



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

— **TELECOM** ESCUELA  
TÉCNICA **VLC** SUPERIOR  
DE INGENIERÍA DE  
TELECOMUNICACIÓN

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería de  
Telecomunicación

Diseño y desarrollo de algoritmos de Deep Learning para la  
segmentación de tejidos en la detección de la Colitis  
Ulcerosa.

Trabajo Fin de Grado

Grado en Ingeniería de Tecnologías y Servicios de  
Telecomunicación

AUTOR/A: López Tudela, Josep

Tutor/a: Amor del Amor, María Rocío del

Cotutor/a: Meseguer Esbrí, Pablo

CURSO ACADÉMICO: 2023/2024



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

— **TELECOM** ESCUELA  
TÉCNICA **VLC** SUPERIOR  
DE INGENIERÍA DE  
TELECOMUNICACIÓN

Escuela Técnica Superior de Ingeniería de Telecomunicación  
Universitat Politècnica de València  
Edificio 4D. Camino de Vera, s/n, 46022 Valencia  
Tel. +34 96 387 71 90, ext. 77190  
[www.etsit.upv.es](http://www.etsit.upv.es)

**VLC/**  
**CAMPUS**  
VALENCIA, INTERNATIONAL  
CAMPUS OF EXCELLENCE





## Resumen

La colitis ulcerosa (CU) es una enfermedad inflamatoria intestinal que provoca inflamación y formación de úlceras en el tracto digestivo. Afecta al recubrimiento más profundo del intestino grueso, también denominado colon, y al recto. Es importante el diagnóstico precoz mediante la toma de biopsias para evitar lesiones en el aparato digestivo y mejorar la respuesta a los tratamientos.

Recientemente, se ha propuesto una puntuación llamada Paddington International virtual ChromoendoScopy ScOre (PiCaSSO) y el Histologic Remission Index (PHRI). El índice PHRI evalúa la presencia de neutrófilos en cuatro componentes histológicos específicos para determinar la actividad de la enfermedad, la cicatrización mucosa y el resultado clínico en la histopatología de la CU. Se analiza la infiltración de neutrófilos en regiones específicas, como la lámina propia, el epitelio superficial, el epitelio criptal y el lumen criptal.

La detección de neutrófilos es una tarea compleja, por lo que se opta por segmentar primero el tejido para identificar las áreas donde podrían encontrarse neutrófilos, y luego detectar los neutrófilos en esas áreas. Este estudio se centra en el desarrollo y diseño de algoritmos de segmentación de tejidos utilizando distintos modelos basados en redes neuronales convolucionales especializadas en imágenes. En la primera fase, nos enfocaremos en encontrar el mejor modelo de segmentación de tejidos utilizando la arquitectura U-Net. Para ello, realizaremos un análisis de ablación de la arquitectura para identificar las características más relevantes y evaluar su impacto en el rendimiento del modelo. Posteriormente, compararemos los resultados para determinar la configuración que mejor se adapta a nuestro problema específico. En la segunda fase, exploraremos otras arquitecturas de redes neuronales, aplicando el mismo proceso de evaluación y comparación utilizado en la primera fase. Esto nos permitirá comparar el rendimiento de diferentes enfoques y seleccionar la arquitectura más adecuada para la segmentación de tejidos en imágenes médicas.

**Palabras clave:** Segmentación semántica, PHRI, Deep Learning, Redes convolucionales.



## Resum

La colitis ulcerosa (CU) és una malaltia inflamatòria intestinal que provoca inflamació i formació d'úlceres en el tracte digestiu. Afecta al recobriment més profund de l'intestí gros, també denominat còlon, i al recte. És important el diagnòstic precoç mitjançant la presa de biòpsies per a evitar lesions en l'aparell digestiu i millorar la resposta als tractaments.

Recentment, s'ha proposat una puntuació anomenada Paddington International virtual ChromoendoScopy ScOre (Picasso) i el Histologic Remission Index (PHRI). L'índex PHRI avalua la presència de neutròfils en quatre components histològics específics per a determinar l'activitat de la malaltia, la cicatrització mucosa i el resultat clínic en la histopatologia de la CU. S'analitza la infiltració de neutròfils en regions específiques, com la làmina pròpia, l'epiteli superficial, l'epiteli criptal i el lumen criptal.

La detecció de neutròfils és una tasca complexa, per la qual cosa s'opta per segmentar primer el teixit per a identificar les àrees on podrien trobar-se neutròfils, i després detectar els neutròfils en eixes àrees. Aquest estudi se centra en el desenvolupament i disseny d'algoritmes de segmentació de teixits utilitzant diferents models basats en xarxes neuronals convolucionals especialitzades en imatges. En la primera fase, ens enfocarem a trobar el millor model de segmentació de teixits utilitzant l'arquitectura U-Net. Per a això, realitzarem una anàlisi d'ablació de l'arquitectura per a identificar les característiques més rellevants i avaluar el seu impacte en el rendiment del model. Posteriorment, compararem els resultats per a determinar la configuració que millor s'adapta al nostre problema específic. En la segona fase, explorarem altres arquitectures de xarxes neuronals, aplicant el mateix procés d'avaluació i comparació utilitzat en la primera fase. Això ens permetrà comparar el rendiment de diferents enfocaments i seleccionar l'arquitectura més adequada per a la segmentació de teixits en imatges mèdiques.

**Paraules clau:** Segmentació semàntica, PHRI, Deep Learning, Xarxes convolucionals.



## Abstract

Ulcerative colitis (UC) is an inflammatory bowel disease that causes inflammation and the formation of ulcers in the digestive tract. It affects the innermost lining of the large intestine, also known as the colon, and the rectum. Early diagnosis through biopsies is crucial to prevent damage to the digestive system and to improve treatment response.

Recently, a scoring system called the Paddington International virtual ChromoendoScopy ScOre (PiCaSSO) and the Histologic Remission Index (PHRI) has been studied. The PHRI index evaluates the presence of neutrophils in four specific histological components to determine disease activity, mucosal healing, and clinical outcome in the histopathology of UC. Neutrophil infiltration is analyzed in specific regions such as the lamina propria, surface epithelium, crypt epithelium, and crypt lumen.

Detecting neutrophils is a complex task, so the tissue is segmented first to identify areas where neutrophils might be present, and then neutrophils are detected in those areas. This study focuses on the development and design of tissue segmentation algorithms using various models based on convolutional neural networks specialized in imaging. In the first phase, we will focus on finding the best tissue segmentation model using the U-Net architecture. To do this, we will perform an ablation analysis of the architecture to identify the most relevant features and assess their impact on the model's performance. We will then compare the results to determine the configuration that best suits our specific problem. In the second phase, we will explore other neural network architectures, applying the same evaluation and comparison process used in the first phase. This will allow us to compare the performance of different approaches and select the most suitable architecture for tissue segmentation in medical images.

**Keywords:** Semantic segmentation, PHRI, Deep Learning, Convolutional networks.



## **Agradecimientos**

Quería expresar mi agradecimiento al equipo de CVB Lab, especialmente a Rocío, Pablo y Valery, por brindarme la oportunidad de trabajar en un proyecto tan significativo. A mis compañeros de clase, quienes han hecho que la carrera sea más amena. A mis amigos y amigas de siempre, por su apoyo en los momentos más difíciles. Y, especialmente, a mi familia, que siempre está a mi lado y ha confiado en mí en todo momento.

## RESUMEN EJECUTIVO

La memoria del TFG del Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación debe desarrollar en el texto los siguientes conceptos, debidamente justificados y discutidos, centrados en el ámbito de la ingeniería de telecomunicación

CONCEPT (ABET)	CONCEPTO (traducción)	¿Cumple? (S/N)	¿Dónde? (páginas)
1. IDENTIFY:	1. IDENTIFICAR:		
1.1. Problem statement and opportunity	1.1. Planteamiento del problema y oportunidad	S	8-12
1.2. Constraints (standards, codes, needs, requirements & specifications)	1.2. Toma en consideración de los condicionantes (normas técnicas y regulación, necesidades, requisitos y especificaciones)	S	8-32
1.3. Setting of goals	1.3. Establecimiento de objetivos	S	10
2. FORMULATE:	2. FORMULAR:		
2.1. Creative solution generation (analysis)	2.1. Generación de soluciones creativas (análisis)	S	32-40
2.2. Evaluation of multiple solutions and decision-making (synthesis)	2.2. Evaluación de múltiples soluciones y toma de decisiones (síntesis)	S	40-55
3. SOLVE:	3. RESOLVER:		
3.1. Fulfilment of goals	3.1. Evaluación del cumplimiento de objetivos	S	40-55
3.2. Overall impact and significance (contributions and practical recommendations)	3.2. Evaluación del impacto global y alcance (contribuciones y recomendaciones prácticas)	S	60-61



## Lista de Acrónimos

<b>IA</b>	Inteligencia Artificial
<b>CU</b>	Colitis Ulcerosa
<b>CNN</b>	Red Neuronal Convolutacional ( <i>Convolutional Neural Network</i> )
<b>CAD</b>	Diagnostico Asistido por Computadora ( <i>Computer Aided Diagnosis</i> )
<b>WSI</b>	Imágenes Histológicas Completas ( <i>Whole Slide Images</i> )
<b>PHRI</b>	Índice de Remisión Histológica PICaSSO
<b>EVC</b>	Cromoendoscopia Virtual Endoscopica ( <i>Endoscopic Virtual Chromoendoscopy</i> )
<b>EII</b>	Enfermedades Inflamatorias Intestinales
<b>ML</b>	Aprendizaje Automático ( <i>Machine Learning</i> )
<b>DL</b>	Aprendizaje Profundo ( <i>Deep Learning</i> )
<b>BN</b>	Normalización por lotes ( <i>Batch Normalization</i> )



## Índice General

### Memoria

Capítulo 1.	Introducción .....	8
1.1	Descripción del problema .....	8
1.2	Contexto y relación con el proyecto PICaSSO .....	9
1.3	Objetivos del proyecto .....	10
Capítulo 2.	Fundamentos Teóricos .....	12
2.1	Enfermedades Inflammatorias Intestinales .....	12
2.2	Colitis Ulcerosa.....	12
2.2.1	Síntomas, Estados y Tratamiento de la Colitis Ulcerosa .....	12
2.2.2	Índices de evaluación y métodos de diagnóstico .....	14
2.3	Inteligencia Artificial.....	19
2.3.1	Aprendizaje profundo.....	20
2.3.2	Redes neuronales convolucionales .....	22
Capítulo 3.	Materiales.....	27
3.1	Base de datos de WSI .....	27
3.2	Software.....	29
3.3	Hardware.....	30
Capítulo 4.	Metodología .....	32
4.1	Arquitecturas.....	32
4.1.1	Definición U-Net.....	32
4.1.2	Implementación U-Net .....	33
4.2	Otras arquitecturas de segmentación .....	34
4.2.1	U-Net 2.0.....	35
4.2.2	U-Net Plus Plus .....	36
4.2.3	Feature Pyramid.....	36
Capítulo 5.	Resultados .....	40
5.1	Métricas de Evaluación.....	40
5.2	Configuración Experimental.....	42
5.3	Parámetros de entrenamiento U-Net.....	43
5.3.1	Evaluación y Ajuste de la Arquitectura del Modelo.....	48
5.4	Resultados otras arquitecturas.....	53
5.4.1	U-Net 2.0.....	53
5.4.2	U-Net Plus Plus .....	54
5.4.3	Feature Pyramid Network (FPN).....	54



5.4.4	Comparativa y Análisis de los Resultados .....	55
Capítulo 6.	Conclusiones y futuras líneas.....	60
6.1	Conclusiones.....	60
6.2	Futuras líneas .....	61
Capítulo 7.	Bibliografía .....	63

## Anexos

A.	Impacto del proyecto en los objetivos de la Agenda 2030 .....	65
----	---	----

## Lista de Figuras

1.1	Partes del intestino grueso. Obtenido de <a href="#">[3]</a> . ....	12
1.2	Diferentes estados de la CU. Proctitis ulcerosa, colitis del lado izquierdo y pancolitis. Obtenido de <a href="#">[2]</a> . ....	14
1.3	Imagen de una colonoscopia de un paciente con colitis ulcerosa moderada-grave. Paciente con CU extensa moderada-grave (izquierda); curación mucosa tras tratamiento. Obtenido de <a href="#">[4]</a> . ....	14
1.4	(A) Imagen endoscópica de mucosa de sigmoides aspecto normal (score Mayo 0). (B) Biopsia mismo paciente mismo paciente mostrando infiltrado inflamatorio en la lámina propia con presencia de neutrófilos. Presencia de neutrófilos en la cripta (score Geobes 3.1) (C) Mucosa de recto con eritema leve (score Mayo 1). (D) Biopsia del mismo paciente mostrando infiltrado de inflamatorio de neutrófilos y absceso de cripta (score Geobes 3.1). Obtenido de <a href="#">[6]</a> . ....	15
1.5	Glóbulos blancos granulocitos: (A) Neutrófilo, (B) Basófilo, (C) Eosinófilo. Obtenido de <a href="#">[11]</a> . ....	16
2.1	Arquitectura de una red neuronal. Obtenida de <a href="#">[13]</a> . ....	20
2.2	Operaciones matemáticas realizadas en las neuronas artificiales. Obtenida de <a href="#">[13]</a> . ....	21
2.3	Funciones de activación. Obtenida de <a href="#">[12]</a> . ....	22
2.4	Estructura capa convolucional. Obtenido de <a href="#">[13]</a> . ....	23
2.5	Proceso obtención del mapa de características con el filtro o kernel. Obtenido de <a href="#">[14]</a> . ....	23
2.6	Funcionamiento capa agrupación. Obtenido de <a href="#">[13]</a> . ....	24
3.1	Imagen WSI del tejido del colon con CU <a href="#">[9]</a> . ....	27
3.2	Imagen WSI anotada por patólogos <a href="#">[9]</a> . ....	28
3.2.1	Código de colores imagen WSI. ....	28
4.1	Arquitectura U-Net: (A) Codificador, (B) Decodificadores. Obtenido de <a href="#">[17]</a> . ....	33
4.2	Estructura U-Net con encoder preentrenado. Obtenido de <a href="#">[16]</a> . ....	35
4.3	Estructura modelo U-Net++. Obtenida de <a href="#">[18]</a> . ....	36
4.4	Estructura modelo FPN. Obtenida de <a href="#">[19]</a> . ....	37
5.1	Grafica con métricas validación con lr 1e-4. Elaboración propia. ....	45
5.2	Grafica con métricas validación con lr 5e-4. Elaboración propia. ....	45
5.3	Grafica con métricas validación con lr 1e-6. Elaboración propia. ....	46
5.4	Grafica con métricas validación con 128 canales. Elaboración propia. ....	47
5.5	Grafica con métricas validación con 32 canales. Elaboración propia. ....	47
5.6	Grafica con métricas validación con input de 1024x1024 32 canales. Elaboración propia. ....	48
5.7	Graficas entrenamiento modelo 2. Elaboración propia. ....	49
5.8	Graficas de entrenamiento modelo 3. Elaboración propia. ....	49



5.9	Graficas de entrenamiento modelo 4. Elaboración propia. ....	50
5.10	Graficas de entrenamiento modelo 5. Elaboración propia. ....	50
5.10.1	WSI's anotadas por patólogos. ....	51
5.11	Imagen anotada por el Modelo 2. Elaboración propia. ....	52
5.12	Imagen anotada por el Modelo 4. Elaboración propia. ....	52
5.13	Imagen WSIs del tejido. Derecha imagen anotada por el Modelo U-Net. Izquierda imagen anotada por patólogos. <a href="#">[9]</a> . ....	56
5.14	Imagen WSIs del tejido. Derecha imagen anotada por el Modelo U-Net 2.0. Izquierda imagen anotada por patólogos. <a href="#">[9]</a> . ....	56
5.15	Imagen WSIs del tejido. Izquierda imagen anotada por el Modelo U-Net PlusPlus libreria. Derecha imagen anotada por patólogos. <a href="#">[9]</a> . ....	57
5.16	Imagen WSIs del tejido. Izquierda imagen anotada por el Modelo FPN libreria. Derecha imagen anotada por patólogos. <a href="#">[9]</a> . ....	58



## Lista de Tablas

1.1	Valoraciones de la CU utilizando el índice de mayo (MES). Obtenido de [7].	16
1.2	Criterios de evaluación del índice PHRI. Obtenida de [9].	17
1.3	Valores de las métricas de evaluación de entrenamiento cuantitativas en la época más óptima del modelo. Lr: Tasa de aprendizaje, PE: Perdida de entrenamiento, PV: Perdida de validación, DE: Dice de entrenamiento, DV: Dice de validación.	44
1.4	Valores de las métricas de evaluación de entrenamiento cuantitativas en la época óptima del modelo. NF: Numero de filtros, PE: Perdida de entrenamiento, PV: Perdida de validación, DE: Dice de entrenamiento, DV: Dice de validación.	46
1.5	Valores de las métricas de evaluación de entrenamiento cuantitativas en la época óptima del modelo. TE: Tamaño de entrada, PE: Perdida de entrenamiento, PV: Perdida de validación, DE: Dice de entrenamiento, DV: Dice de validación.	47
1.6	Valores de la estructura de cada Modelo evaluado. Modelos: Modelos más relevantes, BC: Bloques de codificación, BD: Bloques de decodificación, CC: Capas convolucionales por bloque, NF: Numero de filtros, TE: Tamaño de entrada.	48
1.7	Resultados obtenidos de los modelos óptimos. Métricas de Cat Dice y IOU entre 0 y 1.	51
1.8	Resultados de test de los modelos con sus mejores estructuras, evaluados mediante las métricas IOU y Categorical Dice.	53
1.9	Resultados de test de los modelos U-Net++ con sus mejores estructuras, evaluados mediante las métricas IOU y Categorical Dice.	54
1.10	Resultados de test de los modelos FPN con sus mejores estructuras, evaluados mediante las métricas IOU y Categorical Dice.	54
1.11	Configuración Modelo5_FPN. TC: Tamaño del codificador, F: Numero de filtros inicial, NF: Numero de Filtros utilizados por cada capa, CD: Combinación de datos, D: Dropout, TE: Tamaño de entrada, Lr: Tasa de aprendizaje.	55
1.12	Resultados de métricas de segmentación para los mejores modelos obtenidos.	55



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

— **TELECOM** ESCUELA  
TÉCNICA **VLC** SUPERIOR  
DE INGENIERÍA DE  
TELECOMUNICACIÓN

## Capítulo 1. Introducción

### 1.1 Descripción del problema

La colitis ulcerosa (CU) es una enfermedad inflamatoria crónica y autoinmune que afecta principalmente al colon, incluyendo el recto y el ano. El número de personas afectadas por CU está en aumento, con una media de 24.3 casos por cada 100,000 habitantes a nivel europeo [1]. Este incremento en los casos resalta la necesidad urgente de mejorar los métodos de diagnóstico y tratamiento para una gestión eficaz de la enfermedad.

