



# ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA Y SISTEMAS DE TELECOMUNICACIÓN

## PROYECTO FIN DE GRADO

**TÍTULO:** Aplicación de aprendizaje profundo: clasificación de imágenes médicas basado en servicios en la nube

**AUTOR:** Michael Adrian Hernández Rala

**TITULACIÓN:** Sistemas de Telecomunicación

**TUTOR:** Juana M<sup>a</sup> Gutiérrez Arriola

**DEPARTAMENTO:** Ingeniería Telemática y electrónica

VºBº

**Miembros del Tribunal Calificador:**

**PRESIDENTE:** Rafael Delgado López

**TUTOR:** Juana M<sup>a</sup> Gutiérrez Arriola

**SECRETARIO:** Victor José Osma Ruiz

**Fecha de lectura:**

**Calificación:**

El Secretario,

## Agradecimientos

Llego el día. Hoy escribo este apartado de agradecimientos para finalizar mi trabajo de fin de grado y con ello, poner fin a mi carrera universitaria.

En primer lugar, me gustaría agradecer a Juana, poder haber confiado en mí para la realización de este proyecto. Por guiarme, por apoyarme y por ser capaz de sacar lo mejor de mí.

En segundo lugar, a la Universidad Politécnica de Madrid por acogerme dentro de sus aulas y hacerme sentir como en casa.

También, a mis amigos, a mi familia, a mi madre Daisy y mi padre Adrián por estar siempre apoyándome a lo largo de toda mi vida y animándome a seguir adelante.

Ha sido un largo camino de aprendizaje, no solo a nivel académico y profesional, sino también a nivel personal. Escribir este trabajo ha tenido un gran impacto en mí y por ello me gustaría agradecer a todas aquellas personas que me han ayudado y apoyado durante este largo trayecto.

## Contenido

Agradecimientos.....	3
Índice de figuras .....	7
Índice de tablas .....	9
Lista de acrónimos .....	11
1. Introducción.....	13
2. Marco tecnológico .....	17
2.1 El melanoma.....	17
2.1.1 Definición de melanoma.....	17
2.1.2 Tipos de melanoma.....	19
2.1.3 Etapas del melanoma.....	21
2.1.4 Factores de riesgo.....	22
2.2 Computación en la nube .....	22
2.2.1 ¿Qué es la computación en la nube?.....	22
2.2.2 Características de la computación en la nube.....	24
2.2.3 Ventajas y desventajas .....	25
2.2.4 Tipos de nubes.....	26
2.2.5 Servicios en la nube .....	27
2.3 Redes neuronales.....	28
2.3.1 Aprendizaje automático.....	28
2.3.2 Redes neuronales artificiales .....	29
2.3.3 Redes neuronales convolucionales (CNN).....	33
3. Especificaciones y herramientas de diseño.....	37
3.1 Python.....	37
3.2 Google Colaboratory.....	38
3.3 Microsoft Azure .....	39
3.4 ISIC Challenge 2019.....	39
4. Descripción y desarrollo de la solución propuesta .....	41
4.1 Implementación del modelo en Google Colab y Microsoft Azure.....	41
4.1.1 Implementación en Google Colab .....	41
4.1.2 Implementación en Azure.....	45
4.2 Análisis para distintas configuraciones de la red neuronal .....	47
4.3 Nueva funcionalidad a desarrollar .....	48
5. Resultados .....	49
5.1 Resultados de la implementación del modelo en servicios en la nube ...	49
5.2 Resultados de los análisis para distintas configuraciones de la red .....	51
5.3 Resultados de la implementación de la nueva funcionalidad.....	60

6. Presupuesto .....	65
7. Conclusiones y líneas futuro de trabajo .....	67
8. Bibliografía .....	71

## Índice de figuras

Figura 1. Mercado del Cloud Computing. Fuente: [3].....	14
Figura 2. Estructura de la piel. Fuente: [7] .....	17
Figura 3. Imágenes de melanomas de diferentes colores. Fuente: [7].....	18
Figura 4. Melanoma cutáneo. Fuente: Google .....	19
Figura 5. Melanoma nodular. Fuente: Google .....	20
Figura 6. Melanoma lentigo maligno. Fuente: Google .....	20
Figura 7. Melanoma lentiginoso acral. Fuente: Google .....	20
Figura 8. Etapas del melanoma. Fuente: [7].....	21
Figura 9. Funcionamiento de la computación en la nube. Fuente: [16].....	23
Figura 10. Frontend y Backend. Fuente: Google.....	23
Figura 11. Modelo tradicional vs Cloud Computing. Fuente: [16].....	24
Figura 12. SaaS vs PaaS vs IaaS. Fuente: [16] .....	27
Figura 13. Tipos de aprendizaje automático. Fuente: [25] .....	28
Figura 14. Neurona biológica. Fuente: [29] .....	29
Figura 15. Neurona artificial. Fuente: [29] .....	30
Figura 16. Red neuronal completa. Fuente: Google.....	30
Figura 17. Función lineal .....	31
Figura 18. Función umbral .....	32
Figura 19. Función sigmoide .....	32
Figura 20. Función tangente hiperbólica .....	32
Figura 21. Función ReLu .....	33
Figura 22. Estructura básica de un CNN.....	33
Figura 23. Arquitectura AlexNet. Fuente: [32] .....	34
Figura 24. Arquitectura GoogLeNet. Fuente: [32].....	35
Figura 25. Arquitectura VGGNet. Fuente: [32] .....	35
Figura 26. Python .....	37
Figura 27. Google Colaboratory .....	38
Figura 28. Microsoft Azure .....	39
Figura 29. ISIC 2019 .....	39
Figura 30. Instalación de Colaboratory.....	42
Figura 31. Cuaderno en Colaboratory .....	43
Figura 32. Celdas en Colaboratory.....	43
Figura 33. Flujo de ejecución en Colab .....	44
Figura 34. Conexión a GPU .....	44
Figura 35. Comprobación de conexión correcta GPU .....	45
Figura 36. Selección MV .....	46
Figura 37. Configuración MV.....	47
Figura 38. Resultado nuevo data augmentation.....	48

## Índice de tablas

Tabla 1. Creación de área de trabajo .....	45
Tabla 2. Selección MV .....	46
Tabla 3. Configuración MV .....	47
Tabla 4. Tiempos de ejecución Google Colab vs Microsoft Azure .....	49
Tabla 5. Tipo de datos tras detección .....	50
Tabla 6. dropout 0.1 para 1_test_450x650.....	52
Tabla 7. dropout 0.4 para 1_test_450x650.....	53
Tabla 8. dropout 0.1 para 1_CropCenter_450x650 .....	54
Tabla 9. dropout 0.4 para 1_CropCenter_450x650 .....	55
Tabla 10. dropout 0.1 para 1_AugmentedCropCenter_450x650.....	56
Tabla 11. dropout 0.4 para 1_AugmentedCropCenter_450x650.....	57
Tabla 12. dropout 0.1 para 1_AugmentedCropCenterRotatedAll_450x650 ....	58
Tabla 13. dropout 0.4 para 1_AugmentedCropCenterRotatedAll_450x650 ....	59
Tabla 14. Resultados analisis distintas configuraciones .....	59
Tabla 15. dropout 0.1 para 2_test_450x650.....	60
Tabla 16. dropout 0.1 para 2_CropCenter_450x650 .....	61
Tabla 17. dropout 0.1 para 2_AugmentedCropCenter_450x650.....	62
Tabla 18. dropout 0.1 para 2_AugmentedCropCenterRotatedAll_450x650 ....	63
Tabla 19. Comparación de los resultados red neuronal inicial vs red neuronal final.....	64

## Lista de acrónimos

AWS: *Amazon Web Services*

CNN: *Convolutional Neural Network*

GPU: *Graphics Processing Unit*

IA: *Inteligencia Artificial*

IaaS: *Infrastructure as a Service*

ISIC: *International Skin Imaging Collaboration*

IT: *Information Technology*

MIT: *Massachusetts Institute of Technology*

ML: *Machine Learning*

NIST: *National Institute of Standards and Technology*

OMS: *Organización Mundial de la Salud*

PaaS: *Platform as a Service*

SaaS: *Software as a Service*

TIC: *Tecnología de la información y comunicación*

UV: *Ultravioleta*

VN: *Virtual Network*

VPN: *Virtual Private Network*

## 1. Introducción

Como de todos es sabido, la tecnología se encuentra en constante crecimiento y desarrollo. La creación de nuevas herramientas ha permitido la generación a su vez de nuevas tecnologías que nos han permitido mejorar otros aspectos de nuestra vida y la forma en la que nos comunicamos. La computación en la nube, también conocida como *Cloud Computing*, es una de estas herramientas. Gracias a ella, todos los seres humanos podemos tener acceso a Internet de una forma segura y rápida a un golpe de *click* [1].

El término *Cloud Computing* fue acuñado por John McCarthy, pionero de la Inteligencia Artificial. Durante el centenario del MIT, el cual tuvo lugar en 1961, John introdujo el concepto de uso compartido del tiempo (*Time-Sharing*), gracias al cual, múltiples usuarios podían hacer uso de un mismo ordenador de manera simultánea. De esta forma, se conseguía que el usuario pudiese ahorrar en costos y pagar únicamente por el tiempo de uso de esa tecnología [1][2].

Esta idea de poder vender la tecnología como un servicio tuvo mucha aceptación por parte de las empresas en la década de 1960. Tanto es así, que incluso algunas empresas empezaron a ofrecer diferentes recursos compartidos a modo de oficina de servicios. Sin embargo, a mediados de los años 70, dado que el hardware, software y las diferentes tecnologías de comunicación no estaban todavía muy desarrolladas, esta tecnología empezó a perder presencia e importancia.

En 1996, Douglas Parkhill, tecnólogo canadiense y ex ministro de investigación, en su libro *El desafío de la utilidad de la computadora* trató diferentes aspectos como la elasticidad de los recursos en las máquinas virtuales que posteriormente serían relevantes en el devenir de esta tecnología.

Para la década de 1990, solamente se podían acceder a las VPN (*Virtual Private Network*) a través de diferentes empresas de telecomunicaciones. Estas redes presentaban prestaciones similares, pero a un coste mucho más reducido. Hoy en día, la computación en la nube vas más allá, permitiendo cubrir tanto servidores como la infraestructura de la red.

*Salesforce* fue una de las empresas precursoras en el desarrollo y avance de la computación en la nube. Esta empresa permitió a los usuarios acceder a diferentes aplicaciones empresariales a través de una página web. Más tarde, *Amazon* apareció en escena, lanzando en 2002 *AWS*. En 2006, llegó *Google* presentando *Google Docs*. Ese mismo año, *Amazon* presentó *Elastic Computing Cloud*, un servicio web con el que empresas pequeñas podían alquilar equipos donde poder ejecutar sus propios programas informáticos. En 2008 apareció *Eucalyptus*, una infraestructura de código abierto que permitía implementar computación en nube privada.

En 2009, *Microsoft* introdujo *Windows Azure*. Un año más tarde, aumentaron la cantidad de recursos que podíamos encontrar a nivel de cliente, aplicación, plataforma, infraestructura y servidor. Finalmente, *Apple* presentó su sistema de almacenamiento en la nube, *iCloud*, cuyo objetivo fue cambiar la manera de usar el ordenador.

Con el paso del tiempo, el concepto de computación en la nube ha ido evolucionando y ganando mucho más terreno. Ha dejado de ser un tema de debate y se ha convertido en una de las herramientas más importantes para las empresas. Según datos revelados por *MarketsandMarkets*, en 2018 se estimó que el tamaño del mercado del *Cloud Computing* fue de unos 272 millones de dólares a nivel global y se prevé que para el año 2023, esta cantidad ascienda a los 623,3 millones [3].

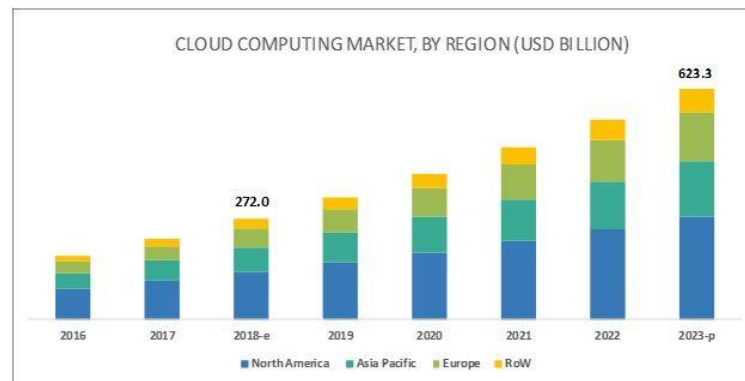


Figura 1. Mercado del Cloud Computing. Fuente: [3]

En la última década, uno de los mejores aliados de la computación en la nube ha sido la Inteligencia Artificial. Ambas tecnologías han formado una combinación perfecta que ha permitido a las empresas propulsar su actividad, hacer un uso más optimizado de sus datos y ofrecer nuevas experiencias enriquecedoras a los clientes [4].

El enorme crecimiento que están experimentando ambos campos traen como resultado la creación de nuevas herramientas y aplicaciones que permiten que áreas como la medicina se vean muy beneficiadas. La unión de la Inteligencia Artificial y el *Cloud Computing* ha permitido el desarrollo de sistemas que facilitan el trabajo de profesionales sanitarios y posibilitan nuevos procedimientos de diagnóstico y tratamiento antes inviables [5][6].

Gracias al estado en el que se encuentra la tecnología y la innumerable cantidad de recursos que tenemos a nuestra disposición, resulta evidente que es nuestro deber como ingenieros hacer uso de todas estas herramientas y conocimientos que vamos adquiriendo con el objetivo de construir un mundo mejor y cada vez más sostenible.

En este contexto, este proyecto tendrá como objetivo desarrollar un sistema clasificatorio de lesiones de piel basado en la nube haciendo uso de los conocimientos y avances tanto en el campo de la computación en la nube como en el de la inteligencia artificial y en concreto, el campo de las redes neuronales convolucionales, que son las que hoy en día se utilizan para el reconocimiento de imágenes.

Este proyecto pretender ser una continuación del proyecto *Optimización de una red convolucional para la detección de melanomas en imágenes dermoscópicas*, desarrollado por el exalumno de la escuela Mario Álvaro Díaz. Por tanto, se hará uso de su red neuronal para la realización de este proyecto.

De esta manera, el presente proyecto estará dividido en las siguientes etapas:

- Implementación de la red neuronal en los siguientes servicios basados en la nube: *Microsoft Azure* y *Google Colab*. Se compararán los resultados obtenidos, así como las prestaciones que nos ofrecen cada uno de ellos. Posteriormente, se elegirá el servicio que mejor nos convenga para realizar las siguientes fases del proyecto.
- Análisis para diferentes configuraciones de la red: Se entrenarán modelos para diferentes configuraciones de los hiperparámetros que rigen su comportamiento y se analizarán los resultados obtenidos en cada uno de ellos y se compararán con los resultados obtenidos para la configuración inicial.
- Adición de nueva funcionalidad: En esta última fase del proyecto se procederá a implementar una nueva funcionalidad que nos permita mejorar el rendimiento de la red y, por tanto, obtener mejores resultados.



## 2. Marco tecnológico

### 2.1 El melanoma

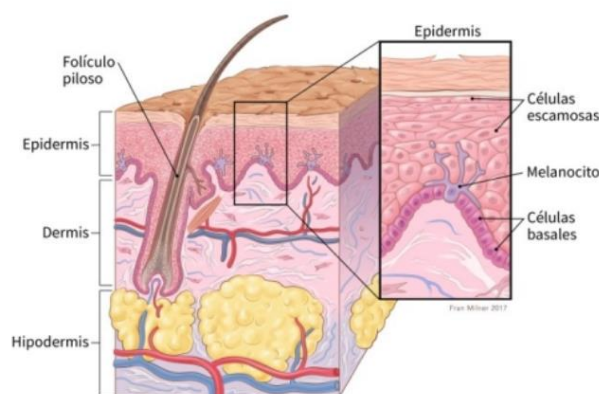
#### 2.1.1 Definición de melanoma

El melanoma se encuentra actualmente entre los tipos de cáncer más graves de piel. Este se produce cuando los llamados melanocitos (células que dan color a nuestra piel y permiten protegerla de los rayos ultravioleta) comienzan a crecer de manera incontrolable hasta el punto de invadir y destruir tejidos cercanos.

