

Sistema para el apoyo en el diagnóstico de enfermedades cutáneas crónicas

Cristian Morillo Losada
Máster en Bioinformática
Universidade da Coruña
A Coruña, España
c.morillo@udc.es

Alejandro Puente Castro
Facultad de Informática
Universidade da Coruña
A Coruña, España
a.puentec@udc.es

Cristian Robert Munteanu
Facultad de Informática
Universidade da Coruña
A Coruña, España
c.munteanu@udc.es

Resumen—La rosácea y la psoriasis son dos enfermedades inflamatorias crónicas de la piel con una prevalencia significativa a nivel global. La rosácea se manifiesta con enrojecimiento facial y lesiones inflamatorias, mientras que la psoriasis se caracteriza por placas eritematosas y escamas. Ambas condiciones tienen una base etiológica compleja, incluyendo factores genéticos, ambientales e inmunológicos. La diferenciación temprana es esencial para proporcionar el tratamiento adecuado. Las Redes Neuronales Convolucionales (CNNs) son una herramienta eficaz capaz de extraer automáticamente características visuales complejas, lo cual es necesario para distinguir entre imágenes dermatológicas similares lo que las hace especialmente aptas para este tipo de clasificación. Este estudio se enfocó en la búsqueda de modelos predictivos para la rosácea y la psoriasis, utilizando diversas arquitecturas de CNNs pre-entrenadas mediante *Transfer Learning*. Se evaluaron y compararon modelos que incluyeron una configuración manual de CNN, ResNet18, Vgg16, EfficientNetV2 y MobileNetV2. Para una evaluación exhaustiva, se implementó la técnica de validación cruzada Leave-One-Out (LOO). Los resultados mostraron que la CNN configurada manualmente obtuvo un rendimiento superior del 96% frente a las arquitecturas pre-entrenadas. El análisis estadístico reveló diferencias significativas entre los modelos, respaldando la superioridad de la CNN propuesta.

Palabras clave—Rosácea, psoriasis, CNNs, dermatología, transfer learning, Leave-One-Out

I. INTRODUCCIÓN

La rosácea es una enfermedad inflamatoria crónica de la piel que afecta aproximadamente al 10% de la población mundial y se presenta generalmente en adultos de edad media y avanzada, con una mayor incidencia en mujeres que en hombres [1]. La enfermedad se caracteriza por episodios de enrojecimiento facial, conocidos como eritema, acompañados de lesiones inflamatorias como pústulas y pápulas [2]. La etiología de la rosácea puede deberse a factores genéticos, ambientales y microbiológicos [3], [4]. Otros factores como la exposición a la luz solar, el consumo de alcohol y el estrés emocional también pueden desencadenar los síntomas [5]. La rosácea se presenta en varias formas clínicas, clasificadas en distintos subtipos: eritematotelangiectásica, papulopustular, fimatosa y ocular [6]. El tratamiento debe ser individualizado y puede incluir medidas tópicas y sistémicas. Una comprensión completa de los aspectos clínicos y etiológicos es crucial para el manejo efectivo de la enfermedad.

La psoriasis es una enfermedad inflamatoria crónica de la piel que afecta al 2-3% de la población mundial y se caracteriza por la proliferación excesiva de queratinocitos y la

inflamación crónica de la piel, lo que produce placas eritematosas y cubiertas de escamas plateadas [7]. Su etiología es compleja y depende de factores genéticos, inmunológicos y ambientales [8]. Se ha demostrado que la respuesta inmune mediada por linfocitos T juega un papel importante en el desarrollo y mantenimiento de las lesiones [9]. Factores ambientales como infecciones, el estrés y ciertos medicamentos también influyen en los brotes de psoriasis [10]. También se manifiesta en diversas formas clínicas: psoriasis en placas, pustulosa, guttata y eritrodérmica [11]. Su tratamiento también implica el uso de medicamentos tópicos y sistémicos y debe ser individualizado [12].

La diferenciación entre la rosácea y la psoriasis es fundamental para el tratamiento de los pacientes. Aunque ambas afecciones pueden presentar síntomas similares, como el enrojecimiento facial y las lesiones inflamatorias, los tratamientos son distintos [4], [8]. La identificación temprana de la enfermedad es fundamental para proporcionar a los pacientes el tratamiento adecuado. En este contexto, las Redes Neuronales Convolucionales (CNNs, por sus siglas en inglés) son una herramienta eficaz para la diferenciación de enfermedades de la piel [13]. Las CNNs son una clase de algoritmos de aprendizaje profundo (Deep Learning) [14] que están diseñados para el procesamiento de imágenes y reconocimiento de patrones visuales complejos [15]. Su efectividad radica en su capacidad para aprender y representar automáticamente características jerárquicas a diferentes niveles de abstracción [16].

