las clases QB y VASC.

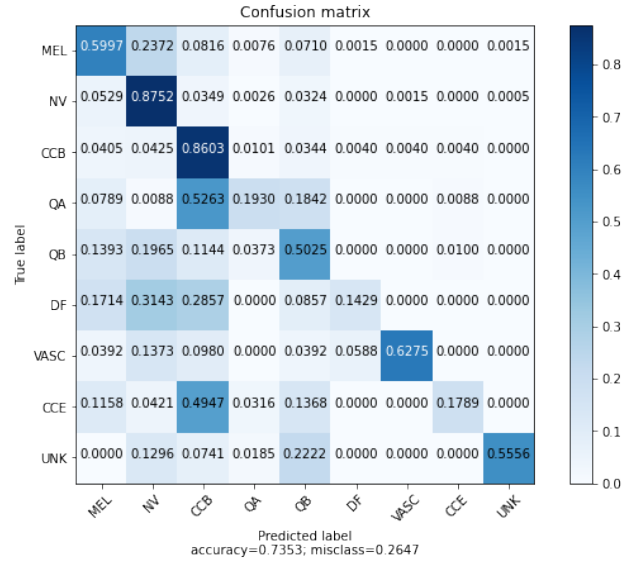
Mirando ahora las curvas de entrenamiento (figura 25), se observa cómo la curva tiene ahora mucha más pendiente que antes, indicando así que está aprendiendo más.

6.3. Introducción de regularización L2

A continuación, se procedió a introducir regularización L2 de 0,01, dando lugar a los modelos F y G (tabla 4). De nuevo, los modelos se entrenaron por 20 épocas. En la tabla 4 se aprecia que esta modificación hizo que la sensibilidad bajase con ambas tasas de aprendizaje.



(a) Modelo B



(b) Modelo D

Figura 24: Matrices de confusión de (a) modelo B y (b) modelo D, mostrando la diferencia tras aumentar las capas entrenables de 3 a 9 respectivamente

No obstante, en las siguientes modificaciones se seguirá probando a incluir regularización para analizar mejor su comportamiento, al igual que con la tasa de aprendizaje.

Modelo	TA	L2	Precisión Test	Sens. UNK
D	1e-4	-	0,735	0,556
F	1e-4	0,01	0,713	0,481
E	1e-5	-	0,710	0,463
G	1e-5	0,01	0,711	0,407

Tabla 4: Comparación de resultados al incluir regularización L2

En cuanto a las matrices de confusión (figura 26) se observa una bajada de la sensibilidad también en la mayoría de categorías, mientras que en las curvas de entrenamiento no se observan apenas diferencias (figura 27).

6.4. Reducción de neuronas

Posteriormente, se redujo el número de neuronas de la primera capa del clasificador de 128 a 64, la mitad, de nuevo entrenando durante 20 épocas. Se hizo para ambas tasas de

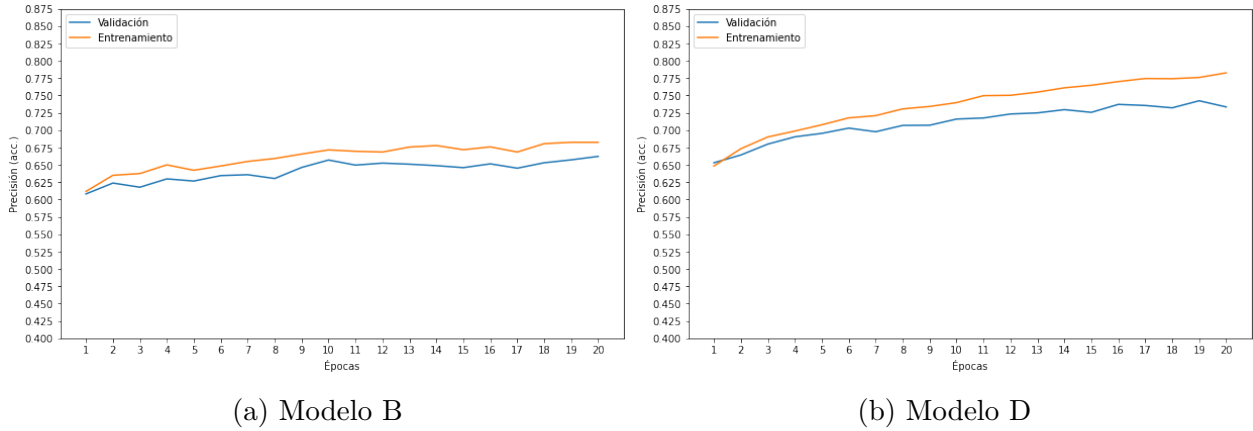


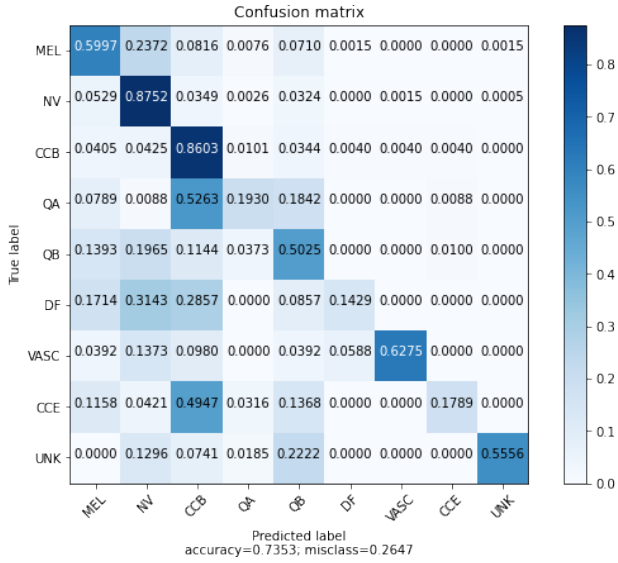
Figura 25: Curvas de entrenamiento de (a) modelo B y (b) modelo D, mostrando la diferencia tras aumentar las capas entrenables de 3 a 9 respectivamente

aprendizaje así como incluyendo regularización y sin incluirla (modelos de la H a la K, tabla 5).

