



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Software

Detección de carcinoma basocelular utilizando red neuronal convolucional y Support Vector Machine

TESIS

Para optar el Título Profesional de Ingeniero de Software

AUTOR

Jorge Alexander ANGELES ROJAS

ASESOR

Hugo David CALDERÓN VILCA

Lima, Perú

2022



Reconocimiento - No Comercial - Compartir Igual - Sin restricciones adicionales

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Usted puede distribuir, remezclar, retocar, y crear a partir del documento original de modo no comercial, siempre y cuando se dé crédito al autor del documento y se licencien las nuevas creaciones bajo las mismas condiciones. No se permite aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros a hacer cualquier cosa que permita esta licencia.

Referencia bibliográfica

Angeles, J. (2022). *Detección de carcinoma basocelular utilizando red neuronal convolucional y Support Vector Machine*. [Tesis de pregrado, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e Informática, Escuela Profesional de Ingeniería de Software]. Repositorio institucional Cybertesis UNMSM.

Metadatos complementarios

Datos de autor	
Nombres y apellidos	Jorge Alexander Angeles Rojas
Tipo de documento de identidad	DNI
Número de documento de identidad	71395600
URL de ORCID	https://orcid.org/0000-0002-0925-5320
Datos de asesor	
Nombres y apellidos	Hugo David Calderón Vilca
Tipo de documento de identidad	DNI
Número de documento de identidad	01317748
URL de ORCID	https://orcid.org/0000-0002-1177-4947
Datos del jurado	
Presidente del jurado	
Nombres y apellidos	Nora Bertha La Serna Palomino
Tipo de documento	DNI
Número de documento de identidad	07665297
Miembro del jurado 1	
Nombres y apellidos	Luzmila Elisa Pró Concepción
Tipo de documento	DNI
Número de documento de identidad	08862360
Datos de investigación	
Línea de investigación	C.0.3.16. Procesamiento de Imágenes y Visión por computadora

Grupo de investigación	No aplica
Agencia de financiamiento	Sin financiamiento
Ubicación geográfica de la investigación	<p>Universidad Nacional Mayor de San Marcos</p> <p>Edificio: Facultad de Ingeniería de Sistemas e Informática</p> <p>País: Perú</p> <p>Departamento: Lima</p> <p>Provincia: Lima</p> <p>Distrito: Cercado de Lima</p> <p>Calle: Av. Carlos Germán Amezaga 375</p> <p>Latitud: -12.053697</p> <p>Longitud: -77.085774</p>
Año o rango de años en que se realizó la investigación	Septiembre 2019 – Marzo 2021
URL de disciplinas OCDE	<p>Sistemas de automatización, Sistemas de Control</p> <p>https://purl.org/pe-repo/ocde/ford#2.02.03</p> <p>Ingeniería de sistemas y comunicaciones</p> <p>https://purl.org/pe-repo/ocde/ford#2.02.04</p>



Universidad Nacional Mayor de San Marcos

Universidad del Perú, DECANA DE AMÉRICA

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Software

Acta de Sustentación Virtual de Tesis

Siendo las quince (15) horas del día 08 (ocho) del mes de julio de 2022, se reunieron en la sala virtual meet.google.com/tqc-ggeo-qkb, presidido por la Dra. Nora Bertha La Serna Palomino, Dra. Luzmila Elisa Pró Concepción (Miembro) y el Dr. Hugo David Calderón Vilca (Miembro Asesor), para la sustentación virtual de la Tesis intitulada **“DETECCIÓN DE CARCINOMA BASOCELULAR UTILIZANDO RED NEURONAL CONVOLUCIONAL Y SUPPORT VECTOR MACHINE”**, por el Bachiller **Jorge Alexander Angeles Rojas**, para optar el Título Profesional de Ingeniero de Software.

Acto seguido de la exposición de la Tesis, la Presidenta invitó al bachiller a dar respuesta a las preguntas establecidas por los Miembros de Jurado.

El bachiller en el curso de sus intervenciones demostró pleno dominio del tema, al responder con acierto y fluidez a las observaciones y preguntas formuladas por los señores miembros del Jurado.

Finalmente habiéndose efectuado la calificación correspondiente por los miembros de Jurado, el bachiller obtuvo la nota de 18 (dieciocho).

A continuación, la Presidenta del Jurado, Dra. Nora Bertha La Serna Palomino, declara al bachiller **Ingeniero de Software**.

Siendo las 15:55 horas, se levantó la sesión.

Dra. Nora Bertha La Serna Palomino
Presidenta

Dra. Luzmila Elisa Pró Concepción
Miembro

Dr. Hugo David Calderón Vilca
Miembro Asesor

DEDICATORIA

Este trabajo está dedicado a mi familia y a mi
buen amigo Jimmy que en paz descanse.

AGRADECIMIENTOS

Agradezco al profesor Calderón por todo su apoyo y orientación para desarrollar la presente tesis y a mis padres por apoyarme en todo lo que necesitaba para desarrollar mi carrera profesional.

RESUMEN

Título	Detección del carcinoma basocelular basado en el modelo híbrido de red neuronal convolucional y Support vector machine
Asesor de tesis	Hugo David Calderón Vilca
Fecha	

El cáncer de piel es uno de los tipos de cáncer más frecuente en los seres humanos, abarca cerca de un tercio total de las neoplasias. Dentro del cáncer de piel encontramos al carcinoma basocelular (CBC) siendo este el tipo de cáncer más frecuente a nivel mundial. Una serie de estudios que involucran enfoques de aprendizaje profundo ya se han desempeñado en un número considerable como la clasificación de imágenes. Los modelos utilizados en dichas tareas emplean la función Softmax (modelo clásico) en la capa de clasificación. Sin embargo, se han realizado estudios que utilizan una alternativa a la función Softmax para la clasificación: la máquina de vectores de soporte (SVM). El uso de SVM en una arquitectura de red neuronal artificial produce resultados relativamente mejores que el uso de la función Softmax convencional. Por este motivo se construyó un sistema que diagnostica el carcinoma basocelular implementando un modelo híbrido de red neuronal convolucional y máquina de vectores de soporte para clasificar el CBC. Los resultados obtenidos fueron medidos con las métricas de precisión, recall, f1-score y exactitud obteniendo 94.51%, 88.42%, 91.36% y 91.54% respectivamente.

Palabras claves:

Carcinoma basocelular, cáncer de piel, Redes neuronales convolucionales, Support Vector Machine, Diagnóstico

SUMMARY

Título	Detection of basal cell carcinoma based on hybrid model of convolutional neural network and Support Vector Machine
Asesor de tesis	Hugo David Calderón Vilca
Fecha	

Skin cancer is one of the most frequent types of cancer in humans, comprising about one third of all neoplasms. Within skin cancer, basal cell carcinoma (BCC) is the most frequent type of cancer worldwide. A number of studies involving deep learning approaches have already been performed in a considerable number such as image classification. The models used in such tasks employ the Softmax function (classical model) in the classification layer. However, studies have been conducted that analyze an alternative to the Softmax function for classification: the support vector machine (SVM). The use of SVM in an artificial neural network architecture produces relatively better results than the use of the conventional Softmax function. For this reason, a system that diagnoses basal cell carcinoma was built by implementing a hybrid model of convolutional neural network and support vector machine to classify BCC. The results obtained were measured with the metrics of precision, recall, f1-score and accuracy obtaining 94.51%, 88.42%, 91.36% and 91.54% respectively.

Keywords:

Basal Cell Carcinoma, Skin Cancer, Convolutional Neural Networks, Support Vector Machine, Diagnostics

ÍNDICE

ÍNDICE DE GRÁFICOS	9
ÍNDICE DE TABLAS	10
ÍNDICE DE FIGURAS	11
CAPÍTULO 1: INTRODUCCIÓN	13
1.1. Antecedentes	13
1.1.1. Carcinoma basocelular en el mundo.....	13
1.1.2. Carcinoma basocelular en Latinoamérica.....	16
1.1.3. Carcinoma basocelular en el Perú	17
1.2. Descripción del problema	19
1.2.1. Formulación del problema	20
1.3. Importancia	21
1.4. Motivación	22
1.5. Objetivo	22
1.6. Objetivos específicos	22
1.7. Propuesta.....	23
1.8. Estructura de la tesis	23
CAPÍTULO 2: MARCO TEÓRICO	24
2.1 Carcinoma basocelular.....	24
2.1.1. Epidemiología.....	24
2.1.2. Etiología y patogenia	25
2.1.3. Comportamiento biológico.....	26
2.1.4. Manifestaciones clínicas.....	26
2.1.5. Diagnóstico.....	26
2.2. Redes neuronales artificiales.....	27
2.3.1. Funciones de activación.....	30
2.3.4. Redes neuronales convolucionales.....	34
2.3.5. Entrenamiento de una red neuronal.....	36
2.3.5.1. Optimización	39
2.3.5.2. Sobreajuste (Overfitting).....	40
2.3.5.3. Aumento de datos (Data augmentation).....	40
2.3.5.4. Parámetros e hiperparámetros	41

2.3.6. Modelos de redes neuronales convolucionales	42
2.3.6.1. AlexNet	42
2.3.6.2. VGG16	43
2.3.6.3. Inception-V3	44
2.3.6.4. ResNet	45
2.3.7. Support Vector Machine	46
2.3.8. Métricas de evaluación	48
CAPÍTULO 3: ESTADO DEL ARTE	51
2.1. Arquitecturas para diagnosticar el cáncer de piel	55
2.2. Framework para diagnosticar el cáncer de piel.....	57
2.3. Métodos para diagnosticar el cáncer de piel.....	59
2.4. Modelos para diagnosticar el cáncer de piel.	69
2.5. Investigaciones que diagnosticaron el CBC	77
CAPÍTULO 4: METODOLOGÍA E IMPLEMENTACIÓN	78
4.1. Dataset	78
4.1.1. Estructura de los registros	79
4.2. Componentes	79
4.2.1 Red neuronal convolucional	79
4.2.2. Support Vector Machine.....	80
4.3. Modelo propuesto para clasificar el CBC.....	80
4.3.1. Pre-procesamiento de la imagen.....	81
4.3.2. Entrenamiento y clasificación con SVM.....	84
4.5. Metodología de desarrollo e implementación del sistema que detecta el carcinoma basocelular	86
4.5.1. Historias de usuario.....	86
4.5.2. Product Backlog	88
4.5.3. Sprints.....	88
4.5.4. Sistema	98
CAPÍTULO 5: RESULTADOS Y DISCUSIÓN	101
5.1. Resultados.....	101

5.1.1. Resultados y evaluación del modelo híbrido (CNN+SVM) para clasificar el CBC	102
5.2. Discusión	104
Conclusiones	107
Conclusiones Específicas	107
Limitaciones.....	108
Trabajos futuros	108
ANEXOS	117

ÍNDICE DE GRÁFICOS

Gráfico 1 Número estimados de nuevos casos de no melanoma en 2018 (Continentes) (Cancer Today, s. f.).....	14
Gráfico 2 Número estimados de nuevos casos de no melanoma en 2018 (Países) (Cancer Today, s. f.).....	15
Gráfico 3 Relación de incidencias y mortalidad del cáncer de tipo no melanoma entre los sexos masculino y femenino (Cancer Today, s. f.).....	16
Gráfico 4 Número estimados de nuevos casos de no melanoma en 2018 en Latinoamérica (Cancer Today, s. f.).....	17
Gráfico 5 Hiperplanos de separación de dos clases: (a) Un hiperplano de separación (b) Otros hiperplanos de separación (Suárez, 2016)	47
Gráfico 6 Margen de un hiperplano de separación: (a) Hiperplano de separación no óptimo (b) Hiperplano de separación óptimo (Suárez, 2016)	48
Gráfico 7 Total de imágenes de carcinoma basocelular. Elaboración propia.	83
Gráfico 8 Evaluación de la exactitud para el modelo híbrido.	102
Gráfico 9 Evaluación de la pérdida para el modelo híbrido	103

ÍNDICE DE TABLAS

Tabla 1 Relación de nuevos casos de cáncer de tipo no melanoma entre personas de sexo masculino y femenino.....	17
Tabla 2 Tipos de lesiones encontrados en las personas según los registros del CIDERM (Sordo & Gutiérrez, 2013).....	18
Tabla 3 Resumen del estado del arte	51
Tabla 4 Cuadro comparativo de HAM10000 con otros datasets. (Tschandl et al., 2018)	78
Tabla 5 Configuración de la red neuronal convolucional.....	85
Tabla 6 Capas totalmente conectadas	85
Tabla 7 Historia de usuario 01, entrenar el modelo que predice el CBC.	86
Tabla 8 Historia de usuario 02, subir una imagen dermatoscópica	87
Tabla 9 Product backlog	88
Tabla 10 Vista de algunas celdas del HAM10000_metadata.csv	92
Tabla 11 Resumen del plan de pruebas del Sprint 1.....	95
Tabla 12 Resumen del plan de pruebas del Sprint 2.....	97
Tabla 13 Clasificación de imágenes dermatoscópicas utilizando el sistema.....	100
Tabla 14 Distribución del conjunto de datos	101
Tabla 15 Matriz de confusión del modelo híbrido	103
Tabla 16 Métricas del modelo híbrido.....	104
Tabla 17 Comparativa de resultados con otras investigaciones que clasificaron el CBC	105

ÍNDICE DE FIGURAS

Figura 1 Características del CBC, las estructuras de izquierda a derecha son: arborización / vasos ramificados, estructuras en forma de rueda de radios, áreas como hojas, nidos ovoides gris azulados, múltiples puntos / glóbulos grises azules, etc. (Marghoob & Jaimes, 2019).....	21
Figura 2 Uso del dermatoscopio (Palacios-Martínez & Díaz-Alonso, 2017)	27
Figura 3 Comparación de una red neuronal biológica y una red neuronal artificial (Stanford University CS231n: Convolutional Neural Networks for Visual Recognition, s. f.).....	28
Figura 4 Red neuronal pre-alimentada simple (D. J. Wu, 2012).....	29
Figura 5 Función sigmoidea (<i>Stanford University CS231n: Convolutional Neural Networks for Visual Recognition</i> , s. f., p. 2).....	32
Figura 6 Función Tanh (Stanford University CS231n: Convolutional Neural Networks for Visual Recognition, s. f.)	32
Figura 7 Función ReLU (Stanford University CS231n: Convolutional Neural Networks for Visual Recognition, s. f.)	33
Figura 8 Agrupamiento promedio (Average pooling).....	36
Figura 9 Entrenamiento de la red neuronal (Torres, 2018)	38
Figura 10 Arquitectura AlexNet. (Krizhevsky et al., 2012)	43
Figura 11 Arquitectura VGG16. (Simonyan & Zisserman, 2015)	44
Figura 12 Módulo Inception. (Szegedy et al., 2014).....	44
Figura 13 Aprendizaje residual. (He et al., 2015)	45
Figura 14 Matriz de confusión. (Barrios, 2019)	49
Figura 15 Modelo para clasificar el CBC en alto nivel	81
Figura 16 Imagen dermatoscópica con rotación aleatoria. Elaboración propia.	82
Figura 17 Imagen dermatoscópica con zoom aleatorio. Elaboración propia.	82
Figura 18 Imagen dermatoscópica con volteo horizontal. Elaboración propia.	82
Figura 19 Imagen dermatoscópica con volteo vertical. Elaboración propia.	83
Figura 20 Diagrama de flujo del pre procesado de las imágenes, elaboración propia. ..	84
Figura 21 Prototipo del sistema. Elaboración propia.	89
Figura 22 Arquitectura en alto nivel. Elaboración propia.	89
Figura 23 Representación de interacción de componentes. Elaboración propia.	90
Figura 24 Diagrama de despliegue. Elaboración propia.....	91
Figura 25 Comprobación de la dimensionalidad de los datos. Elaboración propia.	93

Figura 26 Implementación de la arquitectura propuesta. Elaboración propia	94
Figura 27 Captura de pantalla del sistema.....	98
Figura 28 Captura de pantalla de la selección de una imagen dermatoscópica.....	99
Figura 29 Captura de pantalla de la imagen cargada en el sistema	99
Figura 30 Captura de pantalla de la clasificación realizada	100

CAPÍTULO 1: INTRODUCCIÓN

1.1. Antecedentes

El cáncer de piel es uno de los tipos de cáncer más frecuente en los seres humanos, abarca cerca de un tercio total de las neoplasias. Se distinguen dos grandes grupos: el melanoma maligno y el cáncer cutáneo no melanoma. Este último incluye el carcinoma basocelular, considerado como el tipo más frecuente de cáncer. El carcinoma basocelular (CBC) es un tumor epitelial maligno que se origina a partir de las células pluripotenciales del epitelio. Se caracteriza por un crecimiento lento, pero es localmente invasivo; aunque posee un bajo potencial metastásico (reproducción del cáncer a otra parte del cuerpo), tiene capacidad destructora local y compromete extensas áreas de tejido, cartílago e incluso huesos, en las formas clínicas más severas (Darias Domínguez & Garrido Celis, 2018).

1.1.1. Carcinoma basocelular en el mundo

La Asociación Americana de Cáncer (*Estadísticas importantes sobre el cáncer de piel tipo melanoma*, s. f.; *¿Qué son los cánceres de piel de células basales y de células escamosas?*, s. f.) indica que el melanoma conforma solo 1% de los casos de cáncer de piel, por lo que podemos deducir que el 99% de los casos de cáncer de piel son de tipo no melanoma. El cáncer de piel no melanoma se forma en las células basales o las células escamosas, dependiendo del tipo de célula afectada se denomina carcinoma basocelular o carcinoma de células escamosas. Alrededor de ocho de cada diez casos son cánceres de células basales, mientras que el cáncer de células escamosas ocurre con menos frecuencia.

El gráfico 1 proporcionado por el Centro Internacional de Investigaciones sobre el Cáncer (*Cancer Today*, s. f.) corresponde al número estimados de nuevos casos de cáncer de tipo no melanoma en el año 2018, de ambos sexos a nivel continental.

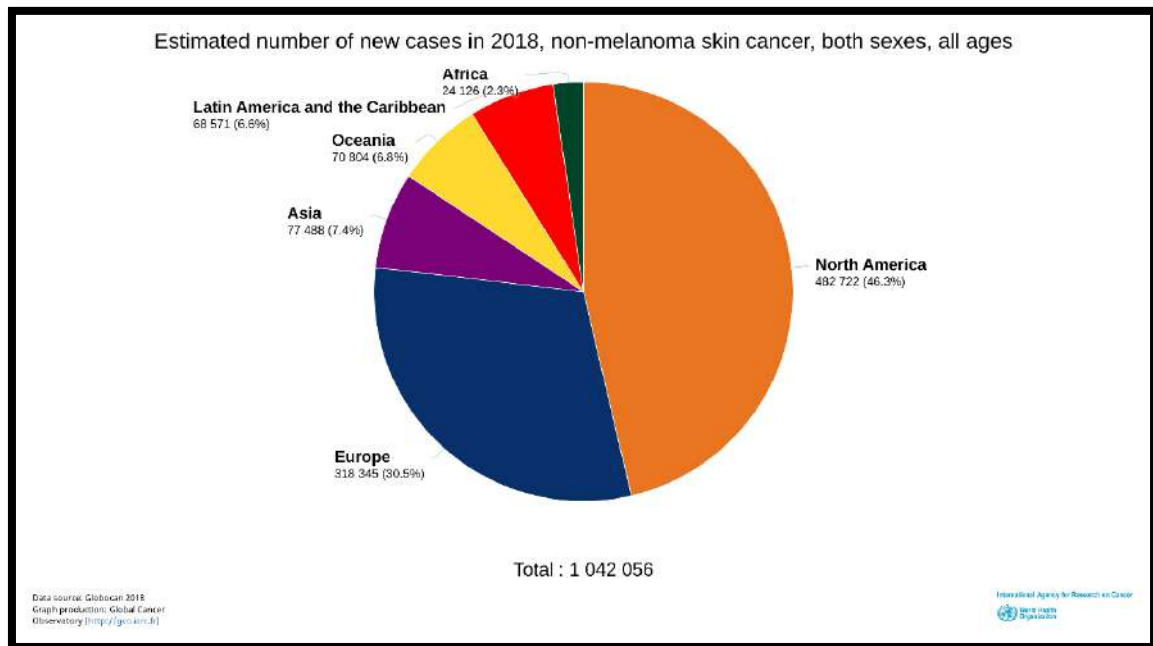


Gráfico 1 Número estimados de nuevos casos de no melanoma en 2018 (Continentes) (Cancer Today, s. f.)

Según el gráfico 1 hay un total de 1,042,056 personas con cáncer no melanoma, si usamos la proporción de casos de cáncer basocelular sostenida por (*¿Qué son los cánceres de piel de células basales y de células escamosas?*, s. f.) hay un aproximado de 833,644 de nuevos casos de carcinoma basocelular. Prosiguiendo con esta proporción para los demás continentes, los continentes que presentan la mayor cantidad de casos es Norte América y Europa con 386,117 y 254,676 respectivamente de nuevos casos de carcinoma basocelular. Latino América y el Caribe ocupa el quinto puesto con 54,856 casos aproximados de carcinoma basocelular.

El gráfico 2 corresponde al número estimado de nuevos casos de cáncer de tipo no melanoma en el año 2018, de ambos sexos a nivel de naciones.

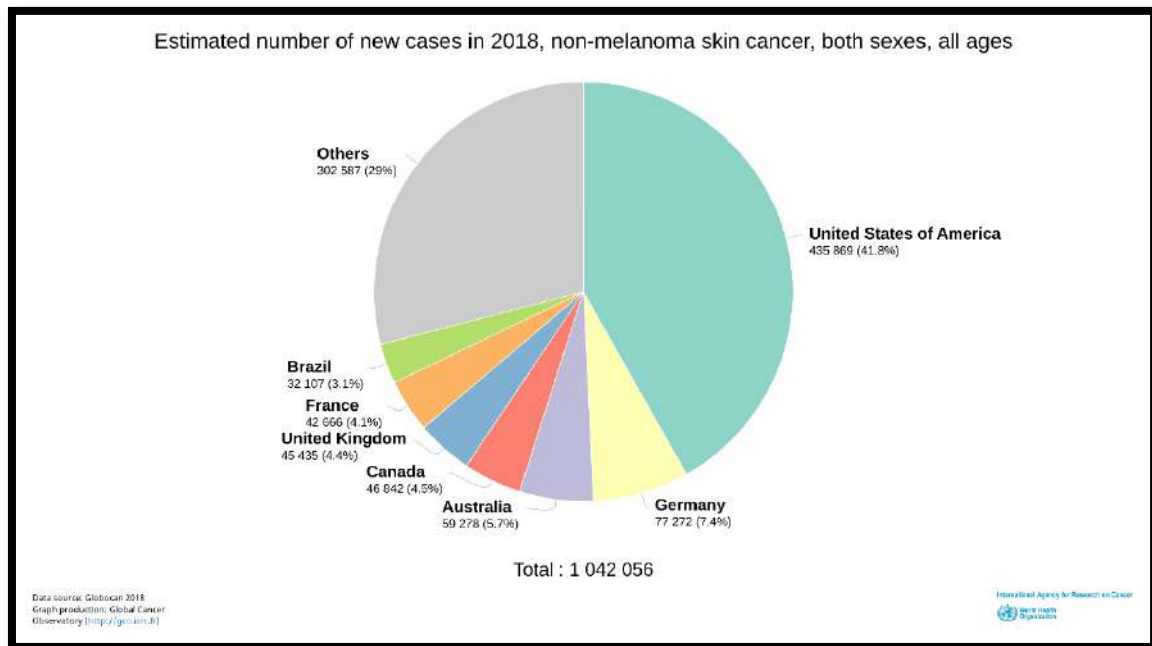


Gráfico 2 Número estimados de nuevos casos de no melanoma en 2018 (Países) (Cancer Today, s. f.)

Según el gráfico 2 se puede observar que Estados Unidos tiene la mayor incidencia de cáncer de tipo no melanoma, seguido por Alemania y Australia. No es casualidad que países donde predomine gente de piel blanca sean los mayores afectados por este tipo de cáncer, ya que, los factores predisponentes más importantes para el desarrollo de este cáncer es la piel blanca (la cual corresponde al fototipo I y II) y a la exposición a la luz ultravioleta proveniente del sol (Darias Domínguez & Garrido Celis, 2018), el Instituto Nacional del Cáncer (NIH) (*Tratamiento del cáncer de piel (PDQ®)–Versión para pacientes - Instituto Nacional del Cáncer, 2021*) añade que si bien la tez clara es un factor de riesgo de este cáncer de piel, este tipo de cáncer se presenta en personas con cualquier color de piel. Con lo que podemos entender que el color de la piel y la exposición al sol aumentan el riesgo tener carcinoma basocelular.

En el gráfico 3 se muestra la relación de incidencias y mortalidad del cáncer de tipo no melanoma entre los sexos masculino y femenino, en el cual se puede apreciar que la incidencia es mayor en el sexo masculino que en el femenino.

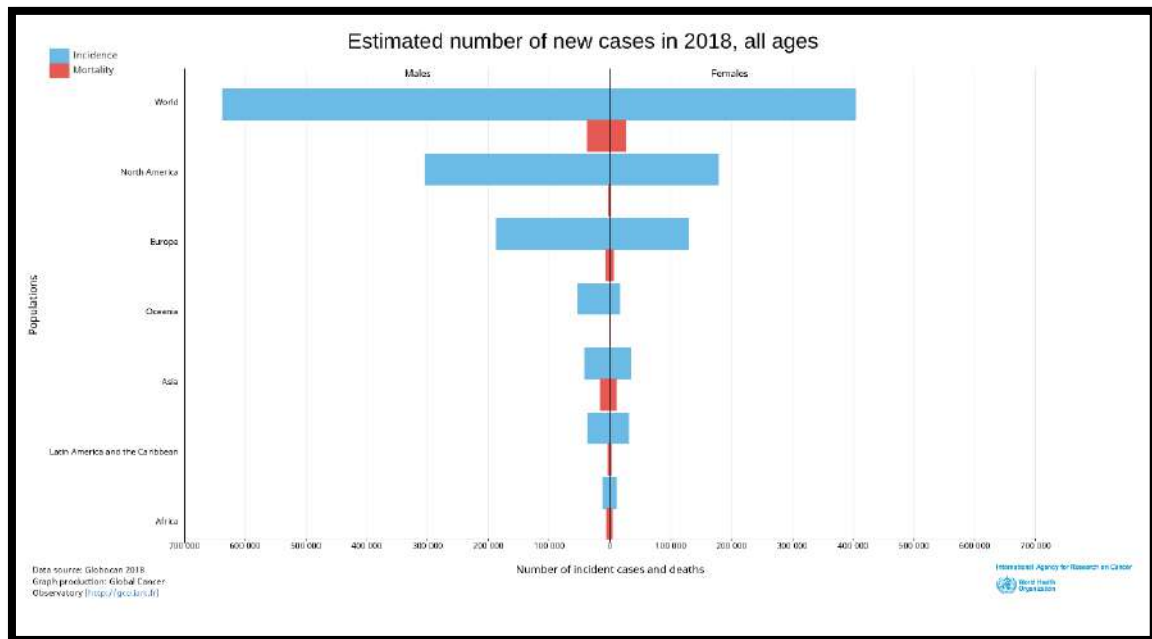


Gráfico 3 Relación de incidencias y mortalidad del cáncer de tipo no melanoma entre los sexos masculino y femenino (Cancer Today, s. f.)

1.1.2. Carcinoma basocelular en Latinoamérica

La incidencia del cáncer de tipo no melanoma en Latinoamérica según el Centro Internacional de Investigaciones sobre el Cáncer (*Cancer Today*, s. f.) es de 51,962 nuevos casos para el año 2018 (Ver gráfico 4). Aplicando la proporción que afirma la Asociación Americana de Cáncer (*¿Qué son los cánceres de piel de células basales y de células escamosas?*, s. f.) existe un aproximado de 41,569 de nuevos casos de carcinoma basocelular en América Latina. Brasil ocupa el primer puesto de incidencias con 25,689 nuevos casos aproximadamente. Colombia y Argentina ocupan el segundo y tercer puesto con 3,741 y 3,242 de nuevos casos respectivamente. Finalmente, el Perú ocupa el quinto puesto con 2,036 nuevos casos.

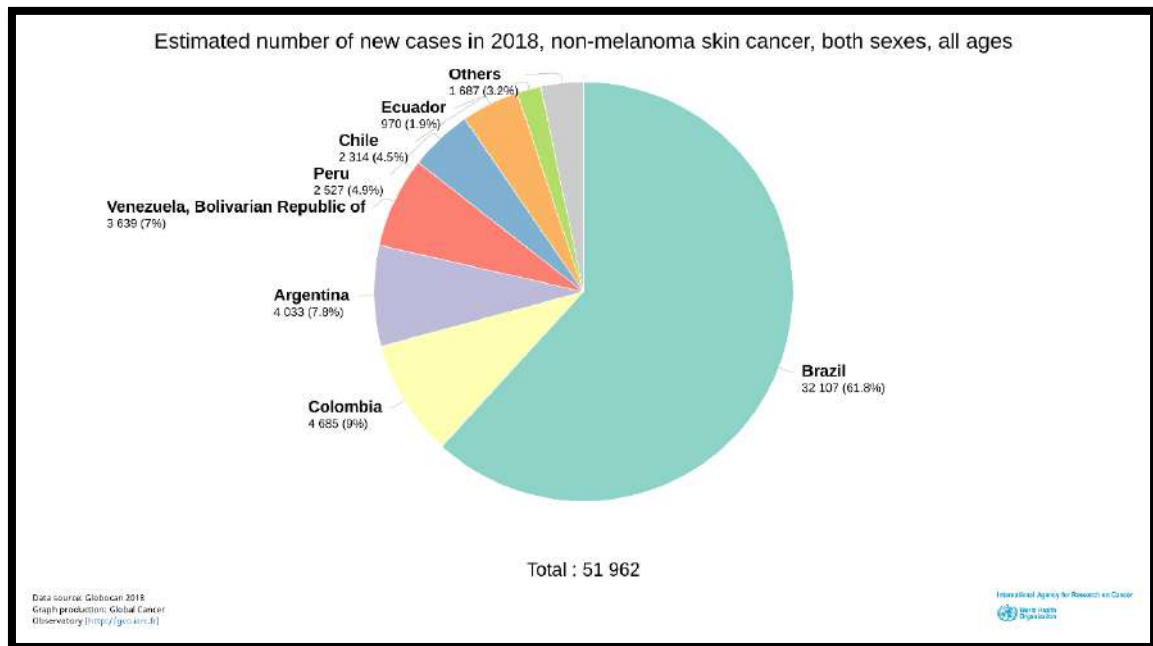


Gráfico 4 Número estimados de nuevos casos de no melanoma en 2018 en Latinoamérica (Cancer Today, s. f.)

En el año 2018, la relación de nuevos casos de cáncer de tipo no melanoma entre personas de sexo masculino y femenino fue como se muestra en la tabla 1, en la cual se puede apreciar que existe una mayor prevalencia de cáncer de tipo no melanoma en hombres que en mujeres en Latinoamérica.

Tabla 1 Relación de nuevos casos de cáncer de tipo no melanoma entre personas de sexo masculino y femenino

	Masculino	Femenino
Brasil	17,309	14,978
Colombia	2,517	2,168
Argentina	2,269	1,768
Venezuela	2,179	1,460
Perú	1,367	1,160

1.1.3. Carcinoma basocelular en el Perú

Según la Revista Peruana de Medicina Experimental y Salud Pública (Sordo & Gutiérrez, 2013) el círculo Dermatológico del Perú (CIDERM) viene realizando desde el año 1995

la “Campaña de Educación, Prevención y Detección Temprana de Cáncer de Piel y Melanoma” denominada el “Día del Lunar” teniendo como objetivos detectar tempranamente las lesiones sospechosas del cáncer de piel y educar a la población sobre los peligros a la sobreexposición al sol. Según los registros del CIDERM entre 1995 y 2011 se atendieron a 118,092 personas. De estas personas atendidas en el 2.8% se identificó alguna lesión cutánea sospechosa de cáncer de piel, y entre estas lesiones la distribución porcentual fue carcinoma basocelular 64,9%, melanoma cutáneo 26,7% y carcinoma espinocelular 8,4%.

