

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO
FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA,
INFORMÁTICA Y MECÁNICA
ESCUELA PROFESIONAL DE INGENIERÍA INFORMÁTICA Y DE
SISTEMAS



TESIS

“EVALUACIÓN DE MODELOS DE REDES NEURONALES
CONVOLUCIONALES PARA LA CLASIFICACIÓN DE LESIONES
CUTANEAS CON MELANOMA”

Presentado por:

Br. Katia Bustinza Salazar

Para optar al título profesional de:

INGENIERO INFORMÁTICO Y DE SISTEMAS

Asesor:

Mgt. Iván César Medrano Valencia

Cusco - Perú

2024

INFORME DE ORIGINALIDAD

(Aprobado por Resolución Nro.CU-303-2020-UNSAAC)

El que suscribe, **Asesor** del trabajo de investigación/tesis titulada: EVALUACION DE
MODELOS DE REDES NEURONALES CONVOLUCIONALES
PARA LA CLASIFICACION DE LESIONES CUTANEAS CON
MELANOMA

presentado por: KATIA BUSTINZA SOLAZAR con DNI Nro.: 07627784 presentado
por: con DNI Nro.: para optar el
título profesional/grado académico de INGENIERO INFORMÁTICO
Y DE SISTEMAS.....

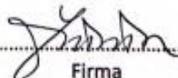
Informo que el trabajo de investigación ha sido sometido a revisión por 2 veces, mediante el
Software Antiplagio, conforme al Art. 6° del **Reglamento para Uso de Sistema Antiplagio de la**
UNSAAC y de la evaluación de originalidad se tiene un porcentaje de 5.....%.

Evaluación y acciones del reporte de coincidencia para trabajos de investigación conducentes a grado académico o
título profesional, tesis

Porcentaje	Evaluación y Acciones	Marque con una (X)
Del 1 al 10%	No se considera plagio.	X
Del 11 al 30 %	Devolver al usuario para las correcciones.	
Mayor a 31%	El responsable de la revisión del documento emite un informe al inmediato jerárquico, quien a su vez eleva el Informe a la autoridad académica para que tome las acciones correspondientes. Sin perjuicio de las sanciones administrativas que correspondan de acuerdo a Ley.	

Por tanto, en mi condición de asesor, firmo el presente informe en señal de conformidad y **adjunto**
la primera página del reporte del Sistema Antiplagio.

Cusco, 15 de FEBRERO de 2024.....


.....
Firma

Post firma IVAN CESAR PEDRANO VALENCIA

Nro. de DNI 23881504.....

ORCID del Asesor 0000-0003-0623-090X.....

Se adjunta:

1. Reporte generado por el Sistema Antiplagio.
2. Enlace del Reporte Generado por el Sistema Antiplagio: old: 27259:332248469

NOMBRE DEL TRABAJO

Tesis_Katia_similitud.docx

AUTOR

KATIA BUSTINZA SALAZAR

RECuento DE PALABRAS

19607 Words

RECuento DE CARACTERES

103907 Characters

RECuento DE PÁGINAS

90 Pages

TAMAÑO DEL ARCHIVO

10.5MB

FECHA DE ENTREGA

Feb 15, 2024 8:20 AM GMT-5

FECHA DEL INFORME

Feb 15, 2024 8:22 AM GMT-5**● 5% de similitud general**

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos.

- 5% Base de datos de Internet
- Base de datos de Crossref
- 3% Base de datos de trabajos entregados
- 0% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

● Excluir del Reporte de Similitud

- Material bibliográfico
- Material citado
- Bloques de texto excluidos manualmente
- Material citado
- Coincidencia baja (menos de 20 palabras)

Acrónimos

CNN: Red Neuronal Convolutacional.

UV: Ultra Violeta.

CBC: Carcinoma Basocelular.

ADN: Acido Desoxirribonucleico.

CDKN2A: Gen cromosoma 9

CDK4: Gen Proteico

ABCDE: Asimetría, Bordes irregulares o mal definidos, Color Heterogéneo, Dimensión Evolución.

RGB: Rojo, Verde, Azul.

BMP: Mapa de Bits

GIF: Formato de Intercambio de Gráficos

PNG: Gráficos de Red Portátiles

TIFF: Formato de Archivos de Imágenes con Etiquetas

JPEG: Formato de Archivo de Imagen.

ML: Aprendizaje Automático.

ANN: Red Neuronal Artificial.

CPU: Unidad Central de Proceso

GPU: Unidad de Procesamiento de Gráficos

TPU: Unidad de Procesamiento Tensorial

RELU: Unidad Lineal Rectificada.

PDF: Formato de Documentos Portátiles.

VSG: Formato de Generador Vectorial de Señales.

CDR: Formato de Registro Detallado de Llamadas

GIF: Formato de Intercambio de Gráfico.

RAW: Formato de Imagen sin Perdida

PSP: Formatos Utilizados por Programas Gráficos

PSD: Formatos Utilizados por Programas Gráficos.

Agradecimiento.

En primer lugar, quiero agradecer a Jehová Dios por la oportunidad de haber realizado mis estudios, a mis padres Silvia, Pedro y mi querida hermana Marianela que están siempre en mis recuerdos, a mi esposo Pedro Martín y a mi hijo Joakim por ser el motivo de mi vida. A mis hermanos Marco por su apoyo incondicional por estar siempre conmigo, Percy por apoyarme en momentos difíciles. También quiero agradecer a mi asesor el Mgt. Iván Cesar Medrano Valencia por guiarme en el desarrollo de este proyecto, a la Universidad San Antonio Abad del Cusco, por brindarme sus instalaciones y disponer de un grupo de excelentes profesores que me compartieron sus conocimientos.

Resumen

Se considera como procedimiento clínico estándar el examen visual para la detección e identificación de lesiones cutáneas, el problema es que este procedimiento es muy subjetivo y puede llevar a diagnósticos incorrectos, aunque haya sido realizado por dermatólogos de mucha experiencia. Por esta razón se han propuesto diversos enfoques para la clasificación de lesiones cutáneas basadas en redes neuronales convolucionales basadas en transferencia de aprendizaje como apoyo al diagnóstico de cáncer de piel. El problema es que estas redes neuronales convolucionales pre entrenadas poseen diversas arquitecturas lo cual afecta su desempeño. Debido a esto el objetivo de esta investigación es evaluar los modelos basados en InceptionV3, MobileNetV2 y DenseNet201 para determinar los modelos más adecuados para clasificar imágenes de lesiones cutáneas con melanoma.

Luego de los experimentos realizados con el conjunto de datos HAMM10000 se comprobó que el modelo basado en InceptionV3 presento una exactitud de 78.5% mientras que MobileNetV2 y DenseNet201 presentaron un 74.5% y 77.63% respectivamente. Sin embargo, debido al desbalance del conjunto de datos y utilizando la métrica Score F1 para evaluar los modelos se pudo verificar que el modelo basado en InceptionV3 se desempeñó mejor clasificando imágenes de las clases minoritarias.

Palabras clave: Redes Neuronales Convolucionales, Cáncer de piel, InceptionV3, MobileNetV2, DenseNet201, Transferencia de aprendizaje.

Abstract

Visual examination for the detection and identification of skin lesions is considered a standard clinical procedure. The problem is that this procedure is very subjective and can lead to incorrect diagnoses, even if it has been performed by highly experienced dermatologists. For this reason, various approaches have been proposed for the classification of skin lesions based on convolutional neural networks based on transfer learning to support the diagnosis of skin cancer. The problem is that these pre-trained convolutional neural networks have diverse architectures which affects their performance. Due to this, the objective of this research is to evaluate the models based on InceptionV3, MobileNetV2 and DenseNet201 to determine the most appropriate models to classify images of skin lesions with melanoma.

After the experiments carried out with the HAMM10000 data set, it was found that the model based on InceptionV3 presented an accuracy of 78.5% while MobileNetV2 and DenseNet201 presented 74.5% and 77.63% respectively. However, due to the imbalance of the data set and using the Score F1 metric to evaluate the models, it was possible to verify that the model based on InceptionV3 performed better classifying images from the minority classes.

Keywords: Convolutional Neural Networks, Skin Cancer, InceptionV3, MobileNetV2, DenseNet201, Transfer Learning.

Índice general

Índice general	vi
1 Introducción	1
1.1 El problema.....	2
1.1.1 Formulación del problema.....	3
1.2 Justificación y motivación.	3
1.3 Objetivo General.....	4
1.3.1 Objetivos Específicos	4
1.4 Organización del texto	4
2 Marco Teórico	6
2.1 La Piel.....	6
2.2 Cáncer de piel.	6
1.1.1 Carcinoma Basocelular.....	7
2.3 Procesamiento de imágenes digitales	10
2.3.1 Imagen digital.....	10
2.3.2 Operaciones típicas del procesamiento de imágenes.....	11
2.3.3 Clasificación de imágenes digitales.....	12
2.3.4 Tipos de archivos de imágenes digitales	13
2.3.5 Aplicaciones del Procesamiento Digital de Imágenes.....	14
2.3.6 Aprendizaje Automático.....	17
2.3.7 Redes Neuronales Artificiales.	21
2.3.8 Evaluación de Modelos.	31
2.3.9 Deep Learning.	34
2.3.10 Redes Neuronales Convolucionales (CNN).	36
2.3.11 Transferencia de Aprendizaje.....	46
2.3.12 Aumento de datos.	47
2.3.13 Redes Neuronales Convolucionales Pre entrenadas.....	50
3 Trabajos Relacionados.....	54
3.1 Detección precoz de cáncer de piel en imágenes basado en redes convolucionales (México).	54
3.2 Clasificación de lesiones en la piel con ensemble de redes neuronales residuales. (Barcelona - España).....	54
3.3 Arquitectura de red neuronal convolucional para diagnóstico de cáncer de piel (Perú).	55

3.4	Herramienta de reconocimiento y clasificación de lesiones cutáneas basada en redes neuronales convolucionales para asistir en el diagnóstico de cáncer de piel tipo melanoma (Argentina).....	55
3.5	Melanoma detection by analysis of clinical images using convolutional neural network.	56
3.6	A comparative study of melanoma skin cancer detection in traditional and current image processing techniques.	56
4	Desarrollo del Proyecto	58
4.1	Metodología	58
4.1.1	Tipo y nivel de investigación.....	58
4.1.2	Diseño metodológico.....	58
4.2	El conjunto de datos.....	58
4.2.1	HAMM10000	58
4.2.2	PH ²	59
4.2.3	ISIC 2016.....	59
4.2.4	Atlas de dermatoscopia.....	59
4.2.5	Dermofit	59
4.2.6	BCN20000	60
4.3	Conjunto de datos utilizado en el proyecto.....	60
4.3.1	Distribución de los datos.	62
4.3.2	Distribución de los datos de prueba por clase.	63
4.4	Preprocesamiento de las imágenes.	63
4.4.1	Espacio de color YUV.....	65
4.4.2	Detección de oclusiones.	65
4.4.3	Retoque.....	66
4.4.4	Filtro bilateral	66
4.4.5	Fuzzy C Means.	67
4.4.6	Supresión de esquinas.....	68
4.4.7	Cierre.	68
4.4.8	Resultados.....	69
4.5	Aumento de datos	69
4.6	Reentrenamiento de los modelos seleccionados.....	70
4.6.1	Determinación de hiperparámetros.....	70
4.6.2	Construcción de los nuevos modelos.	71
4.7	Hardware y Software utilizado.	73

5	Resultados y Discusión.....	74
5.1	Evaluación del modelo basado en InceptionV3.....	74
5.1.1	Resultados del entrenamiento.....	74
5.1.2	Matriz de confusión.....	75
5.2	Evaluación del modelo basado en MobileNetV2.....	76
5.2.1	Resultados del entrenamiento.....	76
5.2.2	Matriz de confusión.....	76
5.3	Evaluación del modelo DenseNet201.....	78
5.3.1	Resultados del entrenamiento.....	78
5.3.2	Matriz de confusión.....	78
5.4	Comparación de resultados de los modelos.....	79
6	Conclusiones y Recomendaciones.....	81
6.1	Conclusiones.....	81
6.2	Recomendaciones.....	82
	Bibliografía.....	83

Índice de Tablas

Tabla 4.1 Tipos de patología de la piel y el número de instancias en el conjunto de datos HAM10000.....	60
Tabla 4.2 Número de Muestras por clase de Entrenamiento	63
Tabla 4.3 Número de muestras por clase	63
Tabla 4.4 Hiperparámetros de entrenamiento de los modelos seleccionados.....	71
Tabla 5.1 Métricas del modelo basado en InceptionV3.....	75
Tabla 5.2 Métricas del modelo basado en MobileNetV2	77
Tabla 5.3 Métricas del modelo basado en DenseNet201	79
Tabla 5.4 Tabla Resultados de exactitud de los modelos estudiados	80
Tabla 5.5 Tabla Score F1 de los modelos estudiados	80

Índice de Figuras

Figura 2.1 Estructura de la piel (Society, 2020)	6
Figura 2.2 Carcinoma basocelular nodular. Imagen con dermatoscopia. (Castañeda Gameros Paola, 2016)	7
Figura 2.3 Carcinoma epidermoide en labio inferior. Imagen macroscópica	8
Figura 2.4 Melanoma de extensión superficial. Imagen con dermatoscopia.....	10
Figura 2.5 Un ejemplo del proceso de adquisición de una imagen digital (Tiagy,2018)	10
Figura 2.6 Imagen en escala de grises (Tiagy, 2018)	11
Figura 2.7 Los tres canales RGB que forman una imagen a color (Tiagy, 2018)	11
Figura 2.8 Zoom de una imagen de mapa de bits (a) Imagen original (b) Parte de la imagen ampliada 4 veces (c) La misma parte de la imagen ampliada 8 veces. (Tiagy, 2018).....	12
Figura 2.9 (a) Una imagen de (b), La imagen de (a) ampliada en formato vectorial (c), La imagen de (a) ampliada en formato Raster. (Tiagy, 2018).....	13
Figura 2.10 Un conjunto de entrenamiento etiquetado para el aprendizaje supervisado	18
Figura 2.11 Un conjunto de entrenamiento sin etiquetas para el aprendizaje no supervisado (Aurélien, 2019).	19
Figura 2.12 Subajuste, Sobreajuste y buen ajuste del modelo (Aurélien, 2019).....	21
Figura 2.13 Arquitectura de red neuronal con una única capa oculta (Salazar, 2020)....	22
Figura 2.14 Una red neuronal artificial toma una entrada y combina las contribuciones de los nodos para calcular una salida y^i con la ayuda de una función no lineal con la suma de sus entradas (Salazar, 2020).	23
Figura 2.15 Arquitectura de red neuronal con una sola capa oculta, incluido el sesgo. Las entradas a un nodo (marcadas en gris) se usan junto con los pesos w para calcular la salida con la ayuda de la función de activación (Salazar, 2020).	25
Figura 2.16 Acercamiento a uno de los nodos ocultos en nuestra arquitectura de red neuronal (Salazar, 2020).....	25
Figura 2.17 Función de activación sigmoide (Salazar, 2020)	26
Figura 2.18 Función de activación Tangente hiperbólica.....	26
Figura 2.19 Función de activación ReLU (Salazar, 2020)	27
Figura 2.20 Función de activación Softmax (Salazar, 2020)	27
Figura 2.21 (a, b, c). Propagación hacia atrás de errores, o backpropagation, permite que la red neuronal aprenda de sus errores (Salazar, 2020).	30

Figura 2.22 La matriz de confusión (Adam, 2017)	31
Figura 2.23 Ejemplos de curvas ROC (Rogers Simón, 2017).....	34
Figura 2.24 Inteligencia Artificial, Aprendizaje Automático y Aprendizaje profundo (Chollet, 2018).....	34
Figura 2.25 Operación de convolución (Mirza Rahim Baig, 2020).....	40
Figura 2.26 Ejemplo de salto (stride) (Mirza Rahim Baig, 2020).....	41
Figura 2.27 Agrupación máxima con stride 2 en una imagen de entrada	43
Figura 2.28 Ejemplo de relleno (Mirza Rahim Baig, 2020).....	44
Figura 2.29 Red Neuronal Convolutiva (Michelucci, 2019)	46
Figura 2.30 Transferencia de aprendizaje (Vasilev, 2019).....	47
Figura 2.31 Ejemplo de voltear horizontalmente (Mirza Rahim Baig, 2020).....	48
Figura 2.32 Ejemplo de voltear verticalmente (Mirza Rahim Baig, 2020).....	48
Figura 2.33 Ejemplo de escalamiento (Mirza Rahim Baig, 2020).....	48
Figura 2.34 Ejemplo de desplazamiento horizontal (Mirza Rahim Baig, 2020).....	49
Figura 2.35 Ejemplo de desplazamiento vertical (Mirza Rahim Baig, 2020)	49
Figura 2.36 Ejemplo de rotación (Mirza Rahim Baig, 2020).....	49
Figura 2.37 Ejemplo de corte (Mirza Rahim Baig, 2020).....	50
Figura 2.38 Modelo general de red pre entrenada para reconocimiento	51
Figura 2.39 Comparación de la operación de convolución estándar y la convolución separable en profundidad utilizada en Howar (2017).....	51
Figura 2.40 Red DenseNet presentada en Huang et. al. (2018).....	52
Figura 2.41 Arquitectura Inception-v3 (Batch Norm y ReLU se usan después de Convolución.)	53
Figura 4.1 Imágenes de Keratosis Actínica (akiec) (P. Tschandl, 2018)	61
Figura 4.2 Carcinoma de Células Basales (bcc) (P. Tschandl, 2018)	61
Figura 4.3 Imágenes de Dermatofibroma (df) (P. Tschandl, 2018)	61
Figura 4.4 Imágenes de Melanoma (mel) (P. Tschandl, 2018)	61
Figura 4.5 Imágenes de Nevus (nv) (P. Tschandl, 2018)	61
Figura 4.6 Imágenes de Queratosis benigna pigmentada (bkl) (P. Tschandl, 2018).....	61
Figura 4.7 Imágenes de Lesiones Vasculares (vasc) (P. Tschandl, 2018).....	62
Figura 4.8 Distribución de las imágenes de entrenamiento y de prueba	62
Figura 4.9 Artefactos en las imágenes del conjunto de datos HAM10000 (Elaboracion propia).....	64

Figura 4.10 diagrama de bloques del preprocesamiento de imágenes de lesiones cutáneas De Castillo A. (2016).....	64
Figura 4.11 . Proceso de la detección de oclusiones a partir del canal de luminancia en la imagen, Obtención de la máscara de desviaciones y su umbralización para la obtención de los pixeles a retocar (Castillo A. 2016).....	66
Figura 4.12 Proceso de retoque a) Imagen original, b) Mascara de retoque, c) Imagen con retoque (Castillo A. 2016).....	626
Figura 4.13 Enfoque para la supresión de esquinas de la segmentación (Castillo A. 2016).	628
Figura 4.14 Resultados del preprocesamiento (Castillo A. 2016).....	7069
Figura 4.15 Aumento de datos (Fuente: Elaboración Propia)	70
Figura 4.17 Creación del modelo base	71
Figura 4.18 Congelamiento de las últimas capas del modelo base	72
Figura 4.19 Adición de capas al modelo base	72
Figura 4.20 Compilación del nuevo modelo.....	723
Figura 4.21 Entrenamiento del nuevo modelo.	723
Figura 5.1 Gráficas de pérdida y exactitud del modelo basado en InceptionV3 Fuente (Elaboración Propia).....	74
Figura 5.2 Matriz de confusión del modelo basado en InceptionV3.....	75
Figura 5.3 Gráficas de pérdida y exactitud del modelo basado en MobileNetV2 Fuente (Elaboración Propia).....	76
Figura 5.4 Matriz de confusión del modelo basado en MobileNetV2 Fuente (Elaboración Propia).....	77
Figura 5.5 Gráficas de pérdida y exactitud del modelo basado en DenseNet201	78
Figura 5.6 Matriz de confusión del modelo basado en DenseNet201	79

Capítulo 1

1 Introducción

El cáncer de piel se ha clasificado como una de las enfermedades más comunes en el mundo, y su forma más dañina es el melanoma. Un diagnóstico temprano puede reducir la tasa de mortalidad si el cáncer se detecta y trata en las primeras etapas (*Skin Cancer Foundation, 2023*), por lo que, realizar esfuerzos de investigación en el desarrollo de estrategias de detección temprana del cáncer es muy importante.

Para el diagnóstico de cáncer de piel, la dermatoscopia es una de las técnicas de imagen más populares utilizadas por los dermatólogos. Aumenta la superficie de la lesión cutánea y su estructura se vuelve más visible para el dermatólogo durante un examen. Sin embargo, esta técnica sólo puede ser utilizada eficazmente por médicos capacitados, porque se basa totalmente en la agudeza visual y la experiencia del dermatólogo (Rebecca et, al, 2020).

Un dermatólogo experto puede lograr una precisión promedio del 65 % al 75 % utilizando la dermatoscopia. Además, la precisión se puede mejorar aún más en casos sospechosos, utilizando una cámara con una alta resolución especial y una lente de aumento para capturar imágenes dermatoscópicas para inspección visual. La luz se regula durante el proceso de captura de imágenes, lo que mejora la visibilidad de las capas profundas de la piel. Se estima que la precisión del diagnóstico de cáncer de piel se puede mejorar en un 50 % con este soporte tecnológico (Rebecca et.al. 2020). Sin embargo, esta técnica no deja de ser subjetiva. Por lo que, el diagnóstico computarizado es una necesidad debido a la creciente tasa de casos positivos, a la subjetividad del procedimiento convencional y los gastos de tiempo y costos.

Por otro lado, los enfoques de aprendizaje profundo han mostrado resultados prometedores en la clasificación de imágenes. Los mecanismos de aprendizaje profundo pueden aprender un conjunto de características de alto nivel a partir de otras de bajo nivel y obtener una alta precisión para aplicaciones de clasificación sin la necesidad de extraer características hechas a mano. Especialmente hay una tendencia a aprovechar el poder del aprendizaje profundo en tareas de imágenes médicas. La red neuronal convolucional (CNN) es un tipo de método de aprendizaje profundo en el que se aplican filtros entrenables y operaciones de agrupación en las imágenes de entrada sin procesar, extrayendo automáticamente un conjunto de características complejas de alto nivel.

En esta investigación, pretendemos aprovechar los métodos de aprendizaje profundo para formar un sistema de diagnóstico automático para la detección del melanoma utilizando Redes Neuronales Convolucionales mediante la Transferencia de Aprendizaje realizando un estudio comparativo del desempeño de redes neuronales pre entrenadas tales como InceptionV3, MobileNetV2 y DenseNet201 para determinar cuál de éstas presenta mejor rendimiento.

1.1 El problema.

En el Perú, el cáncer de piel fue el cuarto tipo de cáncer más frecuente en 2012 y en 2017 fue el segundo (7,9 % vs 10,4%) (Salud., 2013). Cada año se diagnostican más de 66000 casos nuevos y se estima que fallecen más de 32 000 personas por esta enfermedad (Minsa, 2019).

Las medidas preventivas y el diagnóstico temprano de cáncer de piel ayudarían para disminuir los indicadores antes mencionados (de Haas ER, 2010).

Actualmente, el examen visual se considera el procedimiento clínico estándar para la detección e identificación de lesiones cutáneas. Uno de los protocolos más populares para la evaluación médica de las lesiones en la piel es el protocolo 'ABCD', según el cual el dermatólogo examina la (A) asimetría, el (B) borde, el (C) color y el (D) diámetro de la mancha en la piel. En cuanto a la asimetría, se examina si los segmentos del área de la lesión cutánea divergen o no en forma. En cuanto a los bordes, se examina si los bordes de la zona correspondiente de la piel son irregulares o borrosos y, en algunos casos, si presentan muescas. El color puede ser desigual y con varias coloraciones de negro, marrón y rosa. En cuanto al diámetro de la lesión cutánea, la mayoría de los melanomas tienen un diámetro de al menos seis milímetros, siendo cualquier cambio en el tamaño, la forma o el diámetro un dato fundamental que debe conocer el personal médico correspondiente (F. Nachbar et al., 1994). El examen médico de la piel de los pacientes lo llevan a cabo expertos en dermatología, y normalmente es un examen que requiere mucho tiempo. Sin embargo, el diagnóstico de las lesiones de la piel descrito anteriormente, es un procedimiento muy subjetivo, ya que el diagnóstico puede ser impreciso o incorrecto o puede dar lugar a diagnósticos bastante disímiles, aunque haya sido realizado por dermatólogos con mucha experiencia (F. Nachbar et al., 1994). Es por esta razón que, durante la última década, se han publicado diferentes enfoques para la clasificación automática de lesiones cutáneas basada en imágenes utilizando redes

neuronales convolucionales (CNN) como apoyo al diagnóstico del cáncer de piel con diferentes grados de eficiencia. La mayoría de los modelos de CNN se entrenaron mediante aprendizaje por transferencia utilizando modelos inicializados con pesos previamente entrenados por ImageNet¹. El problema radica en que estos modelos pre entrenados, aplicados a la clasificación de lesiones de piel a través de transferencia de aprendizaje, se pueden desempeñar de diferentes formas afectando su capacidad de predicción debido a que tienen diferentes arquitecturas.

1.1.1 Formulación del problema.

Problema General

¿Cuál es el mejor modelo de redes neuronales convolucionales pre entrenadas para clasificación de imágenes de lesiones cutáneas con melanoma utilizando transferencia de aprendizaje?

Problemas específicos.

- ¿Cuál es el conjunto de datos más adecuado para evaluar la clasificación de imágenes de lesiones cutáneas con redes neuronales convolucionales pre entrenadas?
- ¿Cuál es el preprocesamiento necesario en las imágenes de lesiones de piel para obtener un adecuado rendimiento de la red neuronal convolucional para clasificar imágenes de lesiones de piel?

1.2 Justificación y motivación.

En la actualidad, los avances en el aprendizaje automático, se han reconocido como un área importante de investigación en el desarrollo de métodos computarizados efectivos y confiables para ayudar a los radiólogos y médicos en el análisis de imágenes en diversas etapas del diagnóstico y manejo de enfermedades durante el proceso de atención al paciente. En este ámbito, el aprendizaje profundo se ha convertido en el método de aprendizaje automático de última generación en muchas aplicaciones, de tal manera que,

¹ *ImageNet* es un gran conjunto de datos de fotografías comentadas destinadas a la investigación de visión por computadora. El objetivo de desarrollar este conjunto de datos es de proporcionar un recurso para promover la investigación y el desarrollo de métodos mejorados para la visión por computadora. <https://www.image-net.org/>

para tareas de reconocimiento de patrones en imágenes, las redes neuronales convolucionales (CNN) son las redes de aprendizaje profundo más utilizadas.

