



**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA**  
**Grado Ingeniería de la Salud**

**TRABAJO FIN DE GRADO**

**Influencia de Variables Exógenas en la Detección de  
Enfermedades de Piel mediante Redes Neuronales y Análisis  
de Imágenes Médicas**

**Autor/es:**

**Paula Poley Ceballos**

**Tutor/es:**

**Belén Vega Márquez**

**Departamento:**

**Lenguajes y Sistemas Informáticos**

**Sevilla, junio de 2024**

## Resumen

Las enfermedades de la piel abarcan una amplia variedad de afecciones, desde dermatitis hasta formas agresivas como el cáncer de piel. La detección temprana de estas afecciones es crucial para asegurar un tratamiento efectivo y mejorar los resultados clínicos.

El trabajo emplea redes neuronales convolucionales, que incluyen Densenet121, MobileNetV2, ResNet50, EfficientNetB0, Xception y una CNN diseñada desde cero, para analizar imágenes médicas con el fin de mejorar la precisión y eficiencia en el diagnóstico de enfermedades cutáneas.

El cáncer de piel se posiciona como el tipo de cáncer más frecuente a nivel mundial, representando aproximadamente uno de cada tres casos de cáncer diagnosticados. Su incidencia está en aumento, fenómeno atribuido al creciente impacto observable de los efectos adversos del cambio climático. Se estima que el 90% de los casos de este tipo de cáncer pueden ser tratados de manera efectiva si se detectan en etapas tempranas.

La finalidad de este proyecto se centra en desarrollar un sistema altamente preciso para predecir siete clases específicas de cánceres de piel. Además, se investiga la influencia de variables exógenas en el rendimiento de los modelos predictivos. Se espera que este enfoque tenga un impacto positivo en la detección temprana y el tratamiento efectivo de enfermedades de la piel, especialmente el cáncer cutáneo.

Para alcanzar este objetivo, se desarrolla una interfaz de usuario que permite a los usuarios cargar imágenes de lesiones cutáneas y obtener predicciones del modelo. Asimismo, la interfaz les facilita ingresar datos de variables exógenas, lo que les permitirá explorar cómo diferentes factores afectan las predicciones del modelo.

## **Abstract**

Skin diseases encompass a wide variety of conditions, from dermatitis to aggressive forms such as skin cancer. Early detection of these conditions is crucial to ensure effective treatment and improve clinical outcomes.

The work employs convolutional neural networks, including Densenet121, MobileNetV2, ResNet50, EfficientNetB0, Xception, and a CNN designed from scratch, to analyze medical images with the aim of enhancing accuracy and efficiency in the diagnosis of skin diseases.

Skin cancer stands as the most common type of cancer worldwide, accounting for approximately one in every three diagnosed cancer cases. Its incidence is on the rise, a phenomenon attributed to the increasing observable impact of the adverse effects of climate change. It is estimated that 90% of cases of this type of cancer can be effectively treated if detected at early stages.

The purpose of this project focuses on developing a highly accurate system to predict seven specific classes of skin cancers. Additionally, the influence of exogenous variables on the performance of predictive models is investigated. It is expected that this approach will have a positive impact on the early detection and effective treatment of skin diseases, especially skin cancer.

To achieve this goal, a user interface is developed that allows users to upload images of skin lesions and obtain predictions from the model. Additionally, the interface facilitates the input of exogenous variable data, enabling users to explore how different factors affect model predictions.

## Agradecimientos

Quiero expresar mi agradecimiento a las personas que han sido y continúan siendo fundamentales en mi vida y en mi trayectoria académica.

En primer lugar, agradezco a mis padres, hermanas y familia. Su amor incondicional, sacrificio y apoyo constante han sido los pilares sobre los cuales he construido cada uno de mis logros. Cada sacrificio y cada palabra de aliento han sido mi luz en los momentos más oscuros y mi motivación en los desafíos. Vuestra presencia en mi vida es un regalo del que nunca dejaré de estar agradecida.

A mis amigos, mi familia elegida, les agradezco cada risa compartida, cada aventura vivida juntos y cada momento de apoyo mutuo. Asimismo agradezco a mi compañero de vida, su apoyo constante y amor han sido mi refugio en los momentos de incertidumbre

A todos los compañeros de carrera que he conocido, tanto en Málaga como en Sevilla, que han hecho mi camino más llevadero y a los profesores cuya pasión por enseñar elevó la experiencia de aprendizaje en su materia, motivándome a continuar mi camino educativo con entusiasmo.

Por último agradecer a Belén, mi tutora del TFG. Su compromiso, apoyo y orientación han sido fundamentales para llevar a cabo este proyecto con éxito.

# Índice general

<b>Resumen.....</b>	<b>2</b>
<b>Abstract .....</b>	<b>3</b>
<b>1.1. Motivación.....</b>	<b>13</b>
<b>1.2. Objetivos .....</b>	<b>13</b>
<b>1.3. Alcance y relevancia del estudio .....</b>	<b>14</b>
<b>2. Planificación .....</b>	<b>15</b>
<b>2.1. Estructura de trabajo.....</b>	<b>15</b>
<b>2.2. Planificación temporal .....</b>	<b>19</b>
<b>2.3. Presupuesto del proyecto.....</b>	<b>24</b>
<b>2.3.1. Coste del personal.....</b>	<b>24</b>
<b>2.3.2. Coste de software .....</b>	<b>27</b>
<b>2.3.3. Coste de hardware.....</b>	<b>28</b>
<b>2.3.4. Costes indirectos.....</b>	<b>29</b>
<b>2.3.5. Coste total .....</b>	<b>29</b>
<b>2.4. Gestión de riesgos .....</b>	<b>30</b>
<b>2.4.1. Riesgo individual del proyecto (R1) .....</b>	<b>35</b>
<b>2.4.2. Riesgo general del proyecto (R2) .....</b>	<b>37</b>
<b>2.4.3. Riesgo relacionado con evento (R3) .....</b>	<b>39</b>
<b>2.5. Desviación final del proyecto .....</b>	<b>41</b>
<b>3. Contexto .....</b>	<b>43</b>
<b>3.1. Enfermedades de la piel .....</b>	<b>43</b>
<b>3.2. Inteligencia artificial .....</b>	<b>48</b>
<b>3.3. Deep learning.....</b>	<b>50</b>
<b>3.3.1. Aplicaciones del Deep Learning .....</b>	<b>51</b>
<b>3.4. Redes neuronales.....</b>	<b>52</b>
<b>3.4.1. Estructura de una red neuronal .....</b>	<b>54</b>
<b>3.4.2. Tipos de redes neuronales.....</b>	<b>56</b>

<b>3.5. Redes neuronales convolucionales .....</b>	58
<b>3.5.1. Estructura de las CNN .....</b>	59
<b>3.5.2. Aplicación de las CNN para imágenes médicas .....</b>	61
<b>3.5.3. CNN preentrenadas .....</b>	63
<b>3.6. Variables exógenas .....</b>	69
<b>3.7. Estado del arte.....</b>	70
<b>4. Metodología .....</b>	72
<b>4.1. Herramientas y tecnologías .....</b>	72
<b>4.1.1. Lenguaje de programación .....</b>	72
<b>4.1.2. Entorno de desarrollo.....</b>	72
<b>4.1.3. Obtención de dataset .....</b>	73
<b>4.1.4. Librerías .....</b>	74
<b>4.2. Análisis de datos .....</b>	76
<b>4.3. Preprocesamiento del dataset.....</b>	80
<b>4.4. Implementación de la red neuronal convolucional .....</b>	85
<b>4.5. Algoritmos de hiperparametrización.....</b>	87
<b>4.6. Métricas.....</b>	88
<b>4.6.1. Matriz de confusión .....</b>	88
<b>4.6.2. Exactitud .....</b>	90
<b>4.6.3. Precisión .....</b>	91
<b>4.6.4. Recall .....</b>	92
<b>4.6.5. F1-score.....</b>	92
<b>5. RESULTADOS .....</b>	96
<b>5.1. Recopilación de datos.....</b>	96
<b>5.2. Resultado de entrenamiento.....</b>	96
<b>5.3. Resultado de evaluación.....</b>	99
<b>5.3.1. DenseNet121 .....</b>	100
<b>5.3.2. MobileNetV2 .....</b>	100
<b>5.3.3. Resnet50.....</b>	101

<b>5.3.4.</b>	<b>EfficientNetB0</b> .....	101
<b>5.3.5.</b>	<b>Xception</b> .....	102
<b>5.3.6.</b>	<b>CNN desde cero</b> .....	102
<b>5.4.</b>	<b>Comparación y discusión de resultados</b> .....	103
<b>5.5.</b>	<b>Plataforma web predicción</b> .....	105
<b>5.5.1.</b>	<b>Partes de la aplicación</b> .....	105
<b>5.5.2.</b>	<b>Requisitos de la aplicación</b> .....	107
<b>5.5.2.1.</b>	<b>Requisitos funcionales</b> .....	107
<b>5.5.2.2.</b>	<b>Requisitos no funcionales</b> .....	109
<b>5.5.3.</b>	<b>Pruebas realizadas</b> .....	110
<b>5.5.4.</b>	<b>Interfaz gráfica de la plataforma</b> .....	113
<b>5.5.5.</b>	<b>Detalles de implementación</b> .....	117
<b>6.</b>	<b>Conclusión</b> .....	121
<b>6.1.</b>	<b>Trabajo futuro</b> .....	122
	<b>Bibliografía</b> .....	124

## Índice de tablas

Tabla 1: Diccionario EDT con descripción de cada nivel o fase del proyecto.....	18
Tabla 2: Cronograma fechas del trabajo.....	18
Tabla 3: Planificación temporal- Investigación y análisis I .....	19
Tabla 4: Planificación temporal- Investigación y análisis II .....	19
Tabla 5: Planificación temporal- Investigación y análisis III .....	19
Tabla 6: Planificación temporal- Desarrollo del modelo I .....	20
Tabla 7: Planificación temporal- Desarrollo del modelo II .....	20
Tabla 8: Planificación temporal- Resultados I.....	20
Tabla 9:Planificación temporal- Resultados II.....	21
Tabla 10: Planificación temporal- Resultados III.....	21
Tabla 11: Planificación temporal- Documentación I .....	21
Tabla 12: Planificación temporal del proyecto .....	22
Tabla 13: Salario medio en Andalucía y tareas con respecto el perfil profesional... .....	24
Tabla 14: Coste estimado del personal .....	25
Tabla 15: Coste estimado del personal por actividad en proyecto de desarrollo del trabajo .....	26
Tabla 16: Fases tareas-coste .....	27
Tabla 17: Desglose de costos totales .....	29
Tabla 18: Escala de consecuencia o gravedad de riesgos según la guía PMBOK . .....	32
Tabla 19: Escala de probabilidad de riesgos según la guía PMBOK .....	33
Tabla 20: Gestión de riesgos: Factor de impacto .....	34
Tabla 21: Riesgo individual del proyecto .....	36
Tabla 22: Matriz de riesgo individual del proyecto .....	37
Tabla 23: Riesgo general del proyecto .....	38
Tabla 24: Matriz de riesgo general del proyecto .....	39
Tabla 25: Riesgo relacionado con evento.....	40
Tabla 26: Matriz de riesgo relacionado con evento.....	41
Tabla 27: Desviación final del proyecto .....	42
Tabla 28: Nº muertes por melanoma maligno en España [7] .....	44
Tabla 29: Células cutáneas : Benignas y malignas [8].....	44
Tabla 30: Tipos de células cutáneas del conjunto de datos.....	48
Tabla 31: Descripción detallada de las capas en una Red Neuronal [25] .....	55
Tabla 32: Clasificación redes neuronales por el número de capas [27] .....	57
Tabla 33: Clasificación redes neuronales por tipo de conexiones [27].....	57
Tabla 34: Clasificación redes neuronales por grado de conexiones [27] .....	58

Tabla 35: Clasificación redes neuronales por tipo de arquitectura [27] .....	58
Tabla 36: Variables exógenas y endógenas del conjunto de datos .....	70
Tabla 37: Librerías utilizadas en el trabajo .....	75
Tabla 38:Descripción de las variables junto con los valores únicos.....	77
Tabla 39: Resumen parámetros utilizados en la implementación de la CNN .....	85
Tabla 40: Comparación de hiperparámetros entre modelos preentrenados y CNN desde cero [43].....	88
Tabla 41: Descripción de los elementos de evaluación con representaciones gráficas [50] .....	89
Tabla 42: Resultado de entrenamiento de modelos con variables exógenas vs sin variables exógenas .....	98
Tabla 43: Resultado de evaluación DenseNet121 .....	100
Tabla 44: Resultado de evaluación MobileNetV2 .....	100
Tabla 45 : Resultado de evaluación ResNet50.....	101
Tabla 46: Resultado de evaluación EfficientNetB0 .....	102
Tabla 47: Resultado de evaluación Xception.....	102
Tabla 48: Resultado de evaluación CNN desde cero .....	102
Tabla 49: Tabla resultados de f1 score diferencia entre modelos con y sin variables exógenas.....	103
Tabla 50: Tabla resultados de f1 score de los modelos.....	103
Tabla 51: Pruebas realizadas.....	110

# Índice de códigos

Código 1: Implementación preprocesamiento del modelo con variables exógenas I .....	83
Código 2:Implementación preprocesamiento del modelo con variables exógenas II .....	83
Código 3:Implementación preprocesamiento del modelo sin variables exógenas I .....	84
Código 4:Implementación preprocesamiento del modelo sin variables exógenas II .....	84
Código 5: Implementación preprocesamiento del modelo solo con las variables exógenas "age" y "dx_type" .....	84
Código 6:Implementación CNN con variables exógenas, en concreto ResNet50 ...	86
Código 7:Implementación CNN sin variables exógenas, en concreto ResNet50 ....	87
Código 8: Implementación de métricas para evaluar el rendimiento de los modelos con variables exógenas .....	93
Código 9: Implementación de métricas para evaluar el rendimiento de los modelos sin variables exógenas .....	94
Código 10:Implementación de métricas para evaluar el rendimiento de los modelos por clase .....	94
Código 11: Implementación para la visualización de la matriz de confusión por clase .....	95
Código 12: Implementación: Ejemplo de imágenes clasificada .....	95
Código 13: Bibliotecas utilizadas .....	117
Código 14: Implementación de carga del modelo.....	118
Código 15: Preprocesamiento de la imagen .....	118
Código 16: Predicciones basadas en datos exógenos .....	119
Código 17: Implementación de botones, deslizadores y selección múltiple .....	119
Código 18: Implementación presentación de resultados .....	120
Código 19:Implementación imágenes y vídeos .....	120

# Índice de figuras

Figura 1:Diccionario EDT - Estructura de desglose de trabajo .....	15
Figura 2: Diagrama de Gantt estimado.....	23
Figura 3: Gráfico fases tarea-coste .....	27
Figura 4: Gráfico coste total .....	30
Figura 5:Jerarquía de Tecnologías de Inteligencia Artificial [17].....	49
Figura 6:Anatomía de una Neurona: componentes fundamentales. [23] .....	53
Figura 7: Representación fundamental de una neurona artificial. [23] .....	53
Figura 8: Resultado del reconocimiento en una imagen mediante una red neuronal. [24] .....	54
Figura 9: Esquema del recorrido de la ventana del filtro sobre la imagen. [29] .....	59
Figura 10: Ejemplo de una red neuronal con múltiples capas convolucionales [28]	61
Figura 11: Arquitectura de DenseNet – concatenación [33].....	63
Figura 12: Los dos componentes diferentes en el modelo MobileNetV2 [35].....	65
Figura 13: Arquitectura ResNet-50 (Mukherjee 2022) [36].....	66
Figura 14: Desglose de la Arquitectura de EfficientNet-B0, incluye la repetición de los módulos dentro del corchete dos veces (x2) [38].....	67
Figura 15: Elementos esenciales de la arquitectura EfficientNet: los 5 módulos fundamentales [39] .....	68
Figura 16: Arquitectura Xception: Flujo de entrada, Flujo medio y Flujo de salida [40] .....	69
Figura 17: Distribución "cell_type": Representación en barras .....	77
Figura 18: Distribución "dx_type" y "sex": Representación en gráfico de sectores..	78
Figura 19: Distribución "localization" y "age": Representación en barras.....	78
Figura 20: Mapa de calor (Heatmap) de correlación entre variables exógenas .....	79
Figura 21: Importancia de las características en el modelo de clasificación "cell_type_idx" mediante bosques aleatorios (ordenadas por importancia) .....	80
Figura 22: División de datos para entrenamiento, validación y prueba. ....	82
Figura 23: Matriz de confusión (De los Santos 2023) .....	90
Figura 24: Matriz de confusión y cálculo de exactitud (accuracy) [51] .....	90
Figura 25: Matriz de confusión y cálculo de precisión [51].....	91
Figura 26:Matriz de confusión y cálculo de sensibilidad (Recall) [51] .....	92
Figura 27: Fórmula del F1-score [52] .....	93
Figura 28:Gráfica resultados de f1 score de los modelos .....	104
Figura 29: Partes de la aplicación web de predicción .....	106
Figura 30: Prueba selección de los botones de Menú de Navegación.....	111

Figura 31: Prueba carga de imágenes.....	111
Figura 32: Prueba selección de las variables exógenas (género, edad, localización de la lesión cutánea y tipo de diagnóstico).....	111
Figura 33: Prueba restricciones.....	112
Figura 34: Prueba predicción .....	112
Figura 35: Prueba redirección de fuente.....	113
Figura 36: Prueba reproducción de vídeo y redirección a YouTube .....	113
Figura 37: Plataforma web – Inicio .....	114
Figura 38:Plataforma web – Escáner .....	114
Figura 39: Plataforma web - Sobre el escáner I.....	115
Figura 40: Plataforma web - Sobre el escáner II.....	115
Figura 41: Plataforma web - Sobre lesiones cutáneas.....	116
Figura 42:Plataforma web - Factores de riesgo .....	116
Figura 43: Plataforma web - Prevención y detección precoz .....	117

# **1. Introducción**

---

El primer capítulo del trabajo introduce la idea principal del proyecto, contextualiza su desarrollo, expone los objetivos, destaca el alcance y la importancia del estudio, ofreciendo así una visión general del trabajo.

## **1.1. Motivación**

La detección temprana de enfermedades de la piel se ha vuelto una necesidad urgente dado el aumento esperado de casos a nivel mundial y la importancia crucial de mejorar los resultados del tratamiento y reducir la morbimortalidad asociada con estas enfermedades que según [1] es la mortalidad causada por una enfermedad.

Se ha identificado que las enfermedades dermatológicas pueden ser tan diversas como sus causas, lo que dificulta su diagnóstico preciso. Es evidente que la experiencia clínica tradicional puede complementarse con tecnologías avanzadas como la inteligencia artificial (IA) y el aprendizaje profundo (Deep learning). En este contexto, se resalta la importancia de contar con herramientas efectivas de detección, dada la alta tasa de tratamiento exitoso cuando estas enfermedades se detectan en etapas tempranas. Además, se explorará cómo factores externos como la ubicación de la lesión, la edad, el tipo de diagnóstico y el sexo del paciente pueden influir en la efectividad de los modelos predictivos, ofreciendo así una comprensión más amplia y precisa de estos trastornos cutáneos.

## **1.2. Objetivos**

El trabajo se centra en desarrollar un sistema de detección de siete afecciones cutáneas utilizando redes neuronales y análisis de imágenes médicas.

En primer lugar, se busca desarrollar un sistema de detección basado en redes neuronales convolucionales que clasifica con precisión diversas afecciones cutáneas.

El segundo objetivo, y principal, se enfoca en evaluar el impacto de variables externas, como la ubicación de la lesión, la edad ,el tipo de diagnóstico realizado y el sexo del paciente, en la precisión y eficacia de los modelos de detección. Esto proporciona información valiosa sobre cómo estos factores afectan el diagnóstico,

ofreciendo una perspectiva importante para su consideración en el tratamiento y la detección temprana.

Finalmente, se desarrolla una interfaz de usuario intuitiva que permita cargar imágenes, introducir datos sobre las variables exógenas y recibir predicciones del modelo, facilitando la comprensión de la relación entre variables exógenas y resultados de diagnóstico.

### **1.3. Alcance y relevancia del estudio .**

El alcance de este estudio comprende la recopilación de datos médicos pertinentes, el procesamiento inicial de imágenes, así como el diseño y entrenamiento de redes neuronales convolucionales (CNN) específicamente adaptadas para la clasificación de enfermedades dermatológicas utilizando el conjunto de datos HAM10000 [2], abordando siete clases específicas de cáncer de piel.

Además de las imágenes dermatoscópicas, se consideran variables exógenas como la ubicación de la lesión, la edad, el tipo de diagnóstico realizado y el sexo del paciente. Se analiza cómo estas variables impactan en la precisión del modelo como un aspecto fundamental de la investigación.

La relevancia de este estudio radica en su contribución a mejorar los métodos de diagnóstico dermatológico mediante el aprovechamiento de herramientas tecnológicas emergentes. Además, al explorar el impacto de las variables exógenas en el rendimiento del modelo, se proporciona una visión más integral y contextualizada del proceso de diagnóstico dermatológico automatizado.

La aplicación de redes neuronales y análisis de imágenes médicas en la dermatología demuestra el potencial de la inteligencia artificial en la medicina. Esto no solo mejora la precisión del diagnóstico, sino que también establece bases sólidas para futuras investigaciones y aplicaciones en otros ámbitos de la salud.

Mejorar la precisión y rapidez en la detección temprana del cáncer de piel es esencial para salvar vidas y mejorar los resultados clínicos. Este enfoque tecnológico proporciona herramientas más precisas para los profesionales de la salud, lo que podría tener un impacto directo en la efectividad de los tratamientos.

## 2. Planificación

La planificación es un elemento fundamental en la ejecución exitosa de cualquier proyecto. En esta parte del trabajo se desglosan las diversas tareas y actividades necesarias para llevar a cabo el desarrollo del proyecto de manera eficaz y ordenada. Se detalla exhaustivamente el tiempo estimado para la realización de cada una de estas actividades, proporcionando una visión clara de la cronología del proyecto.

Para los proyectos, es vital que el análisis estimado de tiempo y coste sea lo más cercano al real para evitar realizar las entregas fuera de plazo o exceder el presupuesto establecido.

### 2.1. Estructura de trabajo

En este apartado, se emplea la Estructura de Desglose de Trabajo (EDT) como la principal herramienta para descomponer jerárquicamente las tareas del proyecto. La EDT se representa en una tabla que ofrece una visión global de la secuencia y el alcance del proyecto, facilitando la comprensión de las actividades y objetivos de cada fase.

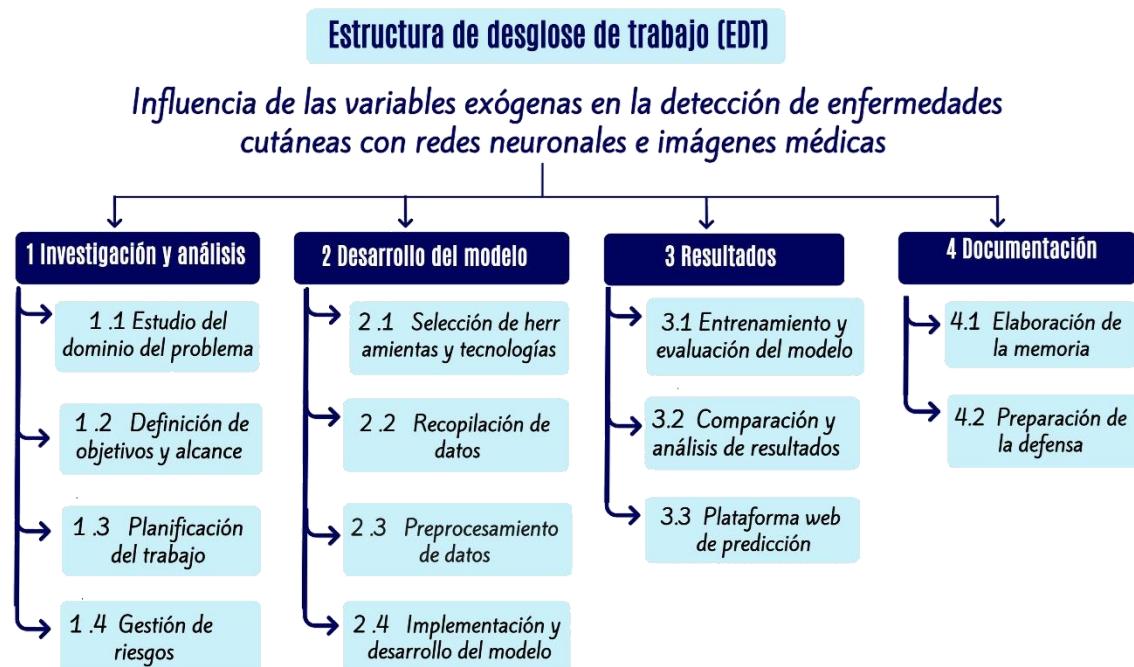


Figura 1:Diccionario EDT - Estructura de desglose de trabajo

El diccionario de la EDT proporciona una descripción de cada nivel o fase del trabajo. A continuación, se describe el significado de cada columna:

- Nivel: Este campo indica el nivel o fase del proyecto, numerado del 1 al 4.
- Código EDT: Es un código único asignado a cada nivel o fase del proyecto.
- Nombre: Muestra el nombre asociado a cada nivel o fase del proyecto. En este caso, se utilizan los nombres "Investigación y análisis", "Desarrollo del modelo", "Resultados" y "Documentación" para representar las diferentes etapas.
- Descripción: Proporciona una breve descripción de cada nivel o fase del proyecto, explicando su propósito y las actividades que se realizan en esa etapa.

Nivel	Código EDT	Nombre	Descripción
1	1	Investigación y análisis	Este nivel implica la investigación sobre el dominio del problema, incluyendo las enfermedades de la piel, las técnicas y análisis de imágenes médicas. Se analizan también las variables exógenas que pueden influir en la detección de enfermedades cutáneas.
1	1.1	Estudio del dominio del problema	Se estudian las enfermedades de la piel, su detección temprana, así como las herramientas de inteligencia artificial y análisis de imágenes médicas .
1	12	Definición de objetivos y alcance	Se establecen los objetivos del trabajo, que incluyen el desarrollo de un sistema de detección de enfermedades de la piel basado en redes neuronales y análisis de imágenes médicas, así como la evaluación de la influencia de las variables exógenas en el desempeño de los modelos.
1	1.3	Planificación del trabajo	Se realiza un plan detallado del trabajo que incluye las tareas, plazos, presupuestos, planificación temporal y de recursos.
1	1.4	Gestión de riesgos	Se identifican los posibles riesgos que pueden surgir durante el desarrollo del trabajo

			y se establecen medidas de mitigación para abordarlos en caso de que ocurran.
2	2	Desarrollo del modelo	Este nivel se centra en el desarrollo del sistema de detección de enfermedades de la piel utilizando redes neuronales y análisis de imágenes médicas.
2	2.1	Selección de herramientas y tecnologías	Se seleccionan las herramientas y tecnologías adecuadas para llevar a cabo el desarrollo del modelo, incluyendo lenguajes de programación, entornos de desarrollo y librerías específicas.
2	2.2	Recopilación de datos	Se recopilan datos de imágenes médicas que incluyan diversas afecciones de la piel, junto con metadatos que describan variables exógenas relevantes.
2	2.3	Preprocesamiento de datos	Se realizan tareas de preprocesamiento en las imágenes y se vinculan con los datos exógenos correspondientes para su posterior análisis
2	2.4	Implementación y desarrollo del modelo	Se diseñan redes neuronales convolucionales para la clasificación de enfermedades de la piel.
3	3	Resultados	En este nivel se presentan y analizan los resultados obtenidos durante el desarrollo.
3	3.1	Entrenamiento y evaluación del modelo	Se lleva a cabo el entrenamiento de los modelos y se evalúa su desempeño utilizando métricas específicas, como la precisión, f1 score y la pérdida.
3	3.2	Comparación y análisis de resultados	Se comparan los resultados obtenidos y se realizan análisis para entender la influencia de las variables exógenas en el desempeño del modelo.
3	3.3	Plataforma web de predicción	Se desarrolla una interfaz intuitiva que permite cargar imágenes de lesiones cutáneas, ingresar datos de variables exógenas y obtener predicciones del modelo.

4	4	Documentación	En este nivel se elabora la memoria del trabajo y se prepara para la defensa de este.
4	4.1	Elaboración de la memoria	Se redacta la memoria del trabajo, incluyendo una descripción detallada de cada etapa del proceso, los resultados obtenidos y las conclusiones.
4	4.2	Preparación de la defensa	Se prepara la presentación del trabajo para su defensa, destacando los aspectos más relevantes y los logros alcanzados

*Tabla 1: Diccionario EDT con descripción de cada nivel o fase del proyecto*

El siguiente cronograma detalla las fechas clave, fundamentales para el desarrollo y la conclusión exitosa del trabajo, relacionadas con las entregas. Cada etapa, desde la selección de la propuesta hasta la defensa final, está marcada con su respectiva fecha final. Proporciona así una visión general de los hitos importantes a lo largo del curso y garantiza el seguimiento de cada fase del trabajo.

Que	Cuando
Escoger propuesta	A lo largo del curso
Adjudicación	8 noviembre de 2023
Realización TFG	A lo largo del curso
Finalización de la adjudicación	1 de mayo de 2024
Entrega de la memoria	27 de mayo de 2024
Inscripción en convocatoria	29 de mayo de 2024
Entrega presentación	5 de junio de 2024
Defensa	Del 10 al 14 de junio de 2024

*Tabla 2: Cronograma fechas del trabajo*

## 2.2. Planificación temporal

En esta sección se detallan las tareas, sus descripciones y las fechas estimadas para su inicio y finalización, así como la duración prevista.

- Investigación y análisis

Tarea	Estudio del dominio del problema, objetivos y alcance
Descripción	Se estudian las enfermedades de la piel, su detección temprana, así como las herramientas de inteligencia artificial y análisis de imágenes médicas. Además se analiza el estado del arte, se establecen objetivos y el alcance de este.
Fecha de inicio estimada	14-11-2023
Fecha de fin estimada	10-12-2023
Estimación de la duración	30 horas

*Tabla 3: Planificación temporal- Investigación y análisis I*

Tarea	Planificación del trabajo
Descripción	Se realiza un plan detallado del trabajo que incluye las tareas, plazos, presupuestos, planificación temporal y de recursos
Fecha de inicio estimada	15-12-2023
Fecha de fin estimada	20-12-2023
Estimación de la duración	20 horas

*Tabla 4: Planificación temporal- Investigación y análisis II*

Tarea	Gestión de riesgos
Descripción	Se identifican los posibles riesgos que pueden surgir durante el desarrollo del trabajo y se establecen medidas de mitigación para abordarlos en caso de que ocurran.
Fecha de inicio estimada	21-12-2023
Fecha de fin estimada	28-12-2023
Estimación de la duración	20 horas

*Tabla 5: Planificación temporal- Investigación y análisis III*

- Desarrollo del modelo

Tarea	Selección de herramientas, tecnologías, recopilación y preprocesamiento de datos
Descripción	Se seleccionan herramientas y tecnologías para el desarrollo del modelo. Incluyendo lenguajes de programación, entornos de desarrollo y librerías. Además se realizan tareas de preprocesamiento en las imágenes para garantizar la calidad y coherencia de los datos.
Fecha de inicio estimada	8-01-2024
Fecha de fin estimada	24-01-2024
Estimación de la duración	50 horas

*Tabla 6: Planificación temporal- Desarrollo del modelo I*

Tarea	Implementación y desarrollo del modelo
Descripción	Se lleva a cabo la implementación y desarrollo de las CNN con el fin de analizar el impacto de las variables exógenas en la precisión de dichos modelos.
Fecha de inicio estimada	25-01-2024
Fecha de fin estimada	06-02-2024
Estimación de la duración	30 horas

*Tabla 7: Planificación temporal- Desarrollo del modelo II*

- Resultados

Tarea	Entrenamiento y evaluación del modelo
Descripción	Se lleva a cabo el entrenamiento y evaluación de las CNN con el fin de analizar el impacto de las variables exógenas en la precisión de dichos modelos.
Fecha de inicio estimada	07-02-2024
Fecha de fin estimada	07-04-2024
Estimación de la duración	65 horas

*Tabla 8: Planificación temporal- Resultados I*

Tarea	Comparación y análisis de resultados
Descripción	Se comparan los resultados obtenidos llevando a cabo un análisis para entender la influencia de las variables exógenas en el rendimiento del modelo.
Fecha de inicio estimada	08-04-2024
Fecha de fin estimada	10-04-2024
Estimación de la duración	8 horas

*Tabla 9: Planificación temporal- Resultados II*

Tarea	Plataforma web de predicción
Descripción	Se ha desarrollado una plataforma utilizando Streamlit en Visual Studio Code para facilitar la carga de imágenes de lesiones cutáneas con el fin de obtener predicciones del modelo. Además, la interfaz permite que los usuarios ingresen datos de variables exógenas, lo que les permite explorar cómo diferentes factores afectan las predicciones.
Fecha de inicio estimada	15-04-2024
Fecha de fin estimada	24-04-2024
Estimación de la duración	15 horas

*Tabla 10: Planificación temporal- Resultados III*

- Documentación

Tarea	Elaboración de la memoria y preparación de la defensa
Descripción	Se redacta la memoria del trabajo, incluyendo una descripción detallada de cada etapa del proceso, los resultados obtenidos y las conclusiones. Además, se prepara la presentación del trabajo para su defensa, destacando los aspectos más relevantes y los logros alcanzados.
Fecha de inicio estimada	26-04-2024
Fecha de fin estimada	19-05-2024
Estimación de la duración	62 horas

*Tabla 11: Planificación temporal- Documentación I*

Para facilitar el seguimiento del tiempo dedicado a cada etapa del proyecto, se ha preparado la siguiente tabla resumen que detalla las horas estimadas para cada fase. Con un total de 300 horas para el proyecto, esta tabla permite visualizar de manera clara y concisa cómo se distribuye este tiempo a lo largo de las distintas etapas.