Las CNNs consisten en capas de neuronas que realizan operaciones de convolución y agrupación (*pooling*) [15]. La convolución hace el escaneo de pequeñas ventanas o filtros sobre la imagen de entrada para extraer características locales. Estos filtros extraen patrones como bordes, texturas y formas simples en las primeras capas y características más complejas en las posteriores. La operación de agrupación reduce la dimensionalidad y preserva información más relevante. Esto se logra mediante la selección del valor máximo (*max pooling*) o el cálculo promedio (*average pooling*) dentro de regiones específicas [17]. La arquitectura típica de una CNN consiste en múltiples capas de convolución seguidas de capas de agrupación alternadas, seguidas de capas completamente conectadas que procesan las características extraídas para la clasificación final [15]. Esto permite que la red aprenda representaciones complejas de las características de las imágenes. El proceso de aprendizaje de una CNN consiste en proporcionarle conjuntos de datos etiquetados (aprendizaje supervisado). Durante el entrenamiento, los pesos de las conexiones entre las neuronas se ajustan iterativamente para

minimizar la diferencia entre los valores predichos y las etiquetas verdaderas de las imágenes [15], [16].

El uso de CNNs pre-entrenadas mediante *Transfer Learning* [18], una técnica que implica aprovechar los conocimientos previamente adquiridos por la red en conjuntos de datos más grandes, ha demostrado ser efectivo en la detección y clasificación de rosácea y psoriasis [19], [20]. Al aprovechar los conocimientos previamente adquiridos por la red, se ha logrado mejorar la precisión de la clasificación en comparación con el entrenamiento desde cero. Estos enfoques han demostrado una alta precisión en la identificación de las características distintivas de cada enfermedad y han sentado las bases para futuras investigaciones en el campo de la bioinformática dermatológica.

II. OBJETIVOS

El objetivo principal de este trabajo ha sido usar técnicas de *Deep Learning* para encontrar un modelo predictivo de enfermedades de la piel como rosácea y psoriasis. Esto comprende una serie de objetivos individuales:

- Búsqueda del conjunto de datos utilizado para el entrenamiento de los modelos.
- Selección de las arquitecturas y creación de los modelos de CNNs pre-entrenadas mediante transfer learning.
- Evaluación de los resultados alcanzados con los distintos modelos.

III. MATERIALES Y METODOLOGÍA

En esta sección se explicarán los materiales utilizados y la metodología aplicada en la realización de este trabajo. Se abordará el conjunto de datos empleado, los entornos de programación que han sido necesarios, la configuración de las redes neuronales utilizadas, las métricas empleadas para validar y comparar los diversos modelos, así como los pasos ejecutados dentro de la metodología.

A. Conjunto de datos

Para realizar este trabajo se llevó a cabo una búsqueda de imágenes biomédicas de la piel (Figura 1) en distintos repositorios de imágenes dermatológicas (Tabla I). La búsqueda se centró en la zona facial, ya que esta región es la que presenta características más similares entre ambas enfermedades. Se obtuvo un conjunto de 60 imágenes de la clase psoriasis facial y 56 imágenes de la clase rosácea facial. El conjunto de datos encontrado fue reducido ya que la mayor parte de las imágenes se descartaron al pertenecer a otras partes del cuerpo, que no son objeto de estudio en este trabajo y que aumentarían la variabilidad.

Tabla I

REPOSITORIOS DE IMÁGENES

Repositorio	URL
DermNet	https://dermnetnz.org
Danderm	https://www.kkh.dk/atlas/index.html
Dermatlas	https://www.dermatlas.net
Dermatoweb	http://dermatoweb.udl.es

Tabla 1- Enlaces a los repositorios utilizados para obtener el conjunto de imágenes.

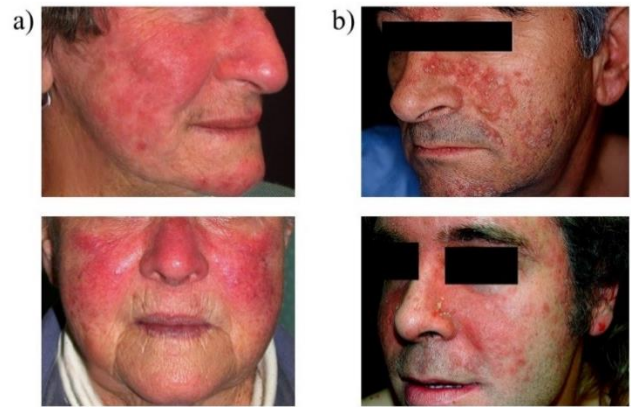


Figura 1- Imágenes representativas del conjunto de datos de a) rosácea y b) psoriasis.