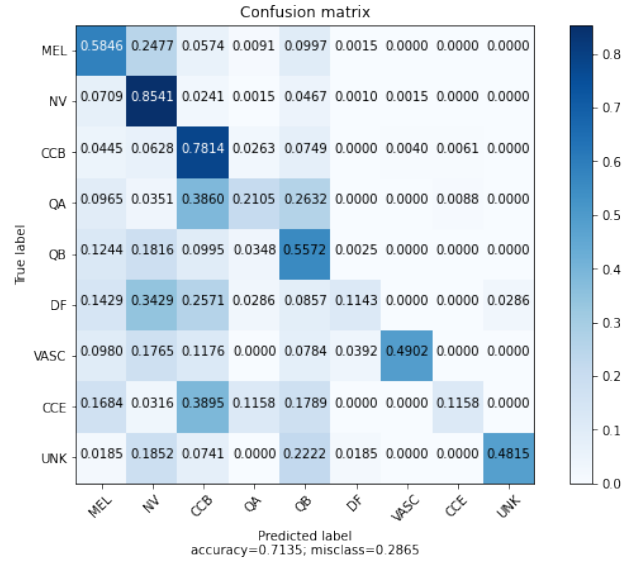
Aquí hubo una notable mejoría en la sensibilidad cuando la tasa de aprendizaje era de $1e-4$, mientras que en el caso de la de $1e-5$ fue todo lo contrario. Por tanto, ya se optó por desechar la tasa de aprendizaje de $1e-5$ para la siguiente y última prueba (añadir otros datasets). Además, se obtuvo el modelo con mejores resultados hasta ese momento (modelo H). Por tanto, esta modificación resultó beneficiosa, por lo que se mantuvo para la siguiente prueba.

Modelo	Neuronas	TA	L2	Precisión Test	Sens. UNK
D	128	$1e-4$	-	0,735	0,556
H	64	$1e-4$	-	0,721	0,630
F	128	$1e-4$	0,01	0,713	0,481
J	64	$1e-4$	0,01	0,715	0,574
E	128	$1e-5$	-	0,710	0,463
I	64	$1e-5$	-	0,697	0,407
G	128	$1e-5$	0,01	0,711	0,407
K	64	$1e-5$	0,01	0,699	0,370

Tabla 5: Resultados de la reducción de neuronas



(a) Modelo D



(b) Modelo F

Figura 26: Matrices de confusión de (a) modelo D y (b) modelo F, mostrando la diferencia tras incluir regularización L2

La precisión general, sin embargo, se vio algo resentida en la mayoría de los modelos, aunque muy poco en comparación con la mejora en la sensibilidad de UNK. Además de en la tabla 5, esto lo podemos observar por ejemplo con las matrices de confusión de los modelos D y H (figura 28).

6.5. Incorporación de datasets

En primer lugar, se incorporaron conjuntamente los datasets *L2* y *7 point* (ver sección 4 referente a Datasets), para después incorporar el SD y finalmente juntarlos todos (modelos de la L a la Q, tabla 6). Dado que esta era ya la última modificación a nuestro modelo, de la cual saldría el definitivo, se entrenaron durante 50 épocas. Los modelos se evaluaron cada 10 épocas, seleccionando la mejor precisión para cada modelo de entre estas evaluaciones. Se probaron tanto con regularización como sin ella.

Como se observa en la tabla 6, la incorporación de datos mejora la sensibilidad respecto a la conseguida anteriormente (modelo H) en dos casos: sin regularización y usando todos los datasets a la vez (modelo N), y con regularización pero sumando solo el dataset SD (modelo

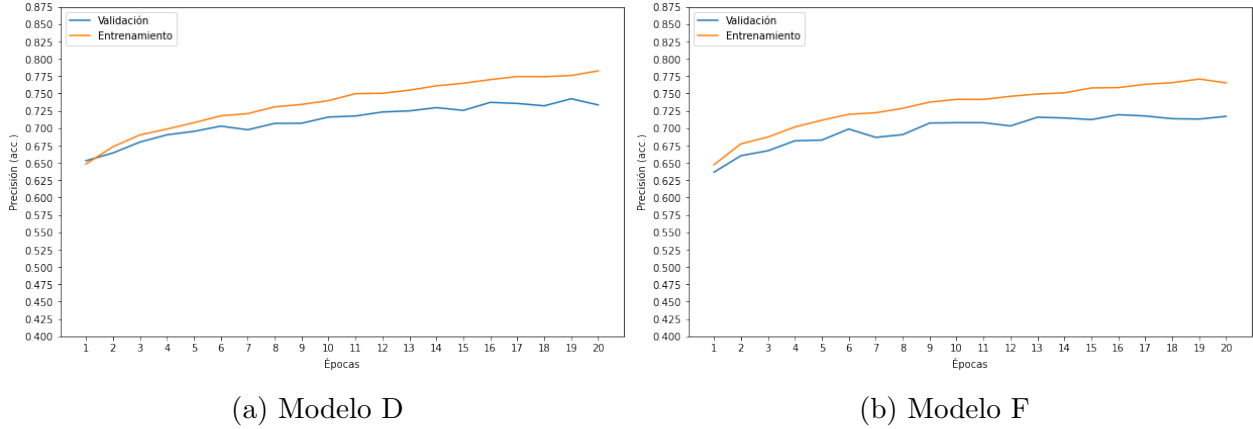


Figura 27: Curvas de entrenamiento de (a) modelo D y (b) modelo F, mostrando la diferencia tras incluir regularización L2

P), siendo por tanto los mejores resultados logrados en el trabajo. Tanto la precisión de test como la sensibilidad de UNK son prácticamente iguales en ambos, por lo que para desempatar se usó el PPV de UNK, el cual es mayor en el modelo N.