Nombre etapas	Nº horas
<b>Investigación y análisis</b>	
Estudio del dominio del problema	30h
Definición de objetivos y alcance	
Planificación del trabajo	20h
Gestión de riesgos	20h
<b>Desarrollo del modelo</b>	
Selección de herramientas y tecnologías	50h
Recopilación de datos	
Preprocesamiento de datos	
Implementación y desarrollo del modelo	30h
<b>Resultados</b>	
Entrenamiento y evaluación del modelo	65h
Comparación y análisis de resultados	8h
Plataforma web de predicción	15h
<b>Documentación</b>	
Elaboración de la memoria	62h
Preparación de la defensa	
<b>Total de horas del proyecto = 300h</b>	

*Tabla 12: Planificación temporal del proyecto*

Utilizando los datos de las tablas previas, el diagrama de Gantt que se muestra a continuación (Figura 2) proporciona una representación cronológica clara de las actividades planificadas, estructurando visualmente las tareas y fechas estimadas. Fundamental para la gestión del tiempo y el seguimiento del trabajo.

Fase	Fecha inicio	Fecha fin	Horas	Noviembre	Diciembre	Enero	Febrero	Marzo	Abri	Mayo
<b>Investigación y análisis</b>										
Estudio del dominio del problema	14-11-23	10-12-23	30 h							
Definición de objetivos y alcance	14-11-23	10-12-23								
Planeificación del trabajo	15-12-23	20-12-23	20 h							
Gestión de riesgos	21-12-23	28-12-23	20 h							
<b>Desarrollo del modelo</b>										
Selección de herramientas y tecnologías	08-01-24	24-01-24								
Recopilación de datos	08-01-24	24-01-24	50 h							
Preprocesamiento de datos	08-01-24	24-01-24								
Implementación y desarrollo del modelo	25-01-24	06-02-24	30 h							
<b>Resultados</b>										
Entrenamiento y evaluación del modelo	07-02-24	07-04-24	65 h							
Comparación y análisis de resultados	08-04-24	10-04-24	8 h							
Plataforma web de predicción	15-04-24	24-04-24	15 h							
<b>Documentación</b>										
Elaboración de la memoria	26-04-24	19-05-24	62 h							
Preparación de la defensa	26-04-24	19-05-24								

Figura 2: Diagrama de Gantt estimado

## **2.3. Presupuesto del proyecto**

En esta sección se analizan los aspectos financieros que involucra la ejecución de este proyecto. Se detalla el presupuesto en distintas áreas, incluyendo el coste del personal, el coste del software, el coste del hardware, los gastos indirectos y en última instancia, el coste total del trabajo.

### **2.3.1. Coste del personal**

Dentro de los costes se encuentran principalmente los costes de personal y otros costes asociados a los mismos. Los costes de personal se centran en el salario del equipo de desarrollo involucrado en el trabajo.

El trabajo contará con los siguientes perfiles desempeñando funciones específicas:

Perfil	Tareas	Salario medio
Jefe de proyecto	Gestión general del proyecto.	37,25€/hora
Tester	Verificación y validación del modelo mediante pruebas.	24,15€/hora
Data Scientist	Trabajo principal de análisis de datos y desarrollo del modelo predictivo.	30,82€/hora

*Tabla 13: Salario medio en Andalucía y tareas con respecto el perfil profesional*

Estos datos de salario medio en Andalucía han sido obtenidos del "INFORME FINAL SOBRE LA CONSULTA PRELIMINAR DEL MERCADO: PERFILES PROFESIONALES ÁMBITO INFORMÁTICO" publicado por la Consejería de Agricultura, Pesca y Desarrollo Rural (CAPDER) por la Junta de Andalucía el 16 de Julio de 2018.

La siguiente tabla presenta las tarifas de los distintos perfiles involucrados en el proyecto, así como el número de horas y el total correspondiente a cada perfil. En ella se detalla la información de cada columna:

- Perfil: Se indica el nombre o título del perfil profesional.
- Tarifa: Se representa la tarifa por hora para cada perfil.

- Horas: Se muestra la cantidad de horas trabajadas por cada perfil en el proyecto.
- Coste estimado: Es el resultado de multiplicar la tarifa por hora por la cantidad de horas trabajadas, dando como resultado el total a pagar para cada perfil.

Coste del personal			
Perfil	Salario medio	Horas	Coste estimado
Jefe de proyecto	37,25€/hora	135h	5.028,75€
Tester	24,15€/hora	39h	941,85€
Data Scientist	30,82€/hora	126h	3.883,32€
<b>TOTAL</b>		<b>300h</b>	<b>9.853,92€</b>

*Tabla 14: Coste estimado del personal*

En la tabla 15 se detalla la información de cada columna de la siguiente manera:

- ID: Es el campo que identifica de forma única cada actividad del proyecto. Se utiliza para organizar y hacer un seguimiento de las distintas tareas que se deben llevar a cabo.
- Nombre: En esta columna se encuentra el nombre o descripción de la actividad.
- Duración: Este campo indica la duración estimada de la actividad, expresada en horas y/o días. Ayuda a planificar el tiempo necesario para completar cada tarea y establecer el cronograma general del proyecto.
- Perfil: Aquí se especifica el perfil profesional responsable de llevar a cabo la actividad. Indica el rol o puesto de la persona encargada de ejecutar la tarea.
- Coste: En esta columna se registra el coste estimado asociado a la realización de la actividad, expresado en euros. Representa el presupuesto asignado para cada tarea o fase del proyecto y ayuda a gestionar los recursos financieros necesarios.

Nº horas	Nombre	Perfil	Nº h/perfil	Coste		
<b>Investigación y análisis</b>						
30h	Estudio del dominio del problema	Jefe de proyecto	70h	2.607,50€		
	Definición de objetivos y alcance					
20h	Planificación del trabajo					
20h	Gestión de riesgos					
<b>Desarrollo del modelo</b>						
50h	Selección de herramientas y tecnologías	Jefe de proyecto	3h	173,39€		
		Data Scientist	2h			
	Recopilación de datos	Data Scientist	45h	1.386,90€		
	Preprocesamiento de datos					
30h	Implementación y desarrollo del modelo	Data Scientist	30h	924,60€		
<b>Resultados</b>						
65h	Entrenamiento y evaluación del modelo	Data Scientist	35h	1.803,20€		
		Tester	30h			
8h	Comparación y análisis de resultados	Data Scientist	4h	219,88€		
		Tester	4h			
15h	Plataforma web de predicción	Data Scientist	10h	428,95€		
		Tester	5h			
<b>Documentación</b>						
62h	Elaboración de la memoria	Jefe de proyecto	62 h	2.309,5€		
	Preparación de la defensa					
<b>Total de horas del proyecto = 300h</b>						
<b>COSTE TOTAL= 9.853,92 € excluyendo el impuesto sobre el valor añadido (IVA)</b>						

Tabla 15: Coste estimado del personal por actividad en proyecto de desarrollo del trabajo

La Tabla 16 presenta un desglose del coste estimado asociado a cada fase principal del proyecto de desarrollo. Cada una de estas fases tiene asignado un coste específico que refleja la suma de los costes de todas las actividades individuales correspondientes. Este desglose facilita la planificación, asegurando que los recursos se distribuyan adecuadamente entre las distintas fases.

Nombre tarea	Coste
Investigación y análisis	2.607,50€
Desarrollo del modelo	2.484,89€
Resultados	2.452,03€
Documentación	2.309,50€

*Tabla 16: Fases tareas-coste*

La Figura 3 muestra un gráfico de pastel que ilustra la distribución porcentual de los costes entre las diferentes fases del proyecto detalladas en la Tabla 16. La representación gráfica facilita la comprensión rápida de la proporción del presupuesto total asignado a cada fase.



*Figura 3: Gráfico fases tarea-coste*

### 2.3.2. Coste de software

En relación con los costos asociados al software, no será necesario adquirir licencias para el sistema operativo, ya que este viene integrado al adquirir el portátil. Además, no será necesaria la compra de ninguna licencia para ningún entorno de desarrollo, puesto que se emplearán tecnologías de código abierto o versiones gratuitas.

Para las tareas de generación de documentación, se utilizará Microsoft Office, junto con Microsoft PowerPoint para la presentación. Cabe mencionar que, a través del convenio entre la Universidad de Sevilla y Microsoft, los estudiantes de la universidad tienen acceso gratuito al portal de Office 365, el cual permite trabajar en línea con Office o descargar Microsoft Office de forma gratuita para la comunidad. Este portal ofrece la última versión de Microsoft Word, Excel, PowerPoint, OneNote, Outlook, Access, Publisher, entre otros, junto con 1 terabyte de almacenamiento en OneDrive y la posibilidad de utilizar Office Online para colaborar en tiempo real desde cualquier lugar.

### 2.3.3. Coste de hardware

El componente hardware requerido para el proyecto seleccionado teniendo en cuenta la durabilidad y eficiencia necesaria, se ha optado por el modelo HP 6S6K6EA, con una memoria RAM de 16 GB, un almacenamiento SSD de 1 TB y un procesador Intel®, brindando un rendimiento óptimo para nuestras necesidades. En la actualidad, el precio de este equipo es de 635,28 unidades monetarias.

Para calcular el costo del equipo en el proyecto, necesitamos considerar su vida útil y la duración del proyecto. Para calcular el costo anual, dividimos el costo total del equipo por su vida útil en años. Dado que la vida útil típica de un ordenador personal es de 4 años:

$$\bullet \quad \text{Costo anual} = \frac{\text{Costo total del equipo}}{\text{Vida útil}} \approx 635,28 / 4 \approx 158,82 \text{ €}$$

Dado que el proyecto tiene una duración de 7 meses, necesitamos calcular el costo del equipo para este período. Dividimos el costo anual por 12 meses para obtener el costo mensual promedio:

$$\bullet \quad \text{Costo mensual} = \frac{\text{Costo anual}}{12} = 158,82 / 12 \approx 13,24 \text{ €}$$

Finalmente, multiplicamos el costo mensual por el número de meses que dura el proyecto:

$$\bullet \quad \text{Costo total del equipo} = \text{Costo mensual} \times \text{Duración del proyecto} = 13,24 * 7 \approx 92,68 \text{ €}$$

Por lo tanto, el costo total del equipo para el proyecto, considerando su duración estimada de 7 meses, sería aproximadamente 92,68 €

### **2.3.4. Costes indirectos**

Los costes indirectos incluyen gastos como el uso de Internet y electricidad, los cuales serán nulos durante el trabajo en la facultad de informática (biblioteca), pero se aplicarán costos cuando se trabaje en el entorno personal, como el hogar. Además, se reserva un fondo mensual para posibles imprevistos, lo que establece un costo mensual de 45€.

Dado que todas las actividades se llevarán a cabo de forma digital, no se requerirá material fungible. Considerando que el proyecto tiene una duración de 7 meses, el costo total de los gastos indirectos se calcula multiplicando el costo mensual por el número de meses, resultando en un total 315€.

### **2.3.5. Coste total**

En esta parte, se presentan en la tabla 17 los diversos componentes que conforman el costo total del proyecto, incluyendo el personal, software, hardware y costes indirectos.

Tipo de coste	Precio
Personal	9.853,92€
Software	0€
Hardware	92,68€
Indirecto	315€
<b>Coste total</b>	<b>10.261,60€</b>

*Tabla 17: Desglose de costos totales*

La Figura 4 muestra un gráfico de pastel que ilustra la distribución porcentual del coste total del proyecto detallado en la Tabla 17. La representación gráfica facilita la comprensión rápida de la proporción del coste total asignado.

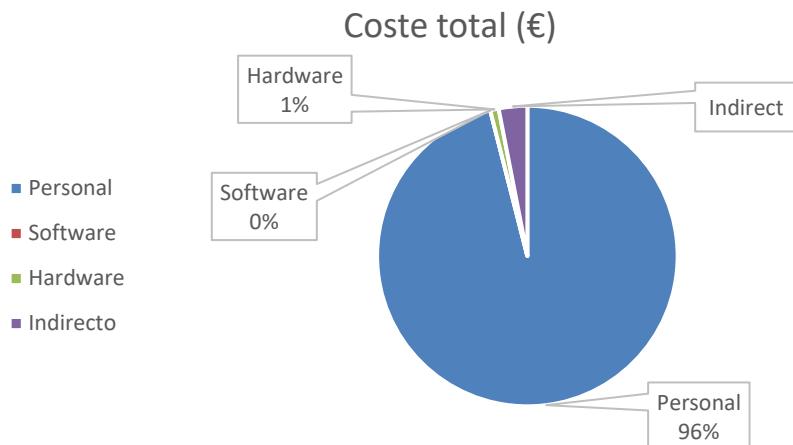


Figura 4: Gráfico coste total

## 2.4. Gestión de riesgos

[3] [4] y según la guía PMBOK, la gestión de riesgos es el proceso por el cual se identifican, se evalúan y se crean planes para disminuir o controlar esos riesgos junto con el efecto que podrían tener en el proyecto.

Al llevar a cabo la gestión de riesgos, se deben tener en cuenta varias consideraciones:

- **Planificar la gestión de riesgos:** Esto implica definir cómo se llevarán a cabo las actividades de gestión de riesgos, lo que mejora las posibilidades de éxito en los procesos posteriores.
- **Identificar los riesgos:** Se trata de determinar qué riesgos podrían afectar al proyecto y documentar sus características, un proceso iterativo que puede descubrir nuevos riesgos o cambios en los existentes.
- **Realizar análisis de riesgos:**
  - Análisis cualitativo: Priorizan los riesgos evaluando su probabilidad de ocurrencia e impacto.
  - Análisis cuantitativo: Analizan numéricamente el efecto de los riesgos sobre los objetivos del proyecto, utilizando métodos adecuados según el tiempo, presupuesto y necesidades.
- **Planificar la respuesta a los riesgos:** Crear alternativas y medidas para potenciar las oportunidades y mitigar los riesgos que podrían afectar los objetivos del proyecto.
- **Monitorear y controlar los riesgos:** Supervisar los riesgos identificados, seguir su desarrollo, descubrir riesgos nuevos y evaluar la eficacia del proceso

durante todo el proyecto, detectando tanto cambios en los riesgos existentes como la aparición de otros nuevos.

Una vez identificados los riesgos, es esencial gestionar tanto las amenazas como las oportunidades. Para ello, se pueden considerar diversas estrategias:

- **Aceptar:** No tomar medidas específicas para abordar el riesgo y asumir el impacto potencial.
- **Mitigar:** Adoptar acciones para reducir la probabilidad o el impacto de un riesgo negativo, o para aumentar la probabilidad o el impacto de un riesgo positivo.
- **Transferir:** Delegar la responsabilidad del riesgo a otra parte mediante seguros, contratos u otros acuerdos, como por ejemplo contratar un seguro.
- **Evitar:** Eliminar completamente los riesgos mediante cambios en el alcance o la planificación del proyecto.
- **Explotar:** Implementar acciones para asegurar el máximo beneficio de una oportunidad.
- **Compartir:** Distribuir la responsabilidad y el impacto del riesgo con otra parte a través de acuerdos de colaboración y mitigación conjunta.
- **Mejorar:** Incrementa la probabilidad y/o los impactos positivos de una oportunidad, identificando y minimizando las fuerzas clave que impulsan estos riesgos positivos.

En el proceso de gestión de riesgos, se categorizan los diversos riesgos detectados mediante un identificador único (ID) y se vinculan con una o más etapas del proyecto, las cuales comprenden:

- **Análisis:** Comprender los requisitos del sistema para determinar qué se necesita realmente.
- **Diseño:** Crear modelos que representen las características del sistema y cómo se implementará.
- **Codificación o implementación:** Seleccionar herramientas adecuadas, entorno de desarrollo y lenguaje de programación. Es crucial escribir un código legible y eficiente, usando estructuras de control adecuadas y comentando adecuadamente.
- **Pruebas:** Realizar pruebas de unidad, integración, aceptación y revisiones finales para encontrar errores en diferentes etapas del proyecto.

- **Entrega del producto:** Planificar la instalación o despliegue del sistema, considerando el entorno de funcionamiento, equipos necesarios, redes de interconexión y sistemas operativos. Además, se debe considerar el mantenimiento adaptativo y perfectivo para ajustar el sistema a nuevas necesidades y agregar nuevas funcionalidades.

Los riesgos se evalúan utilizando descriptores basados en la consecuencia o gravedad de su impacto en los objetivos del proyecto, como alcance, tiempo, costo y calidad. La tabla 18 consta de tres columnas:

- **Nivel:** Enumera los niveles del 1 al 5, indicando el grado de impacto del riesgo si se materializa.
- **Descriptor:** Proporciona una descripción concisa de cada nivel de impacto, resumiendo las consecuencias asociadas.
- **Descripción:** Ofrece una descripción más detallada de cada nivel de impacto, incluyendo información sobre las posibles consecuencias y efectos esperados si el riesgo se materializa.

CONSECUENCIA O GRAVEDAD		
Nivel	Descriptor	Descripción
1	Insignificante	Si el hecho llegara a presentarse, tendría consecuencias o efectos mínimos.
2	Menor	Si el hecho llegara a presentarse, tendría bajo impacto.
3	Moderado	Si el hecho llegara a presentarse, tendría medio impacto
4	Mayor	Si el hecho llegara a presentarse, tendría alto impacto.
5	Catastrófico	Si el hecho llegara a presentarse, tendría desastrosas consecuencias

Basado en la guía de PMBOK. Escala nivel de impacto del riesgo

*Tabla 18: Escala de consecuencia o gravedad de riesgos según la guía PMBOK*

La tabla 18 proporciona una escala de cinco niveles de impacto del riesgo, según la guía de PMBOK (Project Management Body of Knowledge):

1. **Nivel insignificante:** Consecuencias mínimas, impacto prácticamente irrelevante y no afecta significativamente el proyecto o situación.
2. **Nivel menor:** Bajo impacto, de poca importancia y sin grandes perturbaciones.
3. **Nivel moderado:** Impacto medio, genera algunas dificultades adicionales.

4. **Nivel mayor:** Alto impacto, significativo y capaz de causar problemas importantes o retrasos significativos.
5. **Nivel catastrófico:** Consecuencias desastrosas, impacto devastador que pone en peligro el éxito o la viabilidad del proyecto o situación.

La probabilidad de que ocurra se ve como una forma de medir cuán probable es que ocurra cierto riesgo. En la gestión de riesgos, se emplea una escala de probabilidad para detectar y valorar qué tan factible es que se materialice cada riesgo.

La tabla 19 consta de tres columnas:

- **Nivel:** Enumera los niveles del 1 al 5, indicando el grado de impacto del riesgo si se materializa.
- **Descriptor:** Proporciona una descripción concisa de cada nivel de impacto, resumiendo las consecuencias asociadas.
- **Descripción:** Ofrece una descripción más detallada de cada nivel de impacto, incluyendo información sobre las posibles consecuencias y efectos esperados si el riesgo se materializa.

PROBABILIDAD		
Nivel	Descriptor	Descripción
1	Muy improbable	El hecho de que el riesgo ocurra es una posibilidad remota.
2	No es probable	Existe una gran probabilidad de que el riesgo no ocurra.
3	Possible	El riesgo podría ocurrir o no. Las probabilidades de que suceda son 50/50.
4	Probable	Existe una gran probabilidad de que el riesgo ocurra.
5	Muy probable	Se puede estar bastante seguro de que el riesgo ocurrirá en algún momento.

*Tabla 19: Escala de probabilidad de riesgos según la guía PMBOK*

La tabla 19 proporciona una escala de probabilidad del riesgo, que va desde "Muy improbable" hasta "Muy probable". Cada nivel describe la posibilidad de que un riesgo ocurra en un proyecto o situación:

1. **Muy improbable:** Es poco probable que el riesgo ocurra, con una baja probabilidad de materialización y un impacto mínimo en el proyecto o situación.

2. **No es probable:** Existen sólidas razones para creer que el riesgo no se presentará, brindando confianza en la continuidad sin interrupciones del proyecto o situación.
3. **Possible:** Hay incertidumbre sobre si el riesgo ocurrirá o no, con una probabilidad cercana al 50/50. Se requiere un monitoreo cuidadoso para evaluar su evolución.
4. **Probable:** Existe una alta probabilidad de que el riesgo ocurra, indicada por condiciones o factores que sugieren su materialización. Es importante tomar medidas para prevenir o mitigar su impacto.
5. **Muy probable:** Se espera que el riesgo ocurra en algún momento, con circunstancias que apuntan firmemente a su realización. Es crucial tomar medidas inmediatas para abordarlo y minimizar sus consecuencias negativas.

El impacto del riesgo describe las consecuencias que podrían surgir si dicho riesgo se materializa. Se determina según la combinación de la probabilidad y la severidad del riesgo. Se representa mediante una escala de colores, que va desde el verde hasta el rojo, y se clasifica en una escala numérica del 1 al 25.

El criterio sería *Consecuencia x Probabilidad = Factor de impacto*

Escala	Descriptor	Descripción
1-6	Bajo	Los eventos de bajo riesgo son poco probables de ocurrir y, si suceden, tienen pocas consecuencias significativas. Se consideran de baja prioridad.
7-12	Medio	Estos eventos pueden causar contratiempos en el proyecto, pero pueden ser manejados con medidas preventivas y de mitigación durante la planificación. No son prioritarios pero no deben ser ignorados.
13-25	Alto	Los eventos de alto riesgo pueden tener graves consecuencias si no se gestionan adecuadamente. Son cruciales para el plan de gestión de riesgos ya que tienen una alta probabilidad de ocurrir y pueden afectar significativamente al proyecto.

Tabla 20: Gestión de riesgos: Factor de impacto

La tabla 20 proporciona una estructura para clasificar los factores de impacto del riesgo en tres niveles: bajo, medio y alto.

- **Escala:** Utiliza los rangos 1-6, 7-12 y 13-25 para representar los niveles de riesgo bajo, medio y alto, respectivamente.
- **Descriptor:** Proporciona etiquetas breves como "Bajo", "Medio" y "Alto" para resumir el nivel de riesgo y el tipo de consecuencias esperadas.
- **Descripción:** Ofrece detalles sobre cada nivel de riesgo, incluyendo la probabilidad de ocurrencia, las consecuencias esperadas y la importancia relativa del riesgo en el contexto del proyecto o empresa.

#### **2.4.1. Riesgo individual del proyecto (R1)**

Hacen referencia a los riesgos específicos y singulares asociados con cada una de las tareas o actividades del proyecto. Estos riesgos tienen la capacidad de influir en el éxito de una tarea en particular y pueden ser abordados y gestionados a nivel de esa tarea o actividad.

ID - Riesgos	Proba-bilidad	Consecuencia	Nivel de impacto	Plan de mitigación
R1.1 - Problemas en el preprocesamiento de imágenes: calidad, iluminación, artefactos...	Possible	Moderado	Medio	Implementar técnicas de preprocesamiento robustas como normalización, aumento de datos y eliminación de ruido para mejorar la calidad de las imágenes
R1.2 - Problemas con la recopilación de datos	Possible	Moderado	Medio	Realizar una investigación para identificar múltiples fuentes de datos. Utilizar técnicas de aumento de datos para aumentar la diversidad del conjunto de datos.

R1.3 - Dificultades en el diseño de la arquitectura de la red neuronal	Muy probable	Mayor	Alto	Realizar una revisión exhaustiva de la literatura para identificar arquitecturas de redes neuronales convolucionales (CNN) exitosas previamente utilizadas en problemas similares. También se puede consultar a expertos en el campo.
R1.4 – Fallos en la interfaz de usuario	Possible	Moderado	Medio	Realizar pruebas de la interfaz de usuario en diferentes navegadores. Así como también proporcionar documentación clara y tutoriales para los usuarios.

*Tabla 21: Riesgo individual del proyecto*

La tabla 21 presenta una lista de riesgos individuales del proyecto, con información detallada sobre cada uno de ellos. A continuación se describe el significado de cada columna:

- **Riesgos:** Breve descripción del riesgo específico relacionado con el proyecto.
- **ID:** Identificación única para cada riesgo.
- **Probabilidad:** La chance de que ocurra el riesgo, clasificada como "Probable" o "Improbable".
- **Consecuencia:** El impacto o resultado negativo que podría surgir si el riesgo se materializa, categorizado como "Catastrófico" o "Significativo".
- **Nivel de impacto:** Evaluación del nivel de impacto del riesgo, designando "Alto" para riesgos catastróficos y "Medio" para riesgos significativos.
- **Plan de mitigación:** Acciones sugeridas para reducir la probabilidad de ocurrencia del riesgo o mitigar sus consecuencias.

En la gestión de riesgos, la probabilidad indica la posibilidad de que ocurra un evento no deseado, expresándose cualitativa o cuantitativamente. Se evalúa utilizando datos históricos, estadísticas o experiencia experta.

Por otro lado, la severidad se refiere al impacto o consecuencias de un riesgo materializado, considerando daños, pérdidas financieras o impacto en la reputación.

Consecuencia Probabilidad	1	2	3	4	5
1	BAJO	BAJO	BAJO	BAJO	BAJO
2	BAJO	BAJO	BAJO	MEDIO	MEDIO
3	BAJO	BAJO	MEDIO  R1.1  R1.2  R1.4	MEDIO	ALTO
4	BAJO	MEDIO	MEDIO	ALTO	ALTO
5	BAJO	MEDIO	ALTO	ALTO  R1.3	ALTO
Matriz de riesgo individual del proyecto					

*Tabla 22: Matriz de riesgo individual del proyecto*

En la tabla 22, se ha elaborado una matriz donde las columnas indican la severidad (gravedad) de los riesgos, y las filas representan la probabilidad de que ocurran. Dentro de esta matriz, se han marcado y ubicado los riesgos correspondientes. Esta matriz ofrece una visión general de los riesgos individuales identificados, permitiendo una evaluación visual instantánea de su posible repercusión en el proyecto u organización.

#### **2.4.2. Riesgo general del proyecto (R2)**

Se refiere a los riesgos que impactan en la totalidad del proyecto y que pueden influir de manera notable en su éxito general del proyecto. Estos riesgos tienden a ser más estratégicos y pueden afectar a diversas tareas o actividades dentro del proyecto.

ID - Riesgos	Probabi-lidad	Consecuencia	Nivel de impacto	Plan de mitigación
R2.1 - Limitaciones de recursos computacionales	Probable	Moderado	Medio	Optimizar el código y utilizar técnicas como el entrenamiento en lotes pequeños o el uso de servicios en la nube para distribuir la carga computacional.
R2.2 - Sobreajuste de modelos (overfitting)	Possible	Moderado	Medio	Implementar técnicas de regularización como dropout y aumento de datos, y realizar una validación cruzada adecuada para evaluar el rendimiento del modelo.
R2.3 – Cambios en las tecnologías	Possible	Moderado	Medio	Realizar seguimiento regular de las tendencias tecnológicas relevantes y adaptar el proyecto.

*Tabla 23: Riesgo general del proyecto*

La tabla 23 presenta una lista de riesgos generales del proyecto, con información detallada sobre cada uno de ellos. A continuación se describe el significado de cada columna:

- **Riesgos:** Describe de manera concisa el riesgo específico asociado al proyecto.
- **ID:** Identificador único para cada riesgo.
- **Probabilidad:** La posibilidad de que ocurra el riesgo, categorizada como "Probable" o "No es probable".
- **Consecuencia:** El impacto o resultado negativo que podría ocurrir si el riesgo se materializa, clasificado como "Catastrófica" o "Mayor".

- **Nivel de impacto:** La evaluación del nivel de impacto del riesgo, siendo "Alto" para riesgos catastróficos y "Medio" para riesgos mayores.
- **Plan de mitigación:** Las medidas o acciones propuestas para reducir la probabilidad de que ocurra el riesgo o mitigar sus consecuencias.

Consecuencia Probabilidad	1	2	3	4	5
1	BAJO	BAJO	BAJO	BAJO	BAJO
2	BAJO	BAJO	BAJO	MEDIO	MEDIO
3	BAJO	BAJO	MEDIO  R2.2  R2.3	MEDIO	ALTO
4	BAJO	MEDIO	MEDIO  R2.1	ALTO	ALTO
5	BAJO	MEDIO	ALTO	ALTO	ALTO
Matriz de riesgo general del proyecto					

*Tabla 24: Matriz de riesgo general del proyecto*

En la tabla 24 , se ha creado una matriz donde las columnas reflejan la severidad de los riesgos y las filas representan la probabilidad de que ocurran. Dentro de esta matriz, se han señalado y situado los riesgos correspondientes. Esta matriz ofrece una visión general de los riesgos generales identificados, permitiendo una evaluación visual instantánea de su posible repercusión en el proyecto u organización.

#### 2.4.3. Riesgo relacionado con evento (R3)

Estos riesgos están ligados a eventos particulares, como desastres naturales, modificaciones regulatorias o fallos en equipos, entre otros. Se pueden identificar y evaluar con cierta precisión. Su solución es tomar medidas preventivas para reducir la probabilidad de que ocurran.

ID - Riesgos	Probabilidad	Consecuencia	Nivel impacto	Plan de mitigación
R3.01 - Cambio en los requisitos del proyecto	Possible	Moderado	Medio	Establecer una comunicación clara y regular con el cliente o supervisor del proyecto para gestionar expectativas y realizar ajustes según sea necesario.
R3.02 - Retrasos en la entrega	Possible	Moderado	Medio	Establecer hitos claros y realizar un seguimiento regular del progreso del proyecto. Identificar posibles problemas con anticipación y ajustar el plan de trabajo según sea necesario.
R3.03 - Caída de la red que afecta la disponibilidad de la aplicación	Possible	Catastrófico	Alto	Aumentar el número de servidores para evitar el desastre ante caídas y tener servidores en distintas ciudades.

*Tabla 25: Riesgo relacionado con evento*

La tabla 25 presenta una lista de riesgos relacionados con eventos específicos del proyecto, con información detallada sobre cada uno de ellos. A continuación se describe el significado de cada columna:

- **Riesgos:** Describe de manera concisa el riesgo específico asociado al proyecto.
- **ID:** Identificador único para cada riesgo.
- **Probabilidad:** La posibilidad de que ocurra el riesgo, categorizada como "Probable" o "No es probable".
- **Consecuencia:** El impacto o resultado negativo que podría ocurrir si el riesgo se materializa, clasificado como "Catastrófica" o "Mayor".
- **Nivel de impacto:** La evaluación del nivel de impacto del riesgo, siendo "Alto" para riesgos catastróficos y "Medio" para riesgos mayores.
- **Plan de mitigación:** Las medidas o acciones propuestas para reducir la probabilidad de que ocurra el riesgo o mitigar sus consecuencias.