En la tabla 2 se muestra un resumen porcentual de los tipos de lesiones encontrados en las personas, las cuales se separan en neoplasias benignas y neoplasias malignas, en la cual se encuentra ubicado el Carcinoma basocelular que representa el 1.95% del 2.8% total de neoplasias malignas.

Tabla 2 Tipos de lesiones encontrados en las personas según los registros del CIDERM (Sordo & Gutiérrez, 2013)

Tipo de lesión	Porcentaje
NEOPLASIAS BENIGNAS	
Nevus melanocíticos	42.63
Queratosis seborreica	19.18
Léntigo solar	13.37
Queratosis solares	7.99
NEOPLASIAS MALIGNAS	
Carcinoma basocelular	1.95
Melanoma	0.83
Carcinoma espinocelular	0.27

Además, según el género, los hombres presentaron una mayor frecuencia de lesiones sospechosas de neoplasias cutáneas malignas en comparación a las mujeres (3.6% vs. 2.6%). Según los grupos etarios, a mayor edad es mayor la frecuencia de estas lesiones sospechosas. Así, entre los menores de 20 años se tiene una frecuencia del 0.3%, entre las personas de 20 a 39 años de 1.7%, entre las personas de 40 a 59 años de 2.9%, entre las personas de 60 a 79 años de 5.6% y entre las personas de 80 a más años esta frecuencia llega a 13.5%.

1.2. Descripción del problema

Según la Gaceta Médica Espirituana (Fernández Martori et al., 2014) tradicionalmente el carcinoma basocelular se ha diagnosticado por medio de un examen clínico y la biopsia con estudio histopatológico de confirmación, pero muchos de ellos en el momento de la exéresis (extirpación de un tejido corporal) no tienen un diagnóstico clínico bien definido e inclusive se extirpan innecesariamente lesiones aparentemente malignas que corresponden a queratosis seborreicas, queratosis actínicas, nevus melanocíticos, verrugas vulgares, entre otros. Todo esto queda atrás con la introducción de la dermatoscopia, técnica que disminuye el margen de error, esta consiste en un sistema de amplificación de la imagen y de iluminación que permite observar estructuras macroscópicas y asociarlas con las microscópicas aportando nuevos criterios morfológicos para la diferenciación de las lesiones no melanocíticas de las melanocíticas. También se ha demostrado que la dermatoscopia resulta eficaz en el diagnóstico del carcinoma basocelular, con una precisión dermatoscópica-histopatológica de un 84,4 %, al compararlo con una publicación de la Universidad de Queensland, Australia en el 2011. La Journal of the American Academy of Dermatology (Rosendahl et al., 2011) revelaron la exactitud de diagnósticos para neoplasias malignas fue de 0,89 con el dermatoscopio y 0,83 sin él. Sin embargo, según Dermatología Revista Mexicana (Snowflake Schwartzberg & Torres González, 2005) en México la precisión diagnóstica de la dermatoscopia fue pobre en un estudio para determinar el nivel de certeza en el diagnóstico dermatoscópico, lo que se recomendó adiestrar mejor a los dermatólogos para ofrecer un mejor pronóstico.

(Kittler et al., 2002) sostiene que la dermatoscopia ha mejorado la precisión diagnóstica del melanoma en un 10-27% frente a los exámenes a simple vista. Sin embargo, la precisión del análisis de imágenes de dermatoscopia todavía depende de la experiencia de un médico. Se recomienda un diagnóstico de consenso en el que participen dos o más expertos para obtener la mayor precisión diagnóstica posible. (Binder et al., 1997) añade que un dermatólogo no capacitado en lectura de imágenes de dermatoscopia puede ser menos preciso que el análisis a simple vista.

Según la Journal of the American Academy of Dermatology (Reiter et al., 2019) realizó una revisión sistemática de 17 estudios que informan la precisión del examen a simple vista y la dermatoscopia para el diagnóstico de carcinoma basocelular. En 6 estudios, los investigadores eran médicos experimentados; en 2, la mayoría de los investigadores eran

médicos novatos (experiencia de 2 años). Los estudios que no informaron el nivel de experiencia de sus investigadores fueron excluidos de este análisis. La experiencia aumentó significativamente la sensibilidad, pero no la especificidad de la endoscopia para el CBC. La sensibilidad obtenida en la dermatoscopia realizada por médicos experimentados fue del 93,6% en comparación con el 75,6% de los médicos que no tenía experiencia. Finalmente afirman que la sensibilidad se vio muy afectada por el nivel de experiencia de los dermoscopistas.

Actualmente en el campo de la medicina se busca alcanzar altos resultados de clasificación utilizando técnicas de inteligencia artificial, puesto que un diagnóstico erróneo dado por un falso positivo o un falso negativo puede repercutir seriamente en la salud de los pacientes. Según (Agarap, 2019) una serie de estudios que involucran enfoques de aprendizaje profundo (redes neuronales convolucionales con múltiples capas) ya se han desempeñado en un número considerable de tareas. Estos incluyen, entre otros, clasificación de imágenes, procesamiento del lenguaje natural, reconocimiento de voz y clasificación de texto. Los modelos (comúnmente redes neuronales convolucionales) utilizados en dichas tareas emplean la función Softmax en la capa de clasificación. Sin embargo, se han realizado estudios (Ahmed et al., 2019; Haghighi et al., 2020; Kurek, 2018; Liu et al., 2018) que utilizan una alternativa a la función Softmax para la clasificación: la máquina de vectores de soporte (SVM). El uso de SVM en una arquitectura de red neuronal artificial produce resultados relativamente mejores que el uso de la función Softmax convencional. Por lo cual, en esta investigación se usó un modelo híbrido para obtener mejores resultados.

1.2.1. Formulación del problema

- **Problema general**
 - ¿Se puede diseñar un sistema el cual diagnostique eficazmente el carcinoma basocelular utilizando inteligencia artificial?
- **Problemas específicos**
 - ¿Qué tipo de arquitectura de red neuronal convolucional es mejor para clasificar el carcinoma basocelular?
 - ¿Es posible construir el sistema que diagnostica el carcinoma basocelular?

- ¿Cómo podemos validar el sistema que diagnostica el carcinoma basocelular?

1.3. Importancia

El CBC clínicamente se puede presentar con una variedad de morfologías que van desde parches eritematosos hasta nódulos ulcerados. No es raro que el CBC muestre pigmentación en la dermatoscopia y según el grado de pigmentación, algunos CBC pueden simular melanomas u otras lesiones cutáneas pigmentadas. Dependiendo del subtipo de CBC y el grado de pigmentación, el diagnóstico diferencial clínico puede ser bastante amplio, desde lesiones inflamatorias benignas hasta melanoma. Las características del CBC incluyen la presencia de cualquiera de las siguientes estructuras (Marghoob & Jaimes, 2019) :

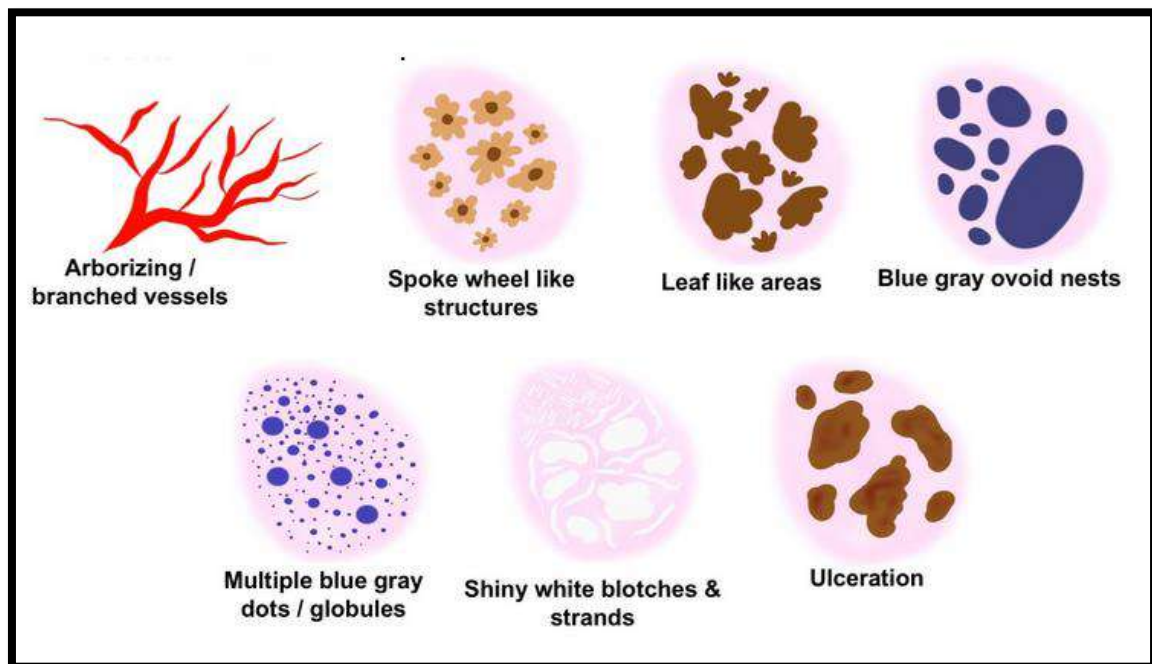


Figura 1 Características del CBC, las estructuras de izquierda a derecha son: arborización / vasos ramificados, estructuras en forma de rueda de radios, áreas como hojas, nidos ovoides gris azules, múltiples puntos / glóbulos grises azules, etc. (Marghoob & Jaimes, 2019)

Como existe una serie de características bien diferenciadas a identificar, el uso de una red neuronal convolucional es adecuada para extraer las características del CBC, puesto que es importante aprender a reconocer estas características para alcanzar un buen diagnóstico del CBC.

Otro punto importante es que exista una herramienta auxiliar que ayude a los dermatólogos poco experimentados para la detección del CBC, ya que la experiencia del

dermatólogo novato está ligada a cuantos casos de carcinoma basocelular ha visto, una red neuronal convolucional puede abstraer rápidamente todo este conocimiento de múltiples casos, lo cual ayudaría a los médicos novatos con el dermatoscopio a dar un diagnóstico correcto. Además, una herramienta basada en redes neuronales convolucionales tendrá un menor costo que otras herramientas que tengan el mismo fin.

Además de la alta incidencia de este cáncer, la detección temprana de CBC es importante para el control de la enfermedad, ya que si no se detecta a tiempo la neoplasia maligna destruirá extensivamente los tejidos circundantes y dañará la estructura de la piel de forma agresiva en etapas avanzadas.

1.4. Motivación

La motivación de esta investigación es implementar un modelo de inteligencia artificial el cual permita clasificar eficazmente el carcinoma basocelular, específicamente, implementar un modelo híbrido el cual permita obtener mejores resultados frente a un modelo clásico de inteligencia artificial, puesto que actualmente hay trabajos (Ahmed et al., 2019; Haghghi et al., 2020; Kurek, 2018; Liu et al., 2018) que clasificaron respectivamente, imágenes de bacterias microscópicas, melanoma, estado de desgaste de taladros en base a imágenes de agujeros perforados y el género de una persona, los cuales obtuvieron mejores resultados utilizando modelos híbridos. Inspirado en estos trabajos, se plantea utilizar un modelo híbrido que busque solucionar este problema de la clasificación del carcinoma basocelular.

1.5. Objetivo

Diseñar el sistema automático de detección de carcinoma basocelular basado en el modelo híbrido de red neuronal convolucional y Support Vector Machine.

1.6. Objetivos específicos

- Diseñar el modelo de detección de CBC utilizando redes neuronales convolucionales y máquina de vectores de soporte que clasifique el CBC.
- Construir un sistema que diagnostica el carcinoma basocelular.

- Validar el modelo que diagnostica el carcinoma basocelular.

1.7. Propuesta

Proponemos un sistema que permita diagnosticar el carcinoma basocelular, utilizando un modelo híbrido de redes neuronales convolucionales y Support Vector Machine.

1.8. Estructura de la tesis

La presente tesis está estructurada en 5 capítulos que se detallan a continuación.

- El capítulo 1 en el cual se detallaron los antecedentes, la problemática, la importancia, la motivación y los objetivos para esta investigación.
- En el capítulo 2, se describe el marco teórico en el que se define qué es el carcinoma basocelular, las redes neuronales convolucionales y el support vector machine.
- En el capítulo 3, se hace un detalle y análisis del estado del arte de investigaciones que han clasificado el carcinoma basocelular u otras lesiones de piel similares como el melanoma.
- En el capítulo 4, se presenta el aporte en el cual se detalla el dataset a utilizar, el modelo propuesto para clasificar el carcinoma basocelular, además de los detalles de implementación del sistema que reconoce el carcinoma basocelular.
- En el capítulo 5 se analiza los resultados obtenidos del entrenamiento del modelo híbrido de red neuronal convolucional con Support Vector Machine, además de presentar las conclusiones y trabajos futuros.

CAPÍTULO 2: MARCO TEÓRICO

2.1 Carcinoma basocelular

El carcinoma basocelular es una neoplasia maligna derivada de células no queratinizadas que se originan en la capa basal de la epidermis. Si no es tratado el carcinoma basocelular sigue su invasión localmente y puede conducir a daño tisular que compromete la función y la estética (Cajina-Vázquez, 2012).

2.1.1. Epidemiología

El carcinoma basocelular es el cáncer más frecuente en los seres humanos representando el 25% de todos los tumores en Estados Unidos de Norte América y el 75% de todos los cánceres de piel no melanomas. La incidencia de cáncer de piel es un fenómeno relacionado directamente con la edad. El aumento en la incidencia de cáncer de piel se ha documentado en todo el mundo. Dicho aumento podría ser producto de diferentes factores, entre los cuales vale la pena mencionar la creciente exposición recreacional al sol, asociada a la cultura del bronceado como un símbolo de belleza, favoreciendo exposiciones prolongadas y continuas. Es más frecuente en individuos ancianos, pero se vuelve cada vez más frecuente en los individuos menores de 50 años, donde se ha demostrado en estudios recientes un incremento significativo en mujeres menores de 40 años. Se desarrolla sobre piel expuesta al sol y es más frecuente en individuos de piel blanca, puesto que la melanina es un protector solar natural. El 20% está localizado en nariz seguido por otros sitios del rostro. Los hombres se ven afectados ligeramente más que las mujeres. Actualmente la diferencia respecto a las mujeres es mucho menos significativa debido al cambio de estilo de vida de muchas de ellas. Los pacientes que presentan carcinoma basocelular tienen 3 veces más riesgo de presentar melanoma (otro tipo de cáncer de piel) (Cajina-Vázquez, 2012).

2.1.2. Etiología y patogenia

El carcinoma basocelular es un tumor epitelial maligno que se origina en las células basales de la epidermis y de sus anexos. En relación con su etiopatogenia se ha reconocido ampliamente el papel de la radiación ultravioleta; donde ésta ha sido ampliamente aceptada como el determinante más importante para el desarrollo del carcinoma basocelular. La exposición a luz UVB (Ultravioleta B) produce daño directo al ADN ya que induce mutaciones en los genes supresores tumorales mientras que la exposición UVA (Ultravioleta A) induce estrés fotooxidativo y mutaciones a través de la generación de especies reactivas de oxígeno generando un nexo potencial entre la luz UV y la disminución de la inmunidad. Es importante tanto la duración como la cantidad de exposición a dichas radiaciones. Las exposiciones muy intensas, aunque de corta duración son las que generan mayor riesgo de carcinoma basocelular. Se dice que cinco quemaduras solares antes de los 18 años aumentan la probabilidad en un 50% de cáncer cuando se llega a la adultez. Se ha descrito que la RUV (radiación ultravioleta) más alta del planeta puede detectarse sobre el trópico en horas cercanas al medio día. El vivir en el ámbito rural tiene una alta correlación con la exposición laboral, debido en gran parte a la mayor probabilidad de estar inmerso en actividades al aire libre en las horas del mediodía. Sin embargo, otros factores como el pobre acceso a campañas preventivas, servicios de salud y medios de comunicación y además hay más propensión a no emplear métodos de protección solar, incrementando el riesgo de padecer esta enfermedad. La sensibilidad cutánea frente a la radiación ultravioleta (fototipo) podría ser un factor para el desarrollo de la enfermedad, específicamente en sujetos fototipo I y II. Pero se ha demostrado en varios estudios que el fototipo III es un grupo de alto riesgo, dado que estas personas experimentan una falsa sensación de seguridad ante la RUV, puesto que en las primeras exposiciones sufren quemaduras leves a moderadas, pero posteriormente desarrollan capacidad de broncearse. Contrariamente a la sensación que tienen ellos y los mismos médicos esta población es extremadamente vulnerable ya que esa falsa sensación de seguridad los lleva a adoptar conductas de riesgo frente al sol, como exponerse a largas jornadas de RUV en busca del color «dorado», no emplear fotoprotector y no utilizar las medidas de barrera física necesarias (sombreros, gafas y mangas largas entre otras). Otros factores que intervienen son la exposición a radiación ionizante, arsénico, clorofenoles, síndromes genéticos, tratamiento durante un largo período con fármacos inmunodepresores: corticoides, azatioprina o ciclosporina. La tendencia a desarrollar carcinomas basocelulares múltiples entre los trastornos hereditarios vinculados se

incluyen el Síndrome de Carcinoma Nevoide de Células Basales o del Nevo de Células Basales, el Síndrome de Bazer y el Síndrome de Rombo (Cajina-Vázquez, 2012).

2.1.3. Comportamiento biológico

El carcinoma basocelular tiene una naturaleza invasora local, la cual es su máximo peligro. Sin tratamiento invade tejido subcutáneo, músculo, hueso e incluso sistema nervioso central. Por medio de los planos de fusión anatómica: surco nasofacial y surco retroauricular; puede invadir más rápidamente. Rara vez realiza invasiones perineurales y cuando ésta se presenta es más frecuente en tumores recurrentes localizados preauricular, malar y periocular. Esta última se manifiesta con síntomas neurológicos focales: dolor, parestesias, debilidad, parálisis. Es extremadamente raro que el carcinoma basocelular de metástasis; cuando lo hace lo realiza con más frecuencia a ganglios linfáticos y pulmones (Cajina-Vázquez, 2012).

2.1.4. Manifestaciones clínicas

Algunos hallazgos en el examen físico pueden ser predictores de riesgo, como han descrito otros autores. La queratosis actínica refleja una historia de exposición solar crónica, por lo que deben orientar a un seguimiento dado su alto papel predictor frente a la aparición de tumores de piel. En general el carcinoma basocelular es una lesión friable que no cicatriza localizada sobre áreas expuestas al sol. Se caracterizan por translucidez, ulceración, presencia de telangiectasias y borde enrollado. Hay cinco subtipos con características específicas; Nodular, es el subtipo más frecuente, representa el 50% de los Carcinomas basocelulares. Pigmentado, es un subtipo del Nodular con más melanización, representa el 25% de los Carcinomas Basocelulares. Estos dos subtipos se localizan mayormente en cabeza y cuello. Superficial es el único subtipo que descama, y se encuentra predominantemente en tronco. Morfeaforme o Esclerosante o Infiltrante, es el más agresivo de las cinco variantes. Fibroepitelioma de Pinkus, se caracteriza por presentarse como una pápula rosada ubicada frecuentemente en la región dorsal baja.

2.1.5. Diagnóstico

Se realiza por medio de dermatoscopia y biopsia cutánea la cual puede ser por rasurado o por sacabocados (procedimiento mediante el cual se extrae un pedazo pequeño de tejido de forma redonda), donde esta última, se prefiere en los casos de carcinoma basocelular morfeaforme y tumores recidivantes (Cajina-Vázquez, 2012).

2.1.5.1. Dermatoscopia

La dermatoscopia es una técnica diagnóstica in vivo no invasiva, desarrollada para estudiar las lesiones cutáneas. Consiste en la visualización de una lesión a través de un dispositivo denominado dermatoscopio. Permite visualizar estructuras cutáneas profundas no visibles a simple vista, denominadas estructuras, parámetros o criterios dermatoscópicos. La dermatoscopia mejora la precisión diagnóstica de las lesiones hiperpigmentadas y el diagnóstico precoz de las lesiones potencialmente malignas, especialmente el melanoma maligno. (Palacios-Martínez & Díaz-Alonso, 2017)

Por otro lado, el dermatoscopio es una lente de aumento a la que se le ha integrado un sistema de iluminación. Se compone de 3 elementos: una lente acromática, una fuente de luz integrada y una fuente de alimentación. Actualmente existen numerosos dispositivos: dermatoscopios manuales o de bolsillo, estereomicroscopios dermatoscopios conectados a cámaras digitales, videodermatoscopios (en ellos la imagen obtenida es enviada directamente a la pantalla de un ordenador). Muchos dispositivos permiten acoplar cámaras de fotos tradicionales o de teléfonos móviles inteligentes para obtener fotografías de las lesiones. Dichas fotografías pueden ser transferidas al ordenador, procesadas o recuperadas.



Figura 2 Uso del dermatoscopio (Palacios-Martínez & Díaz-Alonso, 2017)

2.2. Redes neuronales artificiales

La unidad computacional básica del cerebro es una neurona. Se pueden encontrar aproximadamente 86 mil millones de neuronas en el sistema nervioso humano y están

conectadas con aproximadamente $10^{14} - 10^{15}$ sinapsis. La siguiente figura 3 muestra un dibujo de una neurona biológica (izquierda) y un modelo matemático común (derecha). Cada neurona recibe señales de entrada de sus dendritas y produce señales de salida a lo largo de su axón. El axón eventualmente se ramifica y se conecta a través de sinapsis a las dendritas de otras neuronas. En el modelo computacional de una neurona, las señales que viajan a lo largo de los axones (por ejemplo, x_0) interactúan multiplicándose (por ejemplo, w_0x_0) con las dendritas de la otra neurona en función de la fuerza sináptica en esa sinapsis (por ejemplo, w_0). La idea es que las fuerzas sinápticas (los pesos w) son entrenables y controlan la fuerza de influencia (y su dirección: excitador (peso positivo) o inhibitor (peso negativo)) de una neurona en otra. En el modelo básico, las dendritas llevan la señal al cuerpo celular donde se suman todas. Si la suma final está por encima de cierto umbral, la neurona puede dispararse, enviando un pico a lo largo de su axón. En el modelo computacional, suponemos que los tiempos precisos de los picos no importan, y que solo la frecuencia del disparo comunica información. En base a esta interpretación del código de frecuencia, modelamos la frecuencia de activación de la neurona con una función de activación f , que representa la frecuencia de los picos a lo largo del axón. Históricamente, una opción común de la función de activación es la función sigmoide σ , ya que toma una entrada de valor real (la intensidad de la señal después de la suma) y la convierte para que oscile entre 0 y 1 (*Stanford University CS231n: Convolutional Neural Networks for Visual Recognition*, s. f.)

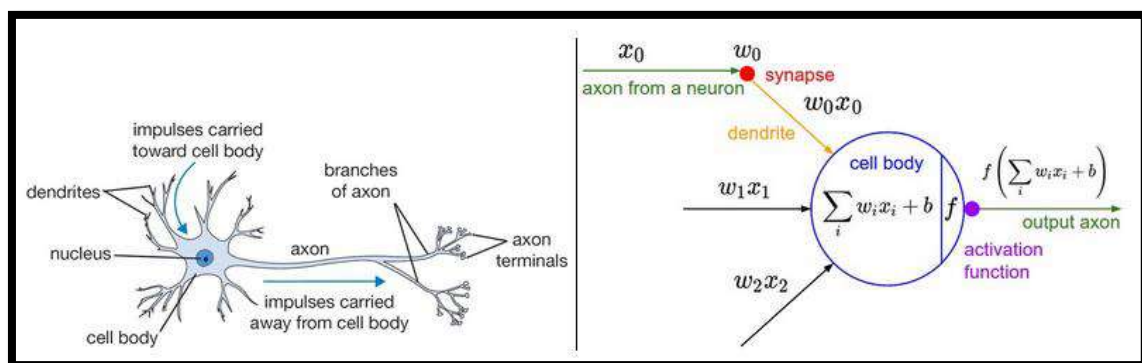


Figura 3 Comparación de una red neuronal biológica y una red neuronal artificial (*Stanford University CS231n: Convolutional Neural Networks for Visual Recognition*, s. f.)

Antes de describir las redes neuronales convolucionales, es necesario ver primero el modelo básico o red neuronal pre-alimentada (feed-forward). Considerando un escenario

de aprendizaje supervisado donde tenemos un conjunto etiquetado de datos (x^i, y^j) , donde x^i e y^j denotan las características y etiquetas respectivamente para el i^{th} ejemplar de entrenamiento. A alto nivel, las redes neuronales proveen una forma compleja de representación, una función no lineal $h_w(x)$ de nuestra variable de entrada x . La función $h_w(x)$ es parametrizada por una matriz de pesos W cuyos valores podemos encontrar para ajustar nuestros datos. En la figura 4 se muestra una red neuronal simple que consiste de tres neuronas de entrada x_{11}, x_{12}, x_{13} y una sola neurona de salida $y = h_w(x_{11}, x_{12}, x_{13})$ (D. J. Wu, 2012)

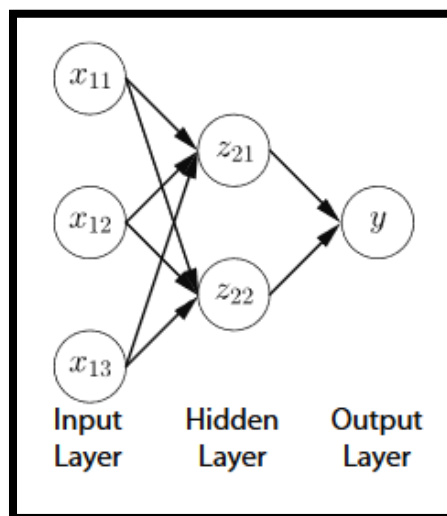


Figura 4 Red neuronal pre-alimentada simple (D. J. Wu, 2012)

Una red neuronal está generalmente organizada en múltiples capas. Por ejemplo, la red neuronal mostrada en la figura 4 consiste de tres capas: La capa de entrada, la capa oculta y una capa de salida, también se muestra un conjunto de aristas conectando las neuronas entre capas adyacentes. Mientras que la figura 4 muestra una red neuronal completamente interconectada donde cada neurona está conectada a todas las neuronas de la capa anterior, no es una condición necesaria en la construcción de la red neuronal. El patrón de conectividad de una red neuronal está generalmente relacionada a la arquitectura de red neuronal (D. J. Wu, 2012).

Aparte de las neuronas en la capa de entrada, cada neurona x_i en la red neuronal es una unidad computacional que toma una entrada de valores de las neuronas de la capa predecesora. Por ejemplo, en la figura 4, la neurona z_{21} tiene como entradas a (x_{11}, x_{12}, x_{13}) y la entrada de y es z_{21} y z_{22} , dadas las entradas, las neuronas primero calculan una

combinación lineal simple de sus entradas, de forma general, dado x_{11}, \dots, x_n que denotan las entradas de la neurona z_j . Entonces primero se calcula (D. J. Wu, 2012):

$$a_j = \sum_{i=1} w_{ij}x_i + b_j$$

Donde w_{ji} es el parámetro que describe la interacción entre z_j y la neurona de entrada x_i . Los términos b_j es un bias o un término de intercepción asociado a la neurona z_j . Luego se aplica una función de activación no lineal a a_j . Algunas funciones de activación comunes incluyen la función sigmoide y la tangente hiperbólica. En particular, la activación o valor de la neurona z_j está definida por (D. J. Wu, 2012):

$$z_j = h(a_j) = \sum_{i=1} w_{ij}x_i + b_j$$

Donde h es la función de activación no lineal. Dado un conjunto de entradas x y pesos W , podemos calcular la activación de cada neurona siguiendo los pasos descritos líneas arriba. La activación de cada neurona depende solo de los valores de las neuronas en las capas predecesoras, se calculan las activaciones comenzando de la primera capa oculta (las cuales a su vez dependen de los valores de sus entradas) y se procede capa por capa de atrás hacia adelante. Este proceso donde la información se propaga a través de la red se denomina paso de propagación hacia adelante. Al final de la propagación hacia adelante, obtenemos un conjunto de salidas $y = h_w(x)$. y viene a ser el resultado de clasificar una entrada x (D. J. Wu, 2012).

Los parámetros W en la red neuronal es tan compuestos por los términos de peso para cada uno de las aristas, así como un término de sesgo o bias para cada una de las neuronas, con exclusión de los que están en la capa de entrada. Dado nuestro conjunto de entrenamiento etiquetado $\{(x^{(i)}, y^{(i)})\}$, el objetivo es aprender o encontrar un valor óptimo para los parámetros W de tal manera que se minimice una función de pérdida. El enfoque estándar para encontrar los valores óptimos de W para minimizar la función de pérdida es el algoritmo de propagación del error hacia atrás (D. J. Wu, 2012).

2.3.1. Funciones de activación

Cada función de activación toma el resultado de la sumatoria de $w_{ij}x_i + b_j$ y se realiza una determinada operación matemática fija en él. Hay varias funciones de activación que se pueden encontrar en la práctica

2.3.1.1. Sigmoidea

La no linealidad sigmoidea tiene la forma matemática $\sigma(x) = 1 / (1 + e^{-x})$ y se muestra en la imagen de arriba a la izquierda. Como se aludió en la sección anterior, toma un número de valor real y lo "aplasta" en un rango entre 0 y 1. En particular, los números negativos grandes se convierten en 0 y los números positivos grandes se convierten en 1. La función sigmoide ha sido utilizada históricamente dado que tiene una buena interpretación como la velocidad de disparo de una neurona: desde no disparar en absoluto (0) hasta disparar completamente saturado a una frecuencia máxima supuesta (1). En la práctica, la no linealidad sigmoidea ha caído recientemente en desuso y rara vez se utiliza. Tiene dos inconvenientes principales: (*Stanford University CS231n: Convolutional Neural Networks for Visual Recognition*, s. f.)

- Los sigmoides saturan y matan los gradientes. Una propiedad muy indeseable de la neurona sigmoidea es que cuando la activación de la neurona se satura en la cola de 0 o 1, el gradiente en estas regiones es casi cero. Durante la propagación hacia atrás, este gradiente (local) se multiplicará por el gradiente de salida de esta puerta para todo el objetivo. Por lo tanto, si el gradiente local es muy pequeño, efectivamente "matará" el gradiente y casi ninguna señal fluirá a través de la neurona a sus pesos y recursivamente a sus datos. Además, se debe tener especial cuidado al inicializar los pesos de las neuronas sigmoideas para evitar la saturación. Por ejemplo, si los pesos iniciales son demasiado grandes, la mayoría de las neuronas se saturarán y la red apenas aprenderá.
- Las salidas sigmoideas no están centradas en cero. Esto no es deseable ya que las neuronas en las capas posteriores de procesamiento en una red neuronal recibirían datos que no están centrados en cero. Esto tiene implicaciones en la dinámica durante el descenso del gradiente, porque si los datos que ingresan a una neurona siempre son positivos (por ejemplo, $x > 0$ por elementos en $f = w^T x + b$), entonces el gradiente en los pesos w durante la retropropagación se volverá positivo o negativo (dependiendo del gradiente de toda la expresión f). Esto podría introducir dinámicas de zigzag no deseadas en las actualizaciones de gradiente para los pesos. Sin embargo, una vez que estos gradientes se suman en un conjunto de datos, la actualización final de los pesos puede tener signos variables, lo que mitiga un poco este problema. Por lo tanto, esto es un inconveniente, pero tiene

consecuencias menos graves en comparación con el problema de activación saturado anterior.