La enfermedad se presenta de manera continua, comenzando en el recto y extendiéndose hacia el colon en distintos grados según el paciente. Se caracteriza por fases de actividad inflamatoria (recaídas) y remisión (sin actividad), lo que implica que los pacientes pueden experimentar largos periodos sin síntomas y otros con síntomas activos.

Para diagnosticar la CU, se realiza una colonoscopia, un procedimiento clínico que permite observar el tracto intestinal. Si se encuentra algún hallazgo, se extraen muestras de tejido del colon mediante un endoscopio. Estas muestras son analizadas posteriormente por patólogos, quienes examinan y detectan los patrones característicos de la enfermedad en el tejido.

Gracias a los avances de las tecnologías digitales en el campo de la histopatología, estas muestras se digitalizan como imágenes histológicas completas, conocidas en inglés como WSIs (*Whole Slide Images*). Estas imágenes son de tamaño gigapíxel y han sido fundamentales para el desarrollo e implementación de algoritmos y métricas para la detección de la CU.

Recientemente, se ha estudiado una puntuación denominada Puntuación de Cromoendoscopia Virtual Internacional de Paddington (PICaSSO) y un índice llamado Índice de Remisión Histológica (PHRI), los cuales se explicarán en detalle más adelante. El índice PHRI ha mostrado excelentes resultados en la detección de la CU y se enfoca en identificar neutrófilos para evaluar la actividad de la enfermedad, la curación de la mucosa y el resultado clínico en la histología de la CU. PHRI se utiliza para analizar la infiltración de neutrófilos en regiones específicas del tejido. Detectar neutrófilos no es una tarea sencilla, por lo que se aborda el problema en dos etapas: primero, segmentar el tejido en áreas donde pueda existir la presencia de neutrófilos y, luego, llevar a cabo la detección de neutrófilos en la zona segmentada. [9]

Este proyecto se enfoca en desarrollar un algoritmo capaz de segmentar las áreas de interés en los tejidos. En la primera fase, se investigará la arquitectura U-Net, realizando un análisis de ablación para identificar las características más relevantes y evaluar su impacto en la precisión del modelo de segmentación. Tras este análisis, se optimizará la arquitectura ajustando su estructura y los hiperparámetros de entrenamiento para mejorar su rendimiento. En la segunda fase, se explorarán otras arquitecturas de aprendizaje profundo utilizando el mismo proceso de evaluación y comparación aplicado a la U-Net, con el fin de seleccionar la arquitectura más adecuada para la segmentación precisa de tejidos.



## 1.2 Contexto y relación con el proyecto PICaSSO

Este trabajo es parte de un proyecto de investigación coordinado por la Universidad de Birmingham. PICASSO es el nombre más común utilizado para este proyecto por el desarrollo de una puntuación endoscópica para la colitis ulcerosa (CU) llamada “Paddington International virtual ChromoendoScopyScore”.

El proyecto de investigación PICASSO se creó con el objetivo de desarrollar un sistema integrado para la gestión de pacientes con colitis ulcerosa (CU) que hayan sido sometidos a una endoscopia y un análisis histológico. En este contexto, utilizando un conjunto de 614 biopsias de 307 pacientes con CU inscritos en un estudio multicéntrico prospectivo, desarrollaron y validaron una nueva puntuación de cromoendoscopia virtual electrónica (EVC) llamada PICaSSO para la clasificación endoscópica de la CU. PICaSSO mostró muy buena concordancia entre observadores, esta puntuación se puede utilizar para definir los hallazgos endoscópicos de la cicatrización mucosa y vascular en la CU, y refleja todo el espectro de cambios histológicos. [\[9\]](#)

También han desarrollado un índice de remisión histológica que muestra una fuerte correlación con las puntuaciones endoscópicas (Puntuación de Mayo e Índice de Severidad Endoscópica de CU y PICaSSO), así como con los resultados clínicos (hospitalización, colectomía y cambios en la terapia médica debido a brotes de CU), llamado PHRI. Este índice evalúa la presencia de neutrófilos en regiones específicas. Dado que es más simplificado, resulta accesible para su uso en Sistemas de diagnóstico asistido por computadora (CAD) basados en aprendizaje profundo.

El Computer Vision and Behaviour Analysis Lab (CVBLab) de la Universidad Politécnica de Valencia (UPV) colabora en el proyecto con el diseño y desarrollo de algoritmos automáticos basados en aprendizaje profundo, para el análisis de las imágenes WSI de pacientes con colitis ulcerosa (CU).



### 1.3 Objetivos del proyecto

El objetivo principal del proyecto es diseñar y conseguir desarrollar un algoritmo de inteligencia artificial basado en aprendizaje profundo capaz de segmentar y predecir las áreas de interés que incluyen lámina propia, el epitelio superficial, el epitelio criptal y el lumen criptal, en las imágenes WSI de pacientes con CU. Para poder alcanzar el objetivo principal planteado, también tenemos algunos objetivos secundarios:

1. Investigar y analizar diversos modelos y técnicas de segmentación de imágenes explorando su posible adaptación en nuestro proyecto. Esto incluirá la revisión de literatura científica y evaluación de algoritmos existentes.
2. Obtener las imágenes WSI junto con sus anotaciones por parte de patólogos en las áreas de interés necesarias para nuestro proyecto. Estas imágenes son proporcionadas por el grupo de investigación PICaSSO y provienen de varios centros internacionales.
3. Implementar los modelos y técnicas investigadas en el desarrollo del algoritmo utilizando nuestra red neuronal convolucional para la segmentación de las áreas de interés de nuestras imágenes WSI
4. Realizar experimentos con cada modelo para optimizar los hiperparámetros y la arquitectura de nuestra red neuronal y obtener los mejores resultados posibles.
5. Comparar los resultados obtenidos de cada experimento utilizando métricas en segmentación de imágenes, con el fin de seleccionar el modelo óptimo para nuestra tarea.
6. Proponer mejoras para futuras líneas de investigación basadas en las limitaciones y desafíos observados durante el desarrollo del proyecto.



## Capítulo 2. Fundamentos Teóricos

### 2.1 Enfermedades Inflammatorias Intestinales

Las Enfermedades Inflammatorias Intestinales (EII) es una enfermedad crónica de causa desconocida que se caracteriza por la inflamación del colon y/o intestino delgado, presentándose en forma de brotes. En España, se estima que hay entre 87 y 110 [20] personas afectadas por enfermedad inflamatoria intestinal por cada 100.000 habitantes.

Las enfermedades inflamatorias intestinales (EII) comprenden dos tipos de trastornos: colitis ulcerosa (CU) y la enfermedad de Crohn (EC). Estas enfermedades se caracterizan por la inflamación crónica e incontrolada de la mucosa intestinal, afectando a cualquier parte del tracto gastrointestinal. Sin embargo, la inflamación crónica también puede ser una característica del intestino normal. El intestino se inflama en respuesta a patógenos potenciales y luego vuelve a un estado de tolerancia una vez que el patógeno es eliminado. Lo que distingue la EII de las respuestas inflamatorias observadas en el intestino normal es la incapacidad de regular a la baja esas respuestas; el sistema inmunitario mucoso permanece crónicamente activado y el intestino permanece crónicamente inflamado.

Los episodios sintomáticos de esta enfermedad crónica se alternan en fases de actividad inflamatoria (brotes) y episodios de remisión (sin actividad). La frecuencia y a la gravedad de los brotes son muy variables e impredecibles.

### 2.2 Colitis Ulcerosa

#### 2.2.1 Síntomas, Estados y Tratamiento de la Colitis Ulcerosa

La colitis ulcerosa es una enfermedad crónica que afecta la mucosa del colon (intestino grueso), incluyendo el recto y el ano, así como otras partes del colon, como el sigmoides, el colon descendente, el colon transverso y el colon ascendente, como se observa en la Figura 1.1. Esta condición daña el revestimiento interno del tejido intestinal. Siempre comienza en el ano y se extiende, en mayor o menor medida, hacia el resto del colon, dependiendo de cada paciente. Se presenta más comúnmente con sangre en las heces y diarrea. Hasta el 15% de los pacientes pueden presentar inicialmente una enfermedad grave [3].

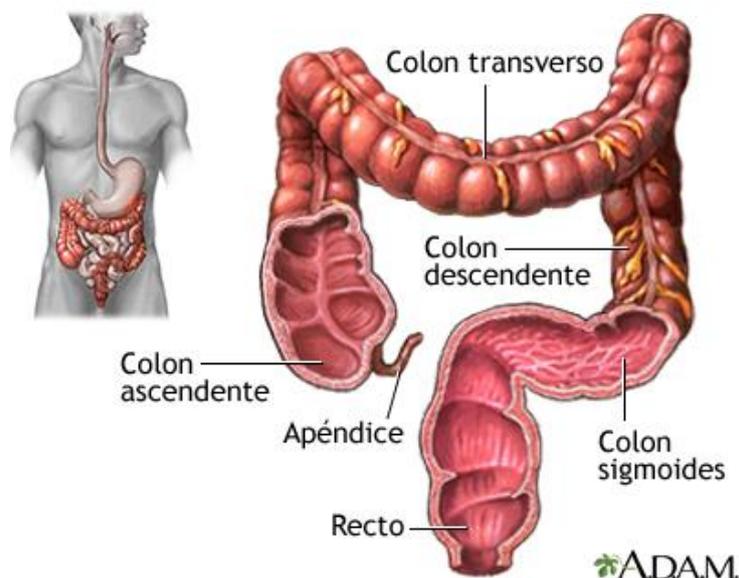


Figura 1.1: Partes del intestino grueso. Obtenido de [3].

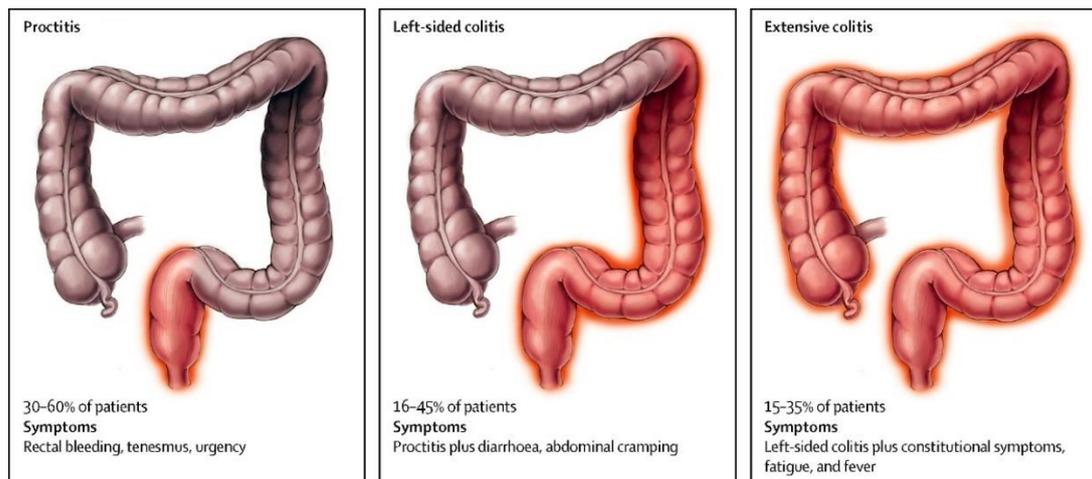
Los síntomas más frecuentes son los siguientes:

- **Diarrea con moco y sangre, puede haber estreñimiento:** Es uno de los síntomas más característicos de la colitis ulcerosa. La diarrea puede ser frecuente y urgente, puede contener moco y sangre debido a la inflamación y ulceración de la mucosa intestinal. Aunque la diarrea es común, en algunos casos dependiendo del estado de la enfermedad, puede haber episodios de estreñimiento, especialmente en aquellos con proctitis ulcerosa, de la cual hablaremos más adelante.
- **Dolor abdominal:** Suele localizarse en la parte baja del abdomen o en el lado izquierdo y puede variar desde molestias leves hasta cólicos severos.
- **Pérdida de peso:** Dada la inflamación crónica y la presencia de diarrea, la absorción de nutrientes se ve considerablemente comprometida, resultando en una pérdida de peso involuntaria.
- **Anemia:** La pérdida de sangre continua en las heces puede causar anemia.
- **Fiebre:** Durante los brotes agudos, puede aparecer fiebre, lo cual indica una respuesta inflamatoria significativa del cuerpo.

La variedad de síntomas mencionados anteriormente refleja cómo se manifiesta la CU en sus diferentes estados. Los estados de la enfermedad son los siguientes:

- **Proctitis ulcerosa:** La inflamación está limitada al recto. Los síntomas pueden incluir urgencia (necesidad repentina e intensa de defecar) y tenesmo (sensación de evacuación incompleta), así como sangrado rectal.
- **Colitis del lado izquierdo:** La inflamación afecta el recto, el colon sigmoide y el colon descendente. Los síntomas incluyen diarrea conteniendo sangre, dolor en el lado izquierdo del abdomen, pérdida de peso y fatiga.
- **Pancolitis:** La inflamación se extiende por todo el colon. Los síntomas son más graves e incluyen diarrea severa con sangre, dolor abdominal, fiebre y pérdida de peso significativa.
- **Colitis fulminante:** Es una forma rara y extremadamente grave de colitis ulcerosa. Los síntomas incluyen todos los mencionados anteriormente y en algunos casos, megacolon tóxico, que es una emergencia médica y puede requerir cirugía inmediata.

En la **Figura 1.2** se observan las distintas inflamaciones de la enfermedad y el número de pacientes afectados en porcentajes, además de los síntomas mencionados anteriormente. Se muestra claramente cómo la inflamación se incrementa en todo el colon y cómo se reduce el número de pacientes afectados en cada estado de la enfermedad.

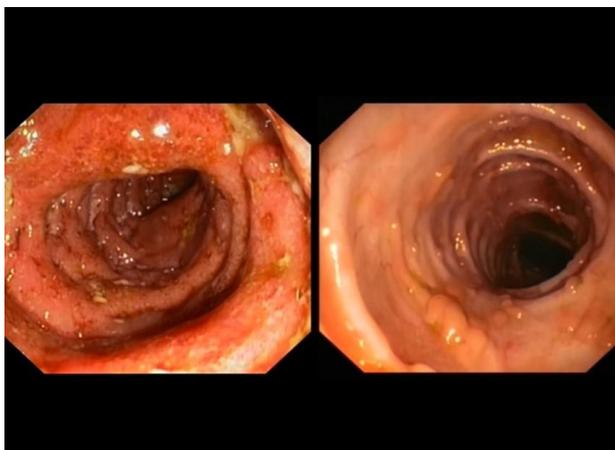


**Figura 1.2:** Diferentes estados de la CU. Proctitis ulcerosa, colitis del lado izquierdo y pancolitis. Obtenido de [2].

Como se comentó anteriormente, actualmente no existe cura para la enfermedad de la CU. Sin embargo, existen tratamientos disponibles que han demostrado producir cambios a corto y largo plazo en la evolución de la enfermedad. Estos tratamientos tienen como objetivo controlar la actividad de la enfermedad, favorecer el estado de remisión y ayudar a evitar las complicaciones. Existen una gran variedad de tratamientos disponibles según la gravedad del brote. Sin embargo, cada caso de la enfermedad es único entre los pacientes afectados. La variabilidad en la respuesta a los tratamientos se debe a factores individuales como la genética, el microbiota intestinal, el grado y la extensión de la inflamación. Debido a estas diferencias, no hay un tratamiento único que sea efectivo para todos los pacientes. Por esta razón, una detección precoz es fundamental para personalizar el tratamiento y mejorar los resultados. [1].