Consecuencia Probabilidad	1	2	3	4	5
1	BAJO	BAJO	BAJO	BAJO	BAJO
2	BAJO	BAJO	BAJO	MEDIO	MEDIO
3	BAJO	BAJO	MEDIO <b>R3.1</b> <b>R3.2</b>	MEDIO	ALTO <b>R3.3</b>
4	BAJO	MEDIO	MEDIO	ALTO	ALTO
5	BAJO	MEDIO	ALTO	ALTO	ALTO
Matriz de riesgo relacionado con evento del proyecto					

Tabla 26: Matriz de riesgo relacionado con evento

En la tabla 26 , se ha creado una matriz donde las columnas reflejan la severidad de los riesgos y las filas representan la probabilidad de que ocurran. Dentro de esta matriz, se han señalado y situado los riesgos correspondientes. Esta matriz ofrece una visión general de los riesgos relacionados con eventos identificados, permitiendo una evaluación visual instantánea de su posible repercusión en el proyecto u organización.

## 2.5. Desviación final del proyecto

La desviación final del proyecto es una medida crucial en la gestión de proyectos que evalúa la precisión de las estimaciones de tiempo. Se calcula como la diferencia entre las horas estimadas para completar una tarea y las horas reales utilizadas para completarla.

$$\text{Desviación \%} = \frac{\text{valor real} - \text{valor estimado}}{\text{valor estimado}} \times 100\%$$

Un resultado positivo indica una desviación negativa, lo que significa que se necesitaron más horas de las estimadas. Un resultado negativo indica una desviación positiva, lo que indica que se necesitaron menos horas de las estimadas.

La Tabla 27 muestra la desviación final del proyecto, presenta los datos correspondientes a las horas estimadas y las horas reales utilizadas para las tareas

del proyecto. En total, el proyecto estima 300 horas para completarse, pero en realidad requiere 390 horas, lo que representa una desviación global del 30,00%. Esta información es esencial para entender las variaciones en la planificación y ejecución del proyecto, y proporciona una base para mejorar las estimaciones en futuros proyectos.

Tarea	Horas estimadas	Horas reales	Desviación
<b>Investigación y análisis</b>			
Estudio del dominio del problema	30 h	45 h	50,00%
Definición de objetivos y alcance			
Planificación del trabajo	20 h	20 h	0,00%
Gestión de riesgos	20 h	20 h	0,00%
<b>Desarrollo del modelo</b>			
Selección de herramientas y tecnologías	50 h	50 h	0,00%
Recopilación de datos			
Preprocesamiento de datos			
Implementación y desarrollo del modelo	30 h	40 h	33,33%
<b>Resultados</b>			
Entrenamiento y evaluación del modelo	65 h	95 h	46,15%
Comparación y análisis de resultados	8 h	8 h	0,00%
Plataforma web de predicción	15 h	20 h	33,33%
<b>Documentación</b>			
Elaboración de la memoria	62 h	92 h	48,38%
Preparación de la defensa			
<b>TOTAL</b>	<b>300 h</b>	<b>390 h</b>	<b>30,00%</b>

*Tabla 27: Desviación final del proyecto*

### **3. Contexto**

---

#### **3.1. Enfermedades de la piel .**

De acuerdo con [5] la piel, siendo el órgano más grande del cuerpo humano, desempeña múltiples funciones vitales para nuestra salud y bienestar. Actúa como una barrera protectora contra la deshidratación y la entrada de gérmenes dañinos, previniendo infecciones y manteniendo la integridad de nuestro organismo. Además, es clave en la percepción sensorial, permitiéndonos sentir sensaciones como calor, frío y dolor, y contribuye activamente en la regulación térmica corporal.

Asimismo, la piel tiene la capacidad de producir vitamina D mediante la exposición solar, esencial para la salud ósea y el sistema inmunológico. No obstante, puede ser afectada por diversos trastornos y enfermedades que van desde leves a potencialmente graves. Entre los trastornos más comunes se encuentran el acné, las quemaduras, cortes y raspaduras, caspa, eccema, urticaria, picaduras de insectos, psoriasis, cáncer de piel e infecciones cutáneas. Estas afecciones pueden provocar síntomas molestos como picazón, ardor, enrojecimiento y erupciones cutáneas, así como afectar la apariencia física.

Según [6] durante las últimas dos décadas, se ha observado un incremento del 50 por ciento en la incidencia del cáncer de piel en España, según revela un estudio realizado por la Red España de Registros de Cáncer (Redecan) y la Sociedad Española de Oncología Médica (SEOM). Este estudio proyecta que en el año 2022 se diagnosticarán aproximadamente 7.500 casos de melanoma cutáneo en el país. (16 casos por cada 100.000 personas)

Los datos muestran una tendencia preocupante en la enfermedad, con un aumento anual del 2,5 por ciento en mujeres y del 1,6 por ciento en hombres entre 2002 y 2020. Colocando al melanoma cutáneo como el noveno cáncer más común en mujeres y el decimoprimer en hombres. La incidencia es más alta en personas mayores de 65 años, que representan el 44 por ciento de los casos.

En cuanto a la mortalidad, en 2020 se registraron 1.079 fallecimientos por melanoma cutáneo, con tasas más altas en hombres que en mujeres. Este tipo de cáncer se sitúa como el 18º en mortalidad en mujeres y el 21º en hombres. Aunque la

tasa de mortalidad ha aumentado ligeramente en hombres, ha disminuido en mujeres. Las personas mayores de 65 años representan la mayoría de las defunciones.

La supervivencia de los pacientes con melanoma ha mejorado entre 2008 y 2013 en comparación con el período anterior. A continuación, se aprecia una imagen que representa el número de muertes por melanoma maligno de la piel en España desde el año 2006 hasta el 2021.

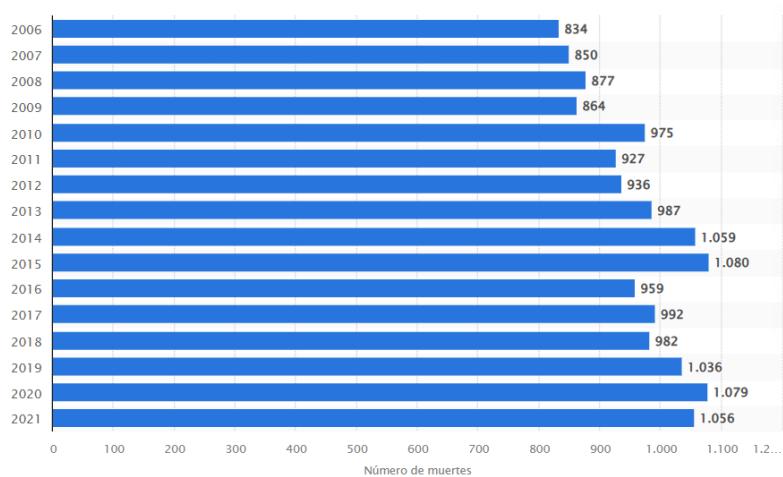


Tabla 28: Nº muertes por melanoma maligno en España [7]

El conjunto de datos comprende siete categorías distintas de células cutáneas, clasificadas en dos grupos principales: benignas y malignas.

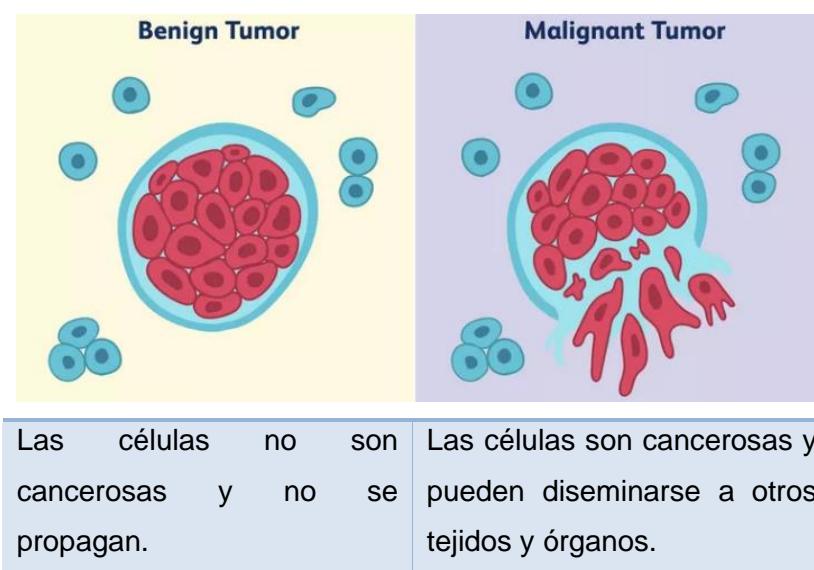


Tabla 29: Células cutáneas : Benignas y malignas [8]

Dentro de las células benignas se encuentran:

- Nevo Melanocítico (nv)
- Queratosis Benigna (bkl)
- Lesiones vasculares (vasc)
- Dermatofibroma (df)

Por otro lado, las células malignas incluyen:

- Melanoma (mel)
- Carcinoma de células basales (bcc)
- Queratosis actínicas (akiec)

Entre estos tipos malignos, el melanoma se considera el más peligroso, requiriendo una atención inmediata debido a su capacidad para propagarse por todo el cuerpo. El carcinoma de células basales, aunque raramente metastatiza, debe ser tratado con prontitud para evitar complicaciones. Las queratosis actínicas, si bien tienden a crecer lentamente, pueden ser tratadas eficazmente, a menudo sin necesidad de cirugía. [9]

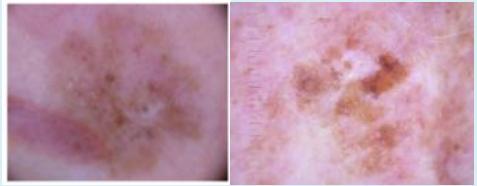
A continuación, se presentan imágenes de cada uno de los tipos de células cutáneas extraídas del conjunto de datos HAM10000, junto con una breve explicación de cada uno, además del número de imágenes que la componen.

Nombre	Descripción - Imagen	
Nv – Melanocytic nevi		
El conjunto de datos consta de 6705 imágenes.		
Los nevus melanocíticos son crecimientos benignos no cancerosos de células pigmentadas conocidas como melanocitos. Se presentan en una variedad de formas y pueden ser notablemente diferentes cuando se observan con dermatoscopio. [10]		
Mel – Melanoma		
El conjunto de datos consta de 1113 imágenes.		

El melanoma es un tipo de cáncer maligno grave que se origina en los melanocitos y puede presentarse en diversas formas, algunas invasivas y otras no (*in situ*). Cuando se detecta en etapas tempranas, puede tratarse con éxito mediante una extirpación quirúrgica simple. Hay distintas formas de melanoma, algunas invasivas y otras no (*in situ*).

Se excluyen del conjunto las formas no pigmentadas, subungueales, oculares o mucosas del melanoma. La exposición a la radiación ultravioleta aumenta el riesgo de desarrollar melanoma, especialmente en personas menores de 40 años. [9]

Bkl – Benign keratosis-like lesions



El conjunto de datos consta de 1099 imágenes.

La queratosis benigna es una clase genérica que incluye:

queratosis seborreica ("verruga senil") lentigo solar (una variante plana de la queratosis seborreica) queratosis tipo liqueno plano (QTL) (queratosis seborreica o un lentigo solar con inflamación y regresión) .

A pesar de sus diferentes apariencias dermatoscópicas, se agrupan debido a similitudes biológicas y al hecho de que a menudo se describen con el mismo término genérico en estudios histopatológicos.

Las QTL presenta un desafío ya que puede mostrar características morfológicas similares al melanoma, lo que a menudo requiere biopsias o extirpaciones para diagnóstico. [9]

Bcc – Basal cell carcinoma



El conjunto de datos consta de 514 imágenes.

El carcinoma basocelular es un tipo común de cáncer de piel, constituyendo entre el 80% y el 90% de todos los carcinomas cutáneos. Se origina en las células basales, las cuales producen nuevas células cutáneas a medida que las viejas mueren. Aunque rara vez se propaga a otras partes del cuerpo, puede extenderse de forma destructiva si no se trata a tiempo.

Se manifiesta en diversas formas morfológicas, como plana, nodular, pigmentada, quística, entre otras, todas las cuales se encuentran en el conjunto [11].

Akiec – Actinic kerastoses		
El conjunto de datos consta de 327 imágenes.		
<p>Las queratosis actínicas (queratosis solar) y el carcinoma intraepitelial (enfermedad de Bowen) son tipos comunes no invasivas del carcinoma de células escamosas que pueden tratarse localmente sin cirugía.</p> <p>Algunos expertos las consideran precursores de carcinomas de células escamosas y no carcinomas reales, pero existe acuerdo en que estas lesiones pueden progresar a carcinomas de células escamosas invasivos, que generalmente no son pigmentados. Ambos neoplasmas suelen presentar descamación en la superficie y carecer de pigmentación.</p> <p>Las queratosis actínicas son más comunes en la cara, mientras que la enfermedad de Bowen es más común en otras áreas del cuerpo.</p> <p>Dado que ambos tipos son inducidos por la luz UV, la piel circundante suele estar caracterizada por daño solar severo, excepto en los casos de enfermedad de Bowen causados por infección por el virus del papiloma humano y no por UV.</p> <p>Existen variantes pigmentadas tanto para la enfermedad de Bowen como para las queratosis actínicas, y todas ellas se encuentran en el conjunto [9, 12].</p>		
vasc – Vascular lesions		
El conjunto de datos consta de 142 imágenes.		
<p>Las lesiones vasculares cutáneas en el conjunto de datos incluyen desde angiomas cereza [13] (pequeños crecimientos rojos causados por vasos sanguíneos dilatados en la superficie de la piel) hasta angiokeratomas (lesiones benignas que resultan de la dilatación de los capilares sanguíneos en la piel) y granulomas piógenos (crecimientos de tejido que aparecen como protuberancias rojas y sangrantes en la piel). También se incluye la hemorragia . [9]</p>		
df - Dermatofibroma		
El conjunto de datos consta de 115 imágenes.		

El dermatofibroma es una lesión cutánea benigna que se caracteriza por ser una proliferación benigna o una reacción inflamatoria mínima ante el trauma. Dermatoscópicamente, se presentan como abultamientos marrones con frecuencia mostrando una zona central de fibrosis. Estas protuberancias son comunes en adultos y suelen aparecer como bultos endurecidos, típicamente en muslos y piernas, especialmente en mujeres. Por lo general, miden menos de 1,25 cm de diámetro y pueden desarrollarse múltiples dermatofibromas. Aunque son inofensivos y generalmente no causan síntomas, pueden provocar picazón o ulceración después de lesiones menores. [14]

*Tabla 30: Tipos de células cutáneas del conjunto de datos*

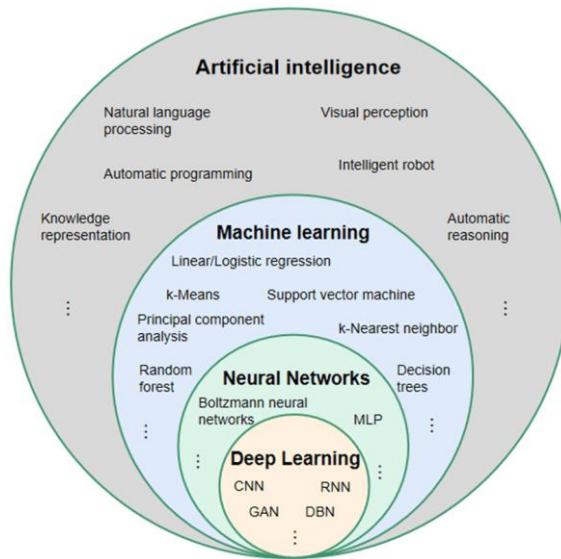
### **3.2. Inteligencia artificial**

La inteligencia artificial (IA) se define como un campo de la informática que se enfoca en desarrollar tecnología capaz de emular diversas funciones cognitivas humanas, como el razonamiento, el aprendizaje, la creatividad y la resolución de problemas. [15]

A diferencia de las concepciones anteriores que planteaban la IA como una amenaza para la humanidad, la perspectiva actual la considera como una herramienta complementaria que puede potenciar nuestras capacidades y mejorar diversos aspectos de la vida cotidiana. [16]

Los sistemas de IA están diseñados para percibir su entorno, procesar datos y tomar decisiones basadas en la información recibida, lo cual implica el uso de algoritmos y modelos especializados que les permiten aprender de experiencias pasadas y adaptarse a nuevas situaciones. Estos avances en IA han sido posibles gracias al desarrollo en el campo de la computación y la ciencia de datos.

Desde entonces, la IA ha experimentado un crecimiento exponencial y ha encontrado aplicaciones en una amplia gama de industrias y sectores. Desde la atención médica y la seguridad cibernética hasta la gestión financiera y la industria del entretenimiento, los sistemas de IA están transformando la forma en que interactuamos con la tecnología y cómo enfrentamos los desafíos cotidianos.



*Figura 5:Jerarquía de Tecnologías de Inteligencia Artificial [17]*

La figura 5 representa una estructura jerárquica que muestra cómo el campo de la inteligencia artificial se desglosa en áreas más específicas. Comienza con la inteligencia artificial en su nivel más amplio, luego se adentra en el aprendizaje automático (Machine Learning), que a su vez se desglosa en redes neuronales (Neuronal Networks). Finalmente, se llega al nivel más específico, el aprendizaje profundo (Deep Learning).

En el ámbito del aprendizaje profundo, se encuentran diversas técnicas y arquitecturas especializadas, siendo las redes neuronales convolucionales (CNN) de particular relevancia para mi trabajo puesto que este se centra en el desarrollo y entrenamiento específico de CNNs con el propósito de clasificar enfermedades de la piel. Además de evaluar cómo variables externas afectan la precisión de los modelos durante el entrenamiento y las pruebas.

[18] El Machine Learning como se ha comentado anteriormente es una rama de la inteligencia artificial que permite a los algoritmos aprender patrones a partir de datos y mejorar su rendimiento con el tiempo. Se compone de cuatro etapas: preparación de datos, selección de algoritmo, entrenamiento y uso del modelo. Los algoritmos pueden ser de regresión para datos etiquetados, árboles de decisiones para recomendaciones, clustering para datos no etiquetados, algoritmos de asociación para descubrir patrones, y redes neuronales, incluyendo las profundas, para tareas complejas como el Deep Learning.

Dependiendo de la naturaleza de los datos y del resultado deseado, existen cuatro modelos principales: supervisado, no supervisado, semisupervisado y de refuerzo. La elección de estos modelos depende de si los datos están etiquetados, si se busca identificar patrones sin etiquetas, si se requiere una combinación de ambos, o si se pretende aprender de la retroalimentación de las acciones realizadas.

### **3.3. Deep learning**

Según [18] el Deep Learning (aprendizaje profundo) es una rama avanzada del aprendizaje automático inventada por Geoffrey Hinton en 1986, que se basa en redes neuronales profundas para aprender y extraer patrones complejos de datos.

A diferencia del aprendizaje automático tradicional, el Deep Learning utiliza redes neuronales con múltiples capas de nodos de cálculo, inspiradas en el funcionamiento del cerebro humano, para realizar tareas como reconocimiento de voz, clasificación de imágenes y análisis de datos [19]. Esta técnica aprovecha grandes volúmenes de datos no etiquetados para aprender de manera no supervisada, extrayendo características jerárquicas a medida que profundiza en las capas de la red neuronal. Por ejemplo, en el reconocimiento de imágenes, una red de Deep Learning puede identificar desde características básicas como bordes y formas hasta conceptos más complejos como objetos específicos.

[20] Los algoritmos están organizados en diferentes capas neuronales, cada una con su función específica:

- Capa de entrada o input layer.
- Capa oculta o hidden layer:
- Capa de salida u output layer.

Resumiendo, mientras que el Machine Learning se basa en el aprendizaje asistido o supervisado donde se guía a la máquina para identificar patrones de forma automática, el Deep Learning implica un aprendizaje autónomo donde la máquina aprende de forma independiente a partir de la retroalimentación que recibe de sus errores.

### **3.3.1. Aplicaciones del Deep Learning**

El aprendizaje profundo ha ganado popularidad en una amplia gama de industrias y campos, incluyendo la atención médica, la farmacéutica, la ciberseguridad y la detección de fraudes, entre otros. [21]

Tres áreas destacadas en la actualidad en [22] son la visión artificial, el reconocimiento de voz y el procesamiento del lenguaje natural (NLP).

- **La visión artificial** implica el uso de técnicas de aprendizaje profundo para que las computadoras comprendan imágenes, lo que les permite moderar contenido, reconocer rostros y clasificar imágenes.
- **El reconocimiento de voz** utiliza modelos de aprendizaje profundo para analizar aspectos como el tono, el lenguaje y el acento en el habla mejorando la experiencia del cliente y haciendo más accesible la transcripción en tiempo real.
- **El procesamiento del lenguaje natural (NLP)**, las computadoras emplean algoritmos de aprendizaje profundo para analizar y extraer información de textos y documentos, útil para resumir textos, identificar opiniones expresadas (positivas o negativas) y generar contenido para asistentes virtuales y chatbots.

Algunos ejemplos de usos del aprendizaje profundo en distintos sectores incluyen:

- **Traducción automática:** Ha mejorado significativamente la traducción automática de texto e imágenes entre idiomas, permitiendo conversaciones más precisas y fluidas.
- **Asistencia sanitaria:** En medicina, se utiliza para el diagnóstico temprano y preciso de enfermedades potencialmente mortales , la personalización de tratamientos y la investigación de curas para enfermedades intratables, lo que puede reducir riesgos y optimizar costos.
- **Detección de fraude:** Fundamental para la detección de patrones anómalos y valores atípicos en las transacciones financieras habituales de los clientes, ayudando a prevenir y detectar fraudes.
- **Automóviles autónomos:** Contribuye al desarrollo de vehículos capaces de percibir el entorno y tomar decisiones seguras en tiempo real.
- **Reconocimiento visual:** Reconocimiento y clasificación de objetos en imágenes (rostros, combinación de personas, fechas etc), así como la organización y etiquetado automático de contenido multimedia.

### **3.4. Redes neuronales**

De acuerdo con [23] las redes neuronales también conocidas como redes neuronales artificiales (ANN) o sistemas neuronales simulados (SNN), son un sistema de procesamiento computacional compuesto por unidades interconectadas, cuya funcionalidad se basa en la representación simplificada de la neurona biológica. Su capacidad de procesamiento se determina por la fuerza de las conexiones entre estas unidades, llamadas pesos, los cuales se ajustan mediante el aprendizaje a partir de un conjunto de datos de entrenamiento. Inspiradas en el cerebro humano, estas redes replican la comunicación entre neuronas biológicas, permitiendo realizar tareas computacionales mucho más rápidas que los sistemas convencionales.

Como ya se ha comentado antes las neuronas son los bloques fundamentales de procesamiento en las redes neuronales, realizando diversas funciones similares a las del cerebro humano. Aunque la complejidad de las neuronas biológicas dificulta su reproducción exacta en sistemas electrónicos, se identifican cuatro componentes principales:

- Sinapsis
- Axón
- Soma (cuerpo celular)
- Dendritas → extensiones en forma de cabello del cuerpo celular, funcionan como canales de entrada, recopilando información a través de sinapsis de otras neuronas, la cual es procesada por el soma para producir una salida.

Los científicos de datos buscan comprender las capacidades naturales de las neuronas para resolver problemas no tratados por la computación convencional. Las neuronas artificiales replican las funciones básicas de las biológicas, pero el objetivo no es emular el cerebro biológico exactamente, sino aprovechar su comprensión para abordar problemas no resueltos por la computación tradicional.

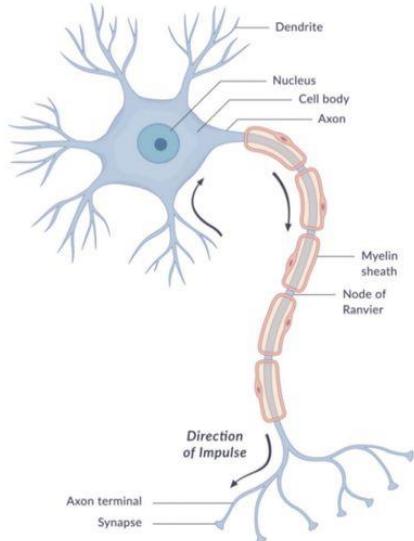


Figura 6:Anatomía de una Neurona: componentes fundamentales. [23]

Con el objetivo de replicar este proceso, las neuronas artificiales imitan las cuatro funciones básicas de las neuronas biológicas. A continuación, se presenta una imagen que ilustra una representación básica de una neurona artificial.

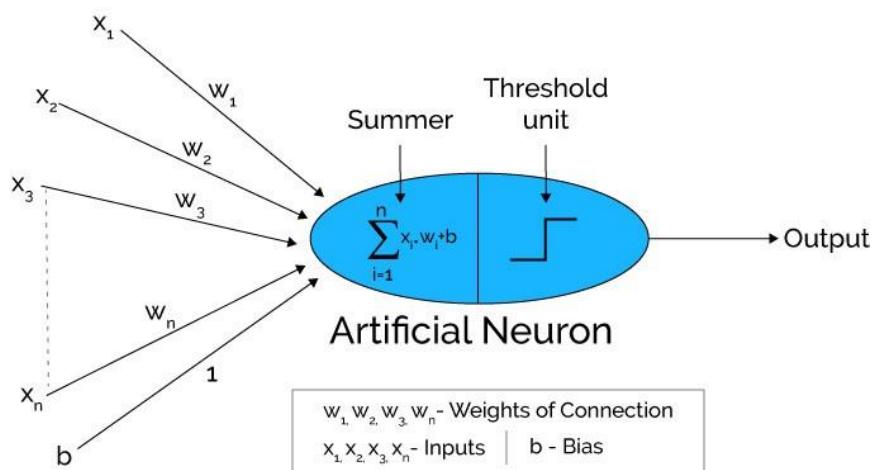


Figura 7: Representación fundamental de una neurona artificial. [23]

En la figura anterior, se observa que  $x_1, x_2, x_3 \dots X_n$  representan varias entradas de la red. Estas entradas se combinan multiplicándolas por los pesos de conexión correspondientes  $w_1, w_2 \dots w_n$ . Posteriormente, los productos se suman de manera ponderada y se pasan a través de una función de transferencia. Esta función, típicamente una unidad de umbral (Threshold unit), genera resultados que producen la salida final.

Las redes neuronales pueden aprender a realizar tareas complejas como reconocimiento de patrones, clasificación de datos, predicción y toma de decisiones. Por ejemplo, un modelo de deep learning basado en una red neuronal y entrenado con una cantidad adecuada de datos podría ser capaz de reconocer objetos en una imagen aunque no los haya visto previamente, tal como se muestra en la Figura 8.

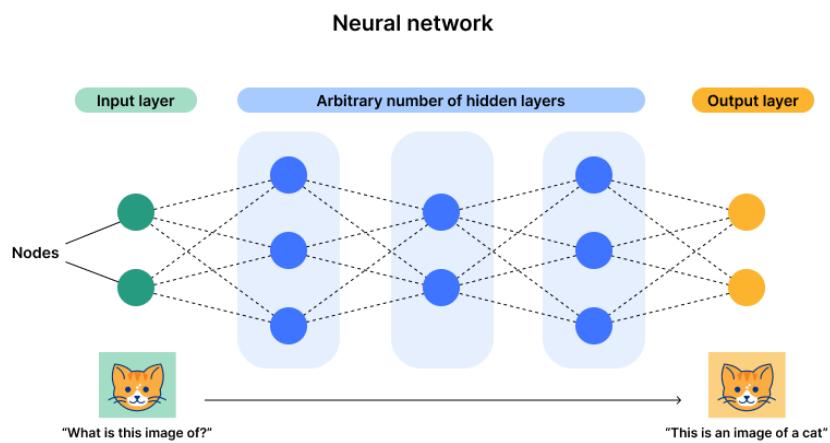


Figura 8: Resultado del reconocimiento en una imagen mediante una red neuronal.

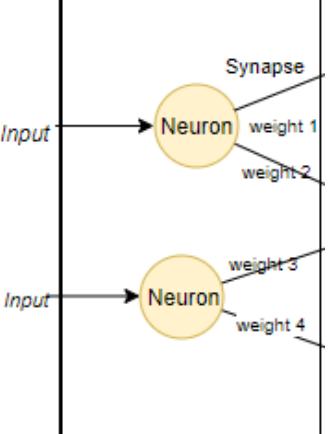
[24]

### 3.4.1. Estructura de una red neuronal

Las redes neuronales están compuestas por una serie de unidades de procesamiento llamadas "nodos" o "neuronas artificiales" organizadas en capas, que intercambian datos entre sí de manera similar a cómo las neuronas en el cerebro transmiten impulsos eléctricos.

Los nodos están organizados mínimo en tres capas distintas: Capa de entrada (input layer), capa oculta (hidden layer) y capa de salida (output layer). A continuación, se detalla este concepto y se proporciona una representación gráfica de la misma.

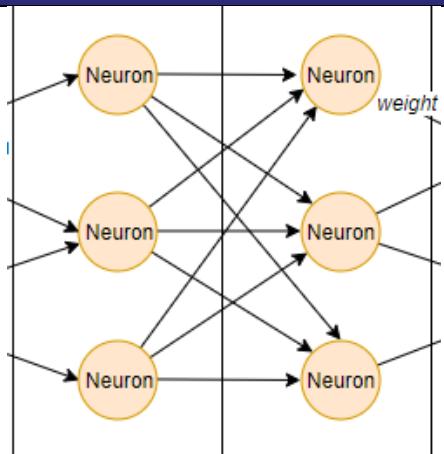
### Capa de entrada (Input layer)



Es la primera capa de la red, la cual recibe los datos de entrada (se pueden cargar desde una fuente externa), a través de sus neuronas realiza los cálculos y la salida se transmite a la capa siguiente.

Cada nodo en esta capa representa una característica o variable de entrada.

### Capa oculta (Hidden layer)

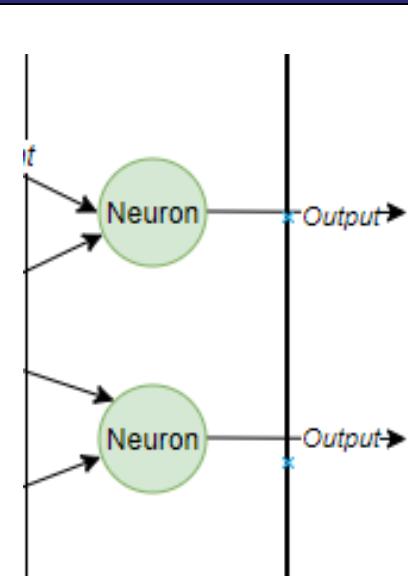


No son visibles ni desde la entrada ni desde la salida.

Se realiza el procesamiento principal de la información y se llevan a cabo los cálculos intermedios.

Esta capa es crucial para la extracción de características y el aprendizaje de patrones.

### Capa de salida (Output layer)



Esta capa final de la red neuronal produce la salida deseada o toma una decisión basada en los cálculos realizados en las capas anteriores. Es la responsable de proporcionar los resultados finales del proceso de aprendizaje.

La capa de salida genera el resultado final del proceso de aprendizaje. Esta capa recibe las entradas provenientes de las capas previas, procesa esta información a través de sus neuronas y luego calcula la salida final.

Siempre debe haber una capa de salida en una red neuronal.

Tabla 31: Descripción detallada de las capas en una Red Neuronal [25]

Cada nodo o neurona, sin importar su ubicación en la red, lleva a cabo algún tipo de operación o proceso sobre la entrada recibida del nodo anterior o de la capa de entrada. Cada nodo opera con una fórmula matemática donde cada variable está ponderada de manera diferente. De manera que determinan la influencia de cada neurona en la salida de las neuronas de la capa siguiente. Si el resultado de esta operación supera cierto umbral, el nodo transmite los datos a la siguiente capa de la red neuronal. Por el contrario, si la salida está por debajo del umbral, no se envía ningún dato a la siguiente capa.