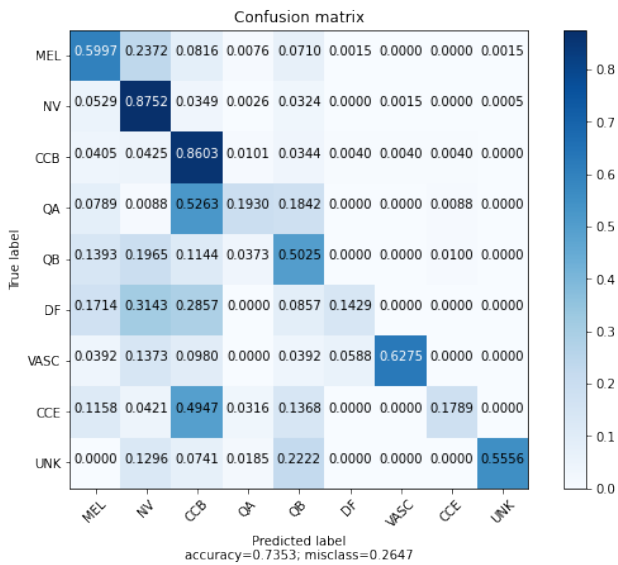
Modelo	L2	Datasets	Precisión Test	Sens. UNK	PPV UNK
H	-	ISIC+UNK	0,735	0,648	0,972
L	-	ISIC+UNK+L2+7P	0,745	0,556	0,968
M	-	ISIC+UNK+SD	0,726	0,648	0,854
N	-	ISIC+UNK+L2+7P+SD	0,737	0,667	0,947
J	0,01	ISIC+UNK	0,734	0,611	0,943
O	0,01	ISIC+UNK+L2+7P	0,737	0,630	0,944
P	0,01	ISIC+UNK+SD	0,730	0,667	0,837
Q	0,01	ISIC+UNK+L2+7P+SD	0,741	0,574	0,838

Tabla 6: Resultados de la incorporación de otros datasets

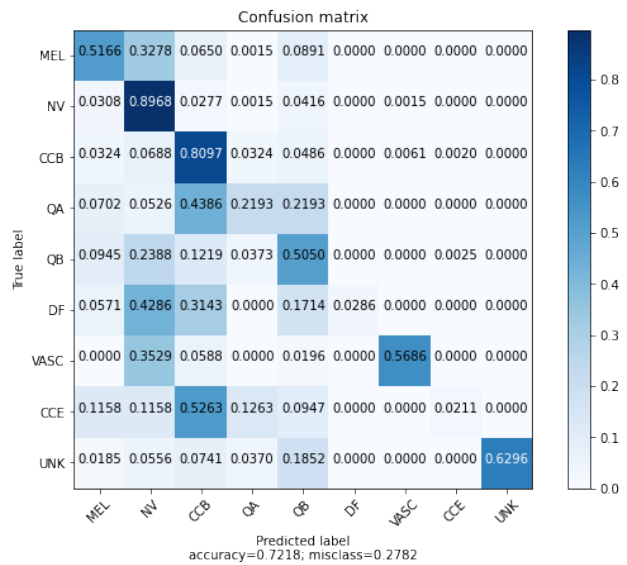
Finalmente, se pueden observar las características completas del modelo final (modelo N) en la tabla 7, así como su matriz de confusión en la figura 29.

Modelo	TA	Capas entrenables	Nº Neuronas	Datasets	Precisión test	Sens. UNK
N	1e-4	9	64	ISIC+UNK+L2+7P+SD	0,737	0,667

Tabla 7: Características del modelo final (modelo N)



(a) Modelo D



(b) Modelo H

Figura 28: Matrices de confusión de (a) modelo D y (b) modelo H, mostrando la diferencia tras reducir el número de neuronas de la primera capa del clasificador de 128 a 64

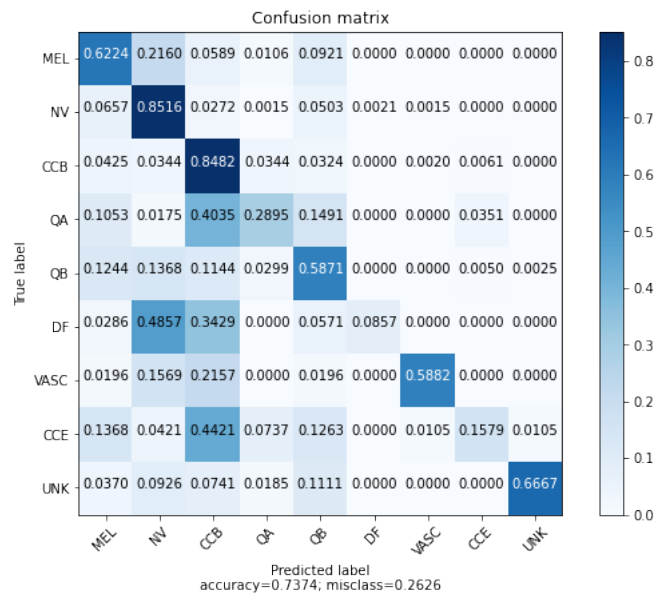


Figura 29: Matriz de confusión del modelo final (modelo N)

7. Discusión

Finalmente el mejor resultado se obtuvo sin regularización L2, una tasa de aprendizaje de $1e-4$, aumentando las capas entrenables, reduciendo el número de neuronas del clasificador e incorporando todos los datasets. Este resultado es de 0,667 de sensibilidad para la clase UNK, por debajo de las mejores clases (NV y CCB con alrededor de un 0,8 de sensibilidad) y por encima de otras como DF, VASC o MEL. En cualquier caso, está por encima de la sensibilidad media entre todas las clases, 0'522. Sin embargo, no es el que mejor precisión presenta para el conjunto de test, pero la desventaja con el que tiene mejor precisión (modelo M) es muy pequeña comparada con la ventaja en sensibilidad de UNK (la diferencia es de 0,055 en la precisión de test y de 0,111 en la sensibilidad UNK)

La precisión general del modelo, 0'737, por sí sola se podría decir que es buena, pues acierta casi 3 de cada 4 imágenes, aunque es obviamente mejorable. Además, claro está que mirando la matriz de confusión hay algunas clases con muy buen rendimiento y otras con un rendimiento muy pobre.