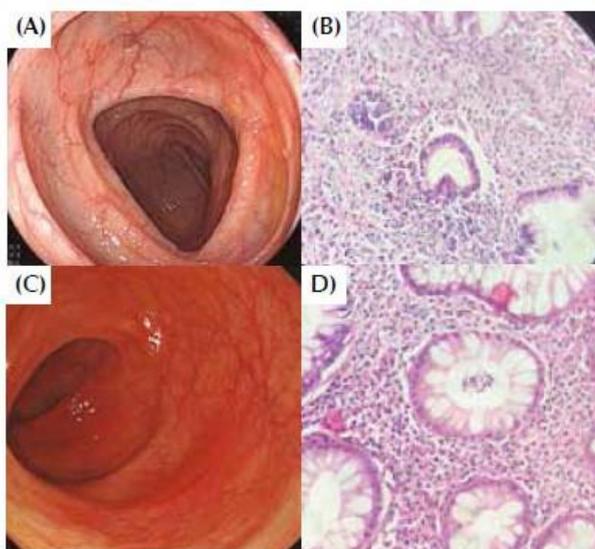
### 2.2.2 Índices de evaluación y métodos de diagnóstico

Para la detección de CU se utilizan distintos métodos como el análisis de heces y el análisis de sangre. La colonoscopia, que proporciona una visión completa del colon como se observa en la **Figura 1.3**, y la sigmoidoscopia, que se centra en el colon sigmoide y el recto, son procedimientos clave para confirmar el diagnóstico de la CU. Este examen permite visualizar las estructuras internas del intestino grueso mediante el uso de una sonda de fibra óptica flexible, conocida como endoscopio. Los endoscopios son dispositivos flexibles que están equipados con una pequeña cámara y una fuente de iluminación, lo que permite al médico obtener una visualización detallada de las mucosas del tubo digestivo.



**Figura 1.3:** Imagen de una colonoscopia de un paciente con colitis ulcerosa moderada-grave. Paciente con CU extensa moderada-grave (izquierda); curación mucosa tras tratamiento. Obtenido de [4].

Durante el procedimiento de una colonoscopia, los médicos pueden usar la endoscopia para realizar biopsias. Una biopsia es un procedimiento que se realiza para extraer una muestra de tejido o de células del cuerpo para su análisis. [5] Posteriormente, estas muestras de tejido son procesadas en un laboratorio de histopatología, donde se preparan cortes delgados del tejido que se tiñen con colorantes específicos. Una vez teñidas, las muestras se examinan bajo un microscopio por un patólogo para evaluar la estructura celular y tisular del tejido. De esta manera, el patólogo puede determinar la presencia de inflamación, úlceras, erosiones u otras características relevantes para el diagnóstico de la CU.



**Figura 1.4:** (A) Imagen endoscópica de mucosa de sigmoides aspecto normal (score Mayo 0). (B) Biopsia mismo paciente mismo paciente mostrando infiltrado inflamatorio en la lámina propia con presencia de neutrófilos. Presencia de neutrófilos en la cripta (score Geobes 3.1) (C) Mucosa de recto con eritema leve (score Mayo 1). (D) Biopsia del mismo paciente mostrando infiltrado de inflamatorio de neutrófilos y absceso de cripta (score Geobes 3.1). Obtenido de [6].

Las evaluaciones endoscópicas e histológicas son fundamentales para diagnosticar la enfermedad, evaluar su gravedad y monitorear la respuesta al tratamiento. Los métodos mencionados proporcionan información detallada sobre el estado del colon y la actividad inflamatoria. Estas evaluaciones se basan en índices específicos que permiten estandarizar los hallazgos clínicos y patológicos. Algunos de los índices utilizados son los siguientes:

1. **Índice endoscópico de Mayo (Mayo Endoscopic Score MES):** Es una herramienta de diagnóstico utilizada para evaluar la actividad de la CU. Este índice se centra en cuatro categorías: frecuencia de las deposiciones, sangrado rectal, mucosa y evaluación global del médico, valorando estos datos en una escala de 0 a 3 (Tabla 1.1). [7]
2. **Índice histológico Robarts (Robarts Histopathology Score RHI):** El índice Robarts se desarrolló utilizando diferentes elementos histológicos del índice de Geboes. Evalúa la actividad histológica de la colitis ulcerosa dividiéndose en cuatro componentes: infiltrado inflamatorio crónico, neutrófilos en la lámina propia, neutrófilos en el epitelio y erosiones/úlceras. Cada componente se gradúa de 0 a 3 y se pondera para calcular el RHI, con una escala de 0 a 3. [8]

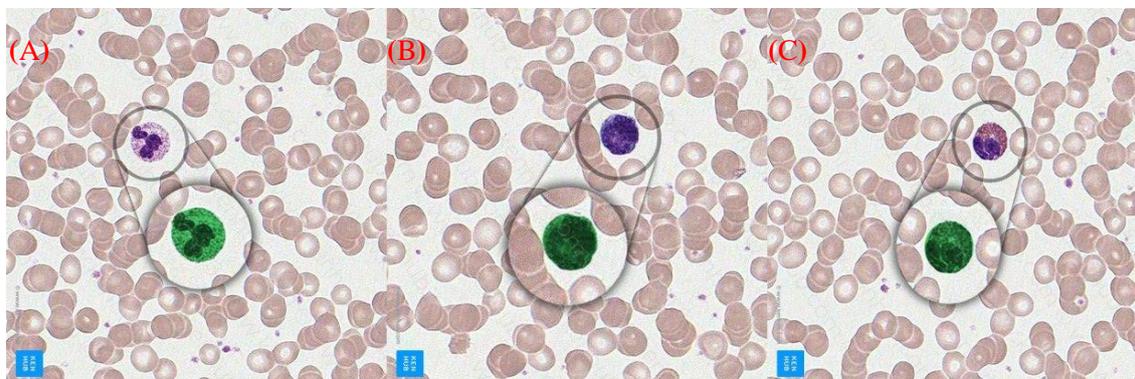
**Tabla 1.1:** Valoraciones de la CU utilizando el índice de mayo (MES). Obtenido de [7].

	Mayo index			
	0	1	2	3
Stool frequency	Normal	1-2/day more than normal	3-4/day more than normal	5/day more than normal
Rectal bleeding	None	Streaks	Obvious	Mostly blood
Mucosa	Normal	Erythema, mild friability	Moderate friability, erosions	Spontaneous bleeding, ulcerations
Physician's global assessment	Normal	Mild	Moderate	Severe

Hemos analizado los índices de Robarts y Mayo, ampliamente utilizados para evaluar la actividad histológica (Robarts) y endoscópica (Mayo) de la colitis ulcerosa (CU). Sin embargo, nuestro enfoque principal en el proyecto es el índice histopatológico de PHRI (Picasso Histologic Remission Index), un índice histológico novedoso y simplificado diseñado para evaluar la actividad de la colitis ulcerosa (CU). Este índice se desarrolló mediante una colaboración entre patólogos y endoscopistas con el objetivo de crear una herramienta que correlacionara con precisión la actividad endoscópica de la enfermedad y los resultados clínicos. [9]

Casi todos los investigadores coinciden ahora en que la ausencia de infiltración neutrofílica (“mucosa libre de neutrófilos”) es fundamental para definir la HR (curación histológica), debido a sus asociaciones con resultados clínicos favorables. [9]. Los neutrófilos son un tipo de glóbulo blanco que forman parte del sistema inmunitario del cuerpo y ayudan a combatir infecciones y otras enfermedades. Pertenecen al grupo de los granulocitos, que también incluye a eosinófilos y basófilos. [10]

Los neutrófilos son una de las primeras células inmunitarias que reaccionan cuando entran al cuerpo microorganismos, además estimulan la respuesta de otras células inmunitarias. Los neutrófilos se distinguen por tener múltiples gránulos y unos núcleos multilobulados (**Ver Figura 1.5**). [11]



**Figura 1.5:** Glóbulos blancos granulocitos: (A) Neutrófilo, (B) Basófilo, (C) Eosinófilo. Obtenido de [11]

El Índice PHRI se basa en un único parámetro: la infiltración de neutrófilos, lo que indica inflamación activa. Este enfoque simplificado permite una evaluación más objetiva y reproducible, identificando la presencia o ausencia de neutrófilos infiltrados en la lámina propia y el epitelio glandular de manera sencilla (“sí” o “no”). En el contexto de la colitis ulcerosa y el índice PHRI, se observará la presencia de neutrófilos en cuatro zonas principales: lámina propia, epitelio de superficie, epitelio de la cripta y lumen de la cripta.

La lámina propia es una de las capas que forma la membrana mucosa que tapiza el tubo digestivo. Es una fina capa de tejido conectivo que se encuentra situada por debajo del epitelio y forma, junto a este, la mucosa. Su función principal es proporcionar soporte y nutrición al epitelio, además de desempeñar un papel inmunológico crucial en la protección contra los gérmenes del intestino. El epitelio glandular se encarga de elaborar y secretar sustancias en el cuerpo, como moco, jugos digestivos y otros líquidos. Este epitelio se encuentra tanto en la superficie como en las criptas intestinales.

El epitelio de superficie cubre el lumen del intestino, mientras que el epitelio de la cripta reviste las criptas intestinales, invaginaciones en la mucosa que se extienden hacia la lámina propia. El lumen de la cripta es el espacio dentro de estas criptas, donde se secretan y transitan diversas sustancias. Su principal función es facilitar el intercambio de nutrientes, secreciones y desechos.

El uso del PHRI evita las escalas visuales subjetivas y la calificación arbitraria de la actividad inflamatoria (leve, moderada, severa), y se centra únicamente en la presencia de neutrófilos. La presencia de neutrófilos es la característica más reproducible en la lectura de biopsias, lo que asegura evaluaciones consistentes entre diferentes observadores.

El PHRI facilita la evaluación de múltiples biopsias de diferentes segmentos del colon, permitiendo una evaluación global de la actividad de la enfermedad. Se puede determinar una calificación global basada en la peor calificación entre todas las biopsias tomadas. Además, el PHRI es fácilmente implementable en modelos de inteligencia artificial (IA), lo que facilita el desarrollo de sistemas de diagnóstico asistidos por computadora para la CU. Su simplicidad y objetividad lo hacen ideal para su uso rutinario en la práctica clínica y su inclusión en informes de patología.

Se observan en la Tabla 1.2 los criterios de evaluación del índice PHRI. A continuación, se explican cada uno de los criterios:

- **Infiltración de neutrófilos** (en la lámina propia o en el epitelio): se acepta cualquier cantidad de neutrófilos, incluso uno solo. Si se determina la ‘ausencia de neutrófilos’, es necesario evaluar con una vista de alta potencia a 40×. El neutrófilo en la lámina propia debe estar fuera del lumen capilar.
- **Absceso criptal**: criptitis con cualquier cantidad de neutrófilos o cualquier cantidad de exudado neutrofílico desbordándose en el lumen de la cripta, y cualquier grado de lesión de las células epiteliales de la cripta.
  - Si una biopsia no tiene epitelio superficial intacto, pero muestra características de erosión/ulceración (por ejemplo, tejido de granulación y/o exudados inflamatorios), también se le asigna una puntuación de 1.
  - Cuando hay múltiples biopsias de diferentes segmentos del intestino, la puntuación máxima/más alta/peor (PHRI\_max) entre todos los sitios de biopsia será la preferida como ‘puntuación global’. [9]

**Tabla 1.2:** Criterios de evaluación del índice PHRI. Obtenida de [9]

PICaSSO Histologic Remission Index (PHRI)	
Histological finding	Score
<b>Neutrophil infiltration in lamina propria</b>	
Absent (no)	0
Present (yes)	1
<b>Neutrophil infiltration in epithelium</b>	
Absent (no)	0
Present (yes)	
Surface epithelium*	1
Cryptal epithelium (cryptitis)	1
Crypt abscess	1
<b>Total score=</b> sum of all above (maximum 4)*	



En conclusión, el PHRI es un índice histológico simple, objetivo y reproducible que se correlaciona bien con la actividad endoscópica y predice resultados clínicos en la CU. Su diseño permite una fácil implementación en la práctica clínica y en modelos de IA, ofreciendo una herramienta eficiente para evaluar la actividad de la enfermedad y apoyar la toma de decisiones clínicas.

## 2.3 Inteligencia Artificial

La inteligencia artificial (IA) es una rama de la ciencia de computación dedicada a desarrollar sistemas capaces de realizar tareas que, en condiciones normales, requerirían inteligencia humana, como el aprendizaje, el razonamiento y la percepción. Mediante la IA, es posible procesar grandes volúmenes de datos y extraer conocimientos y patrones que serían imposibles de identificar para cualquier persona. Como ejemplo, los avances digitales y los algoritmos de inteligencia artificial están aumentando el uso de estos sistemas para la evaluación y diagnóstico en diversas áreas de la medicina, conocidos como sistemas de diagnóstico asistido por computadora (*Computer Assistance Diagnosis, CAD*). [9]

Dentro del campo de la inteligencia artificial (IA) existe una subdisciplina llamada *Machine Learning* o Aprendizaje Automático (ML). El ML se basa en algoritmos y técnicas que permiten aprender de y hacer predicciones o decisiones basadas en datos. Una característica destacada del ML es su capacidad para mejorar el rendimiento en tareas específicas sin necesidad de ser reprogramado. A través del entrenamiento con conjuntos de datos, los modelos de ML pueden mejorar continuamente sus resultados.

El aprendizaje automático (ML) puede dividirse en dos tipos principales: aprendizaje supervisado y no supervisado.

### Aprendizaje Supervisado

En el aprendizaje supervisado, los algoritmos utilizan datos etiquetados, es decir, cada entrada está asociada con una etiqueta que indica la salida deseada. Este enfoque se basa en la función objetivo y la minimización de una función de pérdida. Los datos etiquetados consisten en pares de entrada-salida, donde el objetivo es aprender una función que mapee correctamente las entradas a las salidas. Este tipo de aprendizaje incluye tareas de clasificación y regresión.

Por ejemplo, al entrenar un modelo de clasificación para reconocer gatos en imágenes, cada imagen del conjunto de entrenamiento estará etiquetada con “gato” o “no gato”. El algoritmo aprende a partir de las características extraídas de las imágenes etiquetadas, ajustando sus parámetros internos para minimizar el error de predicción. Métodos comunes en el aprendizaje supervisado incluyen:

- **Regresión Lineal y Logística:** Utilizadas para predecir valores continuos y probabilidades, respectivamente.
- **Máquinas de Vectores de Soporte (SVM):** Utilizadas para clasificación y regresión, se basan en encontrar el hiperplano que mejor separa las clases en el espacio de características.
- **Árboles de Decisión y Bosques Aleatorios:** Utilizados para clasificación y regresión, donde se crean múltiples árboles de decisión para mejorar la precisión y reducir el sobreajuste.
- **Redes Neuronales Profundas (Deep Learning):** Utilizan múltiples capas de neuronas para aprender representaciones complejas de los datos, especialmente útiles en tareas como el reconocimiento de imágenes y procesamiento de lenguaje natural.

Durante el entrenamiento, el algoritmo ajusta sus parámetros utilizando técnicas de optimización como el descenso de gradiente, evaluando su rendimiento en un conjunto de validación y ajustando los hiperparámetros para evitar el sobreajuste.

## Aprendizaje No Supervisado

En contraste, el aprendizaje no supervisado trabaja con datos no etiquetados. El objetivo principal es descubrir estructuras subyacentes o patrones ocultos en los datos sin una guía explícita. Los algoritmos de aprendizaje no supervisado buscan agrupaciones naturales en los datos a través de técnicas como el clustering y la reducción de dimensionalidad. Este tipo de aprendizaje es crucial en aplicaciones como la detección de anomalías y el análisis exploratorio de datos.

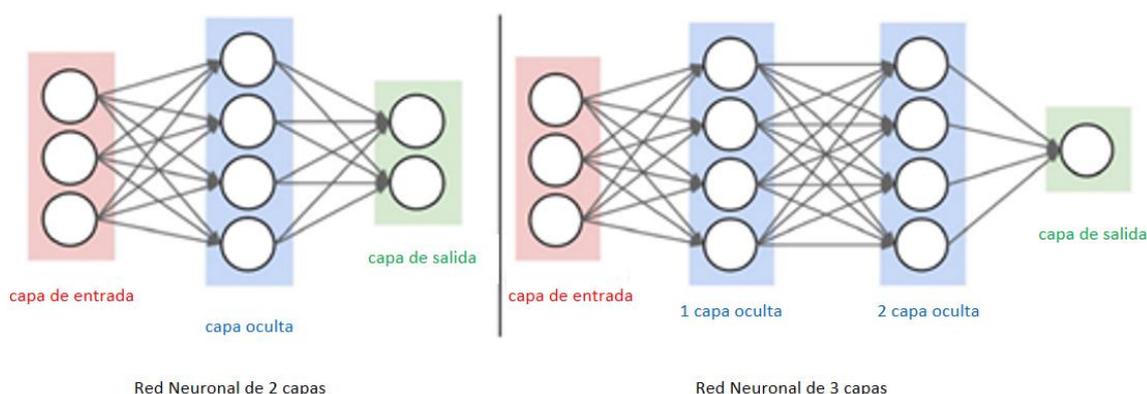
Por ejemplo, utilizando algoritmos de clustering como K-means, un conjunto de datos que proporciona información sobre la altura y el peso de un grupo de personas puede ser automáticamente segmentado en diferentes categorías basadas en sus características físicas. Otros métodos comunes en el aprendizaje no supervisado incluyen:

- **Clustering Jerárquico:** Agrupa los datos en una jerarquía de clústeres anidados, creando un árbol de clústeres.
- **Modelos de Mezcla Gaussiana (GMM):** Modelan la distribución de los datos como una combinación de múltiples distribuciones gaussianas.
- **Reducción de Dimensionalidad:** Técnicas como el Análisis de Componentes Principales (PCA) y el Análisis de Componentes Independientes (ICA) se utilizan para reducir el número de variables en el conjunto de datos, preservando la mayor cantidad de información posible.