Tal como menciona [26] el proceso de funcionamiento de la red neuronal implica dos fases principales:

- **Fase de entrenamiento** → La red neuronal se expone a un conjunto de datos de entrenamiento, proporcionando entradas junto con las salidas esperadas. Utilizando algoritmos de optimización como el descenso de gradiente, la red neuronal ajusta automáticamente los pesos de las conexiones entre neuronas para minimizar una función de costo, que mide la discrepancia entre las salidas predichas y las esperadas. La red neuronal se considera entrenada correctamente.
- **Fase de inferencia** → Durante esta fase, la red puede realizar predicciones o clasificar nuevos datos sin etiquetar. Las entradas se propagan a través de la red neuronal, atravesando las capas ocultas y generando una salida en la capa de salida. Esta salida se obtiene aplicando funciones de activación a la suma ponderada de las entradas, lo que introduce no linealidad en el modelo y permite a la red neuronal aprender relaciones complejas entre las variables de entrada y salida.

### 3.4.2. Tipos de redes neuronales

Según lo mencionado por [27] cuando se trata de categorizar las redes neuronales, es posible hacerlo utilizando diversos criterios. Entre las formas más comunes de clasificación se encuentran: la organización según la cantidad de capas, la agrupación basada en el patrón de conexiones, la segmentación según la complejidad de las conexiones y la categorización según la estructura arquitectónica.

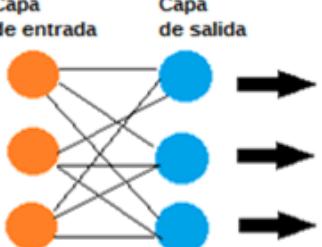
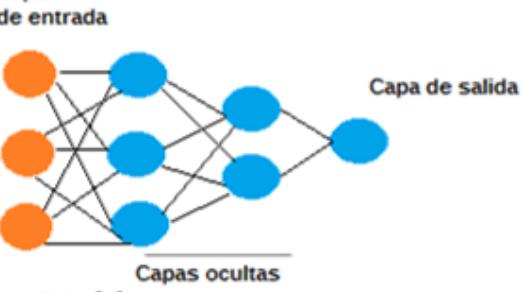
Clasificación por el número de capas	
Redes neuronales monocapas	Redes neuronales multicapas
También conocida como perceptrón simple. Son las más simples, con una capa de entrada y una capa de salida que realiza cálculos.	Tienen capas intermedias, llamadas capas ocultas, que realizan cálculos adicionales entre la entrada y la salida.  Estas capas pueden estar interconectadas entre sí o no.
 <p><b>Monocapa</b></p>	 <p><b>Multicapa</b></p>

Tabla 32: Clasificación redes neuronales por el número de capas [27]

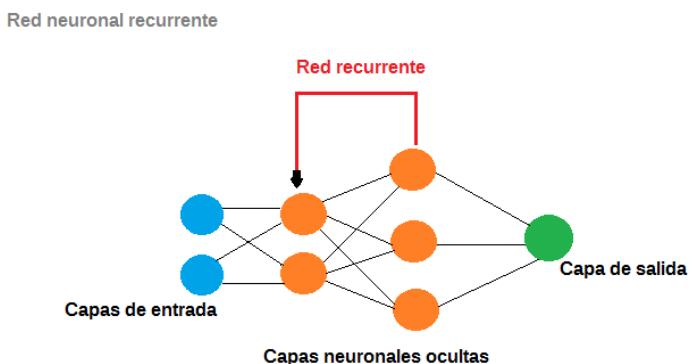
Clasificación por los tipos de conexiones	
Redes neuronales no recurrentes	Redes neuronales recurrentes (RNN)
Surgidas en 1980, operan en una dirección sin realimentación ni memoria, siendo menos comunes.	Al permitir conexiones de retroalimentación entre neuronas, poseen memoria y son más potentes que las no recurrentes. Aunque ambas utilizan algoritmos similares durante el entrenamiento, las recurrentes ofrecen ventajas generales significativas.
 <p><b>Red neuronal recurrente</b></p> <p><b>Red recurrente</b></p> <p><b>Capas de entrada</b></p> <p><b>Capas neuronales ocultas</b></p> <p><b>Capa de salida</b></p>	

Tabla 33: Clasificación redes neuronales por tipo de conexiones [27]

Clasificación por grado de conexiones.	
Redes neuronales totalmente conectadas	Redes parcialmente conectadas
Tienen todas sus neuronas enlazadas entre las capas previas y posteriores	No todas las neuronas tienen conexiones completas entre sí.

Tabla 34: Clasificación redes neuronales por grado de conexiones [27]

Clasificación por el tipo de arquitectura.	
<b>Redes Neuronales Transformers</b>	Como ChatGPT o BERT, son ideales para el procesamiento del lenguaje natural, destacándose en la comprensión contextual, a pesar de su alto consumo de recursos.
<b>Redes Neuronales Recurrentes (RNN)</b>	Comúnmente empleadas en generación de texto y secuencias, siendo reemplazadas por las transformers en muchos casos similares.
<b>Redes Neuronales Convolucionales (CNN)</b>	Se utilizan principalmente en visión por computador, reconocimiento facial y datos bidimensionales, siendo útiles también en conducción autónoma y vigilancia.
<b>Redes Generativas Adversarias (GAN)</b>	Son empleadas para generar contenido como rostros artificiales y dibujos, asegurando calidad mediante una red de evaluación.
<b>Redes Neuronales Siamesas (Siamese NN)</b>	Se usan para comparar similitudes, como en la autenticación de documentos o reconocimiento facial.
<b>Redes Neuronales de Memoria Diferenciable (DNM)</b>	Están diseñadas para aplicaciones que requieren memoria y razonamiento.
<b>Variantes de Redes Neuronales Recurrentes</b>	Hay algunas variedades como LSTM (Long Short-Term Memory) y GRU (Gated Recurrent Unit), que buscan mejorar desventajas de las RNN tradicionales.

Tabla 35: Clasificación redes neuronales por tipo de arquitectura [27]

### 3.5. Redes neuronales convolucionales

Como se ha mencionado anteriormente, las redes neuronales convolucionales (CNN o ConvNet) son una subcategoría dentro de las redes neuronales que han demostrado ser altamente efectivas en la clasificación de imágenes. En base a [28]

estos modelos, compuestos por decenas o incluso cientos de capas, están diseñados para aprender características fundamentales de conjuntos de datos complejos, como imágenes y series temporales.

Su aplicación es especialmente recomendada cuando se cuenta con grandes volúmenes de datos, especialmente en el caso de imágenes. Sin embargo, también pueden adaptarse para el procesamiento de datos de señales o series temporales, siempre y cuando se realice un preprocesamiento adecuado para ajustar los datos a la estructura de la red neuronal.

### 3.5.1. Estructura de las CNN

De acuerdo con lo expuesto por [29] la Red Neuronal Convolutacional (CNN) está compuesta por una parte convolucional y dos partes distintas a diferencia de un modelo clásico de Perceptrón Multicapa (MLP) que tiene solo una etapa de clasificación.

- **Etapa Convolucional:** Su función principal radica en extraer características significativas de cada imagen mediante la aplicación de filtros convolucionales. Este proceso permite comprimir la información inicial de la imagen, generando lo que se conoce como mapas de características. Estos mapas, derivados de la convolución, se combinan en un vector denominado "código CNN".

En una imagen, su efecto es comparable a un proceso de filtrado, tal como se ilustra a continuación:



Figura 9: Esquema del recorrido de la ventana del filtro sobre la imagen.

1. Se establece el tamaño inicial de la ventana de filtrado, ubicada en la esquina superior izquierda de la imagen.
  2. La ventana de filtrado, representativa de la característica buscada, se mueve gradualmente de izquierda a derecha a través de la imagen en pasos predefinidos, hasta que cubre toda su extensión.
  3. En cada sección de la imagen cubierta por la ventana, se realiza un cálculo de convolución. Esto genera un mapa de activación o "feature map", que indica la ubicación de las características relevantes en la imagen. A medida que una característica se encuentra más arriba en el mapa de activación, mayor será la similitud entre la porción de imagen examinada y la característica en cuestión.
- **Etapa de Clasificación:** El código CNN obtenido en la salida de la etapa convolucional se introduce como entrada en una segunda etapa, la cual consiste en capas totalmente conectadas que forman un Perceptrón Multicapa (MLP). En esta fase, el objetivo es utilizar las características relevantes extraídas del código CNN en la etapa convolucional para clasificar la imagen en las categorías pertinentes.

Normalmente, la arquitectura de una Red Neuronal Convolutiva (CNN) sigue una estructura estándar:

1. **Capa de convolución (CONV):** La función principal de esta capa inicial es examinar las imágenes de entrada y reconocer la presencia de un conjunto de características específicas. Al final de esta capa, se obtiene un conjunto de mapas de características que capturan diferentes aspectos de la imagen.
2. **Capa de Pooling (POOL):** Ubicada típicamente entre dos capas de convolución, esta capa reduce el tamaño de los mapas de características mientras preserva sus características más importantes. El Max-pooling y el Average pooling son dos de las técnicas más comunes utilizadas para este propósito.
3. **Capa de activación ReLU (Rectified Linear Units):** Esta capa reemplaza todos los valores negativos de la entrada por ceros, introduciendo así no linealidades en el modelo.
4. **Capa Fully Connected (FC):** Colocadas al final de la CNN, estas capas están conectadas completamente a todas las neuronas de salida. Utilizan una

combinación lineal seguida de una función de activación para clasificar las imágenes de entrada en diferentes categorías.

Al integrar estas capas en la arquitectura de la CNN, se logra un modelo eficiente para el procesamiento de imágenes y la clasificación precisa. Este proceso se repite a lo largo de decenas o cientos de capas, permitiendo que la red capture características complejas de las imágenes.

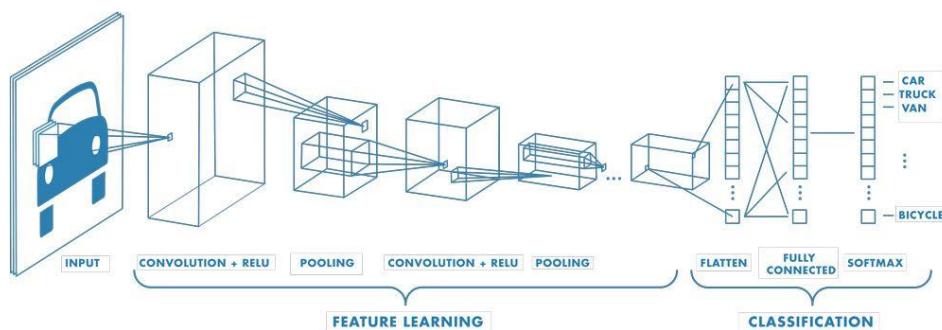


Figura 10: Ejemplo de una red neuronal con múltiples capas convolucionales [28]

Se presenta un ejemplo de una red neuronal con múltiples capas convolucionales. En este enfoque, se emplean filtros para procesar imágenes de entrenamiento a través de distintas resoluciones. La salida de cada capa convolucional se utiliza como entrada para la siguiente capa, permitiendo así una representación jerárquica de las características de las imágenes.

Al contrario de una red neuronal convencional, una CNN emplea pesos y sesgos compartidos entre las neuronas ocultas de una misma capa. Este enfoque implica que todas las neuronas ocultas están sintonizadas para detectar características similares, como bordes o formas, en distintas regiones de la imagen. Esta característica confiere a la red una notable capacidad de generalización y tolerancia a las traslaciones de objetos dentro de una imagen.

### 3.5.2. Aplicación de las CNN para imágenes médicas

Tal como menciona [30] el diagnóstico por imágenes en el ámbito médico implica la creación de representaciones visuales de las estructuras internas del cuerpo con el propósito de análisis clínico y tratamiento médico, así como la visualización de las funciones de órganos y tejidos específicos. Estas imágenes, obtenidas a través de tecnologías como rayos X, tomografía computarizada, mamografía, resonancia

magnética, PET y ultrasonido, son fundamentales para mejorar la precisión diagnóstica.

A pesar de que las imágenes médicas han demostrado ser valiosas para la detección de enfermedades, a menudo presentan desafíos para la interpretación incluso por parte de médicos expertos. La capacidad limitada del ojo humano para detectar detalles sutiles y el fenómeno conocido como 'ceguera por falta de atención', ilustrado por estudios como el realizado por investigadores de la Universidad de Harvard donde se insertó la imagen de un gorila en radiografías, evidencia la importancia de abordar este aspecto en el análisis de imágenes médicas.

Las redes neuronales convolucionales (CNN) han demostrado ser altamente efectivas en el análisis de imágenes médicas. Según investigadores de la Universidad de Stanford, las CNN están especialmente diseñadas para procesar imágenes, lo que las hace ideales para aplicaciones en medicina, como el análisis de resonancias magnéticas o radiografías. Además, estas redes a menudo superan la precisión humana en tareas específicas. Por ejemplo, en el caso de la detección de melanoma por dermatólogos profesionales, la precisión oscila entre el 65% y el 85%. Sin embargo, soluciones basadas en tecnologías como TensorFlow, scikit-learn o keras muestran una precisión más alta, que va desde el 87% hasta el 95%.

Según un estudio realizado por investigadores de la Escuela de Medicina Mount Sinai Icahn, se ha desarrollado una red neuronal que es 150 veces más rápida que los radiólogos humanos en el diagnóstico de afecciones neurológicas críticas como accidentes cerebrovasculares y hemorragias cerebrales.

Un científico español ha creado un algoritmo de aprendizaje profundo basado en redes neuronales para mejorar la resolución de imágenes de resonancia magnética, lo que facilita la identificación de patologías cerebrales complejas, como cáncer, trastornos del habla y lesiones físicas.

Por otro lado, Facebook AI (ahora Meta) y NYU Langone Health han desarrollado una inteligencia artificial llamada fastMRI, que acelera el proceso de escaneo de resonancias magnéticas hasta 4 veces. Los radiólogos no lograron distinguir la diferencia entre las imágenes generadas por fastMRI y las escaneadas de manera tradicional.

### 3.5.3 CNN preentrenadas

Según [31] las CNN preentrenadas son modelos que han sido previamente entrenados para extraer características relevantes de imágenes naturales. Este enfoque aprovecha el conocimiento adquirido por la red neuronal en la tarea de reconocimiento visual a partir de una extensa base de datos, como ImageNet, utilizada en el desafío de Reconocimiento Visual a Gran Escala de ImageNet (ILSVRC). Estas redes han sido entrenadas con millones de imágenes y pueden clasificarlas en una amplia variedad de categorías, desde objetos cotidianos como teclados, tazas de café y lápices, hasta numerosos tipos de animales.

La ventaja principal de utilizar una CNN preentrenada radica en la capacidad de transferir este conocimiento a una nueva tarea con relativa facilidad y rapidez, mediante un proceso conocido como transferencia de aprendizaje.

Las aplicaciones comunes de las redes neuronales preentrenadas incluyen la clasificación de imágenes, la extracción de características relevantes y la transferencia de conocimiento a nuevas tareas.

- **Dense Convolutional Network - Red convolucional densa - DenseNet121**

DenseNet es un tipo de arquitectura de red neuronal convolucional que se caracteriza por tener conexiones densas entre sus capas. Estas conexiones se realizan a través de bloques densos, en los cuales todas las capas están conectadas directamente entre sí, con tamaños de mapa de características coincidentes. Esta estructura permite que cada capa reciba entradas adicionales de todas las capas anteriores y transmita sus propios mapas de características a todas las capas siguientes, manteniendo así la naturaleza de alimentación directa [32] [33]

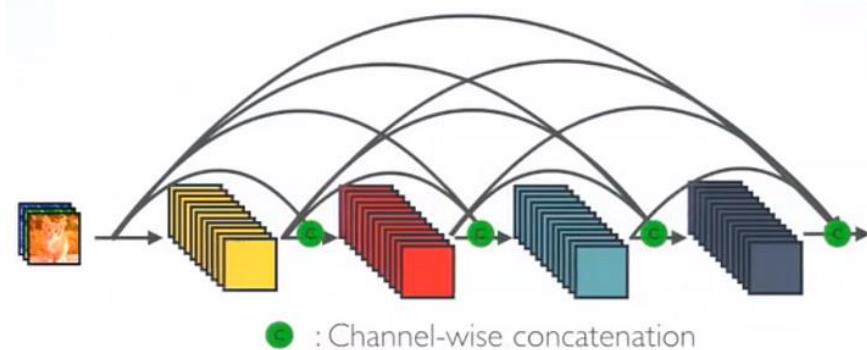


Figura 11: Arquitectura de DenseNet – concatenación [33]

En este proyecto se ha empleado específicamente la arquitectura Densenet121, que es una variante de DenseNet. Esta arquitectura tal como menciona [34] consta de las siguientes capas:

- 1 capa de Convolución de 7x7
- 58 capas de Convolución de 3x3
- 61 capas de Convolución de 1x1
- 4 capas de AvgPool
- 1 capa de Conexión Totalmente Conectada

En resumen, DenseNet-121 tiene 121 convoluciones y 4 AvgPool.

- **MobileNetV2**

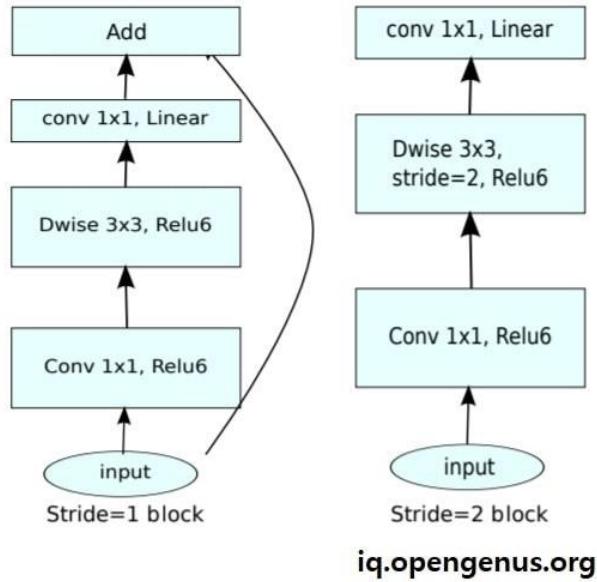
MobileNetV2 [35] representa una evolución en la arquitectura de redes neuronales profundas en comparación con su predecesor, MobileNet original. Integra bloques residuales invertidos y estrategias de estrechamiento, lo que le otorga una eficiencia notable en términos de parámetros. Esta eficiencia la hace particularmente adecuada para despliegues en dispositivos con recursos computacionales limitados.

A diferencia de su versión anterior, MobileNetV2 ofrece la capacidad de procesar imágenes de tamaño variable, admitiendo dimensiones superiores a 32x32. Esta adaptabilidad permite un mejor rendimiento al trabajar con imágenes de alta resolución. Con 53 capas de convolución y una capa de promedio de pool, MobileNetV2 alcanza aproximadamente 350 GFLOP (Operaciones en punto flotante por segundo en gigaflops).

Su diseño se basa en dos bloques principales: el bloque residual invertido y el bloque residual de botella. Estos bloques están compuestos por capas de convolución de 1x1 y 3x3, alternadas para capturar y procesar características a diferentes escalas y niveles de abstracción.

Cada bloque consta de tres capas diferentes:

- una convolución 1x1 con la función de activación Relu6
- una convolución en profundidad
- otra convolución 1x1 sin linealidad.



*Figura 12: Los dos componentes diferentes en el modelo MobileNetV2 [35]*

Hay bloques con un paso de 1 y bloques con un paso de 2. Los componentes internos de estos bloques varían dependiendo del paso, pero siguen la misma estructura general de convoluciones y activaciones.

- **Red Neuronal Residual - ResNet50**

En base a [36] y [37] el modelo ResNet-50 es una red neuronal convolucional diseñada para tareas de reconocimiento de imágenes, presentada en 2015. Se destaca por su profundidad y eficacia al utilizar bloques residuales que permiten el flujo de información sin pérdida a través de las capas, mejorando así el entrenamiento profundo.

La arquitectura ResNet-50 aborda problemas de desvanecimiento de gradientes mediante conexiones de acceso directo y emplea un diseño de "bottleneck" en sus bloques de construcción, lo que acelera el entrenamiento de cada capa. ResNet-50 representa una evolución respecto a la original ResNet-34, con 50 capas ponderadas. Introduce conexiones de acceso directo para evitar problemas de desvanecimiento de gradientes al agregar capas convolucionales.

Presenta elementos clave que incluyen:

- Una convolución inicial de 7x7 seguida de max pooling.
- Diferentes bloques residuales que incluyen convoluciones 1x1, 3x3 y 1x1 con diferentes números de núcleos, repetidos en varias capas.

- Una capa de promedio de valores y una capa completamente conectada con 1000 nodos utilizando la función de activación softmax al final.

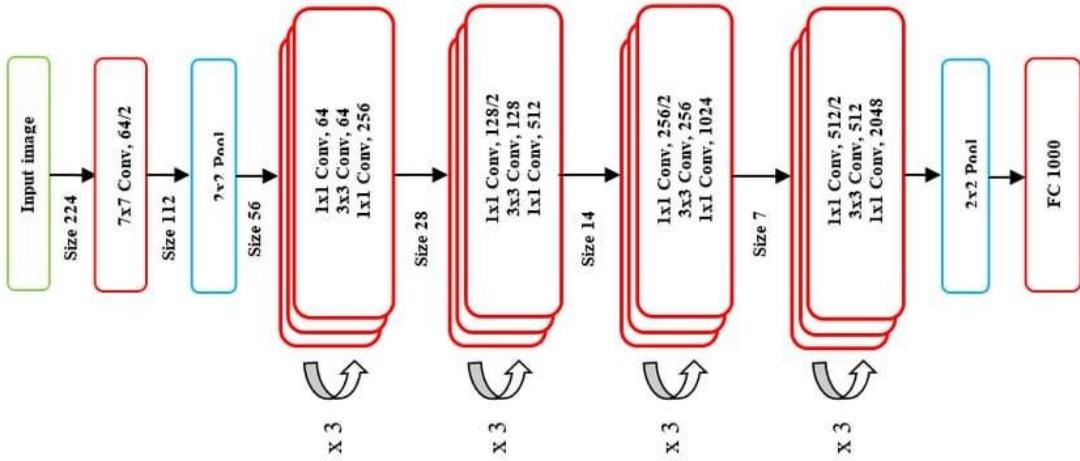


Figura 13: Arquitectura ResNet-50 (Mukherjee 2022) [36]

Esta arquitectura ha demostrado ser altamente efectiva en una variedad de aplicaciones de visión por computadora debido a su capacidad para aprender características complejas de las imágenes.

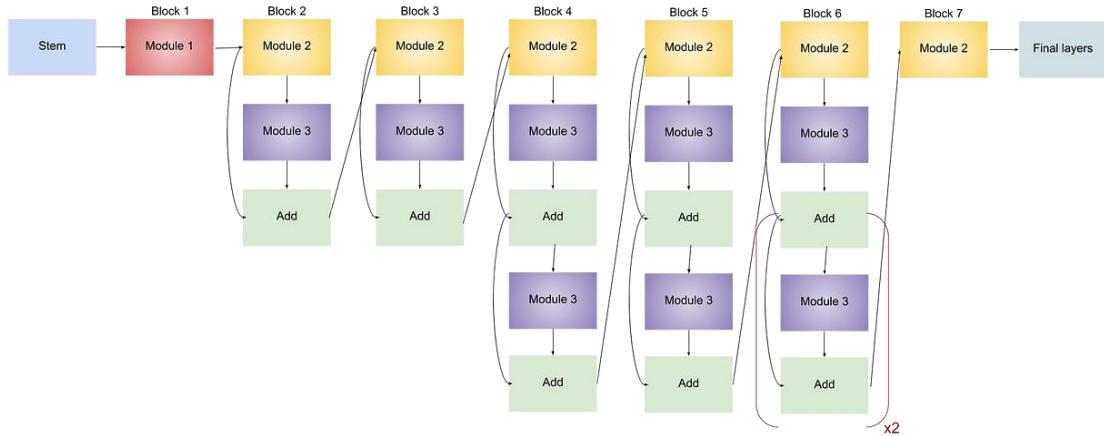
- **EfficientNet-B0**

EfficientNet-B0 [38] la variante base de la familia EfficientNet, se distingue por su eficiencia computacional (FLOPS) para inferencia y su alta precisión en tareas de clasificación de imágenes tanto en ImageNet como en transferencia de aprendizaje de clasificación de imágenes comunes.

Similar a MnasNet, logra un rendimiento cercano al estado del arte con un tamaño de modelo notablemente más pequeño. Introduce un método heurístico para escalar el modelo, ofreciendo una familia de modelos (B0 a B7) que equilibran eficiencia y precisión en diversas escalas. En situaciones de memoria limitada, aumentar la profundidad y/o el ancho del modelo puede mejorar el rendimiento sin comprometer la resolución.

La arquitectura de EfficientNet comienza con una base común a los ocho modelos y las capas finales, donde se llevan a cabo experimentos con la estructura, seguida por

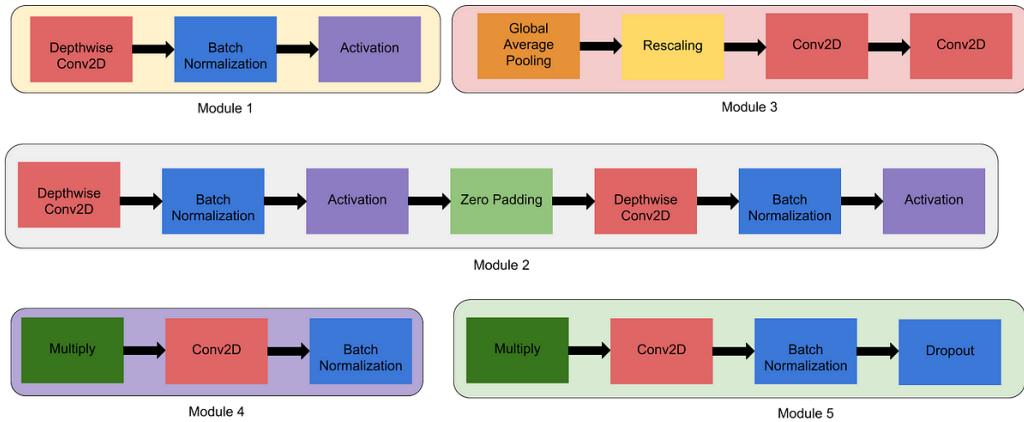
siete bloques, cada uno con una cantidad variable de subbloques. El número de subbloques aumenta gradualmente desde EfficientNetB0 hasta EfficientNetB7.



*Figura 14: Desglose de la Arquitectura de EfficientNet-B0, incluye la repetición de los módulos dentro del corchete dos veces (x2) [38]*

A pesar de que EfficientNet-B0 consta de 237 capas y EfficientNet-B7 de 813, todas estas capas pueden generarse a partir de cinco módulos y una base (tallo inicial). [39] Estos módulos, descritos a continuación, cumplen funciones esenciales en la arquitectura:

- Módulo 1: Cumple la función de punto de partida para los subbloques.
- Módulo 2: Actúa como inicio para el primer subbloque en los siete bloques principales, a excepción del primero.
- Módulo 3: Establece conexiones de omisión con todos los subbloques.
- Módulo 4: Encargado de combinar las conexiones de omisión en los primeros subbloques.
- Módulo 5: Facilita la conexión de omisión entre cada subbloque y su predecesor, permitiendo su combinación.



*Figura 15: Elementos esenciales de la arquitectura EfficientNet: los 5 módulos fundamentales [39]*

Estos módulos se combinan para crear subbloques que desempeñan roles específicos en los bloques. El subbloque 1 se emplea exclusivamente como el primer subbloque del primer bloque. El subbloque 2 se utiliza como el primer subbloque en todos los bloques restantes. Por otro lado, el subbloque 3 se asigna a cualquier subbloque que no sea el primero en todos los bloques.

- **Xception**

Conforme con [40] y [41] Xception es una arquitectura de red neuronal convolucional profunda que introduce innovaciones como las convoluciones separables en profundidad. Fue propuesta por François Chollet, quien también es conocido por crear Keras y trabajar en Google, Inc. Esta arquitectura se inspiró en los conceptos de las redes Inception y lleva esos principios al extremo, tal como sugiere su nombre, que significa "extreme inception".

Al igual que las redes Inception, Xception está diseñada para abordar problemas de eficiencia computacional y de aprendizaje profundo.

Xception invierte el orden de las operaciones: primero aplica filtros en cada mapa de profundidad y luego comprime el espacio de entrada utilizando convoluciones 1x1 a través de la profundidad. Esto es similar a una convolución separable en profundidad, una técnica que se ha utilizado en el diseño de redes neuronales desde hace varios años.

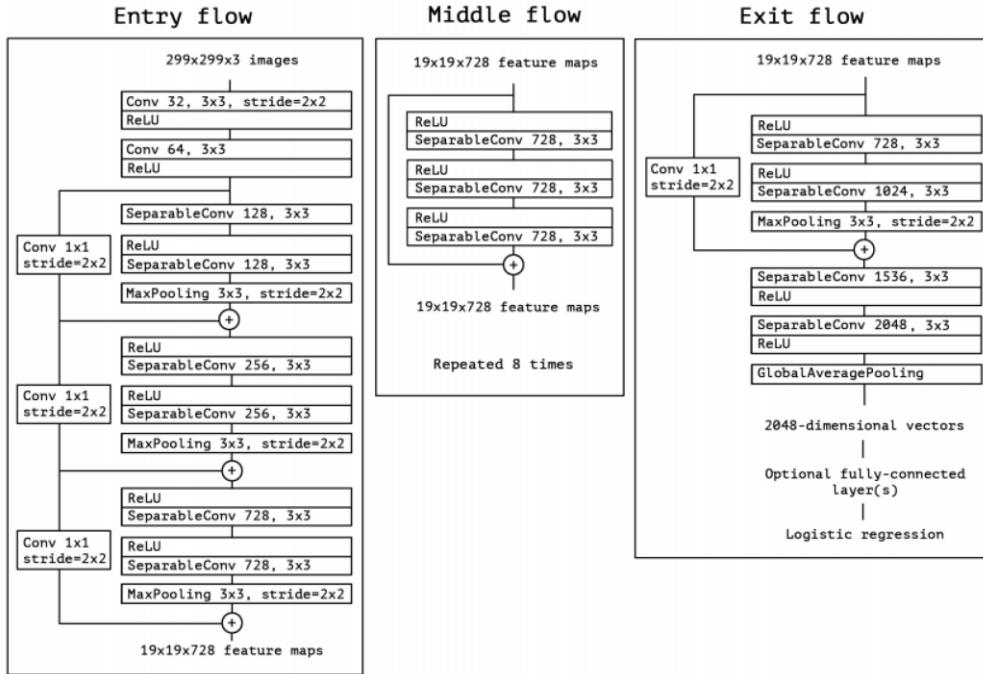


Figura 16: Arquitectura Xception: Flujo de entrada, Flujo medio y Flujo de salida [40]

Los datos inicialmente atraviesan el proceso de entrada, luego se dirigen al flujo intermedio (donde se repiten ocho veces), y finalmente pasan al flujo de salida. Cada capa de Convolución y de Convolución Separable está seguida por normalización por lotes, aunque no se muestre en el diagrama.

Xception fue desarrollado utilizando el framework TensorFlow de Google y fue entrenado utilizando un conjunto de 60 unidades de procesamiento gráfico (GPU) NVIDIA K80.

### 3.6. Variables exógenas

Como se expone en [42] la palabra exógeno, hace referencia a algo que se forma en el exterior de algo. Por lo que se puede concluir que las variables exógenas se forman fuera del modelo, es decir, son factores externos que no son controlables dentro del mismo. Son como las condiciones del entorno que afectan el resultado, pero no son determinadas por las acciones tomadas dentro del sistema en estudio.

La variable opuesta a esta es la variable endógena, la cual influye en los resultados del modelo o sistema y se encuentra dentro de este. Estas pueden ser manipuladas hasta cierto punto dentro del contexto del estudio.

En el proyecto las variables exógenas serían aquellas que no están directamente relacionadas con las características de las imágenes médicas de las enfermedades de la piel, sino que son atributos externos que podrían influir en el diagnóstico o la detección de estas enfermedades.