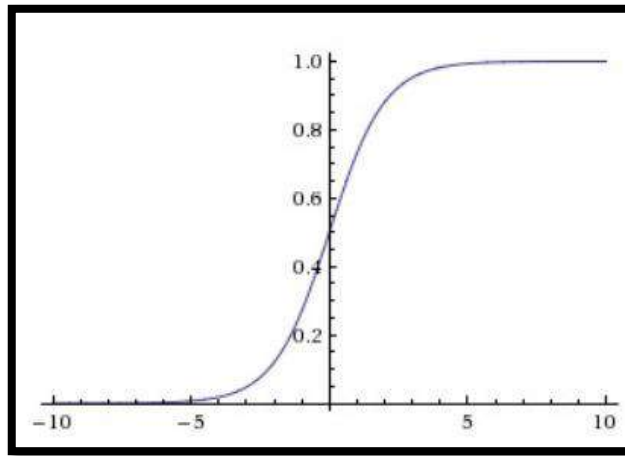


Figura 5 Función sigmoidea (*Stanford University CS231n: Convolutional Neural Networks for Visual Recognition*, s. f., p. 2)

2.3.1.2. Tanh

La no linealidad de Tanh se muestra en la figura 6. Aplasta un número de valor real al rango $[-1, 1]$. Al igual que la neurona sigmoidea, sus activaciones se saturan, pero a diferencia de la neurona sigmoidea, su salida está centrada en cero. Por lo tanto, en la práctica, la no linealidad de tanh siempre se prefiere a la no linealidad sigmoidea. También tenga en cuenta que la neurona tanh es simplemente una neurona sigmoidea escalada, y es definida como: $\tanh(x) = 2\sigma(2x) - 1$. (*Stanford University CS231n: Convolutional Neural Networks for Visual Recognition*, s. f.)

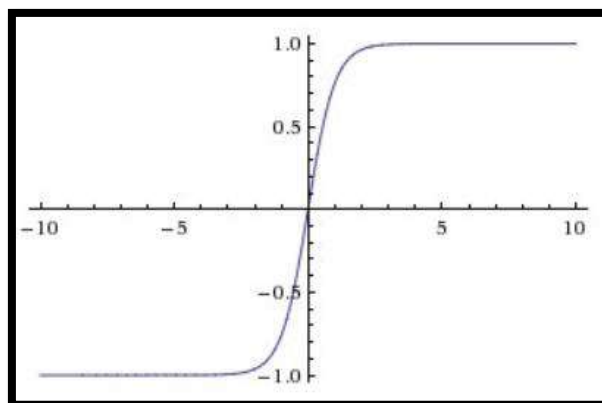


Figura 6 Función Tanh (*Stanford University CS231n: Convolutional Neural Networks for Visual Recognition*, s. f.)

2.3.1.3. ReLU

La unidad lineal rectificada se ha vuelto muy popular en los últimos años. Calcula la función $f(x) = \max(0, x)$. En otras palabras, la activación se limita simplemente a cero (ver figura 7). Existen varios pros y contras para usar las ReLU: (Stanford University CS231n: Convolutional Neural Networks for Visual Recognition, s. f.)

- Se descubrió que acelera en gran medida la convergencia del descenso del gradiente estocástico en comparación con las funciones sigmoideas / tanh. Se argumenta que esto se debe a su forma lineal, no saturante.
- En comparación con las neuronas tanh / sigmoideas que implican operaciones costosas (exponenciales, etc.), la ReLU se puede implementar simplemente con un umbral de una matriz de activaciones en cero.
- Desafortunadamente, las unidades ReLU pueden ser frágiles durante el entrenamiento y pueden "morir". Por ejemplo, un gran gradiente que fluye a través de una neurona ReLU podría hacer que los pesos se actualicen de tal manera que la neurona nunca se active en ningún punto de datos nuevamente. Si esto sucede, entonces el gradiente que fluye a través de la unidad será siempre cero desde ese punto en adelante. Es decir, las unidades ReLU pueden morir irreversiblemente durante el entrenamiento, ya que pueden ser eliminadas del múltiple de datos. Por ejemplo, puede encontrar que hasta el 40% de su red puede estar "muerta" (es decir, neuronas que nunca se activan en todo el conjunto de datos de entrenamiento) si la tasa de aprendizaje es demasiado alta. Con un ajuste adecuado de la tasa de aprendizaje, este problema es menos frecuente.

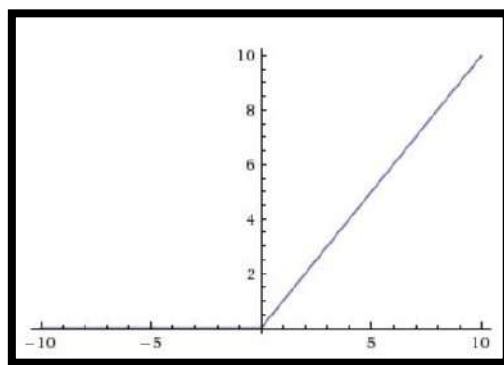


Figura 7 Función ReLU (Stanford University CS231n: Convolutional Neural Networks for Visual Recognition, s. f.)

2.3.4. Redes neuronales convolucionales

En la arquitectura simple, una red neuronal pre-alimentada (feed-forward) descrita en la sección anterior, cada neurona está completamente interconectada a cada uno de las neuronas de las capas siguientes, cada neurona de la capa oculta realiza calcula una función que depende de los valores de cada nodo en la capa de entrada, sin embargo, en visión computacional, es frecuente y ventajoso explorar subestructuras locales en la imagen. Por ejemplo, los píxeles que conforman una vecindad (píxeles adyacentes) tienden a estar fuertemente correlacionados mientras que los píxeles que están más lejos unos de otros en la imagen tienden a estar débilmente correlacionados o simplemente no correlacionados. No es sorpresa entonces que muchas representaciones estándares de características usadas en problemas de visión computacional están basados en características locales de vecindades en la imagen. En una arquitectura CNN (Convolutional Neuronal Network o Red neuronal convolucional en español), se capturan dichas subestructuras locales de la imagen al limitar cada neurona de tal manera que depende solo de un subconjunto espacialmente local de las variables de la capa anterior. Por ejemplo, si la entrada de la CNN es una imagen de 32x32, una neurona en la primera capa oculta podría depender solo de un parche de 8 x 8 del total que es 32 x 32. El conjunto de neuronas en la capa de entrada que afecta la activación de una neurona es denominado como el campo receptivo de la neurona, en una CNN cada neurona individual generalmente tiene un campo receptivo local y no global (D. J. Wu, 2012).

La segunda característica que distingue a las CNNs de las redes neuronales simples es el hecho de que los pesos están compartidos a través de diferentes neuronas en las capas ocultas. Se debe recordar que cada neurona en la red primero calcula una combinación lineal de sus entradas, podemos ver este proceso como la evaluación de un filtro lineal sobre los valores de entrada, en este contexto, compartir los pesos entre múltiples neuronas en la capa oculta se traduce a evaluar el mismo filtro sobre múltiples subventanas de la imagen de entrada, entonces podemos decir que la CNN aprende un conjunto de filtros $F = \{F_i | i=1, \dots, n\}$, cada uno de los cuales es aplicado a todas las subventanas de la imagen de entrada. Usando el mismo conjunto de filtros sobre la imagen entera, la red aprende una codificación o representación general de los datos subyacentes. Restringiendo los pesos para que sean igual es a través de diferentes neuronas también tiene un efecto de regularización sobre la CNN, permitiendo que la red pueda generalizar mejor. Otro beneficio del compartimiento de pesos es el hecho que se reduce

significativamente el número de parámetros libres de la CNN, haciendo el entrenamiento de manera más eficiente. Como nota final, evaluando un filtro F sobre cada ventana en la imagen de entrada I equivale a realizar una convolución de la imagen I con el filtro F , entonces en el paso convolucional de una CNN, se toma la imagen de entrada y se convoluciona con cada filtro F para obtener el mapa de respuesta convolucional (mapa de filtros) (D. J. Wu, 2012).

El componente distintivo final en una CNN es la presencia de capas de agrupamiento (Pooling) cuyo objetivo aquí es doble: reducir la dimensionalidad de las respuestas convolucionales y conferir un pequeño grado de invarianza traslacional al modelo. El enfoque estándar es a través de agrupamiento espacial. En el agrupamiento espacial, el mapa de respuestas convolucionales es primero dividido en un conjunto de $m \times n$ bloques (generalmente disjuntos), luego se evalúa una función de agrupamiento sobre las respuestas en cada bloque. Este proceso da como resultado un mapa de respuestas más pequeño de dimensión $m \times n$ (una respuesta por cada bloque). En el caso de agrupamiento máximo (Max pooling), la respuesta para cada bloque es el valor máximo del bloque de respuestas, y en el caso del agrupamiento promedio (Average pooling), la respuesta es el valor promedio del bloque de respuestas. En la figura 8 se muestra un ejemplo de agrupamiento promedio. En este caso, el mapa de respuestas convolucionales es una entrada de 4×4 , realizamos una función de agrupamiento promedio sobre cuatro bloques de 2×2 , la respuesta agrupada es el promedio de los valores en el bloque. Después de aplicar el procedimiento de agrupamiento promedio, la respuesta es un mapa de respuesta de agrupamiento de 2×2 , comparado al original que era de 4×4 , esto representa una reducción significativa de la dimensionalidad del mapa de respuesta (D. J. Wu, 2012).

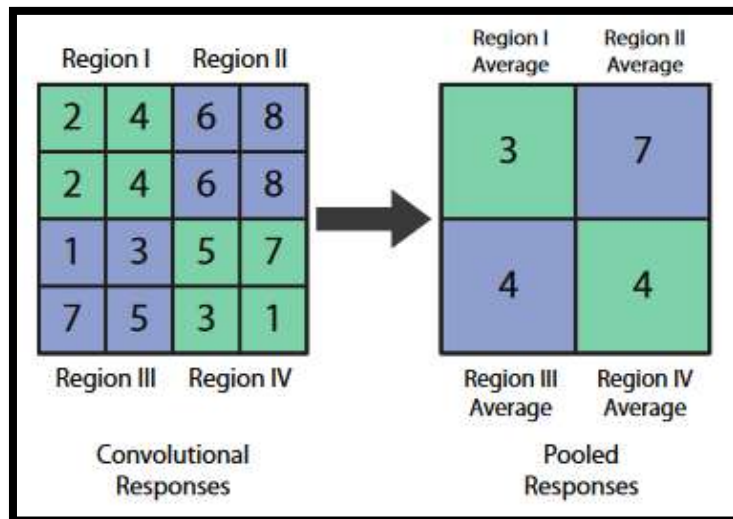


Figura 8 Agrupamiento promedio (Average pooling)

En una CNN típica, tenemos múltiples capas, alternando entre capas de convolución y capas de agrupamiento. Por ejemplo, podemos apilar otra capa de convolución-agrupamiento en la parte superior de las salidas de la primera capa de convolución-agrupamiento, en este caso, simplemente tratamos las salidas del primer conjunto de capas de convolución-agrupamiento como la entrada al segundo conjunto de capas. De esta manera, podemos construir una arquitectura multicapa o profunda. Intuitivamente, los filtros convolucionales de bajo nivel, como aquellos en la primera capa convolucional, se puede pensar que proporcionan un nivel bajo codificación de los datos de entrada. En el caso de los datos de una imagen, estos filtros de bajo nivel pueden consistir en filtros de bordes simples. A medida que nos movemos a capas más profundas en la red neuronal, el modelo comienza a aprender más y más estructuras complicadas. Al usar múltiples capas y grandes números de filtros, la arquitectura CNN puede proporcionar un poder de representación de gran complejidad. Para entrenar una CNN, podemos usar la técnica estándar de propagación del error hacia atrás utilizada para entrenar redes neuronales pre-alimentadas (D. J. Wu, 2012).

2.3.5. Entrenamiento de una red neuronal

Una red neuronal está formada de neuronas conectadas entre ellas; a su vez, cada conexión de nuestra red neuronal está asociada a un peso que dictamina la importancia que tendrá esa relación en la neurona al multiplicarse por el valor de entrada. Cada neurona tiene una función de activación que define la salida de la neurona. La función de

activación se usa para introducir la no linealidad en las capacidades de modelado de la red. Entrenar nuestra red neuronal, es decir, aprender los valores de nuestros parámetros (pesos w_{ij} y sesgos b_j) es la parte más genuina de Deep Learning y podemos ver este proceso de aprendizaje en una red neuronal como un proceso iterativo de “ir y venir” por las capas de neuronas. El “ir” propagando hacia delante lo llamaremos forward-propagation y el “venir” retropropagando información en la red lo llamamos backpropagation. La primera fase forward-propagation se da cuando se expone la red a los datos de entrenamiento y estos cruzan toda la red neuronal para ser calculadas sus predicciones (etiquetas). Es decir, pasar los datos de entrada a través de la red de tal manera que todas las neuronas apliquen su transformación a la información que reciben de las neuronas de la capa anterior y la envíen a las neuronas de la capa siguiente. Cuando los datos hayan cruzado todas las capas, y todas sus neuronas han realizado sus cálculos, se llegará a la capa final con un resultado de predicción de la etiqueta para aquellos ejemplos de entrada. A continuación, usaremos una función de pérdidas para estimar la pérdida (o error) y para comparar y medir cuán bueno/malo fue nuestro resultado de la predicción en relación con el resultado correcto (recordemos que estamos en un entorno de aprendizaje supervisado y disponemos de la etiqueta que nos indica el valor esperado). Idealmente, queremos que nuestro coste sea cero, es decir, sin divergencia entre valor estimado y el esperado. Por eso a medida que se entrena el modelo se irán ajustando los pesos de las interconexiones de las neuronas de manera automática hasta obtener buenas predicciones. Una vez se tiene calculado la pérdida, se propaga hacia atrás esta información. De ahí su nombre, retropropagación, en inglés backpropagation. Partiendo de la capa de salida, esa información de pérdida se propaga hacia todas las neuronas de la capa oculta que contribuyen directamente a la salida. Sin embargo, las neuronas de la capa oculta solo reciben una fracción de la señal total de la pérdida, basándose aproximadamente en la contribución relativa que haya aportado cada neurona a la salida original. Este proceso se repite, capa por capa, hasta que todas las neuronas de la red hayan recibido una señal de pérdida que describa su contribución relativa a la pérdida total. Visualmente, podemos resumir lo que se ha explicado con la figura 9 (Torres, 2018):

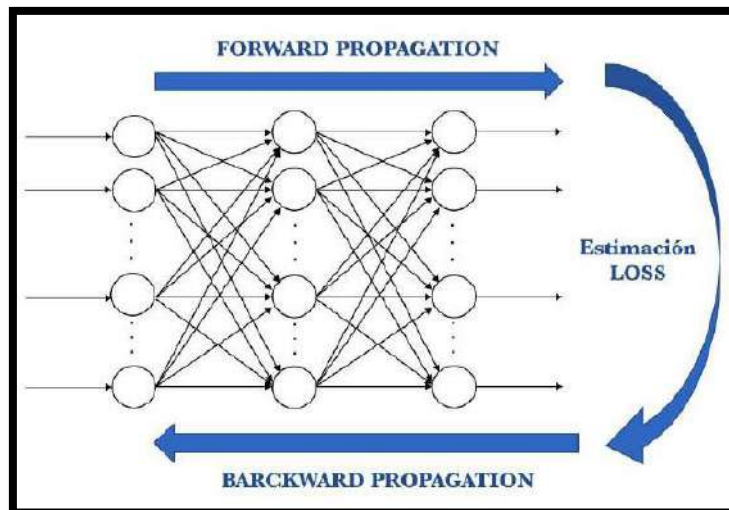


Figura 9 Entrenamiento de la red neuronal (Torres, 2018)

Ahora que ya hemos propagado hacia atrás esta información, podemos ajustar los pesos de las conexiones entre neuronas. Lo que estamos haciendo es que la pérdida se aproxime lo más posible a cero la próxima vez que volvamos usar la red para una predicción. Para ello usaremos una técnica llamada descenso del gradiente. Esta técnica va cambiando los pesos en pequeños incrementos con la ayuda del cálculo de la derivada (o gradiente) de la función de pérdida, cosa que nos permite ver en qué dirección “descender” hacia el mínimo global; esto lo va haciendo en general en lotes de datos (batches) en las sucesivas iteraciones (epochs o épocas) del conjunto de todos los datos que le pasamos a la red en cada iteración. Recapitulando, el algoritmo de aprendizaje consiste en (Torres, 2018):

- Empezar con unos valores aleatorios para los parámetros de la red (pesos w_{ij} y sesgos b_j)
- Coger un conjunto de ejemplos de datos de entrada y pasarlos por la red para obtener su predicción.
- Comparar estas predicciones obtenidas con los valores de etiquetas esperadas y con ellas calcular la pérdida.
- Realizar el backpropagation para propagar esta pérdida a todos y cada uno de los parámetros que conforman el modelo de la red neuronal.
- Usar esta información propagada para actualizar con el descenso del gradiente los parámetros de la red neuronal de manera que reduzca la pérdida total y obtener un mejor modelo.

- Continuar iterando en los anteriores pasos hasta que consideremos que tenemos un buen modelo.

2.3.5.1. Optimización

De forma general, podemos ver el proceso de aprendizaje como un problema de optimización global donde los parámetros (los pesos y los sesgos) se deben ajustar de tal manera que la función de pérdida presentada anteriormente se minimice. En la mayoría de los casos, estos parámetros no se pueden resolver analíticamente, pero en general se pueden aproximar bien con algoritmos de optimización iterativos u optimizadores, como los mencionados anteriormente (Torres, 2018).

2.3.5.1.1. Gradiente descendente

Concretamente el gradiente descendente, la base de muchos y uno de los algoritmos de optimización más comunes en Machine Learning y Deep Learning. El gradiente descendente usa la primera derivada (gradiente) de la función de pérdida cuando realiza la actualización en los parámetros. Recordemos que el gradiente nos da la pendiente de una función en ese punto. El proceso consiste en encadenar las derivadas de la pérdida de cada capa oculta a partir de las derivadas de la pérdida de su capa superior, incorporando su función de activación en el cálculo (por eso las funciones de activación deben ser derivables). En cada iteración, una vez todas las neuronas disponen del valor del gradiente de la función pérdida que les corresponde, se actualizan los valores de los parámetros en el sentido contrario a la que indica el gradiente. El gradiente, en realidad, siempre apunta en el sentido en la que se incrementa el valor de la función de pérdida. Por tanto, si se usa el negativo del gradiente podemos conseguir el sentido en que tendemos a reducir la función de pérdidas (Torres, 2018).

2.3.5.1.2. Adam

Adam es un algoritmo de aprendizaje adaptativo su tasa de aprendizaje se va ajustando automáticamente. Adam almacena un promedio exponencialmente decreciente de gradientes cuadrados que mantiene un promedio exponencialmente decreciente de gradientes pasados m_t (Kingma & Ba, 2017).

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$$

Donde:

- m_t y v_t son estimaciones del primer momento (la media) y el segundo momento (la varianza no centrada) de los gradientes respectivamente
- β_1 y β_2 son las tasas de disminución
- θ es el vector de parámetros inicial
- t es el paso de tiempo (timestep)
- n es el stepsize

Se contrarrestan estos sesgos calculando estimaciones de primer y segundo momento corregidas por sesgo:

$$m'_t = \frac{m_t}{1 - \beta_1^t}$$

$$v'_t = \frac{v_t}{1 - \beta_2^t}$$

Luego se usan para actualizar los parámetros, lo que produce la regla de actualización de Adam:

$$\theta_{t+1} = \theta_t - \frac{n}{\sqrt{v'_t} + \epsilon} m'_t$$

2.3.5.2. Sobreajuste (Overfitting)

Con overfitting nos referimos a lo que le sucede a un modelo cuando este modela los datos de entrenamiento demasiado bien, aprendiendo detalles de estos que no son generales. Esto es debido a que sobreentrenamos nuestro modelo y este estará considerando como válidos solo los datos idénticos a los de nuestro conjunto de entrenamiento, incluidos sus defectos. Es decir, nos encontramos en la situación que el modelo puede tener una baja tasa de error de clasificación para los datos de entrenamiento, pero no se generaliza bien a la población general de datos en los que estamos interesados en realidad (Torres, 2018).

2.3.5.3. Aumento de datos (Data augmentation)

Cuantos menos datos tengamos, menos datos para entrenar y menos posibilidades tendremos de obtener predicciones precisas para datos que nuestro modelo aún no ha visto. Data Augmentation adopta el enfoque de generar más datos de entrenamiento a partir de nuestros datos disponibles. En el caso de imágenes, esto lo consigue aplicando una serie de transformaciones aleatorias a la imagen que producen nuevas imágenes de aspecto creíble. El objetivo es que, en el momento del entrenamiento, nuestro modelo nunca verá exactamente la misma imagen en las diferentes épocas. Esta idea simple, pero potente, ayuda a exponer el modelo a más aspectos de los datos y a generalizar mejor (Torres, 2018).

2.3.5.4. Parámetros e hiperparámetros

En general, se considera un parámetro del modelo como una variable de configuración que es interna al modelo y cuyo valor puede ser estimado a partir de los datos. En cambio, por hiperparámetro se refiere a variables de configuración que son externas al modelo en sí mismo y cuyo valor en general no puede ser estimado a partir de los datos, y son especificados por el programador para ajustar los algoritmos de aprendizaje (Torres, 2018).

2.3.5.4.1. Épocas

Las épocas indica el número de veces en las que todos los datos de entrenamiento han pasado por la red neuronal en el proceso de entrenamiento (Torres, 2018).

2.3.5.4.2. Batch size (Tamaño de lote)

Se puede particionar los datos de entrenamiento en mini lotes para pasarlos por la red. El batch size es el argumento que indica el tamaño que se usará de estos lotes en una iteración del entrenamiento para actualizar el gradiente. El tamaño óptimo dependerá de muchos factores, entre ellos de la capacidad de memoria del computador que usemos para hacer los cálculos (Torres, 2018).

2.3.5.4.3. Learning rate

El vector de gradiente tiene una dirección y una magnitud. Los algoritmos de gradiente descendente multiplican la magnitud del gradiente por un escalar conocido como learning rate (también denominado a veces step size) para determinar el siguiente punto; por ejemplo, si la magnitud del gradiente es 1,5 y el learning rate es 0,01, entonces el algoritmo de gradiente descendente seleccionará el siguiente punto a 0,015 del punto anterior. El valor adecuado de este hiperparámetro es muy dependiente del problema en cuestión, pero en general, si este es demasiado grande, se están dando pasos enormes, lo que podría ser bueno para ir rápido en el proceso de aprendizaje, pero es posible que se salte el mínimo y dificultar así que el proceso de aprendizaje se detenga porque al buscar el siguiente punto perpetuamente rebota al azar en el fondo del “pozo”. Contrariamente, si el learning rate es pequeño, se harán avances constantes y pequeños, teniéndose una mejor oportunidad de llegar a un mínimo local, pero esto puede provocar que el proceso de aprendizaje sea muy lento. En general, una buena regla es que, si nuestro modelo de aprendizaje no funciona, disminuyamos la learning rate. Si sabemos que el gradiente de

la función de pérdidas es pequeño, entonces es seguro probar con learning rate que compensen el gradiente (Torres, 2018).

2.3.5.4.4. Momentum

Para minimizar la función de pérdidas se tiene la garantía de encontrar el mínimo global porque no hay un mínimo local en el que su optimización se pueda atascar. Sin embargo, en realidad, los casos reales son más complejos y visualmente es como si nos pudiéramos encontrar con varios mínimos locales. En este caso, el optimizador puede quedarse atascado fácilmente en un mínimo local y el algoritmo puede pensar que se ha alcanzado el mínimo global, lo que lleva a resultados sub-óptimos. El motivo es que en el momento en que nos atascamos, el gradiente es cero, y ya no podemos salir del mínimo local siguiendo estrictamente lo que hemos contado de guiarnos por el gradiente. El momentum de manera intuitiva, podemos verlo como si un avance tomara el promedio ponderado de los pasos anteriores para obtener así un poco de ímpetu y superar los “baches” como una forma de no atascarse en los mínimos locales. Si consideramos que los promedios de los anteriores eran mejores, quizás nos permita hacer el salto. Pero usar la media ha demostrado ser una solución muy drástica, porque quizás en gradientes anteriores es mucho menos relevante que justo en el gradiente anterior. Por eso se ha optado por ponderar los anteriores gradientes, y el momentum es una constante entre 0 y 1 que se usa para esta ponderación. Se ha demostrado que los algoritmos que usan momentum funcionan mejor en la práctica (Torres, 2018).

2.3.6. Modelos de redes neuronales convolucionales

A través de los últimos años, diversos autores han desarrollado distintos modelos de redes neuronales convolucionales para resolver problemas de clasificación de imágenes, a continuación, se presentaron algunos de los modelos de redes neuronales convolucionales más usados.

2.3.6.1. AlexNet

AlexNet fue un modelo de red neuronal convolucional diseñado por (Krizhevsky et al., 2012). Fue el ganador de la competencia ImageNet Large Scale Visual Recognition Challenge (ILSVRC) en 2012 obteniendo un error de 15.3%. El modelo que propusieron recibe una imagen de 224 x 224 x 3, donde 224 x 224 es la dimensión de las imágenes en píxeles y 3 son los canales de color. La primera capa tiene 96 filtros de 11x11x3 con un stride (ventana) de 4 píxeles, la segunda capa tiene 256 filtros de 5x5x48. La tercera,

cuarta y quinta capas convolucionales están conectadas entre sí sin ninguna intervención de capas de agrupación o normalización. La tercera capa convolucional tiene 384 filtros de tamaño 3x3x256 conectadas a las salidas (normalizadas, agrupadas) de la segunda capa convolucional. La cuarta capa convolucional tiene 384 filtros de tamaño 3x3x192, y la quinta capa convolucional tiene 256 filtros de 3x3x192. Las capas completamente conectadas tienen 4096 neuronas cada una. Finalmente, la capa final tiene 1000 neuronales y usa la función de activación de Softmax.

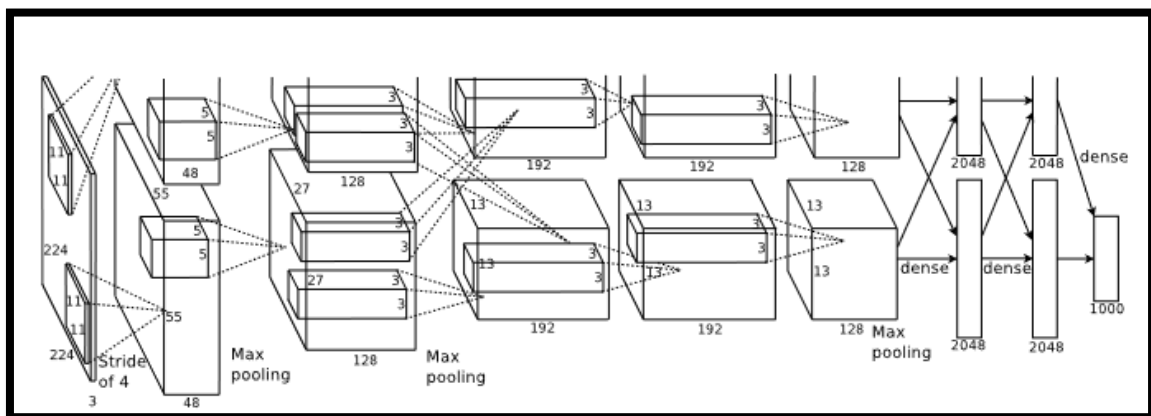


Figura 10 Arquitectura AlexNet. (Krizhevsky et al., 2012)

2.3.6.2. VGG16

VGG16 es un modelo de red neuronal convolucional propuesto por (Simonyan & Zisserman, 2015). Fue usado para la competencia de ImageNet Large Scale Visual Recognition Challenge (ILSVRC) en 2014, el cual fue uno de los mejores modelos para la fecha con 92.7% de exactitud. La arquitectura del VGG16 está compuesto por 13 capas de convolución y 3 capas totalmente conectadas. VGG16 recibe una imagen de entrada de 224 x 224 RGB, y luego la imagen pasa por una pila de capas convolucionales con un filtro de 3x3. Cinco capas de max-pooling siguen después de capa de convolución, con una ventana de 2x2 píxeles con un stride de 2. Luego de las capas de convolución siguen tres capas totalmente conectada, las primeras dos tienen 4096 canales y la última capa (Capa de softmax) tiene 1000 canales. Existe una variación de la red neuronal convolucional VGG16, llamado VGG19 la diferencia con el VGG16 es que VGG19 tiene 3 capas de convolución extras, obteniendo en total 16 capas de convolución y 3 capas totalmente conectadas

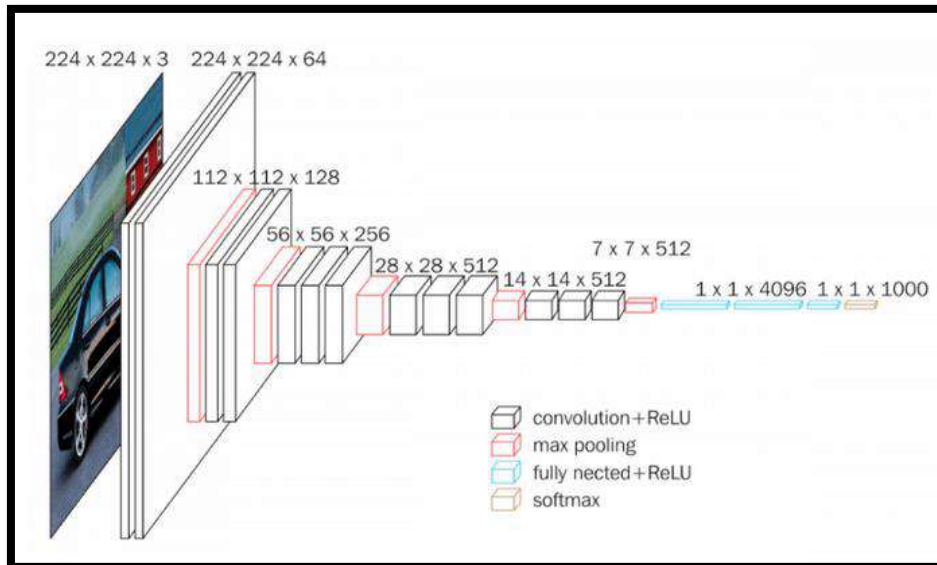


Figura 11 Arquitectura VGG16. (Simonyan & Zisserman, 2015)

2.3.6.3. Inception-V3

En Inception-v3 se hace uso de unos módulos llamados Inception. Estos actúan como múltiples filtros aplicados a un mismo valor de entrada mediante capas convolucionales y de pooling. Esto permite sacar provecho de la extracción de patrones que brindan diferentes tamaños en los filtros. Luego, el resultado de estos filtros es concatenado y utilizado como el valor de salida del módulo. Este modelo aumenta el número de parámetros entrenables y la computación requerida, pero mejora considerablemente la precisión. (Quintero et al., 2018)

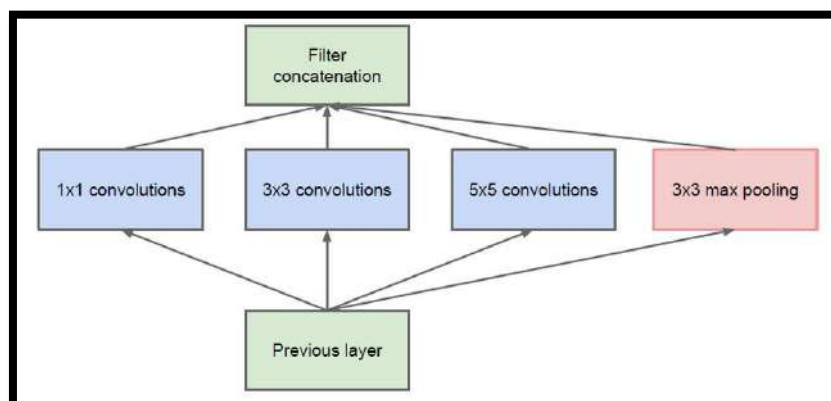


Figura 12 Módulo Inception. (Szegedy et al., 2014)

2.3.6.4. ResNet

ResNet o Red neuronal residual, es un modelo de red neuronal convolucional propuesto por (He et al., 2015). Gracias a esta red por primera vez se pudo entrenar redes muy profundas (Deep learning) con más de 100 capas. ResNet fue presentado en la competición de ImageNet Large Scale Visual Recognition Challenge (ILSVRC) en 2015, ganando la competición con un error de 3.57%. ResNet introdujo el término de bloques residuales debido a que cuando la red comienza a converger, se expone un problema de degradación: con el aumento de la profundidad de la red, la precisión se satura y luego se degrada rápidamente. Tal degradación no es causada por un sobreajuste o por agregar más capas a una red profunda conduce a un mayor error de entrenamiento. Para optimizar esto se usan los bloques residuales, en lugar de esperar que las capas apiladas se ajusten directamente a un mapeo subyacente deseado, dejaron explícitamente que estas capas se ajusten a un mapeo residual. La formulación de $F(x) + x$ se puede realizar mediante redes neuronales pre-alimentadas con conexiones de acceso directo. Las conexiones de acceso directo son aquellas que omiten una o más capas que se muestran en la Figura 13. Las conexiones de acceso directo realizan un mapeo de identidad y sus salidas se agregan a las salidas de las capas apiladas. Al usar las redes neuronales residuales se obtuvieron algunas ventajas, por ejemplo, las ResNets son fáciles de optimizar frente a las redes neuronales clásicas que solo apilan capas, las cuales muestran un mayor error de entrenamiento cuando aumenta la profundidad, además las ResNets puede ganar precisión fácilmente a partir de una profundidad mucho mayor, produciendo resultados que son mejores que las redes anteriores. Existen variaciones de ResNet, según el número de capas ResNet50, ResNet101 o ResNet152.