El aprendizaje no supervisado es particularmente útil cuando se desea explorar los datos y descubrir patrones sin tener una hipótesis previa, y puede ser el primer paso antes de aplicar técnicas de aprendizaje supervisado.

### 2.3.1 Aprendizaje profundo.

El aprendizaje profundo o *Deep learning* (DL), es un subconjunto del aprendizaje automático (ML), que utiliza una arquitectura de red neuronal artificial para (Ver Figura 2.1) procesar datos de entrada. Una red neuronal es un método de la IA que permite imitar a las computadoras a procesar datos inspirada en la forma en la que lo hace el cerebro humano, permitiendo aprender a partir de grandes cantidades de datos. Una red neuronal consta de capas de nodos o neuronas artificiales: una capa de entrada, una o varias capas ocultas y una capa de salida. Estas neuronas están interconectadas entre sí mediante pesos, que son parámetros específicos que, a través de operaciones matemáticas, generan una salida.



**Figura 2.1:** Arquitectura de una red neuronal. Obtenida de [13].

Como se observa en la Figura 2.2, la neurona artificial realiza la suma de las entradas multiplicadas por sus pesos asociados. Los pesos determinan la importancia de cada conexión entre las neuronas y se ajustan durante el proceso de entrenamiento para que la red pueda realizar las tareas necesarias. Al inicio del entrenamiento, los pesos suelen ser asignados aleatoriamente.

Una vez sumadas todas las entradas ponderadas (en algunos casos se añade un término de sesgo o bias), el resultado pasa a través de una función de activación, que determina la salida de la neurona.

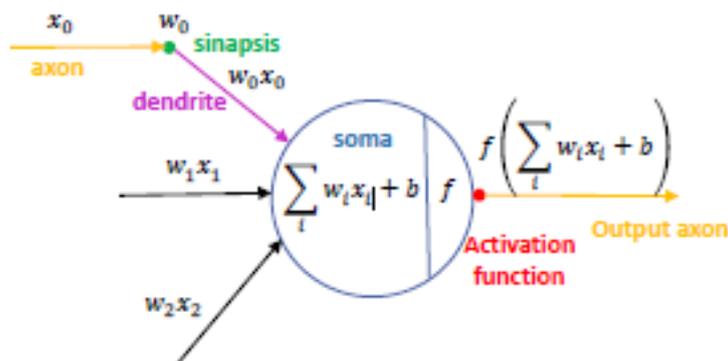


Figura 2.2: Operaciones matemáticas realizadas en las neuronas artificiales. Obtenida de [13]

El ajuste de los pesos se realiza mediante un proceso conocido como retropropagación del error (backpropagation). Este es un algoritmo de optimización utilizado para minimizar la función de pérdida, que mide la diferencia entre la salida predicha por la red y la salida deseada. La retropropagación se lleva a cabo en dos fases principales:

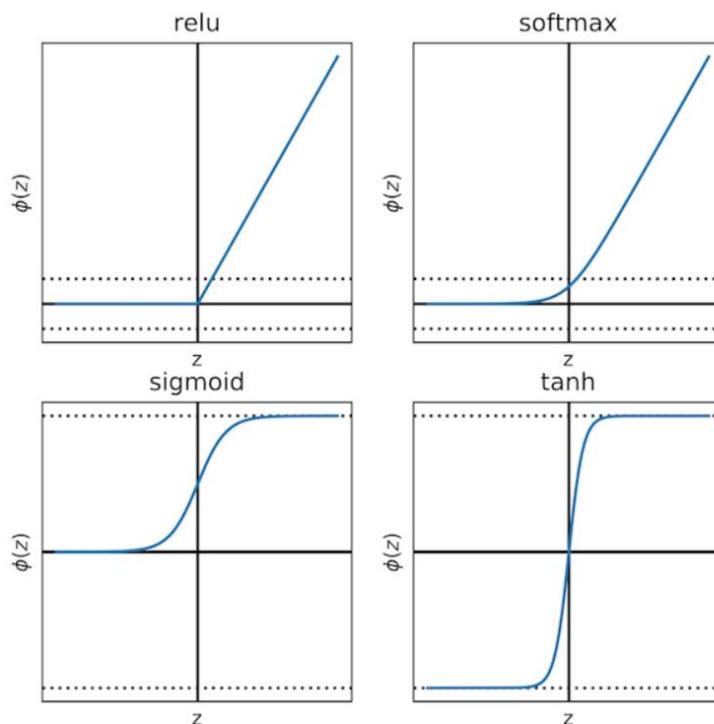
1. **Propagación hacia adelante (Forward Propagation):** En esta fase, los datos de entrada se pasan a través de la red desde la capa de entrada hasta la capa de salida. Cada neurona calcula una suma ponderada de sus entradas, aplica la función de activación y produce una salida. Esta salida se propaga a través de la red hasta que se genera una predicción en la capa de salida.
2. **Propagación hacia atrás (Backward Propagation):** En esta fase, se calcula el error en la capa de salida comparando la predicción de la red con la salida deseada utilizando una función de pérdida (como el error cuadrático medio para regresión o la entropía cruzada para clasificación). Luego, este error se propaga hacia atrás a través de la red, capa por capa, ajustando los pesos en función de su contribución al error total. El ajuste se realiza utilizando el gradiente del error respecto a cada peso, calculado mediante el algoritmo de descenso de gradiente. Los pesos se actualizan en la dirección opuesta al gradiente, reduciendo así el error en cada iteración del entrenamiento.

La retropropagación permite que la red neuronal ajuste sus pesos de manera iterativa, aprendiendo de los datos de entrenamiento para mejorar su precisión en tareas específicas. Las funciones de activación son cruciales para el cálculo de la salida de cada neurona, ya que introducen no linealidad en el modelo, permitiendo que la red aprenda y represente relaciones complejas en los datos. Las cuatro funciones más utilizadas son:

- **Rectified Linear Unit (ReLU):** Transforma los valores negativos a cero y deja pasar los números positivos sin cambios.
- **Sigmoide:** Convierte los valores de entrada en un rango entre 0 y 1, utilizado en clasificación binaria.
- **Softmax:** Utilizado en las capas de salida para problemas de clasificación multiclase, convierte los valores de entrada en probabilidades que suman 1.

- **Tangente hiperbólica (tanh):** Transforma los valores de entrada en un rango entre -1 y 1.

De manera gráfica, las funciones de activación se pueden representar, como se observa en la **Figura 2.3**, para entender con mayor claridad su funcionamiento.



**Figura 2.3:** Funciones de activación. Obtenida de [12]

### 2.3.2 Redes neuronales convolucionales

Las redes neuronales convolucionales (*convolutional neuronal networks*, CNN) son un tipo específico de red de DL diseñado principalmente para procesar y analizar datos como imágenes, señales o series temporales. En el contexto de los sistemas de CAD, las CNN han demostrado ser eficaces en tareas como la clasificación de imágenes, la detección de objetos y la segmentación de imágenes médicas. [14]

Las CNN han mejorado el análisis de imágenes médicas gracias a su capacidad para extraer características relevantes de manera automática. En este trabajo, se utilizarán distintas redes CNN para lograr la segmentación de tejidos en las áreas de interés, facilitando así el diagnóstico mediante el índice PHRI mencionado anteriormente. A continuación, se explicará cómo funcionan las CNN, su estructura y la manera en que resuelven problemas complejos.

La estructura de las CNN está compuesta por tres tipos principales de capas:

- Capa convolucional
- Capa de agrupación
- Capa totalmente conectada

La **capa convolucional** es el bloque principal de la CNN y es responsable de la mayoría de los cálculos. Para su funcionamiento, necesita datos de entrada, un filtro y un mapa de características. Como ejemplo, si la entrada es una imagen de color formada por una matriz de píxeles en 3D, estará compuesta por tres dimensiones: altura, anchura y profundidad, que corresponde con la composición RGB de una imagen (**Observar Figura 2.4 Filter color rojo**). El detector de características, conocido como kernel o filtro, se desplaza a través de diferentes áreas de la imagen

para identificar la presencia de características específicas. Este proceso se conoce como convolución.

### Convolutional layer

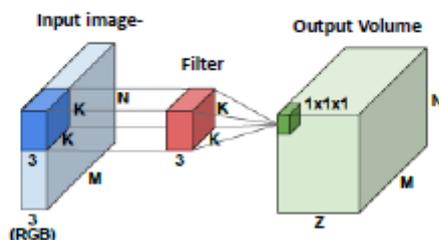


Figura 2.4: Estructura capa convolucional. Obtenido de [13]

El proceso funciona de la siguiente manera: el filtro se coloca sobre un área de la imagen y se calcula el producto escalar entre los valores de los píxeles de entrada y los pesos del filtro. El resultado se introduce en una nueva matriz de salida, también llamado mapa de características. Luego, el filtro se desplaza y repite el proceso recorriendo así toda la imagen. El resultado final del proceso forma el mapa de características. (Ver Figura 2.5)

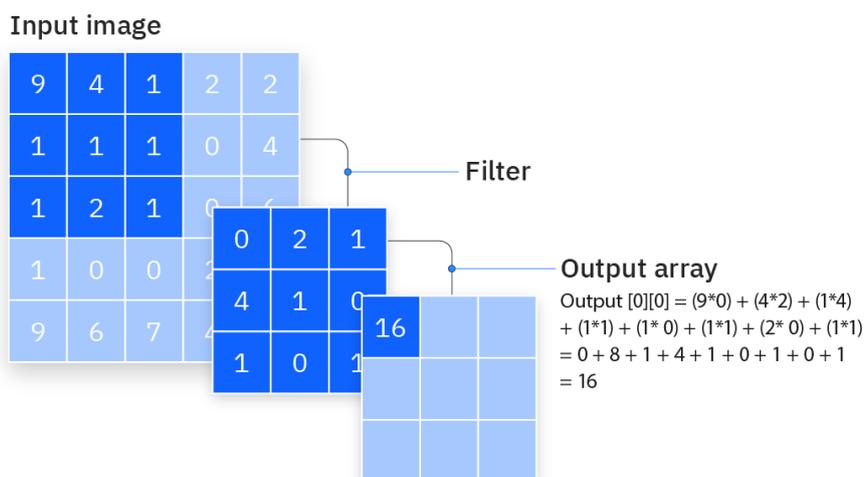


Figura 2.5: Proceso obtención del mapa de características con el filtro o kernel. Obtenido de [14]

Para obtener la salida con su correspondiente tamaño y volumen es necesario configurar tres parámetros antes de comenzar el entrenamiento:

- **Numero de filtros:** Este parámetro indica el número de filtros que se utilizarán en el proceso, determinando así la profundidad de la salida. Por ejemplo, si utilizamos tres filtros, tendremos tres mapas de características distintos, resultando en una salida con profundidad de tres.
- **Stride:** El stride define el número de píxeles de desplazamiento sobre la matriz de entrada en cada paso. Un stride alto implica que el filtro se mueve más lejos en cada paso, reduciendo así el tamaño de la salida. Aunque como comentado anteriormente, suelen ser pequeños de unas dimensiones de 3x3.

- **Zero-Padding:** Este parámetro se utiliza cuando los filtros no se ajustan a la imagen de entrada. Para solucionar este problema, se añaden ceros por todos los elementos que quedan fuera de la matriz de entrada. Existen tres tipos de padding:
  1. **Valid padding:** Descarta los datos que no se ajustan las dimensiones de salida con las de entrada.
  2. **Same padding:** Añade padding para asegurar que la salida tenga las mismas dimensiones que la entrada.
  3. **Full padding:** Añade ceros al borde de la entrada para que el tamaño de la salida sea mayor o igual al de la entrada.

Por cada operación realizada de convolución, se aplica una transformación de unidad lineal rectificadora (ReLU) (Ver Figura 2.3) al mapa de características, para introducir no linealidad al modelo.

En la **capa de agrupación**, también conocida como submuestreo, se reducen las dimensiones de la entrada disminuyendo el tamaño de los mapas de características (Ver Figura 2.6). La operación de agrupación barre toda la entrada con un filtro sin pesos. El filtro aplica una función de agregación a los valores del área de la matriz y así llena la matriz de salida. Hay dos tipos de agrupación principales:

- **Agrupación máxima:** El filtro selecciona el píxel con el valor más alto dentro de su área y lo envía a la matriz de salida.
- **Agrupación media:** El filtro calcula el valor promedio dentro su área lo envía a la matriz de salida.

Aunque se pierde mucha información en la capa de agrupación, esta capa extrae características a diferentes niveles de resolución además de ayudar a reducir la complejidad, mejora la eficiencia y limitar el riesgo de sobreajuste.

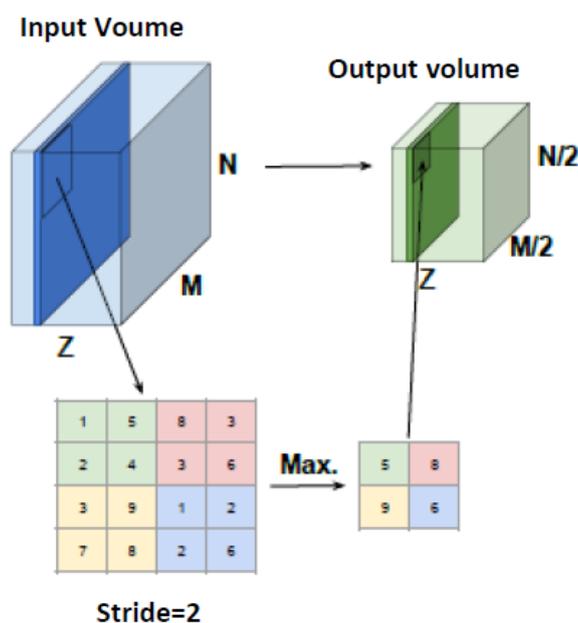


Figura 2.6: Funcionamiento capa agrupación. Obtenido de [13].



Para finalizar, la **capa totalmente conectada** es una estructura en la que cada nodo de la capa actual está conectado a todos los nodos de la capa anterior. Esto se diferencia de las capas convolucionales y de agrupación, donde las conexiones son locales y cada nodo solo se conecta a una pequeña región de la capa anterior.

En la capa totalmente conectada, cada nodo está interconectado con todos los nodos de la capa anterior, formando una red completamente interconectada. En esta capa se realiza la clasificación final basándose en las características extraídas por las capas anteriores. Se utiliza una función de activación softmax (**Ver Figura 2.3**) para convertir las salidas en una probabilidad de entre 0 y 1 para cada clase.

Las Redes Neuronales Convolucionales (CNN) son una base fundamental en la construcción de modelos más complejos, especialmente en tareas de segmentación de imágenes. A medida que avanzamos, exploraremos arquitecturas de segmentación que han evolucionado a partir de las CNN, como FPN (Feature Pyramid Network), U-Net y U-Net Plus Plus. Estas arquitecturas están diseñadas para abordar la segmentación de imágenes de manera más eficiente y precisa, utilizando diferentes estrategias para combinar características extraídas a múltiples escalas y resoluciones.

En la siguiente sección, profundizaremos en cada una de estas arquitecturas, examinando cómo se construyen sobre la base de las CNN y cómo implementan técnicas avanzadas para mejorar el rendimiento en tareas de segmentación. Veremos cómo FPN aprovecha pirámides de características para manejar objetos de diferentes tamaños, mientras que U-Net y U-Net Plus Plus utilizan caminos de codificación y decodificación simétricos para lograr segmentaciones precisas incluso en conjuntos de datos con cantidades limitadas de anotaciones.



## Capítulo 3. Materiales

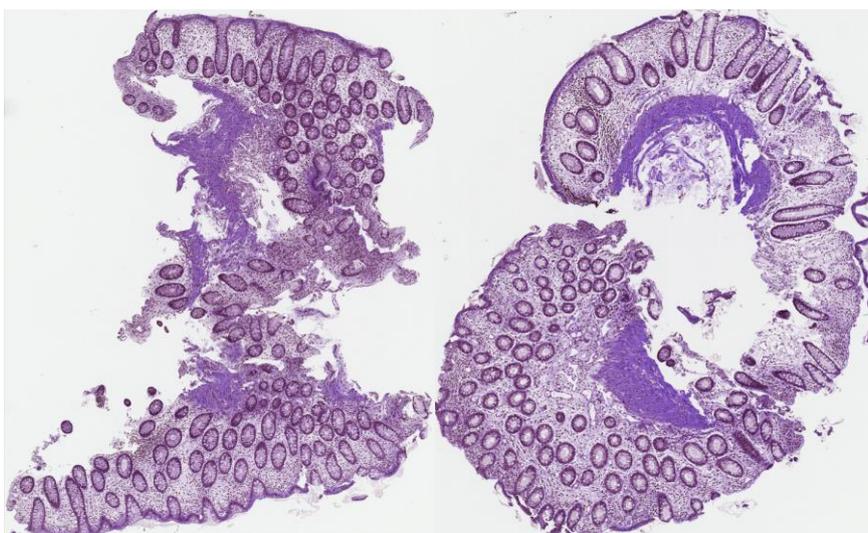
### 3.1 Base de datos de WSI

Las imágenes histológicas completas (WSI, por sus siglas en inglés) representan una innovación en la patología digital. Estas imágenes se obtienen mediante la digitalización de las imágenes histológicas conseguidas de biopsias, facilitando su análisis, almacenamiento y compartición. El proceso de creación de una imagen WSI incluye varios pasos técnicos:

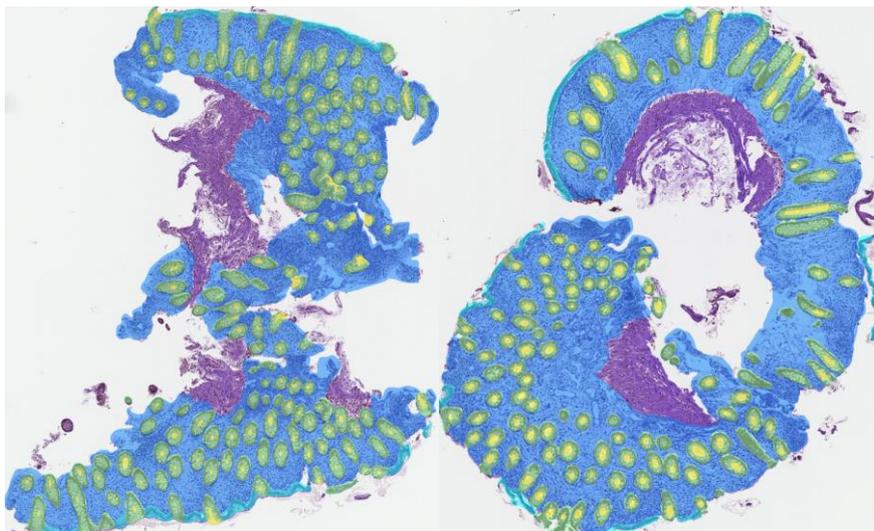
1. **Preparación del Tejido:** Las muestras de tejido son fijadas para preservar la morfología celular, embebidas en parafina para darles consistencia, cortadas en secciones delgadas usando un microtomo, y teñidas comúnmente con hematoxilina y eosina (HE).
2. **Digitalización:** Las secciones teñidas son escaneadas utilizando un sistema de ópticas, cámaras digitales, y robótica para capturar imágenes de alta resolución. Los escáneres pueden variar desde modelos compactos para pocas láminas hasta grandes equipos que manejan cientos de láminas simultáneamente.
3. **Generación de Imágenes Digitales:** Las imágenes capturadas se ensamblan en una sola imagen sin costuras mediante técnicas de escaneo basadas en mosaico o en línea. Esto permite la visualización y el análisis detallado de las muestras en un formato digital [22].