Por otro lado, debido a los límites y la dificultad de la evaluación clínica de las lesiones cutáneas descritas anteriormente y también debido al número de casos relacionados con enfermedades de la piel que aumentan cada año, es necesario el desarrollo de herramientas de aprendizaje automático para la clasificación de lesiones cutáneas para el análisis de manchas relacionadas con melanomas o diferentes patologías cutáneas.

Por lo que la importancia de esta investigación, es la utilización de los logros tecnológicos de vanguardia en las áreas de procesamiento de imágenes digitales, con énfasis en el aprendizaje profundo y big data aplicados para lograr la detección del cáncer de piel. Estas herramientas pueden hacer factible permitir el diagnóstico exitoso de lesiones cutáneas evitando la necesidad de contacto corporal con la piel del paciente o incluso realizar el diagnóstico a distancia enviando al dermatólogo una foto de la zona de piel correspondiente a través de internet.

1.3 Objetivo General

Evaluar modelos de redes neuronales convolucionales pre entrenados para determinar el mejor modelo para clasificar lesiones cutáneas con melanoma utilizando transferencia de aprendizaje

1.3.1 Objetivos Específicos

- Determinar el conjunto de datos más adecuado para evaluar la clasificación de imágenes de lesiones cutáneas con redes neuronales convolucionales pre entrenadas.
- Establecer el preprocesamiento necesario en las imágenes de lesiones de piel para obtener un adecuado rendimiento de la red neuronal convolucional para clasificar imágenes de lesiones de piel.

1.4 Organización del texto

Esta investigación está dividida en 6 capítulos: En el **CAPITULO 1** se presenta una introducción, la definición del problema y los objetivos del trabajo. En el **CAPITULO 2** se realiza una revisión sistemática de los conceptos que dan el soporte teórico a esta

investigación. En el **CAPITULO 3** presentamos el resumen de los aspectos más importantes de los trabajos relacionados con esta investigación. En el **CAPITULO 4** presentamos la metodología utilizada para alcanzar los objetivos propuestos, se presentan la selección de los modelos pre entrenados, el análisis del conjunto de datos, el pre procesamiento de los datos y la construcción de los nuevos modelos. Además, en este capítulo también se realiza la discusión de los resultados obtenidos para determinar el o los modelos que mejor clasifican lesiones de piel con melanoma. En el **CAPITULO 5** presentamos los Resultados y los experimentos y Finalmente, en el **CAPITULO 6** presentamos las conclusiones y recomendaciones de este trabajo.

Capítulo 2

2 Marco Teórico

2.1 La Piel.

La piel es el órgano más grande del cuerpo humano y tiene un peso aproximado de 4 kg en una superficie de 2m^2 (Rosendahl Cliff, 2019). Debido a que la piel cubre la superficie externa, es más vulnerable a lesiones de muchas fuentes, incluida la radiación incidente, por lo que, como era de esperar, es el sitio más común de malignidad. Por la misma razón, esas neoplasias malignas son más accesibles a la inspección visual directa, lo que hace que el desarrollo de herramientas para ayudar a dicha inspección sea muy relevante.

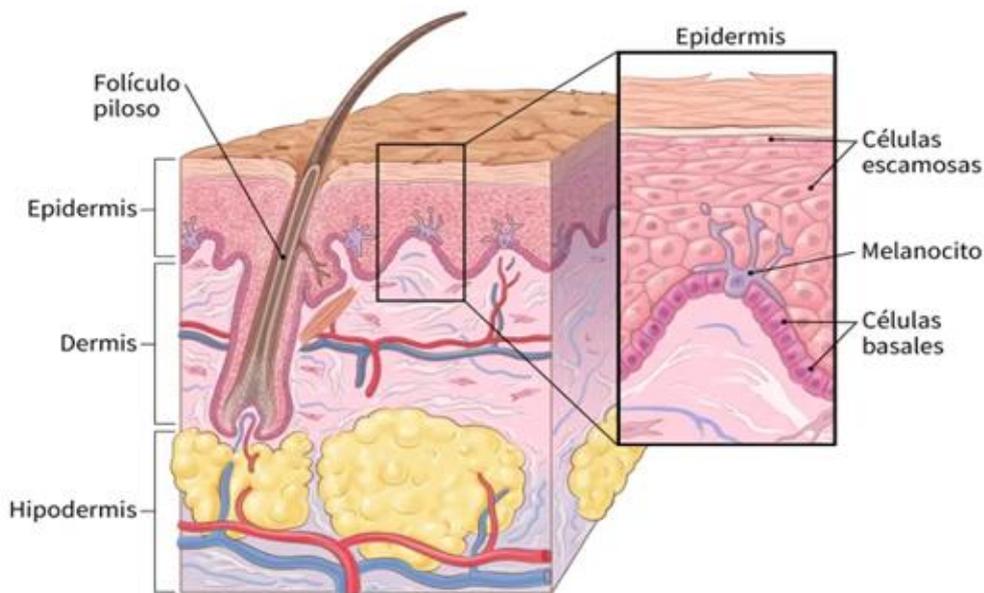


Figura 2.1 Estructura de la piel (Society, 2020)

2.2 Cáncer de piel.

El cáncer de piel es una de las neoplasias más comunes, en algunos países ocupa el primer lugar en frecuencia (Castañeda Gameros Paola, 2016). Su incidencia se ha incrementado en las últimas décadas y es muy importante que el médico general tenga una idea amplia de lo que implica esta patología y referir al paciente con el especialista para el diagnóstico oportuno y el tratamiento adecuado. El riesgo de un sujeto para desarrollar cáncer en la piel depende de factores constitucionales y ambientales. Los factores constitucionales

incluyen la historia familiar, cabello claro o rojo, múltiples nevos melanocíticos, sensibilidad a la exposición solar, entre otros. Mientras que la radiación ultravioleta (UV) es un factor de riesgo ambiental bien establecido, y el más importante. El término de cáncer de piel incluye tumores de diversa estirpe celular y diferente agresividad y pronóstico. Así, tenemos:

- Carcinoma basocelular.
- Carcinoma espinocelular o epidermoide.
- Melanoma maligno.
- Adenocarcinoma de glándulas sebáceas.
- Sarcomas y linfomas.

Debido a su frecuencia, haremos una breve revisión de los tres primeros.

1.1.1 Carcinoma Basocelular

El carcinoma basocelular (CBC) (Castañeda Gámeros Paola, 2016) es la variante más frecuente de cáncer de piel. Como su nombre lo indica, tiene su origen en la capa basal de la epidermis y sus apéndices. Se caracteriza por tener un crecimiento lento, ser localmente invasivo y destructivo y presentar un bajo potencial metastásico. Factores ambientales y genéticos contribuyen al desarrollo del CBC. La exposición a la radiación UV es el factor más importante. Otros factores de riesgo establecidos incluyen la exposición crónica al arsénico, terapia con radiación, terapia inmunosupresora, traumas mecánicos, cicatrices recalcitrantes y el síndrome de nevos de células basales. El 70% de los CBC se presentan en la cara y la cabeza. Las formas más comunes son la nodular y la superficial, que juntas representan el 90% de los casos.

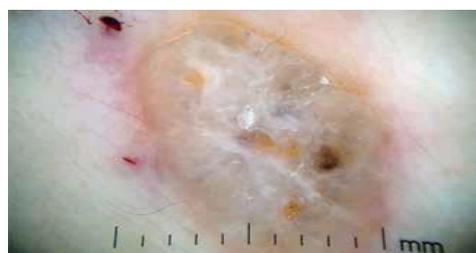


Figura 2.2 Carcinoma basocelular nodular. Imagen con dermatoscopia. (Castañeda Gámeros Paola, 2016)

2.2.1.1 Carcinoma Epidermoide o Espinocelular

Éste es el segundo en frecuencia después del carcinoma basocelular y es resultado de la transformación maligna de los queratinocitos de la epidermis y sus anexos. (Castañeda Gameros Paola, 2016). La exposición crónica a radiación UV representa el factor de riesgo más importante para el desarrollo del carcinoma epidermoide. La radiación UV es capaz de inducir daño al ácido desoxirribonucleico (ADN) que lleva a la transformación de los queratinocitos,

además de alterar la respuesta inmunológica de la piel, volviéndola más susceptible para la formación de tumores. El carcinoma espinocelular es dos a tres veces más frecuente en hombres y aparece la mayoría de las veces en pacientes mayores de 50 años. Sin embargo, en las últimas tres décadas el número de mujeres menores de 40 años afectadas por esta neoplasia ha aumentado de manera importante y se ha atribuido al abuso en el empleo de las camas de bronceado. Se estima que las personas que las utilizan con frecuencia multiplican su riesgo de presentar esta neoplasia hasta 2.5 veces más.



Figura 2.3 Carcinoma epidermoide en labio inferior. Imagen macroscópica (Castañeda Gameros Paola,2016)

2.2.1.2 Melanoma

Es la forma más peligrosa de cáncer cutáneo, se origina en los melanocitos (células productoras de pigmento) de origen neuroectodérmico que se ubican en diferentes partes del cuerpo, incluyendo la piel, el iris y el recto1 (Castañeda Gameros Paola, 2016). Semejan nevos, algunos de ellos incluso se originan en ellos. El incremento en la exposición a la luz UV en una población genéticamente predispuesta parece ser la responsable del incremento en la incidencia y mortalidad del melanoma en las últimas décadas. Dentro de los factores de riesgo se incluyen:

- **Piel blanca.** Al tener menos pigmento (melanina) en la piel se tiene menor protección contra la radiación UV. Presentan mayor riesgo aquellas personas con cabello claro o rojo, ojos claros y múltiples efélides (pecas).
- **Historia de quemadura solar.** Haber presentado una o más quemaduras solares intensas, sobre todo durante la infancia.
- **Exposición excesiva a luz UV.** Proveniente del sol o de camas de bronceado. La luz UV fue identificada como un carcinógeno mayor involucrado en la melanogénesis. El limitar la exposición a la luz UV, así como el uso regular de filtro solar, han demostrado disminuir la incidencia de melanomas primarios de piel.
- **Vivir cerca del ecuador o en lugares con mayor altitud.** Las personas que viven cerca del ecuador, donde los rayos solares son más directos, reciben mayores cantidades de luz UV. Si se vive a una mayor altitud mayor es la radiación ultravioleta.
- **Múltiples nevos y nevos atípicos.** Tener más de 50 nevos en todo el cuerpo aumenta el riesgo de padecer melanoma.
- **Historia familiar de melanoma.** Mutaciones en los genes CDKN2A y CDK4, que han sido documentados en algunas familias con melanoma hereditario, confieren un 60 a 90% de riesgo para presentar melanoma.
- **Inmunosupresión.** Las personas con sistemas inmunitarios debilitados tienen un mayor riesgo de presentar melanoma y otros cánceres de piel.
- **Edad.** El daño al ADN es acumulativo con el paso del tiempo.

Los signos tempranos más comunes de melanoma quedan resumidos en el acrónimo **ABCDE** donde (Castañeda Gameros Paola, 2016):

- **A** = Asimetría.
- **B** = Bordes irregulares o mal definidos.
- **C** = Color heterogéneo.
- **D** = Diámetro mayor a 6 mm.
- **E** = Evolución, cualquier cambio en el tamaño, forma, color o la aparición de algún signo como sangrado, ulceración o la referencia de dolor o prurito deben ser considerados signos de alerta.



Figura 2.4 Melanoma de extensión superficial. Imagen con dermatoscopia.
(Castañeda Gameros Paola,2016)

2.3 Procesamiento de imágenes digitales

2.3.1 Imagen digital

Una imagen es una representación visual de un objeto, una persona o una escena. Una imagen digital es una función bidimensional $f(x, y)$ que es una proyección de una escena tridimensional en un plano de proyección bidimensional, donde x, y representa la ubicación del elemento o píxel de la imagen y contiene el valor de la intensidad (Tiagy, 2018).

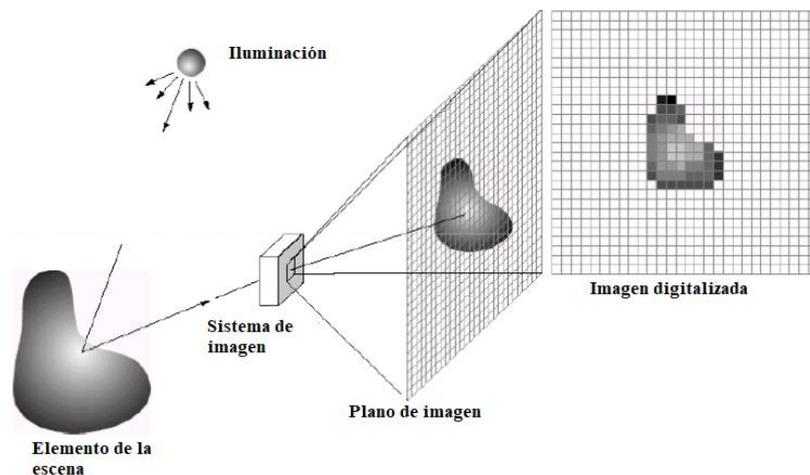


Figura 2.5 Un ejemplo del proceso de adquisición de una imagen digital (Tiagy,2018)

Cuando los valores de intensidad x, y son discretos, se dice que la imagen es una imagen digital. Matemáticamente, una imagen digital es una representación matricial de una imagen bidimensional que utiliza un número finito de elementos de celda de puntos, generalmente denominados píxeles (elementos de imagen o pels).

Cada píxel está representado por valores numéricos: para imágenes en escala de grises, es suficiente que un solo valor represente la intensidad del píxel (generalmente en un

rango [0, 255]); para las imágenes en color, se almacenan tres valores (que representan la cantidad de rojo (R), verde (G) y azul (B)).



Figura 2.6 Imagen en escala de grises (Tiagy, 2018)



Figura 2.7 Los tres canales RGB que forman una imagen a color (Tiagy, 2018)

2.3.2 Operaciones típicas del procesamiento de imágenes

El procesamiento de imágenes implica una serie de técnicas y algoritmos. Las operaciones de tratamiento de imágenes más representativas son (Tiagy, 2018):

- **Binarización:** muchas tareas de procesamiento de imágenes se pueden realizar convirtiendo una imagen en color o una imagen en escala de grises en binario para simplificar y acelerar el procesamiento. La conversión de una imagen en color o en escala de grises a una imagen binaria que tiene solo dos niveles de gris (blanco y negro) se conoce como binarización.
- **Suavizado:** una técnica que se utiliza para desenfocar o suavizar los detalles de los objetos en una imagen. Se logra convolucionando sobre la imagen diferentes filtros para eliminar o disminuir el ruido.
- **Nitidez:** las técnicas de procesamiento de imágenes, mediante las cuales los bordes y los detalles finos de los objetos en una imagen se mejoran para la visualización humana, se conocen como técnicas de nitidez. Estas técnicas son la aplicación de operaciones morfológicas sobre la imagen.

- **Eliminación de ruido y desenfoque:** antes del procesamiento, la cantidad de ruido en las imágenes se reduce mediante filtros de eliminación de ruido. A veces se puede utilizar la técnica de eliminación de imágenes, según el tipo de ruido o desenfoque de la imagen. Se logra también convolucionando diferentes filtros sobre la imagen para eliminar el ruido que puede ser Gaussiano o de Sal y Pimienta.
- **Extracción de bordes:** para encontrar varios objetos antes de analizar el contenido de la imagen, se realiza la extracción de bordes, aplicando diversas técnicas como el algoritmo de Canny.
- **Segmentación:** El proceso de dividir una imagen en varias partes se conoce como segmentación. Para el reconocimiento y clasificación de objetos, la segmentación es un paso previo al procesamiento. Existen varias técnicas para realizar la segmentación, entre los más utilizados son: la umbralización de los valores de grises, y la detección de bordes en las imágenes.

2.3.3 Clasificación de imágenes digitales

Con respecto a la forma en que se almacenan, las imágenes digitales se pueden clasificar en dos categorías: (1) Imagen de trama o mapa de bits, (2) Imagen vectorial (Tiagy, 2018).

Una imagen de mapa de bits o de trama es una matriz rectangular de valores o píxeles muestreados.

Estas imágenes tienen un número fijo de píxeles. En el zoom de una imagen rasterizada, se aplica la interpolación matemática. La calidad de una imagen ampliada se degrada después de un valor particular del factor de zoom, como se muestra en la Fig. 2.8. La resolución de una imagen de mapa de bits está determinada por el dispositivo de detección. Los formatos de imagen comunes BMP, GIF, PNG, TIFF y JPEG son formatos de imagen de mapa de bits o de trama.

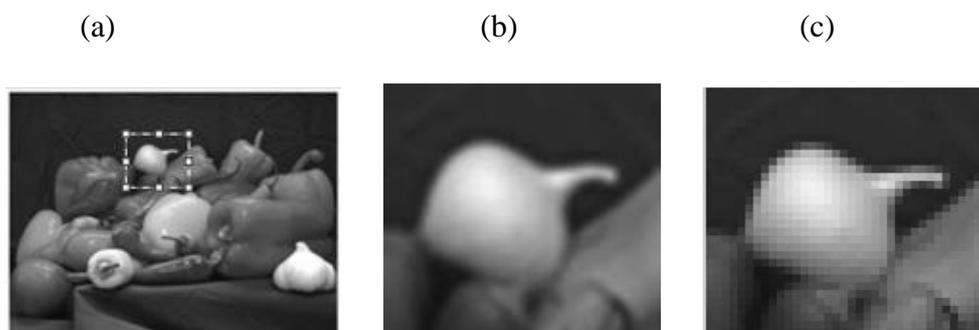


Figura 2.8 Zoom de una imagen de mapa de bits (a) Imagen original (b) Parte de la imagen ampliada 4 veces (c) La misma parte de la imagen ampliada 8 veces. (Tiagy, 2018).

Por otro lado, las imágenes vectoriales se almacenan en forma de líneas y curvas matemáticas. La información como la longitud, el color, el grosor, etc., se almacena en forma de vector. Estas imágenes se pueden mostrar en cualquier tamaño, cualquier resolución y en cualquier dispositivo de salida. Las imágenes vectoriales son adecuadas en ilustraciones, arte lineal, fuente, etc. La diferencia en el zoom de una imagen vectorial y una imagen ráster se puede observar en la Figura. 2.9. La degradación de la calidad debido al zoom es claramente visible en los límites del carácter almacenado en formato de trama. Los formatos de imágenes vectoriales son: Formato .AI (*Adobe Illustrator*), Formato .EPS, Formato .PDF, Formato .SVG, Formato .CDR.

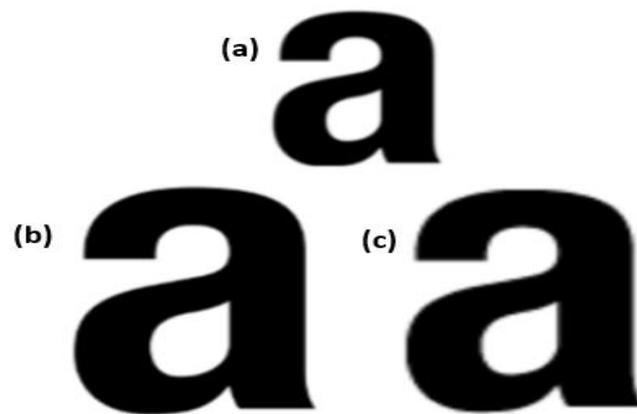


Figura 2.9(a) Una imagen de (b), La imagen de (a) ampliada en formato vectorial (c), La imagen de (a) ampliada en formato Raster. (Tiagy, 2018)

2.3.4 Tipos de archivos de imágenes digitales

Hay una serie de tipos de archivos de imágenes digitales disponibles en estos días. Los tipos de archivos de imagen más utilizados son: JPEG, GIF, TIFF, PNG y BMP. Los tipos de archivos de imagen se basan en la técnica de compresión utilizada para reducir el tamaño del archivo de imagen. Las imágenes en varios tipos de archivos pueden diferir en color, si se ha utilizado color. Una imagen en su forma más simple puede contener solo dos intensidades, es decir, blanco y negro, y necesita solo 1 bit para representar la intensidad en cada píxel (Tiagy, 2018).

- **TIFF (Formato de archivo de imagen etiquetado):** Este formato es muy flexible y puede basarse en una técnica de compresión con o sin pérdida. Los detalles de la técnica de compresión se almacenan en la propia imagen. En general, los archivos

TIFF utilizan un formato de almacenamiento de imágenes sin pérdidas y, por lo tanto, tienen un tamaño bastante grande.

- **Gráficos de red portátiles (PNG):** Es un formato de almacenamiento sin pérdidas y utiliza patrones en la imagen para comprimirla. La compresión de los archivos PNG es exactamente reversible, lo que significa que la imagen sin comprimir es idéntica a la imagen original.
- **Formato de intercambio gráfico (GIF):** Este formato crea una tabla de hasta 256 colores a partir de 16 millones de colores. Si la imagen a comprimir tiene menos de 256 colores, entonces la imagen GIF tiene exactamente el mismo color. Pero si el número de colores es superior a 256, el GIF aproxima los colores de la imagen utilizando una tabla de los 256 colores disponibles.
- **Joint Picture Experts Group (JPG o JPEG):** Este es un formato optimizado para fotografías e imágenes de tonos continuos que contienen una gran cantidad de colores. Los archivos JPEG pueden lograr altos índices de compresión mientras mantienen una calidad de imagen clara.
- **RAW:** Este es un formato de imagen sin pérdida disponible en algunas cámaras digitales. Estos archivos son muy pequeños pero el formato depende del fabricante, por lo tanto, se requiere el software del fabricante para ver las imágenes.
- **Imagen de mapa de bits (BMP):** Es un formato patentado sin comprimir inventado por Microsoft.

Algunos otros formatos de archivo de imagen comunes son PSD, PSP, etc., que son formatos propietarios utilizados por programas de gráficos. Los archivos de Photoshop tienen la extensión PSD, mientras que los archivos de *Paintshop Pro* usan la extensión PSP.

2.3.5 Aplicaciones del Procesamiento Digital de Imágenes

Las técnicas de procesamiento de imágenes digitales ahora se utilizan en varias aplicaciones; algunas aplicaciones comunes se dan a continuación (Tiagy, 2018).

- **En medicina:** Varias herramientas médicas utilizan el procesamiento de imágenes para diversos fines, como la mejora de imágenes, la compresión de imágenes, el reconocimiento de objetos, etc. Radiación X (rayos X), tomografía computarizada (TC), tomografía por emisión de positrones (PET), la tomografía computarizada por emisión de fotón único (SPECT), la espectroscopia de

resonancia magnética nuclear (RMN) y la ultrasonografía son algunos de los equipos médicos populares basados en el procesamiento de imágenes.

- **En agricultura:** El procesamiento de imágenes juega un papel vital en el campo de la agricultura. Varias tareas primordiales, como la detección de malezas, la clasificación de alimentos, el control de la cosecha y la recolección de frutas, se realizan automáticamente con la ayuda del procesamiento de imágenes. El mapeo de tierras irrigadas, la determinación de índices de vegetación, la medición del dosel, etc., son posibles con buena precisión mediante el uso de técnicas de imagen en varios espectros, como imágenes hiperespectrales, infrarrojos, etc.
- **En el pronóstico del tiempo:** El procesamiento de imágenes también juega un papel crucial en el pronóstico del tiempo, como la predicción de lluvias, granizadas e inundaciones. Los radares meteorológicos son ampliamente utilizados para detectar nubes de lluvia y, en base a esta información, los sistemas predicen la intensidad de lluvia inmediata.
- **En fotografía y cine:** Las fotos retocadas y empalmadas se utilizan ampliamente en periódicos y revistas con el fin de mejorar la calidad de la imagen. En las películas, muchas escenas complejas se crean con herramientas de edición de imágenes y videos que se basan en operaciones de procesamiento de imágenes y videos. Los métodos basados en el procesamiento de imágenes se utilizan para predecir el éxito de las próximas películas. Para una empresa global de medios y entretenimiento, *Latent View* extrajo más de 6000 carteles de películas de IMDB junto con sus metadatos (género, elenco, producción, calificaciones, etc.) para predecir el éxito de las películas mediante el análisis de imágenes. Los esquemas de color y los objetos en los carteles de las películas se analizaron utilizando algoritmos de aprendizaje automático o *machine Learning* (ML) y técnicas de procesamiento de imágenes.
- **En el entretenimiento y las redes sociales:** La detección y el reconocimiento de rostros se usan ampliamente en los sitios de redes sociales donde, tan pronto como un usuario sube la fotografía, el sistema identifica automáticamente y sugiere etiquetar a la persona por su nombre.
- **En seguridad:** Los sistemas de verificación biométrica proporcionan un alto nivel de autenticidad y confidencialidad. Las técnicas de verificación biométrica se utilizan para el reconocimiento de humanos en función de sus comportamientos o

características. Para crear alertas de comportamientos especialmente indeseables, se están empleando sistemas de video vigilancia para analizar los movimientos y actividades de las personas. Los bancos y otros departamentos están utilizando estos sistemas de video vigilancia basados en el procesamiento de imágenes para detectar actividades no deseadas.

- **En banca y finanzas:** El uso de técnicas basadas en el procesamiento de imágenes está aumentando rápidamente en el campo de los servicios financieros y la banca. La "captura remota de depósitos" es una instalación bancaria que permite a los clientes depositar cheques electrónicamente utilizando dispositivos móviles o escáneres. Los datos de la imagen del cheque se extraen y utilizan en lugar de un control físico. La detección de rostros también se está utilizando en el proceso de autenticación de clientes bancarios. Algunos bancos utilizan la 'biometría facial' para proteger la información confidencial. La verificación y el reconocimiento de firmas también juegan un papel importante en la autenticación de la firma de los clientes. Sin embargo, todavía se necesita desarrollar un sistema robusto para verificar las firmas manuscritas. Este proceso tiene muchos desafíos porque las firmas manuscritas son imprecisas por naturaleza, ya que sus esquinas no siempre son nítidas, las líneas no son perfectamente rectas y las curvas no son necesariamente suaves.
- **En *marketing* y publicidad:** Algunas empresas utilizan el intercambio de imágenes a través de las redes sociales para rastrear la influencia de los últimos productos/publicidad de la empresa. Las empresas de turismo utilizan imágenes para anunciar destinos turísticos.
- **En defensa:** El procesamiento de imágenes, junto con la inteligencia artificial, está contribuyendo a la defensa en función de dos necesidades fundamentales de las fuerzas armadas: una es la operación autónoma y la otra es el uso de salidas de una diversa gama de sensores sofisticados para predecir peligros/amenazas. En la guerra Irán-Irak, se emplearon tecnologías de teledetección para el reconocimiento del territorio enemigo. Las imágenes de satélite se analizan para detectar, localizar y destruir armas y sistemas de defensa utilizados por las fuerzas enemigas.
- **En la automatización industrial:** Se ha visto un uso sin precedentes del procesamiento de imágenes en la automatización industrial. Los principales

objetivos son determinar la calidad de los objetos y detectar cualquier anomalía en los productos. Muchas industrias también utilizan la clasificación de productos por automatización de formas.