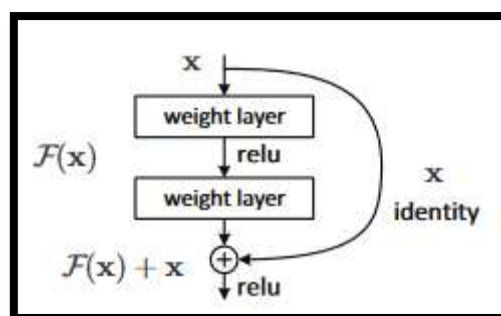


Figura 13 Aprendizaje residual. (He et al., 2015)

2.3.7. Support Vector Machine

(Suárez, 2016) indica que las máquinas de vectores soporte (SVM, del inglés Support Vector Machine) tienen su origen en los trabajos sobre la teoría del aprendizaje estadístico y fueron introducidas en los años 90. Aunque originariamente las SVMs fueron pensadas para resolver problemas de clasificación binaria, actualmente se utilizan para resolver otros tipos de problemas (regresión, agrupamiento, multi-clasificación). También son diversos los campos en los que han sido utilizadas con éxito, tales como visión artificial, reconocimiento de caracteres, categorización de texto e hipertexto, clasificación de proteínas, procesamiento de lenguaje natural, análisis de series temporales. De hecho, desde su introducción, han ido ganando un merecido reconocimiento gracias a sus sólidos fundamentos teóricos. Dentro de la tarea de clasificación, las SVMs pertenecen a la categoría de los clasificadores lineales, puesto que inducen separadores lineales, también llamados hiperplanos, ya sea en el espacio original de los ejemplos de entrada, si éstos son linealmente separables o cuasi-separables (ruido), o en un espacio transformado (espacio de características), si los ejemplos no son linealmente separables en el espacio original. La búsqueda del hiperplano de separación en estos espacios transformados, normalmente de muy alta dimensión, se hará de forma implícita utilizando las denominadas funciones kernel.

Mientras la mayoría de los métodos de aprendizaje se centran en minimizar los errores cometidos por el modelo generado a partir de los ejemplos de entrenamiento (error empírico), el sesgo inductivo asociado a las SVMs radica en la minimización del denominado riesgo estructural. La idea es seleccionar un hiperplano de separación que equidista de los ejemplos más cercanos de cada clase para, de esta forma, conseguir lo que se denomina un margen máximo a cada lado del hiperplano. Además, a la hora de definir el hiperplano, sólo se consideran los ejemplos de entrenamiento de cada clase que caen justo en la frontera de dichos márgenes. Estos ejemplos reciben el nombre de vectores soporte. Desde un punto de vista práctico, el hiperplano separador de margen máximo ha demostrado tener una buena capacidad de generalización, evitando en gran medida el problema del sobreajuste a los ejemplos de entrenamiento.

Dado un conjunto separable de ejemplos $S = \{(x_i, y_i), \dots, (x_n, y_n)\}$, donde $x_i \in R^d$ e $y_i \in \{+1, -1\}$, puede definir un hiperplano de separación (ver gráfico 5) como una función lineal que es capaz de separar dicho conjunto sin error:

$$D(x) = (w_1x_1 + \dots + w_dx_d) + b = \langle w; x \rangle + b$$

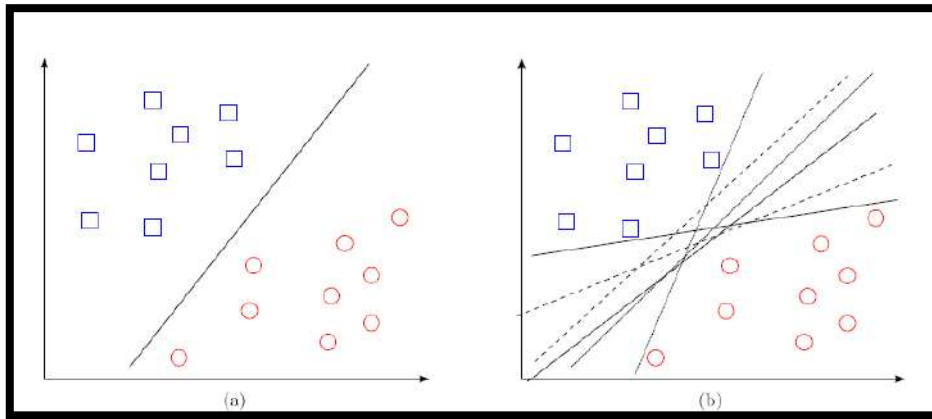


Gráfico 5 Hiperplanos de separación de dos clases: (a) Un hiperplano de separación (b) Otros hiperplanos de separación (Suárez, 2016)

donde $w \in R^d, b \in R$ y el operador $\langle x_1, x_2 \rangle$ representa el producto escalar de los vectores x_1 y x_2 . El hiperplano de separación cumplirá las siguientes restricciones para todo x_i perteneciente al conjunto de ejemplos:

$$\langle w; x_i \rangle + b \geq 0 \text{ si } y_i = +1$$

$$\langle w; x_i \rangle + b \leq 0 \text{ si } y_i = -1, i = 1, \dots, n$$

Tal y como se puede deducir fácilmente del gráfico 6, el hiperplano que permite separar los ejemplos no es único, es decir, existen infinitos hiperplanos separables. Surge entonces la pregunta sobre si es posible establecer algún criterio adicional que permita definir un hiperplano de separación único y óptimo. Para ello, primero, se define el concepto de margen de un hiperplano de separación, denotado por τ , como la mínima distancia entre dicho hiperplano y el ejemplo más cercano de cualquiera de las dos clases (ver gráfico 6a). A partir de esta definición, un hiperplano de separación se denominará óptimo si su margen es de tamaño máximo (ver gráfico 6b). El concepto de margen máximo está relacionado directamente con la capacidad de generalización del hiperplano de separación, de tal forma que, a mayor margen, más equidistará dicho hiperplano de los ejemplos de cada clase. Una propiedad inmediata de la definición de hiperplano de separación óptimo es que éste equidista del ejemplo más cercano de cada clase.

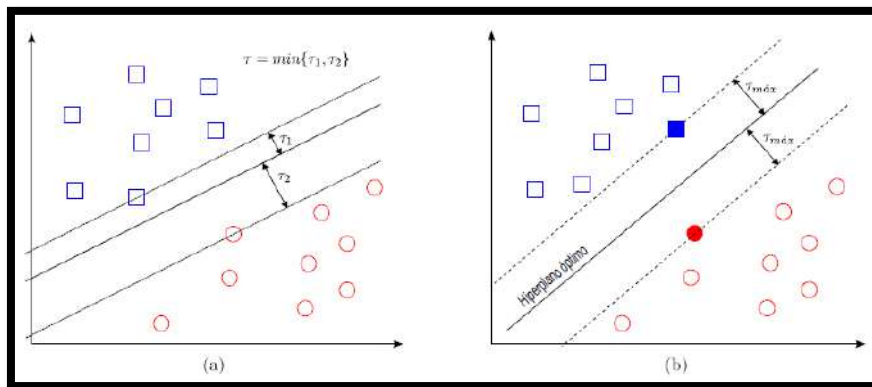


Gráfico 6 Margen de un hiperplano de separación: (a) Hiperplano de separación no óptimo (b) Hiperplano de separación óptimo (Suárez, 2016)

2.3.8. Métricas de evaluación

Según (Borja-Robalino et al., 2020), la disponibilidad de volúmenes masivos de datos ha impulsado el desarrollo de nuevos algoritmos de Machine y Deep Learning que buscan continuamente mejorar el rendimiento en problemas de clasificación multiclase con datos no balanceados. Para resolver un problema de clasificación es necesario seleccionar el algoritmo apropiado mediante una o varias métricas de evaluación del rendimiento en varios algoritmos candidatos. Las métricas más frecuentes utilizadas indistintamente por investigadores en estudios de clasificación predictiva son: exactitud (accuracy), recall (o sensibilidad), matriz de confusión, F1-score, curva ROC y AUC.

- **Matriz de confusión**

Una matriz de confusión es una métrica con la cual se mide el desempeño de un algoritmo de machine learning en el aprendizaje supervisado. Cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fila representa a las instancias en la clase real., o sea en términos prácticos nos permite ver qué tipos de aciertos y errores está teniendo nuestro modelo a la hora de pasar por el proceso de aprendizaje con los datos. La cual está dividida en 4 partes: verdaderos positivos, falsos positivos, falsos negativos y verdaderos negativos. (Barrios, 2019)

- **Verdadero positivo (VP):** El valor real es positivo y la prueba predijo también que era positivo.
- **Verdadero negativo (VN):** El valor real es negativo y la prueba predijo también que el resultado era negativo.

- **Falso negativo (FN):** El valor real es positivo, y la prueba predijo que el resultado es negativo.
- **Falso positivo (FP):** El valor real es negativo, y la prueba predijo que el resultado es positivo.

VALORES PREDICCIÓN	Verdaderos positivos	Falsos Positivos
	Falsos Negativos	Verdaderos Negativos
	VALORES REALES	

Figura 14 Matriz de confusión. (Barrios, 2019)

- **Exactitud**

La exactitud (en inglés, “Accuracy”) se refiere a lo cerca que está el resultado de una medición del valor verdadero. En términos estadísticos, la exactitud está relacionada con el sesgo de una estimación. Se representa como la proporción de resultados verdaderos (tanto verdaderos positivos (VP) como verdaderos negativos (VN)) dividido entre el número total de casos examinados (verdaderos positivos, falsos positivos, verdaderos negativos, falsos negativos)

$$Exactitud = \frac{VP + VN}{VP + FP + FN + VN}$$

- **Precisión**

La precisión (en inglés “Precision”) se refiere a la dispersión del conjunto de valores obtenidos a partir de mediciones repetidas de una magnitud. Cuanto menor es la dispersión mayor la precisión. Se representa por la proporción de verdaderos positivos dividido entre todos los resultados positivos (tanto verdaderos positivos, como falsos positivos).

$$Precisión = \frac{VP}{VP + FP}$$

- **Sensibilidad**

También se conoce como Tasa de Verdaderos Positivos (True Positive Rate) ó TP. Es la proporción de casos positivos que fueron correctamente identificadas por el algoritmo.

$$\text{Sensibilidad} = \frac{VP}{VP + FN}$$

- **Especificidad**

También conocida como la Tasa de Verdaderos Negativos, (“true negative rate”) o TN. Se trata de los casos negativos que el algoritmo ha clasificado correctamente. Expresa cuan bien puede el modelo detectar esa clase.

$$\text{Especificidad} = 2 * \frac{\text{precisión} * \text{sensibilidad}}{\text{precisión} + \text{sensibilidad}}$$

CAPÍTULO 3: ESTADO DEL ARTE

El diagnóstico del cáncer de piel, entre los que se encuentra el carcinoma basocelular ha llamado la atención de varios investigadores, los cuales aplicaron diferentes técnicas de machine learning para solucionar este problema, por lo que se realizó una revisión a la literatura de artículos científicos que busquen aportar sustento científico al problema en distintas revistas y conferencias en los bancos de IEE Xplorer, Science Direct, ACM DL y Springer. Se seleccionaron 20 papers los cuales trataron de resolver el problema del diagnóstico del cáncer de piel. Estos artículos se han dividido en cuatro topologías, la primera topología es arquitectura, la segunda método, la tercera modelo y por último framework. (Ver tabla 3)

Tabla 3 Resumen del estado del arte

Tipo	#Paper	Título	Autor	País	Año	Fuente
Arquitecturas para diagnosticar el cáncer de piel	1	Basal Cell Carcinoma Detection In Full Field OCT Images Using Convolutional Neural Networks	D. Mandache ; E. Dalimier ; J. R. Durkin ; C. Boceara; J.-C. Olivo-Marin; V. Meas-Yedid	Estados Unidos	2018	2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)
	2	Automated segmentation of Mitotic Cells for in vitro histological evaluation of oral squamous cell carcinoma	Dev Kumar Das ; Subhranil Koley ; Chandan Chakraborty ; Asok Kumar Maiti	India	2014	2014 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)
Framework para diagnosticar el cáncer de piel	3	Melanoma Recognition in Dermoscopy Images via Aggregated Deep	Zhen Yu; Xudong Jiang; Feng Zhou ; Jing Qin ; Dong Ni ; Siping Chen ;	Estados Unidos	2019	Journal IEEE Transactions on Biomedical Engineering

		Convolutional Features	Baiying Lei ; Tianfu Wang			
Métodos para diagnosticar el cáncer de piel	4	Automatic detection of translucency using a deep learning method from patches of clinical basal cell carcinoma images	He Huang ; Pegah Kharazmi ; David I. McLean ; Harvey Lui ; Z. Jane Wang ; Tim K. Lee	Estados Unidos	2018	2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)
	5	Deep residual network with regularised fisher framework for detection of melanoma	Nazneen N. Sultana ; Bappaditya Mandal ; N.B. Puan	Reino Unido	2018	Journal IET Computer Vision
	6	Melanoma Screening Using Deep Neural Networks	Jayant Sachdev ; Shashank Shekhar ; S. Indu	India	2018	2018 3rd International Conference for Convergence in Technology (I2CT)
	7	Deep residual neural networks for automated Basal Cell Carcinoma detection	Eric Vander Putten ; Ameer Kambod ; Mobeen Kambod	Estados Unidos	2018	2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)
	8	Skin lesion segmentation using deep convolution networks guided by local unsupervised learning	B. Bozorgtabar ; S. Sedai ; P. K. Roy ; R. Garnavi	Estados Unidos	2017	IBM Journal of Research and Development
	9	An efficient machine learning approach for the	Zahra Waheed ; Amna Waheed ; Madeeha	Pakistan	2017	2017 International Conference on Communication, Computing

		detection of melanoma using dermoscopic images	Zafar ; Farhan Riaz			and Digital Systems (C-CODE)
	10	A Deep Convolutional Neural Network for Melanoma Recognition in Dermoscopy Images	S. N. Haghighi, H. Danyali, M. S. Helfroush and M. H. Karami	Iran	2020	2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)
	11	Keratinocyte Carcinoma Detection via Convolutional Neural Networks	A. Serener and S. Serte	Turquía	2019	2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies
	12	Deep Neural Network for Melanoma Classification in Dermoscopic Images	W. jiahao, J. Xingguang, W. Yuan, Z. Luo and Z. Yu,	China	2021	2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)
	13	Automatic detection of basal cell carcinoma using vascular-extracted features from dermoscopy images	Pegah Kharazmi ; Harvey Lui ; Z. Jane Wang ; Tim K. Lee	Canadá	2016	2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)
Modelos para diagnosticar el cáncer de piel	14	Studies on Different CNN Algorithms for Face Skin Disease Classification Based on Clinical Images	Zhe Wu ; Shuang Zhao ; Yonghong Peng ; Xiaoyu He ; Xinyu Zhao ; Kai Huang ; Xian Wu ; Wei Fan ; Fangfang Li ; Mingliang Chen ; Jie Li ;	Estados Unidos	2019	Journal IEE Access

			Weihong Huang ; Xiang Chen ; Yi Li			
15	Deep Learning and Handcrafted Method Fusion: Higher Diagnostic Accuracy for Melanoma Dermoscopy Images	Jason R. Hagerty ; R. Joe Stanley ; Haidar A. Almubarak ; Norsang Lama ; Reda Kasmi ; Peng Guo ; Rhett J. Drugge ; Harold S. Rabinovitz ; Margaret Oliviero ; William V. Stoecker	Estados Unidos	2019	IEEE Journal of Biomedical and Health Informatics	
16	Convolutional Neural Network Algorithm with Parameterized Activation Function for Melanoma Classification	Abdulaziz Namozov ; Young Im Cho	Sur Corea	2018	2018 International Conference on Information and Communication Technology Convergence (ICTC)	
17	Melanoma Segmentation and Classification in Clinical Images Using Deep Learning	Yunhao Ge; Bin Li; Yanzheng Zhao; Enguang Guan; Weixin Yan	China	2018	ICMLC 2018 Proceedings of the 2018 10th International Conference on Machine Learning and Computing	
18	Automatic skin lesion analysis towards melanoma detection	Le Thu Thao ; Nguyen Hong Quang	Vietnam	2017	2017 21st Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES)	
19	Melanoma Detection Using Convolutional Neural Network	R. Zhang	China	2021	International Conference on Consumer Electronics and Computer Engineering	

	20	Hybrid Model of Convolutional Neural Network and Support Vector Machine to Classify Basal Cell Carcinoma	Alexander Angeles Rojas, Hugo D. Calderon Vilca, Ernesto N. Tumi Figueroa, Kent J. Cuadros Ramos, Steve S. Matos Manguinuri, Edwin F. Calderón Vilca	Perú	2021	Revista de Computación y Sistemas
--	----	--	--	------	------	-----------------------------------

2.1. Arquitecturas para diagnosticar el cáncer de piel

Dada la importancia del problema del diagnóstico del cáncer de piel utilizando los sistemas de información varios investigadores plantearon arquitecturas para resolver este problema.

(Mandache et al., 2018) presentaron una arquitectura de Red neuronal convolucional para detectar el CBC utilizando imágenes de tomografía de coherencia óptica (OCT). Sus datos consistieron en 40 imágenes tomografía de coherencia óptica de campo completo (FFOCT), cada imagen es un corte transversal 2D de una muestra de tejido única con una imagen de 20 μm debajo de la superficie. Las muestras miden entre 2 y 2,5 cm^2 , lo que proporciona imágenes de alta resolución de alrededor de 200 megapíxeles. De todas las imágenes solo 10 presentaron áreas cancerosas, las cuales fueron etiquetadas como CBC. Para el preprocesamiento a las imágenes se le aplicó un filtro Gausseano de 3x3 para proporcionar suavizado a la imagen, después dividieron las imágenes en parches de 256 \times 256 píxeles, con el objetivo de aumentar y también equilibrar el conjunto de datos, sobremuestraron los parches con diferentes valores de paso para las dos clases: 170px para la clase normal, mientras que los parches BCC se superponen más, con un paso de 40px. Esto produjo 108 082 parches: 59,112 normales y 48,970 BCC; El 80% de los cuales sirven como un conjunto de entrenamiento y el resto se utilizó para medir el rendimiento, luego se estandarizaron los datos y se realizó un aumento de datos lo que condujo a duplicar el conjunto de entrenamiento. Paso siguiente desarrollaron una arquitectura clásica de una red neuronal convolucional multicapas. A pesar de que utilizaron una red neuronal clásica aprovecharon ideas de VGG, como: 1) bloques

convolucionales: capas convolucionales consecutivas para capturar entradas más grandes con un repuesto de parámetros; 2) capa de abandono: una fracción de neuronas se elimina al azar para evitar el sobreajuste; 3) unidad lineal rectificadora (ReLU): función de activación utilizada para acelerar los cálculos. Su arquitectura estuvo compuesta de 10 capas que incluye la parte de extracción de características, compuesta de 4 bloques convolucionales (con dos capas convolucionales cada uno) seguidos de max-pooling y un clasificador que consta de dos capas completamente conectadas de 512 y 64 neuronas, respectivamente, por último, hay una neurona de salida cuya activación señala la clasificación del parche de entrada. Por último, definieron que utilizaron el optimizador Adam y escogieron usar mini lotes de 40 muestras para el entrenamiento de la CNN. Sus resultados fueron medidos en función de la precisión, sensibilidad y especificidad con resultados de 95.35%, 95.2% y 96.54% respectivamente.

(Das et al., 2014) afirma que hasta ese momento no se intentó utilizar imágenes histopatológicas de carcinoma escamocelular bucal para la detección de células mitóticas, ya que, encontrar estas desempeña uno de los papeles clave para el diagnóstico de este cáncer. En vista de esto tienen como objetivo desarrollar un esquema de segmentación de células mitóticas para el reconocimiento automático de imágenes microscópicas de carcinoma escamocelular bucal. Para lograr este objetivo los datos fueron recogidos del Departamento de Patología, Midnapur Medical College and Hospital, India. En total se recogieron 15 muestras de tejido oral de pacientes de las cuales se tomaron cinco imágenes de cada muestra. Como siguiente paso, para identificar las posibles células candidatas las imágenes fueron convertidas a una escala de grises, luego segmentaron y filtraron las imágenes para tener una serie de células candidatas y eliminar otras células. Luego se utiliza “los siete momentos de Hu’s” para discriminar las células mitóticas de las células candidatas restantes que serán utilizadas como entrada para el clasificador. Para la clasificación utilizar CART, el cual es un modelo de árbol de decisión no paramétrico. Para los resultados se evaluaron con las métricas de precisión, sensibilidad y F-score con resultados de 83.8%, 73.5% y 78.3% respectivamente. Basándose en los resultados sostuvieron que su metodología propuesta alcanzó una mayor precisión, sensibilidad y F-score que la literatura.

En resumen:

- De estas investigaciones (Mandache et al., 2018) utilizó como conjunto de datos imágenes tomografía de coherencia óptica de campo completo (FFOCT) que ellos mismos consiguieron y (Das et al., 2014) utilizó muestras de tejido oral, con lo que podemos notar que sus datos no fueron imágenes dermatoscópicas como lo hicieron la mayoría de investigaciones.
- También dada las características de su tipo de fuente de datos, estos fueron de poca cantidad con 10 imágenes de áreas cancerosas para (Mandache et al., 2018) y 15 muestras de tejido oral para (Das et al., 2014).
- La técnica utilizada por (Mandache et al., 2018) fue una red neuronal convolucional, a la cual adaptaron ideas de la arquitectura VGG y (Das et al., 2014) utilizó la técnica de CART, el cual es un modelo de árbol de decisión no paramétrico.
- Con respecto a los resultados se puede resaltar que (Mandache et al., 2018) obtuvo los mejores resultados, los cuales fueron medidos en función de la precisión, sensibilidad y especificidad con resultados de 95.35%, 95.2% y 96.54% respectivamente. En cambio (Das et al., 2014) obtuvo el resultado más bajo, los cuales fueron medidos en función de la precisión, sensibilidad y F-score con resultados de 83.8%, 73.5% y 78.3% respectivamente.

2.2. Framework para diagnosticar el cáncer de piel

A continuación, se presenta un framework el cual buscó diagnosticar el cáncer de piel

(Yu et al., 2019) sustentan que la mayoría de los estudios existentes se han centrado principalmente en la ingeniería y clasificación de características, ya sea implícita o explícitamente, suponiendo que la imagen de entrada contiene un objeto de lesión en buen estado. Sin embargo, las imágenes de dermatoscopia pueden no siempre capturar lesiones enteras, o el objeto de la lesión ocupa solo una pequeña parte de una imagen. Varios estudios propusieron adoptar el modelo de bolsa de características (BoF) con características locales para hacer frente con estas situaciones complejas. Aunque la codificación de características en los modelos BoF, como el histograma básico, el vector

de descriptor agregado localmente y el Fisher Vector (FV), se han utilizado ampliamente en varias tareas de clasificación, el rendimiento diagnóstico entregado por estas características hechas a mano sigue siendo insatisfactorio, además, la mayoría de estos métodos contienen procedimientos complejos y tediosos, que conducen a una generalidad deficiente e inaplicabilidad en la práctica clínica. Por lo tanto, presentan un framework para el reconocimiento de imágenes dermatoscópicas utilizando Deep learning y una estrategia de codificación de descriptor local. Para tal fin utilizaron el dataset de ISBI 2016, el cual contiene 1279 imágenes de lesiones dermatoscópicas y de estas se seleccionaron las imágenes que contenían melanoma tomando 173 imágenes de entrenamiento y 75 de prueba. Estas imágenes fueron redimensionadas (224x224 píxeles), normalizadas y se le aplicó el aumento de datos con rotaciones y traslaciones. Luego las imágenes pasan por una Deep residual neural network (ResNet) para la extracción de características, ya que estas han demostrado que pueden acelerar la convergencia de la red profunda y mantener las ganancias de precisión logradas al aumentar sustancialmente la profundidad de la red. Se implementaron las ResNet con diferentes profundidades (con 50 y 101 capas) con un pre-entrenamiento de ImageNet, adicionalmente también se usaron las arquitecturas de AlexNet y VGG16 para evaluar y comparar resultados. Dado la red pre-entrenada y las entradas de las imágenes, se obtiene un vector profundo de características el cual son codificadas en FV (Fisher Vector), el cual es una variación del modelo BoF (bag-of-features), para codificar las características locales y se ha demostrado que tiene un excelente rendimiento en el reconocimiento de imágenes. Finalmente, las representaciones en FV pasan a un clasificador SVM con un kernel chi cuadrado ya que este ha demostrado ser mejor para la clasificación de imágenes la cual se usa para clasificar las imágenes de melanoma. Durante el entrenamiento del SVM, se empleó el algoritmo estocástico de ascenso de coordenadas dobles para minimizar la pérdida. Los resultados se midieron principalmente con las métricas de exactitud y AUC para las arquitecturas de AlexNet con 84.70% y 82.02%; VGG16 con 84.43% y 81.18%; ResNet50 86.54% y 81.49%; ResNet101 con 86.81% y 82.48% respectivamente. Por último, los autores compararon su framework con otras investigaciones del estado del arte que se enfocaron en realizar la extracción de características hechas a mano y otras investigaciones que se enfocaron en métodos de CNN, superando a ambos enfoques.

2.3. Métodos para diagnosticar el cáncer de piel

En la literatura sobre el diagnóstico del cáncer de piel ya existen varios métodos propuestos en los que podemos encontrar los siguientes trabajos.

(Huang et al., 2018) proponen un método basado en un autoencoder disperso apilado (SSAE), el cual es un método de aprendizaje no supervisado con una arquitectura totalmente conectada, para detectar las áreas translúcidas de parches de imágenes clínicas de CBC. Para este objetivo se utilizaron 32 imágenes clínicas de CBC de 32 pacientes del Centro de cuidado de la piel de Vanouwer, estas imágenes tenían una dimensión de 3008x2000 píxeles. De estas imágenes quisieron capturar la translucidez, la cual se define como una apariencia gelatinosa, el cual es un rasgo característico importante de CBC en una etapa temprana porque la característica se puede observar fácilmente. Las áreas translúcidas de estas imágenes fueron segmentadas por un dermatólogo, con el propósito de detectar mejor la translucidez, luego crearon una región de interés (ROI) usando el cuadro delimitador y el tamaño de cada imagen de ROI depende del tamaño de la lesión en la imagen clínica original. De estas imágenes ROI se obtuvieron los parches no superpuestos. Un parche que contiene translucidez se etiquetó como 1 y 0 para no translucidez. El número total de parches fue 4401; hubo 797 parches translúcidos y 3604 parches no translúcidos. Luego se escogieron un parche de 32 x 32 píxeles y el tamaño de entrada para el SSAE fue de 32x32x3. En particular, los SSAE se aplican a parches de imágenes clínicas de la piel. Usar parches tiene dos ventajas. En primer lugar, los parches aumentarán la cantidad de puntos de datos para el proceso de aprendizaje. En segundo lugar, los parches disminuirán la dimensionalidad de los datos de entrada para que la red totalmente conectada pueda transportarse de manera eficiente. Con el propósito de detectar las áreas translúcidas se consideró usar un autoencoder disperso de dos capas, el autoencoder disperso es una red neuronal que consta de múltiples capas de autoencoders dispersos en los que las características se aprenden de una capa por capa. Luego que la imagen pasa por todas las capas de la red neuronal, esta pasa por el clasificador softmax para detectar la translucidez. Finalmente, los resultados se midieron en función de la exactitud, la sensibilidad y la especificidad con 93%, 77% y 97.1% concluyendo que el SSAE trabajó bien para detectar las áreas translúcidas de los parches de las imágenes clínicas.

(Mandal et al., 2018) propusieron un método discriminante regularizado convolucional profundo basado en redes neuronales que extrae características discriminatorias de baja dimensión para la detección de melanoma. Para esto utilizaron los datasets de ISBI (International Symposium on Biomedical Imaging), PH2 y MED-NODE que contienen imágenes de melanoma, las cuales fueron redimensionadas a 224x224 píxeles, luego fueron normalizadas. El aumento de datos fue aplicado para mejorar la performance de la red con transformaciones aleatorias de rotación, traslación y volteo de las imágenes. Usaron una red neuronal residual muy profunda (ResNet50) pre-entrenada en ImageNet, se aplica primero a cada imagen de entrada. Luego se utiliza un enfoque fine-tuning que se usa para el reemplazo de las últimas capas completamente conectadas y el reentrenamiento de toda la red, luego los descriptores profundos locales se extraen de los mapas de activación densos de la última capa convolucional. Las características discriminatorias se extraen de estos descriptores locales profundos utilizando la información de varianza de las muestras de entrenamiento intra e interclase, se regularizan utilizando la codificación de vector de Fisher (codificación que mejora las propiedades invariantes y más discriminatorias de las características profundas extraídas de un modelo previamente entrenado) aplicándole ciertas modificaciones ya que el criterio de Fisher no se puede aplicar directamente en problemas de clasificación binaria y finalmente se extraen las características de baja dimensión para distinguir entre las imágenes de melanoma y las no melanoma. Los resultados fueron medidos con las métricas de exactitud, sensibilidad, especificidad y AUC principalmente y luego compararon con investigaciones relacionadas para cada uno de los datasets. Para el dataset de ISBI 2016 obtuvieron valores máximos de 86.1%, 56% y 92.4% para la exactitud, sensibilidad y especificidad respectivamente utilizando el fine-tuning, sin el fine-tuning los resultados fueron menores. Comparando sus resultados, (Yu et al., 2017) obtuvo mejores resultados en exactitud y especificidad con 86.8% y 97.7% respectivamente, (Codella et al., 2016) obtuvo mejores resultados en sensibilidad con 69.3%. Para el dataset de PH2 obtuvieron valores máximos de 98.5%, 93% y 100% para la exactitud, sensibilidad y especificidad respectivamente. Comparando sus resultados, solo (Barata et al., 2013) lo superó en la especificidad con 100%. Para el dataset de MED-NODE obtuvieron valores máximos de 77.1%, 80.7% y 73.7% para la exactitud, sensibilidad y especificidad respectivamente, obteniendo los mejores resultados posibles en comparación de los otros trabajos que usaron este dataset. Para el dataset de ISBI 2017 obtuvieron valores máximos de 83.2%, 52.9% y 90.5% para la exactitud, sensibilidad y especificidad respectivamente.

Comparando sus resultados, solo (Menegola et al., 2017) lo superó en exactitud y especificidad con 87.2% y 95% respectivamente.