Esta diferencia de resultados entre clases se debe a una de las principales limitaciones de este trabajo: que los datos estaban muy desbalanceados. Mientras en la clase NV hemos contado con casi 14.000 imágenes, otras como la de CCE, DF o QA se han entrenado con imágenes del orden de unos pocos cientos, de forma que esto se ha reflejado claramente en los resultados.

También, el resultado de la clase UNK se debería tomar con prudencia. La mayoría de imágenes de este trabajo pertenecían a un mismo dataset, el ISIC, mientras que de las imágenes de la clase UNK no había ninguna que perteneciera a este. Aunque el ISIC es un dataset relativamente grande, y por tanto las imágenes presentan bastante variabilidad en el tamaño, forma, resolución, color y en general cualquier característica, el hecho de que las imágenes UNK tuvieran otras características distintas puede haber facilitado el que se reconozcan como no pertenecientes a ninguna de las otras clases, pues todas las demás pertenecían a ISIC

en el conjunto de test. En este sentido, si en cualquiera de las matrices de confusión de este documento comparamos la categoría UNK con otras que tenían una cantidad de imágenes similar (como la QA, DF o CCE cuando no se ha usado el dataset SD) estas presentan una precisión mucho menor. Esta diferencia en principio tiene sentido, pues en general las 8 clases principales del problema son patologías con un gran parecido entre sí mientras que las de la clase UNK son más distintas, pero quizás es demasiado amplia como para sustentarse solo en eso.

Otra limitación ha sido la potencia de cálculo. A falta de una computadora con GPU para entrenar los modelos, se ha hecho uso de la herramienta Google Colab, la cual permite usar GPUs en la nube. No obstante, esta herramienta tiene unos límites diarios de tiempo de cálculo así como también presenta otros obstáculos que hacen un poco más engorroso el proceso, como por ejemplo tener que subir todos los datos de entrenamiento a la GPU en cada sesión.

8. Conclusión

En este trabajo se ha desarrollado un sistema de diagnóstico automático de lesiones cutáneas a partir de imágenes, prestando especial atención a la identificación de aquellas patologías distintas de las ocho principales. Esto se ha hecho introduciendo pequeñas modificaciones a una red convolucional previamente entrenada para escoger el sistema más preciso. Estas modificaciones finalmente han sido el uso de una tasa de aprendizaje de $1e-4$, el aumento de capas entrenables, la reducción de neuronas del clasificador, y la inclusión de otros datasets para el entrenamiento.

Teniendo en cuenta la precisión general del modelo, y en especial en algunas de las categorías, queda patente que un sistema de estas características es algo factible y que por tanto es conveniente seguir investigando en esta línea. En cuanto a la identificación de la clase UNK, se ha conseguido que el modelo identifique a gran parte de las imágenes de dicha clase, aunque con las limitaciones explicadas en la discusión.

Como trabajo futuro, convendría probar otras redes convolucionales aparte de la usada aquí, como podrían ser otras de la familia EfficientNet, así como de la VGG o la ResNet, por nombrar algunas. Además de esto, una de las claves para aumentar la precisión es aumentar la cantidad de datos disponibles para el entrenamiento, procurando especialmente que la cantidad de imágenes por cada clase sea más igualitaria que la usada en este trabajo. Esto último sería especialmente importante para la clase UNK, pues por su propia definición engloba a numerosas patologías distintas, por lo que serían necesarias muchas más imágenes de las que se han usado para al menos poder cubrir las más frecuentes.

A nivel de potencia computacional, también es recomendable de cara al futuro aprovechar al máximo todos los avances en este campo, pues actualmente el entrenamiento de las redes convolucionales requiere bastante tiempo de cálculo, lo que sin duda es una limitación a la hora de desarrollar los modelos y a la cantidad de modificaciones que da tiempo a probar en cada uno. En concreto, se podrían usar TPUs en vez de GPUs dado que presentan una

potencia bastante mayor. Estas empiezan ya a estar disponibles, por ejemplo, en la misma herramienta que se ha usado para este trabajo (Google Colab) y también TensorFlow y Keras dan ya soporte para ser usados en estos procesadores.

En un escenario más lejano, cuando estos puntos estén ya superados y se haya podido conseguir un modelo con unos resultados plenamente satisfactorios, sería necesario comparar la precisión de este modelo con la de médicos y dermatólogos para averiguar en qué grado su implementación sería beneficiosa (por ejemplo, como herramienta consultiva para dermatólogos, o no tanto como consultiva pero sí útil para ordenar la prioridad de los pacientes, o como no útil para dermatólogos pero sí para médicos de atención primaria, etc.) y, por último, comprobar cómo funcionaría en un entorno clínico real antes de su materialización.

A. Especificaciones técnicas de Keras

En este apéndice se explica la implementación técnica del trabajo comentando las clases usadas, las cuales se pueden observar en el diagrama de la figura 30.