La base de datos de imágenes histológicas utilizada en este proyecto se obtuvo a través de una colaboración con múltiples centros médicos internacionales, que llevaron a cabo un estudio multicéntrico real y prospectivo sobre la endoscopia e histología en la colitis ulcerosa (UC) [9]. Contamos con un gran número de WSI, sin embargo, como se ha comentado anteriormente la tarea de anotación es una tarea con un alto coste de tiempo, así pues, tenemos un total de 42 WSI anotadas las cuales se dividen en conjuntos de entrenamiento y prueba: 33 imágenes para entrenamiento y 9 para prueba. Las imágenes de prueba también se utilizan para validación, 50% de las imágenes de prueba son para test y el otro 50% para validación.

Estas imágenes han sido anotadas por patólogos expertos a nivel de píxeles diferenciando las áreas de interés de este tejido: lámina propia, epitelio de la superficie, epitelio de la cripta y lumen de la cripta. En la **Figura 3.1** se muestra un ejemplo de una imagen WSI sin anotar, y en la **Figura 3.2** se presenta la misma imagen anotada por el patólogo, con un código de colores que indica las secciones de interés.



**Figura 3.1:** Imagen WSI del tejido del colon con CU. [9]



**Figura 3.2:** Imagen WSI anotada por patólogos [9].

En la **Figura 3.2** se puede observar una imagen WSI anotada por los patólogos, utilizando el siguiente código de colores para identificar las diferentes áreas de interés:

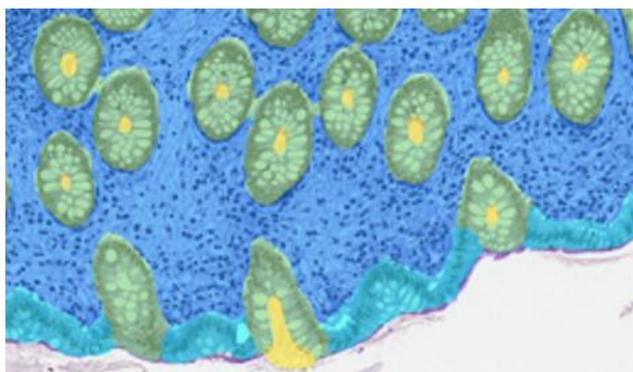


Figura 3.2.1: Código de colores imagen WSI

1. **Color azul:** Lámina Propia
2. **Color azul celeste:** Epitelio de la superficie
3. **Color amarillo-verde:** Epitelio de la cripta
4. **Color amarillo:** Lumen de la cripta

Estas anotaciones detalladas son cruciales para nuestro proyecto, ya que proporcionan una referencia precisa y estandarizada de las distintas estructuras histológicas presentes en las imágenes. La utilización de un código de colores facilita la visualización y diferenciación de las áreas anotadas, lo que es esencial para el desarrollo y entrenamiento de una inteligencia artificial (IA) capaz de segmentar automáticamente estas regiones de interés.

Gracias a estas anotaciones y a las máscaras generadas, podemos entrenar modelos de IA para que aprendan a identificar y segmentar las áreas específicas dentro de las imágenes histológicas. Este proceso es fundamental para avanzar en la automatización del análisis histológico, permitiendo una evaluación más rápida y precisa de las muestras de tejido. La capacidad de la IA para segmentar correctamente las áreas de interés es de vital importancia en nuestro proyecto, ya que mejora la precisión del diagnóstico y la evaluación de la colitis ulcerosa, contribuyendo a una mejor comprensión y manejo de la enfermedad.

La colaboración con los patólogos y la precisión de sus anotaciones aseguran que nuestro modelo de IA tenga una base sólida para el aprendizaje, aumentando la probabilidad de éxito en la segmentación automática de las áreas de interés.

## 3.2 Software

Para el desarrollo de este proyecto, se ha utilizado una combinación de herramientas de software y bibliotecas específicas, diseñadas para facilitar el desarrollo y entrenamiento de modelos de inteligencia artificial. A continuación, se detallan las principales herramientas y bibliotecas empleadas.

### Visual Studio Code (VSCode)

Visual Studio Code, comúnmente conocido como VSCode, es el editor de código fuente elegido para este proyecto. Además de ser un programa libre, VSCode es una herramienta altamente versátil y extensible que ofrece numerosas ventajas:

- **Extensiones y Plugins:** VSCode permite la instalación de extensiones para añadir funcionalidades adicionales, como el soporte para depuración de código, integración con sistemas de control de versiones (como Git) y herramientas específicas para desarrollo en Python.
- **Editor Multilenguaje:** Aunque Python es el principal lenguaje utilizado, VSCode soporta múltiples lenguajes de programación, facilitando el trabajo con archivos de configuración, scripts y otros componentes del proyecto.
- **Depuración Integrada:** La capacidad de depurar código dentro del editor mismo ayuda a identificar y corregir errores de manera eficiente.

### Python

Python versión 3.8, ha sido el lenguaje de programación principal utilizado en este proyecto, debido a sus amplias capacidades y popularidad en el campo de la inteligencia artificial y el aprendizaje automático.

### Bibliotecas de Python para Inteligencia Artificial

- **Keras:** Keras es una biblioteca de alto nivel para el desarrollo de modelos de aprendizaje profundo. Es conocida por su simplicidad y facilidad de uso, permitiendo la creación rápida de prototipos. Keras se ejecuta sobre TensorFlow, proporcionando una interfaz intuitiva para la construcción y entrenamiento de redes neuronales.
- **TensorFlow:** TensorFlow es una biblioteca de código abierto desarrollada por Google, utilizada principalmente para el cálculo numérico y el aprendizaje automático. Sirve como backend para Keras y proporciona una infraestructura robusta para desarrollar y desplegar modelos de aprendizaje profundo.
- **NumPy:** NumPy es una biblioteca fundamental para el cálculo numérico en Python. Ofrece soporte para matrices multidimensionales y una amplia colección de funciones matemáticas de alto nivel para operar con ellas.
- **Pandas:** Pandas es una biblioteca de manipulación y análisis de datos, proporcionando estructuras de datos y herramientas de análisis de alto rendimiento.
- **Matplotlib y Seaborn:** Estas bibliotecas se utilizan para la visualización de datos. Matplotlib es una biblioteca de gráficos 2D que permite crear figuras estáticas, animadas e interactivas. Seaborn, por su parte, está construida sobre Matplotlib y proporciona una interfaz de alto nivel para la visualización de datos estadísticos.



### 3.3 Hardware

Para el desarrollo y entrenamiento de los modelos de inteligencia artificial en este proyecto, se ha utilizado un ordenador MSI con un procesador Intel® Core™ i7-12700H de 12ª generación a 2.70 GHz, sistema operativo Windows 11 de 64 bits y 16 GB de RAM. Este equipo también cuenta con una GPU NVIDIA GeForce RTX 3050 Ti Laptop, la cual es esencial para acelerar el entrenamiento de los modelos mediante cálculos paralelos. Además, dispone de un almacenamiento de 1 TB, lo cual es crucial para manejar las grandes imágenes de Whole Slide Imaging (WSI) debido a su alta resolución y tamaño considerable.



## Capítulo 4. Metodología

Para abordar este problema, se ha diseñado un plan meticuloso para obtener los mejores resultados posibles. En primer lugar, se realizó una búsqueda exhaustiva de modelos de segmentación viables. Después de esta búsqueda, se decidió utilizar la arquitectura U-Net para nuestro código principal. Posteriormente, se efectuaron modificaciones al modelo U-Net para alcanzar los objetivos específicos de nuestro proyecto. Se llevaron a cabo diversas pruebas para optimizar la estructura del modelo y determinar los mejores parámetros posibles.

Para verificar la precisión de los resultados, se midieron los índices de Jaccard, Dice y pérdidas, cuyos valores y significados se detallarán más adelante. Además, se desarrolló un método para realizar inferencias, es decir, generar máscaras de segmentación de las imágenes. Estas máscaras se superponen con las imágenes originales para compararlas con los *overlays* anotados por los patólogos, permitiendo una evaluación visual de los resultados obtenidos.

Una vez que tuvimos el modelo y la arquitectura ajustada a nuestro problema, se emplearon otros modelos de segmentación obtenidos de [\[15\]](#) para realizar pruebas adicionales. Estos modelos fueron evaluados utilizando nuestro método principal como referencia, lo que permitió una comparación detallada de los resultados obtenidos.

### 4.1 Arquitecturas

En este apartado, se llevará a cabo un análisis del funcionamiento y la estructura de la arquitectura U-Net, una red neuronal convolucional utilizada en la segmentación de imágenes. Inicialmente, explicaremos detalladamente los principios fundamentales de la U-Net. Posteriormente, analizaremos otro tipo de arquitecturas y la forma en la que las ajustaremos para poder conseguir los resultados deseados.

#### 4.1.1 Definición U-Net

La U-Net es una arquitectura de red neuronal convolucional (CNN) diseñada para la segmentación de imágenes, especialmente en el ámbito de las imágenes biomédicas. Este modelo fue desarrollado originalmente por Olaf Ronneberger, Philip Fischer y Thomas Brox en 2015. Surgió por la necesidad de mejorar la segmentación en imágenes biomédicas. Las técnicas de segmentación existentes requerían grandes cantidades de datos etiquetados y realizaban segmentaciones poco precisas cuando no se disponía de suficientes datos. La U-Net solucionó este problema permitiendo una segmentación más precisa con menos datos de entrenamiento. [\[16\]](#)

La U-Net se caracteriza por su estructura en forma de U debido a su diseño simétrico. Consta de una etapa de contracción, también llamada codificador (encoder), utilizada para captar el contexto de una imagen, y una etapa de expansión, también llamada decodificador (decoder), que permite una localización precisa mediante la convolución transpuesta.

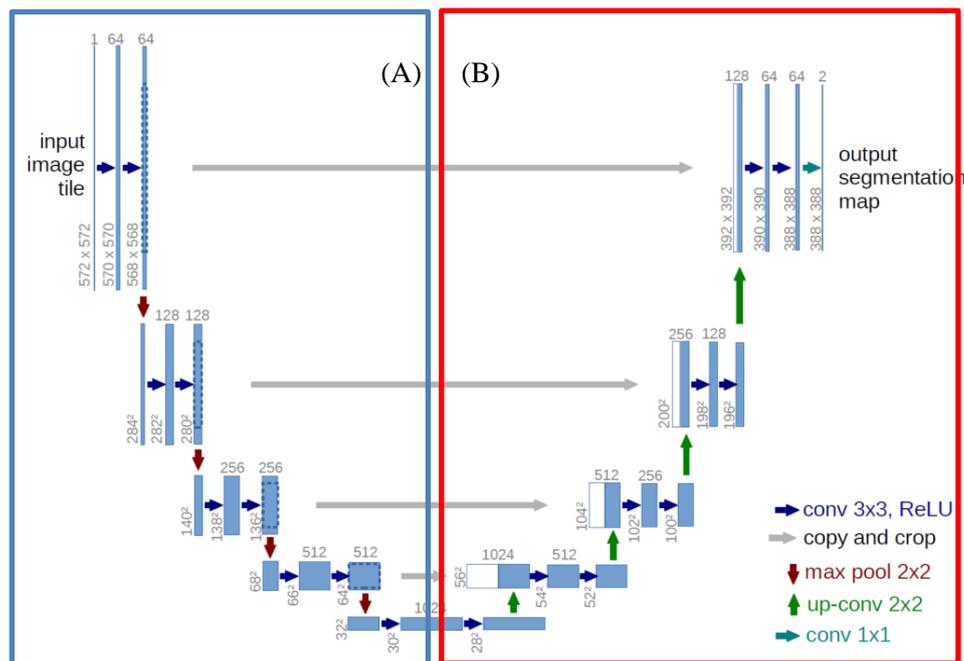


Figura 4.1: Arquitectura U-Net: (A) Codificador, (B) Decodificadores. Obtenido de [17].

En la etapa de codificación, la imagen se procesa a través de una serie de capas convolucionales seguidas de operaciones de agrupación máxima (max pooling) y funciones de activación ReLU. Esto permite al modelo reducir las dimensiones espaciales de la imagen progresivamente mientras aumenta su profundidad, es decir, sus características contextuales. **(Ver Figura 4.1, parte (A) codificador).**

En la segunda etapa, decodificadora, se reconstruye la imagen a partir de las características extraídas, aumentando gradualmente la resolución utilizando capas de upsampling y convoluciones transpuestas. En esta fase se restaura la información espacial detallada de la imagen. (Observar Figura 4.1, parte (B) decodificador).

Una de las claves de la U-Net es la concatenación entre el encoder y el decoder, que permite a la red combinar información detallada de las capas superficiales. Esto facilita la transferencia directa de características de alta resolución desde el encoder al decoder. Estas conexiones se conocen como skip connections (conexiones residuales). **(Ver Figura 4.1, flecha gris conectando las etapas).**

La capa de salida de la U-Net es una capa de convolución que reduce la profundidad a la cantidad de clases de segmentación deseadas y utiliza una función de activación softmax o sigmoid para obtener la probabilidad de cada clase por píxel.

#### 4.1.2 Implementación U-Net

Nuestro modelo de U-Net consta de 4 archivos de código Python que sirven como esqueleto del proyecto. La estructura es la siguiente:

- **main.py**: Es el esqueleto de nuestro código. En este archivo definimos nuestros parámetros de entrenamiento, incluyendo:
  - Tamaño de la imagen de entrada.
  - Tamaño del lote (batch size).
  - Tasa de aprendizaje (learning rate).
  - Número de filtros.
  - Optimizador

Además de definir nuestros parámetros, en el archivo **'main.py'** se inicializan las funciones para cargar los datos de las imágenes dependiendo del estado: entrenamiento, testeo o predicción. Para finalizar, en **'main.py'** se realiza el entrenamiento principal, inicializando el modelo y procesando los resultados. El mismo proceso ocurre en la predicción y testeo.

- **dataset.py:** Se encarga de la carga y preprocesamiento de imágenes. En este archivo se leen las imágenes de entrada, se aplican las transformaciones necesarias y se preparan los lotes de entrada para el entrenamiento.
- **model.py:** Define la arquitectura de nuestro modelo U-Net, especificando las capas convolucionales, las capas de agrupación, las capas de upsampling y las conexiones residuales (skip connections).
- **utilis.py:** Contiene las herramientas necesarias para entrenar el modelo y ejecutar el código. Aquí se realizan normalizaciones, cambios de variables, y se obtienen los índices de Jaccard y Cat-Dice. Además, se generan gráficas de entrenamiento, se guarda el modelo final, se crean las máscaras para la predicción y se superponen las máscaras de salida con las imágenes originales. También incluye el bucle que comprueba que el modelo está entrenando correctamente y detener el entrenamiento cuando no haya mejoras.

## 4.2 Otras arquitecturas de segmentación

En el proyecto, implementaremos los modelos de segmentación disponibles en [\[15\]](#). Estos modelos pueden ser fácilmente integrados mediante el uso de librerías de Python. Una de las ventajas clave de utilizar estos modelos es la posibilidad de emplear encoders específicos con pesos preentrenados, lo que puede significativamente mejorar el rendimiento de la red. Específicamente, utilizaremos encoders basados en las arquitecturas ResNet y VGG11.