(Sachdev et al., 2018) propusieron una comparación de diferentes modelos de arquitecturas de Deep learning como VGG16, ResNet50, InceptionV3 y un modelo combinado que utiliza un enfoque de “Bucket of models” para el resultado combinado de las tres arquitecturas anteriores además de reemplazar las dos últimas capas softmax por un clasificador SVM para clasificar el melanoma. Para esto utilizaron el dataset de ISIC (International Skin Imaging Collaboration dataset) del cual utilizaron una mezcla del dataset de los años 2016 y 2017. La mayoría de estas imágenes estaban formato JPEG, tenían una dimensión de 512x768 píxeles y las dividieron en 2900 imágenes de entrenamiento y 979 de pruebas. Para el preprocesamiento utilizaron pasos estándar, como que redimensionaron las imágenes a 224x224 píxeles, cargaron pesos pre-entrenados de ImageNet en la VGG16, ResNet50 e InceptionV3 y retiraron la capa softmax de cada red. Luego ingresaron las imágenes de entrenamiento por lotes y en el caso de utilizar el aprendizaje de transferencia (Transfer Learning - TL), se congeló la red pre-entrenada y se obtuvieron las características directamente de la última capa conectada y en el caso de ajuste preciso (Fine Tuning - FT) se descongeló la red pre-entrenada y realizaron las iteraciones para permitir que el modelo se adopte más al problema y se obtengan las características de la misma manera que el aprendizaje por transferencia. Luego entrenaron la red con el clasificador SVM en la capa final. Para el clasificador combinado utilizaron un enfoque de “Bucket of models” para obtener las métricas finales. Los resultados fueron obtenidos para cada una de las tres arquitecturas y el modelo combinado presentados anteriormente midiéndolos en Exactitud(TL), Exactitud(FT), True positive Rate(TL) y True Positive Rate (FT) obteniendo para VGG-19 83%, 84%, 75% y 78% respectivamente; para ResNet50 81%, 83%, 73%, y 76% respectivamente; para InceptionV3 82%, 82%, 71% y 73%; y para el modelo combinado 84%, 85%, 75% y 79% respectivamente, concluyendo que el mejor modelo obtenido fue el modelo combinado.

(Putten et al., 2018) discute con (Cheng et al., 2012) por su trabajo de análisis de telangiectasia para diagnosticar el CBC, ya que, en la práctica real, la precisión real será menor debido a falta de telangiectasias en lesiones tempranas y telangiectasias con características similares en lesiones cutáneas benignas. También discute con (Cheng et al., 2013) el cual utilizó una técnica novedosa de combinar la extracción de

características de las lesiones con las características del perfil personal del paciente y las características de las lesiones del examen físico porque no proporcionó valores de precisión, especificidad o sensibilidad por lo que no se puede sacar conclusiones significativas. Por último, comenta que el trabajo de (Kharazmi et al., 2016) que, si bien su sistema por sí solo no sería lo suficientemente completo, este se puede incorporar a otros trabajos en el futuro. Por lo cual (Putten et al., 2018) propuso un método de red neuronal residual que busca diagnosticar automáticamente el CBC. Para tal fin utilizaron dos datasets, un dataset público llamado “Skin Lesion Analysis Towards Melanoma Detection” del ISBI 2016 y la segunda del International Skin Imaging Collaboration (ISIC). Primero como las imágenes provinieron de dos diferentes datasets el primer paso fue normalizar las imágenes redimensionándolas a 480x480, luego se llevó a cabo el proceso de aumento de datos, en el cual al conjunto de datos original se le hace rotaciones, volteos aleatorios, con esto el dataset llegó a 12,160 imágenes. Luego las imágenes pasaron a un modelo de segmentación, el cual busca dividir la imagen en segmentos para simplificar el análisis de imágenes, con esto lograron generar un mapa de lesiones en blanco y negro que luego se utilizó para extraer el segmento de la imagen original. Después de la segmentación la salida es una matriz con valores de 0 y 1 que será convertida en un canal PNG y se escalan entre 0 y 255, con el valor de 0 correspondiente a una alta probabilidad de información extraña y 255 que indica una alta probabilidad de lesión. Usaron este PNG para recortar una superposición de la lesión original usando un algoritmo simple escrito en Python que dibuja un cuadro delimitador alrededor de la lesión original, además de la disminución en el costo de la memoria al cargar una imagen más pequeña en el NN, hay beneficios adicionales, por ejemplo, evitar características que no son importantes para la lesión, como lunares o pelos adyacentes a la lesión, evitando la fuga accidental de datos. Finalmente se utiliza una red residual profunda (ResNet) para la clasificación y la salida fue codificada con un rango de 0 a 1 que representa una probabilidad de si una imagen es CBC o no. Los resultados se probaron utilizando tres profundidades diferentes: 53, 98 y 152. Las profundidades solo se ajustaron en el modelo de clasificación. El mejor modelo fue a una profundidad de 152, en el que la exactitud, sensibilidad y especificidad ponderada general para detectar BCC de no BCC fue del 93%, 97% y 96%, respectivamente.

(Bozorgtabar et al., 2017) propusieron un método que consiste en una estrategia de fine tuning basado en superpíxeles para utilizar eficazmente las características de los píxeles

de la imagen de la piel para extraer con precisión el borde de la lesión y clasificarlas en una red neuronal convolucional. Para tal fin utilizaron el dataset de ISIC (International Skin Imaging Collaboration) del año 2016. Este conjunto de datos ha sido evaluado en la competencia ISBI por muchos otros equipos. Los participantes se clasificaron según el coeficiente Jaccard (JC). El dataset contenía 900 imágenes de entrenamiento y 379 imágenes de prueba, de los cuales 727 eran benignos y 173 eran malignos del melanoma, además de aplicar la técnica de aumento de datos al rotar las imágenes con 45 grados. (Bozorgtabar et al., 2017) implementó la FCN (Fully convolutional network) de (Long et al., 2015) pero haciendo ciertas modificaciones como quitar la red VGG (Visual Geometry Group) y descartando la capa del clasificador final para luego convertir las capas conectadas por completo en convoluciones. Luego atribuyeron una convolución 1x1 con la dimensión del canal que predicen los puntajes para el fondo y las lesiones cutáneas. Luego, usaron una capa de deconvolución para muestrear las últimas capas de predicciones y regresar los píxeles a la imagen, todo esto es para generar un mapa de puntaje inicial. Después se aplica la técnica de Superpixel based fine-tuning que consistió en que, dado que el mapa de puntuación de segmentación de salida obtenido por FCN no es homogéneo, formularon la relación por pares entre parches de imagen vecinos, imponiendo la estructura gráfica. Las puntuaciones de los píxeles de la imagen de predicción de una red totalmente convolucional proporcionan una región de interés (ROI) que muestra las probabilidades de píxeles de la imagen o superpíxeles que pertenecen a la lesión, la idea principal de esto es propagar la etiqueta de esos parches de imagen altamente confiables (superpíxeles) e inferir las etiquetas de los superpíxeles menos seguros basándose en las similitudes entre cada dos superpíxeles. El mapa de predicción final se binariza utilizando el umbral de Otsu para obtener el resultado de segmentación final y este pasa a la red neuronal convolucional para la clasificación. Los resultados del método de SuperPixel basado en fine-tuning, fueron medidos con las métricas de coeficiente de Jaccard, el coeficiente de Dice, precisión y exactitud con resultados de 80.6%, 89.2%, 90.1% y 92.3% logrando superar al trabajo de (Long et al., 2015) el cual alcanzó 79.5%. 88.5%, 88.7% y 91.6%, llegándose a concluir que se mejoró y superó el trabajo de (Long et al., 2015) sobre el cual se habían basado.

(Waheed et al., 2017) argumentan que la mayor parte de los trabajos encontrados en la literatura se basan en visión por computadora y las técnicas de segmentación. Debido a esto proponen un método eficiente de machine learning para detectar el melanoma de

imágenes dermatoscópicas, este método se centra en extraer un conjunto novedoso de características simples de color y textura junto con el uso del clasificador SVM para clasificar las imágenes de melanomas de todas las imágenes dermatoscópicas. Para tal objetivo utilizaron el conjunto de datos de PH2 el cual contiene 200 imágenes dermatoscópicas de la cual 40 son de melanoma. Cada imagen es de 8-bit RGB en formato JPEG con una resolución de 768x560 píxeles. Su propuesta constó de dos partes principales, la extracción de características y la clasificación. El paso de extracción de características utilizó imágenes dermatoscópicas de colores originales para extraer características de color discriminantes (que dan conocimiento sobre la distribución del color y la variación de color presente en las lesiones cutáneas) con el modelo de color HSV ya que con esta se calcula mejor el sistema visual humano y características de textura (que transmiten el conocimiento sobre las diferentes estructuras presentes en la lesión, por ejemplo, puntos, rayas, agujeros, etc.) en la que se utiliza GLCM (matriz de coincidencia de nivel de grises) que atiende bien las lesiones de piel clara y oscura que tienen intensidades variables. En la fase de clasificación, cada imagen dermatoscópica se clasificó como melanoma y no melanoma usando 13 características extraídas de las fases anteriores que pasa al clasificador SVM como un arreglo. Los resultados obtenidos se midieron utilizando solo características de color, solo características de textura y combinación de características de color y textura, obteniendo 95%, 83% y 96% de exactitud respectivamente. Las características de textura y color se determinan calculando la precisión para cada conjunto de características y también combinando ambos conjuntos y se observa que las características de color por sí solas funcionan mejor que usar solo características de textura.

(Kharazmi et al., 2016) presentaron un método de segmentación para la extracción precisa de los vasos sanguíneos cutáneos en imágenes de dermatoscopia para la clasificación de CBC. Para esto utilizaron un dataset compuesto de 659 imágenes de dermatoscopia normales y de lesiones con cáncer, este dataset fue conformado por diferentes fuentes de datos, entre los que se encontraban “Atlas of dermoscopy”, la universidad de Missouri y “Vancouver Skin Care Centre”. Primero realizaron la segmentación de los vasos de la piel, y es que la pigmentación de la piel obstruye la visibilidad de los vasos. Como solución, propusieron un enfoque basado en la descomposición de la piel. La piel humana tiene una estructura en capas, con diferentes pigmentos responsables de diferentes componentes de color. Entre todos los pigmentos de la piel, la melanina y la hemoglobina

son las más dominantes, siendo esta última la responsable del color de la sangre. Utilizaron el Análisis de Componentes Independientes (ICA) para descomponer la imagen del color de la piel en canales de melanina y hemoglobina. El siguiente paso fue aplicar filtros de forma en diferentes escalas. Luego se genera una máscara de vaso para extraer un conjunto de 12 características vasculares de la imagen del vaso de la lesión, de las cuales algunas características que se pueden destacar son la longitud máxima, el área máxima, número de ramas de la lesión. Finalmente, las 12 características pasan como entrada a un clasificador Random Forest con 100 árboles cada uno construido considerando 4 características aleatorias para realizar una clasificación de dos clases de lesiones cancerosas versus benignas. Los resultados obtenidos se midieron en TP Rate, FP Rate, precisión, sensibilidad y ROC Área. La performance de la clasificación fue de 90.3% en términos de área bajo la curva ROC, 82.7% en términos de precisión y 82.4% en términos de sensibilidad, concluyendo que su técnica de segmentación es la primera en el campo capaz de detectar y segmentar vasos en lesiones pigmentadas y no pigmentadas.

(Haghighi et al., 2020) presentaron un método que reconoce las lesiones de melanoma de las que no son melanoma con gran precisión utilizando redes neuronales convolucionales para el entrenamiento y SVM para la clasificación. Para tal objetivo utilizaron el dataset de “Skin lesión analysis toward melanoma detection” presentados por ISBI 2016. El dataset está compuesto por 900 imágenes de lesiones de piel divididos en dos categorías, melanoma con 173 imágenes y benignas con 173 imágenes. Luego se utilizó la técnica de aumento de datos para que obtener un mejor performance al momento de entrenar la red neuronal, para el aumento de datos se utilizó el aumento geométrico el cual consiste en rotar la imagen en diferentes grados y girarla para compensar las pocas imágenes del dataset. Para el entrenamiento utilizaron una red neuronal convolucional de 25 capas, con 5 capas convolucionales, después de cada capa hay capa ReLU, una capa de normalización por lotes y una capa Max pooling, el cual conforma la etapa de extracción de características seguida de la etapa de clasificación, que consiste en una capa totalmente conectada, una capa de dropout, una capa ReLU y finalmente otra capa totalmente conectada. Los valores de píxeles en bruto de las imágenes de entrada, tienen un tamaño de 227×227 , que se introducen en las capas convolucionales a través de la capa de entrada. La salida de la primera capa convolucional (datos normalizados y agrupados) se entrega a la entrada de la segunda capa convolucional y la filtra con 196 kernels de tamaño

$3 \times 3 \times 96$. La tercera capa convolucional sigue el mismo procedimiento con 128 núcleos de tamaño $3 \times 3 \times 196$. La cuarta y la quinta capa convolucional tienen 100 y 64 kernels con un tamaño de $3 \times 3 \times 128$ y $3 \times 3 \times 100$, respectivamente. Las capas de max pooling utilizadas en todas las capas convolucionales tienen un tamaño de 3×3 con un stride de dos píxeles, seguidas de una capa totalmente conectada que tiene como objetivo traducir los mapas de características extraídos en vectores de características. Teniendo en cuenta que la capa totalmente conectada está contaminada por demasiados parámetros que puede provocar un problema de sobreajuste, se emplea una capa de dropout con un valor de 0,5. Para lograr una mejor clasificación utilizaron el support vector machine (SVM). El clasificador SVM se entrena sobre los vectores de características extraídos por la arquitectura CNN propuesta. El clasificador SVM distingue las lesiones de melanoma de las lesiones benignas en las imágenes dermatoscópicas. Se utilizó la matriz de confusión para evaluar los resultados de la clasificación. La evaluación del rendimiento de este método se basa en los criterios, exactitud, sensibilidad, especificidad y precisión, que suelen aplicarse para evaluar el rendimiento de la clasificación binaria. Y se alcanzó una exactitud de 89.5%, 87.7% de sensibilidad y 91.5% de especificidad.

(Serener & Serte, 2019) propuso un método basado en Deep learning donde emplea arquitecturas como AlexNet, ResNet-18 y ResNet-50 para clasificar el carcinoma basocelular, carcinoma de células escamosas y carcinoma de queratinocitos. Utilizó el dataset ISIC 2019, el cual contiene 25331 imágenes dermatoscópicas y está dividida en 8 categorías: melanoma, nevus, carcinoma basocelular, queratosis actínicas, queratosis benigna, dermatofibromas, lesiones vasculares y carcinoma de células escamosas. Luego se aplicó la técnica de aumento de datos, para generar datos adicionales los cuales se utilizarán en el entrenamiento, contando con un total de 12469 datos de entrenamiento de los cuales de carcinoma basocelular fueron 3323, carcinoma de células escamosas fueron 4624 y melanoma fueron 4522. Para la clasificación se compararon los rendimientos de 3 arquitecturas de red neuronal convolucional: AlexNet, ResNet-18 y ResNet-50. Los resultados se midieron con las métricas de exactitud, sensibilidad, especificidad y el área bajo la curva (AUC). Para la clasificación del carcinoma basocelular utilizando la arquitectura AlexNet se alcanzaron los siguientes resultados: el AUC fue 0.70%, mientras que la exactitud fue de 0.77%, la sensibilidad fue de 0.78% y la especificidad fue de 0.76%. Cuando se utilizó la arquitectura ResNet-50, los resultados de AUC y precisión se mantuvieron casi igual, mientras que la sensibilidad aumenta a 0.88 y la especificidad

disminuye a 0.69. Para el carcinoma de células escamosas los resultados son más altos que los del carcinoma de células basales de células basales. Para AlexNet, ResNet-18 y ResNet-50, son de 0.79, 0.80 y 0.84 de AUC, respectivamente. En los resultados de la clasificación del carcinoma de queratinocitos. se observó que en la arquitectura AlexNet el AUC fue de 0.86%, la precisión fue de 0.85%, la sensibilidad fue de 0.82% y la especificidad fue de 0.92%. Para la arquitectura ResNet-18, los resultados fueron 0.77%, 0.88%, 0.90% y 0.84% para AUC, precisión, sensibilidad y especificidad, respectivamente, son en general peores que los de la arquitectura arquitectura AlexNet. Para la arquitectura ResNet-50, los resultados son comparables a los de la arquitectura AlexNet. En general, se concluyó que las arquitecturas AlexNet o ResNet-50 deberían ser las elegidas entre las tres arquitecturas para la detección de carcinoma de queratinocitos. Para el carcinoma de células escamosas, ResNet-50 obtuvo los mejores resultados y para el carcinoma basocelular la mejor opción es utilizar el ResNet-50.

(jiahao et al., 2021) propuso un método para detectar el melanoma utilizando una red neuronal profunda basado en la arquitectura EfficientNet-B5. Utilizaron el dataset ISIC 2020 Challenge Dataset, el cual proviene del archivo ISIC (International Skin Imaging Collaboration) que contiene la más grande colección de imágenes dermatoscópicas con lesiones de piel. El dataset lo dividieron en 70% de las imágenes para el entrenamiento, el 20% de validación y 10% de las imágenes restantes para las pruebas. La metodología utilizada la dividieron en dos partes, primero aplicaron Efficient-B5, el cual es una red neuronal convolucional profunda, para capturar las características de alto nivel y luego estas características fueron transferidas a cuatro capas totalmente conectadas para clasificar el melanoma. Decidieron utilizar el Efficient-B5 porque escala uniformemente la anchura, la profundidad y resolución con un conjunto de coeficientes de escala fijos, la idea fue inspirada por dos observaciones en primer lugar, el aumento de cualquier dimensión de la red -anchura, profundidad o resolución- mejora la precisión, pero la ganancia de precisión disminuye para los modelos más grandes. En segundo lugar, para conseguir una mayor precisión y eficiencia, es fundamental equilibrar todas las dimensiones de la red -anchura, profundidad y resolución- durante el escalado. Y como mencionaron el clasificador basado en cuatro capas totalmente conectadas recopila las características extraídas por Efficient-B5 para clasificar la imagen en un resultado si pertenece a un melanoma. En una capa totalmente conectada, todas las neuronas tienen conexiones completas con todas las activaciones de la capa anterior. Las capas totalmente

conectadas pasan proceso de retropropagación para determinar los pesos más precisos. Así, estas capas pueden aprender un mapeo de representaciones de características visuales de alto nivel en etiquetas de clasificación. Para la evaluación de sus métricas, utilizaron AUC-ROC o área bajo la curva ROC, para evaluar el performance de su método y luego compararon el performance del Efficient-B5 con otras redes neuronales convolucionales populares, como la VGG16 y ResNet50. Los resultados obtenidos fueron que, su método propuesto alcanzó un mejor resultado con 91.9% de AUC-ROC score, con respecto a VGG16 y ResNet50, los cuales obtuvieron respectivamente 89.2% y 91% de AUC-ROC score respectivamente.

En resumen:

- De estas investigaciones (Bozorgtabar et al., 2017; Haghghi et al., 2020; jiahao et al., 2021; Kharazmi et al., 2016; Mandal et al., 2018; Putten et al., 2018; Sachdev et al., 2018; Serener & Serte, 2019; Waheed et al., 2017) usaron como fuente de datos dataset públicos como PH2, ISBI, MED-NODE e ISIC, (Huang et al., 2018; Kharazmi et al., 2016) consiguieron sus datos de instituciones privadas como la Universidad de Missouri o Vancouver Skin Care Centre.
- Las investigaciones de (Bozorgtabar et al., 2017; Haghghi et al., 2020; jiahao et al., 2021; Kharazmi et al., 2016; Mandal et al., 2018; Putten et al., 2018; Sachdev et al., 2018; Serener & Serte, 2019; Waheed et al., 2017) se basaron en imágenes dermatoscópicas, en cambio (Huang et al., 2018) se basaron en imágenes clínicas o entiéndase como imágenes de la lesión que se puede ver a simple vista, capturando la imagen sin algún instrumento en especializado.
- Para el preprocesamiento, (Haghghi et al., 2020; jiahao et al., 2021; Mandal et al., 2018; Putten et al., 2018; Sachdev et al., 2018; Serener & Serte, 2019) realizaron algunos pasos estándar como el redimensionamiento de las imágenes, aumento de datos o segmentación, (Huang et al., 2018) utilizaron técnicas como máscara ROI y autoencoder disperso apilado (SSAE), (Bozorgtabar et al., 2017) usó la técnica de Superpixel based fine-tuning, (Waheed et al., 2017) usaron el modelo de color HSV y GLCM (matriz de coincidencia de nivel de grises) y (Kharazmi et al., 2016) utilizaron el Análisis de Componentes Independientes (ICA).
- Entre las técnicas para la clasificación podemos notar que, (Bozorgtabar et al., 2017; Huang et al., 2018) utilizaron una red neuronal convolucional, (Mandal et al., 2018; Putten et al., 2018) usaron ResNet, (Sachdev et al., 2018) usó VGG16,

ResNet50 e InceptionV3, (Waheed et al., 2017) usó SVM, (Kharazmi et al., 2016) un clasificador Random Forest, (Haghighi et al., 2020) combinó una red neuronal convolucional y un support vector machine, (Serener & Serte, 2019) comparó la performance entre AlexNet, ResNet-18 y ResNet-50, por último, (jiahao et al., 2021) usó el Efficient-B5. De estos trabajos el que obtuvo el mejor resultado fue (Waheed et al., 2017) con 95% de exactitud, frente a (Bozorgtabar et al., 2017; Haghighi et al., 2020; Huang et al., 2018; Mandal et al., 2018; Putten et al., 2018; Sachdev et al., 2018) con 93%, 83.2%, 85%, 93%, 92.3%, 89.5% de exactitud respectivamente, (Kharazmi et al., 2016) no detalló su resultado en términos de exactitud, pero alcanzó 82.7% de precisión, (Serener & Serte, 2019) comparó los resultados de clasificación de distintas enfermedades de lesiones de la piel alcanzando el mayor resultado de 84% de AUC, finalmente, (jiahao et al., 2021) alcanzó 91.9% de AUC-ROC score.

2.4. Modelos para diagnosticar el cáncer de piel.

Varios estudios ya propusieron modelos para diagnosticar el cáncer de piel, de los cuales podemos mencionar a los siguientes investigadores.

(Z. Wu et al., 2019) presentaron diferentes algoritmos de redes neuronales convolucionales para la clasificación de enfermedades de la piel ubicadas en la cara basada en imágenes clínicas. Para lograr esto utilizaron un dataset de 2,656 imágenes faciales obtenidos de Xiangya-Derm, la cual estaba compuesta de seis enfermedades de la piel, entre los que se encontraba 689 imágenes CBC, de las que separaron 623 imágenes de entrenamiento y 66 imágenes de prueba. Luego usaron 5 arquitecturas diferentes de redes neuronales convolucionales que fueron pre-entrenados en ImageNet. Estas 5 arquitecturas fueron ResNet-50, Inception-v3, DenseNet121, Xception and Inception-ResNet-v2. Para el pre procesamiento las imágenes entraron a la red con una dimensión de 300*300 píxeles e incluyeron técnicas de volteo aleatorio y recorte, además para tratar el desequilibrio de datos utilizaron diferentes pesos en la función de costos. ResNet agrega conexiones entre las capas poco profundas y profundas de la red. Dichas conexiones transmiten directamente la información de la capa superficial a la capa profunda. Por otro lado, la propagación del gradiente a la capa superficial durante la retropropagación aumenta considerablemente el número de capas de red. En Inception

hay diferentes núcleos en un bloque, y cada tipo de núcleo tiene una forma diferente; la salida del bloque combina la salida de diferentes núcleos. Esto mejora la diversidad de la red en términos de ancho y la diversidad de la escala del campo receptivo. DenseNet agrega conexiones entre cada dos capas; es decir, los mapas de características de salida de cada capa se utilizarán como entrada para todas las capas posteriores. Al usar estas conexiones densas, la red reutiliza funciones, mejorando así el rendimiento con menos parámetros, lo que hace que el cálculo sea más eficiente. Xception es una versión actualizada de la estructura Inception, el cual mejora el módulo Inception con una convolución separable en profundidad. Este cambio desacopla las correlaciones espaciales y las correlaciones entre canales. Inception-ResNet es una combinación de estructuras Inception y ResNet, al agregar una conexión residual a la red de inicio, puede entrenar redes más profundas mientras mantiene la diversidad de escala de la red, mejorando así el rendimiento. Estas arquitecturas fueron implementadas como en los papers originales. Los resultados para clasificar el CBC fueron presentados en dos partes, utilizando el modelo no pre-entrenado y pre-entrenado. Para los modelos no pre-entrenados se midieron con las métricas de sensibilidad y precisión obteniendo para Resnet50 61.5% y 47.6%, InceptionV3 63.1% y 39.4%, DenseNet121 70.8% y 52.9%, Xception 52.9% y 65.9%, Inception-ResnetV2 64.6% y 63.7 respectivamente para todas las arquitecturas. Para los modelos pre-entrenados se midieron con las métricas de sensibilidad y precisión obteniendo para Resnet50 74.2% y 60.4%, InceptionV3 79.2% y 45.5%, DenseNet121 76.9% y 57.5%, Xception 83.1% y 65.9%, Inception-ResnetV2 89.2% y 63.7 respectivamente para todas las arquitecturas. Observando los resultados se pudo ver que Inception-ResnetV2 obtuvo un mejor resultado para la métrica de sensibilidad con 89.2% y Xception obtuvo un mejor resultado para la métrica de precisión con 65.9 para los modelos pre-entrenados.

(Hagerty et al., 2019) presentan un modelo que combina el procesamiento de imágenes convencional con el aprendizaje profundo, al fusionar las características de las técnicas individuales (procesamiento de imágenes y selección de características utilizadas por médicos) para obtener una mayor precisión para el diagnóstico de melanoma en imágenes de dermatoscopia. Como primera fuente de datos consiguieron 1636 imágenes dermatoscópicas de 4 clínicas entre los años 2007 y 2011. Estas imágenes se redimensionaron a 2014x768 píxeles de las cuales 367 eran de melanoma. Como segunda fuente de datos utilizaron el dataset de ISIC 2018 de las que se obtuvo 1113 imágenes de

melanoma, luego fueron redimensionadas a 450x460 píxeles. Para el preprocesamiento, una de las técnicas utilizadas es para remover el vello de las imágenes dermatoscópicas, esto se hace para aumentar la exactitud de las imágenes al momento de clasificarlas. Luego se aplica el algoritmo de “Median Color Split Algorithm” para segmentar la lesión del resto de la imagen. Después se aplica “Cheng vessel detection” para detectar los vasos sanguíneos, ya que son necesarios para detectar el cáncer. Además, usan “Atypical Pigment Network Detection” para seleccionar características de una red de pigmentos cuya estructura varía en tamaño y forma. El género, edad, la localización, el tamaño de la lesión del pacientes también son tomados en cuenta. Otro método utilizado fue el detector de líneas de Steger, para encontrar puntos sobresalientes. También se contó con un detector de colores específico para la detección del melanoma. Con las imágenes ya preprocesadas pasó por la arquitectura ResNet (Deep residual network) de 50 capas. Para esto se volvió a redimensionar la imagen a 224x224 píxeles, además de usar la técnica de transfer learning, para reducir la cantidad de datos necesarios y el tiempo requerido para realizar un re-entrenamiento completo. Los resultados fueron medidos utilizando la curva ROC, comparando el Resnet50, el modelo de selección de características (técnicas de preprocesamiento), y el ResNet50 más el modelo de selección de características. La fusión de los dos métodos mostró un resultado más alto con 90% AUC, el modelo de selección de características 88% AUC y el ResNet50 89% AUC.

(Namozov & Cho, 2018) presentaron un modelo de red neuronal profunda con la función de activación de unidades lineales por partes adaptativas (APL) para clasificar el melanoma. Para esto utilizaron como datos de entrenamiento 10,015 imágenes de “ISIC 2018: Skin Lesion Analysis towards Melanoma Detection” el cual posee imágenes de distintas enfermedades de la piel. Estas imágenes pasaron a su modelo propuesto el cual es una red neuronal convolucional que usa algunas ideas de LeNet, el cual se aplicó para reconocer caracteres escritos a mano. Su modelo constó de nueve capas el cual tiene cuatro capas convolucionales y dos capas de pooling para la extracción de características y tres capas fully-connected en la parte final para realizar la clasificación. La modificación principal del uso de LeNet en su modelo de red convolucional fue usar la función de activación lineal por partes adaptativa (APL) en lugar de usar las funciones de activación tradicionales en capas convolucionales. Esta contribución mejora el rendimiento de la red, a pesar de tener algunas características adicionales que aumentan

el tiempo de entrenamiento. Como parte de los resultados y para demostrar las ventajas de usar APL como función de activación contra ReLU y la función tangente, realizaron una comparación entre estas tres. Lo único que hicieron fue cambiar la función de activación a su modelo para cada una de las funciones de activación. Utilizaron 8515 imágenes de entrenamiento y 1500 de prueba. Los resultados de exactitud para la función de activación ReLU, tangente y APL fueron 93.25%, 91.76% y 95.86% respectivamente utilizando aumentos de datos y sin aumento de datos fueron 92.36%, 91.15% y 93.42% respectivamente. Por último, afirmaron que su red convolucional consume menos recursos informáticos y que funciona más rápido.

(Ge et al., 2018) propusieron un sistema de diagnóstico asistido por computadora (CAD) de aprendizaje profundo para la segmentación automática y clasificación de lesiones de melanoma, que contiene una red neuronal totalmente convolucional (FCN) y una red neuronal convolucional específica (CNN). Para tal fin utilizaron el dataset de ISBI 2016, el cual contiene imágenes de lesiones de la piel en la que se encuentra el melanoma. Del dataset se obtuvieron 900 imágenes de entrenamiento (797 benignas y 173 malignas) al cual se le aplicó la técnica de aumento de datos llegando a obtener 9000 imágenes, de estas se tomó el 80% para el entrenamiento y el 20% para las pruebas, teniendo ahora 1390 imágenes malignas y 5890 imágenes benignas. Su propuesta incluyó tres componentes principales, la obtención de la máscara ROI después de la segmentación del modelo, la extracción de características automáticas y la combinación de características y clasificación. Entonces la segmentación del modelo estaba compuesta por el FCN que consistió en una estructura neuronal de 28 capas y estaba diseñada para realizar la segmentación y dar como salida la máscara de interés (ROI). Luego la imagen ROI generada por el FCN se le extrajeron las características, tomando características de alto nivel y características GLCM para luego combinarlas y esta combinación pasó a la capa completamente conectada de la CNN donde se obtuvo la clasificación del melanoma, si era maligno o benigno. Para el entrenamiento inicializaron los pesos con la técnica de Xavier, además de utilizar el algoritmo de optimización Adam, un learning rate de 0.003, dropout de 0.5, entropía cruzada y el gradiente descendente estocástico. Por último, los resultados obtenidos fueron medidos con las métricas de precisión, sensibilidad y especificidad con resultados de 92%, 94% y 93 % respectivamente, concluyendo que su

sistema propuesto puede generalizarse fácilmente a otros problemas desafiantes de segmentación y clasificación de imágenes médicas.

(Thao & Quang, 2017) presentan un modelo basado en deep learning para resolver dos problemas en el análisis de lesiones cutáneas utilizando unas imágenes dermatoscópicas que contiene un tumor cutáneo. Para el primer problema, utilizaron una arquitectura totalmente convolucional-deconvolucional para segmentar automáticamente el tumor de piel de la piel circundante. Para el segundo problema, utilizaron una red neuronal convolucional simple y una arquitectura VGG-16 que utiliza el aprendizaje de transferencia para abordar las dos tareas diferentes en la clasificación de tumores de piel. Para esto utilizaron el dataset de ISIC 2017 Challenge, el cual consistía de 2000 imágenes de entrenamiento y 600 de prueba. Para resolver el primer problema, el cual fue la segmentación del tumor, primero prepararon la imagen redimensionándola a 96x128 píxeles luego se le aplicó un aumento de datos con transformaciones aleatorias obteniendo un total de 12,000 imágenes, separándolas en 10,800 imágenes de entrenamiento y 1,200 de prueba. Luego estas imágenes pasan a la arquitectura convolucional-deconvolucional la cual tiene 7 capas convolucionales y 2 de maxpooling, emplearon el optimizador Adam e inicializaron los pesos usando el método de Xavier con 50 épocas y un tamaño de lote de 30. Finalmente se genera una máscara del tumor la cual es redimensionada a su tamaño original de 224x224 píxeles. Esta imagen preprocesada pasa a una arquitectura simple convolucional el cual posee 18 capas el cual se le inicializó los pesos usando el método Kaiming He., además se utilizaron el optimizador Adam, un dropout de 0.5 y 20 épocas con un tamaño de lote de 20 para el entrenamiento. También las imágenes se probaron en la arquitectura VGG16 y utilizaron el aprendizaje de transferencia para tomar los pesos pre-entrenados de ImageNet, además de solo entrenar las capas superiores utilizando fine-tuning ya que estas poseían características más específicas. Finalmente utilizaron el optimizador de Adam, con 20 épocas y un tamaño de lote de 15 para realizar el entrenamiento de la red. Los resultados fueron medidos con las métricas de exactitud, precisión, sensibilidad, especificidad y AUC para luego comparar los resultados de las dos arquitecturas. Para la red convolucional simple los resultados fueron 49.3%, 64.8%, 27.8%, 83.8%, 41% para exactitud, precisión, sensibilidad, especificidad y AUC respectivamente, y para la arquitectura VGG16 los resultados fueron 79.7%, 76.3%,

45.4%, 34.1%, 90.7% para exactitud, precisión, sensibilidad, especificidad y AUC respectivamente, concluyendo que la arquitectura VGG16 dio mejores resultados.