En primer lugar, las imágenes están todas en un mismo directorio, tanto las de entrenamiento como las de test. Por otro lado, las etiquetas se recogen en dos ficheros *.csv* en formato *hot-encoded* (tabla 8), uno para las imágenes de entrenamiento y otro para las de test. Para la entrada de datos se usa un objeto *ImageDataGenerator*, al cual se le pasan las modificaciones que queremos aplicar a las imágenes para hacer aumento de datos (rotación, desplazamiento, etc), así como la función de preprocesado propia de cada red para que la aplique. Se usan dos objetos de esta clase, uno para el conjunto de entrenamiento y otro para el conjunto de test, de forma que al del conjunto de test no le hacemos aumento de datos ya que siempre tiene que ser igual. Esta clase consta del método *flow_from_dataframe()*, el cual a partir del *dataframe* (en el que previamente se ha escrito el *.csv*) selecciona las imágenes de la carpeta y devuelve parejas de imágenes con su correspondiente etiqueta. Esto lo hace en pequeños grupos de imágenes, lo que es especialmente conveniente para conjuntos de datos muy grandes que no entren en la memoria, como en este caso. Por último, gracias al atributo *validation_split* el mismo objeto se encarga de reservar una parte del conjunto de entrenamiento como conjunto de validación.

nombreArchivo	MEL	NV	CCB	QA	QB	DF	VASC	CCE	UNK
mel.jpg	1	0	0	0	0	0	0	0	0

Tabla 8: Muestra de la etiqueta de una imagen tipo melanoma en formato *hot encoded*

Después, se crea el modelo a entrenar (clase *Model*). Este está compuesto por capas (clase *Layer*), de las que además hay subclases representando las distintas capas que componen una CNN, explicadas en la sección 3.2. Las capas convolucionales y densas (clases *Conv2D* y *Dense*) constan de una función de activación y también se les puede suministrar un regularizador (clase *Regularizer*), que en este trabajo ha sido L2. El modelo usa objetos de la clase

Adam y *CategoricalCrossentropy* para calcular el gradiente y el error, así como también usa los objetos *ImageDataGenerator* para obtener los datos¹.

El método *fit()* del modelo se encarga de entrenar al modelo por épocas (se cuenta una época cada vez que en el entrenamiento se recorren todos los datos de entrenamiento). Para determinar la precisión y el error al final de cada época, se ha diseñado un objeto *TrainingHistoryCallback* (subclase de la clase nativa *Callback*), el cual al final de cada época guarda el error y la precisión del modelo para que finalmente se pueda obtener la curva de entrenamiento.

Por último, mediante el método *predict()* de la clase *Model* se obtiene la predicción para los datos de test, que se comparan con las etiquetas reales para extraer la pérdida, la precisión y la matriz de confusión del modelo.

B. Tabla completa de resultados

En la tabla 9 se muestran todos los resultados obtenidos.

¹Realmente el objeto *ImageDataGenerator*, por medio de la función *flow_from_dataframe()*, devuelve un objeto *Iterator* de *Python*, que es el que en realidad se le suministra al modelo, pero esto se ha omitido del texto y del diagrama para dar una idea básica de las clases de Keras usadas y simplificar la explicación al lector.