Gran parte de los melanomas tienen lugar cuando se expone la piel al sol, ya sea de forma continuada o de forma intermitente. Cuando los daños producidos por este tipo de radiaciones superan la capacidad de nuestro cuerpo tiene para repararlos, se producen mutaciones en estas células que hacen que se multipliquen sin ningún tipo de control [7].

Pese a ser menos común que otro tipo de cánceres de piel ya que solo representa el 1% de los casos de cáncer de piel, el melanoma tiene muchas probabilidades de crecer y propagarse si no se identifica y trata a tiempo. Para el año 2021, la Sociedad Americana contra El Cáncer estima que aparecerán 106.110 nuevos casos de melanoma y morirán 7.180 personas a causa de esta enfermedad [8].

Para tener una mejor comprensión del melanoma hemos de tener en cuenta los diferentes componentes que conforman la piel del ser humano. Como podemos ver en la *Figura 2*, la piel es un órgano formado principalmente por tres capas [9][10]:



*Figura 2. Estructura de la piel. Fuente: [7]*

- **Epidermis:** Se trata de la capa más externa de piel. Su función principal es proteger del exterior a las capas más internas. En esta capa podemos encontrar 5 diferentes subcapas entre las que se encuentran: la capa basal, la capa espinosa, la capa granular, la capa clara y la capa córnea.
- **Dermis:** Se trata de una capa gruesa y elástica que se encuentra en la zona intermedia de la piel. Esta capa contiene a los receptores sensoriales

que permiten al cuerpo reaccionar ante la presión, dolor y temperatura. Dentro de la dermis, distinguimos a su vez dos subcapas: la interior y la superior.

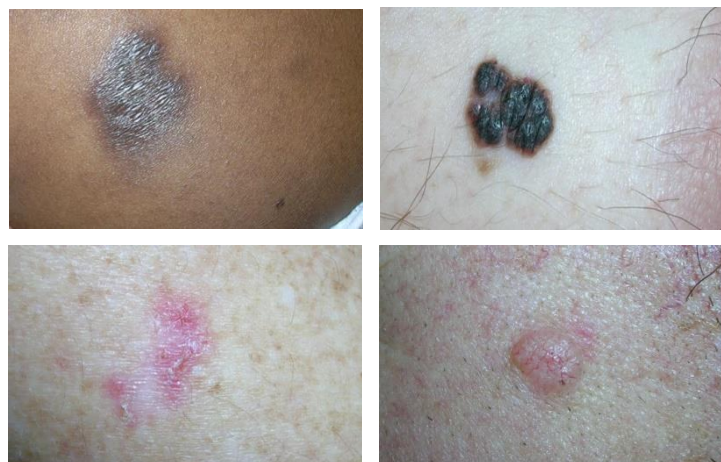
- Hipodermis: Es la parte más interna de la piel. Está formada por células que dotan a la piel de fortaleza, apoyo y permiten aislar al cuerpo y conservar el calor corporal. En esta capa encontramos las células adiposas, fibras colágenas y vasos sanguíneos.

La mayoría de los cánceres de piel comienzan en la epidermis de la piel, la cual está formada por tres tipos principales de células.

- Células escamosas: células situadas en la parte más externa de la epidermis. Estas se desprenden conforme se van generando nuevas células en las capas más internas.
- Melanocitos: células que producen la melanina (pigmento que da color a la piel) y protegen a las capas más internas de la piel contra los efectos de la radiación ultravioleta.
- Células basales: células situadas en la parte más interna de la epidermis.

El cáncer de piel puede desarrollarse en cualquier parte del cuerpo, pero normalmente suelen aparecer en zonas que son más vulnerables a la exposición al sol como el cuello y el rostro, además del pecho y espalda para hombres y piernas para mujeres.

El melanoma, también conocido como melanoma maligno o melanoma cutáneo, como se ha mencionado anteriormente, se origina en los melanocitos, los cuales continúan generando melanina, lo que hace que tumores similares al melanoma suelen ser de color café o negro. Sin embargo, no todos los melanomas producen melanina y pueden tener un color más rosado o incluso blanco [7]. En la *Figura 3* se muestran melanomas que presentan diferentes colores.



*Figura 3. Imágenes de melanomas de diferentes colores. Fuente: [7]*

En general, el cáncer de piel es mucho más común en personas con la piel clara. Esto es debido a que, al tener menos cantidad de melanina, disponen de menos protección frente a las radiaciones ultravioleta. Sin embargo, eso no significa que las personas con una piel más oscura sean inmunes a sufrir este tipo de cáncer,

al contrario, no solo en función del color de piel, si no de la exposición al sol y la de la zona donde se viva, habrá más o menos riesgos de padecer diferentes tipos de melanoma. Según datos de la Sociedad Americana contra el cáncer, el número de personas de raza blanca que padecen este tipo de enfermedad en Estados Unidos es 20 veces mayor que en personas de raza negra. Además, el riesgo de padecer de melanoma a lo largo de su vida es de un 2,6% para los naturales de Europa, norte de África y sudoeste de Asia, un 0,1% para los afroamericanos y un 0,6% para los hispanos [8].

Hasta el momento, solo se ha hablado del cáncer de piel de tipo melanoma que se produce en la piel, pero hay que tener en cuenta:

- Que existen otros tipos de cáncer melanoma que no tienen lugar en la piel, sino que se generan en otras partes del cuerpo como los llamados melanomas mucosos o los melanomas oculares, los cuales como sus propios nombres indican, tienen lugar en las mucosas y en los ojos, respectivamente. Se hará hincapié en este tipo de melanomas en capítulos posteriores.
- Que existen otros tipos de cáncer de piel de tipo no melanoma que tienen lugar en células que no son melanocitos. Estos quedan fuera de la temática del proyecto.

## 2.1.2 Tipos de melanoma

### Melanoma cutáneo

Existen cuatro tipos principales de melanomas cutáneos [11][12]:

- Melanoma de extensión superficial: Representa el 70% de los melanomas registrados. En un primer momento tienen una textura suave pero conforme pase el tiempo se vuelve más rugosa y/o seca. Presenta un crecimiento radial, es decir, transcurrido un período de tiempo de entre 6 a 7 años, crece en profundidad llegando así a la dermis. Es común en personas que se encuentran entre los 30 y 50 años.



Figura 4. Melanoma cutáneo. Fuente: Google

- Melanoma nodular: Representa entre el 15% y el 30% de los casos registrados. Es más agresivo que el melanoma superficial y se desarrolla más rápido. Al igual que los anteriores mencionados, presentan un crecimiento radial. Son duros al tacto y suelen presentar un color azul-negro o negro-rojizo. Suele presentarse en personas que se encuentran entre los 40 y 60 años.



Figura 5. Melanoma nodular. Fuente: Google

- Melanoma lentigo maligno: También conocido como peca melanótica de Hutchison. Representa el 10% de los casos registrados. Suele producirse en partes del cuerpo que más se exponen al sol como pueden ser la cara y los brazos. Es común en mujeres que tienen una piel clara y más de 50 años.

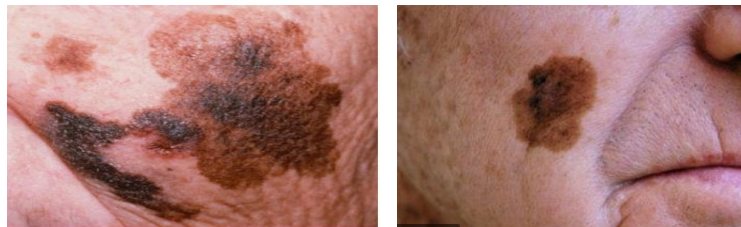


Figura 6. Melanoma lentigo maligno. Fuente: Google

- Melanoma lentiginoso acral: Pese a ser un melanoma poco frecuente (representa solo el 1-2%), es común en personas de piel oscura y menos en personas de piel clara. Suele producirse en la planta de los pies y de las manos. Suelen padecerlo personas mayores de 40 años.

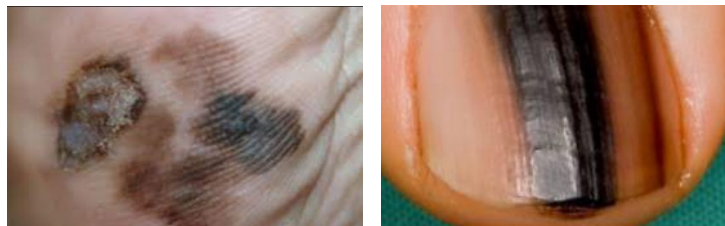


Figura 7. Melanoma lentiginoso acral. Fuente: Google

### Melanoma mucoso

Este tipo de melanoma es muy poco frecuente. De hecho, solo representa el 1% de todos los casos diagnosticados. Como su propio nombre indica, este melanoma tiene lugar en las mucosas, y en especial, aquellas que revisten las cavidades corporales y los órganos huecos. Entre las zonas más comunes, destacamos la cabeza, cuello, recto, vías urinarias y vagina [12].

## Melanoma ocular

Este tipo de cáncer tiende a desarrollarse en la mayoría de los casos en la capa vascular que se encuentra debajo de la esclerótica, la úvea. Sin embargo, también puede desarrollarse en la cavidad que rodea al globo ocular y en los párpados. Al igual que el melanoma mucoso, el melanoma ocular se trata de un tipo de cáncer muy poco común [13].

### 2.1.3 Etapas del melanoma

También son conocidas como los estadios del melanoma. Hacen referencia a que cantidad de cáncer hay en nuestro cuerpo y nos ayuda a determinar la gravedad del cáncer y el mejor tratamiento. Estas se ordenan de 0 al IV. Cuanto menor sea el número de la etapa, menor habrá sido la propagación del melanoma [14].

- Estadio 0: También denominado *melanoma in situ*. En este estadio las células del melanoma únicamente están presentes en la capa más externa de la piel, la epidermis. Es muy poco probable que el melanoma se propague a otras partes del cuerpo.
- Estadio I: En esta etapa el melanoma tiene un grosor menor de 2 milímetros. Es delgado y probablemente se encuentre todavía en la capa más externa de la piel.
- Estadio II: El melanoma presenta un grosor superior al estadio anterior, incluso puede superar los 4 milímetros. Atendiendo al grosor que puede tener el melanoma y a la presencia o no de ulceración (aparición de lesión en la piel) podemos encontrar 3 subgrupos: A, B y C siendo A la más leve y C la más grave.
- Estadio III: En este estadio el melanoma se ha diseminado a los vasos linfáticos y/o ganglios linfáticos cercanos dando lugar a lo que se conoce como metástasis en tránsito, metástasis satélite o tumores microsátelites. Dentro de este estadio distinguimos 4 subgrupos en función del tamaño del melanoma, el número de ganglios linfáticos involucrados, el tipo de lesión que tiene el tumor primario y la presencia o no de ulceración: A, B, C y D.
- Estadio IV: También conocido como melanoma metastásico, se ha expandido a ganglio linfáticos distantes y a otras zonas importantes como los huesos, los músculos, los pulmones, el cerebro o la médula espinal.

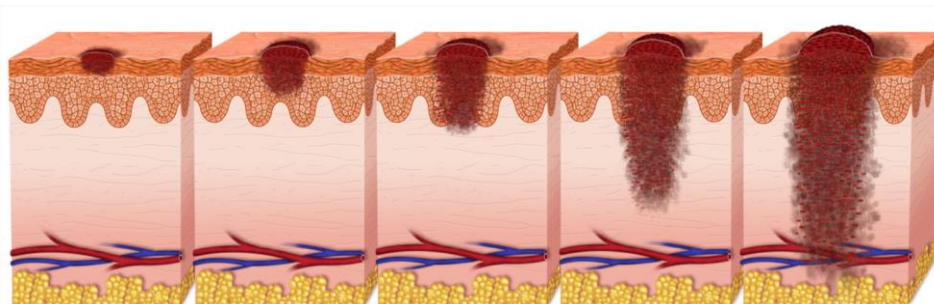


Figura 8. Etapas del melanoma. Fuente: [7]

### 2.1.4 Factores de riesgo

Existen distintos motivos que pueden ocasionar la aparición del melanoma. Entre los más comunes encontramos [15]:

- Piel clara: Tener una piel con pigmentación clara aumenta las posibilidades de padecer melanoma. Esto es debido a que cuanto menos pigmento o melanina se tenga en la piel, menos protección se tiene contra la radiación ultravioleta. Por tanto, el melanoma es más común en personas de pelo rubio, ojos claros y/o con pecas.
- Antecedentes de quemaduras por el sol: El hecho de presentar antecedentes de quemaduras producidas por la exposición al sol durante las primeras etapas de vida aumenta considerablemente el riesgo de padecer cáncer de piel.
- Excesiva exposición a los rayos ultravioleta: Gran parte de los melanomas son debidos a la exposición de la piel al sol.
- Vivir en zonas próximas al ecuador: En zonas cercanas al ecuador los rayos procedentes del sol son más directos y presentan más cantidad de radiación ultravioleta por lo que las personas que viven en este tipo de zonas son más vulnerables a padecer esta enfermedad.
- Vivir en zonas de mayor altitud. Estas zonas están más expuestas a la radiación ultravioleta.
- Tener muchos lunares o lunares poco comunes: Tener una cantidad superior a 50 lunares o tener lunares poco usuales, también denominados nevos displásicos, aumentan considerablemente las probabilidades de padecer el melanoma.
- Antecedentes familiares: Las probabilidades de presentar melanoma son mayores si algún familiar de primer grado, esto es: madre, padre, hermano/a o hijo/a, ha padecido alguna vez esta enfermedad.
- Un sistema inmunitario debilitado: Tener un sistema inmunitario debilitado aumenta las probabilidades de padecer diferentes cánceres de piel, además del melanoma.

## 2.2 Computación en la nube

### 2.2.1 ¿Qué es la computación en la nube?

La computación en la nube, también conocida como *Cloud Computing*, es considerada por algunos como la quinta revolución TIC. Se trata de un modelo de computación a través del cual, los proveedores ponen a disposición del cliente los recursos o servicios a través de Internet. Su principal objetivo es tener toda la información almacenada en la nube de forma que sea accesible desde cualquier parte del mundo, sin necesidad de tener una determinada capacidad de almacenamiento en nuestro ordenador o tener que instalar aplicaciones de forma local.

Una de las principales novedades que introduce este tipo de tecnología es el servicio bajo demanda. Esto es, la capacidad de poder ofrecer al cliente de todos los recursos y servicios requeridos de forma automática, lo que permite abstraer

al cliente sobre cualquier problema relacionado con el tipo de infraestructura requerida para poder ejecutar una determinada aplicación. De esta manera, el cliente, independientemente del lugar en que se encuentre, podrá acceder a todos estos servicios contratados de una forma mucho más dinámica, sin necesidad de realizar ningún desembolso en recursos informáticos determinados [16].



Figura 9. Funcionamiento de la computación en la nube. Fuente: [16]

Dentro del sistema Cloud Computing podemos encontrar dos principales partes que rigen el comportamiento y funcionamiento de esta tecnología:

- *Frontend*: Se trata de la parte visual de una página web. Es la interfaz de usuario. Engloba todas las tecnologías necesarias para el diseño y desarrollo de una página web que corre en la web y que permiten que los usuarios interactúen con esta.
- *Backend*: Se trata de la parte no visual de una página web. Contiene toda la lógica que permite el correcto funcionamiento de una página web. Está formada por los servidores, sistemas y ordenadores que hay en la nube.

Tanto el *Frontend* como el *Backend* se conectan con el servidor encargado de gestionar el funcionamiento y la comunicación entre ambas partes a través de protocolos y *middleware* que permitan establecer una comunicación entre los diferentes ordenadores [17].