B. Herramientas de desarrollo

El desarrollo de este proyecto se realizó con el lenguaje de programación Python (versión 3.9.16) [21]. En particular, se emplearon las librerías *Pytorch* [22], *fastai* [23] y *scikit-learn* [24] para la implementación de los modelos. *Pytorch*, una librería de aprendizaje profundo de código abierto proporciona herramientas flexibles para la creación de redes neuronales y el cálculo de gradientes de manera eficiente. Por otro lado, *fastai*, construida sobre *Pytorch*, ofrece una interfaz de alto nivel que simplifica el proceso de construcción, entrenamiento y evaluación de modelos de aprendizaje automático, abstrayendo gran parte de la complejidad técnica. La librería *scikit-learn* ofrece herramientas y algoritmos para la creación, entrenamiento y evaluación de los modelos de aprendizaje automático.

El entorno de desarrollo Google Colaboratory (Colab) [25] fue utilizado para llevar a cabo el trabajo en este proyecto. Colab ofrece acceso gratuito a GPU y TPU, lo que acelera significativamente el proceso de entrenamiento de modelos de aprendizaje profundo. Además, su integración con Google Drive facilita la colaboración y el intercambio de código y resultados.

C. Configuración de las redes neuronales

Para este trabajo se utilizaron diferentes estructuras de CNNs usando como base distintas redes pre-entrenadas (Figura 2). Estas redes se eligieron en base a resultados previos y su complejidad y sus características son las siguientes:

- ResNet18: pertenece a una arquitectura de CNN que utiliza bloques residuales que permiten un entrenamiento más eficaz al disminuir la pérdida de gradiente entre las capas, introduciendo conexiones de “salto” que permiten que el gradiente fluya a través de las capas sin ser atenuado [26], [27].
- Vgg16: es una arquitectura de red neuronal muy profunda y simétrica con capas convolucionales de pequeño tamaño que ha demostrado ser eficaz en clasificación de imágenes [17].
- EfficientNetV2: es una serie de arquitecturas de redes neuronales que buscan equilibrar el rendimiento y la eficiencia computacional [28], [29].