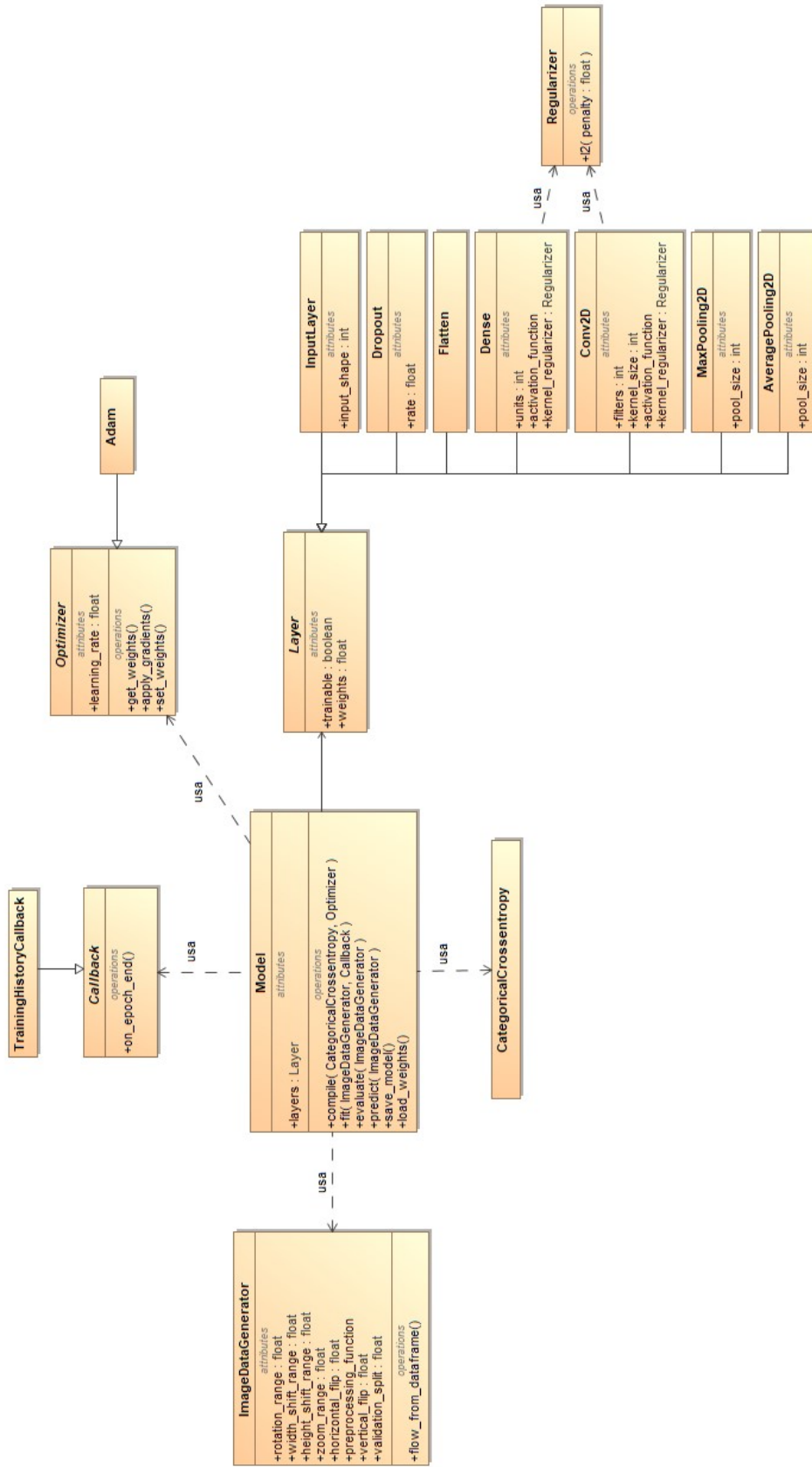


Figura 30: Diagrama UML de las clases usadas de Keras

Modelo	TA	Capas		Neuronas		Error Prec. Test Test	Sens. Esp.		PPV NPV		Sens. Esp.		PPV NPV		
		entrenables	L2	penúlt.	capa		Datasets	Media	Media	Media	Media	Media	Media	Media	Media
A	1e-3	3	-	128	ISIC, UNK	1,115	0,630	0,245	0,913	0,477	0,921	0,056	0,999	0,5	0,979
B	1e-4	3	-	128	ISIC, UNK	0,925	0,681	0,358	0,927	0,687	0,939	0,333	1	1	0,986
C	1e-5	3	-	128	ISIC, UNK	0,879	0,684	0,382	0,932	0,671	0,936	0,278	1	1	0,985
D	1e-4	9	-	128	ISIC, UNK	0,779	0,735	0,504	0,948	0,666	0,949	0,556	0,999	0,938	0,992
E	1e-5	9	-	128	ISIC, UNK	0,827	0,710	0,437	0,938	0,599	0,944	0,463	0,999	0,926	0,989
F	1e-4	9	0,01	128	ISIC, UNK	1,276	0,713	0,466	0,942	0,642	0,943	0,481	1	0,963	0,99
G	1e-5	9	0,01	128	ISIC, UNK	1,861	0,711	0,441	0,939	0,643	0,943	0,407	1	0,957	0,988
H	1e-4	9	-	64	ISIC, UNK	0,781	0,721	0,466	0,941	0,698	0,947	0,630	1	1	0,993
I	1e-5	9	-	64	ISIC, UNK	0,865	0,697	0,415	0,936	0,578	0,940	0,407	0,999	0,846	0,988
J	1e-4	9	0,01	64	ISIC, UNK	1,126	0,715	0,478	0,944	0,733	0,944	0,574	1	0,969	0,992
K	1e-5	9	0,01	64	ISIC, UNK	1,495	0,699	0,400	0,937	0,585	0,940	0,370	1	1	0,987
L	1e-4	9	-	64	ISIC, UNK, L2, 7P	0,779	0,745	0,521	0,949	0,711	0,952	0,556	1	0,968	0,992
M	1e-4	9	-	64	ISIC, UNK, SD	0,777	0,726	0,475	0,945	0,635	0,947	0,648	0,998	0,854	0,993
N	1e-4	9	-	64	ISIC, UNK, L2, 7P, SD	0,766	0,737	0,522	0,950	0,659	0,949	0,667	0,999	0,947	0,994
O	1e-4	9	0,01	64	ISIC, UNK, L2, 7P	1,121	0,737	0,525	0,949	0,692	0,949	0,630	0,999	0,944	0,993
P	1e-4	9	0,01	64	ISIC, UNK, SD	1,096	0,730	0,528	0,949	0,704	0,947	0,667	0,997	0,837	0,994
Q	1e-4	9	0,01	64	ISIC, UNK, L2, 7P, SD	1,094	0,741	0,510	0,947	0,662	0,949	0,574	0,998	0,838	0,992

Tabla 9: Tabla completa de resultados

- [10] American Cancer Society. *Survival Rates for Melanoma Skin Cancer*. URL: <https://www.cancer.org/cancer/melanoma-skin-cancer/detection-diagnosis-staging/survival-rates-for-melanoma-skin-cancer-by-stage.html>. (a 20/05/2020).
- [11] A. Rodriguez-Ruiz y col. “Stand-Alone Artificial Intelligence for Breast Cancer Detection in Mammography: Comparison With 101 Radiologists”. En: *JNCI: Journal of the National Cancer Institute* 111.9 (2019), págs. 916-922. DOI: <https://doi.org/10.1093/jnci/djy222>.
- [12] H. A. Haenssle y col. “Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists”. En: *Annals of Oncology* 29 (2018), págs. 1836-1842. DOI: <https://doi.org/10.1093/annonc/mdy166>.
- [13] S. M. McKinney y col. “International evaluation of an AI system for breast cancer screening”. En: *Nature* 577 (2020), págs. 89-94. DOI: <https://doi.org/10.1038/s41586-019-1799-6>.
- [14] A. Esteva, B. Kuprel, R. Novoa y col. “Dermatologist-level classification of skin cancer with deep neural networks”. En: *Nature* 542 (2017), 115-118. DOI: <https://doi.org/10.1038/nature21056>.
- [15] Y. Fujisawa y col. “Deep-learning-based, computer-aided classifier developed with a small dataset of clinical images surpasses board-certified dermatologists in skin tumour diagnosis”. En: *British Journal of Dermatology* 180.2 (2019), págs. 373-381. DOI: <https://doi.org/10.1111/bjd.16924>.
- [16] A. Hekler y col. “Superior skin cancer classification by the combination of human and artificial intelligence”. En: *European Journal of Cancer* 120 (2019), págs. 114-121. DOI: <https://doi.org/10.1016/j.ejca.2019.07.019>.
- [17] Y. Liu, A. Jain, C. Eng y col. “A deep learning system for differential diagnosis of skin diseases”. En: *Nature Medicine* (2020). DOI: <https://doi.org/10.1038/s41591-020-0842-3>.