(Zhang, 2021) propusieron un modelo de detección de melanoma utilizando EfficientNet-B6, para desarrollar su modelo, utilizaron el dataset ISIC 2020 Challenge Dataset, el cual contiene una colección de imágenes dermatoscópicas. Dividieron el dataset en entrenamiento, validación y prueba con una proporción de 7:2:1. Con EfficientNet intentaron ampliar la profundidad de la red, la anchura y la resolución de entrada en la red de referencia. La red base de EfficientNet-B0 se desarrolló aprovechando una búsqueda de arquitectura neuronal multiobjetivo que optimiza tanto la precisión como los recursos computacionales limitados. La razón por la que EfficientNet-B6 es mejor que otras arquitecturas de redes neuronales puede atribuirse a tres factores. El primero es que la red más profunda puede captar características más ricas y complejas, y generalizar bien en nuevas tareas. El segundo es que la red más amplia tiende a ser capaz de extraer características más finas y es más fácil de entrenar. La última es la mayor resolución de las imágenes de entrada, que significa que se alimentan más píxeles. Con una mayor resolución, mi modelo puede captar potencialmente más patrones de grano fino. En el EfficientNet-B6 también se utilizó el aprendizaje por transferencia, el cual puede acelerar el entrenamiento y ayudar al modelo a encontrar un mejor estado de convergencia, en otras palabras, se utilizó el aprendizaje por transferencia para transferir los pesos de la red EfficientNet pre-entrenada para extraer eficazmente características de una imagen dermatoscópica dada. Luego que se extraen las características extraídas desde el EfficientNet-B6, hay 4 capas totalmente conectadas con 512, 256, 128 y una neurona que se encargó de realizar la clasificación. Para entrenar el modelo se usó el optimizador Adam, una tasa de aprendizaje inicial de $1e-3$ y un tamaño de batch de 32. Para la medición de los resultados del modelo, se utilizó la métrica de AUC-ROC (Área bajo la curva ROC) comparándolo con los modelos VGG16 y VGG19. De los resultados obtuvo que Efficient-B6 alcanzó un 91.7% de AUC-ROC score, mientras que VGG16 y VGG19 alcanzaron 89.1% y 90.2% respectivamente.

(Angeles Rojas et al., 2021) propusieron un modelo híbrido de red neuronal convolucional con support vector machine para diagnosticar el carcinoma basocelular. El dataset utilizado fue el HAM10000, el cual es un conjunto de datos que recopiló imágenes dermatoscópicas de lesiones de la piel pigmentadas de dos sitios diferentes, el Departamento de Dermatología de la Universidad de Medicina de Viena de Austria y de

skin cancer practice of Cliff Rosendahlen en Queensland, Australia. Este dataset contiene 10015 imágenes con diferentes enfermedades de la piel. Como primer procedimiento dividieron las imágenes entre las que tenían carcinoma basocelular y las que no tenían carcinoma basocelular, redimensionaron las imágenes a 64 x 64 píxeles y utilizaron la técnica de aumento de datos para balancear la cantidad de imágenes contando al final con un total de 9766 de carcinoma basocelular y 9501 para no carcinoma basocelular y, por último, las imágenes fueron normalizadas. Su modelo de red neuronal convolucional está compuesto por cuatro capas de convolución, una con 32 filtros, dos capas de 64 filtros y una última capa con 128 filtros, todas con una dimensión de 3x3. Cuatro capas de max pooling y 3 capas de dropout. Para la clasificación del support vector machine utilizaron una capa totalmente conectada con 512 neuronas, otra capa totalmente conectada con 256 neuronas y finalmente una capa totalmente conectada con 2 neuronas. Como función de pérdidas utilizaron la norma L1-SVM. Para evaluar sus resultados compararon su modelo utilizando una red neuronal convolucional clásica frente a una red neuronal con support vector machine. Para esto, dividieron sus imágenes en 10,837 imágenes de entrenamiento y 3,613 imágenes de validación que se utilizan para la evaluación del entrenamiento y se usan 4,817 imágenes de prueba para evaluar el modelo con métricas de exactitud, precisión, recall y f1-score. Para la red neuronal convolucional obtuvieron 95.661%, 95.6730%, 95.6612%, 95.6608% para la exactitud precisión, recall y f1-score respectivamente, y para la red neuronal con support vector machine obtuvieron 96.200%, 96,200% 96.205% y 96.200% para la exactitud, precisión, recall y f1-score respectivamente, evidenciando una mejora de 1% para la clasificación de la red neuronal con support vector machine.

En resumen:

- De estas investigaciones (Z. Wu et al., 2019) utilizó un dataset obtenidos de Xiangya-Derm, (Hagerty et al., 2019) consiguió sus datos de 4 clínicas entre los años 2007 y 2011, (Ge et al., 2018; Namozov & Cho, 2018; Thao & Quang, 2017; Zhang, 2021) usaron como fuente de datos dataset públicos como ISIC o ISBI, (Angeles Rojas et al., 2021) utilizó como dataset HAM10000. Las imágenes de (Z. Wu et al., 2019) estaban compuestas de imágenes faciales, las de (Angeles Rojas et al., 2021; Ge et al., 2018; Hagerty et al., 2019; Namozov & Cho, 2018; Thao & Quang, 2017; Zhang, 2021) eran imágenes dermatoscópicas.

- Para el preprocesamiento, (Angeles Rojas et al., 2021; Namozov & Cho, 2018; Z. Wu et al., 2019) realizaron algunos pasos estándar como el redimensionamiento de las imágenes, aumento de datos o segmentación, (Hagerty et al., 2019) aplicó una serie de técnicas, entre algunas tenemos: “Median Color Split Algorithm”, “Atypical Pigment Network Detection”, “Cheng vessel detection”, (Ge et al., 2018) también utilizó pasos estándar, pero añadió el uso de la máscara ROI, (Thao & Quang, 2017) utilizó una arquitectura convolucional-deconvolucional para obtener una máscara del tumor. (Zhang, 2021) no realizó ningún preprocesamiento en sus imágenes.
- Entre las técnicas para la clasificación podemos notar que, (Z. Wu et al., 2019) probó usando diferentes tipos de arquitecturas como ResNet-50, Inception-v3, DenseNet121, Xception and Inception-ResNet-v2, (Hagerty et al., 2019) usó ResNet50, (Ge et al., 2018; Namozov & Cho, 2018) usaron redes neuronales convolucionales, (Thao & Quang, 2017) usó la arquitectura VGG16, (Zhang, 2021) utilizó el EfficientNet-B6, (Angeles Rojas et al., 2021) comparó una red neuronal convolucional clásica frente a una red neuronal convolucional con support vector machine.
- Los resultados obtenidos fueron: (Z. Wu et al., 2019) usaron las métricas de sensibilidad y precisión, obtuvieron respectivamente para Resnet50 74.2% y 60.4%, InceptionV3 79.2% y 45.5%, DenseNet121 76.9% y 57.5%, Xception 83.1% y 65.9%, Inception-ResnetV2 89.2% y 63.7, (Hagerty et al., 2019) mostró su resultado más alto con 90% AUC para la fusión de sus dos métodos propuestos, (Namozov & Cho, 2018) los resultados de exactitud para la función de activación ReLU, tangente y APL fueron 93.25%, 91.76% y 95.86% respectivamente utilizando aumentos de datos, (Ge et al., 2018) sus resultados fueron medidos con las métricas de precisión, sensibilidad y especificidad con resultados de 92%, 94% y 93 % y (Thao & Quang, 2017) para el VGG16 los resultados fueron 79.7%, 76.3%, 45.4%, 34.1%, 90.7% para exactitud, precisión, sensibilidad, especificidad y AUC respectivamente, (Zhang, 2021) obtuvo un 91.7% de AUC-ROC score para el EfficientNet-B6 mientras que VGG16 y VGG19 alcanzaron 89.1% y 90.2% respectivamente. (Angeles Rojas et al., 2021) obtuvo los máximos resultados con respecto a las demás investigaciones, con unos resultados de 96.200%, 96,200% 96.205% y 96.200% para la exactitud, precisión, recall y f1-score respectivamente

2.5. Investigaciones que diagnosticaron el CBC

De los papers enfocados al carcinoma basocelular, (Angeles Rojas et al., 2021; Kharazmi et al., 2016; Putten et al., 2018) utilizaron imágenes dermatoscópicas. (Putten et al., 2018) usó el dataset de ISBI 2016 y para el pre procesamiento realizaron algunos pasos estándar como el redimensionamiento de las imágenes, aumento de datos o segmentación y utilizó una red neuronal ResNet con lo que obtuvo los resultados más altos con exactitud, sensibilidad y especificidad de 93%, 97% y 96% respectivamente, seguido de (Kharazmi et al., 2016) que usó tres fuentes distintas de datos de “Atlas of dermoscopy”, la universidad de Missouri y “Vancouver Skin Care Centre” los cuales conformaron su dataset, para el pre procesamiento utilizaron el Análisis de Componentes Independientes (ICA) y utilizó un clasificador Random Forest obteniendo 82.7% de precisión y 82.4% de sensibilidad. (Mandache et al., 2018) utilizó imágenes de tomografía de coherencia óptica obtenidas por los mismos investigadores y utilizaron una red neuronal convolucional a la cual adaptaron ideas de la arquitectura VGG, obteniendo resultados que fueron medidos en función de la precisión, sensibilidad y especificidad con resultados de 95.35%, 95.2% y 96.54% respectivamente, (Angeles Rojas et al., 2021) utilizaron como fuente de datos el dataset HAM10000 y aplicaron técnicas como el aumento de datos y redimensionamiento de imágenes. Luego compararon la performance de una red neuronal convolucional clásica frente a una red neuronal convolucional con support vector machine (SVM), de las cuales, la red neuronal con SVM obtuvo mejores resultados para las métricas de exactitud, precisión, recall y f1-score con 96.200%, 96,200% 96.205% y 96.200% respectivamente. (Huang et al., 2018) consiguieron sus datos de instituciones privadas como la Universidad de Missouri y Vancouver Skin Care Centre de donde se obtuvo 32 imágenes clínicas de CBC, para el pre procesamiento se usó la máscara ROI y autoencoder disperso apilado (SSAE) y usaron una red neuronal convolucional para la clasificación obteniendo los resultados que se midieron en función de la exactitud, la sensibilidad y la especificidad con 93%, 77% y 97.1% respectivamente.

CAPÍTULO 4: METODOLOGÍA E IMPLEMENTACIÓN

En este capítulo se describe la metodología relacionada al diseño de la red neuronal convolucional con Support Vector Machine que clasifica el carcinoma basocelular detallando el dataset utilizado, los componentes de la red neuronal y describiendo el modelo utilizado. Finalmente, se detalla la implementación del sistema que clasifica el carcinoma basocelular.

4.1. Dataset

El dataset utilizado es HAM10000, el cual es un conjunto de datos que recopiló imágenes dermatoscópicas de lesiones de la piel pigmentadas de dos diferentes sitios, el Departamento de Dermatología de la Universidad de Medicina de Viena de Austria y de skin cancer practice of Cliff Rosendahlen en Queensland, Australia. Este dataset consiste en 10015 imágenes con diferentes enfermedades de la piel entre las que podemos encontrar la queratosis actínica (akiec), carcinoma basocelular (bcc), queratosis seborreica (bkl), dermatofibroma (df), nevus melanocítico (nv), melanoma (mel), lesiones vasculares (vasc). El dataset puede ser obtenido de Harvard Dataverse o Kaggle. A continuación, se muestra una tabla 4 que resume la cantidad de lesiones de la piel y una comparativa de distintos conjuntos de datos que existen en la actualidad.

Tabla 4 Cuadro comparativo de HAM10000 con otros datasets. (Tschandl et al., 2018)

Dataset	License	Total images	Pathologic verification (%)	akiec	bcc	bkl	df	mel	nv	vasc
PH2	Research&Education ^a	200	20.5%	-	-	-	-	40	160	-
Atlas	No license	1024	unknown	5	42	70	20	275	582	30
ISIC 2017 ^b	CC-0	13786	26.3%	2	33	575	7	1019	11861	15
Rosendahl	CC BY-NC 4.0	2259	100%	295	296	490	30	342	803	3
ViDIR Legacy	CC BY-NC 4.0	439	100%	0	5	10	4	67	350	3
ViDIR Current	CC BY-NC 4.0	3363	77.1%	32	211	475	51	680	1832	82
ViDIR MoleMax	CC BY-NC 4.0	3954	1.2%	0	2	124	30	24	3720	54
HAM10000	CC BY-NC 4.0	10015	53.3%	327	514	1099	115	1113	6705	142

4.1.1. Estructura de los registros

El dataset de HAM10000 también provee un archivo Excel con los metadatos de todas las imágenes dermatoscópicas que contiene el dataset. Por cada registro se tiene la siguiente información:

- `lesion_id`: Id de la lesión.
- `image_id`: Otro tipo de id.
- `dx`: Diagnóstico, el cual puede ser `akiec`, `bcc`, `bkl`, `df`, `nv`, `mel`, `vasc`.
- `dx_type`: El tipo de diagnóstico.
- `age`: Edad.
- `sex`: Sexo.
- `localization`: Localización.

De los estos metadatos, se utilizaron los campos **`image_id`** para identificar cada imagen y **`dx`** para identificar el tipo de lesión de piel de cada imagen.

4.2. Componentes

Para la implementación de la red neuronal convolucional, se utilizó el modelo propuesto por (Angeles Rojas et al., 2021), el cual diseñó una red neuronal convolucional con Support Vector Machine, a continuación, se detallan los componentes que conformaron la red neuronal convolucional.

4.2.1 Red neuronal convolucional

La red neuronal convolucional es el componente que se encarga de entrenar el modelo utilizando las imágenes dermatoscópicas provenientes del HAM10000 y encontrar aquellas características que permitan clasificar si existe el CBC o no.

- **Capa de convolución**

En esta capa se realiza la operación de convolución, la cual sirve para obtener los filtros que permitan identificar la imagen del CBC, en otras palabras, es la capa donde se realiza la extracción de características y se obtiene los mapas de características. Con los distintos mapas de características obtenidos, cada uno contiene información que permite identificar el CBC como las estructuras en forma de rueda de radios o los nidos ovoides gris azulados, etc.

- **Capa de pooling**

Esta capa de pooling se utiliza en este trabajo para reducir el costo computacional de las operaciones y por lo tanto realizar un entrenamiento de la red mucho más rápido, puesto que al disminuir la cantidad de píxeles de las imágenes del HAM10000, también reducimos la cantidad de operaciones que se deben realizar.

- **Capa de dropout**

Esta capa se encarga de desactivar algunas neuronas aleatoriamente, las cuales no se toman en cuenta en el entrenamiento, esto se hace con el motivo de evitar el sobreajuste (overfitting) cuando la red neuronal convolucional está aprendiendo a identificar las características del carcinoma basocelular.

- **Capa Flatten**

Esta capa se usa para aplanar la salida de la red neuronal convolucional, en otras palabras, la entrada multidimensional de la imagen dermatoscópica de 64 x 64 píxeles al final se transforma en una salida unidimensional que pasa a la capa totalmente conectada

- **Capa totalmente conectada**

Luego de aplicar varias capas de convolución y pooling se agrega una capa final para nuestra clasificación del CBC, para esto se usa la capa totalmente conectada. Esta se comporta como una red backpropagation en donde se actualizan y guardan los distintos pesos de cada capa que permiten clasificar el CBC.

4.2.2. Support Vector Machine

Del Support Vector Machine se utiliza la ecuación de la norma L1-SVM como función de pérdidas para realizar la clasificación del CBC. Esta función de pérdidas se implementa en la última capa de la red neuronal convolucional para clasificar el carcinoma basocelular.

4.3. Modelo propuesto para clasificar el CBC

El modelo propuesto por (Angeles Rojas et al., 2021) para la clasificación del carcinoma basocelular (CBC) está compuesto de tres partes, el pre procesamiento, el entrenamiento y la clasificación. En el pre procesamiento, la imagen se preparó para que sirva como input para la red neuronal convolucional, esto se hace con el fin de que la red entrene

eficientemente. Para el entrenamiento utilizamos la red neuronal convolucional utilizando las imágenes del HAM10000. Las imágenes pasaron por cuatro capas de convolucionales con filtros de 3x3 y pasaron por cuatro capas de max pooling con filtros de 2x2 para obtener los filtros que permitan identificar el CBC. Luego para la clasificación, las imágenes pasaron por la capa totalmente conectada (fully connected) compuesta de dos capas densas de 512 neuronas y 256 neuronas para llegar finalmente a una capa densa con 2 neuronas, donde en esta última, se utiliza la función de pérdida L1-SVM y la función de activación lineal para obtener el resultado final. A continuación, en la figura 15 se muestra el modelo para clasificar el CBC.

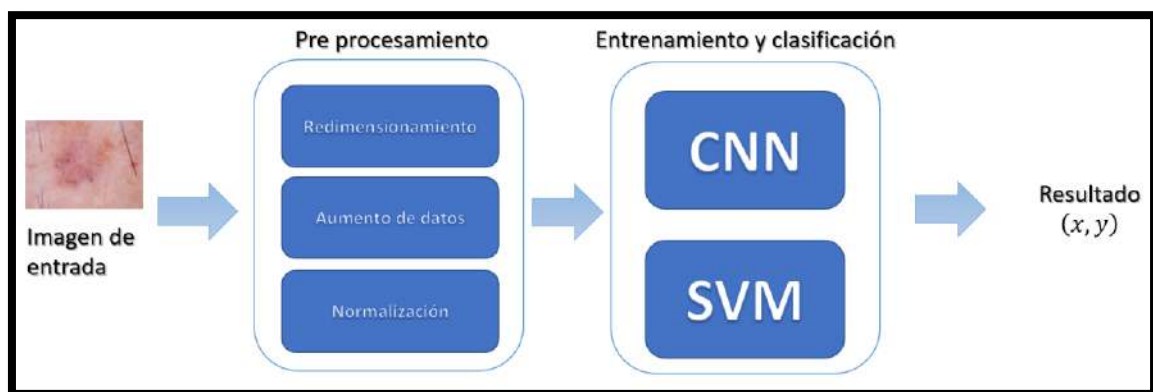


Figura 15 Modelo para clasificar el CBC en alto nivel

4.3.1. Pre-procesamiento de la imagen

Para el pre procesamiento de las imágenes, se las redimensionó a un tamaño de 64 x 64 píxeles, luego si la imagen era de carcinoma basocelular (bcc), se le realizó a la imagen la técnica de regularización de aumento de datos, la cual se utilizó con el fin de disminuir el sobreajuste y tener mayor variabilidad de imágenes. Se aplicaron rotaciones aleatorias, zooms aleatorios, volteos verticales y volteos horizontales para compensar el desbalanceo de datos.

- Rotaciones aleatorias: Se aplicaron 3 rotaciones aleatorias para cada imagen de carcinoma basocelular. (Ver figura 16)

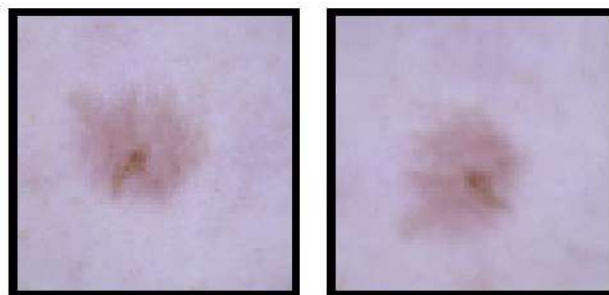


Figura 16 Imagen dermatoscópica con rotación aleatoria. Elaboración propia.

- Zoom aleatorio: Se aplicaron 3 zoom aleatorios para cada imagen de carcinoma basocelular. (Ver figura 17)



Figura 17 Imagen dermatoscópica con zoom aleatorio. Elaboración propia.

- Volteo horizontal: Se aplicó un volteo horizontal para cada imagen de carcinoma basocelular. (Ver figura 18)

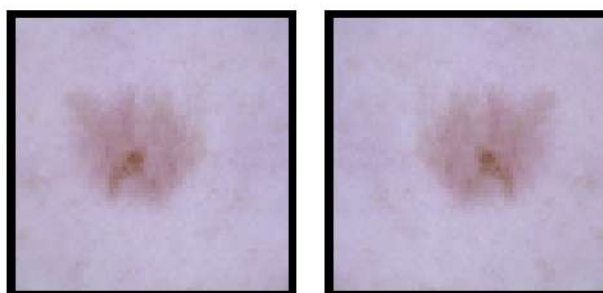


Figura 18 Imagen dermatoscópica con volteo horizontal. Elaboración propia.

- Volteo vertical: Se aplicó un volteo vertical para cada imagen de carcinoma basocelular. (Ver figura 19)



Figura 19 Imagen dermatoscópica con volteo vertical. Elaboración propia.

Luego de aplicar la técnica de aumento de datos se contó con 4,581 imágenes de carcinoma basocelular (bcc) y 4,499 imágenes que no son carcinoma basocelular (nobcc), mejorando así el desbalance de datos de datos (con una diferencia de 0.9%) para así evitar que el resultado final se sesgue al realizar la clasificación. (Ver gráfico 7)

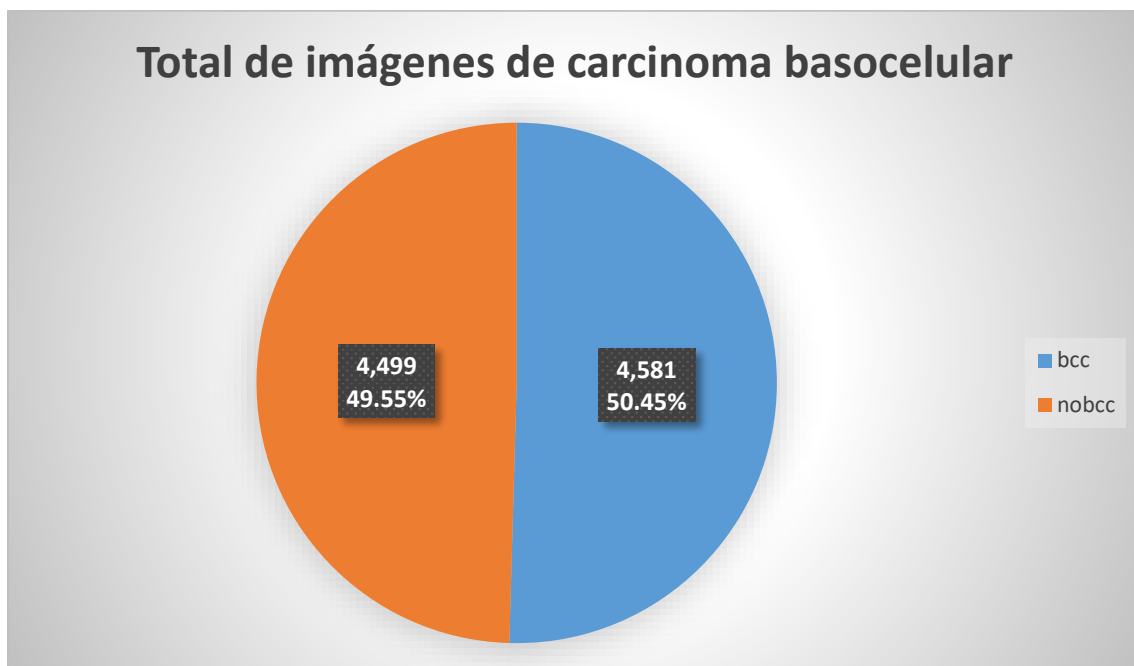


Gráfico 7 Total de imágenes de carcinoma basocelular. Elaboración propia.

Finalmente, a las imágenes se las normalizó para que los valores de cada pixel de la imagen varíen entre 0 y 1, ya que, las entradas con valores grandes pueden ralentizar el

proceso de aprendizaje. En la figura 20 se presenta el diagrama de flujo el cual representa el pre procesamiento de datos a más detalle.

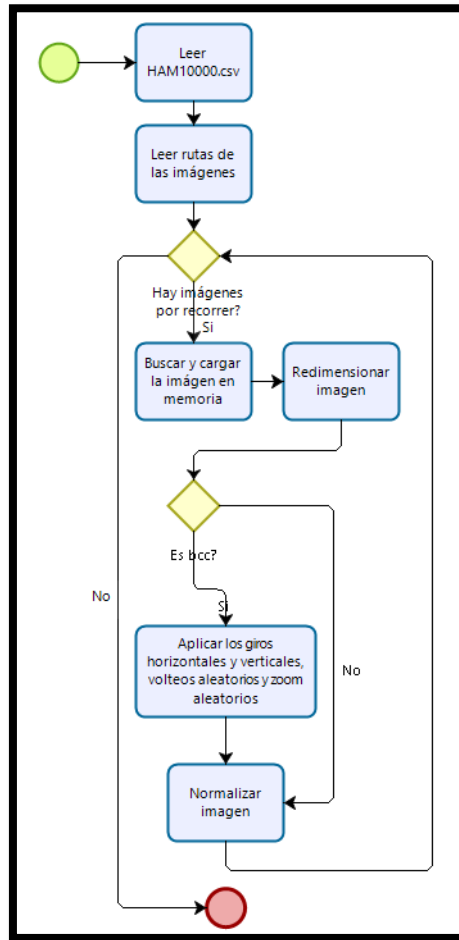


Figura 20 Diagrama de flujo del pre procesado de las imágenes, elaboración propia.

4.3.2. Entrenamiento y clasificación con SVM

Luego que las imágenes fueron pre procesadas, están listas para pasar por la red neuronal convolucional para el entrenamiento. La configuración del modelo de red neuronal convolucional propuesto por (Angeles Rojas et al., 2021) se detallan en las siguientes tablas. La tabla 5 muestra la configuración de la red neuronal convolucional, la cual es la encargada de extraer las características que permiten clasificar el carcinoma basocelular.

Tabla 5 Configuración de la red neuronal convolucional

Capas	Hiperparámetros
Capa de convolución	<ul style="list-style-type: none"> • 32 filtros de 3x3 • Función de activación: ReLU
Capa de max pooling	<ul style="list-style-type: none"> • Filtro 2x2
Capa Dropout	<ul style="list-style-type: none"> • Ratio: 0.1
Capa de convolución	<ul style="list-style-type: none"> • 64 filtros de 3x3 • Función de activación: ReLU
Capa de max pooling	<ul style="list-style-type: none"> • Filtro 2x2
Capa de convolución	<ul style="list-style-type: none"> • 64 filtros de 3x3 • Función de activación: ReLU
Capa de max pooling	<ul style="list-style-type: none"> • Filtro 2x2
Capa Dropout	<ul style="list-style-type: none"> • Ratio: 0.1
Capa de convolución	<ul style="list-style-type: none"> • 128 filtros de 3x3 • Función de activación: ReLU
Capa de max pooling	<ul style="list-style-type: none"> • Filtro 2x2
Capa Dropout	<ul style="list-style-type: none"> • Ratio: 0.1
Capa Flatten	<ul style="list-style-type: none"> • Cantidad: 1 capa

La tabla 6 muestra la capa totalmente conectada, la cual es la encargada de realizar la clasificación. La capa final se configura para que tenga el mismo comportamiento que un Support Vector Machine con la función de activación lineal y función de pérdidas L1-SVM, pero la salida final tiene 2 neuronas en vez de una para obtener los resultados por separado. Este resultado final es representado en un vector (x,y) donde el primer valor “x” devuelve el resultado de “si presenta carcinoma basocelular” y el segundo valor “y” devuelve el resultado de “no es carcinoma basocelular”.

Tabla 6 Capas totalmente conectadas

Capas	Hiperparámetros
-------	-----------------

Capa totalmente conectada	<ul style="list-style-type: none"> • 512 neuronas • Función de activación: ReLU
Capa Dropout	<ul style="list-style-type: none"> • Tasa: 0.5
Capa totalmente conectada	<ul style="list-style-type: none"> • 256 neuronas • Función de activación: ReLU
Capa Dropout	<ul style="list-style-type: none"> • Tasa: 0.5
Capa densa	<ul style="list-style-type: none"> • 2 neuronas • Función de activación: Lineal • Función de pérdidas: L1-SVM

4.5. Metodología de desarrollo e implementación del sistema que detecta el carcinoma basocelular

Para la construcción del aplicativo, motivo de esta investigación, se planteó utilizar la metodología ágil de Scrum, en el cual se define un product backlog compuesto de una serie de sprints que contienen todas las tareas por realizar. El sprint es un periodo de tiempo que va de una a cuatro semanas aproximadamente en donde se desarrolla el producto. Para el desarrollo del sistema que predice el carcinoma basocelular se definieron una serie de historias de usuarios detalladas a continuación.

4.5.1. Historias de usuario

En las historias de usuario se especificaron los requisitos para la construcción del sistema que predice el CBC. (Ver tabla 7 y 8)

Tabla 7 Historia de usuario 01, entrenar el modelo que predice el CBC.

HISTORIA DE USUARIO	
Número: 1	Usuario: Médico
Historia: Diagnosticar el carcinoma basocelular	
Prioridad: Alta	
Tiempo estimado: 2 días	

Descripción: Se implementará y entrenará el modelo de red neuronal convolucional más Support Vector Machine para guardar nuestro modelo que será implementado en el sistema.

Tareas:

- Subir las imágenes de HAM10000 a Google Drive
- Cargar y pre-procesar las imágenes.
- Implementar y entrenar el modelo.
- Guardar el modelo
- Desarrollar el backend.
- Ejecutar pruebas
- Desplegar el backend.

Condiciones:

- Para la construcción de la red neuronal convolucional con Support Vector Machine debe usarse el modelo propuesto por (Angeles Rojas et al., 2021).
- Para la construcción del modelo se utilizó Python 3 y para implementar la red neuronal convolucional se usó la librería Keras.
- El API debe utilizar el framework Flask.

Tabla 8 Historia de usuario 02, subir una imagen dermatoscópica

HISTORIA DE USUARIO	
Número: 2	Usuario: Médico
Historia: Subir una imagen dermatoscópica	
Prioridad: Media	
Tiempo estimado: 3 días	
Descripción: El médico subirá una imagen dermatoscópica al sistema del cual se obtendrá el diagnóstico.	
Tareas:	
<ul style="list-style-type: none"> • Desarrollar el frontend. • Ejecutar pruebas • Desplegar el sistema. 	

Condiciones:

- Las nuevas imágenes de entrada deben tener formato JPG o PNG.
- Se debe pre visualizar la imagen subida.
- El sistema debe utilizar la librería React.

4.5.2. Product Backlog

En el Product Backlog se lista las historias de usuario y cada historia de usuario fue asignado a un único sprint, por lo tanto, el desarrollo del sistema que diagnostica el CBC se ejecutó en un sprint de 4 días. (Ver tabla 9)

Tabla 9 Product backlog

Identificador de la HU	Sprint	Tiempo estimado	Prioridad
HU01	Sprint 1	2 días	Alta
HU02		3 días	Media

4.5.3. Sprints

4.5.3.1. Sprint 0

En este sprint se definieron ciertos preparativos que necesitaba el proyecto como el prototipo de la interfaz de usuario o el diagrama de arquitectura.

a) Prototipo

En la figura 21 se aprecia que el prototipo para predecir el CBC, se sube una imagen dermatoscópica con lesión cutánea al sistema para luego realizar la clasificación. Luego que la imagen fue enviada al servidor, en la parte inferior se muestra la imagen y el resultado de la clasificación.