**ResNet** (Red Residual Network) es una arquitectura de red neuronal profunda introducida por investigadores de Microsoft en 2015. La principal innovación de ResNet es la incorporación de bloques residuales, que permiten a las redes neuronales ser mucho más profundas sin enfrentar el problema de la degradación del rendimiento. Estos bloques residuales se basan en la idea de conexiones residuales (skip connections), que facilitan el flujo de gradientes a través de la red durante el entrenamiento. En lugar de aprender directamente la función deseada, ResNet aprende la diferencia entre la entrada y la salida esperada. Esto permite entrenar redes con cientos o incluso miles de capas, superando los límites de las arquitecturas tradicionales. ResNet ha demostrado ser altamente efectiva en tareas de clasificación y detección de imágenes, y ganó el desafío ImageNet en 2015, estableciendo un nuevo estándar en el campo de la visión por computadora. [\[13\]](#)

**VGG11** es una versión de la familia de redes neuronales convolucionales desarrollada por el Visual Geometry Group (VGG) de la Universidad de Oxford. La arquitectura VGG se caracteriza por su simplicidad y homogeneidad en el diseño. En particular, VGG11 consta de 11 capas de red, combinando convoluciones de tamaño 3x3 y agrupaciones de tamaño 2x2. Las capas convolucionales en VGG11 están organizadas en bloques, seguidos de capas de agrupación máxima (max pooling) para reducir la dimensionalidad de las características. La uniformidad en el tamaño del filtro y la estructura repetitiva de los bloques en VGG11 facilita la implementación y el ajuste del modelo. Aunque VGG11 no es tan profunda como otras arquitecturas modernas, su diseño compacto y eficiente la convierte en una opción popular para tareas de visión por computadora como la clasificación y la detección de características [\[13\]](#).

Los pesos preentrenados en estos encoders provienen del entrenamiento en el conjunto de datos ImageNet, que es una base de datos grande y ampliamente utilizada para la clasificación de imágenes. Aunque ImageNet está diseñado para tareas de clasificación, los encoders preentrenados en este conjunto de datos pueden ser transferidos eficazmente a tareas de

segmentación de imágenes. Esto se debe a que las capas iniciales de una red entrenada en ImageNet aprenden características genéricas y útiles, como bordes, texturas y patrones, que son relevantes para una variedad de tareas de visión por computadora, incluida la segmentación. Utilizar estos pesos preentrenados proporciona una base sólida y puede acelerar el proceso de entrenamiento y mejorar la precisión del modelo en la segmentación de imágenes médicas.

#### 4.2.1 U-Net 2.0

La arquitectura básica del modelo se mantiene igual que la utilizada en nuestro código principal. Sin embargo, el uso de un encoder preentrenado nos ayuda a mejorar el modelo y obtener distintos resultados (Ver Figura 4.2). Para implementar este modelo, utilizamos la librería de segmentación de [15], que nos permite modificar varios aspectos de la estructura del modelo, incluyendo el número de bloques de los encoders, los pesos a utilizar, y el tipo de encoder. Además, podemos ajustar parámetros relacionados con las clases y el tipo de imagen (es decir, RGB).

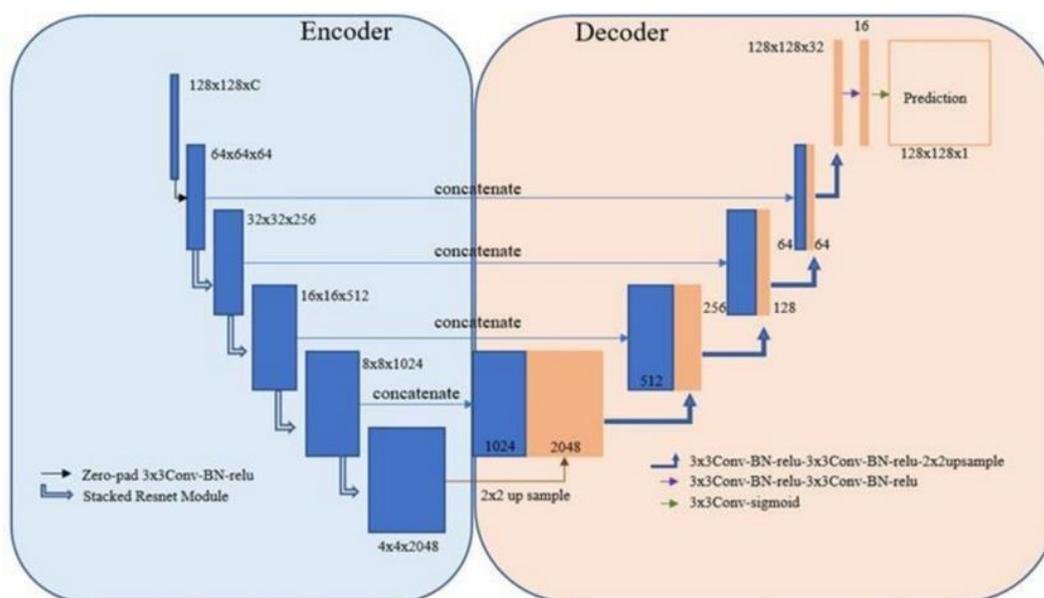


Figura 4.2: Estructura U-Net con encoder preentrenado. Obtenido de [16]

Para investigar cuál arquitectura es mejor, se realizan todas las pruebas mencionadas en el apartado anterior, buscando las mejores estructuras y parámetros para cada tipo de encoder, ResNet y VGG11. Cada uno de estos encoders cuenta con distintas variantes dentro de su familia, como ResNet50, ResNet18, VGG11, y VGG11\_bn. En la arquitectura VGG, la designación VGG11\_bn indica que la red incorpora batch normalization (BN).

Batch normalization es una técnica que normaliza las activaciones de una capa a lo largo de un mini-batch durante el entrenamiento. Específicamente, BN ajusta las activaciones de manera que tengan una media de 0 y una desviación estándar de 1, lo que ayuda a estabilizar y acelerar el proceso de entrenamiento. Esta técnica se introdujo para mitigar problemas como el desvanecimiento o explosión del gradiente, y permite utilizar tasas de aprendizaje más altas, lo que a menudo resulta en una convergencia más rápida y mejores resultados generales. [13]

Estas variantes representan diferentes versiones de la misma arquitectura, que se han actualizado y mejorado a lo largo de los años. En este documento se presentarán los resultados obtenidos con las mejores estructuras.



mapas de características, siendo la salida de la última capa de cada etapa utilizada como referencia para construir la pirámide de características.

La vía descendente del FPN se encarga de aumentar la resolución de los mapas de características de niveles superiores de la pirámide mediante técnicas de upsampling. Estos mapas, aunque más gruesos en términos de resolución espacial, contienen información semánticamente más rica. Luego, estos mapas se enriquecen fusionándolos con características de la vía ascendente a través de conexiones laterales. Cada conexión lateral combina mapas de características del mismo tamaño espacial tanto de la vía ascendente como de la vía descendente.

El enfoque del FPN permite manejar eficientemente objetos y detalles de diferentes escalas dentro de una imagen, mejorando la precisión en tareas como la segmentación de objetos y la detección de múltiples objetos a distintas resoluciones. [19]

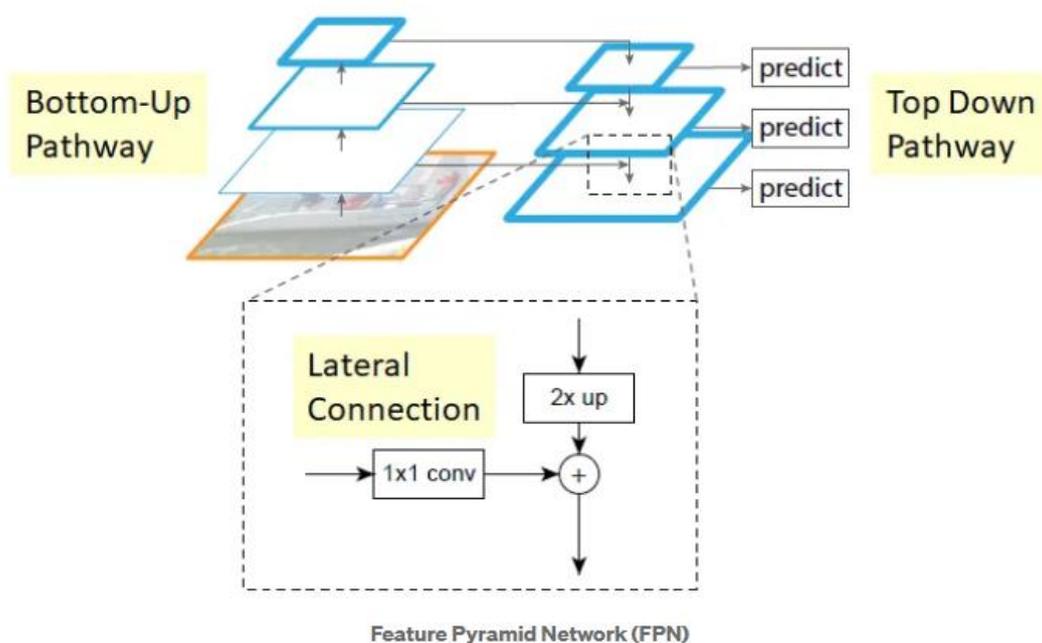


Figura 4.4: Estructura modelo FPN. Obtenida de [19]

En la implementación del modelo de Feature Pyramid Network (FPN), se pueden ajustar varios parámetros relacionados con su estructura fundamental:

- **Tamaño del codificador:** Define el número de etapas o niveles en el encoder, generalmente entre 3 y 5. Cada etapa reduce las dimensiones espaciales de las características extraídas, formando una jerarquía de características.
- **Filtros:** Especifica el número de filtros de convolución en la pirámide de características del FPN. Esta red se utiliza para generar mapas de características a múltiples escalas proporcionales a partir de una única imagen de entrada.
- **Número de canales:** Define el número de filtros de convolución en los bloques de segmentación dentro del FPN. Estos bloques son responsables de producir la segmentación final del objeto en la imagen.
- **Combinación de datos:** Indica cómo fusionar las características de la pirámide dentro del FPN. Las opciones disponibles son "add" (suma) y "cat" (concatenación), determinando cómo se combinan las características de diferentes escalas.



**Dropout del decodificador:** Especifica la tasa de dropout espacial aplicada a la pirámide de características en el FPN. El dropout se utiliza para regularizar la red y evitar el sobreajuste durante el entrenamiento.



## Capítulo 5. Resultados

En este capítulo, explicaremos las métricas de evaluación utilizadas para analizar todas las arquitecturas descritas en el apartado de metodología. Luego, detallaremos su implementación y los resultados obtenidos. Realizaremos diferentes pruebas para identificar el modelo que mejor se adapte a nuestro problema. Utilizaremos, como validación cuantitativa, las métricas explicadas en la sección 5.1 "Métricas de Evaluación" y, de manera cualitativa, evaluaremos los resultados mediante el método de inferencia.

### 5.1 Métricas de Evaluación

En el campo de la IA y, en particular, en los modelos de segmentación de imágenes, la evaluación precisa del rendimiento del modelo es crucial para garantizar que las predicciones sean fiables y útiles. Las métricas de evaluación juegan un papel fundamental en este proceso, ya que proporcionan una medida cuantitativa de cómo se está desempeñando el modelo durante el entrenamiento y la validación.

Para evaluar el rendimiento de nuestros modelos, utilizaremos tanto métricas de similitud como métricas de evaluación del error. Entre las métricas de similitud se encuentran el índice de Jaccard y el coeficiente Dice, que miden la superposición entre las máscaras de segmentación predichas y las etiquetas verdaderas. Estas métricas son esenciales para entender qué tan bien el modelo está capturando las estructuras y patrones en las imágenes. Por otro lado, las métricas de evaluación del error, como la función de pérdida (Loss), nos permiten monitorizar y minimizar el error del modelo durante el entrenamiento.

Para evaluar el rendimiento de nuestros modelos durante el entrenamiento y la validación, utilizaremos métricas de similitud y evaluación del rendimiento (Categorical Dice) y de optimización y evaluación de error (Loss) en cada época de entrenamiento. Una época es una iteración en la que el modelo ha visto y procesado cada ejemplo del conjunto de datos. En nuestro caso, una época será cuando nuestro modelo haya visto las 33 imágenes una vez, dado que nuestro batch es de 1. La manera de obtener las métricas de validación por época es la siguiente:

#### Métricas de similitud:

**Perdida de entrenamiento (Training Loss):** Proporciona información sobre qué tan bien está aprendiendo el modelo durante el entrenamiento con el conjunto de datos de entrenamiento. El objetivo es minimizar esta pérdida durante cada época para realizar predicciones más precisas. Se calcula de la siguiente manera:

1. Se realiza la intersección por clases entre las máscaras de predicción (nuestra salida del modelo) con las etiquetas verdaderas (las imágenes anotadas por los patólogos):

$$\text{intersección} = 2 \times \sum_i y_{\text{true}}^{(i)} \times y_{\text{pred}}^{(i)} \quad (1.1)$$

2. Se calcula la unión por clases entre las máscaras de predicción y las máscaras etiquetadas verdaderas:

$$\text{unión} = \sum_i \left( \sum_j \left( y_{\text{true}}^{(i)}(j) \right)^2 \right) + \sum_i \left( \sum_j \left( y_{\text{pred}}^{(i)}(j) \right)^2 \right) \quad (1.2)$$

3. Posteriormente, se realiza el coeficiente Dice por clase. Se añade un valor pequeño  $\epsilon$  para evitar la división por cero.:

$$\text{dice}^{(i)} = \frac{\text{intersection}^{(i)}}{\text{union}^{(i)} + \epsilon} \quad (1.3)$$

4. Se realiza un promediado del coeficiente Dice de todas las clases, donde N es el número total de clases:

$$\text{cat\_dice} = \frac{1}{N} \sum_{i=1}^N \text{dice}^{(i)} \quad (1.4)$$

5. Se realiza la siguiente resta para obtener una métrica de divergencia, la cual es multiplicada por el número de lotes y sumada a un acumulador (al ser 1, no se ve afectado). Finalmente, se calcula la media de la pérdida considerando el tamaño del batch para obtener la pérdida en cada época de entrenamiento.

$$\text{Categorical Dice Loss} = 1 - \frac{2 \times \sum_{i=1}^n (y_{\text{true},i} \times y_{\text{pred},i})}{\sum_{i=1}^n y_{\text{true},i}^2 + \sum_{i=1}^n y_{\text{pred},i}^2 + \epsilon} \quad (1.5)$$

6. La pérdida de cada época se calcula como (lote de 33 imágenes):

$$\text{epoch\_loss} = \frac{\text{categorical dice loss}_o}{\text{longitud\_lote}} \quad (1.6)$$

**Pérdida de validación (Validation Loss):** El cálculo de la pérdida de validación sigue el mismo proceso que el de la pérdida de entrenamiento, con la diferencia de que se utiliza un conjunto de datos de validación independiente del conjunto de entrenamiento. La pérdida de validación evalúa qué tan bien el modelo generaliza con datos no vistos. Una pérdida de validación baja entre 0-1, indica que el modelo generaliza bien y no está sobreajustando al conjunto de datos de entrenamiento.

**Dice de Entrenamiento (Training Dice):** Mide la similitud entre las máscaras de predicción y las máscaras de etiqueta verdadera (ground truth) utilizando el coeficiente Dice. Se calcula de la misma manera que la pérdida de entrenamiento, pero en el último paso (número 5), no se resta 1 del resultado y tampoco se multiplica por el tamaño del lote en cada iteración. Un valor alto entre 0-1, indica una buena similitud entre la predicción y ground truth.

**Dice de Validación (Validation Dice):** Se calcula con el conjunto de datos de validación, de la misma manera que el Dice de entrenamiento. Evalúa qué tan bien el modelo generaliza la segmentación de imágenes. Un valor alto indica que el modelo es capaz de segmentar con precisión nuevas imágenes no vistas durante el entrenamiento.

Estas métricas se guardarán y graficarán época por época para visualizar el progreso del entrenamiento del modelo.

### Cálculo del índice de Jaccard o IoU:

El índice de Jaccard, también conocido como Intersection over Union (IoU), es otra métrica de similitud utilizada para evaluar la superposición entre las máscaras de segmentación predichas y las etiquetas verdaderas. El IOU mide la superposición entre la máscara de predicción generada por el modelo y la máscara de verdad (ground truth), evaluando la exactitud de las predicciones del modelo al comparar las áreas de intersección y unión de las dos máscaras. Se calcula de la siguiente manera:

1. **Intersección:** La suma de los valores positivos comunes entre la predicción y la etiqueta verdadera:

$$\text{Intersección} = \sum_i (y_{\text{true},i} \times y_{\text{pred},i}) \quad (1.7)$$

2. **Unión:** Es la suma total de los valores positivos en ambas la predicción y la etiqueta verdadera:

$$\text{unión} = \sum_i (y_{\text{true},i} + y_{\text{pred},i} - y_{\text{true},i} \times y_{\text{pred},i}) \quad (1.8)$$

3. **Cálculo del Índice de Jaccard:**

$$\text{IoU} = \frac{\text{intersección}}{\text{unión}} \quad (1.9)$$

Calcularemos el valor del Categorical Dice y del índice de Jaccard/IoU por clases y obtendremos la media de estas métricas como nuestras conclusiones. Estas métricas finales nos permitirán tener una visión cuantitativa y precisa del rendimiento del modelo en términos de similitud y error, asegurando que nuestras predicciones sean lo más exactas posible.

## 5.2 Configuración Experimental

En esta sección, se detallan los parámetros y configuraciones utilizados durante los experimentos para evaluar el rendimiento del modelo de segmentación de imágenes. La configuración experimental es crucial para garantizar la validez y reproducibilidad de los resultados obtenidos.