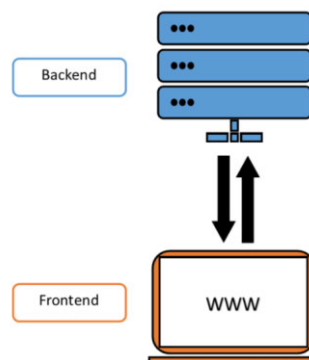


Figura 10. Frontend y Backend. Fuente: Google

Entre los diferentes servicios o recursos que podemos contratar en la nube se encuentran: bases de datos, servidores de correo electrónico, entornos de desarrollo, servidores web o capacidad de almacenamiento. Para poder acceder a ellos, se sigue el modelo “pago por uso” en el que el usuario únicamente deberá pagar por el consumo real que haga de los distintos servicios o recursos que tenga contratados. A diferencia del modelo tradicional en el que las empresas hacían grandes desembolsos tanto hardware como en software necesario para poder ejecutar determinadas aplicaciones, en *Cloud Computing*, todos estos recursos se ponen a disposición del cliente sin necesidad de que este en la obligación de adquirirlos, solo contratándolos como servicio. En la *Figura 11* podemos ver las diferencias existentes entre el modelo tradicional y el modelo de computación en la nube.

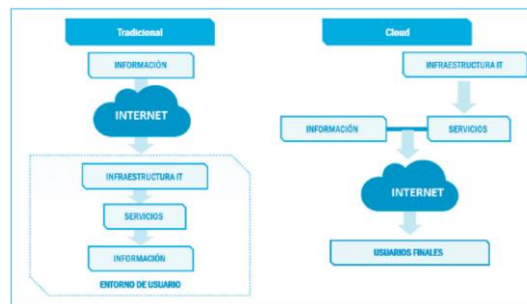


Figura 11. Modelo tradicional vs Cloud Computing. Fuente: [16]

El *Cloud Computing* ha permitido que tanto empresas como autónomos hayan podido aumentar considerablemente su productividad sin necesidad de realizar grandes esfuerzos para ello, dotándoles de agilidad que les ha permitido no solo reducir costes, sino crecer de manera mucho más rápida. Sin embargo, pese a las numerosas ventajas que podemos encontrar en la nube, sigue habiendo un gran desconocimiento sobre esta tecnología. Incluso mucha gente hace un uso diario de distintos servicios almacenados en la nube sin ser realmente conscientes de ello.

### 2.2.2 Características de la computación en la nube

El NIST (*National Institute for Standards and Technology*) destaca los siguientes aspectos como las principales características de la computación en la nube [18][19]:

- Autoservicio bajo demanda: No se requiere de ningún experto IT que proporcione las capacidades de computación requeridas como la velocidad de procesamiento, la capacidad de memoria o la capacidad de almacenamiento, sino que es el propio servicio el que las proporciona de forma automática y a demanda.
- Multiplataforma: Todos los recursos se encuentran disponibles en la red. Por tanto, para hacer uso de ellos, únicamente será necesario tener una conexión a Internet.

- **Multiusuario:** La agrupación de los recursos permiten a los usuarios la posibilidad de compartir los medios y recursos, haciendo un uso más optimizado de ellos.
- **Flexibilidad:** el servicio es capaz de aumentar o disminuir la cantidad de recursos ofrecidos a el cliente en función de las necesidades de este.
- **Pago por uso:** el cliente únicamente paga por los recursos que usa y el precio varía en función del uso que dé al servicio demandado.

### 2.2.3 Ventajas y desventajas

Como se ha mencionado anteriormente, una de las principales novedades que trae consigo la computación en la nube es evitar que sea el propio ordenador el encargado de asumir la fuerte carga de trabajo a la hora de ejecutar cualquier tipo de aplicación dado que es la nube la que se encarga del almacenamiento de dicha aplicación, y, por tanto, es la red de ordenadores la encargada soportar esta carga de trabajo. De esta forma, la labor que debe realizar el usuario se reduce y lo único que debe preocuparle es saber cómo navegar a través de Internet con el objetivo de poder acceder a los datos y procesos. Todo ello desemboca en una serie de ventajas [20]:

- **Reducción de costes:** El uso de la computación en la nube supone una reducción de costes importante en aspectos relacionados con el consumo energético y el mantenimiento tanto de equipos como de personal, ya que como se ha mencionado anteriormente, solo se paga por el uso de recursos [21].
- **Mayor accesibilidad y movilidad:** Gracias a que todos los recursos y servicios se encuentran alojados en la nube, solamente será necesario una conexión a Internet para poder acceder a ellos. Esto permite al usuario satisfacer sus necesidades desde cualquier sitio y desde cualquier dispositivo.
- **Más seguridad:** El usuario no debe preocuparse por realizar ningún tipo de copia de seguridad ya que es el propio servicio el encargado de realizar esta labor automáticamente.
- **Capacidad de almacenamiento ilimitada:** La nube dispone de una capacidad de almacenamiento prácticamente ilimitada. Gracias a esto, el cliente no tiene la necesidad de comprar hardware de almacenamiento externo.
- **Escalabilidad:** Esta es una de las ya mencionadas características principales del *Cloud Computing*.
- **Actualizaciones automáticas:** Es la propia nube la principal responsable de llevar a cabo el mantenimiento del servicio, por lo que el usuario no debe preocuparse por tener que realizar ningún tipo de instalación.
- **Optimización del uso de recursos:** Solamente se paga por el uso que se les da a los recursos contratados. Esto permite reducir los costos en la infraestructura necesaria y, por tanto, hacer un uso más óptimo de los recursos, lo que permite incrementar el rendimiento y la productividad.

Sin embargo, no todo son ventajas. El *Cloud Computing* presentan una serie de desventajas que hay que tener en cuenta y merecen ser mencionadas [16].

- Pérdida de control: Dado que nuestras aplicaciones están almacenadas en la nube, estamos sujetos a la normativa establecida por el proveedor, y por tanto, toda nuestra información corre a cargo de este.
- Confidencialidad y seguridad en los datos: Cualquier tipo de error que pueda producirse en los servidores puede comprometer nuestra información almacenada en ellos.
- Disponibilidad del servicio: En caso de que el servidor se caiga, no podemos acceder a la información.
- Acceso a Internet: Para poder acceder a los recursos es obligatorio disponer de una conexión a Internet.

### 2.2.4 Tipos de nubes

Distinguimos tres diferentes tipos de nubes. Cada nube nos permite dar respuesta a determinados problemas por lo que resulta importante entender cuáles son las ventajas y las desventajas, así como las diferencias existentes entre todas ellas [22][23].

- Nube pública: Se trata del tipo de nube más común de todas. En ella, es el proveedor del servicio el encargado tanto administrar todo el hardware, software y la infraestructura, así como de ofrecer bajo demanda los diferentes recursos a través de Internet. Entre las ventajas que tiene el uso de una nube pública destacamos:
  - Costos inferiores: Únicamente se paga por hacer uso del servicio que se desea por lo que no hay que hacer ningún desembolso en adquirir hardware ni software.
  - Sin mantenimiento: Como se ha mencionado anteriormente, es el proveedor encargado de esta tarea.
  - Escalabilidad casi ilimitada: Existen una gran cantidad de recursos capaces de satisfacer cualquier tipo de necesidad empresarial.
  - Gran confiabilidad: Se disponen de complejas redes de servidores que garantizan la no aparición de problemas.
- Nube privada: En este tipo de red los recursos son de uso exclusivo para los miembros de una institución. Por tanto, es una red privada la encargada de mantener toda la infraestructura. Además, el uso del hardware y software está restringido a la propia empresa u organización. Las ventajas que tiene el uso de una nube privada son:
  - Más flexibilidad: Es la empresa la que puede personalizar la nube con el objetivo de satisfacer sus necesidades específicas.
  - Más control: El hecho de que los recursos son de uso exclusivo para una empresa y no son compartidos con nadie más, les otorga un mayor nivel de control y privacidad.
  - Más escalabilidad: Normalmente, las nubes privadas son más escalables que las nubes públicas.
- Nube híbrida: Se trata de un tipo de nube que combina los dos tipos de nubes anteriormente citadas, con lo que la organización o empresa que

haga uso de este tipo de nubes puede beneficiarse de los servicios que ofrecen ambas nubes.

## 2.2.5 Servicios en la nube

Encontramos tres grandes grupos [16][24]:

- **IaaS (Infrastructure as a Service):** Se trata del modelo más complejo. Sin embargo, es el que más control otorga al propio cliente. En este servicio, los recursos utilizados por el cliente son obtenidos directamente a través del proveedor externo del servicio. Este tipo de servicio es comúnmente usado por administradores de sistemas, pues ofrecen facilidades como poder escoger el sistema operativo o la cantidad de almacenamiento requerida. Algunos ejemplos de *IaaS* son *Microsoft Azure* o *Amazon Web Services*.
- **PaaS (Platform as a Service):** En este modelo, a diferencia del *IaaS*, el cliente no tiene el control sobre la infraestructura de almacenamiento o de las redes, sino que es el propio proveedor el que proporciona todo lo necesario que llevar a cabo el desarrollo de las aplicaciones desde su desarrollo hasta su puesta en producción. Es, por tanto, un servicio muy utilizado por los desarrolladores de software ya que únicamente deben centrarse en el desarrollo del código. Ejemplos de este tipo de servicio son *Heroku* o *CloudFountry*.
- **SaaS (Software as a Service):** En este caso, el cliente únicamente podrá hacer uso de un software desarrollado por una empresa que se encuentra alojado en la nube. El mantenimiento, desarrollo, soporte y operaciones ya no es responsabilidad del cliente, sino de la empresa que ha desarrollado el software. En este modelo, el cliente no debe preocuparse por cuestiones técnicas como la infraestructura del servicio o cómo se lleva a cabo el proceso de mantenimiento, únicamente debe interesarle cómo se usa el servicio. *Microsoft Office 365* o las aplicaciones web de *Google* son ejemplos de este tipo de servicio.

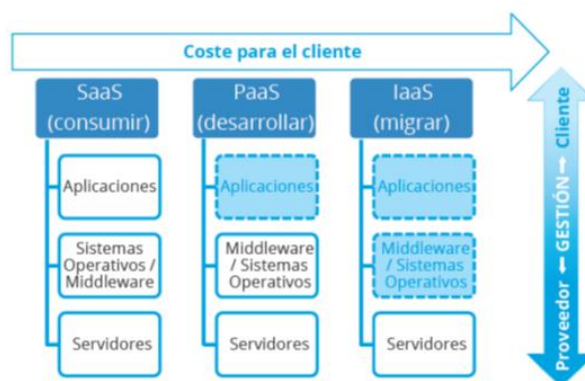


Figura 12. SaaS vs PaaS vs IaaS. Fuente: [16]

## 2.3 Redes neuronales

### 2.3.1 Aprendizaje automático

El aprendizaje automático, también conocido como *Machine Learning*, es un subconjunto de la Inteligencia Artificial cuyo objetivo es analizar e interpretar patrones y estructuras de datos que permiten el aprendizaje, razonamiento y la toma de decisiones sin interacción humana. El usuario, gracias al ML, puede proporcionar una gran cantidad de datos a un determinado algoritmo informático capaz de analizar toda la información, tomar decisiones y hacer recomendaciones basadas en los datos introducidos [25].

El *Machine Learning* es aplicable a todo tipo de sector, pero entre los casos más prácticos se encuentran [26]:

- Energía: Demanda de energía y optimización de suministro.
- Viajes y hostelería: Establecimiento dinámico de precios.
- Fabricación: Mantenimiento predictivo y supervisión condicionada.
- Comercio minorista: Venta adicional y marketing multicanal.
- Servicios sanitarios y ciencias biológicas: Identificación de enfermedades y satisfacción del riesgo.
- Servicios financieros: Análisis de riesgo y regulación.

En función de los datos de partida y del objetivo que se tenga, podemos encontrar diferentes tipos de aprendizaje automático:

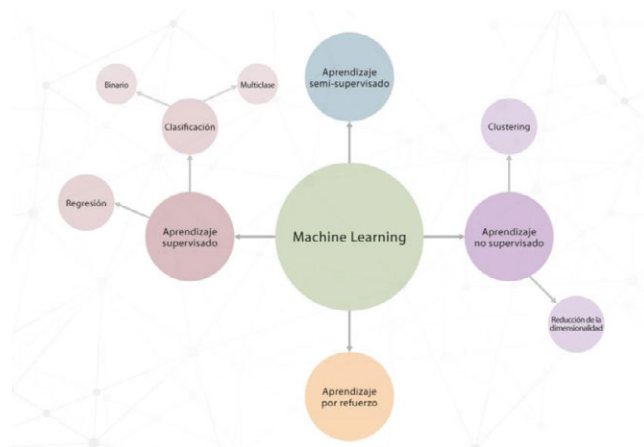


Figura 13. Tipos de aprendizaje automático. Fuente: [25]

- Aprendizaje supervisado: Requiere de datos etiquetados, esto es, un conjunto de datos de entrada y unas etiquetas asociadas a esos datos. Con ellos, el modelo debe encontrar el algoritmo correcto que sea capaz de asociar cada dato de entrada con su etiqueta y poder etiquetar también nuevos datos que pueda recibir en un futuro. En función del tipo de etiqueta, encontramos dos tipos de modelos:
  - Modelos de clasificación: Como salida producen un conjunto finito de etiquetas. Los modelos de clasificación pueden ser binarios o

multiclase. Los primeros son utilizados en caso de querer predecir entre dos clases o etiquetas (enfermedad o no enfermedad, por ejemplo) mientras que el segundo es utilizado cuando se tiene que clasificar más de dos clases.

- Modelos de regresión: Tienen como salida un valor real.
  
- Aprendizaje no supervisado: No requiere de datos etiquetados. El modelo recibe una gran cantidad de datos con los que debe aprender nuevas características y encontrar nuevos patrones que permitan clasificarlos y estructurarlos.
  
- Aprendizaje semi supervisado: Se trata de una combinación de los dos anteriores. En este modelo, se etiquetan una parte de todos los datos con el objetivo de poder encontrar varios algoritmos que sean capaces de asignar cada dato con su etiqueta asociada. Estos nuevos algoritmos se utilizarán para etiquetar las imágenes restantes.
  
- Aprendizaje por refuerzo: En este modelo se fijan una serie de objetivos a largo plazo para obtener una recompensa general máxima y lograr una solución óptima. De esta manera, se premia los comportamientos deseados y penalizar los no deseados.

### 2.3.2 Redes neuronales artificiales

Las redes neuronales forman parte de un campo dentro de la Inteligencia Artificial cuyo objetivo es crear modelos capaces de dar respuesta a problemas difíciles de resolver a través de técnicas algorítmicas convencionales. Están basadas en el funcionamiento de las redes neuronales biológicas. Como podemos ver en la *Figura 14*, una neurona biológica está formada por 3 componentes principales [27][28]:

- Soma: Se trata de la parte principal de la neurona. Presenta una forma variable y su función es producir la energía necesaria para el correcto funcionamiento de la neurona.
- Dendritas: Son prolongaciones del soma. Su función principal es recibir los impulsos procedentes de otras neuronales y enviarlas al soma.
- Axón: Prolongación del soma. Reciben el impulso de las dendritas y las transmiten a otras neuronas, músculos o glándulas del cuerpo humano.

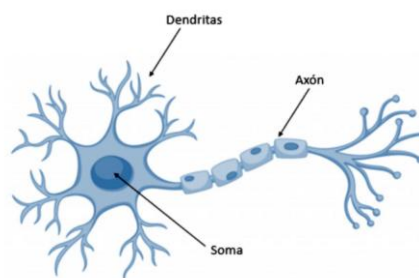


Figura 14. Neurona biológica. Fuente: [29]

En las neuronas artificiales, el funcionamiento es similar. El conjunto de entradas multiplicadas por sus pesos correspondientes representa el “impulso nervioso” recibido por la neurona. El procesamiento del impulso nervioso tiene lugar en una función de activación que devuelve un valor que se envía como salida de la neurona [29].