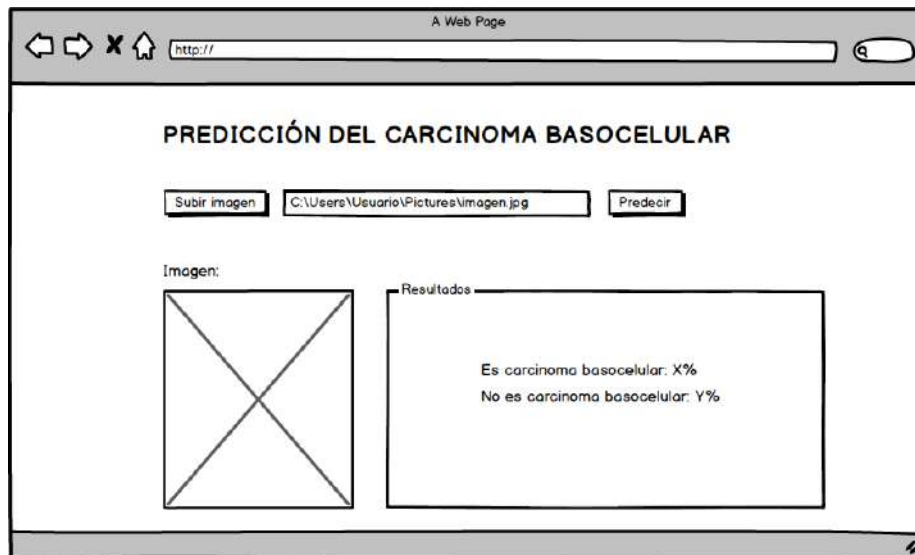


Figura 21 Prototipo del sistema. Elaboración propia.

b) Arquitectura

La figura 22 muestra la arquitectura del sistema, que está representada con el modelo cliente-servidor, puesto que el usuario se conectará por medio de un navegador web para subir su imagen dermatoscópica la cual será enviada al servidor web donde se procesa y regresa su respectivo resultado de si es CBC o no.

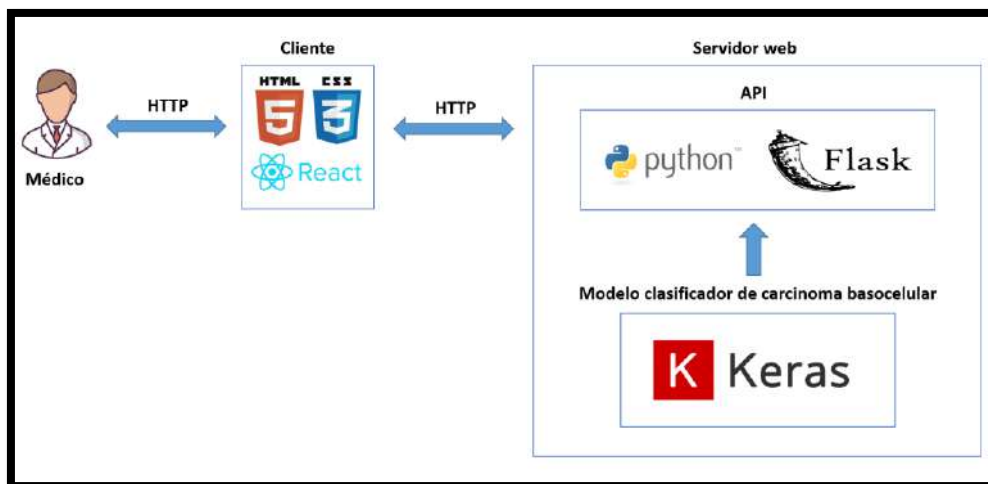


Figura 22 Arquitectura en alto nivel. Elaboración propia.

Podemos separar esta arquitectura en dos partes, la parte de cliente y la parte del servidor web.

- Cliente: Está construido utilizando las tecnologías de HTML, CSS y React 17. El cliente tendrá la responsabilidad de capturar la imagen dermatoscópica y mostrar

la imagen dermatoscópica en pantalla y mostrar el resultado del diagnóstico automático del carcinoma basocelular en pantalla al médico.

- **Servidor:** Está construido utilizando las tecnologías de Python 3.6 y Flask (versión 1.0.2), el cual se encarga de implementar el servidor web REST. Además, en el servidor web se carga el modelo híbrido clasificador del carcinoma basocelular que fue previamente entrenado utilizando la librería Keras (versión 2.6)

Ahora se presenta la figura 23 en el cual se implementa el patrón modelo vista controlador (MVC):

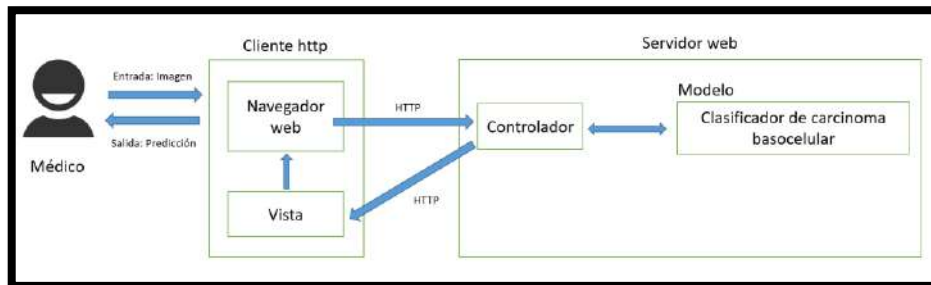


Figura 23 Representación de interacción de componentes. Elaboración propia.

- **Cliente http:** Este componente es el medio por el cual el usuario interactúa con el sistema, la cual puede ser accedida mediante cualquier navegador web actual.
- **Controlador:** Este componente es el encargado de ser el mediador entre la vista y el modelo. Su objetivo es capturar y procesar la imagen dermatoscópica además de comunicarse con el modelo el cual retornará una respuesta, esta respuesta será enviada a la vista por medio del controlador.
- **Modelo:** Este es el fichero que se genera después de hacer el entrenamiento y se encarga de realizar la clasificación. Hecha la clasificación envía la respuesta al controlador.
- **Vista:** Es el componente que se encarga de enviar la imagen dermatoscópica al controlador, que, a su vez recibe el resultado del diagnóstico del carcinoma basocelular que envía el controlador. También se encarga de mostrar la interfaz de usuario del lado del cliente.

En la Figura 24 se muestra el diagrama de despliegue. Está compuesto de dos componentes principales el navegador web y el servidor web. Al acceder al aplicativo mediante un navegador web, primero se ejecuta el componente index.html el cual renderiza la interfaz de usuario, en el momento que el usuario interactúa con el sistema se ejecuta el index.js el cual se encarga recibir los datos del usuario (en este caso la imagen dermatoscópica) y enviarlos por protocolo http al servidor web. El servidor web recibe los datos en formato JSON y son procesados por el componente model.py. Este componente le da formato a la imagen para que el fichero cnn_svm.h5, el cual contiene el modelo que diagnostica el carcinoma basocelular, pueda realizar la clasificación. Luego que se realizó la clasificación, el archivo model.py envía el resultado al navegador web.

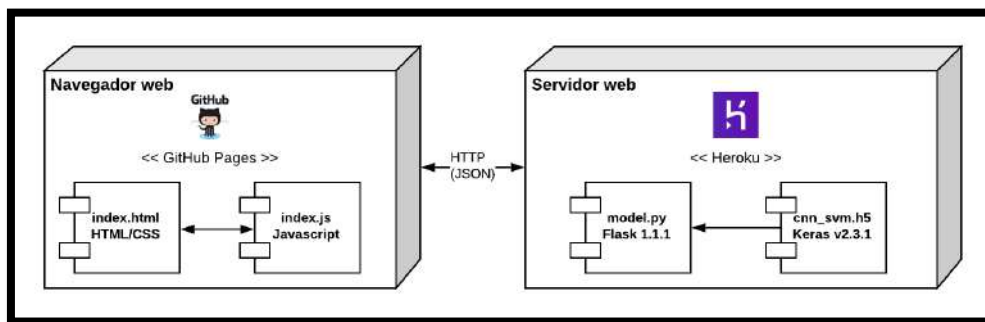


Figura 24 Diagrama de despliegue. Elaboración propia.

4.5.3.2. Sprint 1

El Sprint 1 está compuesto por el HU01 – Diagnosticar el carcinoma basocelular y el HU02 – Subir una imagen dermatoscópica.

a) Desarrollo de HU01 – Diagnosticar el carcinoma basocelular

Para desarrollar la primera historia de usuario se tomaron las siguientes acciones.

- **Tarea 1: Subir las imágenes de HAM10000 a Google Colab**

Para esta tarea se subieron las imágenes en dos carpetas, “HAM10000_images_part1” y “HAM10000_images_part2” a una carpeta de Google Drive, pesando ambas 2.57 GB y con un tiempo de subida aproximado de 5 horas. Adicionalmente se subió el archivo “HAM10000_metadata.csv” el cual contiene las etiquetas para cada una de las imágenes.

- **Tarea 2: Cargar y pre procesar las imágenes**

Antes de precargar las imágenes se modificó el archivo HAM10000_metadata.csv cambiando las etiquetas “akiec”, “bkl”, “df”, “nv”, “mel” y “vasc” por “nobcc”, esto se hace con el fin de crear los grupos para la clasificación binaria. Ya con esta modificación realizada se continua con la carga de imágenes.

Para cargar las imágenes en memoria, se lee el archivo HAM10000_metadata.csv buscando el nombre de la imagen y su etiqueta correspondiente, con estos datos se recorre las carpetas HAM10000_images_part1 y HAM10000_images_part2 buscando la imagen correspondiente.

Tabla 10 Vista de algunas celdas del HAM10000_metadata.csv

lesion_id	image_id	dx	dx_type	age	sex	localization
HAM_0000118	ISIC_0027419	nobcc	histo	80	male	scalp
HAM_0002730	ISIC_0025030	nobcc	histo	80	male	scalp
HAM_0000781	ISIC_0028155	bcc	histo	50	male	back
HAM_0003220	ISIC_0034093	bcc	histo	80	female	face
HAM_0007141	ISIC_0029230	bcc	histo	50	male	trunk

Cuando se encuentra la imagen, se carga en memoria y se redimensiona a una resolución de 64 x 64 píxeles, esto se repite iterativamente hasta terminar de recorrer todas las celdas del HAM100000_metadata.csv. Para tratar el desbalance se utiliza la herramienta de Skimage para realizar los volteos con giros horizontales y verticales, además se utiliza la función ImageDataGenerator de Keras para generar los volteos y zoom aleatorios solo para las imágenes con etiquetas bcc. Luego se separan los datos con 75% muestras de entrenamiento (training) y 25% muestras de prueba (testing), seguidamente a las muestras de entrenamiento se le vuelve a dividir en 75% de entrenamiento y 25% de validación, para consecuentemente utilizar la técnica de “one hot encoding” para codificar las etiquetas. Con el “one hot encoding” nuestras imágenes se etiquetarán como [1,0] y [0,1] y dado que vamos a utilizar una máquina de vectores de soporte para realizar la clasificación, cambiaremos los 0 por un -1, obteniendo nuestras etiquetas finalmente como [1,-1] y [-1,1].

- **Tarea 3: Entrenar e implementar el modelo**

Con los datos ya cargados en memoria se hace una comprobación de la dimensionalidad de los datos utilizando la función shape de la librería Numpy.

```
print(x_train.shape)
print(y_train.shape)
print(x_test.shape)
print(y_test.shape)
print(x_val.shape)
print(y_val.shape)

(5107, 64, 64, 3)
(5107, 2)
(2270, 64, 64, 3)
(2270, 2)
(1703, 64, 64, 3)
(1703, 2)
```

Figura 25 Comprobación de la dimensionalidad de los datos. Elaboración propia.

La figura 25 muestra las variables `x_train`, `x_test` y `x_val`, las cuales poseen las imágenes de entrenamiento, prueba y validación con 5107, 2270 y 1703 imágenes respectivamente. Estas cuentan con una dimensión de 64 x 64 con 3 canales de color. De la misma manera, para las variables `y_train`, `y_test` e `y_val` que en este caso son las etiquetas correspondientes de cada imagen de `x_train`, `x_test` y `x_val` con 5107, 2270 y 1703 etiquetas respectivamente, y el número 2 al lado de la cantidad de etiquetas representa las dos clases a clasificar.

A continuación, se define la arquitectura propuesta por (Angeles Rojas et al., 2021) utilizando la librería Keras. Se define la clase `Sequential` y en la cual se utiliza la función `Conv2D` para crear las capas convolucionales y se definen la cantidad de filtros, la dimensión de cada uno de estos y la función de activación. Luego se utiliza la función `MaxPool2D` para crear la capa convolucional y esto se repite sucesivamente hasta completar las capas de convolución definidas en el capítulo de aporte. También se aplica la función `Dropout` para aplicar este regularizador. Por último, se utiliza la función `Flatten` para realizar el flattening correspondiente y con esto se termina de implementar todas las capas encargadas de la extracción de características.

```

from keras.models import Model, Sequential, load_model
from keras.layers import Dense, Flatten, Conv2D, MaxPool2D, Dropout
model = Sequential()

# Capa convolución
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=(64,64,3)))
model.add(MaxPool2D((2, 2)))
model.add(Dropout(0.1))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPool2D((2, 2)))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPool2D((2, 2)))
model.add(Dropout(0.1))
model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(MaxPool2D((2, 2)))
model.add(Dropout(0.1))
model.add(Flatten())

# Capa totalmente conectada
model.add(Dense(512, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(256, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(2, use_bias=False, activation='linear', name='svm'))

```

Figura 26 Implementación de la arquitectura propuesta. Elaboración propia.

Para la capa totalmente conectada se utiliza la función Dense el cual sirve para definir la capa totalmente conectada, con 512, 256 y 2 neuronas con sus respectivas funciones de activación. (Ver figura 26)

Luego se define la función de pérdidas L1-SVM, para implementarla como función de pérdidas en la última capa. Y finalmente se ajustan los parámetros como el número de épocas, el tamaño de lote (batch size) y el optimizador.

- **Tarea 4: Obtener los resultados**

Para obtener los resultados se utilizó la librería de Matplotlib, el cual permite dibujar la gráfica de entrenamiento y pérdidas, las cuales nos permitió visualizar como avanzaban los porcentajes de exactitud según va aumentando el número de épocas. También se utilizó la librería scikit-learn para obtener la matriz de confusión, las métricas de exactitud, precisión, recall y f1score.

- **Tarea 5: Desarrollar el backend**

En esta tarea se creó el API con la librería Flask, el cual es un microframework de Python que permite crear servicios web. En esta API se recibe la imagen enviada desde el navegador web, se decodifica y redimensiona a 64x64 píxeles, luego se

normaliza para finalmente pasar al modelo que realiza la clasificación. Luego que se realiza la clasificación, se aplica una función sobre el resultado para obtener una respuesta más entendible para el usuario, el cual se representa como un porcentaje de que tanto por ciento es carcinoma y que otro tanto por ciento es no carcinoma. Finalmente, el servidor envía la respuesta si la imagen tiene CBC o no en formato JSON.

- **Tarea 6: Ejecutar pruebas**

En la tabla 11 se presenta el plan de pruebas para el HU01. Para ver las pruebas unitarias, consultar la sección de Anexos.

Tabla 11 Resumen del plan de pruebas del Sprint 1.

Caso de prueba	Efecto deseado	Estado	Observación
Lectura de las imágenes en memoria	Que las imágenes del HAM10000 se carguen en memoria con su etiqueta correspondiente.	OK	Ninguna
Pre visualizar las imágenes antes de entrenar la red neuronal convolucional	Imprimir las imágenes en pantalla para ver su integridad.	OK	Las imágenes se ven en una resolución de 64x64 píxeles
Revisar la dimensionalidad de las imágenes antes de entrenar la red neuronal convolucional	Revisar que la dimensionalidad de las imágenes estén integras	OK	Ninguna
Comprobar el modelo híbrido	Al utilizar la función summary() de Keras, la arquitectura construida debe reflejar lo propuesto en el capítulo de aporte.	OK	Ninguna

Verificar que el archivo que llega al backend sea del formato correcto	Que el archivo que llegue al backend tenga el formato .jpg o .png	OK	Se implementó la prueba unitaria
Verificar que la normalización de la imagen sea correcta	Que la imagen sea normalizada para que sea el input a la red neuronal convolucional	OK	Se implementó la prueba unitaria

- **Tarea 7: Desplegar el backend**

El backend se desplegó en Heroku, el cual es una plataforma como servicio (PaaS) que permite a los desarrolladores construir, ejecutar y operar aplicaciones en la nube. Para esto se crearon tres archivos de configuración llamados Procfile, Requeriments y runtime. Procfile es donde se establecen algunos parámetros para el despliegue en Heroku. El archivo Requeriments es donde se listan todas las librerías a instalar en el servidor web y runtime donde se establece la versión de Python que se va a utilizar, en este caso se configuró con la versión 3.7.4. Con los archivos ya configurados, el proyecto se subió a un repositorio de Github. Luego se creó un nuevo proyecto en Heroku, el cual se sincronizó con nuestro repositorio en Github. Hecho esto el servidor ya está listo para ejecutar las peticiones.

b) Desarrollo de HU02 – Subir una imagen dermatoscópica

Para desarrollar la segunda historia de usuario se tomaron las siguientes acciones.

- **Tarea 1: Desarrollar el frontend**

En esta tarea, se escribe el código html, css y javascript (para el cual se usa la librería React), para construir la interfaz de usuario del aplicativo que predice el carcinoma basocelular. Esta tarea se divide en cuatro funcionalidades:

- Leer la imagen dermatoscópica ingresada.
- Mostrar la imagen en pantalla
- Enviar la imagen al servidor
- Recibir y mostrar el resultado.

- **Tarea 2: Ejecutar pruebas**

En la tabla 12 se presenta el resumen de las pruebas realizadas en el Sprint 2. Para ver las pruebas unitarias y funcionales, consultar la sección de Anexos.

Tabla 12 Resumen del plan de pruebas del Sprint 2

Caso de prueba	Efecto deseado	Estado	Observación
Cargar la imagen	Que las imágenes cargadas sean tanto .jpg como .png	OK	Se ejecutó una prueba unitaria
Convertir la imagen a base64	Que las imágenes tengan el formato base64	OK	Se ejecutó una prueba unitaria
Que renderize el componente para cargar la imagen	El componente para cargar la imagen debe aparecer en la pantalla	OK	Se ejecutó la prueba funcional
Que renderize la imagen	Se debe mostrar la imagen que sube el usuario en pantalla	OK	Se ejecutó la prueba funcional
Que renderize el resultado de la clasificación	Debe mostrar el resultado de la clasificación en la pantalla, el cual llega desde el servidor.	OK	Se ejecutó la prueba funcional

- **Tarea 3: Desplegar el sistema**

El frontend se despliega en GitHub Pages, el cual permite alojar nuestro proyecto web y proporciona un dominio gratis, además tiene una fácil integración con GitHub Actions, el cual es una herramienta para automatizar flujos de CI/CD (Continuous integration / Continuous deployment) y nos permite desplegar fácilmente nuestro sistema.

Primero se instaló el paquete de gh-pages en el proyecto, con el que se prepara y configura el proyecto para que pueda ser reconocido por GitHub Pages, luego se creó un nuevo repositorio en GitHub, donde se subieron nuestros archivos que corresponden al frontend. Posteriormente se creó un archivo YAML donde se realizó la configuración del CI/CD (consultar la sección de Anexos), gracias a esto se verifica que cualquier cambio en el código no rompa el funcionamiento del mismo, puesto que, se ejecutan las pruebas unitarias y funcionales automáticamente (CI) y se evita subir manualmente cualquier cambio al servidor y cada vez que se detecte un cambio en la rama principal, se hace un despliegue automático (CD). Finalmente se hace una configuración de las variables de entorno en el panel de control de GitHub y el frontend está desplegado.

4.5.4. Sistema

Al finalizar el sprint tenemos el sistema que diagnostica el carcinoma basocelular, la cual se observa en la siguiente figura

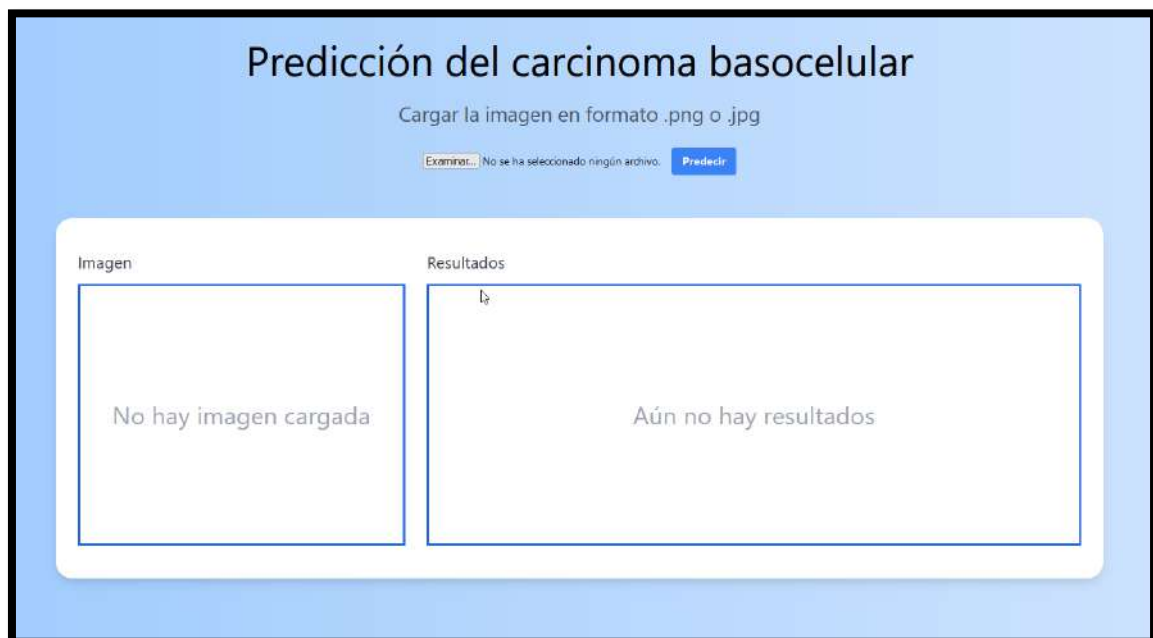


Figura 27 Captura de pantalla del sistema

Ahora se pasa a explicar el funcionamiento de la aplicación.

1. Primero se carga la imagen dermatoscópica en el sistema presionando el botón examinar.

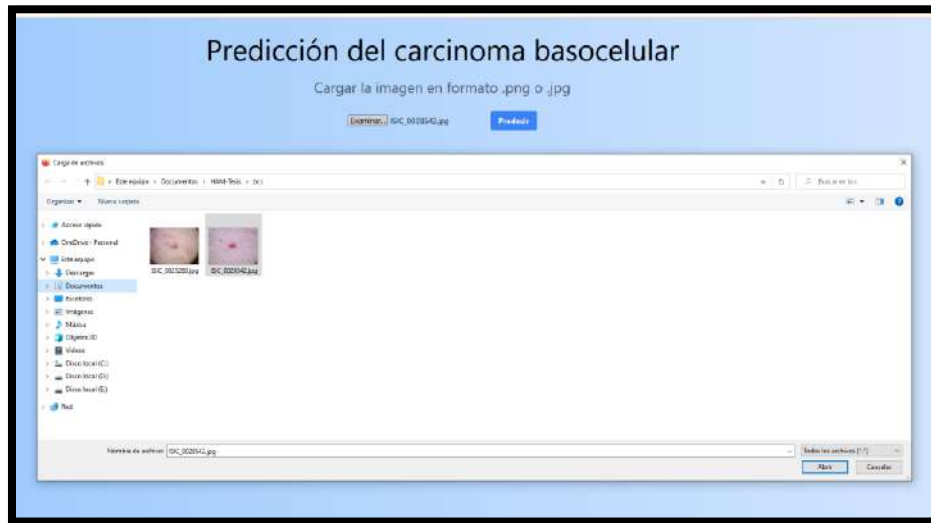


Figura 28 Captura de pantalla de la selección de una imagen dermatoscópica

2. Luego se mostrará la pre visualización de la imagen dermatoscópica en la pantalla



Figura 29 Captura de pantalla de la imagen cargada en el sistema

3. Para realizar la clasificación, se presiona sobre el botón predecir para obtener el resultado.



Figura 30 Captura de pantalla de la clasificación realizada

Para las capturas del sistema se utilizaron unas imágenes del dataset HAM10000 que nunca fueron utilizadas en el entrenamiento, la validación o pruebas del modelo. A modo de ejemplo se muestra los resultados de 4 imágenes distintas que fueron evaluadas por el sistema (ver tabla 13). Estos resultados pueden ser consultados en la sección de Anexos.

Tabla 13 Clasificación de imágenes dermatoscópicas utilizando el sistema

Imagen	Etiqueta	Clasificación: Es bcc	Clasificación: No es bcc	Resultado
ISIC_0028542	bcc	76%	24%	Correcto
ISIC_0025260	bcc	77%	23%	Correcto
ISIC_0032224	nobcc	0%	100%	Correcto
ISIC_0032854	nobcc	24%	76%	Correcto

CAPÍTULO 5: RESULTADOS Y DISCUSIÓN

5.1. Resultados

Como se menciona en el capítulo de aporte, las imágenes utilizadas para el entrenamiento, validación y pruebas de la red neuronal fueron proveídas por el dataset HAM10000. Del dataset se tomó la mitad de imágenes, con un total de 5,008 imágenes dermatoscópicas. Luego se aplicó la técnica de regularización de aumento de datos llegando a tener 9,080 imágenes. Para la división de datos, se tomó el 25% del total de las imágenes para los datos de prueba (para evaluar el modelo), contando con un total de 2,270 imágenes de prueba. Luego de las 6810 imágenes restantes, se tomó el 25% para las imágenes de validación (para evaluar el desempeño del entrenamiento), obteniendo 1,703 imágenes de validación. Las imágenes restantes se tomaron como los datos de entrenamiento (usado para que el modelo aprenda y afine sus parámetros), obteniendo un total de 5,107 imágenes de entrenamiento (ver tabla 14). Para la evaluación del modelo se usaron las métricas de exactitud, precisión, recall y f1-score.

Tabla 14 Distribución del conjunto de datos

Proceso	Número de imágenes
Entrenamiento	5,107
Evaluación	1,703
Prueba	2,270
Total	9,080

Para entrenar el modelo híbrido de red neuronal convolucional con Support Vector Machine (CNN-SVM en adelante) propuesto por (Angeles Rojas et al., 2021) que clasifica el carcinoma basocelular se utilizó la siguiente configuración:

- Épocas: 150
- Batch size (Tamaño de lote): 80

- Optimizador: Adam (learning rate: 0.001)
- Función de activación en la última capa: Lineal
- Función de pérdidas: L1-SVM

5.1.1. Resultados y evaluación del modelo híbrido (CNN+SVM) para clasificar el CBC

Para el modelo híbrido (CNN-SVM) se evaluó el entrenamiento del modelo, para tal fin se utilizaron los datos de entrenamiento y los datos de validación para analizar la exactitud y las pérdidas alcanzadas. Estas se midieron con una gráfica de número de épocas vs la exactitud y otra gráfica de número de épocas vs las pérdidas, de las cuales el eje horizontal representa el número de épocas y el eje vertical representa la exactitud para el gráfico 8 y las pérdidas para el gráfico 9.

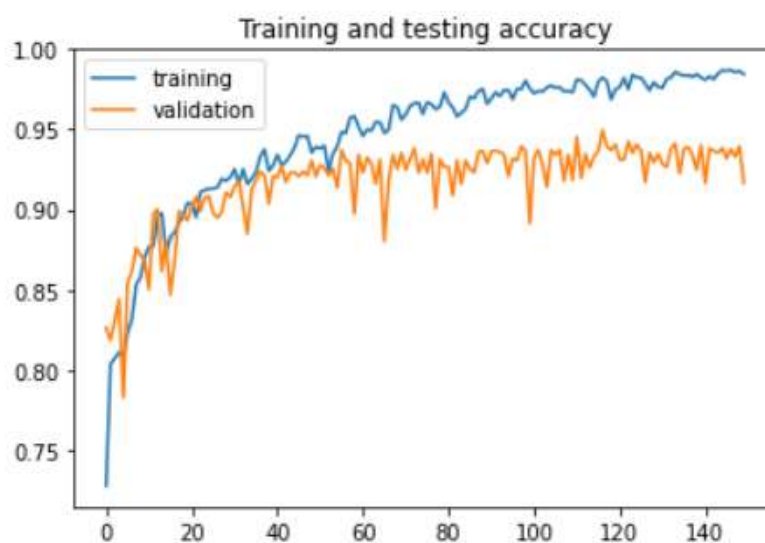


Gráfico 8 Evaluación de la exactitud para el modelo híbrido.

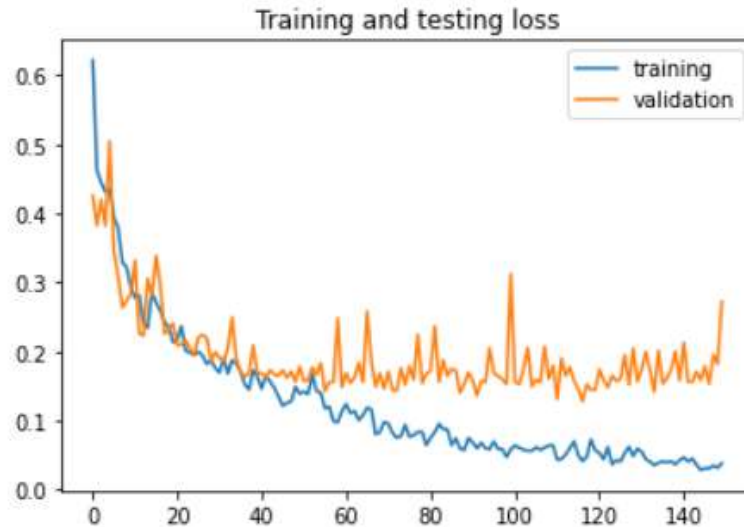


Gráfico 9 Evaluación de la pérdida para el modelo híbrido

Como se puede observar en la gráfica 8, al inicio del entrenamiento la exactitud de los datos de validación (representada por la línea naranja) fluctúa entre 80% y 90% desde la época 1 hasta la 40, luego la exactitud se va estabilizando y aumentando gradualmente mientras avanza el número de épocas, hasta llegar a la época 100 con 93% de exactitud promedio aproximadamente. Del mismo modo para el gráfico 9, las pérdidas de los datos de validación tienden a fluctuar hasta la época 40 y luego las pérdidas poco a poco se van estabilizando mientras avanza el número de épocas. Estas gráficas demuestran un entrenamiento aceptable del modelo, puesto que alcanza un 92% a 93% en promedio, pero que está tendiendo al sobreajuste.

Para estimar el posible sobreajuste se evalúa el modelo híbrido utilizando los 2,270 datos de pruebas o testing para poder calcular las métricas de exactitud, precisión, recall y f1-score. Estas métricas son obtenidas de la matriz de confusión (Ver tabla 15), en donde obtenemos 1016 Verdaderos positivos (VP), 1062 Verdaderos negativos (VN), 133 Falsos negativos (FN) y 59 Falsos positivos (FP).

Tabla 15 Matriz de confusión del modelo híbrido

Clasificación del CBC		Clasificación	
		CBC	No CBC
Observación	CBC	1016	133
	No CBC	59	1062

De los VP, VN, FN y FP podemos obtener las métricas de exactitud, precisión, recall y F1-Score, presentadas en la tabla 16.