- [18] A. Lallas. “Artificial intelligence and melanoma diagnosis: ignoring human nature may lead to false predictions”. En: *Dermatology practical conceptual* 8 (2018), págs. 249-251. DOI: <https://doi.org/10.5826/dpc.0804a0>.
- [19] A. Esteva y E. Topol. “Can skin cancer diagnosis be transformed by AI?” En: *The Lancet* 394.10211 (2019), pág. 1795. DOI: [https://doi.org/10.1016/S0140-6736\(19\)32726-6](https://doi.org/10.1016/S0140-6736(19)32726-6).
- [20] X. Du-Harpur y col. “What is AI? Applications of artificial intelligence to dermatology”. En: *British Journal of Dermatology* (2020). DOI: <https://doi.org/10.1111/bjd.18880>.
- [21] *ISIC Archive*. URL: <https://www.isic-archive.com>. (a 11/06/2020).
- [22] *ISIC Challenge 2019*. URL: <https://challenge2019.isic-archive.com>. (a 11/06/2020).
- [23] *7 point dataset*. URL: <http://derm.cs.sfu.ca/Welcome.html>. (a 11/06/2020).
- [24] J. Kawahara y col. “Seven-point checklist and skin lesion classification using multitask multimodal neural nets”. En: *IEEE Journal of Biomedical and Health Informatics* 23.2 (2019), págs. 538-546. DOI: <https://doi.org/10.1109/JBHI.2018.28243277>.
- [25] *L2 dataset*. URL: <https://www.it.pt/AutomaticPage?id=3459/>. (a 11/06/2020).
- [26] S. M. M. de Faria y col. “Light Field Image Dataset of Skin Lesions”. En: *41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2019, págs. 3905-3908. DOI: <https://doi.org/10.1109/embc.2019.8856578>.
- [27] *SD dataset*. URL: <http://xiaopingwu.cn/assets/projects/sd-198/>. (a 11/06/2020).
- [28] S. Xiaoxiao y col. “A Benchmark for Automatic Visual Classification of Clinical Skin Disease Images”. En: *Computer Vision - ECCV 2016*. Springer International Publishing, 2016, págs. 206-222. URL: <https://xiaoxiaosun.com/docs/2016-eccv-sd198.pdf>.
- [29] F. Chollet. *Deep Learning with Python*. Manning Publications, 2018.

- [30] Google AI Blog. *EfficientNet: Improving Accuracy and Efficiency through AutoML and Model Scaling*. URL: <https://ai.googleblog.com/2019/05/efficientnet-improving-accuracy-and.html>. (a 08/06/2020).

Fuentes de las figuras

- [1] WHO Global Cancer Observatory. *Cancer today*. URL: <https://gco.iarc.fr/today/home>. (a 20/05/2020).
- [2] AECC. *Informe sobre incidencia del cáncer en España*. URL: <https://app.powerbi.com/view?r=eyJrIjoibDVmNjd1OTQtOTE5Zi00YmJmLWJhMjgtNTgxMGJkM2FjYTE3IiwidCI6ImJjYTNjYTJlLTlyNGMtNDNhYS05MTgxLWY2N2YxYzI3OTAyOSIsImMiOjhh9>. (a 13/04/2020).
- [3] J. B. Ahire. *The Artificial Neural Networks handbook: Part 1*. URL: <https://medium.com/coinmonks/the-artificial-neural-networks-handbook-part-1-f9ceb0e376b4>. (a 08/06/2020).
- [4] T. Babs. *The Mathematics of Neural Networks*. URL: <https://medium.com/coinmonks/the-mathematics-of-neural-network-60a112dd3e05>. (a 08/06/2020).
- [5] S. S. *Activation Functions in Neural Networks*. URL: <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>. (a 04/06/2020).
- [6] S. Saha. *A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way*. URL: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>. (a 08/06/2020).
- [7] *3 filters in a Convolutional Neural Network*. URL: https://commons.wikimedia.org/wiki/File:3_filters_in_a_Convolutional_Neural_Network.gif. (a 08/06/2020).
- [8] F. Chollet. *Deep Learning with Python*. Manning Publications, 2018.
- [9] S. Pokhrel. *Beginners Guide to Convolutional Neural Networks*. URL: <https://towardsdatascience.com/beginners-guide-to-understanding-convolutional-neural-networks-ae9ed58bb17d>. (a 08/06/2020).
- [10] *Dive into Deep Learning: 6.3. Padding and Stride*. URL: https://d2l.ai/chapter_convolutional-neural-networks/padding-and-strides.html. (a 08/06/2020).
- [11] A. Dertat. *Applied Deep Learning - Part 4: Convolutional Neural Networks*. URL: <https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2>. (a 08/06/2020).

- [12] Clínica Mayo. *Melanoma*. URL: <https://www.mayoclinic.org/diseases-conditions/melanoma/symptoms-causes/syc-20374884>. (a 20/05/2020).
- [13] Google AI Blog. *EfficientNet: Improving Accuracy and Efficiency through AutoML and Model Scaling*. URL: <https://ai.googleblog.com/2019/05/efficientnet-improving-accuracy-and.html>. (a 08/06/2020).
- [14] *Convolutional neural networks for visual recognition*. URL: <https://cs231n.github.io/neural-networks-3/>. (a 10/06/2020).
- [15] N. Srivastava y col. “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”. En: *Journal of Machine Learning Research* 15 (2014). URL: <http://jmlr.org/papers/volume15/srivastava14a.old/srivastava14a.pdf>.
- [16] M. Elgendy. *Deep Learning for Vision Systems*. Manning Publications, 2020.