### Hiperparámetros

Los hiperparámetros son parámetros configurables que influyen en el proceso de aprendizaje del modelo. La correcta selección de estos parámetros puede mejorar significativamente el rendimiento del modelo. A continuación, se describe el hiperparámetro clave en nuestros experimentos:

- **Tasa de Aprendizaje (Learning Rate):** Ajustaremos la tasa de aprendizaje observando cuál es la que mejor se adapta a nuestro modelo mediante pruebas experimentales. La tasa de aprendizaje controla el tamaño de los pasos que el algoritmo da al descender por la pendiente del gradiente. Una tasa de aprendizaje demasiado alta puede evitar que el modelo converja, mientras que una demasiado baja puede hacer que el entrenamiento sea excesivamente lento.

### Tamaño de las Imágenes

El tamaño de las imágenes es un factor importante en el preprocesamiento y entrenamiento del modelo. En nuestros experimentos, hemos redimensionado las imágenes a tres tamaños diferentes: 512x512, 1024x1024 y 256x256 píxeles. Evaluaremos cuál de estos tamaños ofrece el mejor equilibrio entre la calidad de la imagen y la eficiencia computacional, permitiendo que el modelo procese las imágenes de manera eficiente mientras mantiene suficiente detalle para una segmentación precisa.

### Número de Filtros

El número de filtros, o canales en una capa convolucional, determina cuántas características aprende el modelo en cada capa. Para nuestros experimentos, hemos configurado el número de filtros de la siguiente manera:

- **Primera Capa Convolucional:** N filtros
- **Segunda Capa Convolucional:** 2N filtros
- **Tercera Capa Convolucional:** 3N filtros
- **Cuarta Capa Convolucional:** 4N filtros

Este aumento progresivo en el número de filtros permite al modelo aprender características de bajo nivel en las primeras capas y características de alto nivel más complejas en las capas posteriores. Ajustaremos el valor de N para observar cómo afecta al rendimiento del modelo y analizar los resultados obtenidos.

### **Tamaño del Lote (Batch Size)**

El tamaño del lote determina el número de muestras que el modelo procesa antes de actualizar los parámetros. En nuestros experimentos, utilizamos un tamaño de lote de 1 debido a las limitaciones de memoria y el tamaño de las imágenes. Trabajar con un tamaño de lote de 1 puede hacer que el entrenamiento sea más ruidoso, pero puede ayudar a escapar de mínimos locales en algunos casos.

### **Número de Épocas (Epochs)**

El número de épocas define cuántas veces el modelo verá el conjunto completo de datos durante el entrenamiento. Hemos desarrollado un algoritmo de ajuste de épocas para comparar las métricas de entrenamiento y observar el progreso del modelo. El entrenamiento se detendrá en la época más eficiente, es decir, cuando se observe que el modelo ha alcanzado su máximo rendimiento y ya no está aprendiendo de manera significativa. Un número suficiente de épocas permite al modelo aprender patrones de los datos, pero se debe monitorear para evitar el sobreajuste, donde el modelo se desempeña bien en los datos de entrenamiento, pero no generaliza bien en datos no vistos.

### **Optimización**

Para optimizar el modelo, hemos utilizado el algoritmo Adam, que combina las ventajas de Adaptive Gradient Algorithm (AdaGrad) y Root Mean Square Propagation (RMSProp) [21]. Adam ajusta automáticamente la tasa de aprendizaje para cada parámetro, lo que lo hace robusto y eficiente para una amplia gama de problemas:

- **Algoritmo Adam:** Es conocido por su capacidad para manejar problemas con datos ruidosos y es adecuado para tareas de segmentación de imágenes debido a su capacidad para adaptarse dinámicamente a las condiciones cambiantes del gradiente.

Además de ajustar los parámetros mencionados, también hemos modificado el número de capas convolucionales y realizados ajustes en la estructura de nuestra red U-Net. Los detalles sobre estos ajustes y cómo afectan al rendimiento del modelo se explicarán más adelante en el apartado de resultados. Esto nos permite obtener el mejor resultado posible mediante la optimización y experimentación continuas.

## **5.3 Parámetros de entrenamiento U-Net**

Para entrenar el modelo U-Net, comenzaremos con experimentos para encontrar los mejores parámetros. Una vez establecidos estos parámetros, procederemos a ajustar la arquitectura de la U-Net para alcanzar la configuración óptima.

La estructura inicial de nuestra U-Net consta de tres bloques de codificación (encoders) y dos bloques de decodificación (decoder). Incluye una capa de upsampling para reconstruir la resolución original, como se mencionó anteriormente, y una capa final de clasificación. En la fase de codificación, cada bloque está compuesto por 2 capas convolucionales con padding tipo 'same' seguido de una función ReLU después de cada convolución, y se realiza una conexión residual (skip connection) con la salida inicial de la primera convolución. En la fase de decodificación, cada bloque sigue la misma estructura mencionada anteriormente.

El factor de escala para las dimensiones de la imagen es de 2, lo que incrementa el número de filtros en la fase de codificación y reduciendo el tamaño. En la fase de decodificación, este proceso se invierte para obtener la misma dimensión de salida que la entrada original. Posteriormente, se aplica una convolución final para ajustar y preparar las características de la salida, seguida de una función softmax para obtener las probabilidades por clase. Esta configuración inicial nos proporciona una base sólida para explorar y mejorar la arquitectura de la U-Net.

Los experimentos se dividirán inicialmente en tres partes, cada una variando uno de los tres ajustes principales de nuestro modelo: la tasa de aprendizaje, el número de filtros y las dimensiones de entrada. Entrenaremos cada modelo ajustando estos parámetros y realizaremos pruebas para observar los resultados de cada configuración. Para organizar y explicar el proceso, comenzaremos describiendo los experimentos relacionados con la tasa de aprendizaje, luego abordaremos el número de filtros y, finalmente, las dimensiones de entrada.

Es importante tener en cuenta que los cambios en cada parámetro afectan al rendimiento global del modelo. Aunque se han realizado pruebas con todas las combinaciones posibles de estos parámetros, en este documento se incluirán únicamente los resultados que tuvieron un impacto significativo en el proyecto.

### **Tasa de aprendizaje:**

La tasa de aprendizaje es uno de los parámetros clave que afecta de manera general a todos los ajustes y cambios que se realicen en el modelo. En los experimentos realizados, se observó que, al variar la tasa de aprendizaje, el valor óptimo de esta se mantenía constante a través de todas las combinaciones posibles de ajustes previamente mencionados. Esta consistencia sugiere que hay un valor óptimo de tasa de aprendizaje que es universalmente aplicable a nuestra arquitectura, independientemente de otros parámetros ajustados.

Con esta observación, se decidió mantener la misma tasa de aprendizaje para todos los ajustes subsiguientes. En consecuencia, los resultados se presentan utilizando un tamaño de entrada fijo de 512x512 píxeles y 64 filtros. Este enfoque permite estandarizar el proceso experimental y facilita la comparación directa de resultados, asegurando así la validez de los experimentos realizados.

**Tabla 1.3:** Valores de las métricas de evaluación de entrenamiento cuantitativas en la época más óptima del modelo. Lr: Tasa de aprendizaje, PE: Pérdida de entrenamiento, PV: Pérdida de validación, DE: Dice de entrenamiento, DV: Dice de validación.

<b>Lr</b>	<b>PE</b>	<b>PV</b>	<b>DE</b>	<b>DV</b>
1e-4	0.375	0.433	0.745	0.567
<b>5e-4</b>	<b>0.385</b>	<b>0.422</b>	<b>0.645</b>	<b>0.573</b>
1e-6	0.956	0.978	0.051	0.498

En la Tabla 1.1 se presentan los tres valores de tasa de aprendizaje más significativos obtenidos en los experimentos. Estos valores corresponden a la época final de cada modelo tras su entrenamiento, es decir, la época óptima. Para analizar estos valores, es fundamental que las métricas de validación y entrenamiento tengan la menor diferencia posible entre ellas, que decrezcan en el caso de las pérdidas y que incrementen en el caso de la métrica Dice a lo largo de las épocas.

Al observar los valores iniciales de las tasas de aprendizaje de 1e-4 y 5e-4, se nota que los resultados son bastante buenos, con valores cercanos a 0 para las pérdidas y cercanos a 1 para la métrica Dice. Cabe destacar que alcanzar un valor alto en la métrica Dice es bastante complicado, y un rango de 0.6-0.8 se considera un buen resultado. Ambos valores de tasa de aprendizaje muestran resultados prometedores dentro de este rango.

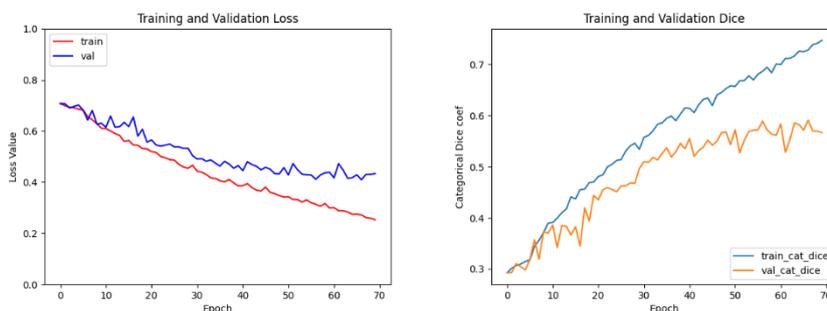
Sin embargo, se observa que la diferencia en los valores de la métrica Dice es menor para la tasa de aprendizaje de 5e-4, lo que sugiere una mejor consistencia y generalización del modelo. Para llegar a una conclusión final, se examinan los valores para cada época de manera gráfica para

obtener más información detallada. Aun así, como primer enfoque, el mejor valor parece ser el de  $5e-4$ .

Es importante destacar un problema en los entrenamientos que debemos tener en cuenta. Si la diferencia entre las métricas de entrenamiento y validación es alta, esto indicará un sobreajuste (overfitting), lo que significa que el modelo funciona bien con el conjunto de entrenamiento, pero falla en el conjunto de validación. Por otro lado, si la diferencia entre las dos métricas es pequeña y no aumenta el valor de la métrica, esto implicará un subajuste (underfitting), señalando que el modelo no ha aprendido lo suficiente.

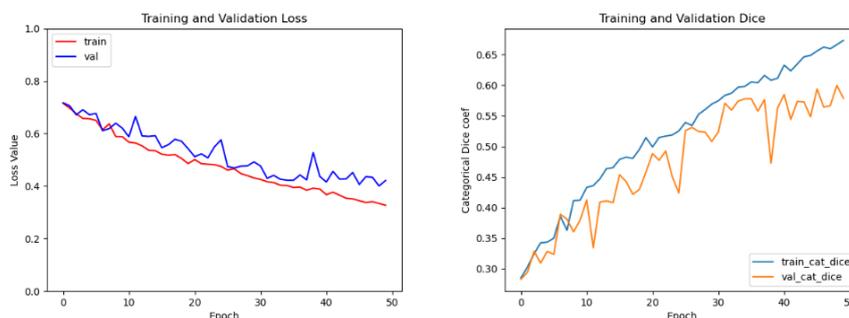
Utilizando una tasa de aprendizaje de  $1e-6$ , observamos un caso claro de subajuste, ya que no hay diferencia alguna entre las métricas de entrenamiento y validación. Esto indica que el modelo no está capturando las características necesarias del conjunto de datos. Posteriormente, examinaremos las gráficas de manera más cualitativa para visualizar y entender mejor el problema de subajuste, lo que confirma que el modelo no está aprendiendo adecuadamente.

- **Tasa de aprendizaje de  $1e-4$**



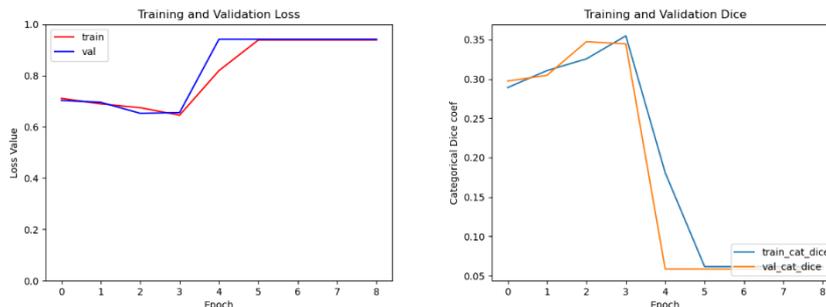
**Figura 5.1:** Grafica con métricas validación con lr  $1e-4$ .

- **Tasa de aprendizaje de  $5e-4$**



**Figura 5.2:** Grafica con métricas validación con lr  $5e-4$ .

- Tasa de aprendizaje de  $1e-6$



**Figura 5.3:** Grafica con métricas validación con lr  $1e-6$ .

Las gráficas muestran claramente que una tasa de aprendizaje demasiado baja provoca que nuestro modelo no aprenda correctamente y no pueda converger, lo que resulta en subajuste (**ver Figura 5.3**). En las **Figuras 5.1 y 5.2** se observa cómo el modelo aprende adecuadamente y converge. Los picos en algunas épocas se deben a la limitada cantidad de datos de entrada; más adelante exploraremos técnicas para abordar este problema y sus efectos en nuestro modelo.

Además de estas pruebas, una observación importante es que al utilizar una tasa de aprendizaje de  $1e-3$  también se produce subajuste, es decir, nuestro modelo no aprende ni converge. Esto indica que tanto una tasa demasiado alta como una demasiado baja resultan en subajuste, destacando que los parámetros son específicos para cada modelo. Analizando los resultados y obviando los picos, se observa que una tasa de  $5e-4$  converge de manera más estable y produce resultados más precisos. Por lo tanto, como se mencionó anteriormente, esta será la tasa de aprendizaje utilizada en los demás experimentos y en nuestro modelo final.

### Numero de filtros:

A continuación, variamos el número de filtros del modelo para aumentar o disminuir el número de parámetros. Se realizaron pruebas con tres modelos 128, 64 y 32 canales respectivamente.

**Tabla 1.4:** Valores de las métricas de evaluación de entrenamiento cuantitativas en la época óptima del modelo. NF: Numero de filtros, PE: Perdida de entrenamiento, PV: Perdida de validación, DE: Dice de entrenamiento, DV: Dice de validación.

NF	PE	PV	DE	DV
128	0.946	0.968	0.065	0.069
<b>64</b>	<b>0.385</b>	<b>0.422</b>	<b>0.645</b>	<b>0.573</b>
32	0.378	0.465	0.657	0.567

Con 128 canales, el modelo tiene 6,953,856 parámetros y claramente no aprende de manera efectiva. Las métricas en la **Tabla 1.2** muestran un subajuste, y en la **Figura 5.4** se puede observar este problema de manera cualitativa. Al comparar modelos con 64 y 32 canales (**Figura 5.5 para 32 canales y Figura 5.2 para 64 canales**), se observa que las diferencias en los resultados no son significativas. Sin embargo, el modelo con 32 canales tiene menos parámetros que el de 64, lo que implica un menor costo computacional para obtener una segmentación de tejido similar. Durante la búsqueda del mejor modelo, se considerarán ambos números de canales (32 y 64) para realizar las pruebas correspondientes.

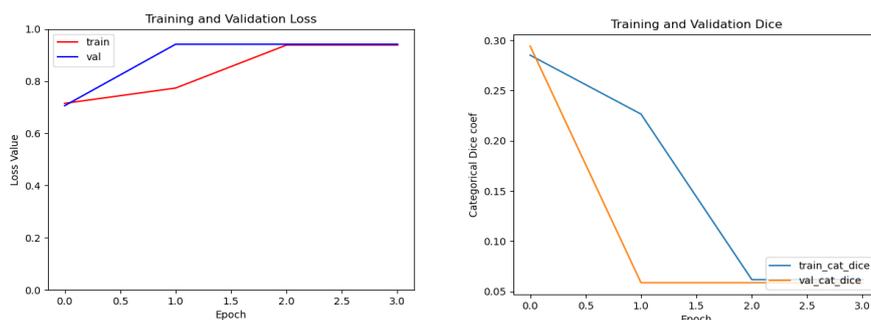


Figura 5.4: Grafica con métricas validación con 128 canales

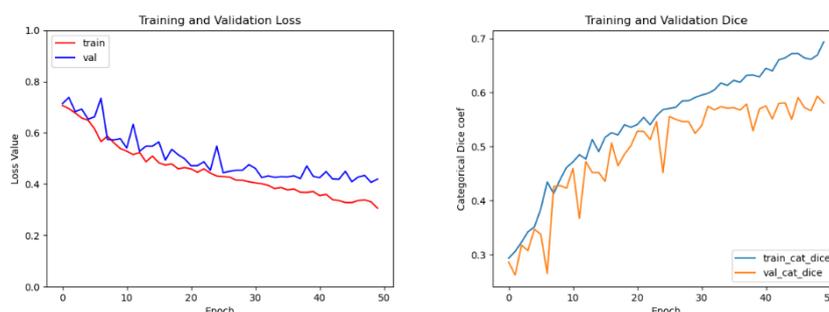


Figura 5.5: Grafica con métricas validación con 32 canales

### Tamaño de la entrada:

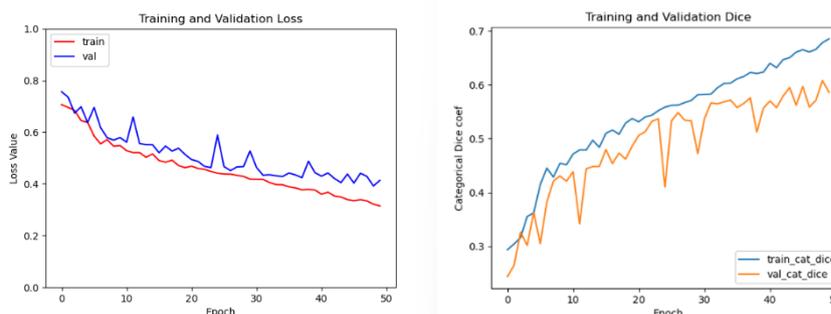
Para completar la optimización de los parámetros de entrenamiento, se han evaluado los resultados obtenidos con distintos tamaños de entrada: 1024x1024, 512x512 y 256x256.