Tabla 16 Métricas del modelo híbrido

	Precisión	Recall	F1-score	Exactitud
Promedio	94.511628 %	88.424717%	91.366906 %	91.541850 %

La exactitud fue de 91.54% y lo podemos entender como la cantidad de predicciones correctas frente al total, la precisión fue de 94.51%, el cual se entiende como la relación entre las observaciones positivas predichas correctamente y el total de observaciones positivas (verdaderos positivos y falsos positivos), en otras palabras, la cantidad de predicciones positivas correctas. Con la métrica de recall se obtuvo 88.42%, que es la proporción de observaciones positivas predichas correctamente a todas las observaciones de la clase real, también se puede entender como el porcentaje de casos positivos detectados y para F1 score se obtuvo 91.36% que es el promedio ponderado de precisión y sensibilidad. Por lo tanto, este puntaje tiene en cuenta tanto los falsos positivos como los falsos negativos. Intuitivamente, no es tan fácil de entender como la precisión, pero F1 suele ser más útil que la precisión, especialmente si tiene una distribución de clase desigual, como en nuestro conjunto de datos existe un pequeño desbalance entre las clases “Si es carcinoma” y “no es carcinoma”, se toma en consideración esta métrica.

Entonces los resultados obtenidos del modelo híbrido, utilizando las métricas de precisión, recall, f1-score y exactitud obtuvimos 94.51%, 88.42%, 91.36% y 91.54% respectivamente.

5.2. Discusión

Con los resultados obtenidos, ahora se realiza un análisis frente a otros trabajos que también clasificaron el CBC. La tabla 17 muestra un resumen de todas las investigaciones, con la técnica utilizada y los resultados alcanzados.

Tabla 17 Comparativa de resultados con otras investigaciones que clasificaron el CBC

Autor	Técnica	Dataset	Exactitud	Precisión	Recall	F1-score
D. Mandache et. al	CNN	Dataset propio	95.93%	-	95.2%	-
H. Huang et. al	CNN	Dataset propio	93%	-	77%	-
E. V. Putten et. al	ResNet	Skin Lesion Analysis Towards Melanoma Detection ISBI 2016	93%	-	97%	-
P. Kharazmi et. al	Random Forest	Dataset propio	-	82.7%	82.4%	-
Esta investigación	CNN + SVM	HAM10000	91.54%	94.51%	88.42 %	91.36%

- (Mandache et al., 2018) utilizó un dataset propio que consistían en imágenes de tomografía de coherencia óptica (OCT), (Huang et al., 2018) utilizó un dataset de imágenes clínicas, entiéndase como imágenes de la lesión que se puede observar a simple vista, (Putten et al., 2018) usó el dataset Skin Lesion Analysis Towards Melanoma Detection del ISBI 2016, (Kharazmi et al., 2016) usó un dataset de imágenes dermatoscópicas conformado por diferentes fuentes de datos, entre los que se encontraban “Atlas of dermoscopy”, la universidad de Missouri y “Vancouver Skin Care Centre”. Nosotros en este trabajo utilizamos el dataset HAM10000 el cual contiene imágenes dermatoscópicas, además la naturaleza de estas imágenes es muy diferentes con respecto a (Mandache et al., 2018) y (Huang et al., 2018), puesto que el método para obtener sus imágenes es muy distinto al nuestro.
- (Mandache et al., 2018) usó una red neuronal convolucional a la cual adaptaron ideas de la arquitectura VGG, (Huang et al., 2018) usó una red neuronal convolucional clásica, (Putten et al., 2018) usó la red neuronal convolucional

ResNet y (Kharazmi et al., 2016) usó el algoritmo Random Forest. Ninguna de estas propuestas planteó utilizar un modelo híbrido (CNN+SVM) para clasificar el CBC como lo hicimos en esta investigación. Además, ninguna de estas investigaciones propuso desarrollar un aplicativo como herramienta para clasificar el CBC.

- (Mandache et al., 2018) alcanzó los resultados de 95.93% de exactitud y 95.2% de recall utilizando su propio dataset, (Huang et al., 2018) obtuvo 93% de exactitud y 77% de recall usando su propio dataset, (Putten et al., 2018) alcanzó los resultados de 93 % de exactitud y 97 % de recall utilizando el dataset Skin Lesion Analysis Towards Melanoma Detection ISBI 2016, (Kharazmi et al., 2016) obtuvo 82.7% de precisión y 82.4% de recall utilizando su dataset propio. Nosotros en esta investigación alcanzamos utilizando las métricas de precisión, recall, f1-score y exactitud obtuvimos 94.51%, 88.42%, 91.36% y 91.54% respectivamente. Además frente a (Huang et al., 2018) logramos superarlos en la métrica de recall, puesto que ellos alcanzaron 77% y nosotros 88.42%, también frente a (Kharazmi et al., 2016) se logró superar sus resultados, puesto que ellos alcanzaron 82.7% y 82.4% con las métricas de precisión y sensibilidad respectivamente y nosotros logramos obtener 94.51% y 88.42% con las métricas de precisión y sensibilidad respectivamente. A otras investigaciones no se logró superar sus resultados como a (Mandache et al., 2018) y (Huang et al., 2018) pero el método de obtención es muy distinto con respecto al nuestro, puesto que nuestras imágenes se obtienen mediante un dermatoscopio, y con respecto a (Putten et al., 2018) el tipo de imágenes son las mismas pero el dataset es distinto.

Conclusiones

En este trabajo se propuso un sistema de detección del carcinoma basocelular implementando un modelo híbrido de redes neuronales convolucional y Support Vector Machine y utilizando el dataset HAM10000, el cual clasifica eficazmente el carcinoma basocelular. La solución propuesta permite al médico subir una imagen dermatoscópica al sistema para obtener un diagnóstico automático del carcinoma basocelular, el cual puede ayudar al médico dar un veredicto final, especialmente a los dermatólogos con poca experiencia.

Conclusiones Específicas

- **Diseñar el modelo de detección de CBC utilizando redes neuronales convolucionales y máquina de vectores de soporte que clasifique el CBC**

Para este fin se utilizó el trabajo de (Angeles Rojas et al., 2021), el cual obtuvo muy buenos resultados en la clasificación del carcinoma basocelular. Debido a esto se optó por usar su diseño híbrido de red neuronal convolucional con Support Vector Machine para clasificar el carcinoma basocelular.

- **Construir el sistema que diagnostica el carcinoma basocelular**

Para la construcción del sistema se utilizó la metodología SCRUM, debido a que la aplicación constaba de pocos requisitos y no se necesitaban detallar o realizar un análisis exhaustivo al sistema, debido a esto se optó por esta metodología ágil que constó de un solo sprint con dos historias de usuarios.

- **Validar el modelo que diagnostica el carcinoma basocelular**

Para evaluar el modelo de red neuronal convolucional con Support Vector Machine para clasificar el carcinoma basocelular se usaron las imágenes del dataset HAM10000. Del dataset se tomó la mitad de imágenes, con un total de 5,008 imágenes dermatoscópicas. Luego se aplicó la técnica de aumento de datos llegando a tener 9,080 imágenes. Las imágenes se dividieron en 2,270 imágenes de prueba, 1,703 imágenes de validación y 5,107 imágenes de entrenamiento. Se analizaron las gráficas de exactitud y pérdidas utilizando los datos de entrenamiento y validación. Además, se utilizaron los datos de prueba para

calcular las métricas de exactitud, precisión, recall y F1-Score. Los resultados obtenidos demostraron que este modelo alcanza buenos resultados para las métricas de precisión, recall, f1-score y exactitud obtenemos 94.51%, 88.42%, 91.36% y 91.54% respectivamente, asegurando así que pueda ser utilizado para diagnosticar nuevos casos de carcinoma basocelular.

Limitaciones

Para el entrenamiento de la red neuronal convolucional se utilizó la plataforma de Google Colab, el cual permite ejecutar código y programar en Python en el navegador. Se tomó esta opción debido a que da un GPU gratuito el cual sirve para entrenar la red neuronal rápidamente. Al ser una herramienta gratuita también tiene sus limitaciones, como un número máximo limitado de horas de uso o limitación en la capacidad de ejecución del GPU. Al optar por esta opción no se puede realizar un entrenamiento demandante, puesto que tenemos limitación de recursos.

Trabajos futuros

Este sistema que clasifica con precisión el carcinoma basocelular podrá ser utilizado por médicos dermatólogos para poder diagnosticar el carcinoma basocelular y podría ser mejorado al incluir los siguientes puntos:

- Se podría probar diferentes técnicas de segmentación de imagen, las cuales permitan aumentar la exactitud de la clasificación.
- Se puede adaptar este modelo híbrido CNN+SVM para clasificar otros tipos de carcinoma.
- Se puede combinar el modelo híbrido con novedosas arquitecturas de redes neuronales convolucionales.
- Integrar un dermatoscopio con una aplicación móvil, la cual permita realizar un diagnóstico automático.

Bibliografía

- Agarap, A. F. (2019). An Architecture Combining Convolutional Neural Network (CNN) and Support Vector Machine (SVM) for Image Classification. *arXiv:1712.03541 [cs, stat]*. <http://arxiv.org/abs/1712.03541>
- Ahmed, T., Wahid, Md. F., & Hasan, Md. J. (2019). Combining Deep Convolutional Neural Network with Support Vector Machine to Classify Microscopic Bacteria Images. *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 1-5. <https://doi.org/10.1109/ECACE.2019.8679397>
- Angeles Rojas, J. A., Calderón Vilca, H. D., Tumi Figueroa, E. N., Cuadros Ramos, K. J., Matos Manguinuri, S. S., Calderón Vilca, E. F., Ángeles Rojas, J. A., Calderón Vilca, H. D., Tumi Figueroa, E. N., Cuadros Ramos, K. J., Matos Manguinuri, S. S., & Calderón Vilca, E. F. (2021). Hybrid Model of Convolutional Neural Network and Support Vector Machine to Classify Basal Cell Carcinoma. *Computación y Sistemas*, 25(1), 83-95. <https://doi.org/10.13053/cys-25-1-3431>
- Barata, C., Ruela, M., Francisco, M., Marques, J., & Mendonça, T. (2013). Two Systems for the Detection of Melanomas in Dermoscopy Images Using Texture and Color Features. *IEEE Systems Journal*, 8. <https://doi.org/10.1109/JSYST.2013.2271540>
- Barrios, J. (2019, julio 26). La matriz de confusión y sus métricas – Inteligencia Artificial –. *Juan Barrios*. <https://www.juanbarrios.com/la-matriz-de-confusion-y-sus-metricas/>
- Binder, M., Puespoeck-Schwarz, M., Steiner, A., Kittler, H., Muellner, M., Wolff, K., & Pehamberger, H. (1997). Epiluminescence microscopy of small pigmented skin lesions: Short-term formal training improves the diagnostic performance of

- dermatologists. *Journal of the American Academy of Dermatology*, 36(2 Pt 1), 197-202. [https://doi.org/10.1016/s0190-9622\(97\)70280-9](https://doi.org/10.1016/s0190-9622(97)70280-9)
- Borja-Robalino, R., Monleon-Getino, A., & Benedé, J. (2020). *Estandarización de Métricas de Rendimiento para Clasificadores Machine y Deep Learning*.
- Bozorgtabar, B., Sedai, S., Roy, P. K., & Garnavi, R. (2017). Skin lesion segmentation using deep convolution networks guided by local unsupervised learning. *IBM Journal of Research and Development*, 61(4/5), 6:1-6:8. <https://doi.org/10.1147/JRD.2017.2708283>
- Cajina-Vázquez, A. (2012). Carcinoma Basocelular. *Revista Médica de Costa Rica y Centroamérica*, 69(603), 325-329.
- Cancer today*. (s. f.). Recuperado 13 de noviembre de 2021, de <http://gco.iarc.fr/today/home>
- Cheng, B., Joe Stanley, R., Stoecker, W. V., Stricklin, S. M., Hinton, K. A., Nguyen, T. K., Rader, R. K., Rabinovitz, H. S., Oliviero, M., & Moss, R. H. (2013). Analysis of clinical and dermoscopic features for basal cell carcinoma neural network classification. *Skin Research and Technology: Official Journal of International Society for Bioengineering and the Skin (ISBS) [and] International Society for Digital Imaging of Skin (ISDIS) [and] International Society for Skin Imaging (ISSI)*, 19(1), e217-222. <https://doi.org/10.1111/j.1600-0846.2012.00630.x>
- Codella, N., Nguyen, Q.-B., Pankanti, S., Gutman, D., Helba, B., Halpern, A., & Smith, J. R. (2016). Deep Learning Ensembles for Melanoma Recognition in Dermoscopy Images. *arXiv:1610.04662 [cs]*. <http://arxiv.org/abs/1610.04662>
- Darias Domínguez, C., & Garrido Celis, J. (2018). Carcinoma basocelular. Un reto actual para el dermatólogo. *Revista Médica Electrónica*, 40(1), 172-182.

Das, D. K., Koley, S., Chakraborty, C., & Maiti, A. K. (2014). Automated segmentation of Mitotic Cells for in vitro histological evaluation of oral squamous cell carcinoma. *2014 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 000354-000357. <https://doi.org/10.1109/ISSPIT.2014.7300614>

Estadísticas importantes sobre el cáncer de piel tipo melanoma. (s. f.). Recuperado 13 de noviembre de 2021, de <https://www.cancer.org/es/cancer/cancer-de-piel-tipo-melanoma/acerca/estadisticas-clave.html>

Fernández Martori, M., Cabrera Corrales, M., & García Gómez, R. (2014). Precisión del diagnóstico dermatoscópico para el carcinoma basocelular en el Servicio de Dermatología. Policlínico Colón-Este. 2012. *Gac. méd. espirit*, 1-12.

Ge, Y., Li, B., Zhao, Y., Guan, E., & Yan, W. (2018). Melanoma Segmentation and Classification in Clinical Images Using Deep Learning. *Proceedings of the 2018 10th International Conference on Machine Learning and Computing*, 252-256. <https://doi.org/10.1145/3195106.3195164>

Hagerty, J. R., Stanley, R. J., Almubarak, H. A., Lama, N., Kasmi, R., Guo, P., Drugge, R. J., Rabinovitz, H. S., Oliviero, M., & Stoecker, W. V. (2019). Deep Learning and Handcrafted Method Fusion: Higher Diagnostic Accuracy for Melanoma Dermoscopy Images. *IEEE Journal of Biomedical and Health Informatics*, 23(4), 1385-1391. <https://doi.org/10.1109/JBHI.2019.2891049>

Haghighi, S. N., Danyali, H., Helfroush, M. S., & Karami, M. H. (2020). A Deep Convolutional Neural Network for Melanoma Recognition in Dermoscopy Images. *2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)*, 453-456. <https://doi.org/10.1109/ICCKE50421.2020.9303684>

- He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]*. <http://arxiv.org/abs/1512.03385>
- Huang, H., Kharazmi, P., McLean, D. I., Lui, H., Wang, Z. J., & Lee, T. K. (2018). Automatic detection of translucency using a deep learning method from patches of clinical basal cell carcinoma images. *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 685-688. <https://doi.org/10.23919/APSIPA.2018.8659685>
- jiahao, W., Xingguang, J., Yuan, W., Luo, Z., & Yu, Z. (2021). Deep Neural Network for Melanoma Classification in Dermoscopic Images. *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, 666-669. <https://doi.org/10.1109/ICCECE51280.2021.9342158>
- Kharazmi, P., Lui, H., Wang, Z. J., & Lee, T. K. (2016). Automatic detection of basal cell carcinoma using vascular-extracted features from dermoscopy images. *2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, 1-4. <https://doi.org/10.1109/CCECE.2016.7726666>
- Kingma, D. P., & Ba, J. (2017). Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*. <http://arxiv.org/abs/1412.6980>
- Kittler, H., Pehamberger, H., Wolff, K., & Binder, M. (2002). Diagnostic accuracy of dermoscopy. *The Lancet. Oncology*, 3(3), 159-165. [https://doi.org/10.1016/s1470-2045\(02\)00679-4](https://doi.org/10.1016/s1470-2045(02)00679-4)
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 25. <https://papers.nips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html>

- Kurek, J. (2018). Hybrid approach towards the assessment of a drill condition using deep learning and the Support Vector Machine. *2018 22nd International Computer Science and Engineering Conference (ICSEC)*, 1-5. <https://doi.org/10.1109/ICSEC.2018.8712622>
- Liu, T., Ye, X., & Sun, B. (2018). Combining Convolutional Neural Network and Support Vector Machine for Gait-based Gender Recognition. *2018 Chinese Automation Congress (CAC)*, 3477-3481. <https://doi.org/10.1109/CAC.2018.8623118>
- Long, J., Shelhamer, E., & Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation. *arXiv:1411.4038 [cs]*. <http://arxiv.org/abs/1411.4038>
- Mandache, D., Dalimier, E., Durkin, J. R., Boceara, C., Olivo-Marin, J.-C., & Meas-Yedid, V. (2018). Basal cell carcinoma detection in full field OCT images using convolutional neural networks. *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, 784-787. <https://doi.org/10.1109/ISBI.2018.8363689>
- Mandal, B., Sultana, N., & Puhan, N. (2018). Deep Residual Network with Regularized Fisher Framework for Detection of Melanoma. *IET Computer Vision*, 12. <https://doi.org/10.1049/iet-cvi.2018.5238>
- Marghoob, A., & Jaimes, N. (2019). *Basal cell carcinoma—Dermoscopedia*. Dermoscopedia.Org. https://dermoscopedia.org/Basal_cell_carcinoma
- Menegola, A., Tavares, J., Fornaciali, M., Li, L. T., Avila, S., & Valle, E. (2017). RECOD Titans at ISIC Challenge 2017. *arXiv:1703.04819 [cs]*. <http://arxiv.org/abs/1703.04819>
- Namozov, A., & Cho, Y. I. (2018). Convolutional Neural Network Algorithm with Parameterized Activation Function for Melanoma Classification. *2018*

- International Conference on Information and Communication Technology Convergence (ICTC)*, 417-419. <https://doi.org/10.1109/ICTC.2018.8539451>
- Palacios-Martínez, D., & Díaz-Alonso, R. A. (2017). Dermatoscopia para principiantes (i): Características generales. *Medicina de Familia. SEMERGEN*, 43(3), 216-221. <https://doi.org/10.1016/j.semerg.2015.11.009>
- Putten, E. V., Kambod, A., & Kambod, M. (2018). Deep residual neural networks for automated Basal Cell Carcinoma detection. *2018 IEEE EMBS International Conference on Biomedical Health Informatics (BHI)*, 337-340. <https://doi.org/10.1109/BHI.2018.8333437>
- ¿Qué son los cánceres de piel de células basales y de células escamosas? (s. f). Recuperado 13 de noviembre de 2021, de <https://www.cancer.org/es/cancer/cancer-de-piel-de-celulas-basales-y-escamosas/acerca/que-es-cancer-de-piel-de-celulas-basales-y-escamosas.html>
- Quintero, C., Merchan, F., Cornejo, A., & Sanchez Galan F, J. (2018). Uso de Redes Neuronales Convolucionales para el Reconocimiento Automático de Imágenes de Macroinvertebrados para el Biomonitorio Participativo. En *KnE Engineering* (Vol. 3). <https://doi.org/10.18502/keg.v3i1.1462>
- Reiter, O., Mimouni, I., Gdalevich, M., Marghoob, A. A., Levi, A., Hodak, E., & Leshem, Y. A. (2019). The diagnostic accuracy of dermoscopy for basal cell carcinoma: A systematic review and meta-analysis. *Journal of the American Academy of Dermatology*, 80(5), 1380-1388. <https://doi.org/10.1016/j.jaad.2018.12.026>
- Rosendahl, C., Tschandl, P., Cameron, A., & Kittler, H. (2011). Diagnostic accuracy of dermoscopy for melanocytic and nonmelanocytic pigmented lesions. *Journal of the American Academy of Dermatology*, 64(6), 1068-1073. <https://doi.org/10.1016/j.jaad.2010.03.039>

- Sachdev, J. A., Shekhar, S., & Indu, S. (2018). Melanoma Screening Using Deep Neural Networks. *2018 3rd International Conference for Convergence in Technology (I2CT)*. <https://doi.org/10.1109/I2CT.2018.8529724>
- Serener, A., & Serte, S. (2019). Keratinocyte Carcinoma Detection via Convolutional Neural Networks. *2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 1-5. <https://doi.org/10.1109/ISMSIT.2019.8932828>
- Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556 [cs]*. <http://arxiv.org/abs/1409.1556>
- Snowflake Schwartzberg, J., & Torres González, S. (2005). Determinación del nivel de certeza en el diagnóstico dermatoscópico entre residentes, dermatólogos y dermatooncólogos de instituciones formadoras de especialistas en dermatología de la Ciudad de México. *Dermatología (México, D.F.)*, 49(5), 197-202.
- Sordo, C., & Gutiérrez, C. (2013). Cáncer de piel y radiación solar: Experiencia peruana en la prevención y detección temprana del cáncer de piel y melanoma. *Revista Peruana de Medicina Experimental y Salud Pública*, 30(1), 113-117.
- Stanford University CS231n: *Convolutional Neural Networks for Visual Recognition*. (s. f.). Recuperado 15 de noviembre de 2021, de <http://cs231n.stanford.edu/>
- Suárez, E. J. C. (2016). *Tutorial sobre Máquinas de Vectores Soporte (SVM)*. 27.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2014). Going Deeper with Convolutions. *arXiv:1409.4842 [cs]*. <http://arxiv.org/abs/1409.4842>
- Thao, L. T., & Quang, N. H. (2017). Automatic skin lesion analysis towards melanoma detection. *2017 21st Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES)*, 106-111. <https://doi.org/10.1109/IESYS.2017.8233570>

- Torres, D. J. (2018). *DEEP LEARNING Introducción práctica con Keras*.
- Tratamiento del cáncer de piel (PDQ®)—Versión para pacientes—Instituto Nacional del Cáncer (nciglobal,ncienterprise). (2021, abril 12). [PdqCancerInfoSummary]. <https://www.cancer.gov/espanol/tipos/piel/paciente/tratamiento-piel-pdq>
- Waheed, Z., Waheed, A., Zafar, M., & Riaz, F. (2017). An efficient machine learning approach for the detection of melanoma using dermoscopic images. *2017 International Conference on Communication, Computing and Digital Systems (C-CODE)*, 316-319. <https://doi.org/10.1109/C-CODE.2017.7918949>
- Wu, D. J. (2012). *END-TO-END TEXT RECOGNITION WITH CONVOLUTIONAL NEURAL NETWORKS*. 60.
- Wu, Z., Zhao, S., Peng, Y., He, X., Zhao, X., Huang, K., Wu, X., Fan, W., Li, F., Chen, M., Li, J., Huang, W., Chen, X., & Li, Y. (2019). Studies on Different CNN Algorithms for Face Skin Disease Classification Based on Clinical Images. *IEEE Access*, 7, 66505-66511. <https://doi.org/10.1109/ACCESS.2019.2918221>
- Yu, Z., Jiang, X., Wang, T., & Lei, B. (2017). *Aggregating Deep Convolutional Features for Melanoma Recognition in Dermoscopy Images* (p. 246). https://doi.org/10.1007/978-3-319-67389-9_28
- Yu, Z., Jiang, X., Zhou, F., Qin, J., Ni, D., Chen, S., Lei, B., & Wang, T. (2019). Melanoma Recognition in Dermoscopy Images via Aggregated Deep Convolutional Features. *IEEE Transactions on Biomedical Engineering*, 66(4), 1006-1016. <https://doi.org/10.1109/TBME.2018.2866166>
- Zhang, R. (2021). Melanoma Detection Using Convolutional Neural Network. *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, 75-78. <https://doi.org/10.1109/ICCECE51280.2021.9342142>

ANEXOS

A. Pruebas unitarias del backend

- i. Tarea: Verificar que el archivo que llega al backend sea del formato correcto

```
11         decoded = base64.b64decode(jpgImage)
12         decodedImageJPEG = Image.open(io.BytesIO(decoded))
13         decoded = base64.b64decode(pngImage)
14         decodedImagePNG = Image.open(io.BytesIO(decoded))
15         self.assertEqual("JPEG", decodedImageJPEG.format)
16         self.assertEqual("PNG", decodedImagePNG.format)
```

- ii. Tarea: Verificar que la normalización de la imagen sea correcta

```
def normalizeImage(self, image):
    image = np.expand_dims(image, axis=0)
    image = image / 255
    return image

def testNormalizeImage(self):
    normalizeImage = np.array(
        [
            [
                [
                    0.87843137,
                    0.52549019,
                    0.53333333,
                ],
                [
                    0.88235294,
                    0.52941176,
                    0.54509803,
                ],
                [
                    0.89019607,
                    0.53333333,
                    0.55686274,
                ],
            ],
        ]
    )
    image = np.array(
        [
            [
                224,
                134,
                136,
            ],
            [
                225,
                135,
                139,
            ],
            [
                227,
                136,
                142,
            ],
        ]
    )
    image = self.normalizeImage(image)
    image = np.trunc(image * 10 ** 8) / (10 ** 8)
    isNormalize = np.array_equal(image, normalizeImage)
    self.assertTrue(isNormalize)
```

iii. Resultados

```
Jorge@MASTERDATA MINGW64 ~/Documents/Python Scripts/flask_app (master)
$ python -m unittest
2021-12-07 11:29:53.589608: W tensorflow/stream_executor/platform/default/dso_loader.cc:44: Could not load dynamic library 'C:\Windows\System32\user32.dll': DllNotFound
2021-12-07 11:29:53.589758: I tensorflow/stream_executor/cuda/cudart_stub.cc:29: Ignore CUDA/CUDART stubs
Cargando modelo..
2021-12-07 11:29:57.972789: W tensorflow/stream_executor/platform/default/dso_loader.cc:44: Could not load dynamic library 'C:\Windows\System32\user32.dll': DllNotFound
2021-12-07 11:29:57.972889: W tensorflow/stream_executor/cuda/cuda_driver.cc:319: failed to initialize CUDA driver: CUDA_ERROR_INVALID_DEVICE
2021-12-07 11:29:57.977989: I tensorflow/stream_executor/cuda/cuda_diagn.cc:100: No GPU/TPU found. Falling back to CPU
2021-12-07 11:29:57.978250: I tensorflow/stream_executor/cuda/cuda_diagn.cc:100: No GPU/TPU found. Falling back to CPU
2021-12-07 11:29:57.978626: I tensorflow/core/platform/cpu_feature_guard.cc:142: Your TensorFlow binary is optimized with a GPU architecture (sm_37) that is not supported by your hardware (sm_70)
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
Modelo cargado
..
-----
Ran 2 tests in 0.017s

OK
```

B. Pruebas unitarias y funcionales del frontend

i. Pruebas unitarias

```
export const getBase64Image = (dataURL) => {
  const extension = getExtension(dataURL);
  const base64Image =
    extension === "png"
      ? dataURL.replace("data:image/png;base64,", "")
      : dataURL.replace("data:image/jpeg;base64,", "");
  return base64Image;
};

export const getExtension = (dataURL) => {
  return dataURL.substring(
    dataURL.indexOf("/") + 1,
    dataURL.indexOf(";base64")
  );
};
```

```
import { getBase64Image, getExtension } from "../utils";
describe("Test in utils", () => {
  test("Must return the correct extension", () => {
    const jpegImage = "data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAQABAAD/2wB";
    const pngImage = "data:image/png;base64,/9j/4AAQSkZJRgABAQAAQABAAD/2wB";

    expect(getExtension(jpegImage)).toBe("jpeg");
    expect(getExtension(pngImage)).toBe("png");
  });

  test("Must return the base64 string", () => {
    const jpegImage = "data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAQABAAD/2wB";
    expect(getBase64Image(jpegImage)).toBe("/9j/4AAQSkZJRgABAQAAQABAAD/2wB");
  });
});
```

ii. Pruebas funcionales

```
import { render, screen } from "@testing-library/react";
import UploadFileContainer from "../UploadFileContainer";
import user from "@testing-library/user-event";
import App from "../App";
import Image from "../Image";
import { Result } from ".";

test("Must render UploadFileContainer", () => {
  render(<UploadFileContainer />);
});

test("should upload the file", async () => {
  const file = new File(["hello"], "hello.png", { type: "image/png" });
  render(<App />);
  const input = screen.getByTestId("fileInput");
  user.upload(input, file);

  expect(input.files[0]).toStrictEqual(file);
  expect(input.files).toHaveLength(1);
});

test("Render image", async () => {
  const file = new File(["hello"], "hello.png", { type: "image/png" });
  render(<Image image={file} />);
  const img = screen.getByRole("img");
  expect(img).toHaveAttribute("alt", "hello.png");
});

test("Should render the results", () => {
  const payload = {
    bcc: 0.6,
    nobcc: 0.4,
  };
  render(<Result prediction={payload} />);
  expect(
    screen.getByText(/Es carcinoma basocelular: 60 %/i)
  ).toBeInTheDocument();
  expect(
    screen.getByText(/No es carcinoma basocelular: 40 %/i)
  ).toBeInTheDocument();
});
```

iii. Resultados

```
PASS src/utils/utils.test.js
PASS src/components/components.test.js

Test Suites: 2 passed, 2 total
Tests: 6 passed, 6 total
Snapshots: 0 total
Time: 3.187 s, estimated 8 s
Ran all test suites.

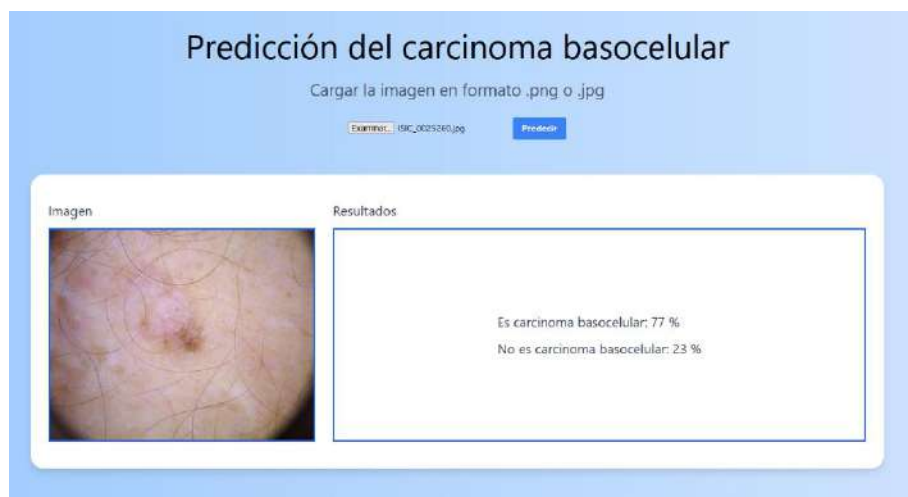
Watch Usage: Press w to show more.
```

C. Capturas de los resultados de la clasificación de carcinoma basocelular utilizando el sistema

i. Imagen ISIC_0028542.jpg



ii. Imagen ISIC_0025260.jpg



iii. Imagen ISIC_0032224.jpg

Predicción del carcinoma basocelular

Cargar la imagen en formato .png o .jpg

Eliminar ISIC_0032224.jpg Predice

Imagen	Resultados
	Es carcinoma basocelular: 0 % No es carcinoma basocelular: 100 %

iv. Imagen ISIC_0032854.jpg

Predicción del carcinoma basocelular

Cargar la imagen en formato .png o .jpg

Eliminar ISIC_0032854.jpg Predice

Imagen	Resultados
	Es carcinoma basocelular: 24 % No es carcinoma basocelular: 76 %

D. Configuración de CI/CD del frontend

```
name: CI

on:
  push:
    branches:
      - master

jobs:
  build:

    runs-on: ubuntu-latest

    steps:
      - name: Checkout 📄
        uses: actions/checkout@v1

      - name: Build 🔨
        run: |
          npm install
          npm run-script build
        env:
          REACT_APP_BACKEND_URL: ${ secrets.REACT_APP_BACKEND_URL }

      - name: Test 🟢
        run: |
          npm run test

      - name: Deploy
        uses: JamesIves/github-pages-deploy-action@releases/v3
        with:
          GITHUB_TOKEN: ${ secrets.GITHUB_TOKEN }
          BRANCH: gh-pages
          FOLDER: build
```