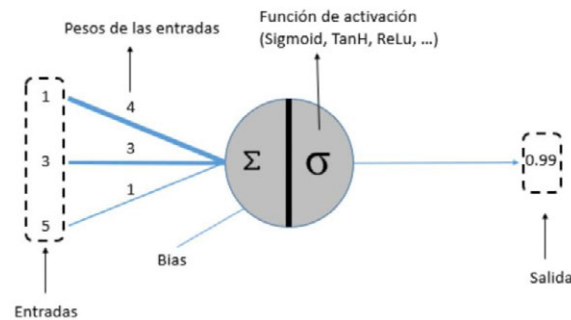


Figura 15. Neurona artificial. Fuente: [29]

Al igual que el cerebro humano, una red neuronal artificial está formada por un conjunto de neuronas conectadas entre sí y agrupadas en diferentes niveles, también denominadas capas. En una red neuronal encontramos tres capas principales:

- Capa de entrada: Capa que recibe como entrada los datos que alimentan a la red neuronal
- Capa de salida: Capa cuya salida es el resultado final de la red.
- Capas ocultas: Son las capas situadas entre la capa de entrada y la capa de salida.

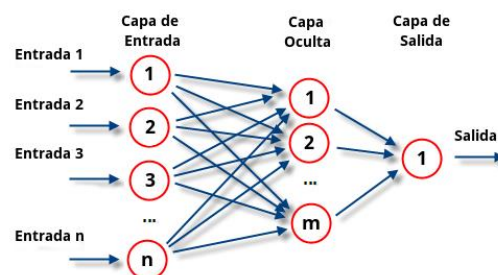


Figura 16. Red neuronal completa. Fuente: Google

La siguiente ecuación rige el comportamiento que presenta una neurona artificial, donde  $y$  es la salida de la función  $A$ , la cual representa la función de activación, de la suma de cada dato de entrada  $x$  multiplicada por su peso  $w$  correspondiente.

$$y = A \left( \sum_{i=1}^5 x_i w_i \right)$$

El peso de una entrada hace referencia al nivel de fuerza que presenta la unión entre la neurona que ha proporcionado dicha entrada y la neurona actual. Teniendo esto en cuenta, el proceso de aprendizaje de una red neuronal consiste en modificar el valor de cada uno de los pesos con el objetivo de alcanzar la salida final deseada a partir de unas determinadas entradas.

Por otra parte, la función de activación tiene como misión el cálculo de la salida en función de todas las entradas y pesos. Por tanto, el tipo de función de activación que elijamos supone una elección muy importante sobre el funcionamiento de la red neuronal. Podemos encontrar dos principales tipos de funciones de activación: lineal y no lineal.

### Función lineal

También llamada identidad, permite que la salida sea igual a la entrada. Por tanto, en una red neuronal de varias capas, si se hace uso de esta función como función de activación, se obtiene como resultado la llamada regresión lineal [30].

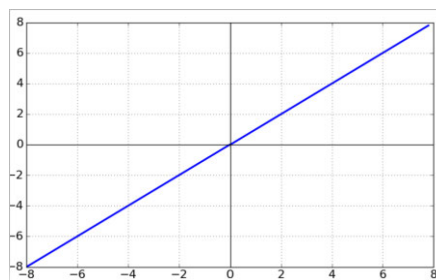


Figura 17. Función lineal

### Funciones no lineales

#### Función umbral

También conocida como función escalón, se emplea cuando se desea clasificar o cuando se tienen salidas categóricas. Es decir, si la entrada  $x$  es menor que 0, la salida va a ser 0 pero cuando la entrada es mayor o igual a 0, la salida será 1.

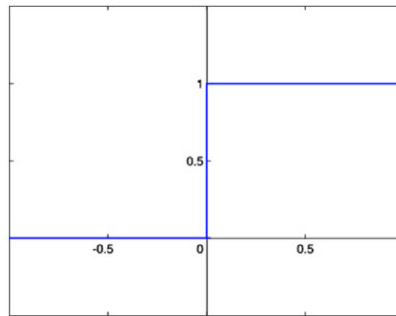


Figura 18. Función umbral

### Función sigmoide

También conocido como función logística. Se encuentra acotada en un rango de valores de salida entre 0 y 1, por tanto, la salida es interpretada como una probabilidad. Pese a no ser una función muy utilizada debido a que no está centrada (afecta en el aprendizaje y entrenamiento de la neurona), tiene aplicaciones en modelos usados para clasificar datos en dos categorías.

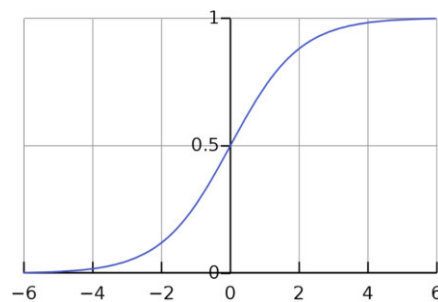


Figura 19. Función sigmoide

### Función tangente hiperbólica

Se trata de una función similar a la sigmoide. En lugar de estar acotada entre 0 y 1, la función tangente hiperbólica se encuentra acotada entre 1 y -1.

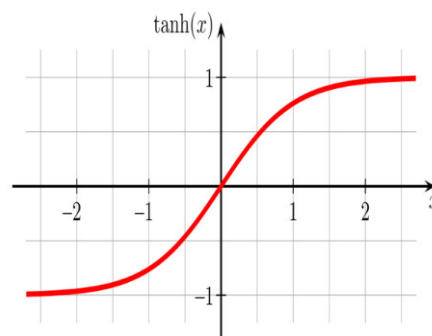


Figura 20. Función tangente hiperbólica

## Función ReLu

Se trata de una de las funciones de activación más utilizadas. Con esta función se obtienen aprendizajes muy rápidos en las redes neuronales. Si esta función recibe valores de entrada inferiores a 0, el resultado es 0. En cambio, si como entrada se reciben valores mayores a 0, la salida será la misma que la entrada.

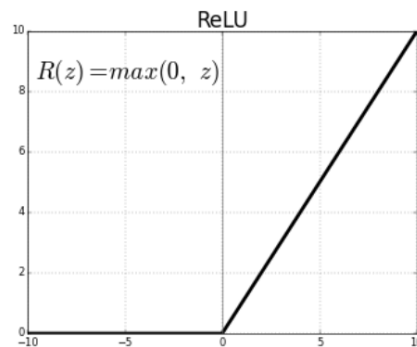


Figura 21. Función ReLu

### 2.3.3 Redes neuronales convolucionales (CNN)

Se trata una de las redes neuronales profundas más conocidas actualmente. La principal característica que presenta este tipo de redes es que cada neurona sólo recibe como parámetros de entrada la salida de algunas neuronas de la capa anterior [31].

A diferencia de las redes neuronales tradicionales, las CNN presentan un mayor número de capas. Por tanto, se debe tener en cuenta las capacidades en términos de GPU que tiene el dispositivo que va a llevar a cabo el proceso de entrenamiento de la red.

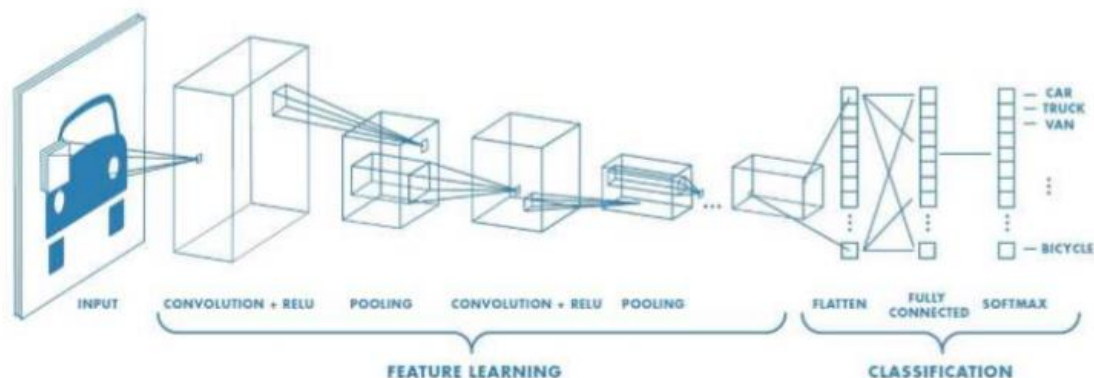


Figura 22. Estructura básica de un CNN

En la *Figura 22* se muestra una estructura básica de un CCN. Como podemos ver, esta red se compone de las siguientes capas [32]:

- **Capa de convolución:** Se trata de una capa oculta en la que se definen ciertos parámetros de ajustes, filtros y la función de activación asociada. La convolución hace referencia a un operador matemático que transforma dos funciones,  $a$  y  $b$ , dando como resultado una tercera,  $c$ , que representa en cierta manera la magnitud en la que se superponen  $a$  y una versión trasladada e invertida de  $b$ .
- **Capa de *pooling*:** Esta capa tienen como misión reducir el tamaño de la imagen. Entre las capas de *pooling* más utilizadas encontramos *max-pooling* y *average-pooling*.
- **Capa *fully connected*:** Constituye la red neuronal completa que se conecta a la última capa de la red. Necesita de una capa *Flatten* que tiene como objetivo realizar una adaptación intermedia para convertir los datos a vector. Posteriormente, tenemos la capa *Softmax* que se encarga de asociar una etiqueta a la imagen de entrada atribuyendo una probabilidad de pertenencia a dicha clase.

A lo largo del tiempo se han ido desarrollando diferentes arquitecturas CCN, cada una con sus ventajas y desventajas. Estas permiten definir el tipo y cantidad de filtros existentes en cada capa, el número de capas que conforman la red neuronal y la forma en la que se conectan dichas capas entre sí.

Uno de los principales inconvenientes que han presentado las arquitecturas a lo largo de su historia ha sido la limitación debido a las características hardware que presentaban las máquinas de aquel entonces. Sin embargo, con el transcurso de los años estas características han ido mejorando, permitiendo la creación de arquitecturas cada vez más grandes y con ello, la mejora de los resultados obtenidos. Entre las arquitecturas CNN más populares encontramos:

- **AlexNet:** Fue creada en 2012 por Alex Krizhevsky. Se trata de la primera arquitectura de redes neuronales estandarizada tras su participación en el ILSVRC. En esta red se utiliza *ReLU* como función de activación. De esta manera, se obtiene un mejor rendimiento de entrenamiento en comparación con la tangente hiperbólica y la sigmoide. En la *Figura 23* podemos ver esta arquitectura.

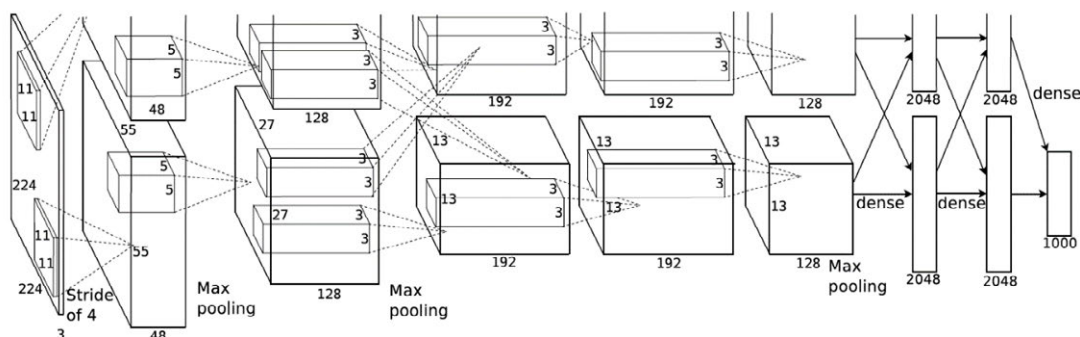


Figura 23. Arquitectura AlexNet. Fuente: [32]

- GoogLeNet: Fue propuesta por Christian Szegedy en 2014. Esta arquitectura se basa en modificar la arquitectura de la red, consiguiendo una disminución en el número de parámetros de la red (de 60 millones a 4 millones). Sustituyó la capa *fully connected* por una capa de *pooling* intermedia a través del método *average*. Esto le permitió obtener ahorro en costes y una considerable mejora en el proceso de entrenamiento. En la *Figura 24* podemos ver esta arquitectura.

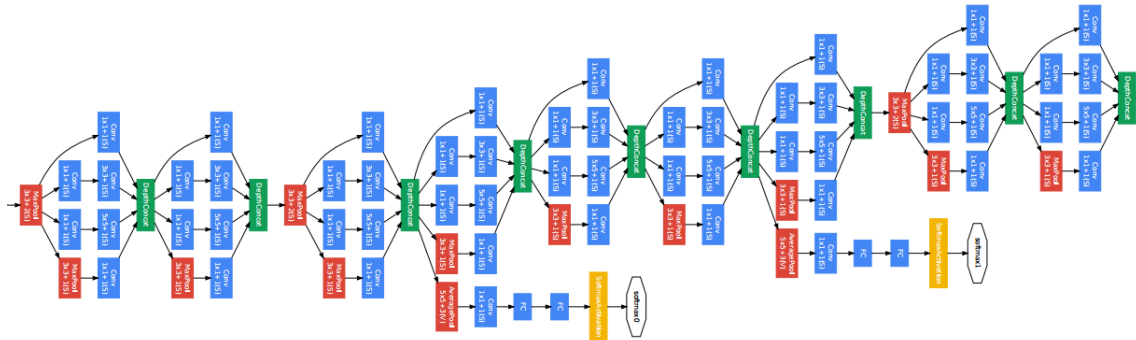


Figura 24. Arquitectura GoogLeNet. Fuente: [32]

- VGGNet: Fue desarrollado en 2014 de la mano de Simonyan y Zisserman. Con esta arquitectura ambos pudieron demostrar la importancia que tiene la profundidad de una red a la hora de obtener mejores resultados. Esta red hace uso de filtros de 3x3, lo que implica una disminución del número de parámetro en comparación con el modelo de Krizhevsky. En la *Figura 25* podemos ver esta arquitectura.

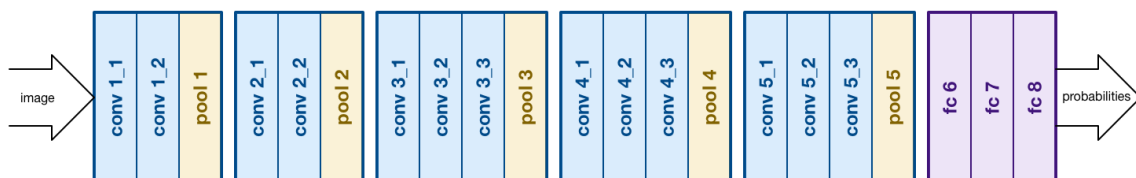


Figura 25. Arquitectura VGGNet. Fuente: [32]



### 3. Especificaciones y herramientas de diseño

Resulta importante mencionar y entender las características que presenta el modelo del que partimos inicialmente y de las diferentes etapas que lo conforman, así como de las herramientas adicionales que se van a usar para tener una mejor comprensión del proyecto.

#### 3.1 Python



Figura 26. Python

Se trata de un lenguaje de programación multiparadigma, orientado a objetos, imperativo y funcional. Fue creado por el ingeniero holandés Guido Van Rossum en los años 90 cuando trabajaba en el Centro de Investigación de Ciencias de la Computación de Holanda [33][34].

Se trata de un lenguaje de programación simple y fácil de utilizar lo que hace que sea el lenguaje favorito para el *scripting*. Debido a su función multiplataforma, gracias a la cual podemos operar con grandes volúmenes de datos, *Python* se trata de un lenguaje ideal para trabajar en el mundo del *Big Data* e Inteligencia Artificial.

Además, gracias a su sencillez, se encuentra entre los lenguajes de programación más utilizados por aquellos que quieren introducirse en el mundo de la programación.