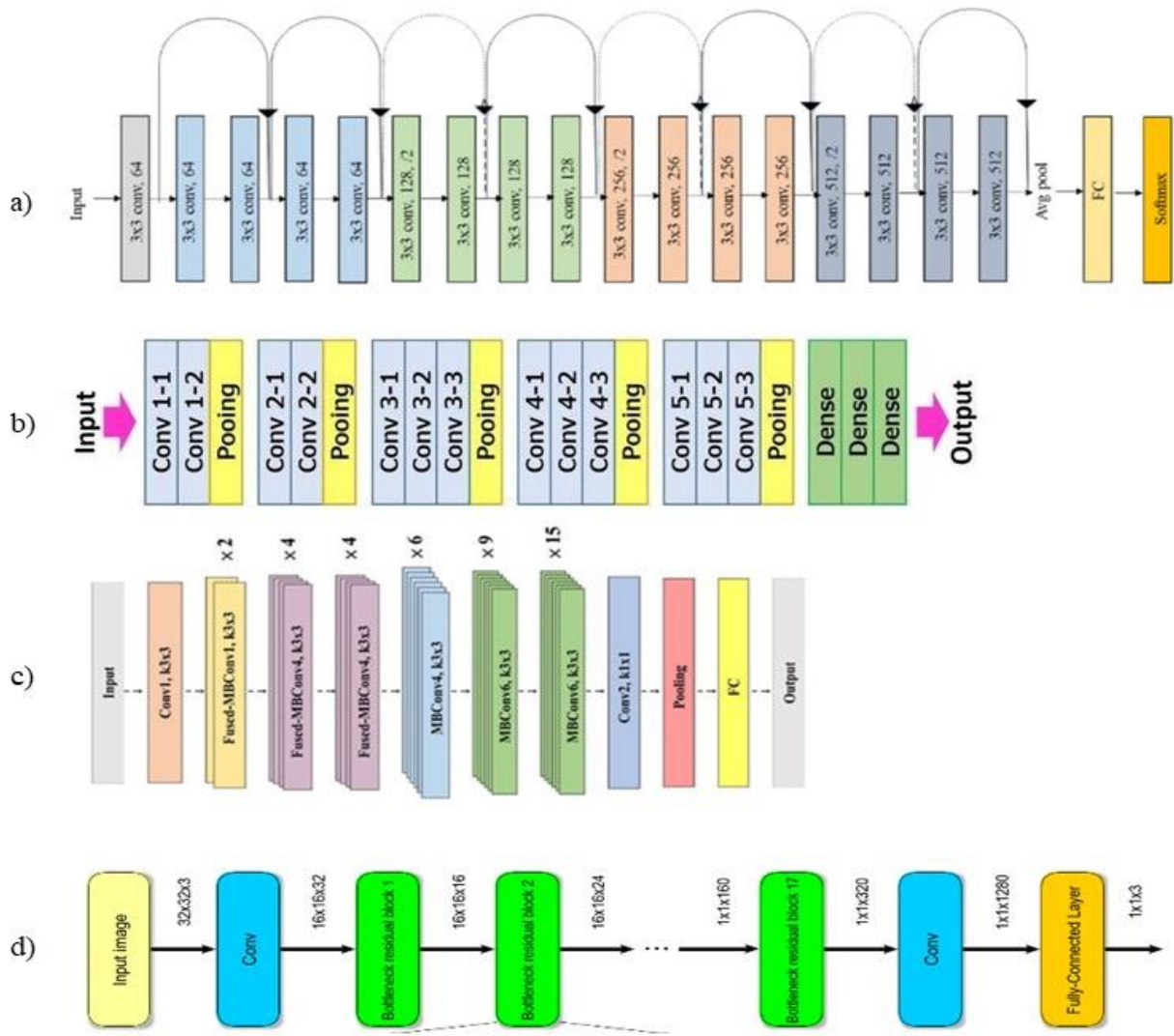


Figura 2- Comparación de la arquitectura de las capas de los cuatro modelos de CNNs utilizados: a) el modelo ResNet18 presenta 18 capas con conexiones residuales que facilitan el entrenamiento; b) el modelo Vgg16 con 16 capas; c) el modelo EfficientNetV2 que pertenece a una familia de modelos escalables y d) el modelo MobileNetV2 con una arquitectura de bloques residuales y diseñada para usarse en dispositivos móviles [17], [27], [29], [31].

- MobileNetV2: red neuronal diseñada específicamente para aplicaciones con restricciones de recursos como los dispositivos móviles [30], [31].

D. Métricas de validación

Se emplearon varias métricas de validación y comparación de los modelos con el objetivo de evaluar el rendimiento de los modelos y poder compararlos para determinar cuál es el más adecuado. Las métricas elegidas ofrecen una evaluación completa del desempeño del modelo y son las más comunes en análisis de clasificación de enfermedades:

- Accuracy (Exactitud): mide la proporción de predicciones correctas en relación con el total de predicciones realizadas.

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

Donde *VP* se refiere a verdaderos positivos, *VN* a verdaderos negativos, *FP* a falsos positivos y *FN* a falsos negativos.

- Precision (Precisión): evalúa cuantas de las predicciones predichas como positivas son realmente positivas.

$$Precision = \frac{VP}{VP + FP}$$

- Recall (Sensibilidad): evalúa la proporción de predicciones positivas reales que el modelo logra identificar y etiquetar correctamente como positivas.

$$Recall = \frac{VP}{VP + FN}$$

- F1 Score (Puntuación F1): calcula la media armónica entre precision y recall, proporcionando una medida equilibrada del rendimiento del modelo.

$$F1\ Score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

- Cohen's Kappa Score (Coeficiente Kappa de Cohen): evalúa la concordancia entre las predicciones de un

modelo y las observaciones reales teniendo en cuenta la proporción esperada por aleatoriedad.

$$\text{Cohen's Kappa Score} = \frac{P_0 - P_e}{1 - P_e}$$

Donde P_0 es la proporción observada entre los clasificadores y P_e es la proporción esperada debido al azar.

E. Metodología

La metodología de este proyecto se llevó a cabo en distintas fases que se explicarán a continuación. En primer lugar, se realizó una fase de captura y preparación de los datos, haciendo una búsqueda exhaustiva de imágenes biomédicas de la piel en diferentes repositorios de imágenes dermatológicas mencionados en la Tabla I. La búsqueda se centró en la zona facial, buscando imágenes de pacientes que se correspondieran con personas caucásicas para acotar el problema y evitar variabilidad. En un primer momento, se decidió dividir las imágenes de forma aleatoria en dos grupos, uno para el conjunto de entrenamiento (*train*) y otro para el conjunto de validación (*valid*). Las imágenes se cargaron en el entorno de Google Colab empleando la biblioteca *google.colab* y la función *drive.mount*.