- **En medicina forense:** Los documentos manipulados se utilizan ampliamente en casos penales y civiles, como testamentos impugnados, papeleo financiero y documentación comercial profesional. Documentos como pasaportes y permisos de conducir se manipulan con frecuencia para usarlos ilegalmente como prueba de identificación. Los departamentos forenses tienen que identificar la autenticidad de tales documentos sospechosos. Identificar la falsificación de documentos se vuelve cada vez más desafiante debido a la disponibilidad de herramientas avanzadas de edición de documentos. El falsificador utiliza la última tecnología para perfeccionar su arte. Los documentos escaneados por computadora se copian de un documento a otro para hacerlos genuinos. La falsificación no solo se limita a los documentos, sino que también está ganando popularidad en las imágenes. Las imágenes tienen un papel destacado en diversas áreas, como la investigación forense, la investigación criminal, los sistemas de vigilancia, los sistemas de inteligencia, los deportes, los servicios legales, las imágenes médicas, las reclamaciones de seguros y el periodismo.
- **En la restauración y mejora de imágenes submarinas:** Las imágenes submarinas a menudo no son claras. Estas imágenes tienen varios problemas, como ruido, bajo contraste, desenfoque, iluminación no uniforme, etc. Para restaurar la claridad visual, se utilizan técnicas de mejora de imágenes.

2.3.6 Aprendizaje Automático.

El aprendizaje automático es la ciencia (y el arte) de programar computadoras para que puedan aprender de los datos (Aurélien, 2019).

A continuación, una definición un poco más general:

“El aprendizaje automático es el campo de estudio que brinda a las computadoras la capacidad de aprender sin ser programadas explícitamente”.

—Arthur Samuel, 1959

Y uno más orientado a la ingeniería:

“Se dice que un programa de computadora aprende de la experiencia E con respecto a alguna tarea T y alguna medida de desempeño P , si su desempeño en T , medido por P , mejora con la experiencia E .”—Tom Mitchell, 1997

Por ejemplo, su filtro de correo no deseado es un programa de aprendizaje automático que puede aprender a marcar el correo no deseado dados ejemplos de correos electrónicos no deseados (p. ej., marcados por los usuarios) y ejemplos de correos electrónicos regulares (que no son correo no deseado, también llamados "ham"). Los ejemplos que utiliza el sistema para aprender se denominan conjunto de entrenamiento. Cada ejemplo de entrenamiento se denomina instancia de entrenamiento.

En este caso, la tarea T es marcar el spam para nuevos correos electrónicos, la experiencia E son los datos de entrenamiento y la medida de rendimiento P debe definirse; por ejemplo, puede utilizar la proporción de correos electrónicos clasificados correctamente. Esta medida de rendimiento en particular se llama precisión y se usa a menudo en tareas de clasificación (Aurélien, 2019).

Los sistemas de aprendizaje automático se pueden clasificar según la cantidad y el tipo de supervisión que reciben durante el entrenamiento. Hay cuatro categorías principales: aprendizaje supervisado, aprendizaje no supervisado, aprendizaje semisupervisado y aprendizaje por refuerzo (Aurélien, 2019).

A continuación, revisaremos los dos primeros:

2.3.6.1 Aprendizaje Supervisado.

En el aprendizaje supervisado, los datos de entrenamiento que alimenta al algoritmo incluyen las soluciones deseadas, llamadas etiquetas (Figura.2.10).

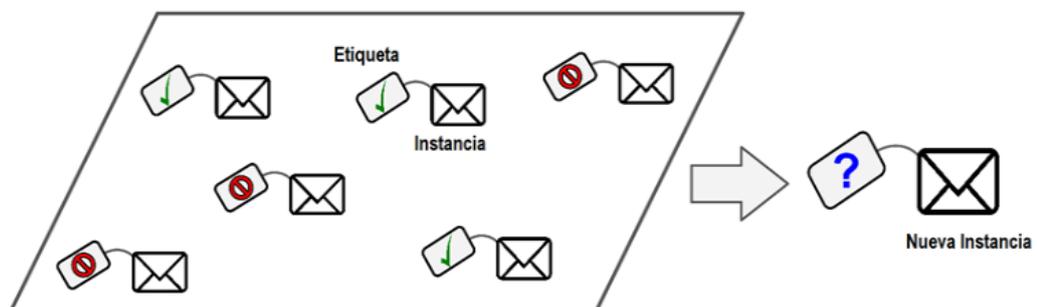


Figura 2.10 Un conjunto de entrenamiento etiquetado para el aprendizaje supervisado (Aurélien, 2019)

Una tarea típica de aprendizaje supervisado es la clasificación. El filtro de spam es un buen ejemplo de esto: está entrenado con muchos correos electrónicos de ejemplo junto con su clase (spam² o ham³), y debe aprender a clasificar los nuevos correos electrónicos. Otra tarea típica es predecir un valor numérico objetivo, como el precio de un automóvil, dado un conjunto de características (kilometraje, antigüedad, marca, etc.) llamadas predictoras. Este tipo de tarea se llama regresión. Para entrenar el sistema, se debe dar muchos ejemplos de automóviles, incluidos sus predictores y sus etiquetas (es decir, sus precios) (Aurélien, 2019).

Estos son algunos de los algoritmos de aprendizaje supervisado más importantes:

- k-vecinos más cercanos
- Regresión lineal
- Regresión logística
- Máquinas de vectores de soporte (SVM)
- Árboles de decisión y bosques aleatorios
- Redes neuronales

2.3.6.2 Aprendizaje No Supervisado.

En el aprendizaje no supervisado, como se puede suponer, los datos de entrenamiento no están etiquetados (Figura. 2.11). El sistema trata de aprender sin un maestro (Aurélien, 2019).

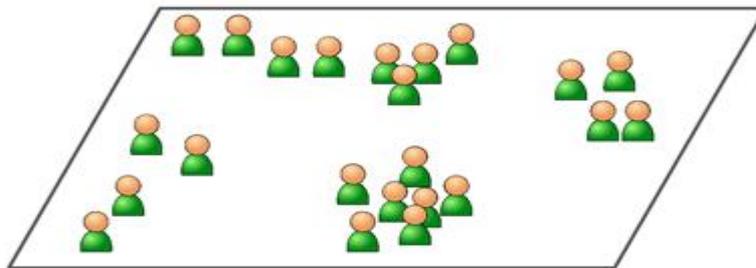


Figura 2.11 Un conjunto de entrenamiento sin etiquetas para el aprendizaje no supervisado (Aurélien, 2019).

En esta técnica, la máquina está diseñada para interactuar con el entorno circundante a través de acciones. La máquina es recompensada o castigada en función de si el entorno reacciona positiva o negativamente a las acciones realizadas por la máquina. La máquina

² Se conoce como "spam" a los correos electrónicos no solicitados que se envían a un gran número de destinatarios con fines publicitarios o comerciales.

³ El término "ham" actualmente se define y entiende como correo electrónico que generalmente se desea y no se considera spam.

aprende de estas respuestas y luego se le enseña a actuar de una manera que maximice las recompensas que ganará en el futuro. El objetivo de la máquina también podría ser minimizar los castigos o reacciones negativas que recibe. Este tipo de aprendizaje está estrechamente relacionado con la teoría del control en ingeniería y la teoría de la decisión en ciencias de la administración y estadística (Sebastian, 2018).

Estos son algunos de los algoritmos de aprendizaje no supervisado más importantes (Aurélien, 2019):

- Agrupación
 - K-Medios
 - DBSCAN
 - Análisis de conglomerados jerárquicos (HCA)
- Detección de anomalías y detección de novedades
 - SVM de una clase
 - Bosque de Aislamiento
- Visualización y reducción de dimensionalidad
 - Análisis de componentes principales (PCA)
 - PCA del núcleo
 - Incrustación lineal local (LLE)
 - Incrustación de vecinos estocásticos distribuidos en t (t-SNE)
- Aprendizaje de reglas de asociación
 - A priori
 - Eclat

2.3.6.3 Subajuste y Sobreajuste.

Subajuste. El subajuste no contribuye a la precisión del modelo de aprendizaje automático. Cuando aparece significa que el modelo o algoritmo no se ajusta suficientemente bien a los datos. Sucede cuando no se tienen los suficientes datos para construir un modelo preciso o cuando se trata de construir un modelo lineal con datos no lineales (Aurélien, 2019).

Sobreajuste. El sobreajuste se presenta cuando el modelo aprende los detalles, esto incluye el ruido de los datos de entrenamiento lo cual tiene un impacto negativo en el rendimiento del modelo cuando trabaja con datos nuevos. Esto quiere decir, que el ruido en los datos de entrenamiento es tomado y aprendido por el modelo lo que afecta la capacidad de generalización del modelo (Aurélien, 2019).

Buen ajuste. Un buen ajuste se considera aquel modelo que se encuentra entre el subajuste y el sobreajuste.

El objetivo es evitar el sobreajuste de los modelos de aprendizaje automático y existen varias maneras para lograrlo, se mencionan seguidamente algunos de ellos (Aurélien, 2019).

- Utilizar mayor cantidad de datos para entrenamiento
- Validación cruzada
- Detección temprana
- Regularización

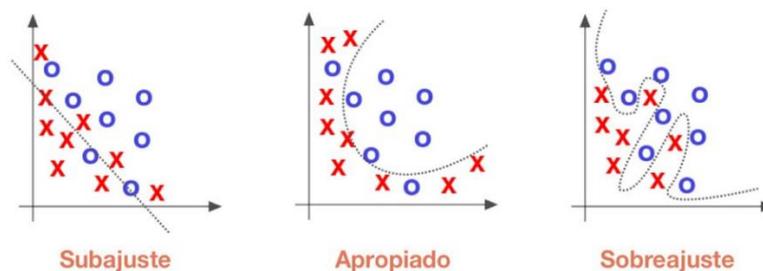


Figura 2.12 Subajuste, Sobreajuste y buen ajuste del modelo (Aurélien, 2019).

2.3.7 Redes Neuronales Artificiales.

Una red neuronal artificial (ANN) es efectivamente un sistema informático que tiene en cuenta entradas que se combinan, normalmente de forma no lineal, para calcular salidas que pueden compararse con los resultados esperados. La motivación detrás de las ANN está vagamente inspirada en las redes neuronales biológicas que constituyen los cerebros de los animales.

En ese sentido, se dice que las redes neuronales son capaces de aprender a realizar tareas a partir de los datos proporcionados (muestras de datos), sin necesidad de ser programadas específicamente con reglas (Salazar, 2020).

Por lo tanto, podemos describir una arquitectura de red neuronal general como una colección de nodos conectados entre sí para permitir la transferencia y manipulación de información. Los nodos se agregan en capas y, por lo tanto, la información fluye de una capa a la siguiente de manera dirigida. En la Figura. 2.13 se puede ver una arquitectura de red neuronal típica, donde los nodos están representados por círculos abiertos y las conexiones entre ellos se muestran como aristas dirigidas.

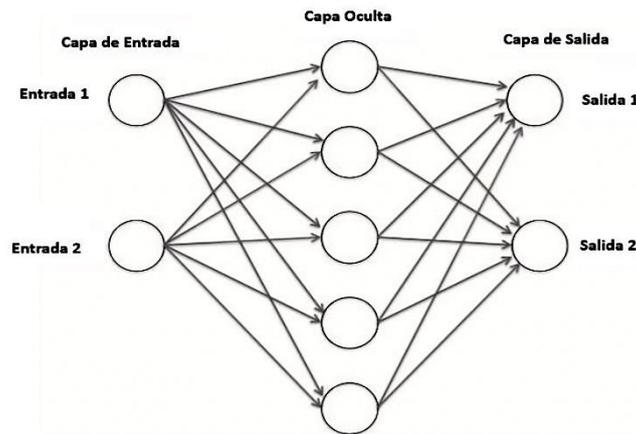


Figura 2.13 Arquitectura de red neuronal con una única capa oculta (Salazar, 2020).

En este caso, tenemos un grafo dirigido donde los nodos pertenecen a capas escalonadas y la información en la red fluye en una sola dirección desde una capa a la siguiente.

Cada uno de los bordes de la red tiene un peso y cada nodo puede tomar las entradas proporcionadas y combinarlas antes de pasarlas al siguiente nivel de nodos. El proceso de aprendizaje nos permite ajustar o “aprender” los pesos óptimos para los bordes a medida que avanza el entrenamiento, para que podamos hacer predicciones con la red (Salazar, 2020).

2.3.7.1 ¿Qué son las Redes Neuronales Artificiales?

Como hemos visto en la sección anterior, las redes neuronales artificiales pueden entenderse en términos de sus representaciones esquemáticas como grafos. Profundizando en el funcionamiento de estos grafos se puede ver cómo pueden aprender patrones, ver imágenes, reconocer el habla y capturar nuestra imaginación colectiva (Salazar, 2020).

Lo primero que debe señalarse es que, la información presente en el grafo no está integrada en la arquitectura de la red. En su lugar, se requiere una entrada externa. Esta información luego se usa para calcular una salida con una función de la suma de sus entradas. Podemos representar esto como se muestra en la Figura 2.14, donde podemos ver n nodos que contribuyen al cálculo de la salida y a través de una función $f(\cdot)$. Se observa que cada borde i tiene su propio peso w_i

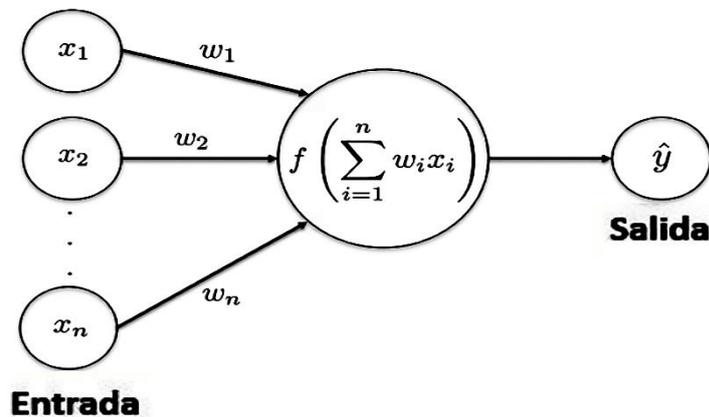


Figura 2.14 Una red neuronal artificial toma una entrada y combina las contribuciones de los nodos para calcular una salida \hat{y} con la ayuda de una función no lineal con la suma de sus entradas (Salazar, 2020).

La arquitectura de red que tenemos es un grafo dirigido ponderado y el objetivo es ajustar los pesos de los bordes a medida que se lleva a cabo el aprendizaje. Podemos pensar en este proceso como si tuviera un conjunto de diales para ajustar la fuerza de una señal en una conexión dada en la red. Como en el caso de su contraparte biológica, las sinapsis entre neuronas pueden dispararse, o activarse, si la señal agregada está por encima de cierto umbral.

2.3.7.2 Arquitectura de red neuronal: capas y nodos.

Una característica importante de la arquitectura de las neuronales es el hecho de que los nodos están dispuestos en capas. En otras palabras, la información en la red fluye de una capa a la siguiente en la dirección prescrita por los bordes dirigidos. Contamos el número de capas comenzando efectivamente desde cero. Esa primera capa suele llamarse capa de entrada y está formada por nodos pasivos que toman la entrada. La última capa suele llamarse capa de salida y está formada por nodos activos. Es decir, toman los resultados de la capa anterior y modifican las señales recibidas. Entre las capas de entrada y salida, podemos tener cualquier número de las llamadas capas ocultas. Estas capas ocultas también están formadas por nodos activos. En la Figura 2.14 se muestra un ejemplo típico de una red neuronal de una sola capa (Salazar, 2020).

Cada una de las capas en la arquitectura de la red neuronal puede tener cualquier cantidad de nodos y, como tal, tanto la cantidad de capas ocultas como la cantidad de nodos en cada capa son algunos de los parámetros que debe decidir primero.

A medida que aumenta el número de capas ocultas en nuestra arquitectura, más y más profundamente debe fluir la información de entrada. Se utiliza la palabra "más profundo" con un poco de intención, ya que esto es lo que da lugar al término aprendizaje profundo para describir el trabajo realizado con grandes redes neuronales artificiales. Hoy en día, las arquitecturas de aprendizaje profundo tienen una amplia gama de aplicaciones que incluyen reconocimiento de voz, visión por computadora, traducción automática, generación de texto, subtítulos de imágenes, etc. (Salazar, 2020).

Diferentes capas en la arquitectura realizarán diferentes tipos de manipulaciones y transformaciones en sus respectivas entradas. Los nodos de la capa de entrada no realizan ningún cálculo; simplemente, pero lo que es más importante, transmiten la información a los nodos en la primera capa oculta. Los nodos ocultos realizan cálculos sobre las entradas recibidas y transfieren el resultado a las siguientes capas, hasta llegar a la capa de salida. Este proceso generalmente se conoce como *feedforward* donde la información se mueve en una sola dirección; no hay ciclos ni bucles.

A medida que aumenta el número de capas ocultas, podemos realizar tareas más complejas. Cuantas más capas añadamos, más eficazmente podremos realizar la ingeniería automática de características.

Aunque las capas ocultas no interactúan directamente con el mundo exterior, sí tienen un profundo impacto en el resultado final devuelto por la red neuronal. Por un lado, tener muy pocos nodos en las capas ocultas dan lugar a un ajuste insuficiente, ya que no hay suficientes nodos para detectar los posibles patrones complejos en nuestros datos. Por otro lado, la presencia de demasiados nodos también puede generar otros problemas. El sobreajuste es uno de ellos, ya que la capacidad adicional de la red permite que el sistema memorice los atributos de los datos, especialmente cuando el conjunto no es lo suficientemente grande. Incluso en los casos en que hay suficientes datos, debemos tener en cuenta que cuantos más nodos tengamos en las capas ocultas, más tiempo de entrenamiento.

2.3.7.3 Activación de las neuronas.

Consideremos una red neuronal de una capa oculta como se muestra en la Figura 2.15. Esta arquitectura no solo es capaz de representar funciones lineales, sino también las no lineales. La capa de entrada tiene tres nodos, uno de los cuales hemos marcado como bias con un valor de 1, los otros dos nodos toman los valores x_1 y x_2 . Estas entradas se pasan a la siguiente capa junto con sus pesos asociados.

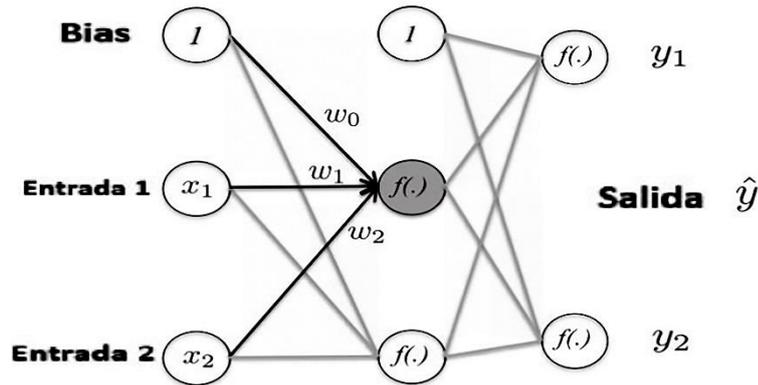


Figura 2.15 Arquitectura de red neuronal con una sola capa oculta, incluido el sesgo. Las entradas a un nodo (marcadas en gris) se usan junto con los pesos w para calcular la salida con la ayuda de la función de activación (Salazar, 2020).

Utilizamos un nodo en la (primera) capa oculta de esta red neuronal (resaltada en gris). Ver Figura 2.16. Este nodo oculto recibe las entradas $(1, x_1, x_2)$ junto con los pesos asociados (w_0, w_1, w_2) y usa los valores para calcular la función $f(\cdot)$ cuyo argumento es una suma de las entradas como $\sum x_i w_i$. Esta función se denomina función de activación.

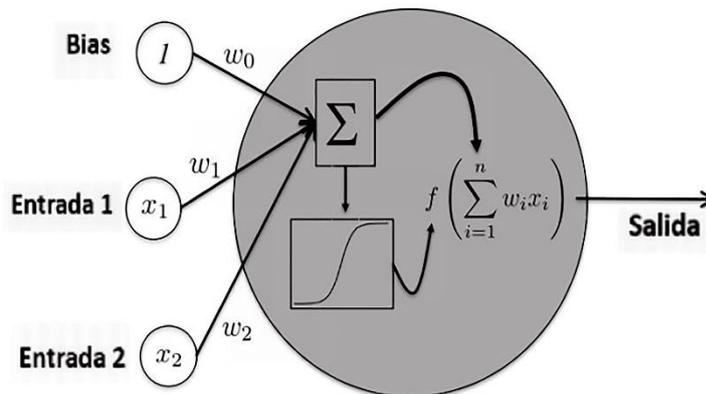


Figura 2.16 Acercamiento a uno de los nodos ocultos en nuestra arquitectura de red neuronal (Salazar, 2020).

El proceso descrito anteriormente se lleva a cabo para cada uno de los nodos de la primera capa y el valor resultante de la función de activación se utiliza como entrada para la siguiente capa, y así sucesivamente. En la arquitectura que se muestra en la Figura 2.15, tenemos dos nodos de salida. Luego de ejecutar el mismo proceso con la función de activación $f(\cdot)$ sus resultados (y_1, y_2) son la salida de la red neuronal.

2.3.7.4 Función de Activación.

La función de activación es una función no lineal. Esto se debe a que estamos interesados en representar datos complejos del mundo real con nuestras redes neuronales. Por lo tanto,

la función de activación introduce la no linealidad en las salidas del nodo, lo que permite que la arquitectura aprenda estas representaciones complejas (Salazar, 2020).

Existen varias funciones de activación, cada una de estas funciones ofrece diferentes formas de combinar las entradas, y las funciones de activación más comunes incluyen las siguientes:

- **Sigmoide:** esta función toma una entrada de valor real y la asigna a un rango de valores entre 0 y 1. La función sigmoide está dada por la siguiente expresión y podemos ver un gráfico en la Figura 2.17.

$$f(x) = \frac{2}{1 + e^{-x}}$$

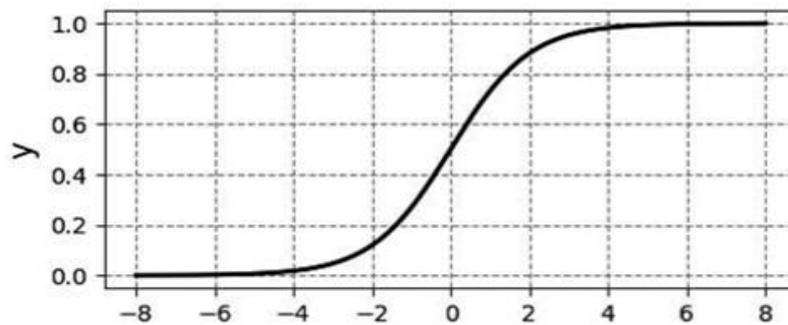


Figura 2.17 Función de activación sigmoide (Salazar, 2020)

- **Tangente hiperbólica:** esta función también toma entradas de valores reales y las asigna a un rango de valores en el intervalo $[-1, 1]$. Es efectivamente un cambio de escala de la función sigmoidea. Podemos ver la forma de la función tanh en la Figura 2.18.

$$f(x) = \frac{2}{1 + e^{-2x}} - 1$$

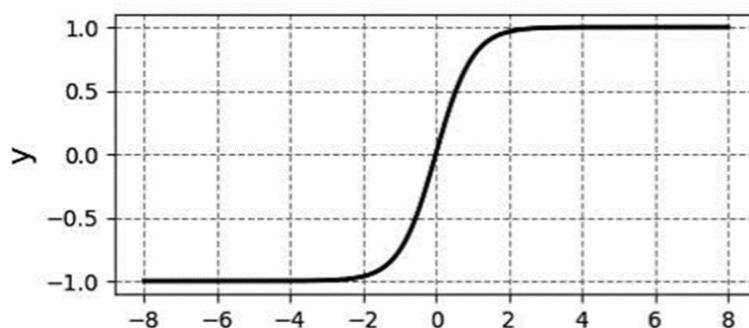


Figura 2.18 Función de activación Tangente hiperbólica

- **Unidad lineal rectificada (ReLU):** esta función establece un umbral mínimo de cero a entradas negativas y asigna valores positivos a sí mismos:

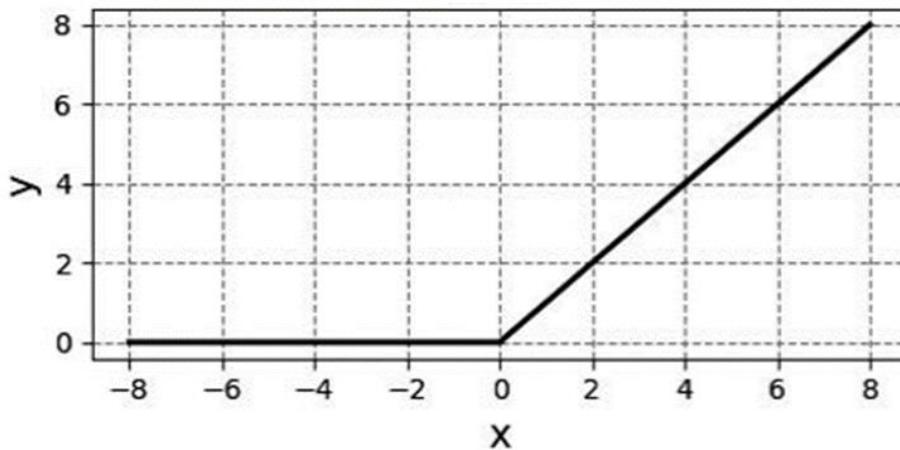


Figura 2.19 Función de activación ReLU (Salazar, 2020)

- **Softmax.** Esta función es efectivamente una generalización de la función sigmoidea. Toma valores reales como entrada y los asigna a una distribución de probabilidad donde la entrada está en el rango (0, 1). Además, todas las entradas suman 1. La función de activación *Softmax* está dada por la siguiente ecuación:

$$f(z)_j = \frac{e^{z_j}}{\sum_{k=1}^k e^{z_k}}$$

Y una gráfica puede verse en la Figura 2.20.