De entre las diferentes librerías que proporciona *Python*, destacamos las siguientes para la realización del proyecto:

- *os*: Proporciona una forma portátil de utilizar la funcionalidad dependiente del sistema operativo [35].
- *csv*: Incorpora clases que permiten leer y escribir datos en formatos *csv* [36].
- *random*: Implementa diferentes generadores de números pseudoaleatorios para varias distribuciones [37].
- *PIL*: Ofrece un amplio soporte de formato de archivo, una representación interna eficiente y capacidades de procesamiento de imágenes [38].
- *matplotlib*: Permite crear visualizaciones estáticas, animadas e interactivas [39].
- *numpy*: Ofrece funciones para trabajar en el dominio del álgebra lineal, transformada de *Fourier* y matrices [40].

- *time*: Proporciona varias funciones que permiten trabajar con el tiempo [41].
- *tensorFlow*: Plataforma *opensource* extremo a extremo destinada a el aprendizaje automático. Es desarrollado por *Google* con el objetivo de construir y entrenar redes neuronales [42].
- *keras*: Funciona como una interfaz de uso intuitivo gracias a la cual poder acceder a *frameworks* de aprendizaje automático. El objetivo de esta biblioteca es acelerar la creación de redes neuronales [43].

## 3.2 Google Colaboratory



Figura 27. Google Colaboratory

También conocida como *Colab*, se trata de una herramienta gratuita desarrollada por *Google* en la nube que nos permite ejecutar y programar código escrito en *Python* en el navegador a través de *Jupyter Notebook*. Estos *notebooks* se almacenan en *Google Drive* y pueden ser compartidos [44].

*Colab* funciona correctamente con los principales navegadores como *Chrome*, *Firefox* y *Safari*. Este servicio solo soporta *Python* como lenguaje de programación. Sin embargo, se está barajando la posibilidad de que la herramienta sea compatible con otros *kernels* de *Jupyter* como *R* o *Scala*, pero no se tiene una fecha estimada de lanzamiento.

La principal ventaja que presenta este servicio es el acceso gratuito de GPUs y TPUs de *Google* a través de librerías como: *Scikit-learn*, *PyTorch*, *tensorFlow*, *keras* y *OpenCV*.

Entre las diferentes GPUs que ofrece *Google*, en el momento de la elaboración del proyecto, encontramos las siguientes: K0, T4, P4 y P100 de NVIDIA. Sin embargo, estas varían con el tiempo y el usuario no puede elegir el tipo de GPU a utilizar, sino que es asignada por el propio *Colab* de manera interactiva en función del histórico del usuario.

Para la ejecución del código, los *notebooks* se conectan a diferentes máquinas virtuales. Los *notebooks* presentan un tiempo de vida útil de hasta 12 horas y estos se desconectan automáticamente de las VM cuando permanecen inactivos durante un largo período de tiempo. Sin embargo, el tiempo de vida útil y tiempo de espera de las máquinas virtuales varían con el tiempo y dependen del uso que se les dé.

### 3.3 Microsoft Azure



Figura 28. Microsoft Azure

Se trata de una nube pública desarrollada por *Microsoft* de pago por uso en la cual podemos compilar, implementar y administrar aplicaciones en una red global desarrollada por la misma empresa. Los servicios que podemos encontrar en *Azure* abarcan servicios web sencillos, equipos totalmente virtualizados para ejecutar soluciones de software personalizados, almacenamiento remoto, hospedaje de bases de datos o administración centralizada de cuentas, entre otras muchas [45]. Al estar trabajando con modelos de redes neuronales convolucionales, haremos uso de uno de los tres principales servicios que *Microsoft* nos ofrece para ello, *Azure Machine Learning*. Se trata de una plataforma que nos permite hacer predicciones. Está formado por diferentes herramientas y servicios gracias a los cuales podemos obtener datos y entrenar redes neuronales con el objetivo de poder encontrar el modelo que mejor prediga un resultado futuro.

### 3.4 ISIC Challenge 2019

Los datos que se han utilizado para el desarrollo y entrenamiento de la red neuronal son proporcionados por el ISIC. En concreto, se va a usar los del año 2019, los cuáles podemos encontrar en la página *ISIC Challenge 2019*. De toda la información que podemos encontrar, vamos a utilizar la siguiente:

- *ISIC\_2019\_Training\_Input*: Una carpeta con un total de 25331 imágenes capturadas en diferentes partes de la piel procedentes de personas de diferente edad, sexo y raza.
- *ISIC\_2019\_Training\_GroundTruth.csv*: Un archivo *csv* con los metadatos asociados a cada una de las imágenes: edad, sexo, parte del cuerpo en la que se ha tomado la foto y el diagnóstico.

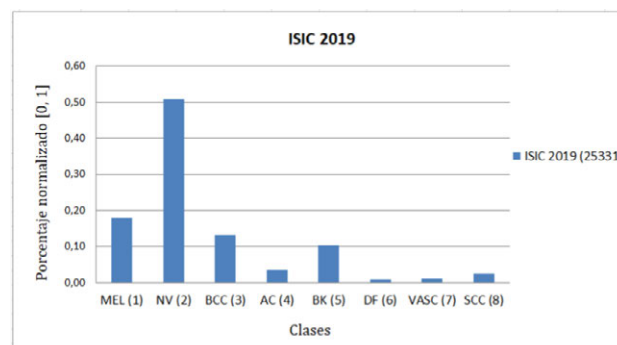


Figura 29. ISIC 2019

Como podemos ver en la *Figura 29*, disponemos de imágenes que pueden pertenecer a 9 clases diferentes entre las que encontramos:

- MEL (*Melanoma*): Gran parte de las imágenes pertenecen a esta clase. Representan el 17.85% del número total de imágenes.
- NV (*Melanocytic Nevus*): En este grupo se encuentran imágenes que hacen referencia a manchas, pecas o lunares. Conforman el 50.83% del número total de imágenes, haciéndolo el grupo más mayoritario de todos.
- BCC (*Basal Cell Carcinoma*): Dentro de esta clase se encuentran los ya mencionados cánceres de piel de células basales. Dentro de esta clase encontramos un total de 3323 imágenes.
- AK (*Actinic Keratosis*): En esta clase encontramos imágenes captadas en una fase previa al cáncer, también denominado precáncer, que se produce cuando se expone la piel a la radiación solar. Este grupo representa el 3,42%.
- BKL (*Benign keratosis*): Se trata de un crecimiento cutáneo libre de cáncer. Este aumenta a medida que pasan los años. Este grupo representa el 10,36% número total de imágenes.
- DF (*Dermatofibroma*): Se trata de un tumor cutáneo benigno que se da comúnmente en las extremidades inferiores de las mujeres. Es la clase más minoritaria de las 9, representando únicamente el 0,94%.
- VASC (*Vascular Lesion*): Se trata de un golpe o traumatismo que tiene como consecuencia una lesión en arterias o venas. Al igual que la clase anterior, disponemos de pocas imágenes de este tipo, solo el 0,998%.
- SCC (*Squamous Cell Carcinoma*): Cáncer que se desarrolla en las ya mencionadas células escamosas. Representa el resto de las imágenes, 2,48%.
- UNK (*Unknow*): Dentro de este grupo se encuentran las imágenes de las que no sabe su origen. Como podemos ver, no disponemos de ninguna imagen de esta clase.

Como se ha mencionado anteriormente, este proyecto tiene como objetivo continuar con el trabajo llevado a cabo por Mario Álvaro Díaz en su proyecto *Optimización de una red neuronal convolucional para la detección de melanomas en imágenes dermoscópicas*. Es por ello, que para la realización de este proyecto se hará uso de su red neuronal. En el *Anexo 1* se podrá encontrar información más detallada acerca de los *scripts* de los que se parte de inicio en este proyecto y que conforman la arquitectura de la red implementada.

## 4. Descripción y desarrollo de la solución propuesta

El principal objetivo de este proyecto es la implementación de una red neuronal convolucional basada en la nube que nos permita identificar y clasificar imágenes en base a un determinado criterio. Para ello, este proyecto se dividirá en tres diferentes etapas: la implementación de la red neuronal convolucional disponible en diferentes plataformas *cloud*, el análisis del comportamiento de la red para distintas configuraciones de sus hiperparámetros y el desarrollo de nuevas funcionalidades que nos permitan mejorar el rendimiento de dicha red.

Para la primera parte del proyecto, los servicios elegidos son *Google Colaboratory* y *Microsoft Azure*. A lo largo de esta etapa, el objetivo será implementar la red de la que partimos en cada uno de estos servicios. Para ello, se subirán los diferentes ficheros que conforman la arquitectura a cada una de estas plataformas realizando los cambios que se consideren necesarios para su correcto funcionamiento. Posteriormente, se ejecutarán dichos ficheros e iremos midiendo los tiempos de ejecución que obtenemos en ambos servicios para cada uno de los ficheros. En base a los resultados obtenidos y prestaciones que nos ofrecen cada uno de ellos, elegiremos el que mejor nos convenga para llevar a cabo el desarrollo de las siguientes partes del proyecto.

Para la segunda parte, haciendo uso de la plataforma seleccionada en la primera parte, se realizarán las modificaciones que se crean convenientes en la red de la que partimos inicialmente con el objetivo de ver la respuesta que se obtiene por su parte ante diferentes variaciones en los parámetros que rigen su comportamiento y poder así obtener conclusiones finales. Finalmente, se desarrollarán nuevas funcionalidades que permiten mejorar de cierta forma el comportamiento de la red y con ello, la posibilidad de obtener resultados mucho más precisos.

### 4.1 Implementación del modelo en Google Colab y Microsoft Azure

Esta primera fase tendrá como objetivo definir cuál va a ser la plataforma en la que se lleve a cabo el análisis de la red para distintas configuraciones y el desarrollo de la nueva funcionalidad que tendrá lugar en la tercera parte del proyecto. Para ello, tendremos que implementar el modelo en cada una de estas plataformas.

#### 4.1.1 Implementación en Google Colab

##### Creación de una cuenta Google Drive

El primer paso para la implementación de la red neuronal en *Google Colab* es hacer uso de una cuenta de *Google Drive*. *Google Drive* es un servicio de almacenamiento de datos en la nube ofrecida por *Google*. Por tanto, necesitaremos un correo electrónico de *Gmail* para acceder a este servicio.

Por defecto, *Google Drive* ofrece de manera gratuita un servicio de almacenamiento de hasta 15 GB, en los cuales podemos almacenar cualquier tipo de archivo (fotos, videos o archivos de presentación) independientemente del sistema operativo de nuestro dispositivo. Además, disponemos de tres diferentes planes de pago que nos ofrecen una mayor capacidad de almacenamiento entre las que encontramos: 100 GB con un precio de 1,99 €/mes o 19,99 €/año, 200 GB con un precio de 2,99 €/mes o 29,99 €/año y 2 TB con un precio de 9,99 €/mes o 99,99 €/año. Adicionalmente, podemos encontrar suscripciones de nos otorgan más capacidad de almacenamiento en caso de que sea necesario, llegando a los 10 TB, 20 TB o 30 TB.

Dado que tan solo la carpeta que contiene todas las imágenes con la que vamos a trabajar tiene un peso de 9,12 GB y que además vamos a crear diferentes directorios que a su vez van a contener una gran cantidad de imágenes cada uno, la versión gratuita de 15 GB de almacenamiento no es suficiente. Por tanto, para la realización de este proyecto se ha hecho uso de una cuenta de *Google Drive* de 100 GB de espacio de almacenamiento.

### Creación de un libro

Un cuaderno es un documento que contiene código ejecutable y elementos de texto que enriquecen nuestro código como pueden ser *links* o figuras. Para la creación de un nuevo cuaderno que contenga el código de nuestra red neuronal, pulsaremos en *Nuevo* y en *Más* para posteriormente seleccionar la opción *Colaboratoy* como vemos en la *Figura 30*.

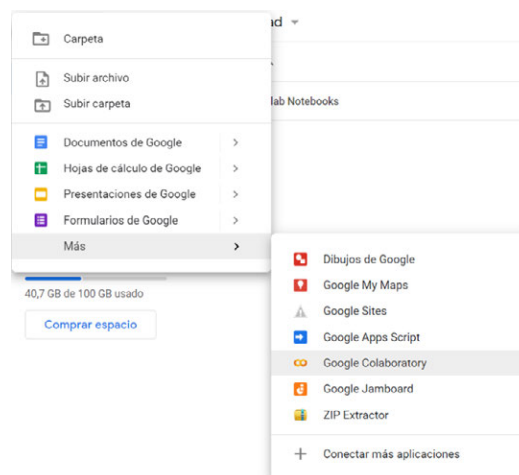


Figura 30. Instalación de Colaboratory

En caso de que no aparezca esta opción pulsaremos en *Conectar más aplicaciones*. Posteriormente escribiremos *Colaboratory* en el buscador y añadiremos ese complemento a nuestro *Google Drive*.

Una vez hechos los pasos anteriores se nos abrirá un fichero con formato *.ipynb*. Este formato viene de *IPython Notebook* y es un formato que nos permite ejecutar código tanto en *IPython* como en *Jupyter* y *Colab*. En la parte superior de este fichero podemos cambiar el nombre de este cuaderno.

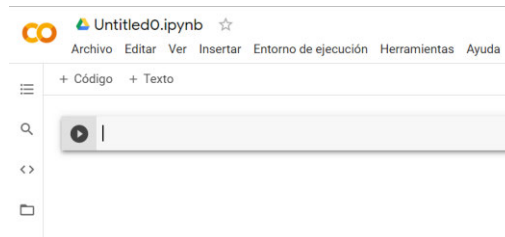


Figura 31. Cuaderno en Colaboratory

## Celdas

Una de las particularidades que tiene *Google Colab* es la capacidad de poder dividir nuestro código en pequeñas unidades independientes que pueden ser dictadas y ejecutadas. Las celdas nos permiten separar nuestro código en bloques lógicos. Encontramos diferentes tipos de celdas:

- Código: Contiene el código ejecutable.
- Texto: Contiene información descriptiva.

### ▼ Principio

```
[2] print("primera linea")
primera linea

[3] print("segunda linea")
segunda linea

▶ print("tercera linea")
tercera linea
```

### Final

Figura 32. Celdas en Colaboratory

## Entornos de ejecución

Pese a que las celdas son independientes entre sí, todas ellas utilizan el mismo *kernel*. Este se trata del motor de computación encargado de ejecutar todo el código y devolver el resultado para que sea mostrado por pantalla. Dado que el estado del *kernel* persiste durante el tiempo, una celda puede acceder a las variables declaradas en otra celda diferente.

Cuando creamos un cuaderno, tenemos que conectarnos a un entorno de ejecución para poder llevar a cabo una ejecución. En *Google Colab*, cuando ejecutamos una celda o pulsamos el botón *conectar*, el cuaderno se conecta a una VM (*Virtual Machine*) de *Google Compute Engine* que cuenta con 12 GB de RAM y 50 GB de almacenamiento.

El flujo de ejecución de las celdas es de arriba hacia abajo, es decir, el orden en que han sido creadas, pero siempre cabe la posibilidad de querer cambiar dicho flujo de ejecución. Para ello hacemos uso de los números que encontramos en la parte izquierda de la celda. Este número indica el orden de ejecución que tiene esa celda.

A modo de ejemplo, cómo podemos ver en la *Figura 33*, en primer lugar, se importa la librería *numpy* y luego se declara una función para elevar números al cuadrado. En celdas siguientes, se utiliza esa función y la librería para elevar al cuadrado números aleatorios.

```
▶ import numpy as np

def elevar_al_cuadrado(x):
    return x*x

[3] x = np.random.randint(1,10)
    y = elevar_al_cuadrado(x)

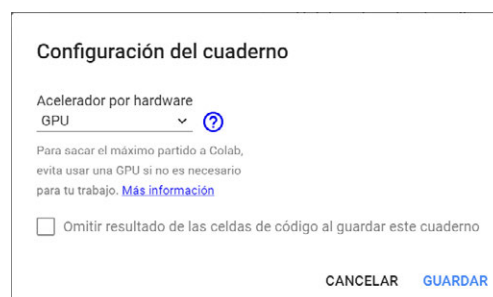
[4] print('%d elevado al cuadrado es %d' % (x, y))
    5 elevado al cuadrado es 25
```

*Figura 33. Flujo de ejecución en Colab*

Hay que tener en cuenta que sea cual sea el entorno de ejecución escogido, las máquinas virtuales a las que nos conectaremos tienen un tiempo de vida máximo de 12 horas, es decir, el tiempo máximo de conexión a estas máquinas es de 12 horas. Además, existe un tiempo máximo de inactividad a partir del cual el entorno se desconecta automáticamente. Este tiempo es de 90 minutos.