Variables exógenas	Variables endógenas
<b>Edad (age):</b> La edad del paciente.	<b>Lesión ID (lesion_id):</b> Identificador único de la lesión.
<b>Sexo (sex):</b> El género del paciente.	<b>Image ID (image_id):</b> Identificador único de la imagen médica.
<b>Localización (localization):</b> La parte del cuerpo donde se encuentra la lesión.	<b>Diagnóstico (dx) y Cell Type (cell_type):</b> Tipo de enfermedad de la piel diagnosticada en la imagen.
<b>Tipo de diagnóstico (dx_type):</b> El tipo de diagnóstico realizado.	<b>Path:</b> La ruta del archivo de la imagen médica.

*Tabla 36: Variables exógenas y endógenas del conjunto de datos*

### 3.7. Estado del arte

En esta sección, se analiza el estado actual de la investigación relacionada con la influencia de las variables exógenas en la detección de enfermedades de la piel mediante redes neuronales y análisis de imágenes médicas

Se analizaron estudios recientes que abordan aspectos relevantes para el desarrollo de este trabajo, incluyendo la selección de modelos y el impacto de variables exógenas.

- **Evaluación de modelos de redes neuronales convolucionales para la clasificación de lesiones cutáneas con melanoma [43]**

En el estudio llevado a cabo por Bustinza Salazar (2024), se examinan diversos modelos de redes neuronales convolucionales (CNN) para la clasificación de lesiones cutáneas con melanoma. Se destaca que el método visual tradicional puede conducir a diagnósticos incorrectos, por lo que se propone el empleo de CNN preentrenadas.

Además, se resalta la importancia de seleccionar la arquitectura del modelo adecuada, evidenciando que ciertas arquitecturas, como InceptionV3, pueden ser más

efectivas en condiciones de desbalance de datos. Este estudio destaca la importancia de seleccionar cuidadosamente la arquitectura del modelo (CNN) para optimizar el rendimiento en la detección de enfermedades de la piel.

- **Evaluación del impacto de las variables exógenas en la predicción de pacientes en un servicio de urgencias mediante redes neuronales de atención [44]**

El artículo evalúa el impacto de variables exógenas en la predicción de la afluencia de pacientes en servicios de urgencias utilizando redes neuronales de atención (ScienceDirect). Se encontró que la inclusión de variables como el calendario, el clima y la calidad del aire mejora la precisión de los modelos, mientras que ciertos datos, como los de Google Trends, pueden afectar negativamente el rendimiento.

Este estudio destaca la importancia de seleccionar cuidadosamente las variables exógenas para lograr una precisión óptima en la predicción de la afluencia de pacientes.

## **4. Metodología**

---

### **4.1. Herramientas y tecnologías**

#### **4.1.1. Lenguaje de programación**

Python y R son dos de los idiomas más destacados en el ámbito del aprendizaje automático y el análisis de datos médicos, gracias a su extensa variedad de bibliotecas especializadas tal como menciona [45].

Las ventajas que ofrece son:

- Lenguaje de programación intuitivo y fácil de aprender, lo que facilita el desarrollo de algoritmos complejos para la detección de enfermedades cutáneas, permitiendo expresar ideas de manera clara y concisa.
- Cuenta con una amplia variedad de bibliotecas y frameworks como TensorFlow, Keras y OpenCV, que ofrecen herramientas para el procesamiento de imágenes y la implementación de redes neuronales. Esto no solo reduce el tiempo de programación, sino que también mejora la legibilidad del código.
- Altamente versátil, funcionando sin problemas en diferentes sistemas operativos, lo que facilita la colaboración y el despliegue en diversos entornos.
- Flexibilidad en la integración con otros sistemas y tecnologías, como C y Java , lo que permite una mayor interoperabilidad.
- La capacidad de escalabilidad permite manejar grandes conjuntos de datos y entrenar modelos complejos de manera eficiente.

#### **4.1.2. Entorno de desarrollo**

El entorno de desarrollo elegido para la implementación del sistema es Jupyter Notebook. En un principio, se explora la plataforma de Google Colab como alternativa, sin embargo, se encuentran dificultades relacionadas con la interrupción del proceso de ejecución después de un tiempo determinado, así como problemas de gestión de memoria.

Estos obstáculos llevan a la transición hacia Jupyter Notebook, que ofrece una mayor estabilidad y consistencia en la ejecución del código. Además, Jupyter Notebook proporciona la capacidad necesaria para ejecutar código de manera incremental y mantener el estado de ejecución incluso después de períodos

prolongados, lo que resulta esencial para el proceso de entrenamiento, evaluación y ajuste del modelo de la Red Neuronal Convolucional (CNN) utilizado en el proyecto.

Una de las formas más rápidas y sencillas de instalar Jupyter notebook y la cual se ha llevado a cabo es mediante Anaconda, una distribución libre y gratuita diseñada específicamente para la ciencia de datos y el aprendizaje automático. Con más de 8000 paquetes y bibliotecas de código abierto disponibles. [46]

En el desarrollo de la plataforma web de predicción, se utiliza Visual Studio Code (VS Code), editor de código libre y multiplataforma desarrollado por Microsoft. Ofrece integración con Git, depuración de código y una amplia gama de extensiones para programar en varios lenguajes.

Además de facilitar la escritura y depuración de código, VS Code permite una integración fluida con herramientas de desarrollo como Streamlit utilizada en el proyecto para crear aplicaciones web interactivas con Python. La combinación de estas tecnologías dentro del entorno de VS Code agiliza el proceso de desarrollo y garantiza una experiencia eficiente para el equipo de desarrollo de la plataforma de predicción. [47]

#### **4.1.3. Obtención de dataset**

El conjunto de datos HAM10000 (Human Against Machine with 10000 training images, [2] de imágenes dermatoscópicas se obtuvo del ISIC Archive y está disponible de forma gratuita en Kaggle, además de otras plataformas.

Kaggle es una plataforma web que reúne una amplia comunidad de profesionales y entusiastas de Data Science, ofreciendo una variedad de herramientas y recursos esenciales para el progreso en este campo. Con millones de usuarios registrados en todo el mundo, Kaggle es un destino principal para proyectos de ciencia de datos.

Por otro lado, el ISIC Archive es una base de datos especializada en imágenes dermatoscópicas, lo que lo convierte en una fuente valiosa para la investigación en dermatología y aprendizaje automático aplicado a la salud de la piel.

#### 4.1.4. Librerías

Se utilizaron diversas librerías en Python, incluyendo Pandas para la manipulación de datos, Scikit-learn para la división del conjunto de datos y evaluación de modelos, TensorFlow/Keras para la implementación de la red neuronal convolucional y Streamlit, para convertir scripts de datos en aplicaciones web interactivas.

Librería	Descripción-Aplicación	URL	Icono
Pandas	Utilizada para la manipulación y análisis de datos tabulares. Se emplea para cargar y preprocessar los metadatos del conjunto de datos HAM10000, así como para realizar operaciones de limpieza y transformación de los mismos.	<a href="https://pandas.pydata.org/">https://pandas.pydata.org/</a>	
Numpy	Se utiliza para el manejo de arrays y operaciones matemáticas en Python, así como para la manipulación de las imágenes en formato de arrays y para la normalización de los datos.	<a href="https://numpy.org/">https://numpy.org/</a>	
TensorFlow	Utilizada como el backend principal para la implementación de la red neuronal convolucional. Proporciona una plataforma completa para la construcción y entrenamiento de modelos de aprendizaje automático, incluyendo herramientas para realizar cálculos numéricos eficientes y ejecutar el entrenamiento y la evaluación de modelos.	<a href="https://www.tensorflow.org/?hl=es">https://www.tensorflow.org/?hl=es</a>	
Keras	Utilizada como una interfaz de alto nivel para la construcción y entrenamiento de modelos de redes neuronales en TensorFlow. Keras permite definir y configurar modelos de forma rápida y sencilla, lo que la hace ideal para prototipado rápido y desarrollo.	<a href="https://keras.io/">https://keras.io/</a>	

	experimental. En este trabajo se utiliza para definir la arquitectura del modelo de red neuronal convolucional, así como para compilar, entrenar y evaluar el modelo.		
Matplotlib	Se utiliza para visualizar datos y gráficos, lo que incluye la representación de imágenes y la visualización de resultados de análisis y modelos.	<a href="https://matplotlib.org/">https://matplotlib.org/</a>	
Scikit-learn	Biblioteca de aprendizaje automático que proporciona herramientas simples y eficientes para el análisis de datos y la construcción de modelos predictivos. Se utiliza para dividir el conjunto de datos en conjuntos de entrenamiento, validación y prueba, así como para evaluar la precisión del modelo.	<a href="https://scikit-learn.org/stable/#">https://scikit-learn.org/stable/#</a>	
Scikit-image	Proporciona una colección de algoritmos para el procesamiento de imágenes. Se utiliza para redimensionar las imágenes a un tamaño uniforme antes de alimentarlas al modelo de red neuronal convolucional.	<a href="https://scikit-image.org/">https://scikit-image.org/</a>	
Streamlit	Marco gratuito y de código abierto que convierte scripts de datos en aplicaciones web interactivas en minutos. Con una API intuitiva, agregar elementos es fácil. Despliega y comparte las aplicaciones sin esfuerzo. Todo en Python, sin necesidad de experiencia en front-end.	<a href="https://streamlit.io/">https://streamlit.io/</a>	

Tabla 37: Librerías utilizadas en el trabajo

## 4.2. Análisis de datos

En este apartado, se presenta un análisis del conjunto de datos HAM10000 utilizado en el proyecto.

En la Tabla 38 se describen las variables disponibles junto con una breve explicación de su significado y los valores que pueden tomar. Este análisis proporciona una comprensión inicial de la estructura y la información contenida en el conjunto de datos.

Variables	Descripción	Valores
lesion_id	Identificador único de la lesión.	['HAM_0000118', 'HAM_0002730', 'HAM_0001466' ..... 'HAM_0005579', 'HAM_0001576', 'HAM_0002867']
image_id	Identificador único de la imagen asociada a la lesión.	['ISIC_0027419', 'ISIC_0025030', 'ISIC_0026769' ..... 'ISIC_0033536', 'ISIC_0032854', 'ISIC_0032258']
dx	Representa el diagnóstico de la lesión cutánea.	['bkl', 'nv', 'df', 'mel', 'vasc', 'bcc', 'akiec']
dx_type	Indica el tipo de diagnóstico realizado.	['histo', 'consensus', 'confocal', 'follow_up']
age	La edad de los pacientes en el momento del diagnóstico.	[80., 75., 60., 70., 55., 85., 65., 40., 50., 45., 35., 0., 30., nan 5., 25., 20., 10., 15.]
sex	El sexo de los pacientes.	['male', 'female', 'unknown']
localization	La ubicación en el cuerpo donde se encuentra la lesión cutánea.	['scalp', 'ear', 'face', 'back', 'trunk', 'chest', 'upper extremity', 'abdomen', 'unknown', 'lower extremity', 'genital', 'neck', 'hand', 'foot', 'acral']
path	Ruta del archivo de la imagen.	['D:\\Users\\....\\ISIC_0027419.jpg' ..... 'D:\\Users\\....\\ISIC_0032258.jpg']

cell_type	Tipo celular de la lesión.	['Benign keratosis-like lesions ', 'Melanocytic nevi' , 'Dermatofibroma' , 'Melanoma', 'Vascular lesions', 'Basal cell carcinoma', 'Actinic keratoses']
cell_type_idx	Índice del tipo celular de la lesión codificado.	[2, 4, 3, 5, 6, 1, 0]

Tabla 38:Descripción de las variables junto con los valores únicos

La Figura 17 presenta la distribución de los diferentes tipos celulares (cell\_type) de las lesiones cutáneas en el conjunto de datos. Se observa que los melanocitos nevos representan la mayoría de las lesiones, seguidos por el melanoma y las lesiones similares a queratosis benignas. Otros tipos celulares, como el carcinoma de células basales y las queratosis actínicas, también están presentes, aunque en menor frecuencia.

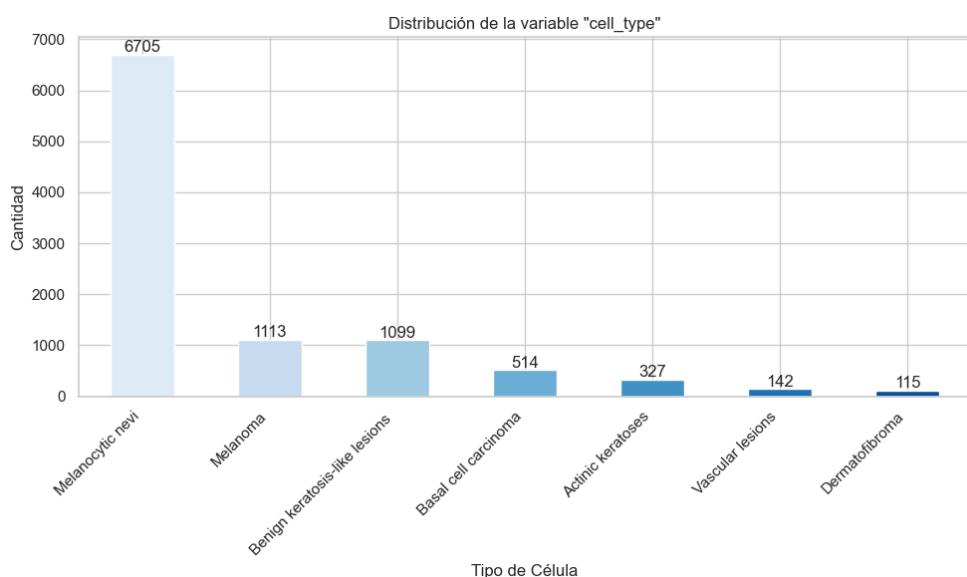
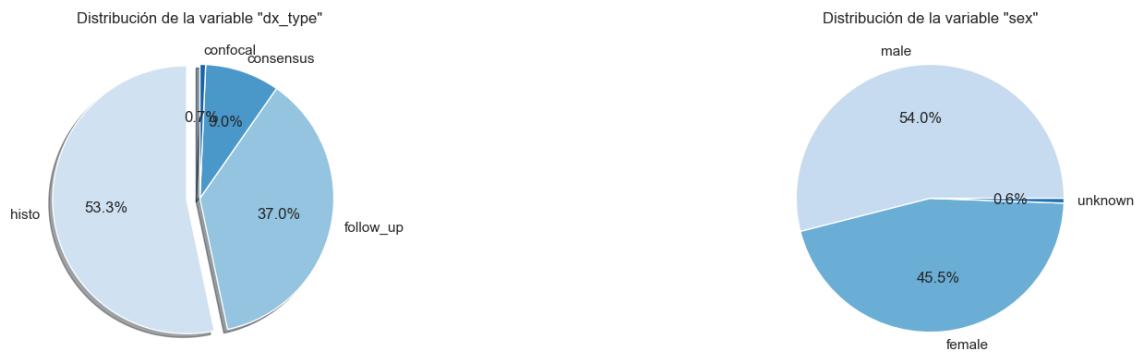


Figura 17: Distribución "cell\_type": Representación en barras

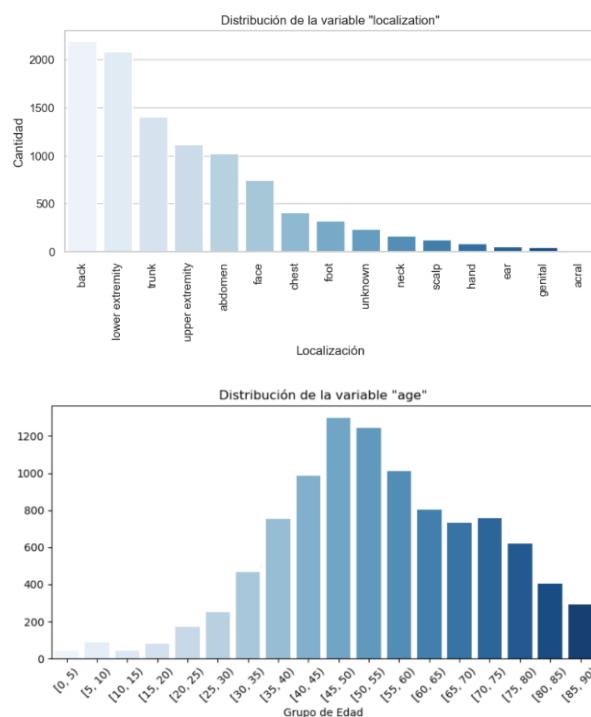
En la Figura 18 se presenta la distribución de los tipos de diagnóstico ("dx\_type") y el sexo de los pacientes en un gráfico de sectores. Se observa que la mayoría de los diagnósticos se realizaron mediante histopatología, seguidos por el seguimiento clínico. En cuanto al sexo de los pacientes, hay una ligera predominancia de pacientes masculinos en comparación con los femeninos.



*Figura 18: Distribución "dx\_type" y "sex": Representación en gráfico de sectores*

La Figura 19 muestra la distribución de las ubicaciones en el cuerpo donde se encuentran las lesiones cutáneas y la edad de los pacientes en un gráfico de barras. Se observa que las ubicaciones más comunes para las lesiones son la espalda, las extremidades inferiores y el tronco.

Además, se puede apreciar cómo se distribuye la edad de los pacientes en relación con estas ubicaciones, destacando que la mayoría se encuentra en el rango de edad entre los 40 y 55 años, con una disminución gradual en las frecuencias a medida que aumenta la edad.

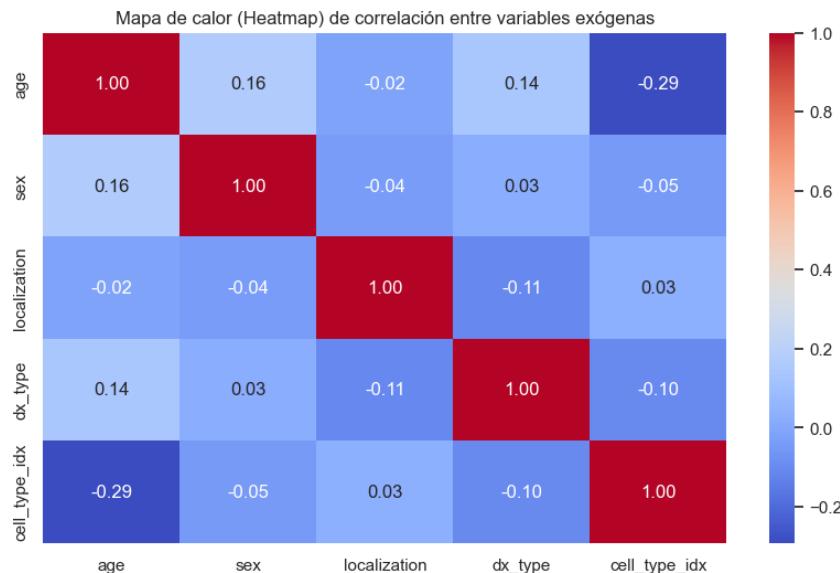


*Figura 19: Distribución "localization" y "age": Representación en barras*

La Figura 20 presenta un mapa de calor que ilustra la matriz de correlación entre las variables analizadas. Este mapa de calor resalta las relaciones lineales entre las variables, proporcionando una visión inicial de cómo cada una se relaciona con la variable objetivo, "cell\_type\_idx".

Se observa que la edad presenta una correlación negativa moderada con el tipo celular de la lesión, sugiriendo que es un factor significativo en su determinación. Le sigue en importancia el tipo de diagnóstico. Por otro lado, el sexo y la ubicación de la lesión muestran una influencia más limitada.

Por lo tanto, las variables "Age" y "Dx\_type" muestran una correlación más fuerte con "Cell\_type\_idx", lo que sugiere que podrían ser más útiles como características para la CNN. En cambio, "Sex" y "Localization", aunque proporcionan información adicional, muestran correlaciones más débiles y podrían ser menos relevantes para la tarea de clasificación de imágenes.



*Figura 20: Mapa de calor (Heatmap) de correlación entre variables exógenas*

La Figura 21 muestra la importancia de las características del modelo de clasificación de tipos celulares de lesiones cutáneas mediante bosques aleatorios. Según el análisis, las características más relevantes son la edad y el tipo de diagnóstico. La localización también desempeña un papel importante, aunque en menor medida, mientras que el sexo del paciente muestra la menor influencia en la clasificación.

Los resultados coinciden con las observaciones anteriores presentadas en la Figura 20, lo que sugiere la posibilidad de considerar el entrenamiento de los modelos utilizando imágenes conjuntamente con las variables más relevantes , con el fin de evaluar si esto conlleva a una mejora en la precisión y eficacia de los sistemas de clasificación.

	Característica	Importancia
0	age	0.376850
3	dx_type	0.344178
2	localization	0.241279
1	sex	0.037693

*Figura 21: Importancia de las características en el modelo de clasificación “cell\_type\_idx” mediante bosques aleatorios (ordenadas por importancia)*

### 4.3. Preprocesamiento del dataset

El preprocesamiento del conjunto de datos incluye varias etapas.

- En primer lugar, se carga el conjunto de datos desde el archivo CSV utilizando la función pd.read\_csv('HAM10000\_metadata.csv') de la biblioteca Pandas. Este archivo contiene metadatos sobre las imágenes de lesiones cutáneas, como la edad del paciente, el tipo de lesión, el género, etc.
- Luego, se realiza el relleno de los valores faltantes en la columna 'age' (edad del paciente) con la media de las edades existentes. Esto se realiza para asegurar que no haya valores faltantes en el conjunto de datos, lo que podría afectar el entrenamiento del modelo. Sin embargo, si el modelo se enfoca únicamente en utilizar las imágenes sin las variables exógenas, este procedimiento no resulta necesario.
- Se define un diccionario lesion\_type\_dict que mapea los códigos de tipo de lesión ('nv', 'mel', 'bkl', etc.) a sus nombres completos ('Melanocytic nevi', 'Melanoma', 'Benign keratosis-like lesions', etc.). Además, se crea un diccionario imageid\_path\_dict que mapea el nombre de la imagen de la lesión a su ruta de archivo correspondiente en el sistema de archivos. Esto se hace utilizando la función glob para encontrar todas las rutas de archivos de imágenes en un directorio específico. Luego, se agrega una nueva columna 'path' al DataFrame que contiene las rutas de archivo de las imágenes correspondientes a cada entrada en el DataFrame. Esto se realiza

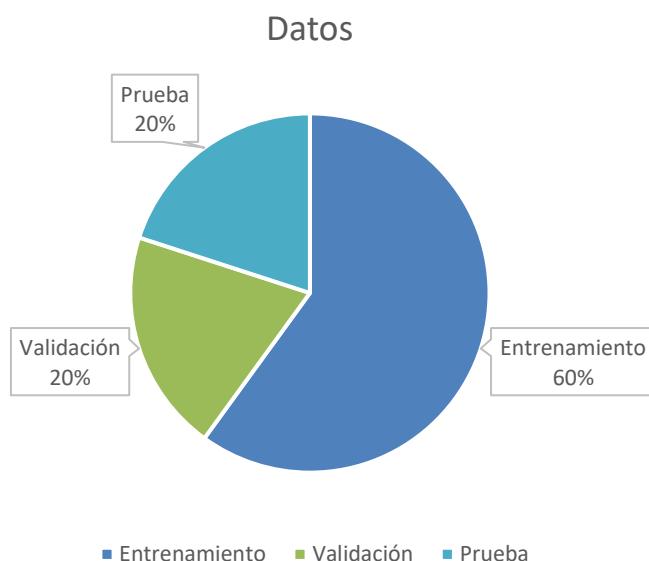
mediante la función map que mapea los identificadores de imagen a las rutas de archivo utilizando el diccionario imageid\_path\_dict.

- Posteriormente, se codifican los tipos de lesiones y otras variables categóricas en el DataFrame utilizando la función pd.Categorical(df['columna']).codes. Esto asigna un código numérico único a cada categoría, lo que facilita el procesamiento posterior del modelo.
- Para cada ruta de archivo en la columna 'path' del DataFrame, se carga la imagen y se convierte en una matriz NumPy de tamaño (64,64). Luego, los valores de píxeles de la imagen se normalizan dividiendo cada elemento por 255.0 para asegurar que estén en el rango [0,1]. Estas imágenes procesadas se agregan a una lista llamada `images`. Finalmente, todas las imágenes se convierten en una matriz NumPy bidimensional llamada `X\_images`.
- Se utilizan las funciones `pd.get\_dummies()` para convertir las variables categóricas del DataFrame en columnas binarias separadas, lo que facilita su manejo en el modelo. Este proceso se realiza específicamente para las variables 'localization' y 'dx\_type'.
- La variable categórica 'gender' (género del paciente) se codifica asignando un valor numérico a cada categoría ('female' se asigna a 0, 'male' a 1 y 'unknown' a 2) para facilitar el procesamiento del modelo.
- Se definen las características que se utilizarán para predecir la variable objetivo ('cell\_type\_idx'). Las características incluyen la variable codificada 'gender\_male', la edad ('age') y las columnas dummy creadas para las variables categóricas 'localization' y 'dx\_type'. Estas características se almacenan en la matriz X para realizar predicciones, mientras que la variable objetivo se guarda en la matriz y, representando el valor a predecir.

Estos tres últimos pasos no son necesarios en el modelo que prescinde de las variables exógenas , este solo convierte las etiquetas en códigos numéricos. Asigna los valores numéricos correspondientes a las etiquetas en la columna 'cell\_type\_idx' del DataFrame a la variable y, lo que la convierte en la variable objetivo del modelo.

- Por otra parte, también se estudiará la precisión de los modelos sin la utilización de las variables exógenas "sex" y "localization". El único cambio con respecto a lo anteriormente mencionado será el mostrado en el Código 5.

- Finalmente, los datos se dividen en conjuntos de entrenamiento, validación y prueba (figura 22) utilizando la función `train_test_split` de `sklearn`. Esto se hace para tener conjuntos de datos separados para entrenamiento, evaluación y prueba del modelo. En este caso, se utiliza un 80% para entrenamiento y un 20% para prueba, y luego se divide el conjunto de entrenamiento en un 75% para entrenamiento y un 25% para validación. Resumidamente, esta distribución se representa como 60% para el entrenamiento, 20% para la validación y 20% para la prueba, tal como se muestra en la Figura 22.



*Figura 22: División de datos para entrenamiento, validación y prueba.*

En los Códigos 1,2,3 y 4 se presenta el código correspondiente al preprocesamiento del conjunto de datos de lesiones cutáneas, el cual ha sido descrito previamente. Estas implementaciones abordan dos enfoques distintos: uno que considera variables exógenas (Códigos 1 y 2) y otro que prescinde de ellas (Códigos 3 y 4).

```

# Cargar el DataFrame
df = pd.read_csv('HAM10000_metadata.csv')

# Rellenar valores faltantes en la columna 'age' con la media
df['age'].fillna(int(df['age'].mean()), inplace=True)

# Definir un diccionario de tipos de lunares
lesion_type_dict = {
    'nv': 'Melanocytic nevi',
    'mel': 'Melanoma',
    'bkl': 'Benign keratosis-like lesions ',
    'bcc': 'Basal cell carcinoma',
    'akiec': 'Actinic keratoses',
    'vasc': 'Vascular lesions',
    'df': 'Dermatofibroma'
}

# Definir el directorio base de las imágenes
base_skin_dir = 'C:\\\\Users\\\\usuario\\\\Downloads\\\\TFG\\\\HAM10000_images_'

# Crear un diccionario de rutas de imagen
imageid_path_dict = {os.path.splitext(os.path.basename(x))[0]: x
                     for x in glob(os.path.join(base_skin_dir, '*', '*.jpg'))}

# Mapear las rutas de imagen en el DataFrame
df['path'] = df['image_id'].map(imageid_path_dict.get)
# Mapear los tipos de lesiones en el DataFrame
df['cell_type'] = df['dx'].map(lesion_type_dict.get)
# Codificar los tipos de Lesiones
df['cell_type_idx'] = pd.Categorical(df['cell_type']).codes

```

```

# Cargar imágenes, convertirlas a matrices numpy
images = []
for path in tqdm(df['path']):
    image = img_to_array(load_img(path, target_size=(64,64)))
    image = image / 255.0
    images.append(image)

X_images = np.array(images)

```

100% [██████████] 10015/10015 [01:28<00:00, 112.96it/s]

```

# Crear columnas dummy para las variables categóricas
df = pd.get_dummies(df, columns=['localization'], prefix='loc')
df = pd.get_dummies(df, columns=['dx_type'], prefix='dx_type')

# Codificar la variable categórica 'gender'
df['gender_male'] = df.sex.map({'female': 0, 'male': 1, 'unknown': 2})

# Seleccionar las características y la variable objetivo
features = ['gender_male', 'age'] +
           [col for col in df.columns if 'loc_' in col] +
           [col for col in df.columns if 'dx_type_' in col]
X = df[features].values
y = df['cell_type_idx'].values

```

Código 1: Implementación preprocessamiento del modelo con variables exógenas I

```

# Dividir los datos en conjuntos de entrenamiento, validación y prueba (20% - 80%(75%-25%))
X_train_images,X_test_images,X_train_tabular,X_test_tabular,y_train,y_test=train_test_split(X_images,X,y,test_size=0.2, random_state=0)
X_train_images,X_val_images,X_train_tabular,X_val_tabular,y_train,y_val=train_test_split(X_train_images,X_train_tabular,y_train, test_size=0.25, random_state=0)

# Normalizar datos tabulares
scaler = StandardScaler()
X_train_tabular = scaler.fit_transform(X_train_tabular)
X_val_tabular = scaler.transform(X_val_tabular)
X_test_tabular = scaler.transform(X_test_tabular)

```

Código 2: Implementación preprocessamiento del modelo con variables exógenas II

```

# Cargar el DataFrame
df = pd.read_csv('HAM10000_metadata.csv')

# Definir un diccionario de tipos de lunares
lesion_type_dict = {
    'nv': 'Melanocytic nevi',
    'mel': 'Melanoma',
    'bkl': 'Benign keratosis-like lesions ',
    'bcc': 'Basal cell carcinoma',
    'akiec': 'Actinic keratoses',
    'vasc': 'Vascular lesions',
    'df': 'Dermatofibroma'
}

# Definir el directorio base de las imágenes
base_skin_dir = 'D:\\\\Users\\\\paula\\\\Escritorio\\\\TFG\\\\DATASET-TFG\\\\HAM10000_images_'

# Crea un diccionario de rutas de imagen
imageid_path_dict = {os.path.splitext(os.path.basename(x))[0]: x
                     for x in glob(os.path.join(base_skin_dir, '*', '*.jpg'))}

# Mapea las rutas de imagen en el DataFrame
df['path'] = df['image_id'].map(imageid_path_dict.get)
# Mapea los tipos de lesiones en el DataFrame
df['cell_type'] = df['dx'].map(lesion_type_dict.get)
# Codifica los tipos de lesiones
df['cell_type_idx'] = pd.Categorical(df['cell_type']).codes

# Cargar imágenes, convertirlas a matrices numpy
images = []
for path in tqdm(df['path']):
    image = img_to_array(load_img(path, target_size=(64,64)))
    image = image / 255.0 # Normaliza la imagen
    images.append(image)

X_images = np.array(images)

```

100% |██████████| 10015/10015 [01:32< 00:00, 108.23it/s]

Código 3: Implementación preprocessamiento del modelo sin variables exógenas I

```

# Convertir las etiquetas a códigos numéricos
y = df['cell_type_idx'].values

# Dividir los datos en conjuntos de entrenamiento, validación y prueba
X_train_images, X_test_images, y_train, y_test = train_test_split(X_images, y, test_size=0.2, random_state=0)
X_train_images, X_val_images, y_train, y_val = train_test_split(X_train_images, y_train, test_size=0.25, random_state=0)

```

Código 4: Implementación preprocessamiento del modelo sin variables exógenas II

```

# Crea columnas dummy para las variables categóricas
df = pd.get_dummies(df, columns=['dx_type'], prefix='dx_type')

# Selecciona las características y la variable objetivo (la cual se desea predecir)
features = ['age'] + [col for col in df.columns if 'dx_type_' in col]
X = df[features].values
y = df['cell_type_idx'].values

```

Código 5: Implementación preprocessamiento del modelo solo con las variables exógenas "age" y "dx\_type"

#### **4.4. Implementación de la red neuronal convolucional**

En la implementación de la red neuronal convolucional (CNN), se han empleado diversas arquitecturas preentrenadas, como DenseNet121, MobileNetV2, ResNet50, EfficientNetB0 y Xception, junto con una red creada desde cero como parte del conjunto de arquitecturas empleadas.