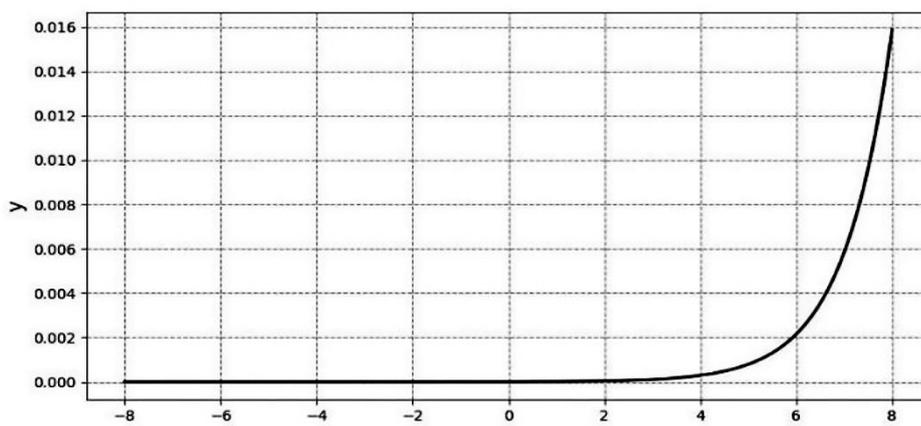


Figura 2.20 Función de activación Softmax (Salazar, 2020)

Una vez que hemos aplicado la función de activación a las entradas del nodo en cuestión, estamos listos para pasar el resultado a la siguiente capa de nuestra red neuronal. Hacemos

esto hasta llegar a la última capa, donde los resultados son en realidad las predicciones realizadas por toda la arquitectura de la red neuronal.

2.3.7.5 Propagación hacia adelante y hacia atrás.

La salida de las redes neuronales que hemos discutido hasta ahora se ha obtenido llevando las entradas de una capa a la siguiente. Este tipo de red neuronal se denomina red de avance y el flujo de información va en una sola dirección, sin permitir bucles ni ciclo (Salazar, 2020). Por lo tanto, la retroalimentación no es posible. ¿Qué sucede si la respuesta que obtenemos al entrenar nuestra red neuronal no es satisfactoria? En el caso de una red *feedforward*, como se describe arriba, no hay mucho que podamos hacer.

No obstante, es posible considerar el siguiente escenario: una vez que hemos obtenido la salida final de nuestra red neuronal, podemos compararla con los datos etiquetados utilizados para el entrenamiento. Si el error es insignificante, hemos terminado; sin embargo, si el error no es aceptable, nos gustaría proporcionarlo como retroalimentación a la red neuronal. En otras palabras, nos gustaría que la red neuronal “aprendiera de sus errores” (Salazar, 2020).

El proceso para habilitar esta forma de aprendizaje en una red neuronal se conoce como propagación hacia atrás de errores, o retropropagación para abreviar. Siguiendo con la analogía de aprender de nuestros errores, en cierto sentido le estamos pidiendo a la máquina que adivine el valor de los datos etiquetados. El error en la conjetura se calcula y se propaga hacia atrás para que se pueda hacer una mejor estimación (Salazar, 2020).

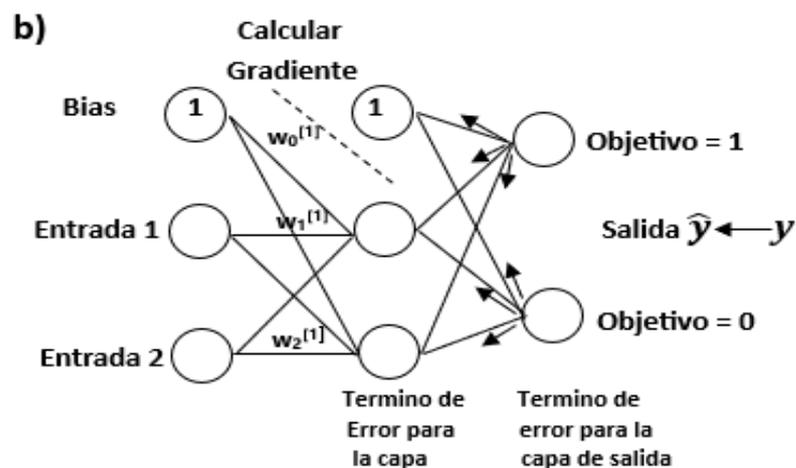
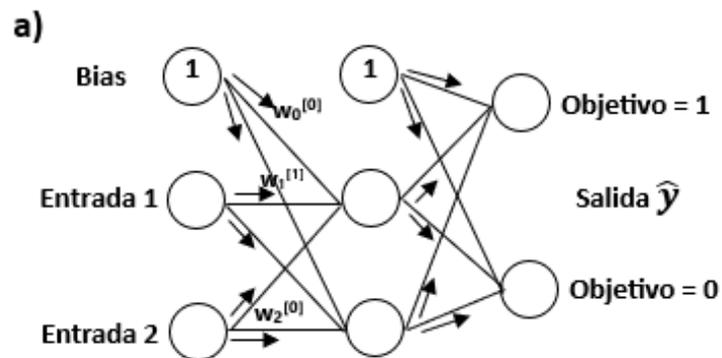
De esta forma, vamos adelante y atrás, y adelante de nuevo hasta que el error esté dentro de un nivel de tolerancia aceptado. La forma en que la retropropagación estima el error es minimizando una función de costo y, por lo tanto, necesitamos hacer uso del cálculo. Un método de optimización muy conocido utilizado en este tipo de tareas es el descenso de gradiente.

Comencemos con nuestra red de propagación directa como se muestra en la Figura 2.21, en el panel a) tenemos la situación descrita en la sección anterior. Para nuestros propósitos, tenemos una red neuronal con una capa oculta y tres nodos en la capa de entrada.

Necesitamos juntar las entradas (y el sesgo) con los pesos iniciales w_0 , w_1 y w_2 y hacer los cálculos apropiados con ellos. Para realizar un seguimiento de lo que sucede con los pesos, agregamos el superíndice [0] para indicar el paso inicial hacia adelante. De esta forma, los pesos $w_i^{[0]}$ y las entradas se combinan en los nodos ocultos por una determinada

función de activación. Luego, los resultados se pasan a la capa de salida, donde obtenemos una predicción objetivo, en este caso, la salida es un 1 o un 0. Tenga en cuenta que en la Figura 2.21 solo estamos etiquetando los pesos que van desde la entrada hasta la primera capa oculta. Los otros aspectos de la arquitectura también tienen su propio peso.

Ahora estamos interesados en verificar si las etiquetas de predicción obtenidas del pase hacia adelante número 0 son buenas. Podemos denotar el resultado de salida como \hat{y} i compararlo con el objetivo real y . El resultado de la comparación entre y i \hat{y} contiene información útil para la red neuronal, ya que le permite aprender de sus errores (Salazar, 2020).



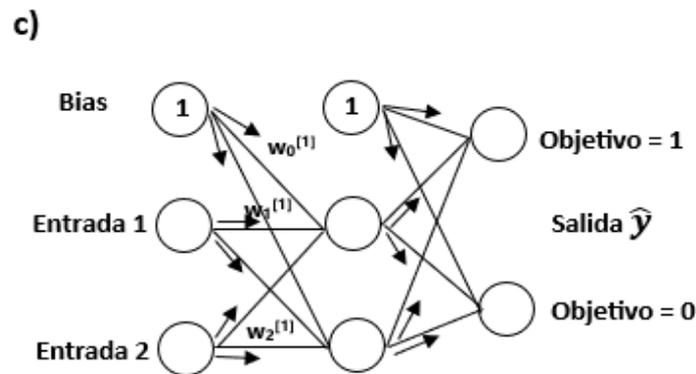


Figura 2.21(a, b, c). Propagación hacia atrás de errores, o backpropagation, permite que la red neuronal aprenda de sus errores (Salazar, 2020).

Para hacer esto, hacemos uso de una función de pérdida (o costo), que nos permite evaluar qué tan bien nuestro algoritmo elegido modela los datos de entrenamiento proporcionados. Si nuestras predicciones están fuera de lugar, la función de pérdida devolverá un valor más alto. Sin embargo, si son buenos, el resultado de la función de pérdida será un número menor (Salazar, 2020).

Es posible propagar esta información desde la capa de salida hacia los nodos ocultos, hasta que alcance los bordes de la capa de entrada. De esta forma es posible que la red neuronal identifique los pesos que es necesario ajustar para mejorar las predicciones realizadas. Por un lado, en los casos en que la predicción es diferente a la etiqueta objetivo, la red neuronal puede ajustar el peso que hizo esta predicción y mejorar el resultado. Por otro lado, para las etiquetas que se han pronosticado correctamente, no es necesario realizar ningún ajuste. Para propagar la información, estamos resolviendo un problema de optimización para la función de pérdida elegida (Salazar, 2020).

Como se mencionó anteriormente, se puede usar un algoritmo bien conocido como el descenso de gradiente para el paso de optimización, y requiere que podamos calcular los gradientes. El paso de retropropagación se representa en la Figura 2.21, panel b).

Ahora estamos en la situación que se muestra en el panel c) de la Figura 2.21, donde llegamos al comienzo de nuestro ciclo de aprendizaje. Ahora que hemos pasado información sobre los pesos, la red neuronal puede hacer ajustes y los nuevos pesos nos permiten comenzar un nuevo paso hacia adelante. Este paso comienza y sigue exactamente la misma lógica que antes, excepto que esta vez tenemos nuevos valores

$w_0^{[1]}$, $w_1^{[1]}$ y $w_2^{[1]}$. Observe que estamos usando el superíndice 1 para indicar el hecho de que tenemos una nueva iteración (Salazar, 2020).

De aquí en adelante, el proceso continúa como antes: usamos los pesos y las entradas para hacer cálculos usando la función de activación, y pasamos los resultados a la siguiente capa. Cuando llegamos a la capa de salida, los nuevos resultados de predicción se comparan con los objetivos. El hecho de que estemos usando un proceso de minimización indica que esperamos valores cada vez más bajos de nuestra función de pérdida. Continuamos iterando sobre nuestra red neuronal enviando información de un lado a otro hasta que alcancemos una tolerancia elegida en los valores de nuestra función de pérdida. En ese punto, podemos detener las iteraciones y tenemos una red neuronal que ha aprendido de sus errores al hacer predicciones al comparar con las etiquetas objetivo proporcionadas (Salazar, 2020).

2.3.8 Evaluación de Modelos.

La evaluación de modelos es el proceso de comprender qué tan bien dan la clasificación correcta y luego medir el valor de la predicción en un contexto determinado. A veces, solo nos importa la frecuencia con la que un modelo obtiene una predicción correcta; otras veces, es importante que el modelo obtenga cierto tipo de predicción correcta con más frecuencia que las demás. La herramienta básica para evaluar modelos es la matriz de confusión (Adam, 2017).

2.3.8.1 Matriz de confusión.

La matriz de confusión (Figura 2.22), es una tabla de filas y columnas que representa las predicciones y los resultados reales (etiquetas) para un clasificador. Usamos esta tabla para comprender mejor qué tan bien se está desempeñando el modelo o clasificador en función de dar la respuesta correcta en el momento adecuado (Adam, 2017).

	P' (Predecir)	N' (Predecir)
P (ACTUAL)	VERDADERO POSITIVO (TP)	FALSO NEGATIVO (FN)
N (ACTUAL)	FALSO POSITIVO (FP)	VERDADERO NEGATIVO (TN)

Figura 2.22 La matriz de confusión (Adam, 2017)

Medimos estas respuestas contando el número de los siguientes:

- Verdaderos positivos
 - Predicción positiva
 - La etiqueta fue positiva
- Falsos positivos
 - Predicción positiva
 - La etiqueta era negativa
- Verdaderos negativos
 - Predicción negativa
 - La etiqueta era negativa
- Falsos negativos
 - Predicción negativa
 - La etiqueta fue positiva

Podemos calcular diferentes evaluaciones del modelo basándonos en combinaciones de los cuatro recuentos antes mencionados en la matriz de confusión, tal como se detalla a continuación.

- Exactitud (*Accuracy*)
- Precisión
- Recuperación (*Recall*)
- Puntaje F1

Exactitud (Accuracy): La exactitud es el grado de cercanía de las mediciones de una cantidad al valor real de esa cantidad.

$$\text{Exactitud} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{FN} + \text{TN})$$

La exactitud puede ser engañosa en la calidad del modelo cuando el desequilibrio de clases es alto. Si simplemente clasificamos todo como la clase más grande, nuestro modelo obtendrá automáticamente un gran número de conjeturas correctas y nos proporcionará una puntuación de alta exactitud, pero una indicación de valor engañosa basada en una aplicación real del modelo (por ejemplo, nunca predecirá la clase más pequeña o evento raro) (Adam, 2017).

Precisión: El grado en que las mediciones repetidas en las mismas condiciones nos dan los mismos resultados se llama precisión en el contexto de la ciencia y la estadística. La precisión también se conoce como el valor de predicción positivo.

$$\text{Presición} = \text{TP} / (\text{TP} + \text{FP})$$

Una medición puede ser exacta pero no precisa, no exacta pero todavía precisa, ni exacta ni precisa, o exacta y precisa a la vez. Consideramos que una medida es válida si es exacta y precisa (Adam, 2017).

Recuperación (*Recall*): La verdadera tasa positiva mide la frecuencia con la que clasificamos un registro de entrada como clase positiva y es la clasificación correcta. Esto también se llama sensibilidad o recuerdo. La sensibilidad cuantifica qué tan bien el modelo evita los falsos negativos (Adam, 2017).

$$\text{Sensibilidad} = \text{TP} / (\text{TP} + \text{FN})$$

Puntaje F1: En la clasificación binaria, consideramos que la puntuación F1 (o puntuación F, medida F) es una medida de la precisión de un modelo. La puntuación F1 es la media armónica de las medidas de precisión y recuperación (descritas anteriormente) en una sola puntuación, como se define aquí:

$$\text{F1} = 2\text{TP} / (2\text{TP} + \text{FP} + \text{FN})$$

Vemos puntajes para F1 entre 0.0 y 1.0, donde 0.0 es el peor puntaje y 1.0 es el mejor puntaje que nos gustaría ver. La puntuación F1 se usa normalmente en la recuperación de información para ver qué tan bien un modelo recupera resultados relevantes. En el aprendizaje automático, vemos que la puntuación F1 se usa como una puntuación general sobre el rendimiento de nuestro modelo (Adam, 2017).

2.3.8.2 Área Bajo la Curva ROC.

De (Rogers Simón, 2017) la curva característica operativa del receptor (ROC) nos permite examinar cómo varía el rendimiento de un modelo a medida que cambiamos un umbral que por encima del cual se predice todo como positivos y por debajo como negativos. La sensibilidad y la especificidad se calculan para un rango de valores de umbral, y la sensibilidad se representa frente a la especificidad complementaria o la tasa de falsos positivos ($1 - Sp$), dando una curva que normalmente se verá como las que se muestran en la Figura 2.23.

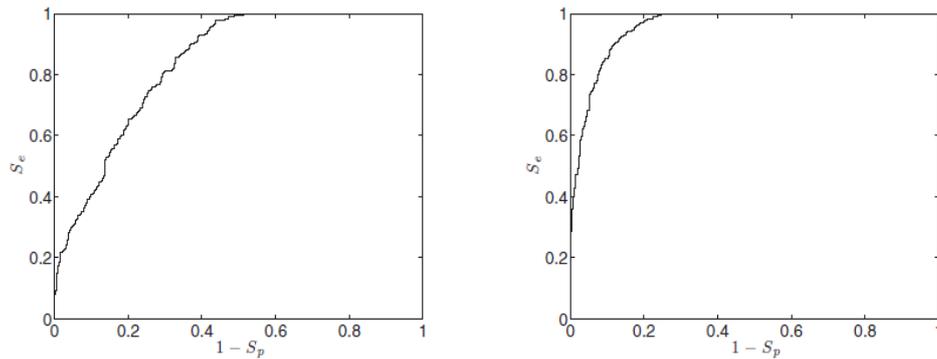


Figura 2.23 Ejemplos de curvas ROC (Rogers Simón, 2017)

Lo que se quiere es que S_e y S_p sean lo más alto posible. Por lo tanto, cuanto más se acerque la curva a la parte superior izquierda de la gráfica ($S_e = 1$; $1 - S_p = 0$), es mejor. Si la curva toca la esquina superior izquierda, nos dice que hay un umbral que podemos elegir que clasificará los datos perfectamente. La curva siempre comenzará en $S_e = 0$; $1 - S_p = 0$, correspondiente a un umbral que nunca clasifica nada como perteneciente a la clase 1, y finaliza en $S_e = 1$; $1 - S_p = 1$, correspondiente a un clasificador que nunca clasifica nada como perteneciente a la clase 0. A medida que el clasificador empeora, la curva se acercará a una línea recta de $(0; 0)$ a $(1; 1)$. Esto es equivalente a adivinar aleatoriamente la clase.

2.3.9 Deep Learning.

El aprendizaje profundo es un subcampo específico del aprendizaje automático, una nueva forma de aprender representaciones a partir de datos que pone énfasis en aprender capas sucesivas de representaciones cada vez más significativas. Lo profundo en el aprendizaje profundo no es una referencia a ningún tipo de comprensión más profunda lograda por el enfoque; más bien, representa esta idea de capas sucesivas de representaciones. La cantidad de capas que contribuyen a un modelo de los datos se denomina profundidad del modelo (Chollet, 2018).

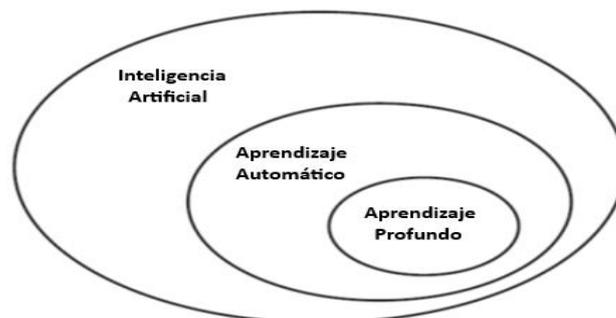


Figura 2.24 Inteligencia Artificial, Aprendizaje Automático y Aprendizaje profundo (Chollet, 2018).

El aprendizaje profundo moderno a menudo implica decenas o incluso cientos de capas sucesivas de representaciones, y todas se aprenden automáticamente a partir de la exposición a los datos de entrenamiento. Mientras tanto, otros enfoques del aprendizaje automático tienden a centrarse en aprender solo una o dos capas de representaciones de los datos; por lo tanto, a veces se les llama aprendizaje superficial.

En el aprendizaje profundo, estas representaciones en capas se aprenden (casi siempre) a través de modelos llamados redes neuronales, estructurados en capas literales apiladas una encima de la otra. El término red neuronal es una referencia a la neurobiología, pero, aunque algunos de los conceptos centrales del aprendizaje profundo se desarrollaron en parte inspirándose en nuestra comprensión del cerebro, los modelos de aprendizaje profundo no son modelos del cerebro. No hay evidencia de que el cerebro implemente algo parecido a los mecanismos de aprendizaje utilizados en los modelos modernos de aprendizaje profundo.

Para nuestros propósitos, el aprendizaje profundo es un marco matemático para aprender representaciones a partir de datos (Chollet, 2018).

2.3.9.1 Características del Deep Learning.

En comparación con los algoritmos tradicionales de aprendizaje automático y las redes neuronales superficiales, los algoritmos modernos de aprendizaje profundo suelen tener las siguientes características (Liangqu Long, 2022).

- **Volumen de Datos.**

Los primeros algoritmos de aprendizaje automático son relativamente simples y rápidos de entrenar, y el tamaño del conjunto de datos requerido es relativamente pequeño, como el conjunto de datos de flores Iris recopilado por el estadístico británico Ronald Fisher en 1936, que contiene solo tres categorías de flores, con cada categoría teniendo 50 muestras. Con el desarrollo de la tecnología informática, los algoritmos diseñados son cada vez más complejos y la demanda de volumen de datos también está aumentando.

Aunque el aprendizaje profundo tiene una gran demanda de grandes conjuntos de datos, la recopilación de datos, especialmente la recopilación de datos etiquetados, suele ser muy costosa. La formación de un conjunto de datos generalmente requiere la recopilación manual, el rastreo de datos sin procesar y la limpieza de muestras no válidas, y luego anotar las muestras de datos con inteligencia humana, por lo que

inevitablemente se introducen sesgos subjetivos y errores aleatorios (Liangqu Long, 2022).

- **Poder Computacional.**

El aumento de la potencia informática es un factor importante en el tercer renacimiento de la inteligencia artificial. De hecho, la teoría básica del aprendizaje profundo moderno se propuso en la década de 1980, pero el potencial real del aprendizaje profundo no se realizó hasta el lanzamiento de AlexNet basado en el entrenamiento en dos GPU GTX580 en 2012. Por lo general, el entrenamiento en serie en la CPU puede obtener resultados satisfactorios. Pero el aprendizaje profundo depende en gran medida de los dispositivos informáticos de aceleración paralela. La mayoría de las redes neuronales actuales utilizan chips de aceleración paralelos, como *NVIDIA* GPU y Google TPU, para entrenar los parámetros del modelo (Liangqu Long, 2022).

- **Escala de red.**

Los primeros modelos de perceptrón y las redes neuronales multicapa solo tienen una o dos a cuatro capas, y los parámetros de la red también rondan las decenas de miles. Con el desarrollo del aprendizaje profundo y la mejora de las capacidades informáticas, se han propuesto sucesivamente modelos como AlexNet (8 capas), VGG16 (16 capas), *GoogleNet* (22 capas), ResNet50 (50 capas) y DenseNet121 (121 capas), mientras que el tamaño de las imágenes de entrada también ha aumentado gradualmente de 28×28 a 224×224 a 299×299 e incluso más grandes. El aumento de la escala de la red mejora la capacidad de las redes neuronales correspondientemente, de modo que las redes pueden aprender modalidades de datos más complejas y el rendimiento del modelo puede mejorarse en consecuencia. Por otro lado, el aumento de la escala de la red también significa que necesitamos más datos de entrenamiento y potencia computacional para evitar el sobreajuste (Liangqu Long, 2022).

2.3.10 Redes Neuronales Convolucionales (CNN).

Una red neuronal convolucional es una red neuronal de propagación hacia adelante que generalmente se usa para analizar imágenes visuales mediante el procesamiento de datos con una topología similar a una cuadrícula. Una red neuronal convolucional se utiliza para detectar y clasificar objetos en una imagen (Mirza Rahim Baig, 2020).

2.3.10.1 *Kernels* y Filtros.

Uno de los componentes principales de las CNN son los filtros, que son matrices cuadradas que tienen dimensiones $n_K \times n_K$, donde n_K es un número entero y suele ser un número pequeño, como 3 o 5. A veces, los filtros también se denominan núcleos. El uso de *kernels* proviene de las técnicas clásicas de procesamiento de imágenes (Michelucci, 2019).

- El siguiente núcleo permitirá la detección de bordes horizontales.

$$J_H = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}$$

- El siguiente núcleo permitirá la detección de bordes verticales

$$J_V = \begin{pmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{pmatrix}$$

- El siguiente *kernel* permitirá la detección de bordes cuando la luminosidad cambie drásticamente.

$$J_L = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

- El siguiente núcleo desenfocará los bordes de una imagen.

$$J_B = -\frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

2.3.10.2 Convolución.

El primer paso para comprender las CNN es comprender la convolución. Primero, en el contexto de las redes neuronales, la convolución se realiza entre tensores. La operación

obtiene dos tensores como entrada y produce un tensor como salida. La operación se suele indicar con el operador *.

Considerando dos tensores, ambos con dimensiones 3×3. La operación de convolución se realiza aplicando la siguiente fórmula (Michelucci, 2019):

$$\begin{pmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{pmatrix} * \begin{pmatrix} k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 \\ k_7 & k_8 & k_9 \end{pmatrix} = \sum_{i=1}^9 a_i k_i$$

En el apartado anterior hablamos de núcleos, y la razón es que la convolución se suele hacer entre un tensor, que podemos indicar aquí con A, y un núcleo. Por lo general, los núcleos son pequeños, 3×3 o 5×5, mientras que los tensores de entrada A normalmente son más grandes. En el reconocimiento de imágenes, por ejemplo, los tensores de entrada A son las imágenes que pueden tener dimensiones tan altas como 1024 × 1024 × 3, donde 1024 × 1024 es la resolución y la última dimensión (3) es el número de canales RGB (Michelucci, 2019).