## Conexión a GPU

*Colab* nos permite cambiar el entorno de ejecución de manera gratuita. Para ello, en el menú *Editar* seleccionamos la opción *Configuración del cuaderno* y en la opción *Acelerador por hardware* elegimos la opción GPU.



*Figura 34. Conexión a GPU*

De esta manera, *Google Colab* hará los cambios necesarios para ofrecernos una máquina con estos requisitos. Posteriormente, ejecutaremos en una celda el código que se muestra en la *Figura 35*. Con este trozo de código comprobamos que la conexión al nuevo entorno de GPU se ha llevado a cabo de manera correcta.

```
%tensorflow_version 2.x

import tensorflow as tf
device_name = tf.test.gpu_device_name()
if device_name != '/device:GPU:0':
    raise SystemError('GPU device not found')
print('Found GPU at: {}'.format(device_name))
```

Found GPU at: /device:GPU:0

*Figura 35. Comprobación de conexión correcta GPU*

Sin embargo, por temas de limitación de recursos puede darse la situación de que no podamos conectarnos a una máquina con GPU. Entre los motivos, encontramos la posibilidad de que, puede haber múltiples usuarios conectados al mismo al tiempo. De esta manera, existe la posibilidad de que en medio de una ejecución el entorno se desconecte.

## 4.1.2 Implementación en Azure

### Creación de una cuenta Azure

Para crear y hacer uso de los servicios que podemos encontrar en *Azure*, necesitamos en primer lugar, una suscripción. Para la realización de este proyecto, la Universidad Politécnica de Madrid se ha hecho cargo de los pagos necesarios para disponer de una cuenta con un fondo inicial de 100 €. Posteriormente, se necesitó una suscripción adicional. Esta vez, de 280 €.

### Creación de un área de trabajo

En *Azure ML*, un área de trabajo es un recurso de nivel superior que proporciona un lugar centralizado para trabajar con los recursos que han sido creados. A la hora de crear un nuevo área de trabajo, el usuario debe especificar la información mostrada en la *Tabla 1*.

*Tabla 1. Creación de área de trabajo*

Campo	Descripción
Nombre del área de trabajo	Nombre que identifica unívocamente el área de trabajo

<b>Subscripción</b>	Subscripción <i>Azure</i> que se desea usar
<b>Grupo de recursos</b>	Almacena los recursos relacionados con una solución <i>Azure</i>
<b>Localización</b>	Ubicación más cercana a los usuarios y recursos de datos
<b>Edición del área de trabajo</b>	Características a las que tendrá acceso y los precios.

### Creación de un proceso

Una instancia de proceso se trata de una estación de trabajo administrada basada en la nube que permite a los científicos de datos hacer uso de *Azure Machine Learning* de una forma mucho más fácil. Esta herramienta proporciona herramientas que permite la administración y la preparación empresarial de los administradores de IT. Encontramos diferentes maneras de instanciar un nuevo proceso. En este proyecto se hará desde la sección *Notebooks* a la hora de ejecutar uno de los cuadernos.

Para crear un proceso en primer lugar se debe seleccionar la máquina virtual que se quiera usar para la instancia del proceso. Para ello se debe rellenar el formulario de la *Figura 36* que contiene la siguiente información:

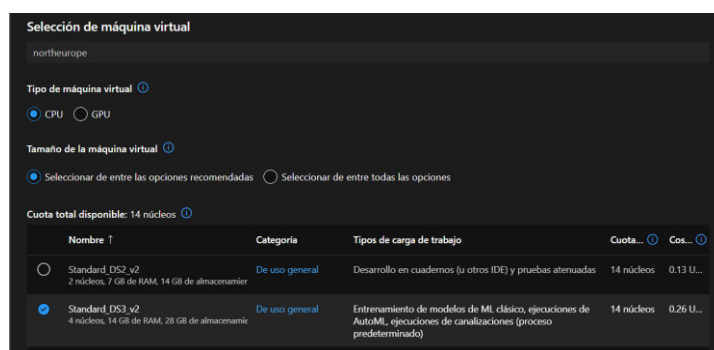


Figura 36. Selección MV

Tabla 2. Selección MV

Campo	Descripción
<b>Tipo de máquina virtual</b>	Se debe elegir entre CPU o TPU. Una vez creado el proceso, esta opción no podrá cambiarse
<b>Tamaño de la máquina virtual</b>	Este parámetro está restringido a la región

Una vez elegida la máquina virtual, debemos configurar todas opciones de la instancia de proceso para el tamaño de la máquina virtual seleccionada.

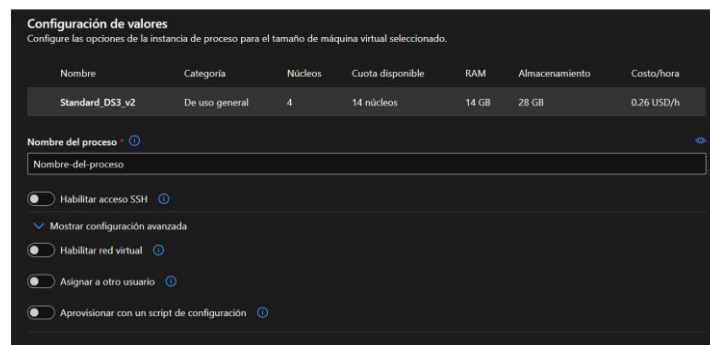


Figura 37. Configuración MV

Tabla 3. Configuración MV

Campo	Descripción
<b>Nombre del proceso</b>	<ul style="list-style-type: none"> <li>• Parámetro obligatorio. Debe tener una longitud de entre 3 y 24 caracteres</li> <li>• Los caracteres válidos son mayúsculas, minúsculas, dígitos y el carácter –</li> <li>• Debe empezar en mayúscula</li> <li>• Debe ser único en todos los procesos existentes dentro de una región de <i>Azure</i></li> <li>• Si se usa el carácter -, debe ir seguido de al menos una letra más adelante en el nombre</li> </ul>
<b>Habilitar/deshabilitar el acceso SSH</b>	De forma predeterminada, el acceso SSH se encuentra deshabilitado. De igual manera que en el tamaño de la máquina virtual, tras la creación del proceso este campo no se podrá modificar.
<b>Configuración avanzada</b>	Campo opcional. En esta opción encontramos aspectos relacionados con el grupo de recursos, red virtual y subredes para crear un proceso dentro de una VNET ( <i>Azure Virtual Net</i> ).

## 4.2 Análisis para distintas configuraciones de la red neuronal

El hiperparámetro con el que se va a trabajar es el *dropout*. Este se trata de un método cuyo objetivo es desactivar de manera aleatoria una cantidad de neuronas de la red neuronal. Gracias a ello, se es capaz de reducir el llamado *overfitting*.

El *dropout* se indica por medio de un parámetro que hace referencia a la probabilidad de que las neuronas queden activadas. Toma valores entre 0 y 1. Para valores próximos a 0 se desactivarán pocas neuronas mientras que para valores próximos a 1 se desactivarán muchas más.

De esta manera, los análisis que se van a hacer serán para diferentes valores de este parámetro, en concreto, para valores de 0.1, y 0.4. Se entrenarán a los modelos para estos valores y posteriormente se procederá a comparar los resultados obtenidos en cada uno de ellos con los obtenidos para la configuración inicial de la red neuronal, la cual presenta un *dropout* de 0.2. Para ello, se usarán diferentes conjuntos de entrada. Estos son:

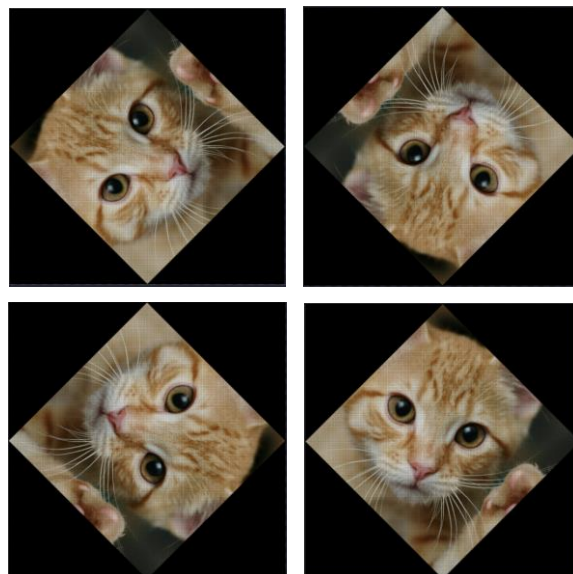
- *test*
- *testCropCenter*
- *testAugmentedCropCenter*
- *testAugmentedCropCenterRotatedAll*

### 4.3 Nueva funcionalidad a desarrollar

Con el objetivo de poder tener más imágenes de entrada con la que poder entrenar nuestra red neuronal, se ha decidido que la nueva funcionalidad a desarrollar en este proyecto sea una nueva técnica de generación artificial de datos a partir de modificaciones en los datos originales, es decir, añadir una nueva técnica de *data augmentation*.

Hasta ahora se ha contado con dos etapas de *data augmentation*: la primera de ellas tiene como objetivo de obtener imágenes simétricas con respecto a los ejes vertical y horizontal para posteriormente, en la segunda etapa generar una imagen rotada 90° con respecto a cada una de las imágenes nuevas obtenidas y la original.

En esta última fase de este proyecto se tiene como objetivo realizar una rotación de cuatro ángulos: 45°, 135°, 225° y 315° con respecto a la imagen original. En *Figura 38* podemos ver los resultados que obtendríamos con esta nueva funcionalidad



*Figura 38. Resultado nuevo data augmetation*

## 5. Resultados

En este apartado se van a exponer los resultados obtenidos en las diferentes partes del proyecto.

### 5.1 Resultados de la implementación del modelo en servicios en la nube

En esta primera parte, se va a utilizar como métrica de evaluación el tiempo de ejecución. Este parámetro dependerá del tipo de entorno de ejecución empleado.

Para medir dichos tiempos se harán uso de la librería *time* que como se ha especificado en el apartado 3.1, proporciona varias funciones relacionadas con el tiempo. De todas ellas, se va a hacer uso de la función *perf\_counter()*.

En la *Tabla 4* están reflejados los diferentes tiempos de ejecución obtenidos para cada uno de los diferentes ficheros en los servicios de nube utilizados en este proyecto.

*Tabla 4. Tiempos de ejecución Google Colab vs Microsoft Azure*

		Plataformas	
		Google Colab	Microsoft Azure
FASE 1	tranformed	5 segundos	10 segundos
	add_jpg_extension	26 segundos	11 segundos
	rename_images_and_csv_by_class	1 hora 30 minutos	45 minutos
	create_train_val_test	6 horas	2 horas 15 minutos
FASE 2	crop_center	35 minutos	45 minutos
	augmentation_simmetries	1 hora 30 minutos	2 horas 15 minutos
	augmentation_Rotate_Square	5 horas 20 minutos	5 horas 30 minutos
FASE 3	Entrenamiento	84 horas	49 horas
<b>TOTAL</b>		<b>98 horas 55 minutos 31 segundos</b>	<b>60 horas 30 minutos 21 segundos</b>

Como podemos ver, en casi todos los ficheros vamos obteniendo resultados muy parecidos. Sin embargo, se observa que, para tiempos de ejecución largos, *Microsoft Azure* presenta una mejor respuesta, tardando casi 40 horas menos que *Google Colab* a la hora de entrenar el modelo, resultando decisivo a la hora de elegir cual es el mejor servicio. No obstante, cabe recordar que para *Microsoft Azure* se ha tenido que pagar por hacer uso de una máquina virtual mientras que el entorno de ejecución para *Google Colab* resulta gratuito. Esta primera fase supuso un coste económico de aproximadamente 90€ en *Microsoft Azure*.

De todas formas, este desembolso económico no supone ningún obstáculo viendo el ahorro que obtenemos en tiempos de ejecución. De esta manera, la plataforma elegida para llevar a cabo la segunda parte es *Microsoft Azure*.

Una vez implementada la red neuronal en estos servicios y la posterior elección de la plataforma idónea, se va a proceder a modificar alguno de los hiperparámetros que rigen el comportamiento de la red neuronal con el objetivo de ver cómo se comporta dicha red y sacar conclusiones.

### Métricas de evaluación para la segunda y tercera fase del proyecto

Para llevar a cabo el análisis de los resultados que se vayan obteniendo a lo largo de las diferentes pruebas realizadas en la segunda y tercera fase, se va a hacer uso de cuatro indicadores de calidad: exactitud, precisión, sensibilidad y F score. Estos indicadores dependen de la comparación entre los resultados obtenidos por el modelo y los resultados reales proporcionados por los expertos. Encontramos 4 tipos de datos que podemos encontrar en los resultados reales. En la *Tabla 5* podemos ver más en detalle cuáles son estos cuatro parámetros:

Tabla 5. Tipo de datos tras detección

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	TP	FP
	Negativo	FN	TN

- *True Positive* (TP): Verdadero positivo. Tanto el modelo como el diagnóstico detectan la imagen como positiva.
- *False Positive* (FP): Falso positivo. El modelo detecta la imagen como positiva mientras que el diagnóstico real lo detecta como negativo.
- *True Negative* (TN): Negativo verdadero. Tanto el modelo como el diagnóstico detectan la imagen como negativa.
- *False Negative* (FN): Falso negativo. El modelo detecta la imagen como negativo mientras que el diagnóstico real lo detecta como positivo.

Atendiendo a estos parámetros, podemos definir los indicadores de calidad:

- Exactitud (*accuracy*): Es la relación existente entre el número de aciertos totales entre el número total de elementos de trabajo.

$$Exactitud = \frac{TP + TN}{TP + FP + TN + FN}$$

- Precisión (*precision*): Es la relación entre los positivos verdaderos (TP) y el número de elementos que han sido detectados como positivos.

$$Precisión = \frac{TP}{TP + FP}$$

- Sensibilidad (*recall*): Es la relación entre los positivos verdaderos (TP) y el número de elementos que son realmente positivos (TP + FN).

$$Sensibilidad = \frac{TP}{TP + FN}$$

- F score: Este parámetro relaciona la sensibilidad y la precisión.

$$F \text{ score} = \frac{2 * Precisión * Sensibilidad}{Precisión + Sensibilidad}$$

De esta manera, se calcularán estos parámetros para la configuración inicial de la red neuronal de la que partimos de base, que como se ha mencionado, está configurado con un *dropout* de 0.2 y se compararán con los resultados obtenidos para un *dropout* de 0.1 y 0.4. Los resultados obtenidos para la configuración inicial se encuentran en el *Anexo 2* de este proyecto.

## 5.2 Resultados de los análisis para distintas configuraciones de la red

### 1\_test\_450x650

Para el primer entrenamiento se va a hacer uso del conjunto de datos *test* creado en la fase *1\_create\_train\_val\_test*. El principal objetivo de este primer entrenamiento es ver cómo responde el modelo sin realizar ningún tipo de transformación o modificación en las imágenes de entrada.

Para un *dropout* de 0.1 obtenemos unos resultados más que aceptables en todos los indicadores de calidad, superando en todos ellos los valores obtenidos en la configuración inicial de la red neuronal. En la *Tabla 6* podemos ver más

detalladamente los resultados obtenidos para cada uno de los cuatro indicadores de calidad.

Tabla 6. dropout 0.1 para 1\_test\_450x650

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	851	480
	Negativo	530	1331

$$Exactitud = \frac{851 + 1331}{851 + 530 + 480 + 1331} = 0.68$$

$$Precisión = \frac{851}{851 + 480} = 0.64$$

$$Sensibilidad = \frac{851}{851 + 530} = 0.62$$

$$F\ score = \frac{2 * 0.64 * 0.62}{0.64 + 0.62} = 0.63$$

Por otra parte, para un dropout de 0.4 obtenemos unos valores finales prácticamente iguales a los obtenidos con un dropout de 0.1. Ambos presentan la misma exactitud. Sin embargo, para un dropout de 0.1 obtenemos una ligera mejora en cuanto a la precisión y F score, mientras que para un dropout de 0.4 la mejora la obtenemos en la sensibilidad. En la Tabla 7, vemos los resultados obtenidos para un dropout de 0.4.