La siguiente fase consistió en el diseño y desarrollo de los los experimentos. Tras diferentes pruebas con los diferentes modelos de aprendizaje profundo los resultados no fueron satisfactorios y el conjunto de datos era reducido, por lo que se optó por seguir una estrategia de validación cruzada Leave-One-Out (LOO) [32] para evaluar el rendimiento. Esta elección se basó en la ventaja que ofrece este método al maximizar la información proporcionada por el conjunto de datos, con estimaciones más robustas y una disminución en el sesgo al utilizar la totalidad del conjunto de datos. Con la técnica LOO en cada iteración se utiliza un único dato como conjunto de prueba y el resto como conjunto de entrenamiento. Este proceso se repite hasta que se haya utilizado cada imagen como conjunto de prueba una vez, lo que garantiza que la evaluación sea más exhaustiva. Para cada iteración, se generan los índices de las imágenes de entrenamiento y prueba, y se obtienen los nombres de los archivos correspondientes. Se crea un objeto *DataBlock* para cargar el conjunto total de imágenes sin dividir, redimensionarlas y aplicar *data augmentation* con el método *aug_transforms* de la biblioteca *fastai*. Este método aplica transformaciones aleatorias a las imágenes para aumentar la variabilidad, como rotaciones aleatorias, cambios de perspectiva, ajuste de contraste y brillo, entre otras. A continuación, se crea un objeto *DataLoader* que selecciona las imágenes modificadas correspondientes con los índices seleccionados para entrenamiento y validación. Después se crea un modelo de aprendizaje automático basado en las diferentes arquitecturas utilizadas: ResNet18, Vgg16, EfficientNetV2 y MobileNetV2. Luego se lleva a cabo el entrenamiento utilizando el método *fine_tune* de la biblioteca *fastai*, que realiza un ajuste ‘fino’ de los modelos actualizando los pesos de las capas, lo que mejora el rendimiento. Esta técnica de entrenamiento aprovecha el conocimiento previamente adquirido por el modelo en tareas similares, adaptando el problema de una manera más eficaz.

A parte de los modelos pre-entrenados, se utilizó una arquitectura de CNN sin pre-entrenamiento (CNN manual),

ajustando la arquitectura manualmente (Figura 3). Esta red neuronal se construye utilizando el módulo *nn.Sequential* de *Pytorch*. En este trabajo se utilizó una arquitectura que consta de tres capas convolucionales intercaladas con capas de activación *ReLU* [33] y capas de agrupación máxima (*MaxPooling*).

Posteriormente se realiza la fase de evaluación, utilizando las predicciones en el conjunto de prueba y calculando las diversas métricas de evaluación de cada iteración. Se crean listas vacías que van almacenando las predicciones, las etiquetas verdaderas y las métricas. Finalmente, se calculan las métricas de desempeño con su media y desviación estándar. Después de haber obtenido los resultados de cada modelo, se crea un diccionario con los nombres de las métricas, la media y la desviación estándar, y utilizando la librería *pandas* [34] se crea un *DataFrame* para almacenar los resultados de manera tabular en un archivo CSV, para poder analizarlos en cualquier momento.

Ya con los resultados almacenados, se procedió a realizar una última fase de comparación con un análisis estadístico comparativo entre los modelos. Primero se realiza un análisis de normalidad para cada conjunto de datos utilizando la prueba de Shapiro-Wilk [35]. Para cada modelo, se imprime el estadístico de prueba y el p-valor. Si este último es mayor que 0.05, se concluye que la muestra sigue una distribución normal. En caso contrario, se concluye que la muestra no sigue una distribución normal. Como las muestras no seguían una distribución normal, se realiza la prueba de Kruskal-Wallis [36] para determinar si hay diferencias significativas entre los grupos de datos de los diferentes modelos. Si el p-valor de esta prueba es menor de 0.05, existen diferencias significativas entre los grupos y por tanto se pasa a realizar una comparación post hoc utilizando la prueba de Tukey HSD [37] para identificar que pares de grupos difieren significativamente entre sí.

IV. RESULTADOS Y DISCUSIÓN

Según lo comentado en la metodología, se utilizaron modelos basados en CNNs utilizando diferentes redes pre-entrenadas. Tras el entrenamiento y prueba de los modelos se obtuvieron los resultados mostrados en la Tabla II.

Tabla II

RESULTADOS DE LAS MÉTRICAS DE LOS MODELOS

Modelos	Accuracy	Precision	Recall	F1-Score	Cohen Kappa
CNN manual	0.9618	0.9626	0.9617	0.9598	0.9235
	±	±	±	±	
	0.0735	0.0724	0.0743	0.0771	
ResNet18	0.8272	0.8332	0.8269	0.8209	0.6554
	±	±	±	±	
	0.0738	0.0754	0.0754	0.0777	
Vgg16	0.8456	0.8552	0.8478	0.8402	0.6921
	±	±	±	±	
	0.0824	0.0780	0.0817	0.0853	
EfficientNetV2	0.8632	0.8696	0.8645	0.8577	0.7268
	±	±	±	±	
	0.0693	0.0670	0.0714	0.0764	
MobileNetV2	0.8463	0.8540	0.8474	0.8405	0.6934
	±	±	±	±	
	0.0783	0.0759	0.0783	0.0829	

Tabla 2- Resultados de los modelos empleados para el análisis de los datos.