Para entender cómo aplicar la convolución cuando tenemos matrices con diferentes dimensiones, consideremos una matriz A que es 4 × 4

$$A = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 \\ a_5 & a_6 & a_7 & a_8 \\ a_9 & a_{10} & a_{11} & a_{12} \\ a_{13} & a_{14} & a_{15} & a_{16} \end{pmatrix}$$

Y un *Kernel* K que tomaremos para este ejemplo como 3×3.

$$K = \begin{pmatrix} k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 \\ k_7 & k_8 & k_9 \end{pmatrix}$$

La idea es comenzar en la esquina superior izquierda de la matriz A y seleccionar una región de 3 × 3. En el ejemplo sería.

$$A_1 = \begin{pmatrix} a_1 & a_2 & a_3 \\ a_5 & a_6 & a_7 \\ a_9 & a_{10} & a_{11} \end{pmatrix}$$

Alternativamente, los elementos marcados en **negrita** aquí:

$$A = \begin{pmatrix} \mathbf{a_1} & \mathbf{a_2} & \mathbf{a_3} & a_4 \\ \mathbf{a_5} & \mathbf{a_6} & \mathbf{a_7} & a_8 \\ \mathbf{a_9} & \mathbf{a_{10}} & \mathbf{a_{11}} & a_{12} \\ a_{13} & a_{14} & a_{15} & a_{16} \end{pmatrix}$$

Luego realizamos la convolución, como se explicó al principio entre esta matriz más pequeña A_1 y K , obteniendo (indicaremos el resultado con B_1):

$$B_1 = A_1 * K = a_1k_1 + a_2k_2 + a_3k_3 + k_4a_5 + k_5a_5 + k_6a_7 + k_7a_9 + k_8a_{10} + k_9a_{11}$$

Luego, debemos cambiar la región seleccionada de 3×3 en la matriz A de una columna a la derecha y seleccionar los elementos marcados en negrita aquí:

$$A = \begin{pmatrix} a_1 & \mathbf{a_2} & \mathbf{a_3} & \mathbf{a_4} \\ a_5 & \mathbf{a_6} & \mathbf{a_7} & \mathbf{a_8} \\ a_9 & \mathbf{a_{10}} & \mathbf{a_{11}} & \mathbf{a_{12}} \\ a_{13} & a_{14} & a_{15} & a_{16} \end{pmatrix}$$

Esto nos dará la segunda submatriz A_2 :

$$A_2 = \begin{pmatrix} a_2 & a_3 & a_4 \\ a_6 & a_7 & a_8 \\ a_{10} & a_{11} & a_{12} \end{pmatrix}$$

Luego realizamos la convolución entre esta matriz más pequeña A_2 y K nuevamente:

$$B_2 = A_2 * K = a_2k_1 + a_3k_2 + a_4k_3 + a_6k_4 + a_7k_5 + a_8k_6 + a_{10}k_7 + a_{11}k_8 + a_{12}k_9$$

Ya no podemos desplazar nuestra región 3×3 hacia la derecha, ya que hemos llegado al final de la matriz A , así que lo que hacemos es desplazarla una fila hacia abajo y empezar de nuevo desde el lado izquierdo. La siguiente región seleccionada sería.

$$A_3 = \begin{pmatrix} a_5 & a_6 & a_7 \\ a_9 & a_{10} & a_{11} \\ a_{13} & a_{14} & a_{15} \end{pmatrix}$$

Nuevamente, realizamos la convolución de A_3 con K .

$$B_3 = A_3 * K = a_5k_1 + a_6k_2 + a_7k_3 + a_9k_4 + a_{10}k_5 + a_{11}k_6 + a_{13}k_7 + a_{14}k_8 + a_{15}k_9$$

El último paso es cambiar nuestra región seleccionada de 3×3 a la derecha de una columna y realizar la convolución nuevamente. Nuestra región seleccionada ahora será:

$$A_4 = \begin{pmatrix} a_6 & a_7 & a_{87} \\ a_{10} & a_{11} & a_{12} \\ a_{14} & a_{15} & a_{16} \end{pmatrix}$$

Además, la convolución dará este resultado:

$$B_4 = A_4 * K = a_6k_1 + a_7k_2 + a_8k_3 + a_{10}k_4 + a_{11}k_5 + a_{12}k_6 + a_{14}k_7 + a_{15}k_8 + a_{16}k_9$$

Ahora ya no podemos desplazar nuestra región 3×3 , ni a la derecha ni hacia abajo. Hemos calculado cuatro valores: B1, B2, B3 y B4. Esos elementos formarán el tensor resultante de la operación de convolución dándonos el tensor B (Michelucci, 2019):

$$B = \begin{pmatrix} B_1 & B_2 \\ B_3 & B_4 \end{pmatrix}$$

El mismo proceso se puede aplicar cuando el tensor A es mayor. Simplemente se obtendrá un tensor B resultante más grande, pero el algoritmo para obtener los elementos Bi es el mismo.

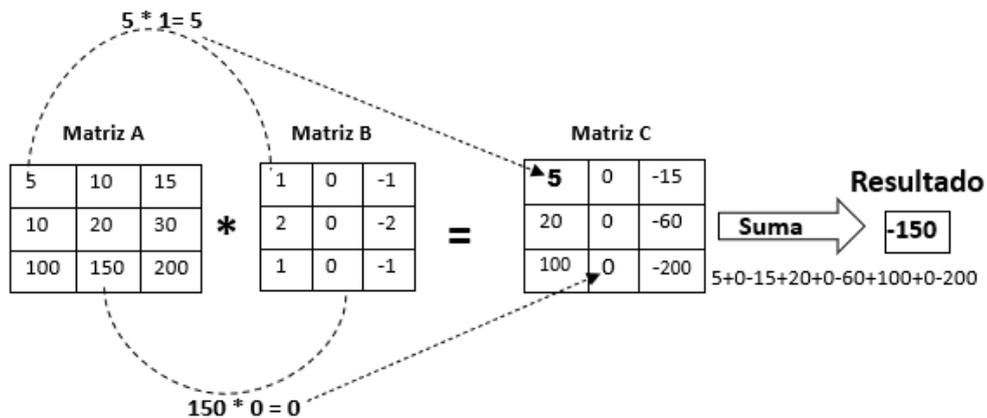


Figura 2.25 Operación de convolución (Mirza Rahim Baig, 2020).

Hay un pequeño detalle que se debe discutir, y es el concepto de *salto*. En el proceso anterior, se movió la región 3×3 siempre una columna a la derecha y una fila hacia abajo. El número de filas y columnas, en este ejemplo fue 1, se llama *salto* (*stride*) y a menudo se indica con *s*. Un salto $s = 2$ significa simplemente que desplazamos nuestra región de 3×3 dos columnas a la derecha y dos filas hacia abajo en cada paso.

Otra cosa que se debe discutir es el tamaño de la región seleccionada en la matriz de entrada A. Las dimensiones de la región seleccionada que cambiamos en el proceso deben ser las mismas que las del núcleo utilizado. Si usa un núcleo de 5×5 , deberá seleccionar una región de 5×5 en A. En general, dado un núcleo de $n_K \times n_K$, se debe seleccionar una región de $n_K \times n_K$ en A.

En una definición más formal, la convolución con *salto* *s* en el contexto de la red neural es un proceso que toma un tensor A de dimensiones $n_A \times n_A$ y un kernel K de dimensiones $n_K \times n_K$ y da como salida una matriz B de dimensiones $n_B \times n_B$ (Michelucci, 2019).

$$n_B = \frac{n_A - n_K}{S} + 1$$

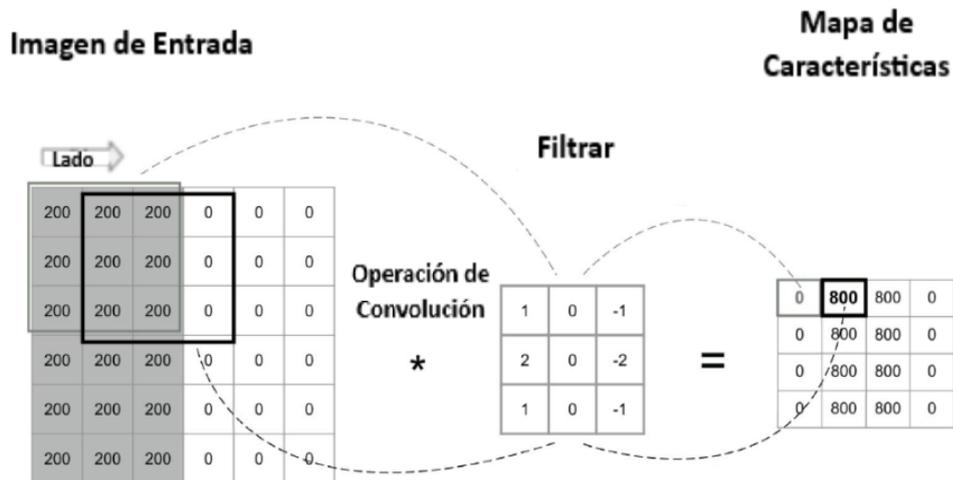


Figura 2.26 Ejemplo de salto (stride) (Mirza Rahim Baig, 2020).

2.3.10.3 Agrupación Máxima (*Pooling*).

El *pooling* es la segunda operación fundamental en las CNN. Para entenderlo, veamos un ejemplo concreto y consideremos lo que se llama agrupación máxima. Considerando nuevamente la matriz de 4×4 que vimos durante nuestra discusión sobre la convolución (Michelucci, 2019):

$$A = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 \\ a_5 & a_6 & a_7 & a_8 \\ a_9 & a_{10} & a_{11} & a_{12} \\ a_{13} & a_{14} & a_{15} & a_{16} \end{pmatrix}$$

Para realizar la agrupación máxima, necesitamos definir una región de tamaño $n_K \times n_K$, análoga a lo que hicimos para la convolución. Consideremos $n_K = 2$. Lo que debemos hacer es comenzar en la esquina superior izquierda de nuestra matriz A y seleccionar una región $n_K \times n_K$, en nuestro caso 2×2 de A. Aquí seleccionaríamos:

$$\begin{pmatrix} a_1 & a_2 \\ a_5 & a_6 \end{pmatrix}$$

Estos son los elementos marcados en **negrita** en la matriz A que se muestran a continuación:

$$A = \begin{pmatrix} \mathbf{a_1} & \mathbf{a_2} & a_3 & a_4 \\ \mathbf{a_5} & \mathbf{a_6} & a_7 & a_8 \\ a_9 & a_{10} & a_{11} & a_{12} \\ a_{13} & a_{14} & a_{15} & a_{16} \end{pmatrix}$$

De los elementos seleccionados, a_1 , a_2 , a_5 y a_6 , la operación de agrupación máxima selecciona el valor máximo. El resultado se indica con B1.

$$B_1 = \max_{i=1,2,5,6} a_i$$

Luego, debemos desplazar nuestra ventana de 2×2 dos columnas a la derecha, normalmente el mismo número de columnas que tiene la región seleccionada, y seleccionar los elementos marcados en negrita:

$$A = \begin{pmatrix} a_1 & a_2 & \mathbf{a_3} & \mathbf{a_4} \\ a_5 & a_6 & \mathbf{a_7} & \mathbf{a_8} \\ a_9 & a_{10} & a_{11} & a_{12} \\ a_{13} & a_{14} & a_{15} & a_{16} \end{pmatrix}$$

O, en otras palabras, la matriz más pequeña.

$$\begin{pmatrix} \mathbf{a_3} & \mathbf{a_4} \\ \mathbf{a_7} & \mathbf{a_8} \end{pmatrix}$$

El algoritmo *max-pooling* seleccionará entonces el máximo de los valores y dará un resultado que indicaremos con B2.

$$B_2 = \max_{i=3,4,7,8} a_i$$

En este punto ya no podemos mover la región 2×2 hacia la derecha, así que la movemos dos filas hacia abajo y comenzamos el proceso nuevamente desde el lado izquierdo de A, seleccionando los elementos marcados en negrita y obteniendo el máximo y llamándolo B3.

$$A = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 \\ a_5 & a_6 & a_7 & a_8 \\ \mathbf{a_9} & \mathbf{a_{10}} & a_{11} & a_{12} \\ \mathbf{a_{13}} & \mathbf{a_{14}} & a_{15} & a_{16} \end{pmatrix}$$

El salto s en este contexto tiene el mismo significado que ya hemos discutido en convolución. Es simplemente la cantidad de filas o columnas que se mueve la región al seleccionar los elementos. Finalmente, seleccionamos la última región 2×2 en la parte inferior de A, seleccionando los elementos a_{11} , a_{12} , a_{15} y a_{16} . Luego obtenemos el

máximo y lo llamamos B4. Con los valores que obtengamos en este proceso, en el ejemplo los cuatro valores B1, B2, B3 y B4, construiremos un tensor de salida (Michelucci, 2019):

$$B = \begin{pmatrix} B_1 & B_2 \\ B_3 & B_4 \end{pmatrix}$$

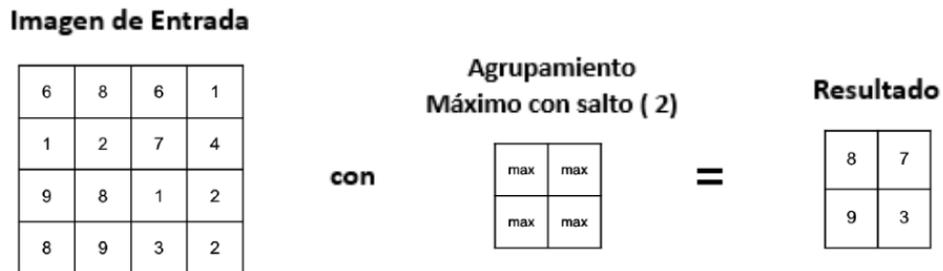


Figura 2.27 Agrupación máxima con stride 2 en una imagen de entrada

2.3.10.4 Relleno (*Padding*).

A veces, cuando se trata de imágenes, no es óptimo obtener un resultado de una operación de convolución que tenga dimensiones diferentes a la imagen original. Es cuando el relleno es necesario. La idea es sencilla: agrega filas de píxeles en la parte superior e inferior y columnas de píxeles a la derecha e izquierda de las imágenes finales para que las matrices resultantes tengan el mismo tamaño que el original. Algunas estrategias rellenan los píxeles añadidos con ceros, con los valores de los píxeles más cercanos, etc. Solo como referencia, en caso de que se use *padding*, si p es el ancho de las filas y columnas que usa como *padding*, las dimensiones finales de la matriz B , en caso de convolución y agrupación, están dadas por:

$$n_B = \left\lfloor \frac{n_A + 2p - n_K + 1}{s} \right\rfloor$$

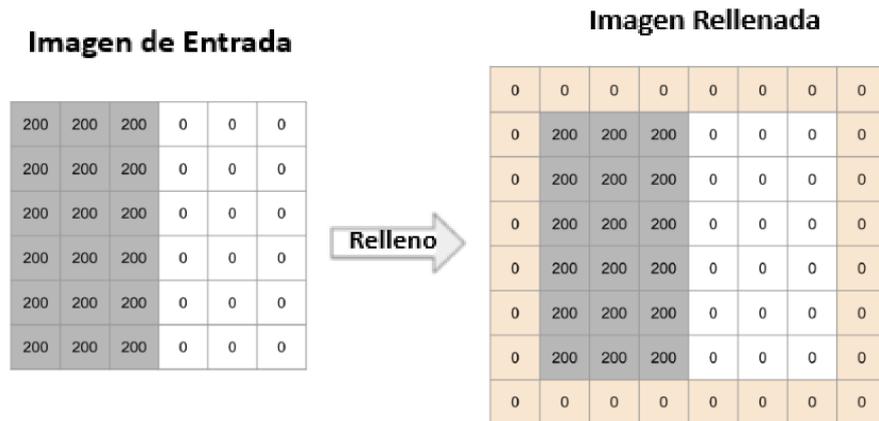


Figura 2.28 Ejemplo de relleno (Mirza Rahim Baig, 2020).

2.3.10.5 Arquitectura de una Red Neuronal Convolutiva.

De acuerdo a Michelucci (2019), una red neuronal convolutiva tiene múltiples capas ocultas que ayudan a extraer información de una imagen. Las tres capas importantes de una CNN son:

- Capa convolutiva.
- Capa de agrupación máxima,
- Capa completamente conectada.

- **Capa Convolutiva.**

De (Michelucci, 2019), una capa convolutiva toma como entrada un tensor (que puede ser tridimensional, debido a los tres canales de color), por ejemplo, una imagen, aplica una cierta cantidad de núcleos, normalmente 10, 16 o incluso más, agrega un sesgo, aplica la función de activación *RELU* para introducir no linealidad al resultado de la convolución, y produce una matriz conteniendo mapas de características (ver Fig. 2.29).

Las capas convolutivas en una red neuronal convolutiva aplican sistemáticamente filtros aprendidos a las imágenes de entrada para crear mapas de características que resuman la presencia de esas características en la entrada.

Las capas convolutivas resultan muy efectivas, y el apilamiento de las capas convolutivas en modelos profundos permite que las capas cercanas a la entrada aprendan características de bajo nivel (por ejemplo, líneas) y capas más profundas en el modelo para aprender características de alto orden o más abstractas, como formas u objetos específicos.

- **Capa de Agrupación.**

En (Michelucci, 2019), una capa de agrupación es una nueva capa agregada después de la capa convolucional. Específicamente, después de que se aplica una función de activación o no linealidad (por ejemplo, *RELU*) a la salida de mapas de características por una capa convolucional

La adición de una capa de agrupación después de la capa convolucional es un patrón común utilizado para ordenar capas dentro de una red neuronal convolucional que puede repetirse una o más veces en un modelo dado.

La capa de agrupación funciona en cada mapa de características por separado para crear un nuevo conjunto del mismo número de mapas de características agrupados.

Implica seleccionar una operación de agrupación, al igual que un filtro que se aplicará a los mapas de características. El tamaño de la operación o filtro de agrupación es más pequeño que el tamaño del mapa de características; Específicamente, casi siempre es de 2×2 píxeles aplicados con un paso de 2 píxeles.

Esto significa que la capa de agrupación siempre reducirá el tamaño de cada mapa de características en un factor de 2, p. Cada dimensión se reduce a la mitad, reduciendo el número de píxeles o valores en cada mapa de características a un cuarto del tamaño. Por ejemplo, una capa de agrupación aplicada a un mapa de características de 6×6 (36 píxeles) dará como resultado un mapa de características agrupado de salida de 3×3 (9 píxeles) (ver Fig.2.29).

- **Capa Completamente Conectada.**

De (Michelucci, 2019), la capa totalmente conectada (también conocida como capa oculta) es la última capa en la red neuronal convolucional.

Se puede agregar múltiples capas de este tipo en función de la profundidad a la que se quiere tomar el modelo de clasificación. Se debe tener en cuenta que esto depende completamente del conjunto de datos de entrenamiento. La salida de la capa oculta final se envía a la función *Softmax* o sigmoidea para la distribución de probabilidad sobre el conjunto final del número total de clases (ver Figura. 2.29).

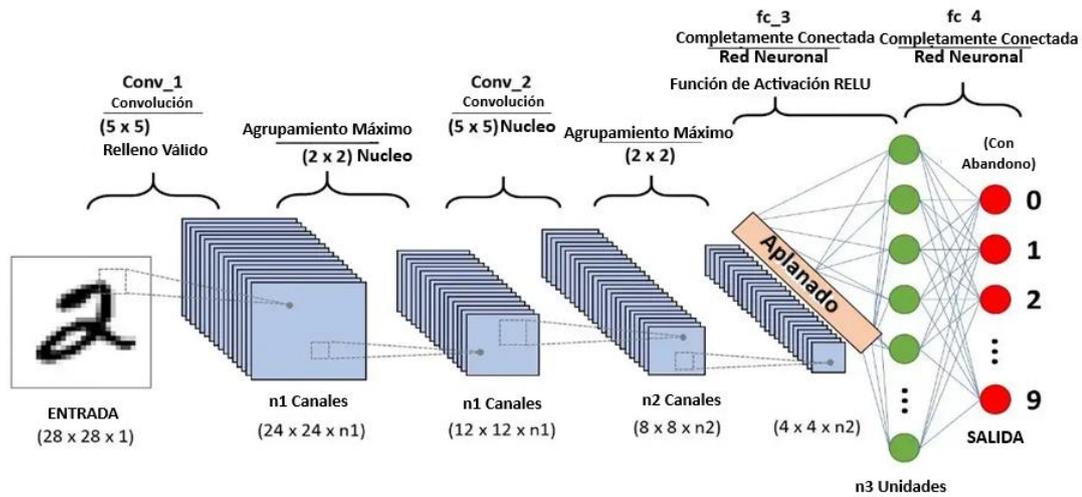


Figura 2.29 Red Neuronal Convolutiva (Michelucci, 2019)

2.3.11 Transferencia de Aprendizaje.

En (Vasilev, 2019) la Transferencia de Aprendizaje es el proceso de aplicar un modelo Machine Learning entrenado existente a un problema nuevo, pero relacionado. Por ejemplo, podemos tomar una red entrenada en *ImageNet*⁴ y reutilizarla para clasificar imágenes correspondientes a otro problema que tenga relación. Alternativamente, podríamos usar un juego de simulación de conducción para entrenar una red neuronal para conducir un automóvil simulado y luego usar la red para conducir un automóvil real. La Transferencia de aprendizaje es un concepto general que se aplica a todos los algoritmos de Machine Learning. En nuestro caso aplicaremos este concepto a las Redes Neuronales Convolucionales.

El escenario más común es tomar una red previamente entrenada con *ImageNet*, pero podría ser cualquier conjunto de datos. *TensorFlow* y *PyTorch* tienen arquitecturas neuronales populares preentrenadas con *ImageNet* que se pueden usar. Alternativamente, podemos entrenar nuestra propia red con un conjunto de datos de nuestra elección.

Las capas totalmente conectadas al final de una CNN actúan como traductores entre el lenguaje de la red (las representaciones de características abstractas aprendidas durante

⁴ El proyecto *ImageNet* es una gran base de datos visual diseñada para su uso en la investigación de software de reconocimiento de objetos visuales. El proyecto ha anotado a mano más de 14 millones de imágenes para indicar qué objetos se representan y en al menos un millón de imágenes, también se proporcionan cuadros delimitadores. *ImageNet* contiene más de 20.000 categorías, (<https://www.image-net.org/>).

el entrenamiento) y nuestro lenguaje, que es la clase de cada muestra. Se puede pensar en transferencia de aprendizaje como una traducción a otro idioma.

Se comienza con las características de la red, que es la salida de la última capa convolucional o de agrupación. Luego, se traduce a un conjunto diferente de clases de la nueva tarea. Se puede hacer esto eliminando la última capa completamente conectada (o todas las capas completamente conectadas) de una red preentrenada existente y reemplazándola con otra capa, que representa las clases del nuevo problema (ver Figura. 2.30).

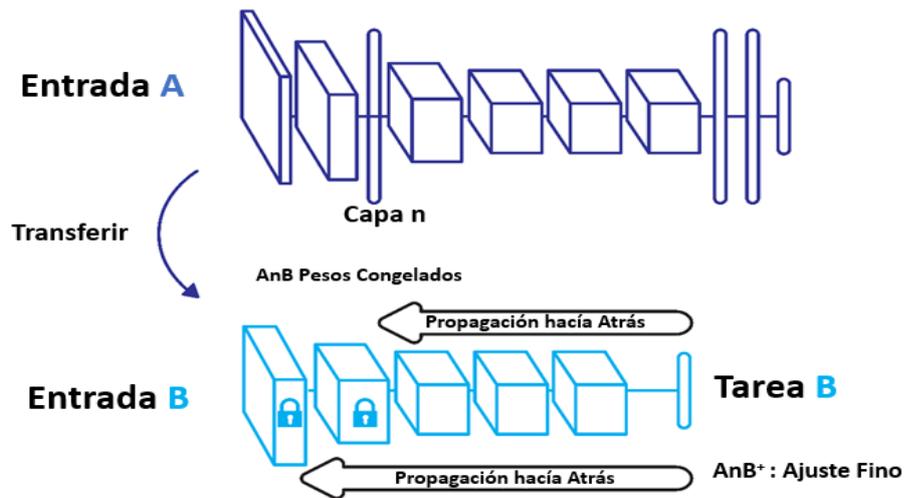


Figura 2.30 Transferencia de aprendizaje (Vasilev, 2019)

2.3.12 Aumento de datos.

Al realizar el aumento de datos, se aumenta la cantidad de imágenes de un conjunto de datos. Para una imagen se puede generar, por ejemplo, 10 variantes diferentes de la misma imagen. Entonces, el tamaño del conjunto de datos se multiplicará por 10 (Mirza Rahim Baig, 2020).

Además, con el aumento de datos, se tiene un conjunto de imágenes con una gama más amplia de elementos visuales. Al realizar el aumento de datos, ayuda a que el modelo generalice mejor a diferentes tipos de imágenes. Sin embargo, hay un inconveniente: el aumento de datos también aumentará el tiempo de entrenamiento, ya que se tendrá que realizar transformaciones de datos adicionales.

Algunos de los diferentes tipos de aumento de datos que se pueden hacer son:

- **Voltear horizontalmente.**

El giro horizontal devuelve una imagen volteada horizontalmente:



Figura 2.31 Ejemplo de voltear horizontalmente (Mirza Rahim Baig, 2020)

- **Voltear verticalmente.**

El giro vertical volteará una imagen verticalmente:

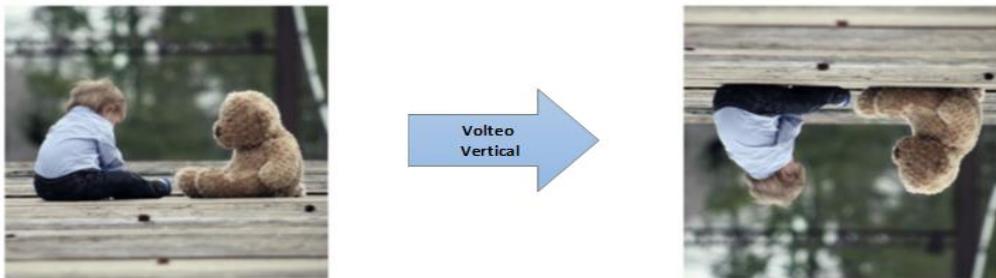


Figura 2.32 Ejemplo de voltear verticalmente (Mirza Rahim Baig, 2020)

- **Escalamiento.**

Se puede ampliar una imagen y proporcionar diferentes tamaños de objetos en la imagen:

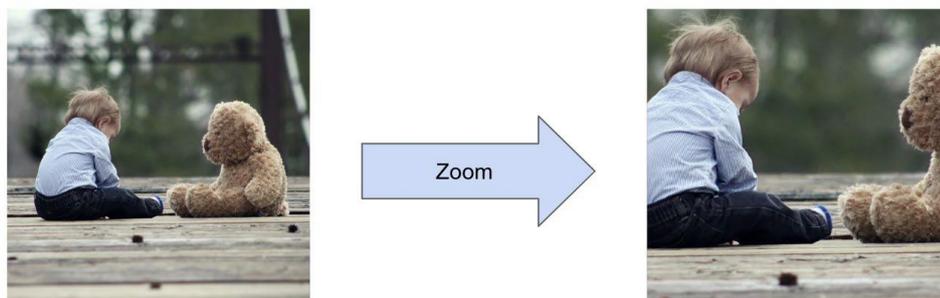


Figura 2.33 Ejemplo de escalamiento (Mirza Rahim Baig, 2020).

- **Desplazamiento horizontal.**

El desplazamiento horizontal, como su nombre lo indica, desplazará la imagen a lo largo del eje horizontal, pero la mantendrá del mismo tamaño. Con esta transformación, la imagen puede recortarse y es necesario generar nuevos píxeles para llenar el vacío. Una técnica común es copiar los píxeles vecinos o llenar ese espacio con píxeles negros:



Figura 2.34 Ejemplo de desplazamiento horizontal (Mirza Rahim Baig, 2020).