Tabla 7. dropout 0.4 para 1\_test\_450x650

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	843	536
	Negativo	488	1331

$$Exactitud = \frac{843 + 1331}{843 + 488 + 536 + 1331} = 0.68$$

$$Precisión = \frac{843}{843 + 536} = 0.61$$

$$Sensibilidad = \frac{843}{843 + 488} = 0.63$$

$$F\ score = \frac{2 * 0.61 * 0.63}{0.61 + 0.63} = 0.62$$

Para esta primera prueba, obtenemos los mejores resultados para un *dropout* de 0.1 y los peores para la configuración inicial de 0.2. Sin embargo, hay que tener en cuenta que el número de datos con el que trabajamos es relativamente bajo. Por tanto, a medida que vayamos aumentando el número de imágenes con el que trabajamos iremos viendo que los resultados serán más diferentes.

### 1\_CropCenter\_450x650

Este segundo entrenamiento tendrá como objetivo ver la influencia que tiene la relación de aspecto de las imágenes que se utilizan como entrada de la red neuronal. En este segundo entrenamiento se va a hacer uso del conjunto de datos *testCropCenter* creado en la fase *2\_Data\_Augmentation*.

Para un *dropout* de 0.1 vemos que obtenemos peores valores en todos los parámetros de calidad en comparación con la configuración inicial y en especial, en la exactitud. Es decir, con esta configuración y haciendo uso de imágenes con una relación de aspecto 1:1, el modelo asocia menos imágenes correctamente que con la relación de aspecto original.

En la *Tabla 8* vemos de forma más detallada los valores obtenidos para los indicadores de calidad.

*Tabla 8. dropout 0.1 para 1\_CropCenter\_450x650*

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	847	484
	Negativo	537	1331

$$Exactitud = \frac{847 + 1331}{847 + 537 + 484 + 1331} = 0.54$$

$$Precisión = \frac{847}{847 + 484} = 0.63$$

$$Sensibilidad = \frac{847}{847 + 537} = 0.61$$

$$F\ score = \frac{2 * 0.63 * 0.61}{0.63 + 0.61} = 0.62$$

Comparando estos valores con los obtenidos con el primer conjunto de datos *test*, vemos que se obtienen cambios significantes en los resultados obtenidos. Se obtienen peores valores en todos los indicadores de calidad, y en especial, en la exactitud, disminuyendo esta un 0.14. En este caso, la relación de aspecto 1:1 de las imágenes de entrada influye negativamente en los resultados finales.

Como podemos ver en la *Tabla 9*, para un *dropout* de 0.4, obtenemos una mejor exactitud que para un *dropout* de 0.1 y 0.2. En cuanto a los otros indicadores de calidad, son los mismos que para un *dropout* de 0.1.

Para esta configuración, al igual que para un *dropout* de 0.2, la relación de aspecto 1:1 no afecta mucho al comportamiento del modelo, ya que obtenemos valores prácticamente parecidos a los obtenidos con las imágenes originales.

Tabla 9. dropout 0.4 para 1\_CropCenter\_450x650

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	838	493
	Negativo	545	1331

$$Exactitud = \frac{838 + 1331}{838 + 545 + 493 + 1331} = 0.68$$

$$Precisión = \frac{838}{838 + 493} = 0.63$$

$$Sensibilidad = \frac{838}{838 + 545} = 0.61$$

$$F\ score = \frac{2 * 0.63 * 0.61}{0.63 + 0.61} = 0.62$$

En esta segunda etapa, hemos visto como la relación de aspecto ha influido de manera negativa cuando trabajamos con un *dropout* de 0.1. Sin embargo, esta influencia apenas se nota para un *dropout* de 0.4 y para una configuración inicial de 0.2.

A diferencia de los obtenidos con el primer conjunto de imágenes, los mejores resultados para el conjunto *testCropCenter* se obtienen para un *dropout* de 0.4.

### 1\_AugmentedCropCenter\_450x650

Para este entrenamiento se hará uso del conjunto de datos *testAugmentedCropCenter*. Como se ha mencionado anteriormente, este conjunto estará formado por 4 nuevas imágenes nuevas generadas a partir de la original: imagen original, simetría vertical, simetría horizontal y simetría vertical + horizontal. De esta manera, pasamos de tener 1331 imágenes de entrada a 5324 imágenes.

Para un *dropout* 0.1 conseguimos una mejora considerable en todos los indicadores de calidad, aumentando un 0.4 en exactitud y precisión y un 0.5 en sensibilidad y F score.

Tabla 10. *dropout* 0.1 para 1\_ *AugmentedCropCenter\_450x650*

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	3426	1898
	Negativo	2118	5324

$$Exactitud = \frac{3426 + 5324}{3426 + 2118 + 1898 + 5324} = 0.69$$

$$Precisión = \frac{3426}{3426 + 1898} = 0.64$$

$$Sensibilidad = \frac{3426}{3426 + 2118} = 0.62$$

$$F\ score = \frac{2 * 0.64 * 0.62}{0.64 + 0.62} = 0.63$$

Si comparamos estos resultados con los obtenidos para el conjunto de imágenes anterior, vemos como el aumento de imágenes debido a las técnicas de *data\_augmentation* influyen de manera positiva en el rendimiento de la red, aumentando en un 0.15 el valor de la exactitud.

Sin embargo, al igual que ocurre con un *dropout* de 0.2, para un *dropout* de 0.4 se consiguen los mismos valores que con el conjunto de entrenamiento anterior.

En la *Tabla 11*, podemos ver los resultados obtenidos para los 4 indicadores de calidad para un *dropout* de 0.4.

Tabla 11. dropout 0.4 para 1\_AugmentedCropCenter\_450x650

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	3348	1976
	Negativo	2179	5324

$$Exactitud = \frac{3348 + 5324}{3348 + 1976 + 2179 + 5324} = 0.68$$

$$Precisión = \frac{3348}{3348 + 1976} = 0.63$$

$$Sensibilidad = \frac{3348}{3348 + 2179} = 0.61$$

$$F\ score = \frac{2 * 0.63 * 0.61}{0.63 + 0.61} = 0.62$$

Como conclusión de esta tercera prueba, hemos visto que para este nuevo conjunto de imágenes hemos conseguido mejorar el rendimiento del modelo para un *dropout* de 0.1, convirtiéndolo en el modelo con mejores resultados hasta el momento. No podemos decir lo mismo para los otros dos modelos, en los cuales, los resultados siguen siendo los mismos que para el conjunto de datos anterior.

### 1\_AugmentedCropCenterRotatedAll\_450x650

Para el último entrenamiento usaremos las imágenes contenidas en el directorio *testAugmentedCropCenterRotatedAll*. Cabe recordar que, para la generación de este conjunto se ha hecho uso del directorio *testAugmentedCropCenter*.

En esta ocasión para cada una de las imágenes contenidas en este directorio se van a generar dos nuevas: imagen original e imagen rotada 90°. Por tanto, el nuevo conjunto de datos estará formado por 10648 imágenes,

Para un *dropout* de 0.1 los resultados siguen siendo considerablemente mejores que para la configuración inicial. En la *Tabla 12* podemos ver los resultados de una manera mucho más detallada.

*Tabla 12. dropout 0.1 para 1\_ AugmentedCropCenterRotatedAll\_450x650*

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	6872	3776
	Negativo	4202	10648

$$Exactitud = \frac{6819 + 10648}{6819 + 3829 + 4266 + 10648} = 0.68$$

$$Precisión = \frac{6819}{6819 + 3829} = 0.64$$

$$Sensibilidad = \frac{6819}{6819 + 4266} = 0.61$$

$$F\ score = \frac{2 * 0.64 * 0.61}{0.64 + 0.61} = 0.62$$

Finamente, para el *dropout* de 0.4 obtenemos mejores métricas que para un *dropout* de 0.2 pero peores que para un *dropout* de 0.1. En la *Tabla 13* podemos ver dichos valores de una manera mucho más detallada.

Tabla 13. dropout 0.4 para 1\_ AugmentedCropCenterRotatedAll\_450x650

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	6730	3918
	Negativo	4332	10648

$$Exactitud = \frac{6730 + 10648}{6730 + 3918 + 4332 + 10648} = 0.68$$

$$Precisión = \frac{6730}{6730 + 3918} = 0.63$$

$$Sensibilidad = \frac{6730}{6730 + 4332} = 0.61$$

$$F\ score = \frac{2 * 0.63 * 0.61}{0.63 + 0.61} = 0.62$$

Teniendo en cuenta todos los resultados obtenidos para cada grupo de imágenes de entrada y para cada valor de *dropout*, podemos concluir que la configuración con la que se han obtenido las mejores métricas y que, por tanto, será la configuración que se va a utilizar en la fase siguiente del proyecto es, el modelo con el valor de *dropout* de 0.1. En la *Tabla 14* podemos ver un resumen de los resultados obtenidos.

Tabla 14. Resultados análisis distintas configuraciones

		Exactitud	Precisión	Sensibilidad	F score
Dropout 0.1	test	0,68	0,64	0,62	0,63
	testCropCenter	0,54	0,63	0,61	0,62
	TestAugmentedCropCenter	0,69	0,64	0,62	0,63
	TestAugemntesCropcenterRotatedAll	0,68	0,64	0,61	0,62
	test	0,65	0,6	0,57	0,58

Conf. Inicial Dropout 0.2	testCropCenter	0,65	0,6	0,57	0,58
	TestAugmentedCropCenter	0,65	0,6	0,57	0,58
	TestAugemntesCropcenterRotatedAll	0,65	0,59	0,57	0,58
Dropout 0.4	test	0,68	0,61	0,63	0,62
	testCropCenter	0,68	0,63	0,61	0,62
	TestAugmentedCropCenter	0,68	0,63	0,61	0,62
	TestAugemntesCropcenterRotatedAll	0,68	0,63	0,61	0,62

### 5.3 Resultados de la implementación de la nueva funcionalidad

En esta segunda parte se recogen los resultados obtenidos tras haber realizado las pruebas anteriores pero esta vez para el modelo resultante de la nueva funcionalidad añadida mencionada en el apartado 4.3 *Nueva funcionalidad a desarrollar*.

#### 2\_test\_450x650

Para este nuevo modelo podemos ver que, haciendo uso de las imágenes originales como datos de entrada obtenemos una mejora muy considerable con respecto al anterior modelo, incrementado en un 0.2 todos los indicadores de calidad excepto la F score, donde obtenemos una reducción.

En la *Tabla 15* vemos de forma más detallada los valores obtenidos para los indicadores de calidad.

Tabla 15. dropout 0.1 para 2\_test\_450x650

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	871	460
	Negativo	506	1331

$$Exactitud = \frac{871 + 1331}{871 + 506 + 460 + 1331} = 0.70$$

$$Precisión = \frac{871}{871 + 460} = 0.65$$

$$Sensibilidad = \frac{871}{871 + 506} = 0.63$$

$$F\ score = \frac{2 * 0.65 * 0.63}{0.65 + 0.63} = 0.64$$

## 2\_CropCenter\_450x650

En la *Tabla 16* se muestran los resultados obtenidos cuando aplicamos la relación de aspecto 1:1. Como podemos ver en los cálculos de los indicadores de calidad, obtenemos una disminución de un 0.1 en la exactitud. Con respecto a los demás indicadores, los valores siguen siendo los mismos.

*Tabla 16. dropout 0.1 para 2\_CropCenter\_450x650*

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	870	461
	Negativo	511	1331

$$Exactitud = \frac{870 + 1331}{870 + 511 + 461 + 1331} = 0.69$$

$$Precisión = \frac{870}{870 + 461} = 0.65$$

$$Sensibilidad = \frac{870}{870 + 511} = 0.63$$

$$F\ score = \frac{2 * 0.65 * 0.63}{0.65 + 0.63} = 0.64$$

## 2\_ AugmentedCropCenter\_450x650

Cuando para este modelo pasamos de un conjunto de 1331 imágenes de entrada a un conjunto de 5324, como podemos ver en las métricas de calidad, no presentamos ninguna variación con respecto al entrenamiento anterior.

Tabla 17. dropout 0.1 para 2\_ AugmentedCropCenter\_450x650

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	3429	1895
	Negativo	2106	5324

$$Exactitud = \frac{3429 + 5324}{3429 + 2106 + 1895 + 5324} = 0.69$$

$$Precisión = \frac{3429}{3429 + 1895} = 0.64$$

$$Sensibilidad = \frac{3429}{3429 + 2106} = 0.62$$

$$F\ score = \frac{2 * 0.64 * 0.62}{0.64 + 0.62} = 0.63$$

## 2\_ AugmentedCropCenterRotatedAll\_450x650

Finalmente, cuando hacemos uso del conjunto de imágenes de entradas más grande, vemos que a partir de los resultados reflejados en la *Tabla 18*, hemos conseguido mejorar en un 0.1 el valor de la precisión. Con respecto a las otras métricas, no se ven alteradas.

*Tabla 18. dropout 0.1 para 2\_ AugmentedCropCenterRotatedAll\_450x650*

		Diagnóstico real	
		Positivo	Negativo
Diagnóstico modelo	Positivo	6872	3776
	Negativo	4202	10648

$$Exactitud = \frac{6872 + 10648}{6872 + 3776 + 4202 + 10648} = 0.69$$

$$Precisión = \frac{6872}{6872 + 3776} = 0.65$$

$$Sensibilidad = \frac{6872}{6872 + 4202} = 0.62$$

$$F\ score = \frac{2 * 0.65 * 0.62}{0.65 + 0.62} = 0.63$$

Si comparamos estos valores con los obtenidos para el modelo anterior desarrollado por Mario Álvaro Díaz, vemos como hemos conseguido mejorar considerablemente los valores de los indicadores de calidad, aumentando en un 0.4 el valor de la exactitud, un 0.6 en términos de precisión y un 0.5 más tanto para la sensibilidad como para la F score. De esta manera, se ha conseguido el objetivo de esta última fase. En la *Tabla 19* encontramos información más detallada acerca de los valores obtenidos para los indicadores de calidad para cada uno de los conjuntos de imágenes de entrada.

Tabla 19. Comparación de los resultados red neuronal inicial vs red neuronal final

		Exactitud	Precision	Sensibilidad	F score
<b>Resultados del modelo inicial</b>	Resultados inicales	0,65	0,6	0,57	0,58
	Resultados finales	0,65	0,6	0,57	0,58
	TestAugmentedCropCenter	0,65	0,6	0,57	0,58
	TestAugemntesCropcenterRotatedAll	0,65	0,59	0,57	0,58
<b>Resultados del modelo final</b>	test	0,7	0,65	0,63	0,64
	testCropCenter	0,69	0,65	0,63	0,64
	TestAugmentedCropCenter	0,69	0,64	0,62	0,63
	TestAugemntesCropcenterRotatedAll	0,69	0,65	0,62	0,63

## 6. Presupuesto

La realización de este proyecto se ha llevado a cabo gracias a la utilización de una serie de recursos físicos y licencias de software que han supuesto un determinado coste económico. Este apartado tiene como objetivo calcular la inversión necesaria para poder llevar a cabo el desarrollo de este proyecto. El presupuesto planteado está desglosado principalmente en: el coste de personal, las licencias software y otros recursos correspondientes.

El tiempo estimado para la elaboración de este proyecto es de 350 horas aproximadamente. Para su realización es necesario la contratación de un ingeniero de telecomunicaciones. Para no realizar un gran desembolso salarial, se contratará a un ingeniero junior. Según los datos recogidos por *infoempleo*, en España un ingeniero junior especialista en *cloud computing* tiene un sueldo medio de 34.000 euros brutos anuales. Esto equivale a un salario mensual medio de 2.148,2 €, esto es, 13,43 €/hora. De esta manera, en 350 horas será necesario realizar un desembolso total de 4700 €.