Para ambos modelos se crea una entrada de imagen con forma (64,64,3) utilizando la función 'Input', que define el formato de las imágenes que el modelo recibirá. Luego, se agrega una capa global de pooling (Flatten) para convertir la salida de las capas convolucionales en un vector unidimensional, seguida de una capa densa de salida con activación softmax. La activación softmax se utiliza en la capa de salida para realizar una clasificación multclase, donde cada clase representa una de las posibles categorías de salida.

Se compila el modelo utilizando el optimizador Adam y se utiliza la función de pérdida de entropía cruzada categórica escasa (sparse categorical crossentropy) como medida de la discrepancia entre las distribuciones de probabilidad predichas por el modelo y las distribuciones reales de las etiquetas. Esta función de pérdida es adecuada para problemas de clasificación multclase.

El modelo se entrena utilizando los datos de entrenamiento y validación durante 10 épocas utilizando un tamaño de lote (batch size) de 32. Una vez que el modelo está entrenado, se evalúa el modelo en el conjunto de prueba y se muestran los resultados de la pérdida y la precisión.

Parámetros	Valor
Arquitecturas preentrenadas	DenseNet121, MobileNetV2, ResNet50, EfficientNetB0, Xception
Input size	(64,64,3)
Capa global de pooling	Flatten
Capa densa de salida	Activación: Softmax
Optimizador	Adam
Función de pérdida	Sparse Categorical Crossentropy
Épocas de entrenamiento	10
Tamaño de lote (batch size)	32

*Tabla 39: Resumen parámetros utilizados en la implementación de la CNN*

Para los modelos que consideran las variables exógenas se define entradas separadas para las imágenes y las características tabulares (variables exógenas) y se concatenan las salidas de estos dos tipos de datos antes de pasarlas a la capa de salida final.

En el Código 6 se presenta la implementación del modelo CNN, específicamente para ResNet50. Se emplea el mismo código para todas las CNN preentrenadas mencionadas anteriormente, variando únicamente cuando se cambia el modelo base. Es decir, se en "base\_model = ResNet50(weights='imagenet', include\_top=False, input\_tensor=image\_input)" en lugar de "ResNet50", se reemplaza con el modelo deseado.

```
# Definir la entrada para la imagen
image_input = Input(shape=(64, 64, 3), name='input_image')

# Cargar el modelo base sin la capa densa final
base_model = ResNet50(weights='imagenet', include_top=False, input_tensor=image_input)

# Agregar una capa global de pooling y una capa densa de salida
x = base_model.output
x = Flatten()(x)

# Define la entrada para las características tabulares fuera del bucle
tabular_input = Input(shape=(X_train_tabular.shape[1]), name='tabular_input')

# Concatena las salidas de los modelos de imágenes y tabulares
combined = concatenate([x, tabular_input])

# Agrega una capa de salida final
output = Dense(7, activation='softmax', name='output')(combined)

# Crea el modelo combinado fuera del bucle
model = Model(inputs=[image_input, tabular_input], outputs=output)

# Compila el modelo
model.compile(optimizer=Adam(), loss='sparse_categorical_crossentropy', metrics=['accuracy'])

# Entrenar el modelo
print("\nEntrenamiento del modelo:")
model.fit([X_train_images, X_train_tabular], y_train, validation_data=[[X_val_images, X_val_tabular], y_val], epochs=10, batch_size=32)

# Evaluar el modelo
print("\nEvaluación del modelo en conjunto de prueba:")
test_loss, test_accuracy = model.evaluate([X_test_images, X_test_tabular], y_test)
print(f'Test Loss: {test_loss}, Test Accuracy: {test_accuracy}'')
```

Código 6: Implementación CNN con variables exógenas, en concreto ResNet50

Como se puede apreciar en el Código 7, en la implementación para los modelos que no consideran las variables exógenas, solamente se define una entrada para las imágenes, dado que no se están considerando características tabulares adicionales.

```

# Definir la entrada para la imagen
image_input = Input(shape=(64, 64, 3), name='input_image')

# Cargar el modelo base sin la capa densa final
base_model = ResNet50(weights='imagenet', include_top=False, input_tensor=image_input)

# Agregar una capa global de pooling y una capa densa de salida
x = base_model.output
x = Flatten()(x)

# Agregar una capa de salida final
output = Dense(7, activation='softmax', name='output')(x)

# Crear el modelo combinado fuera del bucle
model = Model(inputs=image_input, outputs=output)

# Compilar el modelo
model.compile(optimizer=Adam(), loss='sparse_categorical_crossentropy', metrics=['accuracy'])

# Entrenar el modelo
print("\nEntrenamiento del modelo:")
model.fit(X_train_images, y_train, validation_data=(X_val_images, y_val), epochs=10, batch_size=32)

# Evaluar el modelo
print("\nEvaluación del modelo en conjunto de prueba:")
test_loss, test_accuracy = model.evaluate(X_test_images, y_test)
print(f'Test Loss: {test_loss}, Test Accuracy: {test_accuracy}')

```

Código 7: Implementación CNN sin variables exógenas, en concreto ResNet50

## 4.5. Algoritmos de hiperparametrización

La Tabla 40 presenta una comparación detallada de los hiperparámetros utilizados en el entrenamiento de modelos seleccionados para la clasificación de lesiones cutáneas con melanoma. Se distinguen entre modelos preentrenados, incluidos DenseNet121, MobileNetV2, ResNet50, EfficientNetB0 y Xception, y aquel construido desde cero.

Hiperparámetros	Modelos preentrenados	CNN desde cero
Numero de épocas	10	10
Numero de capas intermedias	1 (capa densa después de la capa base)	2 capas intermedias en la red neuronal
Numero de neuronas capa 1	256 neuronas	128 neuronas
Número de neurona capa 2	No aplica, solo hay una capa intermedia	64 neuronas
Función de activación en capas intermedias	ReLU (Rectified Linear Unit)	ReLU (Rectified Linear Unit)
Tasa de aprendizaje	Por defecto, 0.001	Por defecto, 0.001

Optimizador	Adam	Adam
Tamaño de lote	32	32
Función de pérdida	Sparse Categorical Crossentropy	Sparse Categorical Crossentropy
Tamaño de imágenes de entrada	64x64 píxeles	64x64 píxeles

*Tabla 40: Comparación de hiperparámetros entre modelos preentrenados y CNN desde cero [43]*

## 4.6. Métricas

Una vez se ha entrenado los modelos, se utilizan varias métricas para evaluar el rendimiento de los modelos de clasificación en la tarea de clasificación de lesiones cutáneas. A continuación, se explicarán de acuerdo con [48] y [49]

### 4.6.1. Matriz de confusión

La matriz de confusión es una herramienta fundamental en la evaluación de modelos de clasificación, ya que proporciona una visión detallada de cómo el sistema está clasificando las muestras. Se compone de un cuadro que muestra el número de verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos. Se caracteriza por ser una matriz cuadrada, con el número de filas y columnas igual al número de clases presentes en el problema de clasificación.

El escenario ideal que se busca es aquel en el que el modelo arroje 0 falsos positivos (FP) y 0 falsos negativos (FN), lo que indicaría una clasificación perfecta. Sin embargo, en la práctica, esto puede ser difícil de lograr y, por lo tanto, se busca minimizar tanto los falsos positivos como los falsos negativos para mejorar la precisión del modelo de clasificación.

En la Tabla 41 se presenta una descripción detallada de los elementos de evaluación, tales como TP, FP, FN y TN, acompañados de representaciones gráficas para una mejor comprensión.

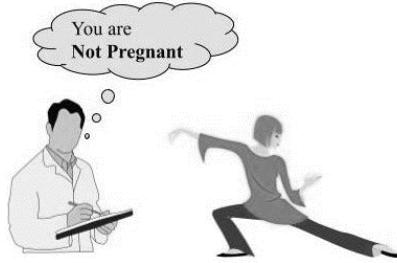
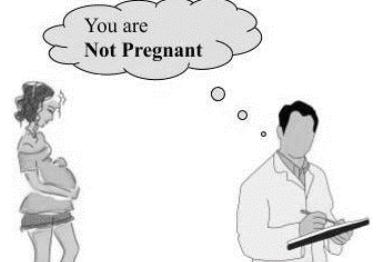
Descripción de los elementos	Ejemplo ilustración
<p><b>Verdaderos positivos (True positives -TP):</b> Ocurren cuando la clase real de un punto de datos es 1 (verdadero) y la predicción del modelo también es 1 (verdadero), representando los casos positivos correctamente clasificados por el modelo.</p>	 <p>A doctor in a white coat is writing on a clipboard. A pregnant woman is standing next to him. A thought bubble above them says "You are Pregnant". Below the illustration is the text "TRUE POSITIVE".</p>
<p><b>Verdaderos negativos (True negatives -TN):</b> Se dan cuando la clase real de un punto de datos es 0 (falso) y la predicción del modelo también es 0 (falso), indicando los casos negativos correctamente identificados por el modelo.</p>	 <p>A doctor in a white coat is writing on a clipboard. A non-pregnant woman is performing a stretching exercise. A thought bubble above them says "You are Not Pregnant". Below the illustration is the text "TRUE NEGATIVE".</p>
<p><b>Falsos positivos (False positives -FP):</b> Surgen cuando la clase real de un punto de datos es 0 (falso) pero la predicción del modelo es 1 (verdadero), reflejando los casos negativos incorrectamente clasificados como positivos por el modelo.</p>	 <p>A doctor in a white coat is writing on a clipboard. A non-pregnant woman is sitting on the floor. A thought bubble above them says "You are Pregnant!". Below the illustration is the text "FALSE POSITIVE".</p>
<p><b>Falsos negativos (False negatives - FN):</b> Ocurren cuando la clase real de un punto de datos es 1 (verdadero) pero la predicción del modelo es 0 (falso), representando los casos positivos incorrectamente clasificados como negativos por el modelo.</p>	 <p>A doctor in a white coat is writing on a clipboard. A pregnant woman is standing. A thought bubble above her says "You are Not Pregnant". Below the illustration is the text "FALSE NEGATIVE".</p>

Tabla 41: Descripción de los elementos de evaluación con representaciones gráficas [50]

En la Figura 23 se presenta los elementos clave en una matriz de confusión, donde se identifican “a” como los Verdaderos Negativos (TN), “b” como los Falsos Positivos (FP), “c” como los Falsos Negativos (FN), y “d” como los Verdaderos Positivos (TP).

Matriz de confusión		Estimado por el modelo	
		Negativo (N)	Positivo (P)
Real	Negativo	a: (TN)	b: (FP)
	Positivo	c: (FN)	d: (TP)

Figura 23: Matriz de confusión (De los Santos 2023)

Además de proporcionar una visión detallada de la clasificación del sistema, la matriz de confusión sirve como base para calcular métricas esenciales como la precisión, la sensibilidad y la especificidad. Permite visualizar el desempeño del modelo en diferentes clases y revelar patrones de error en la clasificación. Prácticamente todas las métricas de evaluación del modelo se basan en los datos extraídos de esta matriz.

#### 4.6.2. Exactitud

La exactitud, también conocida como "accuracy" en inglés, es una métrica crucial en la evaluación de modelos de clasificación. Representa la proporción de casos correctamente clasificados respecto al total de casos evaluados. Su cálculo se basa en la fórmula:  $\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)}$ . Estos conceptos se muestran en la Figura 24.

Matriz de confusión		Estimado por el modelo		
		Negativo (N)	Positivo (P)	
Real	Negativo	a: (TN)	b: (FP)	
	Positivo	c: (FN)	d: (TP)	
		Exactitud ("accuracy") Porcentaje de predicciones correctas (No sirve en datasets poco equilibrados)		
		$(a+d)/(a+b+c+d)$		

Figura 24: Matriz de confusión y cálculo de exactitud (accuracy) [51]

La exactitud ofrece una visión general del desempeño del modelo en la tarea de clasificación. Una alta exactitud sugiere una buena precisión en las clasificaciones, mientras que una baja exactitud indica errores en la clasificación. Es importante tener en cuenta que la exactitud puede ser engañosa en situaciones de desequilibrio de datos, especialmente con clases minoritarias.

En estos casos, un modelo que predice la clase mayoritaria puede mostrar una alta exactitud pero ser poco útil en la práctica. Por ello, se recomienda complementar la evaluación de la exactitud con otras métricas para obtener una visión más completa del rendimiento del modelo.

#### 4.6.3. Precisión

La precisión evalúa la proporción de casos positivos correctamente identificados por un modelo en relación con el total de casos identificados como positivos. Se calcula:

$$Precision = \frac{TP}{(TP + FP)}$$

Este indicador resulta fundamental en situaciones donde los costos asociados con los falsos positivos son significativos.

No obstante, la precisión no aborda los falsos negativos, es decir, los casos positivos que el modelo no identifica correctamente. Por lo tanto, se recomienda complementar la evaluación con otras métricas para obtener una visión más completa del desempeño del modelo en tareas de clasificación.

Como se ilustra en la Figura 25, se presenta la matriz de confusión y el cálculo de precisión en la evaluación del modelo de clasificación.

Matriz de confusión		Estimado por el modelo			
		Negativo (N)	Positivo (P)		
Real	Negativo	a: (TN)	b: (FP)	Precisión ("precision") Porcentaje predicciones positivas correctas:	d/(b+d)
	Positivo	c: (FN)	d: (TP)		

Figura 25: Matriz de confusión y cálculo de precisión [51]

#### 4.6.4. Recall

La sensibilidad (recall en inglés) también conocida como tasa de true positive (TPR), es una métrica esencial en la evaluación de modelos de clasificación. Se calcula :  $Recall = \frac{TP}{(TP+FN)}$ . En otras palabras, mide la capacidad del modelo para identificar correctamente los casos positivos en relación con el total de casos positivos en los datos. Estos conceptos se muestran en la Figura 26.

La sensibilidad es crucial en situaciones donde identificar correctamente todos los casos positivos es fundamental debido al impacto significativo de un falso negativo. Sin embargo, no aborda los falsos positivos, por lo que se recomienda combinarla con otras métricas para una evaluación completa del modelo en clasificación.

Matriz de confusión		Estimado por el modelo	
Real	Negativo	Negativo (N)	Positivo (P)
	Positivo	c: (FN)	d: (TP)
		Sensibilidad, exhaustividad ("Recall") Porcentaje casos positivos detectados  d/(d+c)	

Figura 26:Matriz de confusión y cálculo de sensibilidad (Recall) [51]

#### 4.6.5. F1-score

El F1 Score es una métrica ampliamente empleada en problemas de clasificación, combinando la precisión y la sensibilidad en una única medida.

Calculado como la media armónica de la precisión y la sensibilidad, el F1 Score busca un equilibrio entre ambas métricas. Se define como el doble del producto de precisión y sensibilidad, dividido por la suma de precisión y sensibilidad, como se observa en la Figura 27.

$$\begin{aligned} \text{F1 Score} &= \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} \\ &= \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \end{aligned}$$

Figura 27: Fórmula del F1-score [52]

Con valores que van de 0 a 1, donde 1 es el óptimo, un alto F1 Score señala un equilibrio efectivo entre precisión y sensibilidad, reflejando la capacidad del modelo para identificar tanto casos positivos como negativos con precisión. Particularmente útil en situaciones de desequilibrio de datos, donde hay más casos negativos o positivos, el F1 Score ofrece una evaluación más precisa del desempeño del modelo en la clasificación.

Una vez que se han explicado las métricas, en los Códigos 8 y 9 se presenta el código utilizado en el proyecto para su implementación. En el Código 8, se observa que para los modelos que incorporan variables exógenas, se suministran tanto imágenes (`X_test_images`) como variables exógenas (`X_test_tabular`) como entrada al modelo. Por otro lado, en el Código 9, destinada a los modelos que no requieren variables exógenas, solo se proporcionan imágenes (`X_test_images`) como entrada.

```
# Predecir las etiquetas para el conjunto de prueba
y_pred = model.predict([X_test_images, X_test_tabular])
y_pred_classes = np.argmax(y_pred, axis=1)

# Calcular métricas
accuracy = accuracy_score(y_test, y_pred_classes)
precision = precision_score(y_test, y_pred_classes, average='weighted')
recall = recall_score(y_test, y_pred_classes, average='weighted')
f1 = f1_score(y_test, y_pred_classes, average='weighted')
conf_matrix = confusion_matrix(y_test, y_pred_classes)

# Imprimir métricas
print(f'Accuracy: {accuracy}')
print(f'Precision: {precision}')
print(f'Recall: {recall}')
print(f'F1 Score: {f1}')
print('Confusion Matrix:')
print(conf_matrix)
```

Código 8: Implementación de métricas para evaluar el rendimiento de los modelos con variables exógenas

```

# Predecir las etiquetas para el conjunto de prueba
y_pred = model.predict(X_test_images)
y_pred_classes = np.argmax(y_pred, axis=1)

# Calcular métricas
accuracy = accuracy_score(y_test, y_pred_classes)
precision = precision_score(y_test, y_pred_classes, average='weighted')
recall = recall_score(y_test, y_pred_classes, average='weighted')
f1 = f1_score(y_test, y_pred_classes, average='weighted')
conf_matrix = confusion_matrix(y_test, y_pred_classes)

# Imprimir métricas
print(f'Accuracy: {accuracy}')
print(f'Precision: {precision}')
print(f'Recall: {recall}')
print(f'F1 Score: {f1}')
print('Confusion Matrix:')
print(conf_matrix)

```

*Código 9: Implementación de métricas para evaluar el rendimiento de los modelos sin variables exógenas*

Se realizaron las mismas métricas de evaluación del rendimiento del modelo por clase, es decir, en la clasificación de 7 diferentes clases de lesiones cutáneas, como se muestra en el Código 10.

```

# Calcular precisión, recall y F1 por clase
precision, recall, f1, _ = precision_recall_fscore_support(y_test, y_pred_classes, average=None)

# Crear un DataFrame para mostrar los resultados
results_df = pd.DataFrame({
    'Clases': df['cell_type'].unique(),
    'Precisión': precision,
    'Recall': recall,
    'Score - F1': f1})

# Imprimir el DataFrame
print(results_df)

```

*Código 10: Implementación de métricas para evaluar el rendimiento de los modelos por clase*

Para evaluar la capacidad de clasificación del modelo en las diferentes clases de lesiones cutáneas, se generó una matriz de confusión, como se muestra en el Código 11. Esta matriz proporciona una representación visual de las predicciones del modelo en comparación con las etiquetas verdaderas para cada clase.

```
# Calcular la matriz de confusión
conf_matrix = confusion_matrix(y_test, y_pred_classes)

# Crear un DataFrame para mostrar la matriz de confusión
conf_matrix_df = pd.DataFrame(conf_matrix, index=df['cell_type'].unique(), columns=df['cell_type'].unique())

# Visualizar la matriz de confusión utilizando seaborn
plt.figure(figsize=(10, 8))
sns.heatmap(conf_matrix_df, annot=True, cmap='Blues', fmt='d')
plt.xlabel('Predicción')
plt.ylabel('Etiqueta verdadera')
plt.title('Matriz de confusión')
plt.show()
```

Código 11: Implementación para la visualización de la matriz de confusión por clase

Como se muestra en el Código 12, se presentaron ejemplos de imágenes clasificadas incorrectamente por el modelo, junto con sus etiquetas predichas y verdaderas.

```
# Obtener las etiquetas de lesiones cutáneas correspondientes a las predicciones y etiquetas reales
predicted_labels = [str(label) for label in y_pred_classes]
true_labels = [str(label) for label in y_test]
misclassified_indices = np.where(y_pred_classes != y_test)[0]

# Mostrar algunas imágenes mal clasificadas junto con las predicciones
plt.figure(figsize=(12, 8))
for i, idx in enumerate(misclassified_indices[:9]):
    plt.subplot(3, 3, i + 1)
    plt.imshow(X_test_images[idx])
    predicted_label = predicted_labels[i]
    true_label = true_labels[i]
    plt.title(f'Predicción: {predicted_label}\nEtiqueta: {true_label}')
    plt.axis('off')

# Ajustar espacios entre las subtramas
plt.subplots_adjust(hspace=0.5)

plt.show()
```

Código 12: Implementación: Ejemplo de imágenes clasificada

## **5. RESULTADOS**

---

### **5.1. Recopilación de datos**

El conjunto de datos HAM10000 ("Human Against Machine with 10000 training images") ha recopilado 10,015 imágenes dermatoscópicas de diversas poblaciones, adquiridas y almacenadas mediante diferentes modalidades durante un periodo de 20 años en dos sitios diferentes: el Departamento de Dermatología de la Universidad Médica de Viena, Austria, y la práctica de cáncer de piel de Cliff Rosendahl en Queensland, Australia. Estas imágenes representan una amplia variedad de diagnósticos importantes en el ámbito de las lesiones pigmentadas de la piel, como queratosis actínica, carcinoma intraepitelial / enfermedad de Bowen (akiec), carcinoma de células basales (bcc), lesiones benignas similares a queratosis (lentigos solares / queratosis seborreicas y queratosis liquenoides, bkl), dermatofibroma (df), melanoma (mel), nevus melanocíticos (nv) y lesiones vasculares (angiomas, angiokeratomas, granulomas piogénicos y hemorragias, vasc).

Con más del 50% de las lesiones confirmadas mediante histopatología (histo), y el resto verificadas a través de seguimiento (follow\_up), consenso de expertos (consenso) o microscopía confocal in vivo (confocal), el conjunto de datos ofrece una valiosa fuente de información clínica.

Las imágenes proceden de pacientes austriacos referidos a un centro especializado en melanoma, quienes a menudo presentan un alto número de nevos y antecedentes de melanoma. Mientras tanto, las imágenes australianas provienen de un centro de atención primaria en un área con alta incidencia de cáncer de piel, donde el daño solar crónico es prevalente. Los pacientes australianos se caracterizan por un daño solar crónico grave. La piel dañada crónicamente por el sol se caracteriza por múltiples lentigos solares y vasos ectásicos, que a menudo están presentes en la periferia de la lesión objetivo.

### **5.2. Resultado de entrenamiento**

Durante el proceso de entrenamiento, se ha seguido tanto la pérdida en el conjunto de entrenamiento (loss) como en el conjunto de validación (val\_loss). Se busca analizar cómo están convergiendo y si el modelo está aprendiendo de manera efectiva sin sobreajuste o subajuste.

En el título de cada gráfica se muestra el nombre del modelo CNN preentrenado utilizado, junto con las abreviaturas CVE (con variables exógenas) y SVE (sin variables exógenas).



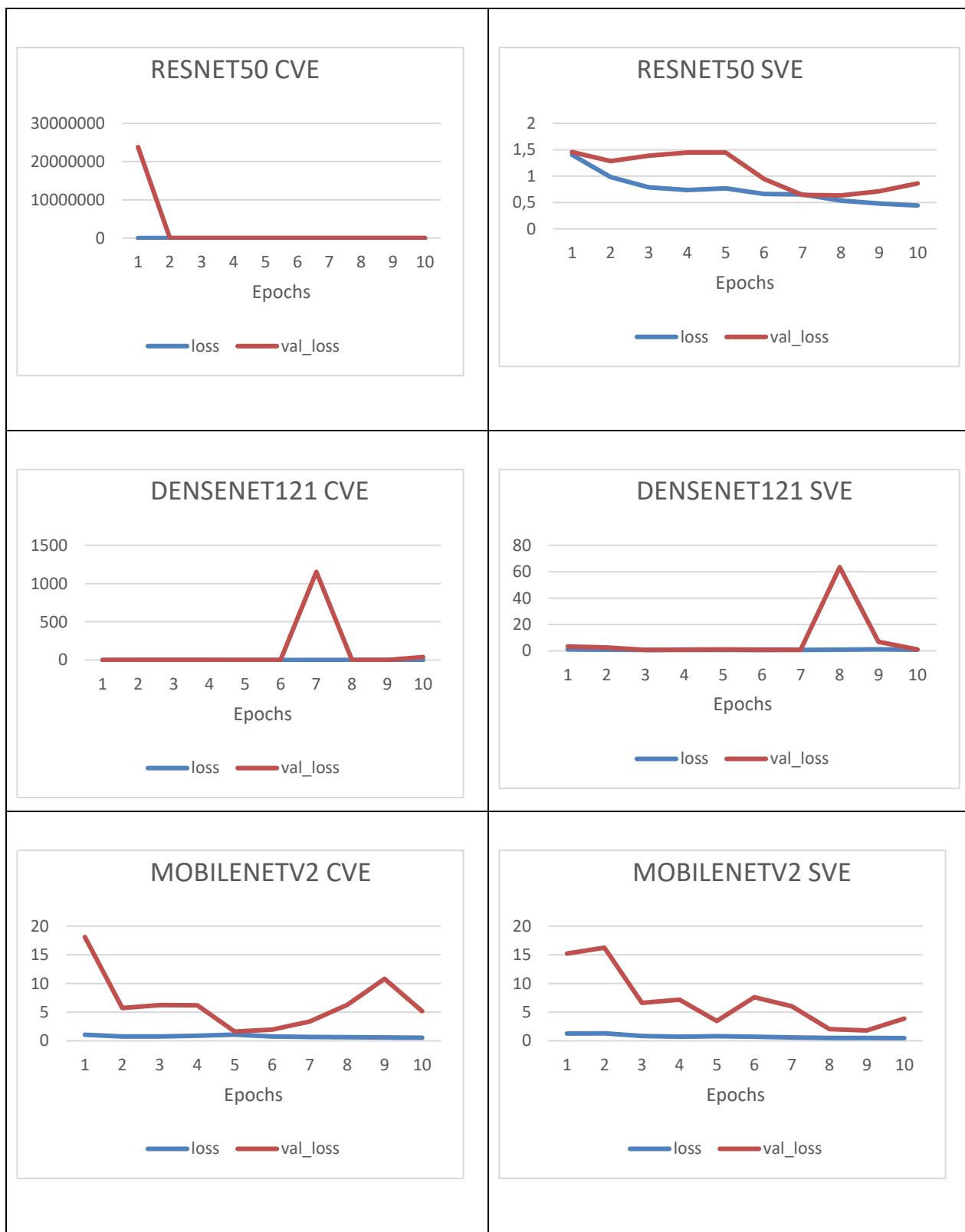


Tabla 42: Resultado de entrenamiento de modelos con variables exógenas vs sin variables exógenas

La pérdida (loss) representa una medida de cuán lejos están las predicciones del modelo de las etiquetas reales en el conjunto de entrenamiento, mientras que la pérdida de validación (val\_loss) indica el rendimiento del modelo en un conjunto de datos no visto durante el entrenamiento.

En general, los resultados de las gráficas presentes en la Tabla 42 sugieren que la inclusión de variables exógenas mejora el rendimiento de los modelos de clasificación de imágenes en términos de pérdida y pérdida de validación. Sin embargo, se observa cierta variabilidad en el rendimiento entre diferentes arquitecturas de modelos.

Los modelos entrenados desde cero (CNN desde cero) muestran un buen rendimiento en general, con pérdidas y pérdidas de validación relativamente bajas, especialmente cuando se incluyen variables exógenas. Por otro lado, modelos como Xception y EfficientB0 también muestran resultados prometedores, aunque algunos modelos presentan picos en la pérdida de validación, lo que podría indicar un potencial sobreajuste.

### **5.3. Resultado de evaluación**

En este apartado se presenta el resultado de la evaluación de los modelos, considerando las diversas configuraciones. Se analizan aquellos que incorporan variables exógenas (CVE), así como aquellos que prescinden de esta inclusión (SVE). Además, se examinan los modelos que incluyen variables exógenas, pero excluyen "sexo" y "localización", dado que previamente se observó que la correlación de estas variables es baja. Por ende, el modelo se construye únicamente con "age" y "dx\_type" (CVE - age y dx\_type).

Las métricas presentadas en la tabla han sido previamente explicadas. En cuanto a la matriz de confusión, se organiza de la siguiente manera:

- Eje horizontal (columnas) se encuentran las clases predichas por el modelo
- Eje vertical (filas) se muestran las clases verdaderas o reales.

Cada celda de la matriz representa el número de instancias clasificadas correcta o incorrectamente.

- La primera fila corresponde a la clase "bkl" (Benign keratosis-like lesions).
- La segunda fila corresponde a la clase "nv" (Melanocytic nevi).
- La tercera fila corresponde a la clase "df" (Dermatofibroma)

- La cuarta fila corresponde a la clase "mel" (Melanoma).
- La quinta fila corresponde a la clase "vasc" (Vascular lesions).
- La sexta fila corresponde a la clase "bcc" (Basal cell carcinoma).
- La séptima fila corresponde a la clase "akiec" (Actinic keratoses).

Para interpretar la matriz de confusión, se presta atención a la diagonal principal, ya que muestra el número de predicciones correctas para cada clase. Los valores fuera de la diagonal principal indican los casos mal clasificados. Cuanto más cercano a cero sea el valor fuera de la diagonal principal, mejor será el rendimiento del modelo, ya que indica que hay menos casos mal clasificados. Para una comprensión más clara, la Tabla 46 detalla la matriz de confusión del modelo EfficientNetB0.