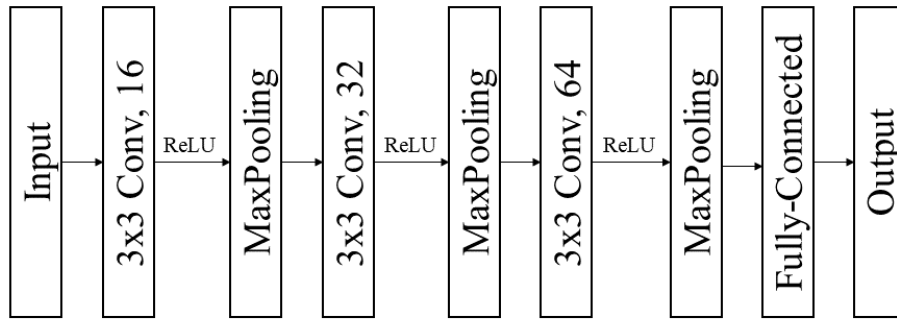


Figura 3- Esquema de la arquitectura de la red neuronal convolucional (CNN) configurada manualmente con las capas de convolución y agrupamiento (*pooling*) (elaboración propia).

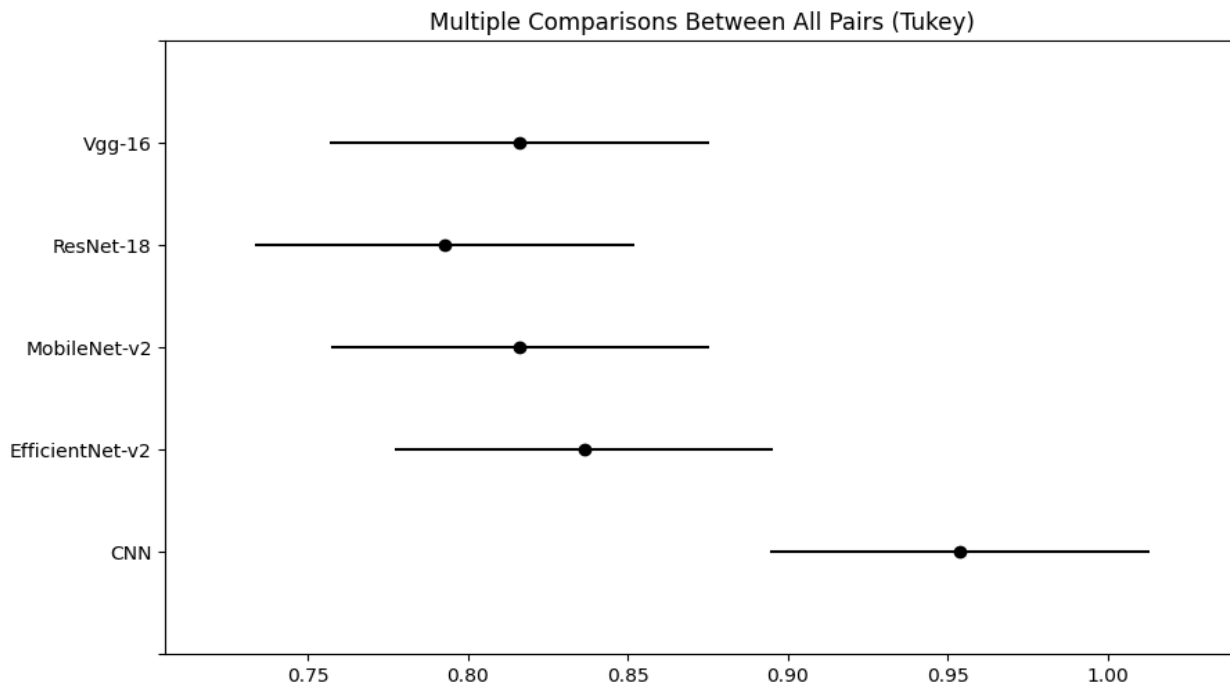


Figura 4. Gráfico con los resultados de las comparaciones post hoc realizadas mediante la prueba de Tukey HSD.

De acuerdo con lo descrito en la metodología, se consiguieron 116 imágenes tras el proceso de *data augmentation* con las cuales se entrenaron y validaron cada modelo con la estrategia de validación cruzada LOO resultando que el modelo que alcanzó un mejor rendimiento fue la CNN que se configuró de forma manual, obteniendo valores en torno al 0.96 en todas las métricas, y con un coeficiente Kappa de Cohen de 0.9235, indicando un grado significativo de concordancia. Teniendo la estrategia de validación seguida, los modelos se entrenaron y validaron 116 veces con datos distintos, por lo que se consigue aproximadamente que 112 imágenes hayan sido clasificadas correctamente. El resto de los modelos basados en redes pre-entrenadas mostraron buenos resultados, aunque con rendimientos inferiores al de la CNN. Entre estos, el que

mejores resultados dio fue EfficientNetV2 con valores alrededor de 0.86 y un coeficiente Kappa de Cohen de 0.7268. Los modelos Vgg16 y MobileNetV2 mostraron resultados similares, y el peor modelo fue el ResNet18 con resultados alrededor de 0.82 en todas las métricas.