- **Desplazamiento vertical.**

El desplazamiento vertical es similar al desplazamiento horizontal, pero a lo largo del eje vertical:



Figura 2.35 Ejemplo de desplazamiento vertical (Mirza Rahim Baig, 2020)

- **Rotación.**

Se puede realizar una rotación con un ángulo particular en una imagen de la siguiente manera:

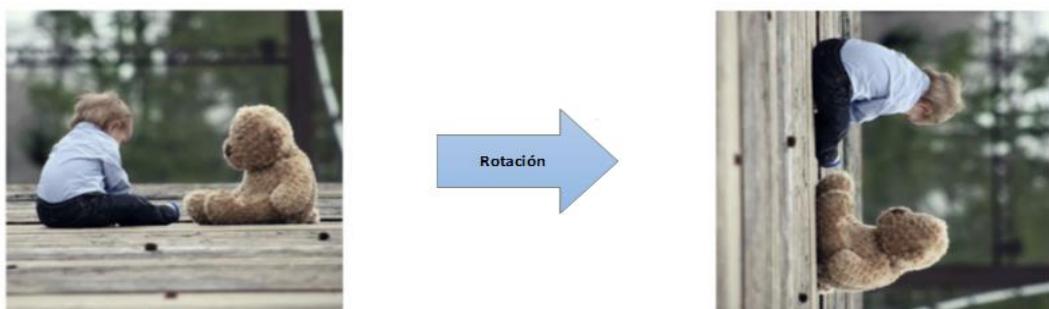


Figura 2.36 Ejemplo de rotación (Mirza Rahim Baig, 2020)

- **Corte.**

El corte transforma la imagen moviendo uno de los bordes a lo largo del eje del borde.

Después de hacer esto, la imagen se distorsiona de un rectángulo a un paralelogramo:

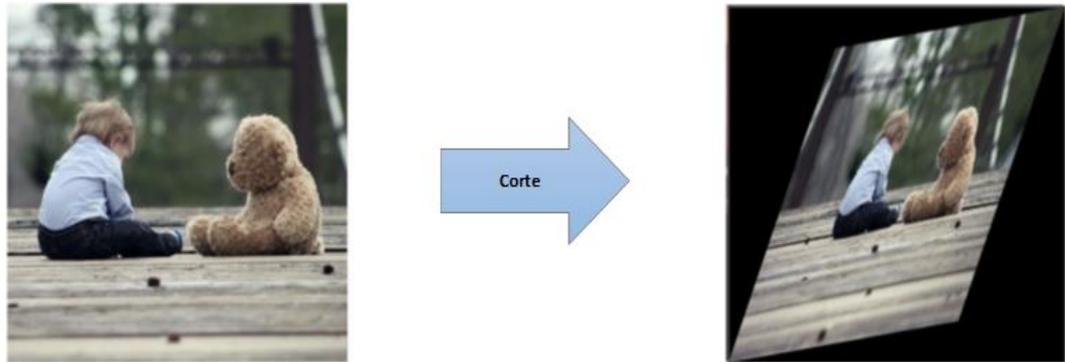


Figura 2.37 Ejemplo de corte (Mirza Rahim Baig, 2020)

2.3.13 Redes Neuronales Convolucionales Pre entrenadas.

En (Vasilev, 2019) estas redes suelen combinar múltiples operaciones primitivas de convolución y/o agrupación en un bloque de construcción novedoso que sirve como base para una arquitectura compleja. Esto nos permite construir redes muy profundas con un alto poder de representación que funcionan bien en tareas complejas como la clasificación de imágenes, la segmentación de imágenes, el reconocimiento de voz, etc. Existen varias redes neuronales convolucionales pre entrenadas con ImageNet y que pueden ser muy bien aprovechadas para resolver problemas de clasificación de imágenes mediante transferencia de aprendizaje, reemplazando sus últimas capas para extraer características del conjunto de datos en particular (ver Figura. 2.38). Estas son algunas de las principales redes pre entrenadas:

- AlexNet
- Visual Geometry Group (VGG)
- Residual Network (ResNet)
- Inception Networks
- Xception
- MobileNet
- DenseNet

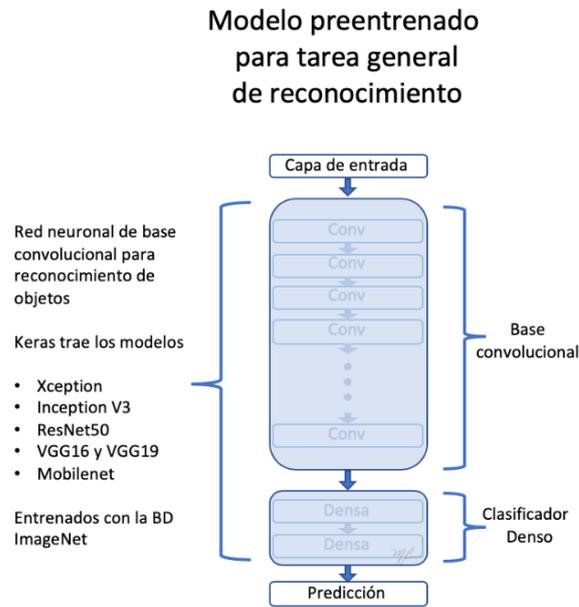


Figura 2.38 Modelo general de red pre entrenada para reconocimiento

<http://personal.cimat.mx:8181/~mrivera/cursos/aprendizaje/profundo/preentrenadas/preentrenadas.html>

2.3.13.1 MobileNet

Howard y colaboradores (Howard, 2017) presentaron una red ligera, MobileNet, para aplicaciones móviles. MobileNet reemplaza la operación de convolución 3x3 en CNN estándar con la operación de convolución profunda 3x3 y convolución puntual 1x1. El uso de una operación de convolución separable en profundidad en lugar de una operación de convolución estándar ayuda a reducir la cantidad de parámetros de entrenamiento. La diferencia entre la operación de convolución estándar y la convolución separable en profundidad utilizada en MobileNet se muestra en la Figura 2.39



Figura 2.39 Comparación de la operación de convolución estándar y la convolución separable en profundidad utilizada en Howar (2017)

2.3.13.2 DenseNet

Huang et al. (2018) presentó una red convolucional densamente conectada en la que cada capa estaba conectada a todas las demás capas, como se muestra en la Figura 2.40. En DenseNet, los mapas de características de todas las capas anteriores se utilizan como entradas para cada capa, y el mapa de características de esa capa se utiliza como una entrada para todas las capas que vienen después. La arquitectura *DenseNet* ayudó a resolver el problema del gradiente que desaparece y permitió la reutilización de funciones. Conectar cada capa entre sí también ayudó a fortalecer la propagación de características y disminuir la cantidad de parámetros necesarios para entrenar una red neuronal profunda.

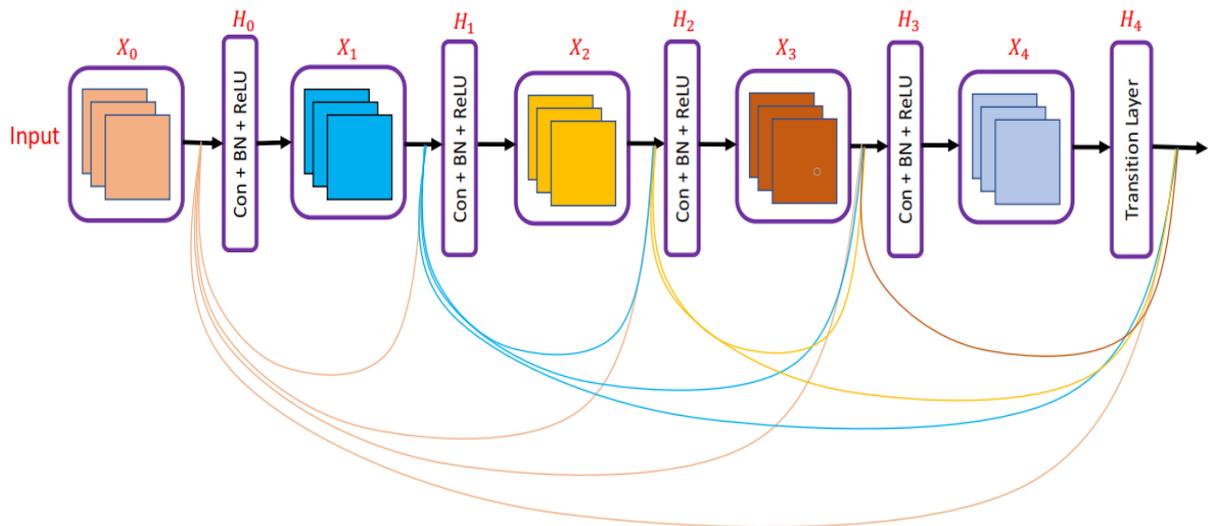


Figura 2.40 Red DenseNet presentada en Huang et. al. (2018)

2.3.13.3 Inception V3

En Vasilev (2019), es una arquitectura de red neuronal convolucional de la familia Incepción que realiza varias mejoras, incluido el uso de suavizado de etiquetas, convolucional factorizadas 7×7 y el uso de un clasificador auxiliar para propagar información de etiquetas más abajo en la red

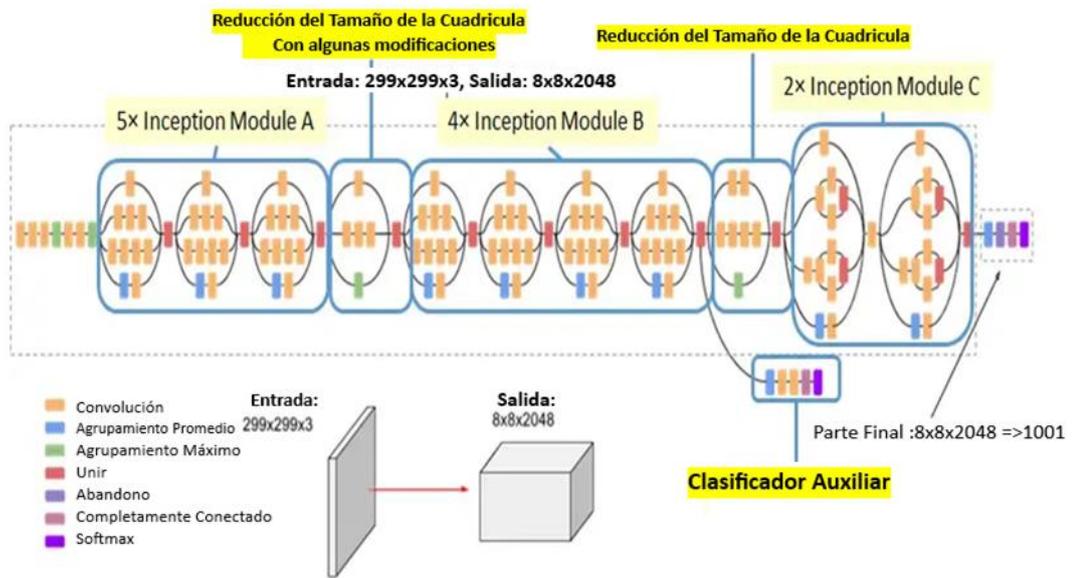


Figura 2.41 Arquitectura Inception-v3 (Batch Norm y ReLU se usan después de Convolución.) (Vasilev, 2019).

Capítulo 3

3 Trabajos Relacionados

Se revisaron los siguientes antecedentes como conocimiento base para la investigación.

3.1 Detección precoz de cáncer de piel en imágenes basado en redes convolucionales (México).

En esta investigación (Cristina, 2019) implementa una Red Neuronal Convolucional para clasificar imágenes de melanomas. Para implementar esta red neuronal utilizó la metodología de transferencia de aprendizaje (Transfer Learning) utilizando las redes AlexNet, VGG y ResNet con la librería Pytorch. Con cada una de estos modelos realiza experimentos variando la proporción de los datos de entrenamiento y validación. Encuentra que los mejores resultados se obtienen cuando la proporción de los datos de entrenamiento y validación es de 55% y 45% respectivamente. Con la red AlexNet se obtiene una exactitud de 74%, con VGG se obtiene 69% y con ResNet 72%.

Como conjunto de datos utiliza HAM10000 ("*Human Against Machine with 10000 training images*") que consiste de 13786 imágenes dermatoscópicas.

Entre sus conclusiones importantes indica que se ha logrado una certeza de 74% en las predicciones las cuales considera insuficientes.

3.2 Clasificación de lesiones en la piel con ensemble de redes neuronales residuales. (Barcelona - España)

En este trabajo (Tarrés, 2019), se estudia el problema de clasificación de lesiones en la piel utilizando redes neuronales residuales. El conjunto de datos utilizado es el proporcionado por el challenge ISIC2019. Su objetivo principal es la obtención de un modelo que clasifique correctamente las imágenes de lesiones en la piel aplicando redes neuronales residuales y la técnica de ensamble para mejorar los resultados. El estudio se centra en la red ResNet para generar predicciones y las técnicas de ensamble.

La primera modificación que realiza es referente a la métrica de error, ya que se trata de un problema complejo y en este proyecto solo se ha utilizado la *balanced accuracy*.

Otra modificación, relacionada con los datos de entrenamiento y de validación, es la utilización de validación cruzada, para obtener un resultado más robusto y más extrapolable a nuevas imágenes introducidas a la base de datos.

Considera que el hiperparámetro más importante es la tasa de aprendizaje por lo que realiza varios experimentos entrenando la red pre entrenada ResNet con diferentes valores de tasa de aprendizaje. Establece luego de los experimentos que la tasa de aprendizaje que mejores resultados proporciona es 1×10^{-6} con un valor de *accuracy* de 78%

3.3 Arquitectura de red neuronal convolucional para diagnóstico de cáncer de piel (Perú).

En este trabajo (Giorzinio Maikol Tejada Layme, 2020) se describe el problema para la detección de cáncer de piel a partir de imágenes ya clasificadas en melanoma maligno y benigno utilizando un modelo de aprendizaje profundo. Para la solución a este problema, se evaluó las redes pre entrenadas AlexNet, ZFNet, GoogleNet y VGGNet entrenándolas con dos conjuntos de datos denominados ISIC con 2650 imágenes y PH2 con 200 imágenes. El modelo establecido para el problema está basado en una clasificación binaria utilizando los valores 1 en caso de maligno y 0. Se menciona que se llegó a obtener una exactitud de 92% pero no se indica con cuál de los modelos pre entrenados.

3.4 Herramienta de reconocimiento y clasificación de lesiones cutáneas basada en redes neuronales convolucionales para asistir en el diagnóstico de cáncer de piel tipo melanoma (Argentina).

En esta investigación (Binder, 2021) se desarrolló una herramienta de asistencia diagnóstica, basada en procesamiento de imágenes dermatológicas e inteligencia artificial que asiste al dermatólogo en el diagnóstico de cáncer de piel tipo melanoma. El algoritmo de clasificación binaria suministrado a la herramienta analiza y clasifica las lesiones cutáneas melanocíticas proporcionando una probabilidad diagnóstica de pertenencia a la clase “benigna” o “maligna”. Este fue desarrollado para el sistema operativo Windows, empleando lenguaje open-source (Python) y herramientas específicas del área de la IA (TensorFlow y Keras). Para entrenar la CNN se utilizaron imágenes dermatoscópicas digitales obtenidas de un repositorio de imágenes internacional libre denominado ISIC.

Se obtuvieron métricas comparables al desempeño de dermatólogos profesionales y a aquel de otros modelos citados en la bibliografía. Las métricas AUC obtenidas fueron 0.88 para la ROC AUC y 0.90 para la ROC PR. Por otro lado, se calcularon los valores de exhaustividad (TPR), especificidad (TNR) y exactitud, siendo estos 91.15%, 61.98% y 77.20% respectivamente.

3.5 Melanoma detection by analysis of clinical images using convolutional neural network.

En este artículo (E. Nasr-Esfahani, 2016) se utilizan los métodos de aprendizaje profundo para implementar un sistema de diagnóstico automático para la detección del melanoma. Para ello se preprocesan las imágenes digitales de entrada, normalmente sometidas a efectos de ruido e iluminación. Después, las imágenes se envían a una arquitectura CNN pre entrenada para clasificar la entrada como melanoma o benigna. El artículo no menciona cuál es la red pre entrenada utilizada en la investigación.

El conjunto de datos que utiliza consta de 170 imágenes no dermatoscópicas (70 de melanoma, 100 de nevo) y pertenece al archivo de imágenes digitales del Departamento de Dermatología del Centro Médico Universitario de Groningen (UMCG).

Para hacer frente a las limitaciones del conjunto de datos de entrenamiento y al número relativamente bajo de imágenes, considera el enfoque de aumento de datos llegando a obtener 6120 imágenes.

El resultado del método propuesto es comparado con otros tres métodos existentes que han reportado sus resultados sobre el mismo conjunto de datos, presentando un 81% de exactitud superando a los resultados de las otras investigaciones.

3.6 A comparative study of melanoma skin cancer detection in traditional and current image processing techniques.

Esta investigación (B. Sreedhar, 2020) realiza un estudio comparativo entre el procesamiento de imágenes tradicional y las tecnologías actuales de procesamiento de imágenes para la clasificación de imágenes de cáncer de piel.

Dentro de los métodos tradicionales de procesamiento de imágenes para detectar cáncer de piel considera los siguientes pasos: La adquisición de la imagen que puede ser mediante una cámara digital o un dermatoscopio, el preprocesamiento de las imágenes para reducir el ruido presentado, el post procesamiento enfocado a mejorar la imagen y

su tamaño, la segmentación de las imágenes con el propósito de separar la imagen en múltiples partes y resaltar las más relevantes, la extracción de características que tiene la finalidad de obtener las propiedades de las imágenes, y finalmente la clasificación de las imágenes aplicando técnicas de aprendizaje automático.

Dentro de las tecnologías actuales de clasificación de imágenes dermatoscópicas, considera también las mismas etapas anteriores a excepción de la última en la que considera la aplicación de redes neuronales artificiales.

Capítulo 4

4 Desarrollo del Proyecto

4.1 Metodología

4.1.1 Tipo y nivel de investigación

Esta investigación es de tipo cuasiexperimental (Pallela S., 2006) porque se utilizarán datos que no pueden ser obtenidos al azar para medir el desempeño de redes neuronales convolucionales para clasificar imágenes de lesiones de piel. Y debido a que producirá nuevo conocimiento es de nivel aplicado tecnológico.

4.1.2 Diseño metodológico.

- La primera fase de la investigación es la selección de un conjunto de tres redes neuronales convolucionales pre entrenadas asumiendo criterios tales como su rendimiento en la clasificación de imágenes y su accesibilidad y facilidad de uso.
- La segunda fase es experimental, para la cual primero se exploran los datos para conocerlos de cerca y ver la necesidad de su preprocesamiento y aumento de datos.
- En una tercera fase procedemos a entrenar los tres modelos seleccionados con el mismo conjunto de datos y los mismos hiperparámetro que se pueden manipular.
- En la última fase se evalúa el desempeño de cada modelo, para lo cual se obtienen métricas como la pérdida, la exactitud, la matriz de confusión y el ROC.

4.2 El conjunto de datos

En esta sección se proporciona una descripción general de los conjuntos de datos más comunes utilizados en el diagnóstico del cáncer de piel.

4.2.1 HAMM10000

El conjunto de entrenamiento HAM10000 incluye 10.015 imágenes dermatoscópicas para detectar lesiones cutáneas pigmentadas. Este conjunto de datos está disponible públicamente e incluye 142 imágenes de lesiones cutáneas vasculares, 327 imágenes de queratosis actínica, 514 imágenes de carcinomas de células basales, 1099 imágenes de

queratosis benignas, 115 imágenes de dermatofibromas, 1113 imágenes de nevos melanocíticos y 6705 imágenes de melanomas (P. Tschandl, 2018).

4.2.2 PH²

La base de datos PH2 contiene 200 imágenes dermatoscópicas. La segmentación manual, el diagnóstico clínico y la identificación de muchas estructuras dermatoscópicas, realizados por dermatólogos expertos, se incluyen en la base de datos PH2 (Mendonça, 2013).

4.2.3 ISIC 2016

ISIC5 2016 consta de 900 imágenes de nevos benignos y melanomas malignos en el conjunto de entrenamiento y 379 imágenes dermatoscópicas en su subconjunto de prueba. Un total del 30,3% de las imágenes utilizadas en ISIC 2016 son de melanoma, mientras que el resto de las imágenes son de clase nevo benigno (Codella, 2017).

4.2.4 Atlas de dermatoscopia

El Atlas de dermatoscopia incluye cinco imágenes de AK, 42 imágenes de BCC, 70 imágenes de queratosis benignas, 20 imágenes de dermatofibromas, 275 imágenes de nevos melanocíticos, 582 imágenes de melanoma y 30 imágenes de lesiones cutáneas vasculares (Lio, 2004).

4.2.5 Dermofit

Dermofit tiene 1300 imágenes totales de lesiones cutáneas divididas en diez clases. Tiene 45 imágenes de queratosis actínica (AK), 239 imágenes de carcinoma de células basales (BCC), 331 imágenes de nevo melanocítico/mola (ML), 88 imágenes de carcinoma de células escamosas (SCC), 257 imágenes de queratosis seborreica, 78 imágenes de carcinoma intraepitelial (IEC), 24 imágenes de granuloma piógeno (PYO), 97 imágenes de hemangioma, 65 imágenes de dermatofibromas, 76 imágenes de melanoma maligno (Ballerini, 2020).

⁵ The International SkinImaging Collaboration (<https://www.isic-archive.com>)

4.2.6 BCN20000

Este conjunto de datos consta de 19.424 imágenes dermatoscópicas de lesiones cutáneas tomadas en el Hospital de Barcelona entre 2010 y 2016. Utilizando la base de datos BCN20000 se pueden aplicar las siguientes clasificaciones a las imágenes: nevis, melanoma, carcinoma basocelular, queratosis seborreica, queratosis actínica, carcinoma de células escamosas, dermatofibroma, lesión vascular (Combalia, 2019)

4.3 Conjunto de datos utilizado en el proyecto

Para evaluar la precisión de la clasificación de las lesiones cutáneas, se utilizará el conjunto de datos HAM10000 (P. Tschandl, 2018), debido a que la base de datos HAM10000 es un gran conjunto de datos de imágenes digitales con lesiones cutáneas, que consta de 10015 imágenes y todas las imágenes se han etiquetado con su tipo de patología correspondiente, como se muestra en la Tabla 4.1. Las imágenes dermatoscópicas del conjunto de datos se recopilieron durante un período de 20 años en dos sitios diferentes, por el médico Cliff Rosendahl en la Universidad de Queensland Australia, y el Departamento de Dermatología de la Universidad Médica de Viena, en Austria.

Tabla 4.1 Tipos de patología de la piel y el número de instancias en el conjunto de datos HAM10000

Tipo de patología	Nro. de imágenes
Queratosis actínica (akiec)	327
Carcinoma de células basales (bcc)	514
Dermatofibroma (df)	115
Melanoma (mel)	1113
Nevus (nv)	6705
Queratosis benigna pigmentada (bkl)	1099
Lesiones vasculares (vasc)	142
Total	10015

Fuente: (P. Tschandl, 2018)

En las siguientes figuras se puede apreciar las imágenes contenidas en el conjunto de datos.

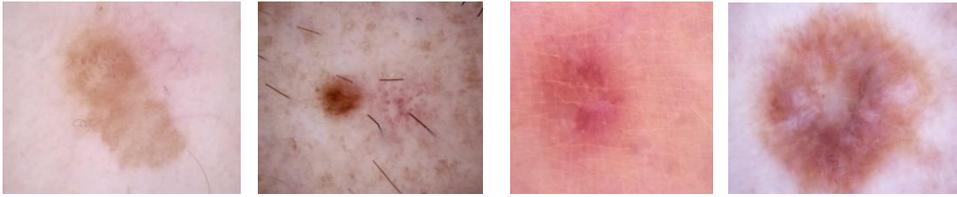


Figura 4.1 Imágenes de Keratosis Actínica (akiec) (P. Tschandl, 2018)

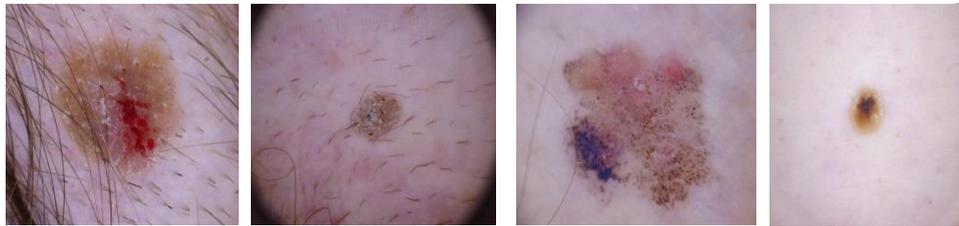


Figura 4.2 Carcinoma de Células Basales (bcc) (P. Tschandl, 2018)

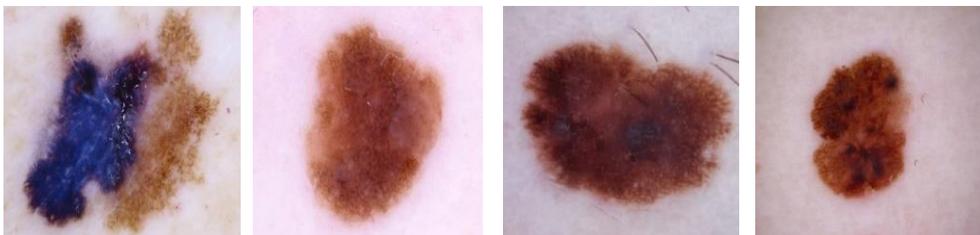


Figura 4.3 Imágenes de Dermatofibroma (df) (P. Tschandl, 2018)



Figura 4.4 Imágenes de Melanoma (mel) (P. Tschandl, 2018)



Figura 4.5 Imágenes de Nevus (nv) (P. Tschandl, 2018)

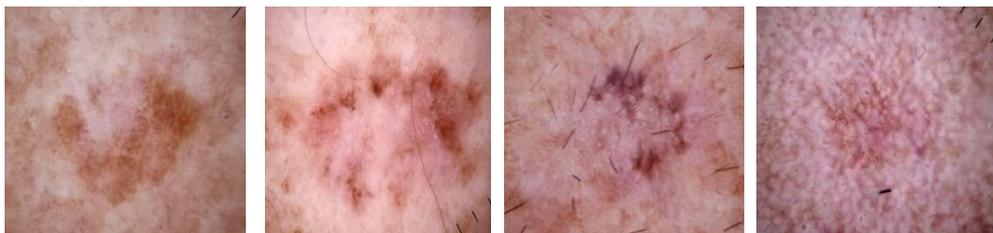


Figura 4.6 Imágenes de Queratosis benigna pigmentada (bkl) (P. Tschandl, 2018)

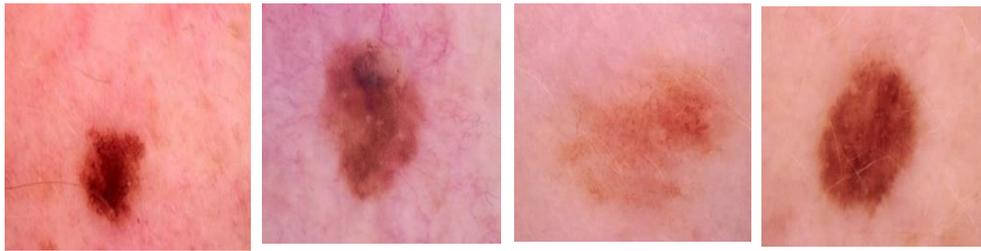


Figura 4.7 Imágenes de Lesiones Vasculares (vasc) (P. Tschandl, 2018)

4.3.1 Distribución de los datos.

El conjunto de datos HAM10000 consta de 10015 imágenes, las cuales se han dividido en el conjunto de entrenamiento y en el conjunto de pruebas y validación según el siguiente detalle:

El 80% de las imágenes para entrenamiento haciendo un total de 8,012 imágenes

El 20% de las imágenes para pruebas y validación haciendo un total de 2003 imágenes.