Fuentes del dataset *Unknown*

- [1] *Dermoscopyedia*. URL: <https://dermoscopyedia.org>.
- [2] *DermNet NZ*. URL: <https://dermoscopyedia.org>.
- [3] *Dermatoweb*. URL: <http://www.dermatoweb.net>.
- [4] *Entodermoscopy*. URL: <https://www.entodermoscopy.net>.
- [5] *Dermoscopy Atlas*. URL: <http://www.dermoscopyatlas.com>.
- [6] B. M. Piraccini, F. Bruni y M. Starace. “Dermoscopy of non-skin cancer nail disorders”. En: *Dermatologic Therapy* 25.6 (2012), págs. 594-602. DOI: <https://doi.org/10.1111/j.1529-8019.2012.01521.x>.
- [7] A. Lallas y col. “Accuracy of dermoscopic criteria for the diagnosis of psoriasis, dermatitis, lichen planus and pityriasis rosea”. En: *British Journal of Dermatology* 166.6 (2012), págs. 1198-1205. DOI: <https://doi.org/10.1111/j.1365-2133.2012.10868.x>.
- [8] F. Vázquez-López y col. “A dermoscopy subpattern of plaque-type psoriasis: red globular rings”. En: *Archives of Dermatology* 143.12 (2007), págs. 1612-1612. DOI: <https://doi.org/10.1001/archderm.143.12.1612>.
- [9] F. Vázquez-López, J. Kreusch y A. Marghoob. “Dermoscopic semiology: further insights into vascular features by screening a large spectrum of nontumoral skin lesions”. En: *British Journal of Dermatology* 150.2 (2004), págs. 226-231. DOI: <https://doi.org/10.1111/j.1365-2133.2004.05753.x>.
- [10] R. M. Bakos, A. Cartell y L. Bakos. “Dermatoscopy of early-onset necrobiosis lipoidica”. En: *Journal of the American Academy of Dermatology* 66 (2012). DOI: <https://doi.org/10.1016/j.jaad.2011.01.02>.
- [11] R. Segal y col. “Report: Dermoscopy as a diagnostic tool in demodicidosis”. En: *International Journal of Dermatology* 49.9 (2010), págs. 1018-1023. DOI: <https://doi.org/10.1111/j.1365-4632.2010.04495.x>.

- [12] G. Micali y col. “Dermoscopy: Alternative uses in daily clinical practice”. En: *Journal of the American Academy of Dermatology* 64.6 (2011). DOI: <https://doi.org/10.1016/j.jaad.2010.03.010>.
- [13] J. Bae y col. “Differential diagnosis of plantar wart from corn, callus and healed wart with the aid of dermoscopy”. En: *British Journal of Dermatology* 160.1 (2009), págs. 220-222. DOI: <https://doi.org/10.1111/j.1365-2133.2008.08937.x>.
- [14] A. Lallas y col. “Dermoscopy of discoid lupus erythematosus”. En: *British Journal of Dermatology* 168.2 (2013), págs. 284-288. DOI: <https://doi.org/10.1111/bjd.12044>.
- [15] M. Song y col. “Structural correlations between dermoscopic and histopathological features of juvenile xanthogranuloma”. En: *Journal of the European Academy of Dermatology and Venereology* 25.3 (2011), págs. 259-263. DOI: <https://doi.org/10.1111/j.1468-3083.2010.03819.x>.
- [16] E. Errichetti y G. Stinco. “Dermoscopy in General Dermatology: A Practical Overview”. En: *Dermatology and Therapy* 6.4 (2016), págs. 471-507. DOI: <https://doi.org/10.1007/s13555-016-0141-6>.
- [17] A. K. Jha, S. Sonthalia y A. Lallas. “Dermoscopy as an evolving tool to assess vitiligo activity”. En: *Journal of the American Academy of Dermatology* 78.5 (2018), págs. 1017-1019. DOI: <https://doi.org/10.1016/j.jaad.2017.12.009>.
- [18] I. Zalaudek, J. Giacomel, H. Cabo y col. “Entodermoscopy: a new tool for diagnosing skin infections and infestation”. En: *Dermatology* 216.1 (2008), págs. 14-23. DOI: <https://doi.org/10.1159/000109353>.
- [19] A. Llambrich y col. “Dermoscopy of cutaneous leishmaniasis”. En: *British Journal of Dermatology* 160.4 (2009), págs. 756-761. DOI: <https://doi.org/10.1111/j.1365-2133.2008.08986.x>.
- [20] A. Lallas y col. “Scabies escaping detection until dermoscopy was applied”. En: *Dermatology Practical and Conceptual* 7.1 (2017), págs. 49-50. DOI: <https://dx.doi.org/10.5826%2Fdpc.0701a09>.

- [21] A. Lallas y col. “Dermoscopy of early stage mycosis fungoides”. En: *Journal of the European Academy of Dermatology and Venereology* 27.5 (2013), págs. 617-621. DOI: <https://doi.org/10.1111/j.1468-3083.2012.04499.x>.
- [22] F. Vázquez-López y col. “Dermoscopy for the screening of common urticaria and urticaria Vasculitis”. En: *Archives of Dermatology* 144.4 (2008), págs. 568-568. DOI: <https://doi.org/10.1001/archderm.144.4.568>.
- [23] A. A. Garrido-Ríos y col. “Dermoscopy of extragenital lichen sclerosus”. En: *Archives of Dermatology* 145.12 (2009), págs. 1468-1468. DOI: <https://doi.org/10.1001/archdermatol.2009.261>.
- [24] C. Bombonato y col. “Orange color: A dermoscopic clue for the diagnosis of granulomatous skin disease”. En: *Journal of the American Academy of Dermatology* 72.1 (2015), págs. 60-63. DOI: <https://doi.org/10.1016/j.jaad.2014.07.059>.
- [25] R. Pellicano y col. “Dermoscopy of necrobiosis lipoidica and granuloma annulare”. En: *Journal of the American Academy of Dermatology* 226.4 (2013), págs. 319-323. DOI: <https://doi.org/10.1159/000350573>.
- [26] S. Idoudi y col. “Dermoscopy of discoid lupus erythematosus”. En: *European Journal of Pediatric Dermatology* 28.3 (2018), págs. 135-138. DOI: <https://doi.org/10.26326/2281-9649.28.3.1866>.



UNIVERSIDAD
DE MÁLAGA

| uma.es

E.T.S de Ingeniería Informática
Bulevar Louis Pasteur, 35
Campus de Teatinos
29071 Málaga

E.T.S. DE INGENIERÍA INFORMÁTICA