### 5.3.1. DenseNet121

	CVE	CVE - age y dx_type	SVE
<b>Accuracy</b>	0.7913	0.7868	0.7129
<b>Precision</b>	0.7827	0.7764	0.6865
<b>Recall</b>	0.7913	0.7868	0.7129
<b>F1 Score</b>	0.7839	0.7772	0.6676
<b>Matriz de confusión</b>	[[ 25 18 8 0 6 4 0] [ 20 62 9 1 14 2 3] [ 10 15 95 0 49 28 1] [ 6 4 0 10 6 1 2] [ 7 6 25 1 1264 48 0] [ 5 3 21 0 89 105 0] [ 0 2 1 0 3 0 24]]	[[ 17 10 14 0 5 15 0] [ 16 56 13 0 17 7 2] [ 5 8 105 0 55 24 1] [ 2 10 6 2 6 2 1] [ 2 2 36 1 1256 53 1] [ 4 3 19 0 72 119 6] [ 0 2 0 0 6 1 21]]	[[ 31 9 7 0 10 4 0] [ 29 16 10 0 55 1 0] [ 13 4 59 0 109 13 0] [ 7 2 5 0 15 0 0] [ 11 3 28 0 1277 32 0] [ 10 1 31 0 136 45 0] [ 2 0 0 0 27 1 0]]

Tabla 43: Resultado de evaluación DenseNet121

### 5.3.2. MobileNetV2

	CVE	CVE - age y dx_type	SVE
<b>Accuracy</b>	0.6814	0.6789	0.4063
<b>Precision</b>	0.6525	0.5461	0.8386
<b>Recall</b>	0.6814	0.6789	0.4063
<b>F1 Score</b>	0.5606	0.5535	0.4516
<b>Matriz de confusión</b>	[[ 0 3 0 0 58 0 0] [ 0 2 2 0 107 0 0] [ 0 1 11 0 186 0 0] [ 0 1 0 0 28 0 0] [ 0 0 0 0 1351 0 0] [ 0 2 2 0 218 1 0] [ 0 0 1 0 29 0 0]]	[[ 0 0 0 0 61 0 0] [ 0 0 0 0 111 0 0] [ 0 0 9 0 189 0 0] [ 0 0 0 0 29 0 0] [ 0 0 0 0 1351 0 0] [ 0 0 1 0 222 0 0] [ 0 0 0 0 30 0 0]]	[[ 1 4 56 0 0 0 0] [ 1 21 86 0 3 0 0] [ 0 1 195 0 2 0 0] [ 1 5 22 1 0 0 0] [ 0 1 756 0 594 0 0] [ 0 0 211 0 11 1 0] [ 0 4 16 1 8 0 1]]]

Tabla 44: Resultado de evaluación MobileNetV2

### 5.3.3. Resnet50

	CVE	CVE - age y dx_type	SVE
Accuracy	0.7868	0.7563	0.6749
Precision	0.7724	0.7588	0.7559
Recall	0.7868	0.7563	0.6749
F1 Score	0.7727	0.7352	0.6978
Matriz de confusión	<pre>[[ 18  25  10   0   4   4   0]  [ 10  72   5   0  17   7   0]  [  6  19  93   1  59  19   1]  [  6   9   1   7   6   0   0]  [  2   6  27   0 1280  36   0]  [  3   8  23   0 100  87   2]  [  0   1   3   0   5   2 19]]</pre>	<pre>[[  7  28  20   0   4   1   1]  [  2  73  14   0  15   1   6]  [  0  19 127   2  44   5   1]  [  0  14   4   1   8   0   2]  [  0  16  65   1 1248  21   0]  [  2  23  70   0  81  39   8]  [  0   3   3   1   2   1 20]]</pre>	<pre>[[ 27  12   7   1   3  11   0]  [ 13  74   4   0   6  13   1]  [ 11   7  84   0  29  67   0]  [  4  15   2   0   5   3   0]  [ 17  29  77   1 989 237   1]  [  5   2  23   0 31 162   0]  [  1   5   0   0   4   4 16]]</pre>

Tabla 45 : Resultado de evaluación ResNet50

### 5.3.4. EfficientNetB0

	CVE	CVE - age y dx_type	SVE
Accuracy	0.7808	0.7738	0.6719
Precision	0.7784	0.7794	0.6817
Recall	0.7808	0.7738	0.6719
F1 Score	0.7646	0.7580	0.5426
Matriz de confusión	<pre>[[  8  12  27   0   9   5   0]  [  3  45  20   0  31   7   5]  [  1   1 127   0  49  19   1]  [  0   3   8   6   9   2   1]  [  0   3  47   0 1265  35   1]  [  0   2  43   0  88  89   1]  [  0   1   1   0   4   0 24]]</pre>	<pre>[[ 17  19   3   1   9  12   0]  [  9  73   2   7  14   4   2]  [  6  17  42   1  80  52   0]  [  2   6   0  10  11   0   0]  [  4  21   2   0 1248  75   1]  [  3  13   5   1  66 135   0]  [  0   2   0   0   2   1 25]]</pre>	<pre>[[  0   0   0   0   0  61   0   0]  [  0   0   1   0 110   0   0]  [  0   0   0   0 198   0   0]  [  0   0   0   0   0  29   0   0]  [  0   0   5   0 1346   0   0]  [  0   0   2   0 221   0   0]  [  0   0   0   0   0  30   0   0]]</pre>
Explicación matriz de confusión	<ul style="list-style-type: none"> <li>- <b>Primera fila, "bkl":</b> el modelo predijo correctamente 8 casos como "bkl", 12 casos como "nv", 27 casos como "df", 0 casos como "mel", 9 casos como "vasc", 5 casos como "bcc" y 0 caso como "akiec".</li> <li>- <b>Segunda fila, "nv":</b> el modelo predijo correctamente 45 casos como "nv", 3 casos como "bkl", 20 como "df", 0 casos como "mel", 31 casos como "vasc", 7 casos como "bcc" y 5 casos como "akiec".</li> <li>- <b>Tercera fila, "df":</b> el modelo predijo correctamente 127 casos como "df", 1 casos como "bkl", 1 casos como "nv", 0 casos como "mel", 49 casos como "vasc", 19 casos como "bcc" y 1 casos como "akiec".</li> <li>- <b>Cuarta fila, "mel":</b> el modelo predijo correctamente 6 casos como "mel", 0 casos como "bkl", 3 casos como "nv", 8 caso como "df", 9 casos como "vasc", 2 caso como "bcc" y 1 casos como "akiec".</li> <li>- <b>Quinta fila, "vasc":</b> el modelo predijo correctamente 1265 casos como "vasc", 0 casos como "bkl", 3 casos como "nv", 47 casos como</li> </ul>		
CVE			

	<p>"df", 0 casos como "mel", 35 casos como "bcc" y 1 caso como "akiec".</p> <ul style="list-style-type: none"> <li><b>Sexta fila, "bcc":</b> el modelo predijo correctamente 89 casos como "bcc", 0 casos como "bkl", 2 casos como "nv", 43 casos como "df", 0 casos como "mel", 88 casos como "vasc" y 1 casos como "akiec".</li> <li><b>Séptima fila, "akiec":</b> el modelo predijo correctamente 24 casos como "akiec", 0 casos como "bkl", 1 casos como "nv", 1 casos como "df", 0 casos como "mel", 4 casos como "vasc" y 0 casos como "bcc".</li> </ul>
--	--

Tabla 46: Resultado de evaluación EfficientNetB0

### 5.3.5. Xception

	CVE	CVE - age y dx_type	SVE
<b>Accuracy</b>	0.7978	0.7873	0.7698
<b>Precision</b>	0.8036	0.7886	0.7513
<b>Recall</b>	0.7978	0.7873	0.7698
<b>F1 Score</b>	0.7917	0.7833	0.7514
<b>Matriz de confusión</b>	<pre>[[ 24  20  14   1   1   1   0]  [ 9  82   5   5   8   0  2]  [ 7  12 131   1  41   6   0]  [ 0   5   1  20   3   0   0]  [ 1  20  57   1 1244  27  1]  [10  15  52   1  71  73  1]  [ 0   1   2   0   2   1 24]]</pre>	<pre>[[ 37   5   15   0   1   3   0]  [13  56  13  11  11   6  1]  [15   1 105   1  45  31  0]  [ 5   4   2  15   3   0   0]  [ 7   3  49   0 1254  38  0]  [ 9   0  34   3  81  96  0]  [ 0   2   4   2   8   0 14]]</pre>	<pre>[[ 13  17  21   1   7   2   0]  [ 7  65  10   5  20   1  3]  [ 3  14 106   0  64  11  0]  [ 3   6   3   9   8   0   0]  [ 0  10  41   0 1266  34  0]  [ 2   8  41   0 111  60  1]  [ 0   6   0   0   1   0 23]]</pre>

Tabla 47: Resultado de evaluación Xception

### 5.3.6. CNN desde cero

	CVE	CVE - age y dx_type	SVE
<b>Accuracy</b>	0.7563	0.7498	0.7169
<b>Precision</b>	0.7802	0.7810	0.6975
<b>Recall</b>	0.7563	0.7498	0.7169
<b>F1 Score</b>	0.7628	0.7505	0.6880
<b>Matriz de confusión</b>	<pre>[[ 27  14    8   1   2   8   1]  [17  62    2   3   10  15  2]  [10   8  80   5  46  49  0]  [ 4   6   1  12   6   0   0]  [ 2  15  26   3 1169 136  0]  [ 4   6  18   0  47 147  1]  [ 2   2   3   1   2   2 18]]</pre>	<pre>[[ 8   8  29   0   5  11   0]  [ 8  41  36   0  15   9  2]  [ 2   2 132   0  29  33  0]  [ 2   7  10   2   7   1  0]  [ 0   3  74   0 1174 100  0]  [ 2   1  37   0  53 130  0]  [ 1   0   6   0   3   5 15]]</pre>	<pre>[[ 4   7  29   0  15   5   1]  [ 6  15  25   2  54   4  5]  [ 0   2 116   0  63  16  1]  [ 1   2  14   1  10   1  0]  [ 0   0  90   0 1222 38  1]  [ 1   0  39   0 123  60  0]  [ 1   0   1   0  10   0 18]]</pre>

Tabla 48: Resultado de evaluación CNN desde cero

## 5.4. Comparación y discusión de resultados

Como los datos están desbalanceados, el F1 Score destaca como una de las métricas que mejor evalúan la clasificación porque consideran tanto los falsos positivos como los falsos negativos, lo que los hace robustos frente a desequilibrios en las clases. [53]

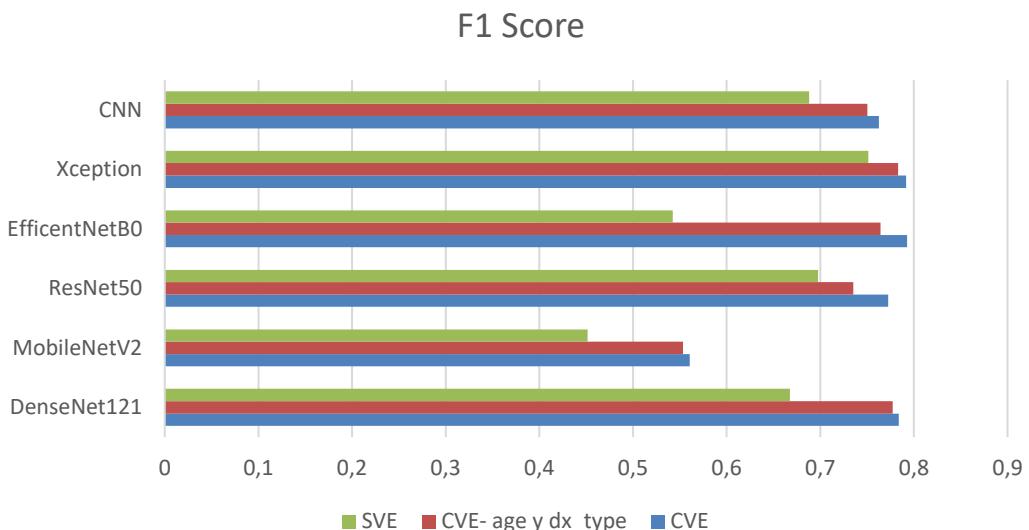
Por consiguiente, en la Tabla 49,50 y Figura 28 se recogen los resultados de esta métrica para todos los modelos.

F1 Score			
	CVE	SVE	Diferencia
<b>DenseNet121</b>	0.7839	0.6676	0.1163
<b>MobileNetV2</b>	0.5606	0.4516	0.1090
<b>ResNet50</b>	0.7727	0.6978	0.0749
<b>EfficientNetB0</b>	0.7646	0.5426	0.2220
<b>Xception</b>	0.7917	0.7514	0.0403
<b>CNN</b>	0.7628	0.6880	0.0748

Tabla 49: Tabla resultados de f1 score diferencia entre modelos con y sin variables exógenas

F1 Score			
	CVE	CVE - age y dx_type	SVE
<b>DenseNet121</b>	0.7839	0.7772	0.6676
<b>MobileNetV2</b>	0.5606	0.5535	0.4516
<b>ResNet50</b>	0.7727	0.7352	0.6978
<b>EfficientNetB0</b>	0.7646	0.7580	0.5426
<b>Xception</b>	0.7917	0.7833	0.7514
<b>CNN</b>	0.7628	0.7505	0.6880

Tabla 50: Tabla resultados de f1 score de los modelos



*Figura 28: Gráfica resultados de f1 score de los modelos*

La comparación de los modelos revela que diferentes arquitecturas tienen diferentes niveles de rendimiento. Por ejemplo, Xception y DenseNet tienen F1 Scores más altos en comparación con otros modelos como MobileNetV2. Esto sugiere que la elección del modelo puede influir significativamente en el rendimiento de la clasificación, y es importante seleccionar el modelo más adecuado para la tarea específica.

La inclusión de variables exógenas en el modelo también tiene un impacto significativo en su rendimiento. La comparación entre el conjunto de variables exógenas (CVE) y el conjunto sin variables exógenas (SVE) revela diferencias significativas en los resultados. F1 Score es más alto cuando se incluyen las variables exógenas en comparación con cuando no se incluye ninguna.

Por otro lado, al utilizar todas las variables exógenas (CVE), se observan F1 Scores ligeramente superiores en comparación con el uso exclusivo de las variables "age" y "dx\_type", prescindiendo así de las variables exógenas "sex" y "localization". Este hallazgo resulta intrigante, dado que las correlaciones entre las variables sugieren que "dx\_type" y "age" son las más significativas para el modelo, mientras que las otras dos, "sex" y "localization", muestran correlaciones más bajas con la variable objetivo.

A pesar de estas relaciones, el modelo parece beneficiarse de la inclusión de todas las variables exógenas, lo que sugiere que podrían estar proporcionando información

adicional que resulta útil para la predicción. Es posible que estas variables contengan señales relevantes que el modelo puede aprovechar para mejorar su capacidad predictiva.

En resumen, estos hallazgos sugieren que el rendimiento de los modelos de clasificación puede variar según la selección de variables y que el F1 Score es una métrica útil para evaluar el rendimiento de los modelos, especialmente en situaciones de desbalance en las clases. Además, la inclusión de variables exógenas mejora el rendimiento predictivo de los modelos en la clasificación.

## 5.5. Plataforma web predicción

Esta sección expone las partes y los requisitos de la aplicación web, así como las pruebas realizadas, interfaz gráfica de la plataforma y algunos detalles de la implementación.

### 5.5.1. Partes de la aplicación

Como se ve en la Figura 29 las partes de la aplicación son:

- **Inicio** = En esta sección, los usuarios son recibidos con un menú de navegación que les permite explorar las diferentes secciones de la aplicación y con una pantalla de bienvenida que les proporciona instrucciones sobre cómo utilizar el escáner para obtener la predicción del diagnóstico.
- **Escáner** = En esta sección, los usuarios pueden cargar una imagen de la lesión cutánea que desean analizar. Después de cargar la imagen, deben completar los datos requeridos, como el género, la edad, la localización de la lesión y el tipo de diagnóstico. Una vez que todos los datos han sido proporcionados, los usuarios pueden hacer clic en el botón "Realizar predicción" para obtener un diagnóstico. Esta sección proporciona una herramienta interactiva para la evaluación inicial de las lesiones cutáneas, que no debe sustituir la consulta médica profesional.
- **Sobre el escáner** = Esta sección permite a los usuarios cargar una imagen de su lunar y agregar información sobre factores como edad, sexo, localización de la lesión y tipo de diagnóstico para obtener una clasificación basada en siete clases diferentes de lunares. Se muestran siete imágenes, cada una representando uno de los tipos de lesiones cutáneas presentes en el conjunto de datos.

- **Sobre lesiones cutáneas** = En esta sección se da información general sobre las lesiones cutáneas, su importancia y cómo reconocerlas. Se incluye una imagen que muestra la diferenciación entre tumores benignos y malignos para ayudar a los usuarios a comprender mejor.
- **Factores de riesgo** = En esta sección se enumeran varios factores de riesgo asociados con las lesiones cutáneas, como la exposición al sol, los antecedentes familiares de cáncer de piel y la inmunosupresión. Se proporciona un vídeo con información sobre el cáncer de piel, sus causas, síntomas y prevención, ofrecido por la AECC (Asociación Española Contra el Cáncer).
- **Prevención y detección precoz** = Esta sección presenta estrategias para prevenir las lesiones cutáneas y detectarlas tempranamente. Se adjunta un vídeo sobre la regla del ABCD para la autoexploración, proporcionado por la Asociación Española Contra el Cáncer. Los usuarios pueden aprender sobre la importancia de realizar autoexámenes regulares de la piel y la detección temprana de posibles problemas cutáneos.



*Figura 29: Partes de la aplicación web de predicción*

## **5.5.2. Requisitos de la aplicación**

Los requisitos de la aplicación son cruciales, según (Affde 2021) ya que proporcionan una guía clara sobre lo que se debe desarrollar, permitiendo un entendimiento común entre los equipos de desarrollo y los clientes. Establecen las bases para la planificación del proyecto, la estimación de costos y la evaluación del éxito del producto. Además, ayudan a priorizar tareas y recursos, lo que contribuye a la eficiencia y al cumplimiento de los objetivos comerciales.

Los requisitos funcionales detallan las acciones específicas que los usuarios llevan a cabo en un sistema o producto, delineando lo que necesita ser implementado. Estos requisitos abarcan todas las funciones esenciales que el usuario final específicamente solicita como componentes básicos para operar el sistema. Normalmente, se documentan en el contrato y se presentan en forma de datos que se integrarán en el sistema. Son los requisitos expresados por el usuario que pueden ser verificados y observados directamente en el producto final, en contraste con los requisitos no funcionales.

Por otro lado, los requisitos no funcionales describen las características y propiedades del sistema o producto de software, especificando cómo opera y qué cualidades tiene en particular. Principalmente se refieren a una variedad de características relacionadas con la calidad del producto, tales como su fiabilidad, capacidad de escalabilidad, rendimiento y más. Estos requisitos no funcionales buscan satisfacer las expectativas del usuario, ya que representan atributos esenciales del producto.

### **5.5.2.1. Requisitos funcionales**

#### **RF-01. Inicio**

- **RF-01.1:** El sistema muestra las diferentes acciones disponibles, incluyendo opciones como "Inicio", "Escáner", "Sobre el escáner", "Sobre lesiones cutáneas", "Factores de riesgo" y "Prevención y detección precoz".

#### **RF-02. Escáner**

- **RF-02.1:** El sistema permite a los usuarios cargar una imagen de una lesión cutánea. Debe subir la imagen antes de realizar la predicción obligatoriamente.

- **RF-02.2:** El sistema solicita a los usuarios completar datos como género, edad, localización de la lesión cutánea y tipo de diagnóstico. Todos los campos deben completarse antes de realizar la predicción obligatoriamente.
- **RF-02.3:** El sistema realiza una predicción utilizando el modelo de aprendizaje automático proporcionado.
- **RF-02.4:** El sistema muestra el resultado de la predicción, proporcionando un diagnóstico de la lesión cutánea.

#### **RF-03. Sobre el escáner**

- **RF-03.1:** El sistema proporciona información sobre el funcionamiento del escáner de lesiones cutáneas, incluyendo qué tipos de lesiones puede identificar.
- **RF-03.2:** El sistema redirige a una nueva pestaña que abre la página web de donde se obtuvo la información al hacer clic en la fuente de la información.

#### **RF-04. Sobre lesiones cutáneas**

- **RF-04.1:** El sistema presenta información sobre los diferentes tipos de lesiones cutáneas, incluyendo imágenes y descripciones de lesiones benignas y malignas. Se resalta la importancia de monitorear cualquier cambio en la piel y consultar a un médico ante la presencia de lesiones sospechosas.
- **RF-04.2:** El sistema redirige a una nueva pestaña que abre la página web de donde se obtuvo la información al hacer clic en la fuente de la información.

#### **RF-05. Factores de riesgo**

- **RF-05.1:** El sistema ofrece información sobre los factores de riesgo asociados con las lesiones cutáneas, como la exposición al sol, antecedentes familiares de cáncer de piel y características personales de la piel.
- **RF-05.2:** El sistema redirige a una nueva pestaña que abre la página web de donde se obtuvo la información al hacer clic en la fuente de la información.
- **RF-05.3:** El sistema reproduce el vídeo y también puede redirigir al usuario al vídeo en YouTube

#### **RF-06. Prevención y detección precoz**

- **RF-06.1:** El sistema proporciona consejos y estrategias para la prevención y detección precoz de lesiones cutáneas.
- **RF-06.2:** El sistema reproduce el vídeo y también puede redirigir al usuario al vídeo en YouTube.

### **5.5.2.2. Requisitos no funcionales**

#### **RNF-01. Disponibilidad**

- **RNF-01.1.** El sistema mantiene su disponibilidad en todo momento para los usuarios, maximizando su accesibilidad.

#### **RNF-02. Seguridad**

- **RNF-02.1.** El sistema implementa medidas de seguridad para garantizar la protección de la privacidad y confidencialidad de los datos de los usuarios, especialmente durante la carga y procesamiento de imágenes de lesiones cutáneas.

#### **RNF-03. Rendimiento**

- **RNF-03.1.** El sistema proporciona resultados precisos en un tiempo de respuesta rápido y fluido, asegurando una experiencia eficiente para los usuarios.

#### **RNF-04. Interfaz**

- **RNF-04.1.** El sistema presenta una interfaz con un diseño claro y sencillo, permitiendo que los usuarios accedan intuitivamente a sus funcionalidades.

#### **RNF-05. Usabilidad**

- **RNF-05.1.** El sistema requiere una conexión a internet para su funcionamiento adecuado.

#### **RNF-06. Portabilidad**

- **RNF-06.1.** El sistema es compatible con diversos sistemas operativos y navegadores web, garantizando su accesibilidad para una amplia variedad de usuarios.

## RNF-07. Mantenimiento

- **RNF-07.1.** El sistema está estructurado y comentado de manera adecuada, facilitando su mantenimiento y posibles actualizaciones en el futuro.

### 5.5.3. Pruebas realizadas

De acuerdo con [54] “Detrás de cada plataforma digital de éxito, hay un robusto proceso de pruebas que garantiza la funcionalidad, seguridad y facilidad de uso. Este método es imperativo para asegurar que el producto final no sólo responda a las expectativas de los desarrolladores, sino que también exceda las del usuario final.”

Las pruebas realizadas en este proyecto se centran en revisar manualmente los contenidos del proyecto a través de la navegación de sus vistas, verificando que responde a las expectativas generadas, asegurando el correcto funcionamiento y la calidad de la misma. Estas pruebas se ejecutan en cada nueva versión de la aplicación.

La Tabla 51 proporciona un resumen claro y conciso de los resultados obtenidos durante estas pruebas. Enumera las acciones o funciones realizadas en la aplicación, junto con una indicación de si cada acción ha sido verificada (representada por "Si") y una referencia a la figura asociada.

Acción	Validación	Figura
Selección de los botones de Menú de Navegación	Si	Figura 30
Carga de imágenes	Si	Figura 31
Selección género, edad, localización de la lesión cutánea y tipo de diagnóstico	Si	Figura 32
Restricción para subir la imagen antes de la predicción y para seleccionar los datos. (variables exógenas)	Si	Figura 33
Realizar predicción	Si	Figura 34
Redirigir cuando se le da a la fuente de cada módulo	Si	Figura 35
Reproducir vídeo y redirigir al usuario al vídeo en YouTube	Si	Figura 36

Tabla 51: Pruebas realizadas

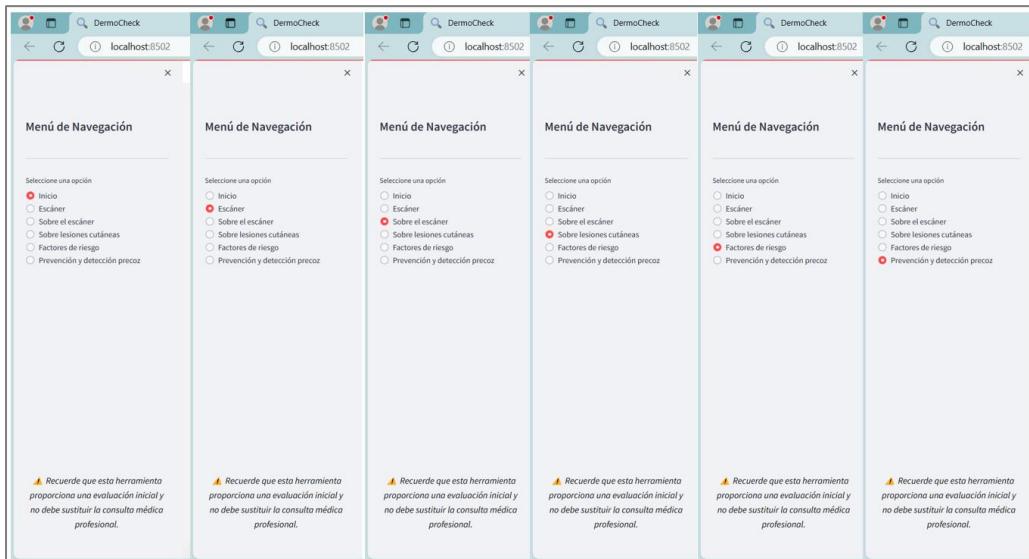


Figura 30: Prueba selección de los botones de Menú de Navegación

**Cargue la imagen de la lesión cutánea**

Seleccione o arrastre y suelte la imagen de la lesión cutánea:

Drag and drop file here  
Limit 200MB per file

lunar.jpeg 12.9KB X Browse files

Figura 31: Prueba carga de imágenes

**Complete los datos**

Género:

female  
 male  
 unknown

Edad:

0  100

Localización de la lesión cutánea:

foot X ▼

Tipo de Diagnóstico:

consensus X ▼

Realizar predicción

Figura 32: Prueba selección de las variables exógenas (género, edad, localización de la lesión cutánea y tipo de diagnóstico)

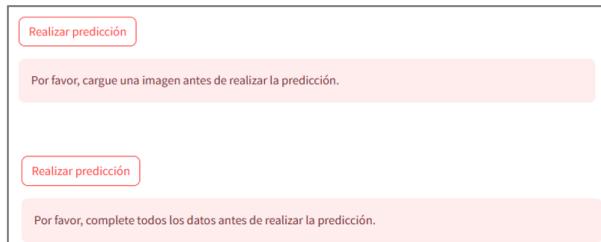


Figura 33: Prueba restricciones

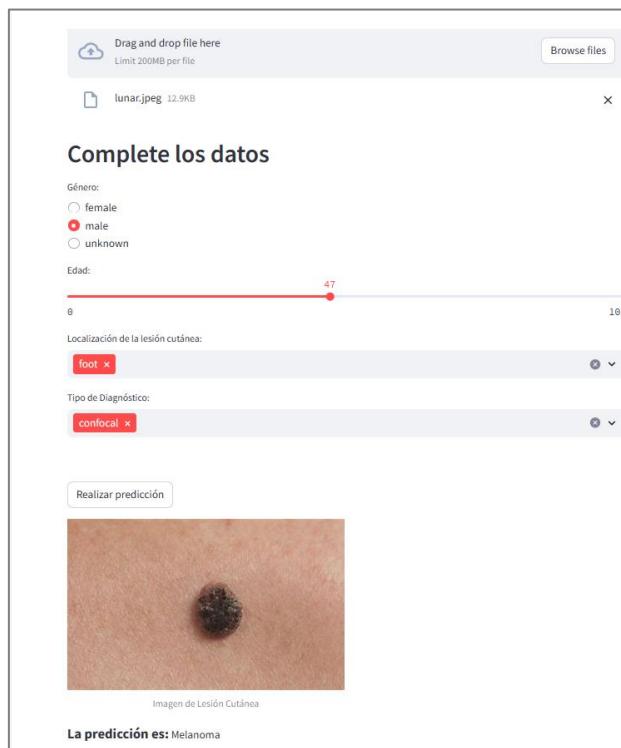


Figura 34: Prueba predicción

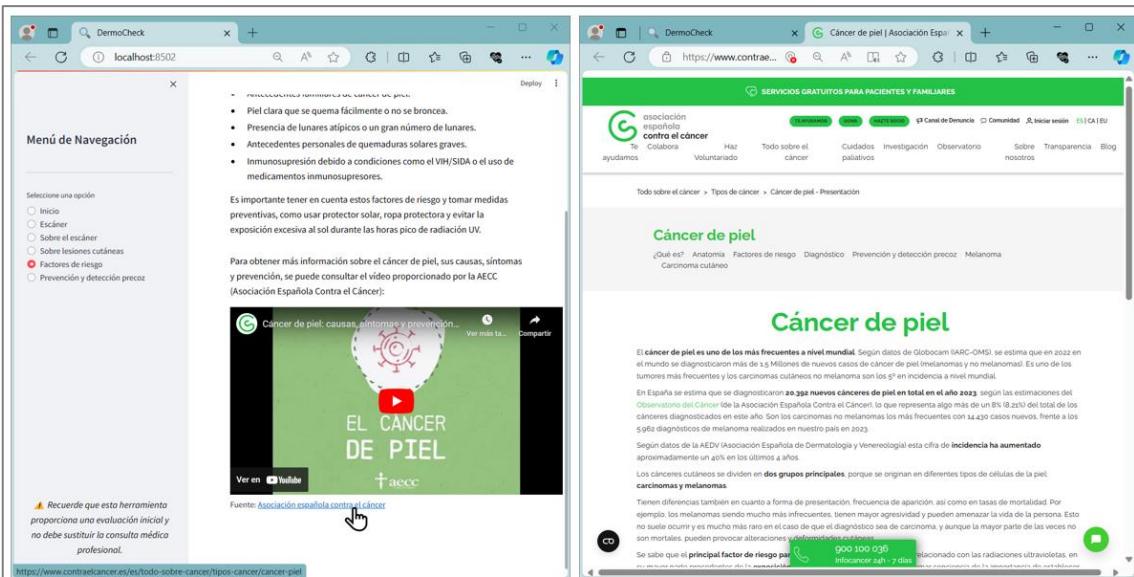


Figura 35: Prueba redirección de fuente

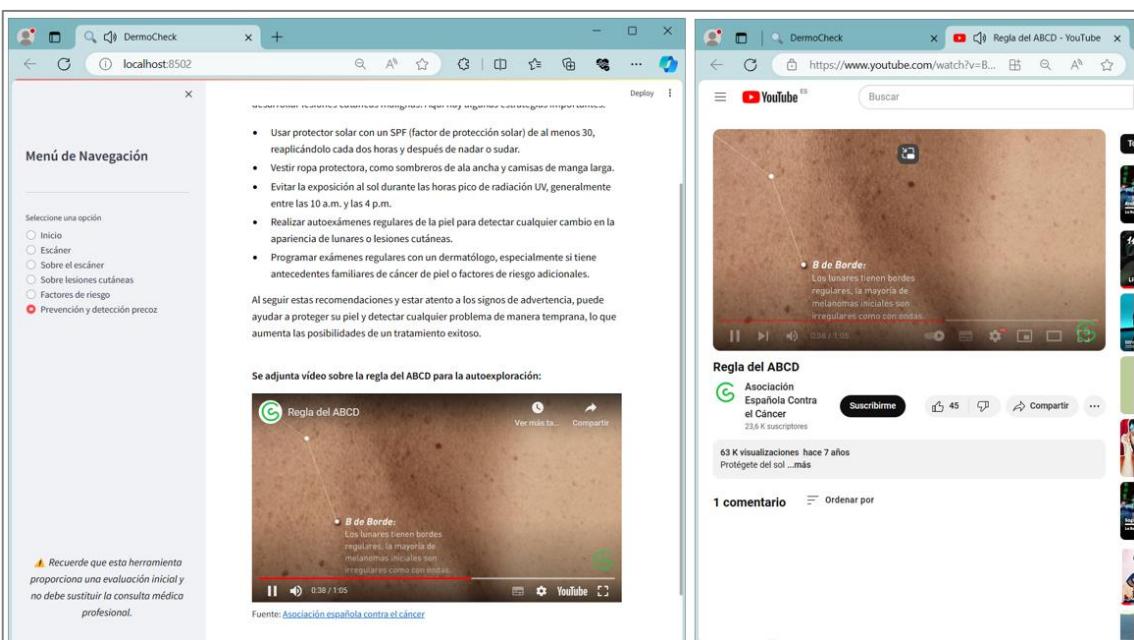


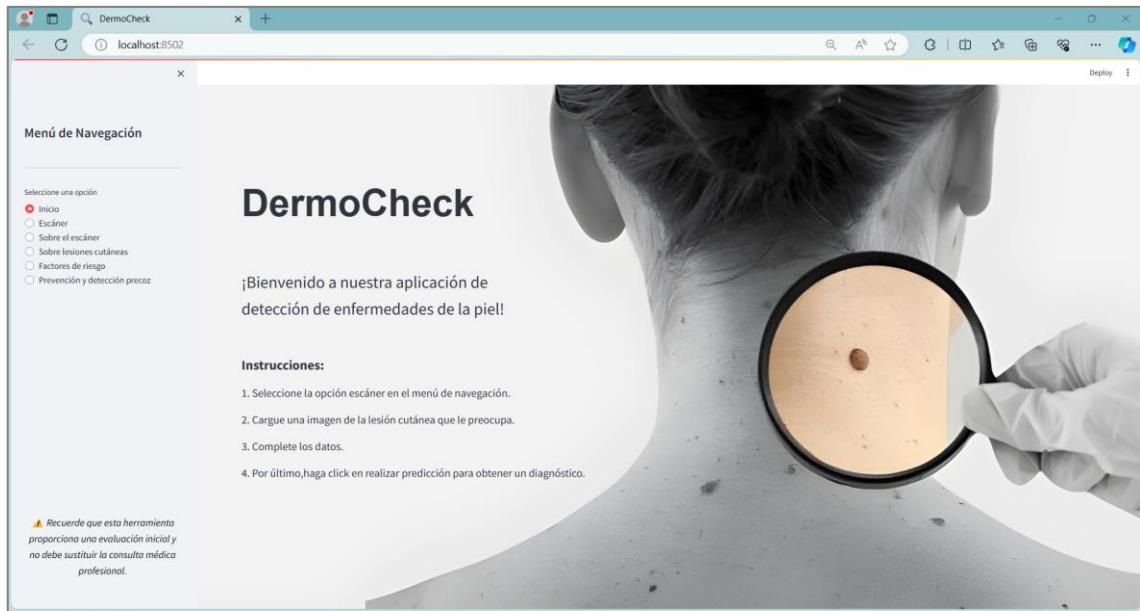
Figura 36: Prueba reproducción de vídeo y redirección a YouTube

#### 5.5.4. Interfaz gráfica de la plataforma

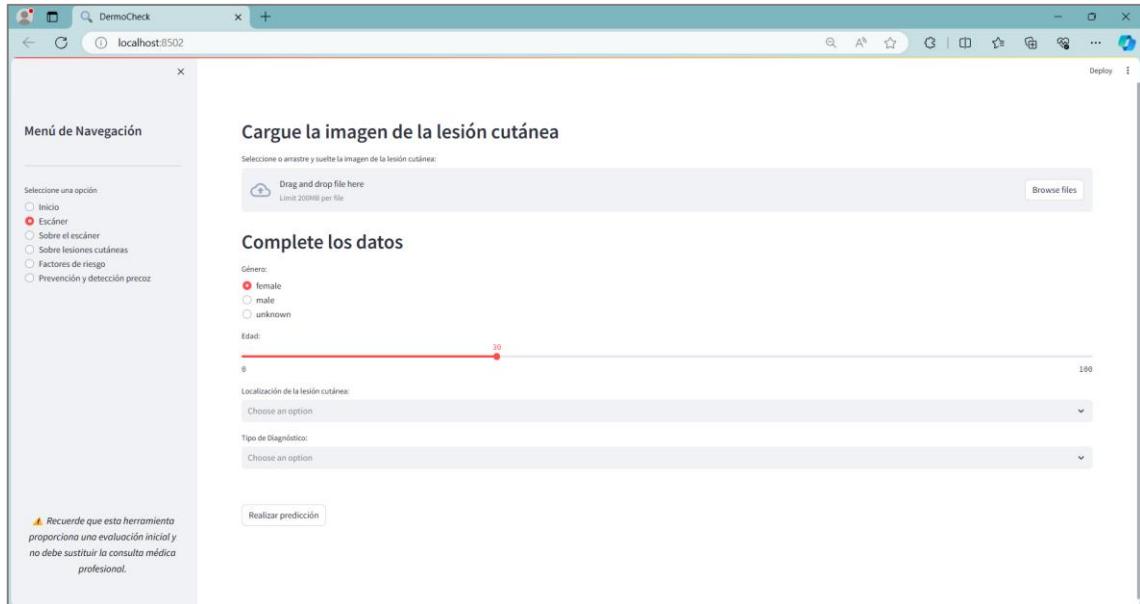
La interfaz de usuario refleja la primera impresión que los usuarios experimentan al interactuar con cualquier aplicación web.