Después de contrastar los resultados, se hizo un análisis estadístico comparativo entre los modelos para comprobar si las diferencias observadas entre los modelos son estadísticamente significativas o resultado de variaciones aleatorias. Lo primero fue evaluar la normalidad de los resultados de cada modelo mediante la prueba de Shapiro-Wilk, ya que necesitamos saber si los datos siguen una distribución normal o no para elegir las pruebas estadísticas adecuadas. Ninguno de los modelos seguía una distribución

normal, por lo que se pasó a realizar la prueba no paramétrica de Kruskal-Wallis para determinar si al menos un grupo tenía una distribución diferente al resto. El estadístico arrojó un p-valor de 0.00259 por lo que existían diferencias significativas entre los grupos. Posteriormente, se pasó a realizar una comparación múltiple de medias con la prueba de Tukey HSD [37] ya que proporciona robustez a la hora de comparar varios grupos. Los resultados (Figura 4) muestran que el modelo CNN presenta diferencias significativas con todos los modelos excepto con EfficientNetV2. El resto de modelos no presentaron diferencias significativas entre sí.

V. CONCLUSIONES Y TRABAJOS FUTUROS

Este estudio abordó la diferenciación entre dos enfermedades dermatológicas, la rosácea y la psoriasis, que comparten similitudes clínicas pero requieren tratamientos distintos. Para ello se usaron CNNs pre-entrenadas, una herramienta eficaz para el reconocimiento de patrones visuales complejos. Se implementaron y evaluaron diferentes modelos basados en arquitecturas pre-entrenadas, así como una arquitectura de CNN configurada manualmente.

Los resultados destacaron el alto rendimiento alcanzado con la CNN manual, indicando su capacidad para identificar de manera precisa y confiable las características distintivas de cada enfermedad. Asimismo, se observó que los modelos basados en arquitecturas pre-entrenadas también demostraron buenos resultados, aunque con rendimientos inferiores.

El análisis estadístico reveló diferencias significativas en el rendimiento de los modelos, resaltando la importancia de la selección de la arquitectura. Este estudio proporciona información sobre la evaluación de distintas arquitecturas de CNNs para el diagnóstico de enfermedades dermatológicas, poniendo énfasis en el papel prometedor de las CNNs en el campo de la bioinformática.

Por otro lado, el mayor limitante de este trabajo ha sido el conjunto de datos, ya que ha resultado difícil el acceso a repositorios públicos. Como trabajos futuros se podría aumentar el conjunto de datos obteniendo imágenes desde repositorios de hospitales donde se hayan revisado y clasificado por expertos. También se podría ampliar el campo de análisis hacia pacientes de diferentes etnias, así como analizar otras partes del cuerpo distintas a la zona facial.