La distribución de las imágenes de entrenamiento y pruebas se aprecian en la Figura 4.8.

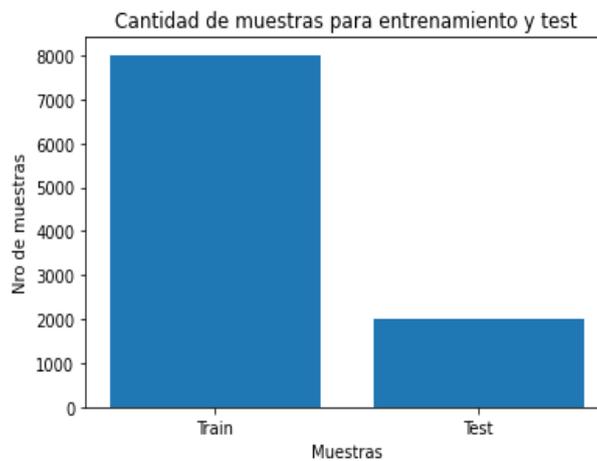


Figura 4.8 Distribución de las imágenes de entrenamiento y de prueba (Fuente Elaboración Propia)

La distribución de los datos de entrenamiento por clase es la siguiente:

Tabla 4.2 Número de Muestras por clase de Entrenamiento

Clase	Nro. de imágenes
akiec	262
bcc	411
bkl	879
df	92
mel	890
nv	5,364
vasc	114
Total	8,012

(Fuente: Elaboración Propia)

4.3.2 Distribución de los datos de prueba por clase.

La distribución de los datos de prueba por clase es la siguiente:

Tabla 4.3 Número de muestras por clase

Clase	Nro. de imágenes
akiec	65
bcc	103
bkl	220
df	23
mel	223
nv	1341
vasc	28
Total	2,003

(Fuente: Elaboración Propia)

4.4 Preprocesamiento de las imágenes.

Diagnosticar el cáncer de piel mediante métodos asistidos por computadora no es sencillo y se debe considerar muchos factores para un diagnóstico preciso. Por ejemplo, artefactos como pelos, esquinas oscuras, burbujas de agua, signos de marcadores, marcas de tinta y signos de regla, como se muestra en la Figura 4.9 pueden dar lugar a una clasificación errónea y una segmentación inexacta de las lesiones cutáneas. Por lo que es necesario preprocesar las imágenes con la finalidad de disminuir el efecto de estos artefactos en la extracción de características y mejorar la precisión del modelo.



Figura 4.9 Artefactos en las imágenes del conjunto de datos HAMM10000.
(Elaboración Propia)

De Castillo A. (2016), en la Figura 4.10 se ilustra el diagrama de bloques para el procesamiento de imágenes con lesiones cutáneas para la obtención de la región de interés,

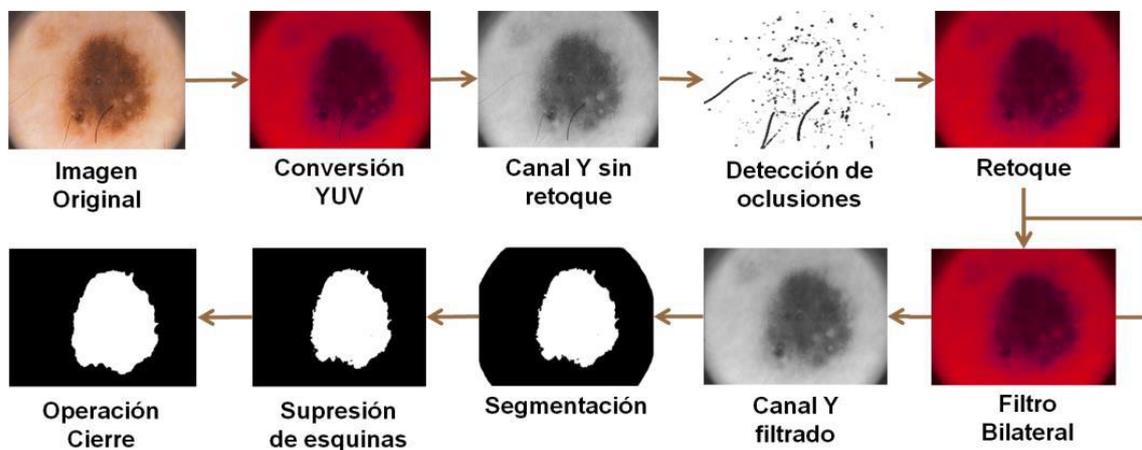


Figura. 4.10 diagrama de bloques del preprocesamiento de imágenes de lesiones cutáneas De Castillo A. (2016)

El esquema consiste en tomar la imagen y realizar una conversión de espacio de color, en este caso, de RGB a YUV. Al tener nuestra imagen en el nuevo espacio de color se toma el canal de luminancia y se procede a la detección de oclusioniones para obtener la máscara de los pixeles que serán procesados en el retoque. La etapa de retoque utiliza la imagen

en el nuevo espacio de color y, basado en la máscara de oclusiones, se hace el retoque de la imagen con el fin de eliminar vello y burbujas presentes que puedan afectar en la identificación de la lesión (Castillo A. 2016).

Una imagen, por la gran cantidad de cambios de contraste, textura o color que puede contener, se puede considerar de naturaleza ruidosa, es cuando se decide hacer un filtrado bilateral para disminuir las variaciones sin perder los detalles de la imagen pasando a una etapa de segmentación donde se pretende clasificar los pixeles que pertenecen a la lesión. Cuando se adquiere la imagen se acompaña de la presencia de cuatro esquinas que pueden afectar la visualización causadas por el acoplamiento de las lentes al sistema de adquisición de imágenes, por lo que se requiere que los datos en estas esquinas sean discriminados con una etapa de supresión de datos o regiones. Se utiliza una última etapa donde se busca eliminar regiones de interés muertas que se encuentren dentro de la lesión para proceder con la extracción de las características que estén presentes en las imágenes y estos datos puedan ser usados para una clasificación (Castillo A. 2016).

4.4.1 Espacio de color YUV.

Comienza con colores RGB y asume que una luz blanca fue utilizada para adquirir la imagen. Es usado en codificaciones de color para la transmisión de señales analógicas de televisión. Este espacio separa las componentes cromáticas por medio de la substracción de iluminación Y de los canales rojo y azul respectivamente. La Matriz de transformación asociada es la siguiente (Castillo A. 2016):

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147141 & -0.288869 & 0.436010 \\ 0.614975 & -0.514965 & -0.100010 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Es un enfoque de espacio de color ampliamente usado en el desarrollo de algoritmos de procesamiento de imágenes a escala de grises.

4.4.2 Detección de oclusiones.

Consideramos por oclusión a todo elemento en la imagen que afecte o tenga poca importancia para el procesamiento de la misma. La detección de oclusiones se realiza por

medio de la desviación estándar, tomando como base el canal de luminancia en la imagen, en un pixel con una vecindad de $N \times N$ obteniendo una máscara de desviaciones en la cual se aplicará un umbral T_h obteniendo al final de este proceso una máscara indicando los pixeles que deberán ser retocados (ver Figura 4.11) (Castillo A. 2016).

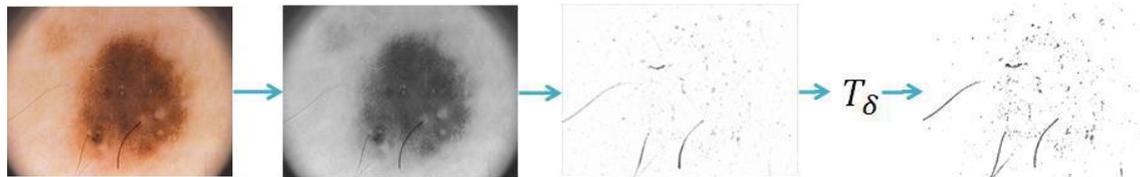


Figura 4.11. Proceso de la detección de oclusiones a partir del canal de luminancia en la imagen, Obtención de la máscara de desviaciones y su umbralización para la obtención de los pixeles a retocar (Castillo A. 2016)

4.4.3 Retoque

El retoque es una técnica que consiste en el relleno de regiones de datos perdidas, dañadas o que se busque modificar para obtener un efecto deseado. Sus aplicaciones van desde la restauración, eliminación de elementos no deseados e incluso efectos especiales.

En este caso se usa la imagen original y una máscara en la que se muestran los elementos a retocar, como el vello o burbujas (Figura. 4.12) (Castillo A. 2016).

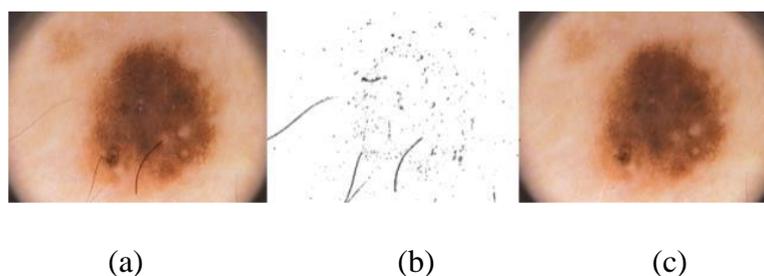


Figura 4.12. Proceso de retoque a) Imagen original, b) Mascara de retoque, c) Imagen con retoque (Castillo A. 2016)

4.4.4 Filtro bilateral

El filtro bilateral difumina imágenes mientras conserva los bordes por medio de una combinación no lineal de valores en la imagen. El método es local, simple y puede ser iterativo, combina colores basados en su cercanía geométrica y similitud fotométrica teniendo más ponderación valores cercanos que lejanos en ambos dominios. La imagen filtrada es obtenida por la siguiente ecuación (Castillo A. 2016):

$$x'_i = \frac{\sum_{j \in N_i} \omega(i, j) x_j}{\sum_{j \in N_i} \omega(i, j)},$$

donde ω son ponderaciones aplicadas a cada pixel x_j en la vecindad N_i . La ponderación se compone de la ponderación espacial ω_s y la ponderación de color ω_c .

$$\omega(i, j) = \omega_s(i, j) \times \omega_c(i, j),$$

donde ω_s y ω_c se definen por:

$$\omega_s(i, j) = e^{-\left(\frac{d_s(i, j)}{2\sigma_s^2}\right)},$$

$$\omega_c(i, j) = e^{-\left(\frac{d_c(i, j)}{2\sigma_c^2}\right)},$$

Teniendo que σ_s está relacionado con el radio de difuminado, si σ_s es alto el difuminado es mayor, pero si σ_s es muy grande puede difuminar bordes importantes. Y σ_c determina cuanto contraste será conservado o difuminado. Para valores bajos de σ_c casi todos los contrastes serán conservados mientras que para valores altos el comportamiento sería un emborronamiento gaussiano lineal (Castillo A. 2016).

4.4.5 Fuzzy C Means.

Es un algoritmo de clasificación que permite a un dato pertenecer a dos o más grupos con diferente grado de pertenencia. Dado un conjunto de datos $X = \{x_1, x_2, \dots, x_k\}$ conteniendo n número de datos de dimensión d , Fuzzy C Means pretende minimizar la siguiente función objetivo (Castillo A. 2016).

$$J_m = \sum_{k=1}^n \sum_{i=1}^c u_{ik}^m \|x_k - v_i\|^2,$$

donde v es el centroide del grupo i , u_{ik} es el grado de pertenencia de x_k en el grupo i mientras que m es el factor de *fuzzyficación*. La minimización se obtiene mediante actualizaciones sucesivas de v y u_{ik} utilizando:

$$v_i = \frac{\sum_{k=1}^n u_{ik}^m x_k}{\sum_{k=1}^n u_{ik}^m},$$

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{\|x_k - v_i\|}{\|x_k - v_j\|} \right)^{2/(m-1)}}.$$

4.4.6 Supresión de esquinas.

Como se muestra en la Figura 4.13, para suprimir las esquinas en la segmentación se considera que, por efecto del acoplamiento de las lentes, la lesión está dentro de una circunferencia de radio r por lo que las regiones son discriminadas si no se encuentran dentro de esta circunferencia concéntrica a la imagen (Castillo A. 2016).

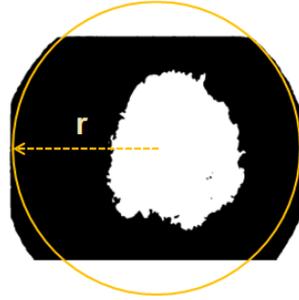


Figura 4.13. Enfoque para la supresión de esquinas de la segmentación (Castillo A. 2016).

4.4.7 Cierre.

Está una operación compuesta por las operaciones morfológicas dilatación y erosión respectivamente denotada por la siguiente ecuación (Castillo A. 2016).

$$I \bullet H = (I \oplus H) \ominus H.$$

La dilatación es una operación que corresponde al crecimiento, donde se agregan capas de píxeles a la imagen I dependiendo de un elemento estructural H y se denota por la siguiente ecuación.

$$I \oplus H \equiv \bigcup_{p \in I} H_p.$$

La erosión es una operación inversa a la dilatación cuya función es remover píxeles respecto al elemento estructural H donde su notación es la siguiente

$$I \ominus H \equiv \{p \in \mathbb{Z}^2 | H_p \subseteq I\},$$

donde p es un par coordinado en $I(p)=I$ y H_p denota el elemento estructural desplazado por p .

El elemento estructural es similar a la matriz de coeficientes de un filtro lineal, las propiedades de un filtro morfológico son especificadas en la matriz de elementos H . En una imagen binaria H contendrá valores de 0 y 1 solamente.

4.4.8 Resultados.

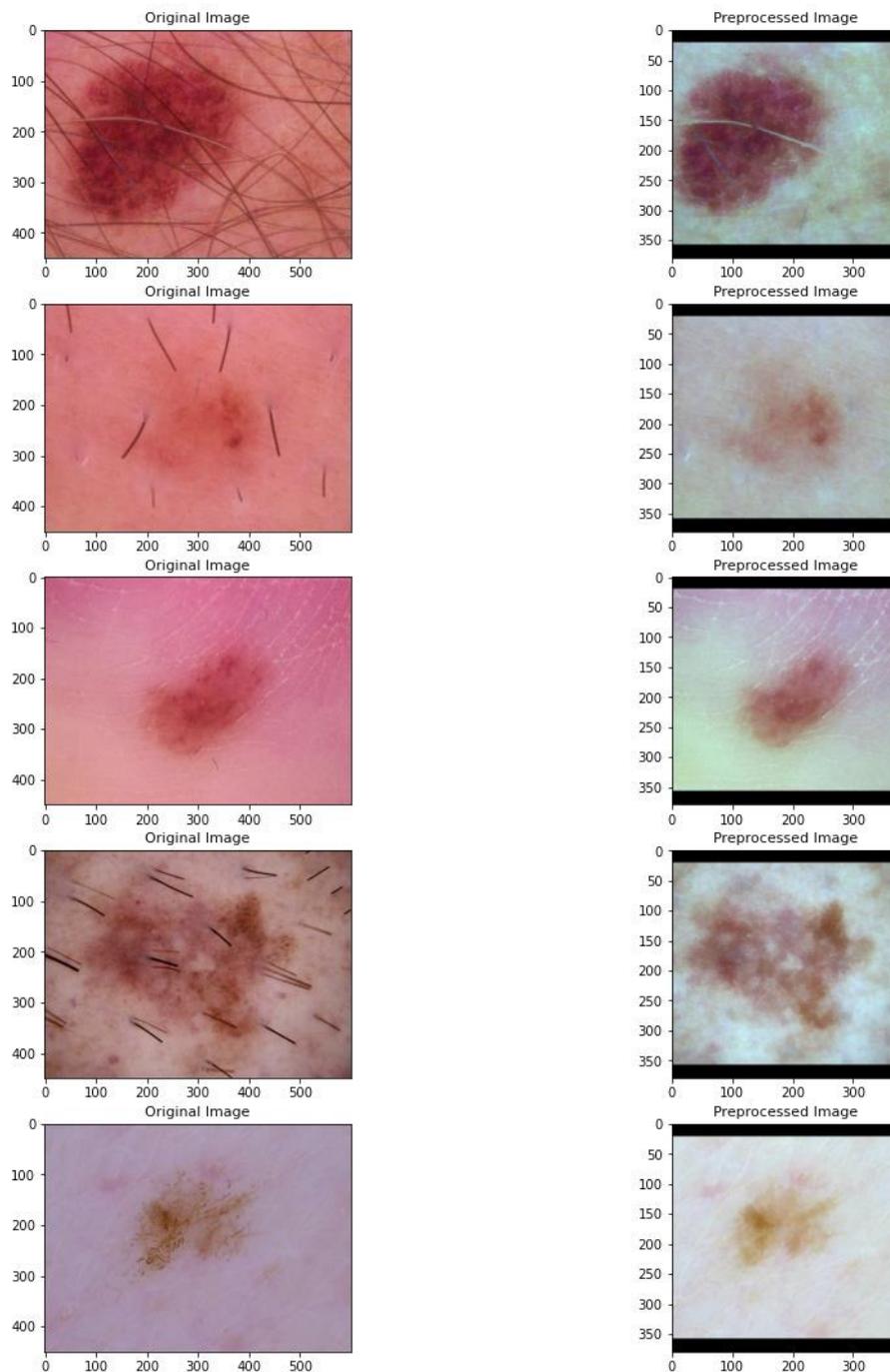


Figura 4.14. Resultados del preprocesamiento (Castillo A. 2016).

4.5 Aumento de datos

Para evitar el sobreajuste y el modelo pueda generalizar mejor es importante realizar el aumento de datos. En este trabajo utilizamos la librería *TensorFlow* y el aumento de datos se realiza definiendo un generador como se muestra en la Figura 4.15

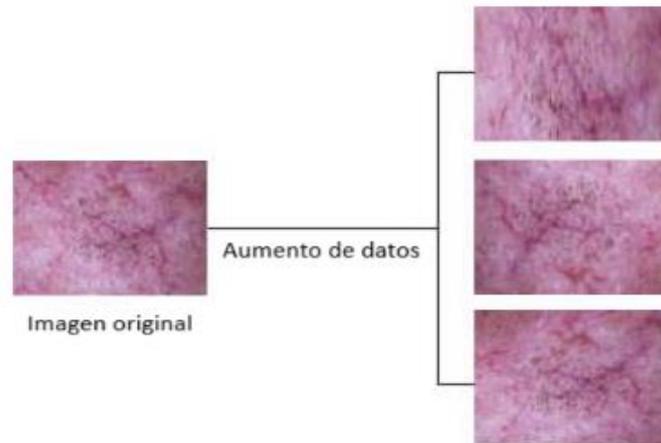


Figura 4.95 Aumento de datos (Fuente: Elaboración Propia)

```
import tensorflow as tf

from tensorflow.keras.preprocessing.image import
ImageDataGenerator, load_img

train_datagen = ImageDataGenerator(rescale=1. / 255,
                                   rotation_range=40,
                                   width_shift_range=0.2,
                                   height_shift_range=0.2,
                                   shear_range=0.2,
                                   zoom_range=0.2,
                                   horizontal_flip=True)
```

Figura 4.16 Preprocesamiento y aumento de datos (Fuente: Elaboración Propia)

4.6 Reentrenamiento de los modelos seleccionados.

4.6.1 Determinación de hiperparámetros.

En la Tabla 4.4 se detallan los hiperparámetros con los cuales se entrenó cada uno de los modelos seleccionados. Estos hiperparámetros se seleccionaron en base a los antecedentes de la investigación tales como (Cristina, 2019) y (Binder,2021)

Tabla 4.4 Hiperparámetros de entrenamiento de los modelos seleccionados

Hiperparámetros	InceptionV3	MovileNetV2	DenseNet201
Nro. de épocas	25	25	25
Nro. de capas intermedias	2	2	2
Nro. de neuronas capa 1	1024	1024	1024
Nro. de neuronas capa 2	512	512	512
Función de activación en capas intermedias	<i>ReLU</i>	<i>ReLU</i>	<i>ReLU</i>
Función de activación en capa de salida	<i>Softmax</i>	<i>Softmax</i>	<i>Softmax</i>
Tasa de aprendizaje	0.001	0.001	0.001
Optimizador	Adam	Adam	Adam

(Fuente: Elaboración Propia)

4.6.2 Construcción de los nuevos modelos.

Los nuevos modelos se construyen por transferencia de aprendizaje sobre modelos base previamente entrenados con el conjunto de datos ImageNet. Los modelos base considerados, sobre los que se construirán los nuevos modelos para clasificar imágenes de lesiones de piel son InceptionV3, MobileNetV2 y DenseNet201.

En general los modelos se construyen siguiendo el siguiente procedimiento.

1. Creación del nuevo modelo.

A partir del modelo pre entrenado se obtiene el nuevo modelo base especificando los siguientes parámetros (ver Figura 4.17):

- *weights = 'imagenet'*. Significa que se utilizará los pesos obtenidos con el conjunto de datos ImageNet.
- *include_top = False*. Indica que se eliminará la capa de salida del modelo pre entrenado.
- *Input_shape(ancho, alto, canales)*. El tamaño de las imágenes en pixeles y el número de canales

```
modelo_base = red_pre_entrenada (weights='imagenet',
include_top=False, input_shape(ancho, alto, canales))
```

Figura 4.107 Creación del modelo base

2. Congelamiento de capas del nuevo modelo.

Es necesario congelar las últimas capas del modelo base para evitar la pérdida de información obtenida en el pre entrenamiento. Estas capas no son entrenables (ver Figura 4.18).

```
for layer in modelo_base.layers[:-15]:  
    layer.trainable = False
```

Figura 4.118 Congelamiento de las últimas capas del modelo base

3. Adición de capas entrenables al nuevo modelo.

A continuación, se agregan nuevas capas encima del modelo pre entrenado para la nueva tarea. Se agrega también una capa *Flaten* para convertir la salida del modelo pre entrenado en una matriz unidimensional, capas *Dropout* para evitar el sobreajuste, y capas densas intermedias para la clasificación y una capa de salida con el número de neuronas correspondientes a las clases (ver Figura 4.19).

```
x = base_model.output  
x = GlobalAveragePooling2D()(x)  
x = Flatten()(x)  
x = Dense(units=1024, activation='relu')(x)  
x = Dropout(0.3)(x)  
x = Dense(units=512, activation='relu')(x)  
x = Dropout(0.3)(x)  
salida = Dense(units=7, activation='softmax')(x)  
modelo red pre entrenada = Model(base_model.input, salida)
```

Figura 4.19 Adición de capas al modelo base

4. Compilación del nuevo modelo.

Para la compilación del nuevo modelo se tiene que especificar la función de pérdida, el optimizador con la tasa de aprendizaje utilizada y la métrica de evaluación como se puede apreciar en la Figura 4.20.

```
# Función de pérdida
loss = tf.keras.losses.CategoricalCrossentropy()
# Optimizador
optimizer = tf.keras.optimizers.Adam(learning_rate=0.001)
# compilación
modelo_inceptionV3.compile(optimizer=optimizer, loss=loss,
metrics= ['accuracy'])
```

Figura 4.20 Compilación del nuevo modelo.

5. Entrenamiento del nuevo modelo

El entrenamiento del nuevo modelo se realiza especificando el número de épocas y el tamaño del lote como se puede apreciar en la Figura 4.21.

```
historia_modelo = modelo_base.fit( train_generador,
validacion_data=test_generador,epochs=50, batch_size=32)
```

Figura 4.21 Entrenamiento del nuevo modelo.

4.7 Hardware y Software utilizado.

- **Hardware**

Estación de trabajo: HP ZBOOK

Procesador: Intel(R) Xeon(R) E-2176M CPU @ 2.70GHz 2.71 GHz

Memoria: 16.0 GB

Sistema Operativo: Windows 11 de 64 bits, procesador x64

Video: *Nvidia Quadro* P1000 de 4 GB GDDR5.

- **Software.**

Lenguaje de programación: *Python 3.9*

Framework: *Tensorflow, Keras*

Plataforma: *Google Colaboratory, Visual Studio Code*

Capítulo 5

5 Resultados y Discusión.

En este capítulo evaluaremos y discutiremos los resultados de los modelos construidos a partir de las redes pre entrenadas InceptionV3, MobileNetV2 y DenseNet201 para clasificar lesiones de la piel con melanoma a través de transferencia de aprendizaje utilizando los hiperparámetros de la Tabla 5.4.