Por otro lado, en lo relativo a los diferentes programas informáticos que se van a utilizar encontramos:

- Cuenta con una suscripción a *Microsoft Azure* con un crédito de 400 €.
- Licencia *Microsoft Excel*, para el análisis de los resultados obtenidos.

Todo lo mencionado anteriormente no sería posible si no se contara con un ordenador. Cabe recordar que este proyecto supone una elevada carga computacional. Sin embargo, dado que este proyecto se va a llevar a cabo utilizando herramientas basadas en la nube, no necesitaremos un ordenador que disponga de muy altas prestaciones. Se recomiendan las siguientes características:

- Dispositivo: *Slimbook* con *Linux*
- 16 GB de RAM

Para satisfacer estas especificaciones encontramos numerosas soluciones en el mercado, desde ordenadores portátiles hasta ordenadores de sobremesa, con precios entre 549,00 € hasta 1899,00 €. Así pues, para el desarrollo de este proyecto seleccionaremos la mejor oferta que podemos encontrar y de esta manera contar con un dispositivo que garantice el cumplimiento de los requisitos necesarios.

Teniendo todos los aspectos en cuenta, el coste total estimado para la realización del proyecto es de 7.034 €. No obstante, hay que tener en cuenta que los recursos de personal y recursos físicos incluyen valores variables, por lo que este presupuesto total puede variar en función de estos factores.



## 7. Conclusiones y líneas futuro de trabajo

Durante estos últimos años, la nube ha sido el lugar donde la Inteligencia Artificial ha tenido su desarrollo exponencial gracias al enorme potencial de cálculo que tiene lugar en los *data centers*. Eso nos permite llevar a cabo diferentes tareas computacionalmente muy exigentes a un coste muy reducido.

Hemos visto como servicios basados en la nube como *Microsoft Azure* o *Google Colab* nos han permitido reducir considerablemente los tiempos de ejecución a la hora de desarrollar y entrenar una red neuronal convolucional.

Además, hemos podido analizar el comportamiento de esta red ante variaciones en los diferentes hiperparámetros de configuración con el objetivo de sacar conclusiones que nos permitan entender mejor su funcionamiento. Finalmente, hemos añadido una técnica de *data augmentation* con el objetivo de poder mejorar su funcionamiento.

La investigación y el desarrollo que se han llevado a cabo a lo largo de la realización de este proyecto abren numerosas líneas de trabajo futuro. El análisis del comportamiento de la red neuronal en otros servicios basados en la nube como *AWS* o *Google Cloud* se ha visto condicionado por el tiempo y presupuesto disponible.

Por otro lado, técnicas de *data augmentation* ofrecen muchos beneficios en tareas donde resulte complicado obtener nuevos conjuntos de datos. Todo ello junto con una gran cantidad de pruebas para distintas configuraciones de los hiperparámetros nos permitiría optimizar los resultados obtenidos.







## 8. Bibliografía

- [1]"Computación en la nube, ¿sabes quién lo inventó y cómo surgió?", *Sky.One*, 2021. [Online]. Available: <https://skyone.solutions/es/hub/conocer-la-computacion-en-la-nube/>. [Accessed: 03- May- 2021].
- [2]"El origen de: El Cómputo en la Nube", *Publimetro*, 2021. [Online]. Available: <https://www.fayerwayer.com/2012/01/el-origen-de-el-computo-en-la-nube/>. [Accessed: 03- May- 2021].
- [3]C. Wagner, "9 Datos y estadísticas de computación en la nube interesantes (2021)", *Hostingpill*, 2021. [Online]. Available: <https://hostingpill.com/es/statistics/cloud-computing-statistics/>. [Accessed: 03- May- 2021].
- [4]R. PowerData, "¿Cómo la inteligencia artificial mejora Cloud Computing en tu empresa?", *Blog.powerdata.es*, 2021. [Online]. Available: <https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/como-la-inteligencia-artificial-mejora-cloud-computing-en-tu-empresa>. [Accessed: 03- May- 2021].
- [5]"IOMED - Retos de la inteligencia artificial en medicina", *lomed.es*, 2021. [Online]. Available: <https://iomed.es/2020/01/08/retos-de-la-inteligencia-artificial-en-medicina/>. [Accessed: 03- May- 2021].
- [6]"LA INTELIGENCIA ARTIFICIAL EN LA NUBE, EL NUEVO MODELO DE NEGOCIO DE LAS GIGANTES TECNOLOGÍAS | EI EFETE", *Elfete.com*, 2021. [Online]. Available: <http://www.elfete.com/la-inteligencia-artificial-en-la-nube-el-nuevo-modelo-de-negocio-de-las-gigantes-tecnologicas/>. [Accessed: 03- May- 2021].
- [7]"La Sociedad Americana Contra El Cancer", *Cancer.org*, 2021. [Online]. Available: <https://www.cancer.org/es/>. [Accessed: 03- May- 2021].
- [8]"Estadísticas importantes sobre el cáncer de piel tipo melanoma", *Cancer.org*, 2021. [Online]. Available: <https://www.cancer.org/es/cancer/cancer-de-piel-tipo-melanoma/acerca/estadisticas-clave.html>. [Accessed: 03- May- 2021].
- [9]"Eucerin: Acerca de la piel | Estructura y función de la piel: Una introducción.", *Eucerin.es*, 2021. [Online]. Available: <https://www.eucerin.es/acerca-de-la-piel/conocimientos-basicos-sobre-la-piel/estructura-y-funcion-de-la-piel#:~:text=La%20piel%2C%20%C3%B3rgano%20din%C3%A1mico%20constantemente,est%C3%A1%20formada%20por%20varias%20subcapas>. [Accessed: 03- May- 2021].
- [10]"Componentes de la piel - Videos de salud: MedlinePlus enciclopedia médica", *Medlineplus.gov*, 2021. [Online]. Available: <https://medlineplus.gov/spanish/ency/anatomyvideos/000029.htm#:~:text=L>

- a%20capa%20m%C3%A1s%20profunda%20de, llamada%20la%20%22piel%20verdadera%22. [Accessed: 03- May- 2021].
- [11]"Melanoma cutáneo maligno - Publicaciones Científicas", *Enfermeriadeciudadreal.com*, 2021. [Online]. Available: [https://www.enfermeriadeciudadreal.com/articulo\\_imprimir.asp?idarticulo=631&accion=](https://www.enfermeriadeciudadreal.com/articulo_imprimir.asp?idarticulo=631&accion=). [Accessed: 03- May- 2021].
- [12]E. Press, "¿Qué es el melanoma? | Estadios de la enfermedad | Infosalus", *Infosalus.com*, 2021. [Online]. Available: <https://www.infosalus.com/enfermedades/oncologia/melanoma/que-es-melanoma-13.html>. [Accessed: 03- May- 2021].
- [13]"Melanoma ocular - Síntomas y causas - Mayo Clinic", *Mayoclinic.org*, 2021. [Online]. Available: <https://www.mayoclinic.org/es-es/diseases-conditions/eye-melanoma/symptoms-causes/syc-20372371>. [Accessed: 03- May- 2021].
- [14]E. Press, "¿Qué es el melanoma? | Estadios de la enfermedad | Infosalus", *Infosalus.com*, 2021. [Online]. Available: <https://www.infosalus.com/enfermedades/oncologia/melanoma/que-es-melanoma-13.html>. [Accessed: 03- May- 2021].
- [15]"Factores de riesgo para el cáncer de piel tipo melanoma", *Cancer.org*, 2021. [Online]. Available: <https://www.cancer.org/es/cancer/cancer-de-piel-tipo-melanoma/causas-riesgos-prevencion/factores-de-riesgo.html#:~:text=Antecedente%20familiar%20de%20melanoma,antecedentes%20familiares%20de%20esta%20enfermedad>. [Accessed: 03- May- 2021].
- [16]*Incibe.es*, 2021. [Online]. Available: [https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia-cloud-computing\\_0.pdf](https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia-cloud-computing_0.pdf). [Accessed: 03- May- 2021].
- [17]"¿Qué es y para qué sirve el Cloud Computing? - OpenSistemas", *OpenSistemas*, 2021. [Online]. Available: <https://opensistemas.com/que-es-y-para-que-sirve-el-cloud-computing/>. [Accessed: 03- May- 2021].
- [18]"Cloud Computing: Que és, características y ventajas", *OpenWebinars.net*, 2021. [Online]. Available: <https://openwebinars.net/blog/cloud-computing-que-es-caracteristicas-y-ventajas/>. [Accessed: 03- May- 2021].
- [19]"¿Qué es Cloud Computing?", *Salesforce.com*, 2021. [Online]. Available: <https://www.salesforce.com/mx/cloud-computing/>. [Accessed: 03- May- 2021].
- [20]"Ventajas del cloud computing", *Intelequia*, 2021. [Online]. Available: <https://intelequia.com/blog/post/2055/ventajas-del-cloud-computing>. [Accessed: 03- May- 2021].
- [21]"Reducción de costes con Cloud Computing", *Descom.es*, 2021. [Online]. Available: <https://www.descom.es/blog/cloud/reduccion-de-costes-con-cloud-computing.html>. [Accessed: 03- May- 2021].

- [22]"Tipos de cloud computing, ¡conócelos todos! - Einatec", *Einatec*, 2021. [Online]. Available: <https://einatec.com/tipos-de-cloud-computing/>. [Accessed: 03- May- 2021].
- [23]"Diferencias entre una nube pública, una nube privada y una nube híbrida | Microsoft Azure", *Azure.microsoft.com*, 2021. [Online]. Available: <https://azure.microsoft.com/es-es/overview/what-are-private-public-hybrid-clouds/#deployment-options>. [Accessed: 03- May- 2021].
- [24]"Los 3 tipos de servicios que existen dentro del cloud computing en las empresas • Blog de Viafirma", *Viafirma.com*, 2021. [Online]. Available: <https://www.viafirma.com/blog-xnoccio/es/tipos-servicios-cloud-computing-empresas/>. [Accessed: 03- May- 2021].
- [25]"¿Cómo aprenden las máquinas? Machine Learning y sus diferentes tipos | datos.gob.es", *Datos.gob.es*, 2021. [Online]. Available: <https://datos.gob.es/es/blog/como-aprenden-las-maquinas-machine-learning-y-sus-diferentes-tipos>. [Accessed: 15- May- 2021].
- [26]"¿Qué es el aprendizaje automático (ML) y por qué es importante? | NetApp", *Netapp.com*, 2021. [Online]. Available: <https://www.netapp.com/es/artificial-intelligence/what-is-machine-learning/>. [Accessed: 15- May- 2021].
- [27]"Redes Neuronales artificiales: Qué son y cómo se entrenan", *Xeridia*, 2021. [Online]. Available: <https://www.xeridia.com/blog/redes-neuronales-artificiales-que-son-y-como-se-entrenan-parte-i>. [Accessed: 15- May- 2021].
- [28]"¿Cuáles son las partes de una neurona? | LaReserva.com", *Lareserva.com*, 2021. [Online]. Available: [https://www.lareserva.com/Cuales\\_son\\_las\\_partes\\_de\\_una\\_neurona](https://www.lareserva.com/Cuales_son_las_partes_de_una_neurona). [Accessed: 15- May- 2021].
- [29]"Redes Neuronales artificiales: Qué son y cómo se entrenan", *Xeridia*, 2021. [Online]. Available: <https://www.xeridia.com/blog/redes-neuronales-artificiales-que-son-y-como-se-entrenan-parte-i>. [Accessed: 15- May- 2021].
- [30]"Redes neuronales", *Medium*, 2021. [Online]. Available: <https://bootcampai.medium.com/redes-neuronales-13349dd1a5bb#:~:text=Funciones%20de%20activaci%C3%B3n,entrada%20o%20conjunto%20de%20entradas.&text=Adem%C3%A1s%2C%20se%20debe%20considerar%20que,la%20mayor%20cantidad%20de%20datos>. [Accessed: 15- May- 2021].
- [31] B. Moons, R. Uytterhoeven, W. Dehaene y M. Verhelst, "14.5 Envision: un procesador de red neuronal convolucional escalable de precisión de voltaje de 0,26 a 10TOPS / W paralelo paralelo con precisión de frecuencia dinámica escalable en FDSOI de 28nm", 2017 IEEE International 34 Solid-State Circuits Conference (ISSCC), San Francisco, CA, 2017, pp. 246247, doi: 10.1109 / ISSCC.2017.7870353.

- [32] Detección automática de especies bentónicas en fondos profundos mediante imágenes obtenidas con vehículos remotos y técnicas de deep-learning, de Celia Díaz Cuesta, 2018, Universidad de Cantabria
- [33]S. Programacion en Castellano, "Qué es Python y para qué sirve?", *Programación en Castellano.*, 2021. [Online]. Available: [https://programacion.net/articulo/que\\_es\\_python\\_y\\_para\\_que\\_sirve\\_2795](https://programacion.net/articulo/que_es_python_y_para_que_sirve_2795). [Accessed: 15- May- 2021].
- [34]"Welcome to Python.org", *Python.org*, 2021. [Online]. Available: <https://www.python.org/>. [Accessed: 15- May- 2021].
- [35]"os — Miscellaneous operating system interfaces — Python 3.9.5 documentation", *Docs.python.org*, 2021. [Online]. Available: <https://docs.python.org/3/library/os.html>. [Accessed: 15- May- 2021].
- [36]"csv — Lectura y escritura de archivos CSV — documentación de Python - 3.10.0b1", *Docs.python.org*, 2021. [Online]. Available: <https://docs.python.org/es/3.10/library/csv.html>. [Accessed: 15- May- 2021].
- [37]"random — Generate pseudo-random numbers — Python 3.9.5 documentation", *Docs.python.org*, 2021. [Online]. Available: <https://docs.python.org/3/library/random.html>. [Accessed: 15- May- 2021].
- [38]"Pillow", *PyPI*, 2021. [Online]. Available: <https://pypi.org/project/Pillow/>. [Accessed: 15- May- 2021].
- [39]"Matplotlib: Python plotting — Matplotlib 3.4.2 documentation", *Matplotlib.org*, 2021. [Online]. Available: <https://matplotlib.org/>. [Accessed: 15- May- 2021].
- [40]"Introduction to NumPy", *W3schools.com*, 2021. [Online]. Available: [https://www.w3schools.com/python/numpy/numpy\\_intro.asp](https://www.w3schools.com/python/numpy/numpy_intro.asp). [Accessed: 15- May- 2021].
- [41]"time — Time access and conversions — Python 3.9.5 documentation", *Docs.python.org*, 2021. [Online]. Available: <https://docs.python.org/3/library/time.html>. [Accessed: 15- May- 2021].
- [42]"TensorFlow", *TensorFlow*, 2021. [Online]. Available: <https://www.tensorflow.org/?hl=es-419>. [Accessed: 15- May- 2021].
- [43]O. Marketing, M. búsqueda and ¿. Keras?, "Keras: biblioteca de código abierto para crear redes neuronales", *IONOS Digitalguide*, 2021. [Online]. Available: <https://www.ionos.es/digitalguide/online-marketing/marketing-para-motores-de-busqueda/que-es-keras/>. [Accessed: 15- May- 2021].
- [44]"Colaboratory – Google", *Research.google.com*, 2021. [Online]. Available: <https://research.google.com/colaboratory/faq.html#resource-limits>. [Accessed: 15- May- 2021].

[45]"¿Qué es Azure? - Learn", *Docs.microsoft.com*, 2021. [Online]. Available: <https://docs.microsoft.com/es-es/learn/modules/intro-to-azure-fundamentals/what-is-microsoft-azure>. [Accessed: 15- May- 2021].

[46]*Oa.upm.es*, 2021. [Online]. Available: [http://oa.upm.es/66778/7/TFG\\_MARIO\\_ALVARO\\_DIAZ.pdf](http://oa.upm.es/66778/7/TFG_MARIO_ALVARO_DIAZ.pdf). [Accessed: 20-Jun- 2021].