BIBLIOGRAFÍA

- [1] J. Wilkin, M. Dahl, M. Detmar, L. Drake, A. Feinstein, R. Odom and F. Powell, "Standard classification of rosacea: Report of the National Rosacea Society Expert Committee on the Classification and Staging of Rosacea," *Journal of the American Academy of Dermatology*, vol. 46, no. 4, pp. 584-587, 2002.
- [2] S. Jarmuda, N. O'Reilly, R. Zaba, O. Jakubowicz, and A. Szkaradkiewicz, "Potential role of Demodex mites and bacteria in the induction of rosacea," *Journal of Medical Microbiology*, vol. 61, no. 11, pp. 1504-1510, 2012.
- [3] Aksoy, A. Altaykan-Hapa, D. Egemen, F. Karagöz and N. Atakan, "The impact of rosacea on quality of life: effects of demographic and clinical characteristics and various treatment modalities," *British Journal of Dermatology*, vol. 163, no. 4, pp. 719-725, Oct. 2010.
- [4] M. Steinhoff, M. Schmelz, and J. Schaubert, "Different pathophysiologies and current treatment options to address the unmet medical needs of patients with facial erythema of rosacea," *Acta Derm Venereol*, vol. 96, no. 5, pp. 579-586, Feb. 2016.
- [5] E. J. van Zuuren, Z. Fedorowicz, and B. Carter, "Interventions for rosacea," *Cochrane Database of Systematic Reviews*, no. 4, 2015.
- [6] Y. R. Woo, J. H. Lim, D. H. Cho, and H. J. Park, "Rosacea: Molecular Mechanisms and Management of a Chronic Cutaneous Inflammatory Condition," *International Journal of Molecular Sciences*, vol. 20, no. 24, p. 6156, 2019.
- [7] R. G. Langley, B. S. Krueger, and M. Griffiths, "Psoriasis: epidemiology, clinical features, and quality of life," *Annals of the Rheumatic Diseases*, vol. 64, no. suppl 2, pp. ii18-ii23, 2005.
- [8] M. A. Lowes, D. A. Bowcock, and J. G. Krueger, "Pathogenesis and therapy of psoriasis," *Nature*, vol. 445, no. 7130, pp. 866-873, 2007.
- [9] K. N. Kanada, C. W. Schupp and A. W. Armstrong, "Association between psoriasis and viral infections in the United States: focusing on hepatitis B, hepatitis C and human immunodeficiency virus," *Journal of the European Academy of Dermatology and Venereology*, vol. 27, no. 10, pp. 1312-1316, 2013.
- [10] G. G. Pujalte, L. M. Costa, A. D. Clapp, R. J. Presutti and J. C. Sluzevich, "More than skin deep: dermatologic conditions in athletes," *Sports Health*, vol. 15, no 1, pp. 74-85, 2023.
- [11] C. E. Griffiths and J. N. Barker, "Pathogenesis and clinical features of psoriasis," *The Lancet*, vol. 370, no. 9583, pp. 263-271, 2007.
- [12] C. Montesinos-Guevara, A. A. Miranda, E. Bedoya-Hurtado, C. E. Licitay, J. V. A. Franco, D. Simancas-Racines, and A. Viteri-García, "Evaluación de la calidad de guías de práctica clínica para el tratamiento de psoriasis mediante la herramienta AGREE II," *Actas Dermo-Sifiliográficas*, vol. 113, no. 3, p. 222-235, 2022.
- [13] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, Blau, H. M., and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115-118, 2017.
- [14] I.H. Sarker, "Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions," *SN Comput. Sci.*, vol. 2, pp. 420, 2021.
- [15] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *International Conference on Neural Information Processing Systems*, pp. 1097-1105, Jan. 2012.
- [16] Y. LeCun, Y. Bengio and G. E. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.
- [17] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *Cornell University*, Apr. 2015.
- [18] H. C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, ... and R. M. Summers, "Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285-1298, May 2016.
- [19] S. Park, A. L. Chien, B. Lin, and K. Li, "FACES: A Deep-Learning-Based Parametric Model to Improve Rosacea Diagnoses," *Applied Sciences*, vol. 13, no. 2, p. 970, Jan. 2023.
- [20] P. R. Kshirsagar, H. Manoharan, S. Shitharth, A. M. Alshareef, N. Albishry, and P. K. Balachandran, "Deep Learning Approaches for Prognosis of Automated Skin Disease," *Life*, vol. 12, no. 3, pp. 426, Mar. 2022.
- [21] M. Lutz, "Learning Python: Powerful Object-Oriented Programming," *O'Reilly Media, Inc.*, 2013.
- [22] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, ... and S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [23] J. Howard and S. Gugger, "Fastai: A Layered API for Deep Learning," *Information*, vol. 11, no. 2, pp. 108, 2020.
- [24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, ... and É. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.
- [25] E. Bisong, "Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners" *Berkeley, CA: Apress*, pp. 59-64, 2019.
- [26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.
- [27] F. Ramzan, M. U. G. Khan, A. Rehmat, S. Iqbal, T. Saba, A. Rehman and Z. Mehmood, "A Deep Learning Approach for Automated Diagnosis and Multi-Class Classification of Alzheimer's Disease

Stages Using Resting-State FMRI and Residual Neural Networks,” *Journal of Medical Systems*, vol. 44, pp. 1-16, 2020.

- [28] M. Tan and Q. V. Le, “EfficientNetV2: Smaller Models and Faster Training,” *International conference on machine learning. PMLR*, pp. 10096-10106, 2021.
- [29] M. Huang, and Y. Liao, “Stacking Ensemble and ECAEfficientNetV2 Convolutional Neural Networks on Classification of Multiple Chest Diseases Including COVID-19,” *Academic Radiology*, vol. 30, no. 9, 2023.
- [30] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510-4520, 2018.
- [31] U. Seidaliyeva, D. Akhmetov, L. Ilipbayeva, and E. T. Matson, “Real-Time and Accurate Drone Detection in a Video with a Static Background,” *Sensors*, vol. 20, no. 14, p. 3856, Jul. 2020.
- [32] T. Hastie, R. Tibshirani, J. H. Friedman and J. H. Friedman, “The elements of statistical learning: data mining, inference, and prediction,” *New York: springer*, vol. 2, pp. 1-758, 2009.
- [33] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 807-814, 2010.
- [34] W. McKinney, “Data structures for statistical computing in Python,” *Proceedings of the 9th Python in Science Conference*, vol. 445, no. 1, pp. 51-56, June 2010.
- [35] S.S. Shapiro and M.B. Wilk, “An Analysis of Variance Test for Normality (Complete Samples),” *Biometrika*, vol. 52, pp. 591-611, 1965.
- [36] W. H. Kruskal and W. A. Wallis, “Use of Ranks in One-Criterion Variance Analysis,” *Journal of the American Statistical Association*, vol. 47, pp. 583-621, 1952.
- [37] J. W. Tukey, “Comparing individual means in the analysis of variance,” *Biometrics*, pp. 99-114, 1949.