5.1 Evaluación del modelo basado en InceptionV3.

5.1.1 Resultados del entrenamiento.

Los resultados de pérdida y exactitud durante el entrenamiento del modelo InceptionV3 con datos de entrenamiento y pruebas se muestran en la Figura 5.1

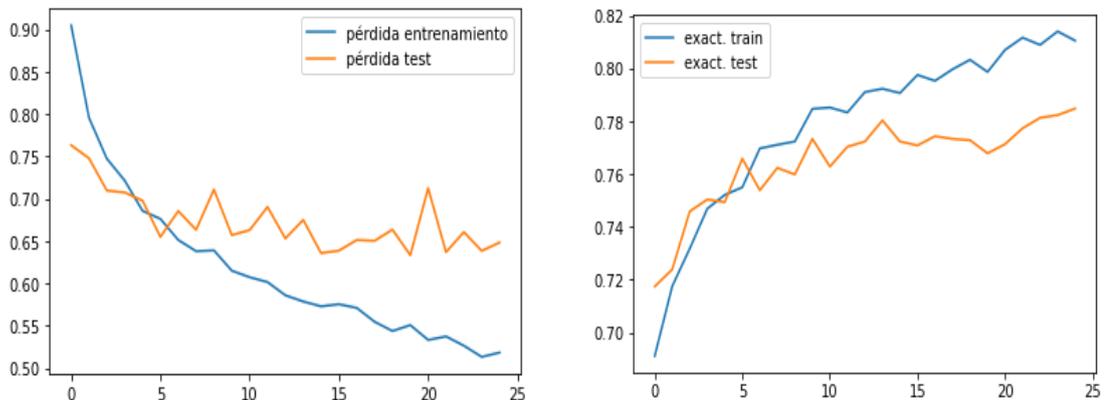


Figura 5.1 Gráficas de pérdida y exactitud del modelo basado en InceptionV3 Fuente (Elaboración Propia)

Se puede observar que la pérdida para los datos de entrenamiento es constante llegando a su punto más bajo en 0.53. Sin embargo, con los datos de prueba la función de pérdida es variable, pero con una tendencia descendente llegando a 0.66 en la iteración 25.

En cuanto a la exactitud del modelo, se puede observar en el gráfico de la derecha que para los datos de entrenamiento es ascendente, llegando a 81% al final del entrenamiento. Con respecto a los datos de prueba, la exactitud también muestra una tendencia ascendente llegando a su valor final en 78%.

5.1.2 Matriz de confusión.

La matriz de confusión resultante del entrenamiento de este modelo se puede apreciar en la Figura 5.2.

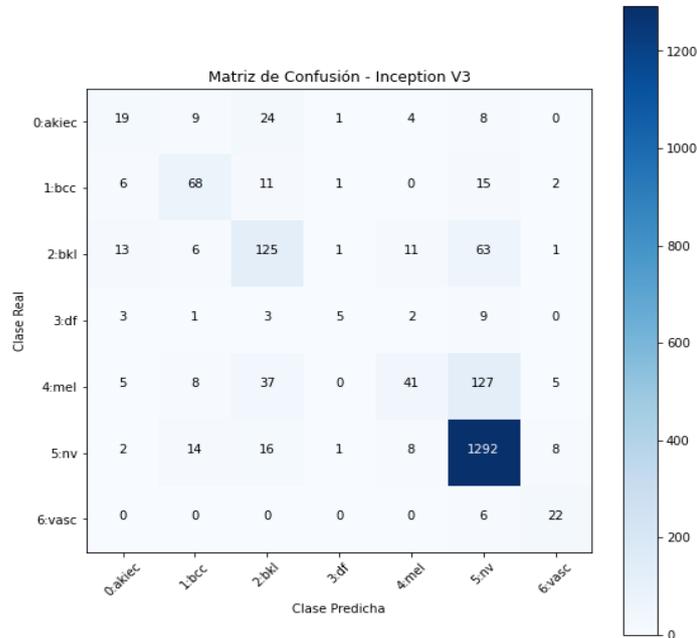


Figura 5.2 Matriz de confusión del modelo basado en InceptionV3

De la matriz mostrada en la Figura 5.2 se pueden obtener los resultados mostrados en la Tabla 5.1

Tabla 5.1 Métricas del modelo basado en InceptionV3

Clases	Precisión	Recall	Score -F1	
0	akiiec	0.40	0.29	0.34
1	bcc	0.64	0.66	0.65
2	bkl	0.58	0.57	0.57
3	df	0.56	0.22	0.31
4	mel	0.62	0.18	0.28
5	nv	0.85	0.96	0.90
6	vasc	0.58	0.19	0.67

Fuente:(Elaboración propia)

De la tabla 5.1. se puede observar que el modelo basado en InceptionV3 reconoce adecuadamente lesiones de la piel de tipo nv debido a que la métrica Score F1 es la más alta, sin embargo, imágenes de las demás clases se reconocerán en un porcentaje más bajo.

5.2 Evaluación del modelo basado en MobileNetV2.

5.2.1 Resultados del entrenamiento.

Los resultados del entrenamiento de este modelo se pueden observar en la Figura 6.3

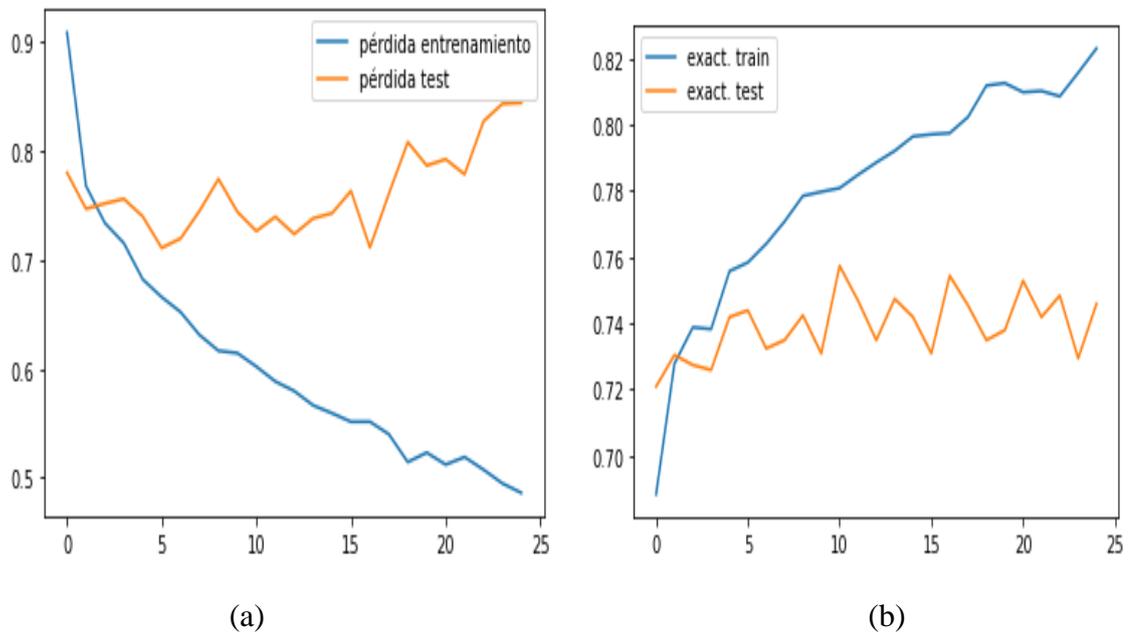


Figura 5.3 Gráficas de pérdida y exactitud del modelo basado en MobileNetV2 Fuente (Elaboración Propia)

La función de pérdida para datos de entrenamiento se muestra descendente llegando a tener un valor de 0.48 finalmente, sin embargo, la función de pérdida para los datos de prueba se muestra con tendencia ascendente lo que significaría que no está disminuyendo el error durante el entrenamiento.

En cuanto a la exactitud del modelo basado en MobileNetV2 de acuerdo a la Figura 5.3 (b) se muestra un comportamiento ascendente para datos de entrenamiento llegando a un valor de 82%, sin embargo, para datos de test o prueba es muy variable llegando a su punto más alto (76%) en la iteración 11 aproximadamente.

5.2.2 Matriz de confusión.

La matriz de confusión para el modelo basado en MobileNetV2 se muestra en la Figura 5.4

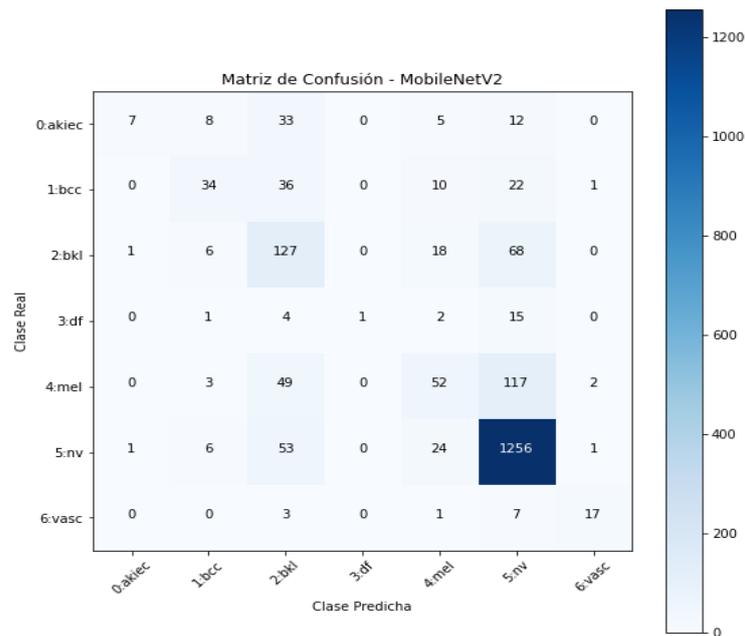


Figura 5.4 Matriz de confusión del modelo basado en MobileNetV2 Fuente (Elaboración Propia)

Las métricas obtenidas a partir de la matriz de confusión de la Figura 5.4. se muestran en la Tabla 5.2.

Tabla 5.2 Métricas del modelo basado en MobileNetV2

Clases		Precision	Recall	Score -F1
0	akiec	0.78	0.11	0.19
1	bcc	0.59	0.33	0.42
2	bkl	0.42	0.58	0.48
3	df	1.00	0.04	0.08
4	mel	0.46	0.23	0.31
5	nv	0.84	0.94	0.89
6	vasc	0.81	0.61	0.69

Fuente: (Elaboración Propia.)

Analizando la Tabla 5.2 podemos verificar que también en este modelo las imágenes de la clase “nv” serán reconocidas con mayor exactitud, mientras que imágenes de la clase “df” serán las que no reconocerá muy bien este modelo por cuanto el Score F1 para este modelo es 0.08.

5.3 Evaluación del modelo DenseNet201.

5.3.1 Resultados del entrenamiento.

En la Figura 5.5 se muestran las gráficas de la función de pérdida y exactitud del modelo basado en DenseNet201.

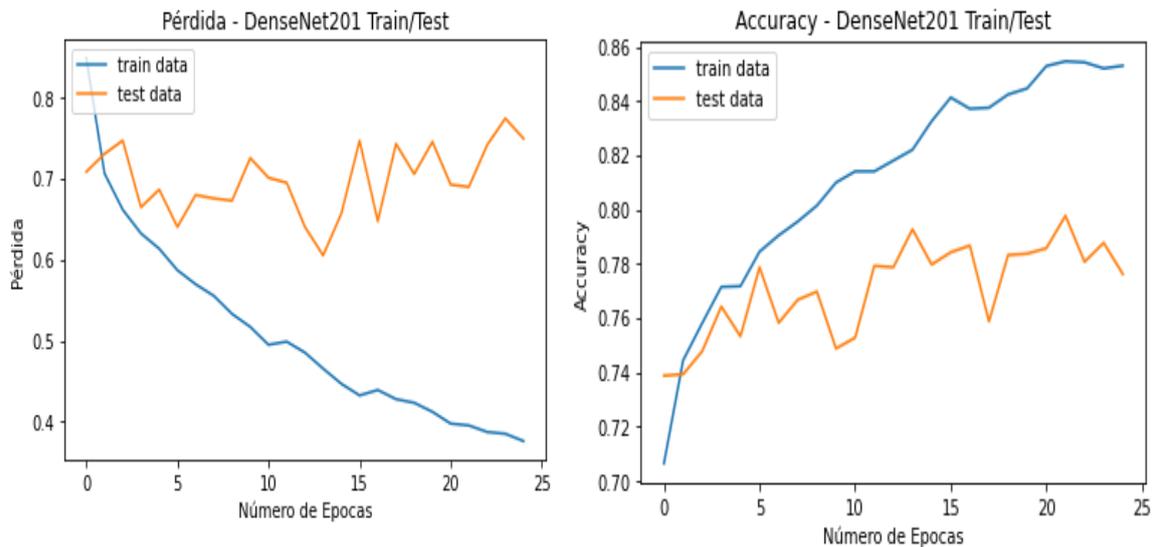


Figura 5.5 Gráficas de pérdida y exactitud del modelo basado en DenseNet201

En este modelo la función de pérdida para los datos de entrenamiento es también descendente llegando a un valor de 0.37 al final de las iteraciones, pero la función de pérdida para los datos de prueba muestra un comportamiento con tendencia ascendente, lo que demuestra también que en este modelo no hay disminución del error en el entrenamiento con datos de prueba. Con respecto a la exactitud tiene un comportamiento ascendente para datos de entrenamiento llegando a un valor de 85% mientras que para datos de prueba y validación su tendencia también es ascendente llegando a su pico más alto en la iteración 22 con una exactitud de 78%.

5.3.2 Matriz de confusión.

La matriz de confusión para el modelo basado en DenseNet201 se muestra en la Figura 5.6.

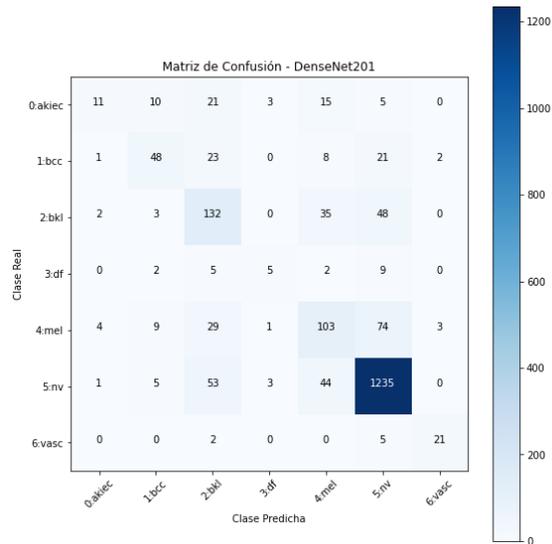


Figura 5.6 Matriz de confusión del modelo basado en DenseNet201

La Tabla 5.3 muestra las métricas obtenidas de la matriz de confusión mostrada en la Figura 5.6.

Tabla 5.3 Métricas del modelo basado en DenseNet201

Clases		Precision	Recall	Score -F1
0	akiiec	0.58	0.17	0.26
1	bcc	0.62	0.47	0.53
2	bkl	0.50	0.60	0.54
3	df	0.42	0.22	0.29
4	mel	0.50	0.46	0.48
5	nv	0.88	0.92	0.90
6	vasc	0.81	0.75	0.78

Fuente: (Elaboración Propia)

Los resultados de la Tabla 5.3. demuestran que la clase “nv” es mejor detectada por este modelo por ser la clase mayoritaria con Score F1 de 0.90, luego la clase “vasc” es la siguiente mejor detectada con Score F1 de 0.78 y las imágenes de la clase “akiiec” son las peores detectadas con un Score F1 de 0.26.

5.4 Comparación de resultados de los modelos.

En la tabla 5.4 se muestran los resultados de la métrica exactitud para los modelos estudiados.

Tabla 5.4 Tabla Resultados de exactitud de los modelos estudiados

Modelo	Exactitud (Accuracy)
InceptionV3	80.5%
MobileNetV2	76.5%
DenseNet201	79.3%

Fuente: (Elaboración Propia)

Del análisis de la Tabla 6.4 se puede advertir que el modelo basado en InceptionV3 tiene la mejor exactitud, sin embargo, debido a que los datos están desbalanceados un mejor análisis se realiza teniendo en cuenta la métrica Score F1, pero aun así no deja de ser un indicador del desempeño de esta red al clasificar imágenes de lesiones de piel.

La métrica que mejor evalúa la clasificación cuando los datos están desbalanceados es el Score F1, por lo que en la tabla 5.5 recogemos los resultados de esta métrica para todos los modelos estudiados

Tabla 5.5 Tabla Score F1 de los modelos estudiados

Clases		Score F1		
		InceptionV3	MobileNetV2	DenseNet201
0	akiec	0.34	0.19	0.26
1	bcc	0.65	0.42	0.53
2	bkl	0.57	0.48	0.54
3	df	0.31	0.08	0.29
4	mel	0.28	0.31	0.48
5	nv	0.90	0.89	0.90
6	vasc	0.67	0.69	0.78

Fuente: (Elaboración Propia)

Del análisis de la Tabla 6.5 se puede concluir que el modelo basado en InceptioV3 obtiene mejores resultados de clasificación por cuanto las clases minoritarias son detectadas con mayor precisión que los otros modelos, aunque para las clases “mel” y “vasc” el modelo basado en DenseNet201 demuestra mejor desempeño con Score F1 de 0.48 y 0.78 respectivamente. Para la clase “nv” que es la mayoritaria los tres modelos estudiados presentan excelentes condiciones para su reconocimiento debido a que tiene un Score F1 de 0.90 en promedio.

Capítulo 6

6 Conclusiones y Recomendaciones

6.1 Conclusiones.

1. El conjunto de datos utilizado para esta investigación es el denominado HAM10000 (P. Tschandl, 2018) que consta de 10015 imágenes a colores en 7 clases de cáncer de piel que fue recopilado durante 20 años en la Universidad de Queensland Australia, y el Departamento de Dermatología de la Universidad Médica de Viena, en Austria. Del análisis que se realizó a este conjunto de datos se pudo verificar que esta desbalanceado. De este conjunto el 80% se utilizó para entrenamiento y el 20% para pruebas y validación.
2. Las imágenes del conjunto de datos HAMM10000 presentan muchos elementos extraños tales como pelos, esquinas oscuras, marcas de tinta, etc. Por lo que se necesitó realizar un preprocesamiento con la finalidad de disminuir el efecto de estos elementos en la extracción de características y así aumentar la precisión de los modelos.
3. Reducir el efecto de elementos extraños en las imágenes de lesiones de piel mediante su pre procesamiento, incrementa la exactitud de las redes neuronales convolucionales, por lo que realizar esta etapa antes del entrenamiento es muy importante.
4. Para clasificar imágenes de lesiones de piel con melanoma, se seleccionaron los modelos pre entrenados InceptionV3, MobileNetV2 y DenseNet201 debido a que estos modelos no fueron estudiados en los trabajos de investigación de los antecedentes y además estos modelos mantienen una baja tasa de error y mejor precisión al clasificar imágenes.
5. Se reentrenaron los modelos estudiados y para que puedan generalizar mejor las imágenes de lesiones cutáneas, se realizó aumento de datos mediante transformaciones geométricas aplicadas a las imágenes lo cual permitió contar con mayor número de imágenes. La desventaja de aumentar datos fue que los modelos se demoraron más tiempo en la fase de entrenamiento.
6. Evaluando los modelos estudiados en base a la métrica exactitud se pudo comprobar que el modelo basado en InceptionV3 presentó 80.5% siendo ésta mejor que las que presentaron el modelo basado en MobileNetV2 con 76.5% y

DenseNet201 con 79.3%. Por lo tanto, el modelo basado en InceptionV3 puede servir como una buena herramienta de apoyo al diagnóstico de cáncer de piel.

7. Debido a que el conjunto de datos utilizado tiene la característica de estar desbalanceado, la métrica con la que mejor se puede evaluar este tipo de conjuntos de datos es el Score F1. Utilizando esta métrica se pudo comprobar como se muestra en la Tabla 5.5. el modelo basado en InceptionV3 presenta un mejor desempeño para clasificar imágenes de lesiones cutáneas de las clases “akiec”, “bcc”, “bkl” y “df”, mientras que el modelo basado en DenseNet201 presenta una mejor tasa de reconocimiento para las clases “mel” y “vasc”, mientras que los tres modelos se desempeñan bien para reconocer imágenes de lesiones cutáneas de la clase “nv”.
8. Los hiperparámetros considerados en el entrenamiento de los nuevos modelos basados en InceptionV3, MobileNetV2 y DenseNet201 permitieron obtener buenos resultados. Sin embargo, realizar en el futuro un mejor ajuste sería importante con la finalidad de mejorar la precisión de los modelos para clasificar imágenes de lesiones cutáneas.

6.2 Recomendaciones.

1. Balancear el conjunto de datos utilizando técnicas como las de submuestreo o sobremuestreo con la finalidad de obtener una mejor exactitud con los modelos estudiados o con otros basados en transferencia de aprendizaje.
2. Preprocesar las imágenes convirtiéndolas a escala de grises, disminución del ruido, detección de bordes o características antes del entrenamiento para intentar mejorar el desempeño de los modelos estudiados.
3. Existen muchos otros modelos de redes pre entrenadas de uso libre, considerar su aplicación para la clasificación de imágenes de lesiones cutáneas y evaluar su desempeño, por ejemplo, VGG, LeNet, etc.
4. Construir un conjunto de datos de prueba con imágenes de lesiones de piel locales y probar el modelo basado en InceptionV3 para determinar su desempeño.
5. Implementar una aplicación en base a los modelos estudiados para poner a disposición de los profesionales de la salud esta herramienta, debe ser mejorada en cuanto a su rendimiento

Bibliografía

- Adam, P. J. a. G. (2017). *Deep Learning A Practitioner's Approach*. USA: O'Reilly Media, Inc.
- Ahmad Naeem, M. S. F., Adel Khelifi, Adnan Abid. (2020). Malignant Melanoma Classification Using Deep Learning: Datasets, Performance Measurements, Challenges and Opportunities. *IEEE Access*. doi:10.1109/ACCESS.2020.3001507
- Aurélien, G. (2019). *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*. USA: O'Reilly Media, Inc.,.
- Aurobindo Gupta, S. T., Ajay Rana. (2020). Study of Melanoma Detection and Classification Techniques. *2020 8th International Conference on Reliability, Infocom Technologies and Optimization*.
- B. Sreedhar, M. S. B. E. (2020). A Comparative Study of Melanoma Skin Cancer Detection in Traditional and Current Image Processing Techniques. *Proceedings of the Fourth International Conference on I-SMAC, IEEE Xplore*. doi:10.1109/I-SMAC49090.2020.9243501
- Binder, A. (2021). *Herramienta de reconocimiento y clasificación de lesiones cutáneas basada en redes neuronales convolucionales para asistir en el diagnóstico de cáncer de piel tipo melanoma.*, Universidad Nacional de San Martín,
- Castañeda Gameros Paola, E. T., Juliana. (2016). El cáncer de piel, un problema actual. *Revista de la Facultad de Medicina. UNAM*, 59, 6-14.
- Chollet, F. (2018). *Deep Learning with Python*: Manning Publications Co.
- Cristina, P. L. (2019). *Detección precoz de cáncer de piel en imágenes basado en redes convolucionales*. Universidad Autónoma de Madrid,
- de Haas ER, N. T., de Vries E. (2010). Population education in preventing skin cancer: from childhood to adulthood. *Journal of Drugs in Dermatology*.
- E. Nasr-Esfahani, S. S., N. Karimi, S.M.R. Soroushmehr, M.H. Jafari, K. Ward, K. Najarian. (2016). Melanoma Detection by Analysis of Clinical Images Using Convolutional Neural Network. *IEEE*.
- F. Nachbar et al. (1994). The ABCD rule of dermatoscopy: high prospective value in the diagnosis of doubtful melanocytic skin lesions. *J. Am. Acad. Dermatol*.
- Giorzino Maikol Tejada Layme, R. P. G. C. (2020). *Arquitectura de Red Neuronal Convolucional para Diagnóstico de Cáncer de Piel*. Universidad Tecnológica del Perú,
- Huang Gao, L. Z., van der Maaten Laurens, Q. Weinberger Kilian. (2018). Densely Connected Convolutional Networks. *Computer Vision and Pattern Recognition*.
- Jia Deng, e. a. (2019). ImageNet: A Large-Scale Hierarchical Image Database.
- Liangqu Long, X. Z. (2022). *Beginning Deep Learning with TensorFlow*: Apress.
- Michelucci, U. (2019). *Advanced Applied Deep Learning: Convolutional Neural Networks and Object Detection*. Dübendorf, Switzerland: Apress.
- Minsa. (2019). Epidemiológica S. *Boletín Epidemiológico Del Perú*.
- Mirza Rahim Baig, T. V. J., Nipun Sadvilkar, Mohan Kumar Silaparasetty, and Anthony So. (2020). *The Deep Learning Workshop*. Birmingham B3 2PB, UK: Packt Publishing.
- P. Tschandl, C. R., H. Kittler. (2018). *The HAM10000 dataset, a large collection of multisource dermatoscopic images of common pigmented skin lesions*. Retrieved from: <https://doi.org/10.7910/DVN/DBW86T>
- Rogers Simón, G. M. (2017). *A First Course in Machine Learning*: Chapman & Hall/CRC.

-
-
- Rosendahl Cliff, M. A. (2019). *Dermatoscopy and Skin Cancer*.: Scion Publishing Ltd.
- Salazar, J. R. (2020). *Advanced Data Science and Analytics with Python*: CRC Press Taylor & Francis Group.
- Salud., M. d. (2013). *Análisis de la situación del cáncer en el Perú*. Retrieved from https://www.dge.gob.pe/portal/docs/asis_cancer.pdf
- Sandler Mark, H. A., Zhu Menglong, Zhmoginov Andrey, Chen Liang-Chieh (2019). MobileNetV2: Inverted Residuals and Linear Bottlenecks. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*,, pp. 4510-4520.
- Sebastian, D. (2018). *Aprendizaje Automático*.
- Society, A. C. (2020). Estadísticas importantes sobre el cáncer de piel tipo melanoma. Retrieved from <https://www.cancer.org/es/cancer/tipos/cancer-de-piel-tipo-melanoma/acerca/estadisticas-clave.html>
- Skin Cancer Foundation | Skin Cancer Information (2023). Fecha de acceso: Oct. 16, 2023. [Online]. Dirección: <https://www.skincancer.org/skincancer-information/skin-cancer-facts>
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Tarrés, L. (2019). *Clasificación de lesiones en la piel con ensemble de redes neuronales residuales*. Universitat Politècnica de Catalunya, Barcelona.
- Tiagy, V. (2018). *Understanding Digital Image Processing*. India: CRC Press Taylor & Francis Group.
- Vasilev, I. (2019). *Advanced Deep Learning with Python*. Birmingham, UK.: Packt Publishing.