En el caso de "DermoCheck", desempeña un papel crucial al ofrecer una experiencia sencilla y accesible para la detección de enfermedades de la piel.

Las siguientes capturas de pantalla ofrecen una visión detallada de cómo la plataforma presenta la información, desde el inicio (figura 37) hasta la sección de prevención y detección precoz (figura 43) pasando por el escáner (figura 38), sobre el escáner (figura 39 y 40), sobre lesiones cutáneas (figura 41) y factores de riesgo (figura 42)



*Figura 37: Plataforma web – Inicio*



*Figura 38:Plataforma web – Escáner*



Figura 39: Plataforma web - Sobre el escáner I

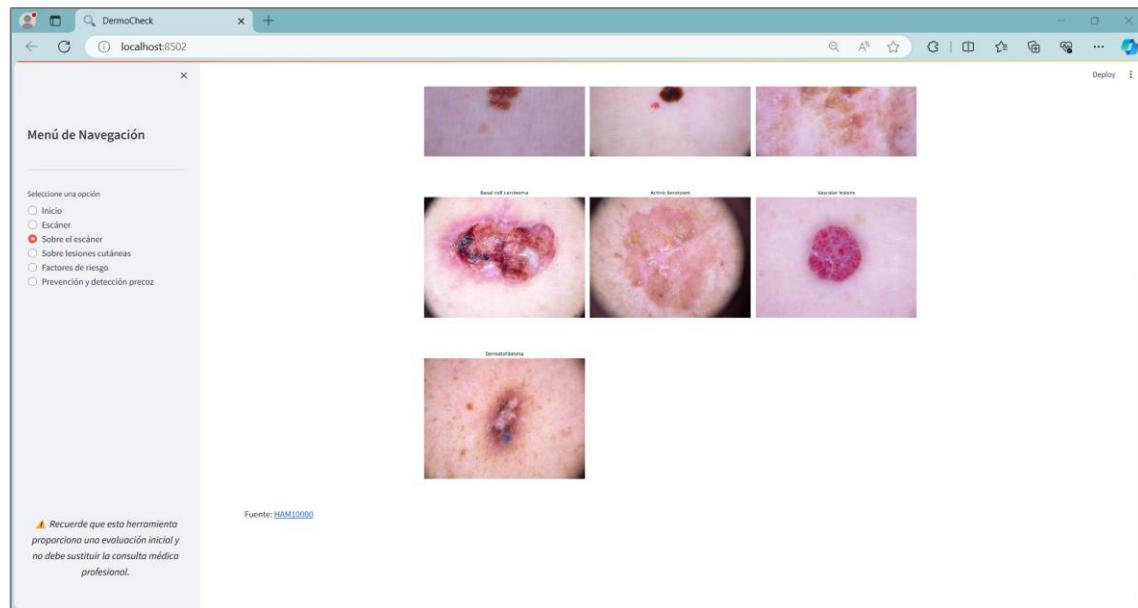


Figura 40: Plataforma web - Sobre el escáner II

**Sobre lesiones cutáneas**

**Menú de Navegación**

Seleccione una opción

- Inicio
- Escáner
- Sobre lesiones cutáneas
- Factores de riesgo
- Prevención y detección precoz

**Recuerde que esta herramienta proporciona una evaluación inicial y no debe sustituir la consulta médica profesional.**

La piel es el órgano más grande del cuerpo humano, actuando como una barrera protectora contra agentes externos como bacterias y sustancias químicas. Contiene melanina, que protege contra los rayos ultravioleta, y regula la temperatura corporal. Aunque la mayoría de las lesiones cutáneas no son graves, algunas pueden ser severas y representar un riesgo para la vida.

Es fundamental monitorizar cualquier cambio en la piel y consultar a un médico ante la presencia de lesiones sospechosas, ya que pueden ser benignas o malignas.

La imagen adjunta proporciona una representación visual de la diferenciación entre tumores benignos y malignos.

- En 'Benign Tumor', se muestra una célula que ilustra la naturaleza no cancerosa de este tipo de tumor, caracterizada por su incapacidad para propagarse a otros tejidos o órganos
- Por otro lado, en 'Malignant Tumor', se representa otra célula que refleja la naturaleza cancerosa de este tipo de tumor, con la capacidad de diseminarse a través del cuerpo, afectando otros tejidos y órganos

Fuentes: Universidad Europea | Imagen

Figura 41: Plataforma web - Sobre lesiones cutáneas

**Factores de riesgo**

**Menú de Navegación**

Seleccione una opción

- Inicio
- Escáner
- Sobre el escáner
- Factores de riesgo
- Prevención y detección precoz

**Recuerde que esta herramienta proporciona una evaluación inicial y no debe sustituir la consulta médica profesional.**

Los factores de riesgo para las lesiones cutáneas pueden variar desde la exposición al sol hasta antecedentes familiares de cáncer de piel. Aquí hay algunos factores de riesgo comunes a considerar:

- Exposición a la radiación ultravioleta (UV) del sol o camas de bronceado.
- Antecedentes familiares de cáncer de piel.
- Piel clara que se quema fácilmente o no se broncea.
- Presencia de lunares atípicos o un gran número de lunares.
- Antecedentes personales de quemaduras solares graves.
- Inmunosupresión debido a condiciones como el VIH/SIDA o el uso de medicamentos inmunosupresores.

Es importante tener en cuenta estos factores de riesgo y tomar medidas preventivas, como usar protector solar, ropa protectora y evitar la exposición excesiva al sol durante las horas pico de radiación UV.

Para obtener más información sobre el cáncer de piel, sus causas, síntomas y prevención, se puede consultar el video proporcionado por la AECC (Asociación Española Contra el Cáncer):

Fuente: Asociación española contra el cáncer

Figura 42:Plataforma web - Factores de riesgo

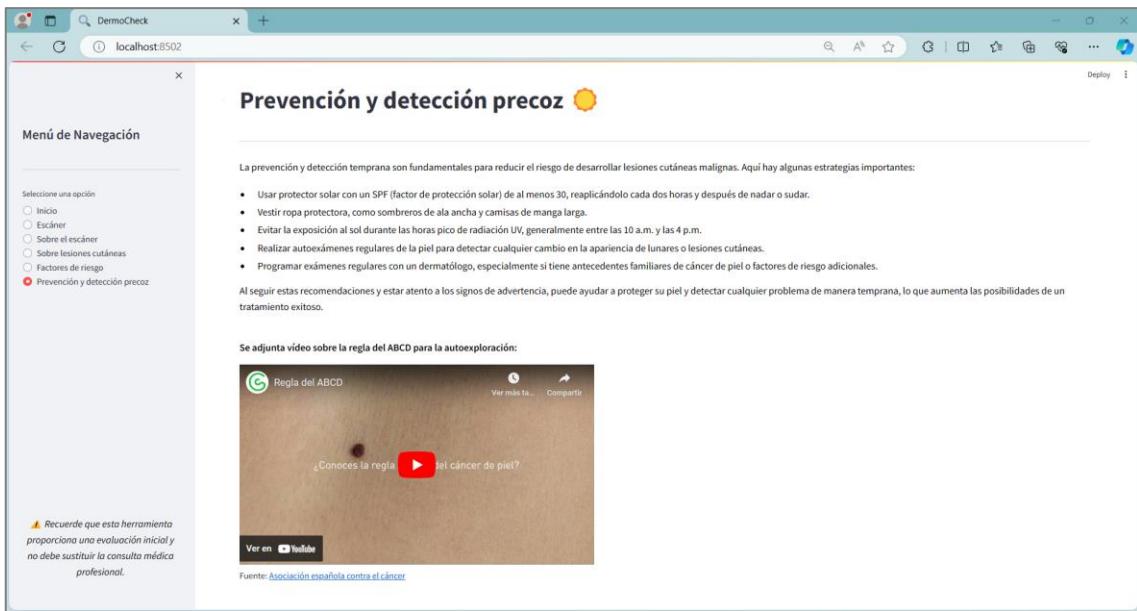


Figura 43: Plataforma web - Prevención y detección precoz

### 5.5.5. Detalles de implementación

En este apartado, se exploran en detalle los elementos de implementación que dan vida a la aplicación "DermoCheck". Desde la carga del modelo preentrenado hasta la presentación de resultados y la personalización del diseño, cada aspecto resulta vital en la experiencia del usuario y la efectividad del sistema.

#### Carga de modelo preentrenado:

- Se utiliza la biblioteca Keras para cargar un modelo de red neuronal preentrenado en este caso se ha elegido el de mayor precisión así como otras bibliotecas como Streamlit, io,base64 , y PIL

```
import streamlit as st
import io
import base64
import numpy as np
from PIL import Image
from keras.models import load_model
from keras.applications.efficientnet import preprocess_input
```

Código 13: Bibliotecas utilizadas

- El modelo con mayor precisión (EfficientNetB0) se carga desde una ruta de archivo especificada en la variable `model\_path` y se utiliza la función `load\_model` de Keras para cargar el modelo desde el archivo especificado.

```

def render_escaner():
    #ruta
    model_path = 'D:\\Users\\paula\\Escritorio\\modelo_preentrenado.h5'

    # Cargar el modelo
    @st.cache_resource
    def load_saved_model(model_path):
        model = load_model(model_path)
        return model

    model = load_saved_model(model_path)

```

*Código 14: Implementación de carga del modelo*

### **Preprocesamiento de imágenes:**

- Antes de realizar predicciones, las imágenes de las lesiones cutáneas se preprocesan para que coincidan con el formato esperado por el modelo. Esto implica cambiar el tamaño de la imagen a 64x64 píxeles. La función `preprocess\_image` se encarga de este preprocesamiento.

```

# Función para preprocesar la imagen
def preprocess_image(image):
    image = image.resize((64, 64)) # Cambiar tamaño para que coincida
    image = np.array(image)
    image = preprocess_input(image)
    return image

```

*Código 15: Preprocesamiento de la imagen*

### **Predicciones basadas en datos exógenos:**

- Además de las imágenes de las lesiones cutáneas, la aplicación solicita información adicional al usuario, como género, edad, localización de la lesión y tipo de diagnóstico. Esta información se utiliza junto con la imagen para realizar predicciones más precisas.
- Se codifican los datos categóricos (género, localización y tipo de diagnóstico) para que puedan ser utilizados por el modelo.

```

# Función para hacer predicciones
def predict(image, gender, age, localization, dx_type):
    processed_image = preprocess_image(image)
    # Convertir gender y age a números
    gender_num = 1 if gender == 'male' else 0 # Asignar 1 a masculino y 0 a femenino
    age_num = float(age) # Convertir edad a float
    # Convertir localization a números
    localization_num = [1 if loc in localization else 0 for loc in ['abdomen', 'scalp', 'back', 'ear', 'face',
    'foot', 'hand', 'lower extremity', 'neck', 'trunk', 'upper extremity', 'unknown']]
    # Asegurar que localization tenga una longitud de 12 (el mismo que en la creación del modelo)
    localization_num += [0] * (12 - len(localization_num))
    # Asegurar que dx_type tenga una longitud de 4 (el mismo que en la creación del modelo)
    dx_type_encoded = [0, 0, 0, 0] # Inicializar con ceros
    if dx_type == 'histo':
        dx_type_encoded[0] = 1
    elif dx_type == 'consensus':
        dx_type_encoded[1] = 1
    elif dx_type == 'confocal':
        dx_type_encoded[2] = 1
    elif dx_type == 'follow_up':
        dx_type_encoded[3] = 1
    # Concatenar todas las características tabulares
    tabular_input = [gender_num, age_num] + localization_num + dx_type_encoded
    # Asegurar que tabular_input tenga una longitud de 21 (el mismo que en la creación del modelo)
    tabular_input += [0] * (21 - len(tabular_input))
    # Convertir a numpy array
    tabular_input = np.array(tabular_input).reshape(1, -1)

    # Modelo toma dos entradas
    prediction = model.predict([np.expand_dims(processed_image, axis=0), tabular_input])
    return np.argmax(prediction)

```

Código 16: Predicciones basadas en datos exógenos

### Interfaz de usuario interactiva:

- Se utiliza Streamlit para una interfaz de usuario interactiva. Proporciona elementos como botones de radio, deslizadores y selección múltiple para una interacción intuitiva. Los usuarios pueden seleccionar opciones y realizar acciones fácilmente, mejorando la experiencia.

```

# Sección de variables exógenas
st.header('Complete los datos')
gender = st.radio('Género: ', ['female', 'male', 'unknown'])
age = st.slider('Edad:', min_value=0, max_value=100, value=30)
localization = st.multiselect('Localización de la lesión cutánea:', ['abdomen', 'scalp', 'back',
    'ear', 'face', 'foot', 'hand', 'lower extremity', 'neck', 'trunk',
    'upper extremity', 'unknown'])
dx_type = st.multiselect('Tipo de Diagnóstico:', ['histo', 'consensus', 'confocal', 'follow_up'])

# Botón de predicción
if uploaded_image is not None and st.button('Realizar predicción'):

```

Código 17: Implementación de botones, deslizadores y selección múltiple

### Presentación de resultados:

- Después de que el usuario carga una imagen y completa los datos, la aplicación muestra la imagen cargada junto con la predicción realizada por el modelo.
- Se utiliza un diccionario de nombres de clases para mapear los resultados numéricos de la predicción a nombres legibles para el usuario.

```

# Botón de predicción
if uploaded_image is not None and st.button('Realizar predicción'):
    # Verificar si se han completado todos los campos
    if gender != 'unknown' and age != 0 and localization and dx_type:
        st.image(uploaded_image, caption='Imagen de Lesión Cutánea', width=400)
        # Resto del código para la predicción...
        # Codificar dx_type como one-hot
        dx_type_encoded = [0, 0, 0] # Inicializar con ceros
        if dx_type == 'histo':
            dx_type_encoded[0] = 1
        elif dx_type == 'consensus':
            dx_type_encoded[1] = 1
        elif dx_type == 'confocal':
            dx_type_encoded[2] = 1
        elif dx_type == 'follow_up':
            dx_type_encoded[3] = 1

        # Convertir género y edad a cadenas
        gender = str(gender)
        age = str(age)

        # Definir el diccionario de nombres de clases
        class_names = {
            0: 'Benign keratosis-like lesions',
            1: 'Melanocytic nevi',
            2: 'Dermatofibroma',
            3: 'Melanoma',
            4: 'Vascular lesions',
            5: 'Basal cell carcinoma',
            6: 'Actinic keratoses'
        }

        # Hacer predicción
        prediction_num = predict(Image.open(uploaded_image), gender, age, localization, dx_type_encoded)

        # Obtener el nombre correspondiente a la predicción numérica
        prediction_name = class_names.get(prediction_num, 'Clase Desconocida')

        # Mostrar la predicción
        st.write('<span style="font-size:20px; font-weight:bold;">La predicción es:</span>', prediction_name, unsafe_allow_html=True)
    else:
        st.error("Por favor, complete todos los datos antes de realizar la predicción.")
elif uploaded_image is None and st.button('Realizar predicción'):
    st.error("Por favor, cargue una imagen antes de realizar la predicción.")

```

Código 18: Implementación presentación de resultados

### Estilo y personalización:

- Se utiliza CSS para personalizar el aspecto de la aplicación, incluida la adición de una imagen de fondo y estilos de texto.
- Se agregan imágenes y videos incrustados para proporcionar información adicional y visualizar conceptos relevantes.

```

st.markdown(
    '<div style="display: flex; justify-content: center;">
    
    </div>',
    unsafe_allow_html=True
)
st.write("Fuentes: [Universidad Europea](https://universidadeuropea.com/blog/tipos-lesiones-piel/) | "
    "[Imagen](https://medicinabasica.com/diferencias-entre-un-tumor-maligno-y-benigno)")

st.markdown('<iframe width="640" height="360" src="https://www.youtube.com/embed/Tp_WaONFEIU" frameborder="0" allowfullscreen></iframe>', unsafe_allow_html=True)
st.write("Fuente: [Asociación española contra el cáncer](https://www.contraelcancer.es/es/todo-sobre-cancer/tipos-cancer/cancer-piel)")

```

Código 19: Implementación imágenes y vídeos

## 6. Conclusión

---

En este proyecto se ha llevado a cabo el estudio del impacto de variables exógenas, como la ubicación de la lesión, la edad, el sexo del paciente y el tipo de diagnóstico, en la precisión y eficacia de los modelos de detección de enfermedades de la piel. Se han desarrollado dos enfoques de detección utilizando redes neuronales convolucionales (CNN): uno que considera variables exógenas y otro que no las considera. Diversas arquitecturas de modelos preentrenados, incluyendo DenseNet121, MobileNetV2, ResNet50, EfficientNetB0, Xception, así como una CNN diseñada desde cero se emplearon para clasificar siete afecciones cutáneas, utilizando una colección de imágenes de alta resolución de lesiones cutáneas etiquetadas.

Se han logrado cumplir los objetivos propuestos para este proyecto, ya que los resultados muestran que la inclusión de variables exógenas mejora la precisión de los modelos de clasificación de imágenes. Tanto los modelos entrenados desde cero como algunos modelos preentrenados, como Xception y DenseNet, presentan resultados prometedores en la detección de enfermedades cutáneas. La elección del modelo influye significativamente en el rendimiento de la clasificación, lo que subraya la importancia de seleccionar la arquitectura más adecuada.

Con respecto a la utilidad de la información exógena, se observa que la inclusión de todas las variables exógenas, como la ubicación de la lesión, la edad, el sexo del paciente y el tipo de diagnóstico, resulta beneficiosa para mejorar la capacidad predictiva de los modelos. Aunque las correlaciones entre estas variables y la variable objetivo varían, se aprecia que la inclusión de todas las variables exógenas proporciona información adicional que el modelo puede aprovechar para mejorar su rendimiento predictivo.

En cuanto a la implementación de una plataforma web de predicción, se ha desarrollado una aplicación llamada "DermoCheck" que ofrece una herramienta interactiva para la evaluación inicial de las lesiones cutáneas. Los usuarios pueden cargar imágenes de lesiones cutáneas, completar los datos de las variables exógenas y obtener un diagnóstico de la lesión.

En resumen, este estudio demuestra que la combinación de técnicas de aprendizaje profundo y la inclusión de variables exógenas mejora significativamente la precisión y eficiencia en el diagnóstico de enfermedades cutáneas. Estos hallazgos tienen el potencial de impactar positivamente en la detección temprana y el tratamiento efectivo de estas afecciones, mejorando los resultados clínicos en dermatología.

Además, el desarrollo de la aplicación web "DermoCheck" proporciona una herramienta práctica y accesible que puede ser utilizada por profesionales de la salud y pacientes para evaluar y monitorear la salud de la piel de manera conveniente y precisa. Esto representa un avance significativo en la telemedicina y la atención dermatológica, permitiendo una mayor accesibilidad y comodidad para los pacientes, así como una colaboración más estrecha entre médicos y tecnología para mejorar los resultados de salud cutánea.

## 6.1. Trabajo futuro

En futuras investigaciones, se podrían explorar diversas áreas para mejorar la detección de enfermedades dermatológicas y obtener resultados satisfactorios:

- **Adaptar la aplicación a móvil:** Aumentaría su accesibilidad y utilidad para profesionales de la salud, pacientes o usuarios. Esto permitirá un acceso rápido y conveniente a la herramienta en cualquier lugar y momento, mejorando la detección temprana y el seguimiento de las enfermedades cutáneas
- **Validar los resultados en vivo con personal sanitario:** Aunque los resultados obtenidos son prometedores, es importante validar la precisión y eficacia del sistema en entornos clínicos reales. La colaboración con profesionales de la salud permitiría evaluar la efectividad del modelo en situaciones prácticas, identificar posibles desafíos y realizar ajustes necesarios para mejorar su desempeño en el mundo real
- **Estudiar otros datasets de imágenes que tengan más variables exógenas:** Puede enriquecer el análisis y mejorar la precisión del modelo, proporcionando una comprensión más completa de las enfermedades de la piel y mejorando su capacidad de generalización del modelo.
- **Funcionalidad de usuario y almacenamiento de imágenes:** Añadir funcionalidades adicionales para los usuarios, como la capacidad de crear perfiles individuales y almacenar imágenes de lesiones cutáneas junto con

información relevante, podría mejorar la experiencia del usuario y facilitar el seguimiento de las afecciones a lo largo del tiempo. Esto también podría permitir la creación de bases de datos más amplias y diversificadas para futuros análisis y mejoras del modelo.

## Bibliografía

- [1] R. A. Española, «Diccionario de la lengua española,» 13 Mayo 2023. [En línea]. Available: <https://dle.rae.es/morbimortalidad>.
- [2] K. Mader., «Kaggle,» [En línea]. Available: <https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000>.
- [3] M. Bara, «OBS Business School,» 4 Mayo 2017. [En línea]. Available: <https://www.obsbusiness.school/blog/7-estrategias-para-las-amenazas-y-oportunidades-en-proyectos>.
- [4] P. Poley Ceballos, G. M. González y E. C. Moreno Lamadrid, *Salud Coding - PROYECTO DELFOS - Sistema de apoyo a la decisión clínica.*, 2023.
- [5] MedlinePlus, «Enfermedades de la Piel,» 11 Mayo 2023. [En línea]. Available: <https://medlineplus.gov/spanish/skinconditions.html>.
- [6] E. F. y. J. Galceran, «Redacción médica,» 14 Julio 2022. [En línea]. Available: <https://www.redaccionmedica.com/secciones/dermatologia/la-incidencia-del-cancer-de-piel-en-espana-se-ha-duplicado-en-20-anos-1782>.
- [7] S. R. Department, «Statista,» 25 Marzo 2024. [En línea]. Available: <https://es.statista.com/estadisticas/590516/numero-de-muertes-por-cancer-de-piel-en-espana/>.
- [8] B. Splane, «Verywellhealth,» 2 Mayo 2023. [En línea]. Available: <https://www.verywellhealth.com/what-does-malignant-and-benign-mean-514240>.
- [9] S. Data., «National Library of Medicine, NIH,» 14 Agosto 2018. [En línea]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6091241/>.
- [10] «Wikipedia,» 27 Enero 2024. [En línea]. Available:

[https://en.wikipedia.org/wiki/Melanocytic\\_nevus](https://en.wikipedia.org/wiki/Melanocytic_nevus).

<https://www.sap.com/latinamerica/products/artificial-intelligence/what-is-machine-learning.html>.

- [20] «Jesuites educació en colaboración Universitat Oberta de Catalunya,» [En línea]. Available: <https://fp.uoc.fje.edu/blog/que-es-deep-learning-y-cuales-son-sus-aplicaciones/>.
- [21] «Unir, la universidad de Internet,» 10 06 2021. [En línea]. Available: <https://www.unir.net/ingenieria/revista/deep-learning/>.
- [22] «RedHat,» 3 Abril 2023. [En línea]. Available: <https://www.redhat.com/es/topics/digital-transformation/what-is-deep-learning>.
- [23] S. Ravindra, «HdfsTutorial,» [En línea]. Available: <https://hdfstutorial.com/blog/artificial-neural-networks-basics-guide/>.
- [24] «Cloudflare,» [En línea]. Available: <https://www.cloudflare.com/es-es/learning/ai/what-is-neural-network/>.
- [25] «ICHI.PRO,» 2020. [En línea]. Available: <https://ichi.pro/es/capas-de-redes-neuronales-129624486048658>.
- [26] «IBM,» [En línea]. Available: <https://www.ibm.com/es-es/topics/neural-networks>.
- [27] J. Finance, «Inteligencia-Artificial.dev,» 6 Enero 2021. [En línea]. Available: <https://inteligencia-artificial.dev/tipos-redes-neuronales/>.
- [28] «MathWorks,» [En línea]. Available: <https://es.mathworks.com/discovery/convolutional-neural-network.html>.
- [29] «DataScientest,» [En línea]. Available: <https://datascientest.com/es/convolutional-neural-network-es>.

- [30] Y. Musienko, «MEREHEAD,» 9 Noviembre 2022. [En línea]. Available: <https://merehead.com/es/blog/redes-neuronales-en-industria-salud/>.
- [31] «MathWorks,» [En línea]. Available: <https://es.mathworks.com/help/deeplearning/ug/pretrained-convolutional-neural-networks.html>.
- [32] H. e. al., «Paperswithcode,» [En línea]. Available: <https://paperswithcode.com/method/densenet#:~:text=A%20DenseNet%20is%20a%20type%20of%20convolutional%20neural,%28with%20matching%20feature-map%20sizes%29%20directly%20with%20each%20other..>
- [33] «ICHI.PRO,» [En línea]. Available: [https://ichi.pro/es/revision-densenet-red-convolucional-densa-clasificacion-de-imagenes-200536763594755#google\\_vignette](https://ichi.pro/es/revision-densenet-red-convolucional-densa-clasificacion-de-imagenes-200536763594755#google_vignette).
- [34] A. Ahmed, «Iq.opengenus,» [En línea]. Available: <https://iq.opengenus.org/architecture-of-densenet121/>.
- [35] «iq.opengenus,» [En línea]. Available: <https://iq.opengenus.org/mobilenetv2-architecture/>.
- [36] S. Mukherjee, «Medium - Towards Data Science,» 18 Agosto 2022. [En línea]. Available: <https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758>.
- [37] «Datagen,» [En línea]. Available: <https://datagen.tech/guides/computer-vision/resnet-50/>.
- [38] Y. Fu, «Keras,» 10 Julio 2023. [En línea]. Available: [https://keras.io/examples/vision/image\\_classification\\_efficientnet\\_fine\\_tuning/](https://keras.io/examples/vision/image_classification_efficientnet_fine_tuning/).
- [39] V. Agarwal, «Medium,» 24 Mayo 2020. [En línea]. Available: <https://towardsdatascience.com/complete-architectural-details-of-all-efficientnet-models-5fd5b736142>.

- [40] Z. Akhtar, «iq.opengenus,» [En línea]. Available: <https://iq.opengenus.org/xception-model/>.
- [41] F. Chollet, «The computer vision foundation,» 2017. [En línea]. Available: [/openaccess.thecvf.com/content\\_cvpr\\_2017/html/Chollet\\_Xception\\_Deep\\_Learning\\_CVPR\\_2017\\_paper.html](https://openaccess.thecvf.com/content_cvpr_2017/html/Chollet_Xception_Deep_Learning_CVPR_2017_paper.html).
- [42] J. F. López, «Economipedia,» 22 Junio 2019. [En línea]. Available: <https://economipedia.com/definiciones/variable-exogena.html>.
- [43] K. Bustinza Salazar, «Universidad Nacional de San Antonio Abad del Cusco . Repositorio institucional,» 2024. [En línea]. Available: <https://repositorio.unsaac.edu.pe/handle/20.500.12918/8648>.
- [44] «ScienceDirect,» 15 Abril 2024. [En línea]. Available: <https://www.sciencedirect.com/science/article/pii/S0957417423029986>.
- [45] «Aprendeprogramando,» [En línea]. Available: <https://aprendeprogramando.es/python-ventajas-y-desventajas/>.
- [46] A. Morales, «MappingGIS,» [En línea]. Available: <https://mappinggis.com/2022/10/como-instalar-anaconda-y-jupyter-notebook-para-ciencia-de-datos-y-gis/#:~:text=C%C3%B3mo%20instalar%20Anaconda%20y%20jupyter%20notebook%20para%20ciencia,de%20datos%20y%20GIS%C2%0es%20hacerlo%20mediante%20Anaconda..>
- [47] F. Flores, «OpenWebinars,» 22 Julio 2022. [En línea]. Available: <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/#qu%C3%A9-es-visual-studio-code>.
- [48] «SITIOBIGDATA.COM,» 19 01 2019. [En línea]. Available: <https://sitiobigdata.com/2019/01/19/machine-learning-metrica-clasificacion-parte-3/>.

- [49] «DataBitAI,» 17 Abril 2023. [En línea]. Available: [https://databitai.com/machine-learning/metricas-de-evaluacion-en-machine-learning/#Para\\_algoritmos\\_de\\_Clasicacion\\_generalmente\\_se\\_utilizan\\_las\\_siguientes\\_metricas](https://databitai.com/machine-learning/metricas-de-evaluacion-en-machine-learning/#Para_algoritmos_de_Clasicacion_generalmente_se_utilizan_las_siguientes_metricas).
- [50] H. N. B, «Medium,» 10 Diciembre 2019. [En línea]. Available: <https://medium.com/analytics-vidhya/confusion-matrix-accuracy-precision-recall-f1-score-ade299cf63cd>.
- [51] P. R. D. L. SANTOS, «Telefonicatech,» 13 Diciembre 2021. [En línea]. Available: <https://telefonicatech.com/blog/como-interpretar-la-matriz-de-confusion-ejemplo-practico>.
- [52] R. Kundu, «v7labs,» 16 Diciembre 2022. [En línea]. Available: <https://www.v7labs.com/blog/f1-score-guide#:~:text=How%20to%20calculate%20F1%20score%3F%201%20Macro-averaged%20F1,class-imbalanced%20data%20distribution.%20...%204%20F%CE%B2%20score%20>.
- [53] R. Díaz, «The Machine Learners,» [En línea]. Available: [https://www.themachinelearners.com/metricas-de-clasificacion/#F1\\_Score](https://www.themachinelearners.com/metricas-de-clasificacion/#F1_Score).
- [54] ITCWEBS, «ITC Web Solutions,» 6 Enero 2024. [En línea]. Available: <https://itcwebsolutions.com/pruebas-y-optimizacion/la-importancia-de-las-pruebas-en-el-desarrollo-de-sitios-web-y-aplicaciones/>.
- [55] Affde, «La Importancia De Los Requisitos Funcionales Y No Funcionales,» 25 Junio 2021. [En línea]. Available: <https://www.affde.com/es/the-importance-of-functional-and-non-functional-requirements-in-software-development.html#:~:text=Dos%20categor%C3%ADas%20de%20requisitos%20Los%20requisitos%20funcionales,propiedades%20y%20caracter%C3%ADsticas%20tiene%20un%20d>.