Inicialmente, la idea de tener un tamaño de entrada mayor, es decir, 1024x1024, teóricamente debería proporcionar más características detalladas, lo que podría mejorar la precisión de nuestro modelo. Sin embargo, en nuestra evaluación, hemos observado que aumentar el tamaño de entrada no tiene un impacto significativo, ni positivo ni negativo, en el rendimiento del modelo (**ver Figura 5.6 y Tabla 1.3**). Lo que sí aumenta considerablemente es el costo computacional.

Finalmente, tras realizar los entrenamientos y observar las métricas, con un tamaño de 256x256 observamos el mismo problema de subajuste que hemos tenido anteriormente.

**Tabla 1.5:** Valores de las métricas de evaluación de entrenamiento cuantitativas en la época óptima del modelo. TE: Tamaño de entrada, PE: Perdida de entrenamiento, PV: Perdida de validación, DE: Dice de entrenamiento, DV: Dice de validación.

TE	PE	PV	DE	DV
1024x1024	0.373	0.412	0.689	0.563
<b>512x512</b>	<b>0.385</b>	<b>0.422</b>	<b>0.645</b>	<b>0.573</b>
256x256	0.956	0.968	0.051	0.047



**Figura 5.6:** Grafica con métricas validación con input de 1024x1024 32 canales

En conclusión, para la optimización de los parámetros de entrenamiento se considerarán tanto el tamaño de entrada de 512 y 1024 píxeles, así como configuraciones con 64 o 32 canales. Mantendremos el valor de la tasa de aprendizaje en  $5e-4$ . Además de las pruebas mencionadas, se han explorado diversas combinaciones adicionales, confirmando los resultados previamente discutidos.

En la siguiente sección, ajustaremos la arquitectura del modelo para conseguir los resultados deseados. Evaluaremos los resultados utilizando las métricas de entrenamiento y las métricas de evaluación cuantitativas Jaccard y Dice, realizando la media de las clases para identificar la mejor estructura posible. Además, superpondremos las máscaras obtenidas de la salida de nuestros modelos para verificar de manera cualitativa, comparando las imágenes anotadas por los patólogos con las máscaras generadas.

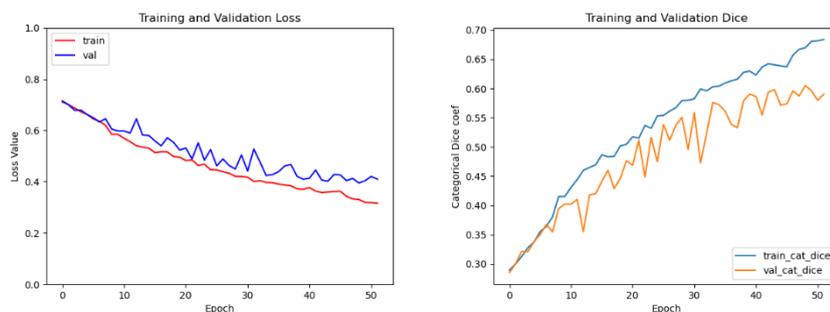
### 5.3.1 Evaluación y Ajuste de la Arquitectura del Modelo

Para ajustar la arquitectura de nuestro modelo, se realizarán diversas pruebas modificando la estructura de distintas maneras. Las modificaciones que más han impactado en nuestro proyecto incluyen la adición de bloques de codificación y decodificación, así como la incorporación de capas convolucionales en estos bloques. Dado el gran número de pruebas, no es práctico comentar cada una de ellas en detalle. Por ello, se explicarán las estructuras más relevantes junto con sus parámetros correspondientes y los resultados obtenidos. A continuación, hablaremos de cinco estructuras distintas explicadas en la **Tabla 1.4**.

**Tabla 1.6:** Valores de la estructura de cada Modelo evaluado. Modelos: Modelos más relevantes, BC: Bloques de codificación, BD: Bloques de decodificación, CC: Capas convolucionales por bloque, NF: Numero de filtros, TE: Tamaño de entrada.

Modelos	BC	BD	CC	NF	TE
Modelo_1	3	2	2	64	512x512
Modelo_2	3	2	3	64	512x512
Modelo_3	4	3	3	64	512x512
Modelo_4	3	2	3	32	512x512
Modelo_5	3	2	2	32	512x512
Modelo_6	3	2	2	32	1024x1024

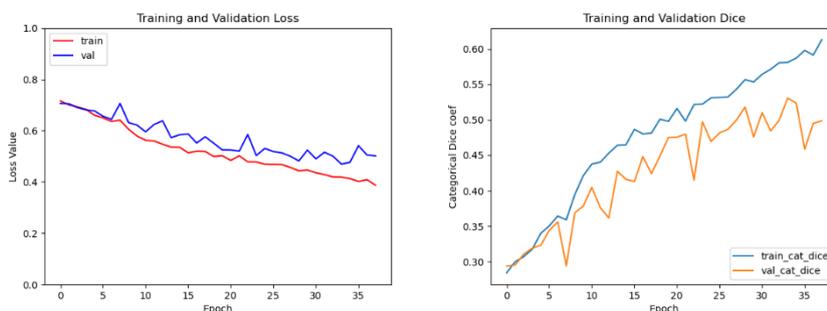
Comenzaremos con el análisis del entrenamiento del segundo modelo, ya que hemos evaluado los resultados del primer modelo anteriormente durante la optimización de parámetros. Más adelante, examinaremos los resultados del test. En este segundo modelo, se ha añadido una capa adicional tanto en los bloques de encoder como en los de decoder, lo que implica una capa extra de convolución. Se han utilizado 64 canales en este modelo, con un tamaño de entrada de 512x512 píxeles.



**Figura 5.7:** Graficas entrenamiento modelo 2.

Se observa una mejora en el entrenamiento, ya que los picos son más suaves y están más ajustados tanto en la validación como en el entrenamiento (**Figura 5.7**).

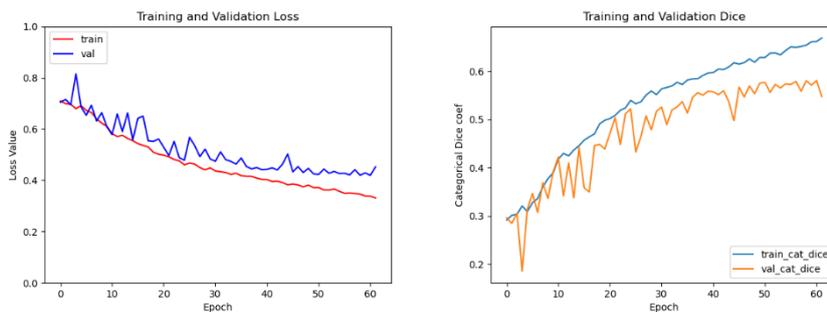
Para el modelo 3, se añadió un bloque adicional tanto en la codificación como en la decodificación, además de incluir tres capas convolucionales en cada bloque. Este modelo utiliza 64 canales, con un tamaño de entrada de 512x512 píxeles.



**Figura 5.8:** Graficas de entrenamiento modelo 3

En este modelo, las épocas disminuyen, lo que indica que la diferencia entre la validación y el entrenamiento supera el umbral óptimo antes que en las demás. También se observa que los picos son más abruptos y el modelo se sobreajusta con mayor rapidez (**Ver Figura 5.8**).

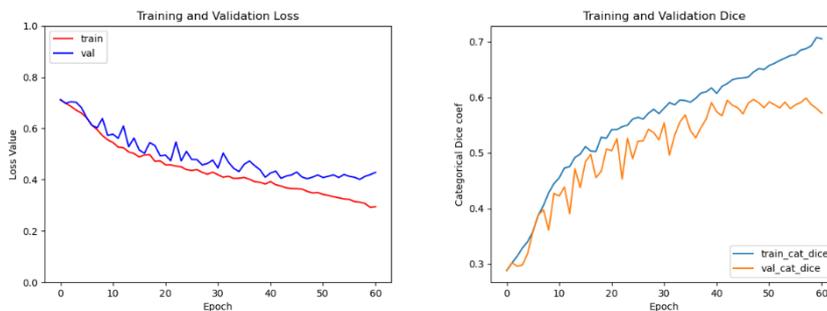
El modelo 4 utiliza 3 codificadores y 2 decodificadores, en las que en cada bloque se implementan 3 capas convolucionales. Se ha utilizado un tamaño de entrada de 512x512 píxeles y 32 canales.



**Figura 5.9:** Graficas de entrenamiento modelo 4

Los resultados obtenidos demuestran un mejor aprendizaje del modelo, lo que se traduce en líneas más ajustadas (**Ver Figura 5.9**).

El quinto modelo está estructurado con tres codificadores y dos decodificadores. Cada bloque del modelo incluye dos capas convolucionales. Se introducen imágenes de un tamaño de 512x512 píxeles y 32 canales.



**Figura 5.10:** Graficas de entrenamiento modelo 5

Las gráficas se asemejan a las del entrenamiento anterior, lo que sugiere una mejora en el aprendizaje del modelo. El último modelo tiene la misma estructura que la figura 3.5, ya que después de realizar varias pruebas, no encontramos que ningún modelo con una entrada de 1024x1024 obtuviera resultados mejores (**Ver Figura 5.10**).

En conclusión, se observa una mejora en algunos modelos mencionados debido a los cambios realizados en la estructura, aun así, no se puede observar claramente con las gráficas y métricas de entrenamiento que modelo es el que más se ajusta a nuestro problema.

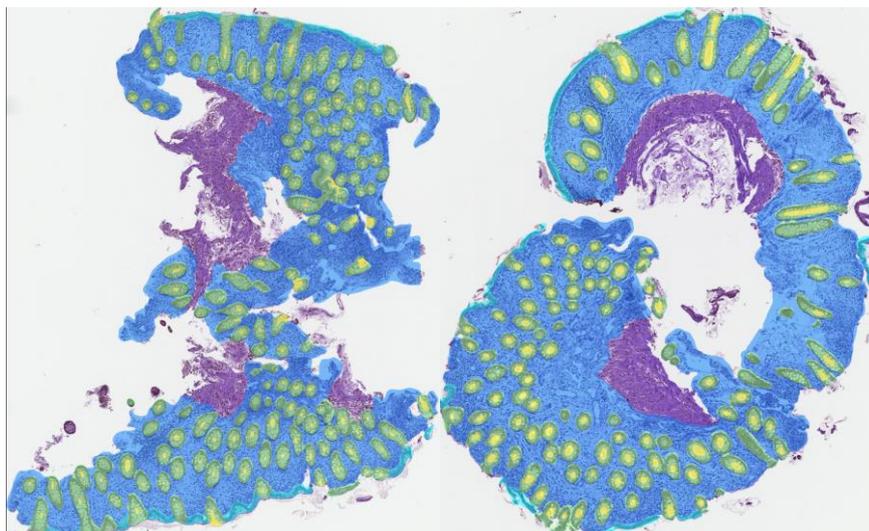
Además de los cambios mencionados, se realizaron ajustes en el factor aplicado a la conexión skip; sin embargo, no se observó ninguna mejora significativa. Se probaron ambas opciones de extracción de características, agrupación media y agrupación máxima, y se determinó que la agrupación media ofrece los mejores resultados. También se intentó aumentar la base de datos mediante técnicas de *data augmentation*, pero no se logró ninguna mejora, ya que nuestras imágenes contienen muchas características, lo cual limita este tipo de técnicas.

Como se explicó anteriormente, no es posible determinar cuál es el mejor modelo únicamente analizando las gráficas de entrenamiento. Es necesario utilizar métricas de validación para tomar una decisión informada. En este sentido, emplearemos las métricas IoU y Cat Dice, que fueron detalladas previamente, para ayudarnos a identificar el mejor modelo. A continuación, se presentan los resultados obtenidos en la Tabla 1.5.

**Tabla 1.7:** Resultados obtenidos de los modelos óptimos. Métricas de Cat Dice y IoU entre 0 y 1.

Modelos U-Net	Cat Dice	IoU Score
Modelo 1	0.5012	0.2985
<b>Modelo 2</b>	<b>0.6748</b>	<b>0.4648</b>
Modelo 3	0.4863	0.2812
<b>Modelo 4</b>	<b>0.6291</b>	<b>0.4176</b>
Modelo 5	0.5645	0.3201
Modelo 6	0.5289	0.352

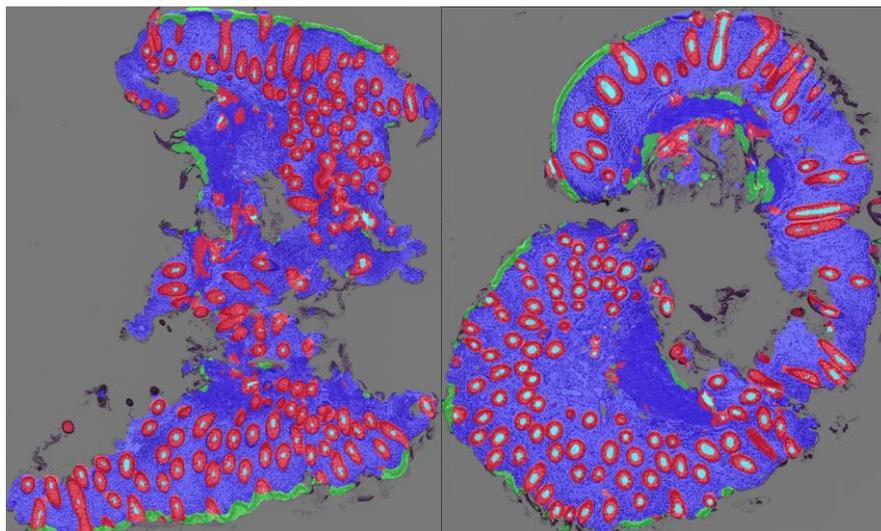
Los resultados obtenidos demuestran que los dos mejores modelos son el Modelo 2 y el Modelo 4. Ambos tienen una estructura de 3 codificadores y 2 decodificadores, con 3 capas convolucionales en cada bloque, y un tamaño de entrada de 512x512. La única diferencia es el tamaño de los canales: el modelo 2 tiene 64 canales y el modelo 4 tiene 32 canales. Para decidir cuál es el modelo final entre los dos, utilizaremos la máscara de salida del modelo y la superpondremos con la imagen sin anotar. Luego compararemos los resultados con las imágenes anotadas por los patólogos. Esta comparación nos ayudará a determinar cuál de los dos modelos es más preciso y se ajusta mejor a las necesidades del proyecto.



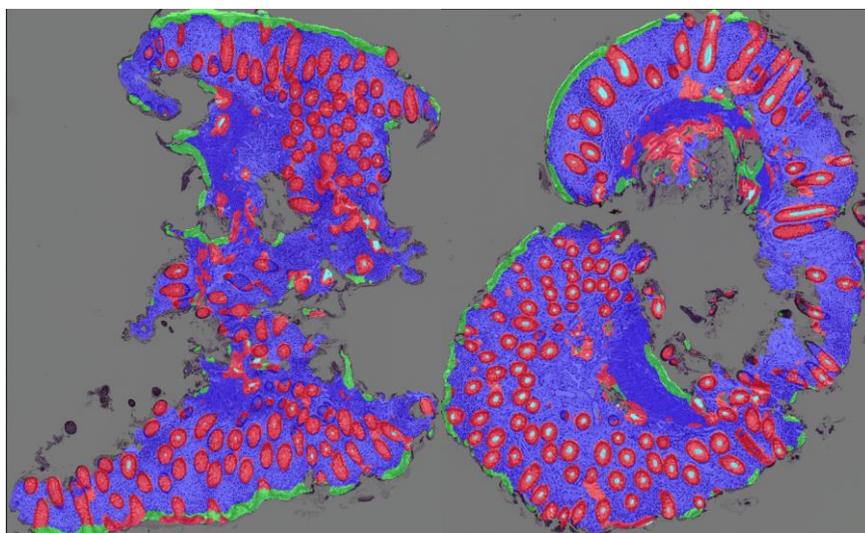
**Figura 5.10.1.** WSI's anotadas por patólogos.

En las imágenes anotadas por nuestro código el código de colores es el siguiente:

1. Sin Mascara: Fondo
2. Color Azul: Lamina propia
3. Color verde: Epitelio de la superficie
4. Color Rojo: Epitelio de la cripta
5. Color azul celeste: Lumen de la cripta



**Figura 5.11:** Imagen anotada por el Modelo 2



**Figura 5.12:** Imagen anotada por el Modelo 4

Al observar los resultados, la evaluación cualitativa sugiere que el Modelo 2 obtiene mejores resultados en comparación con el Modelo 4. Esto concuerda con los valores obtenidos de la evaluación de métricas de segmentación. Se puede observar que nuestro modelo anota más en algunas zonas en las que los patólogos no han anotado nada. Este problema fue comentado con los patólogos y se llegó a la conclusión de que es un error asumible en nuestro modelo. Se observa que la clase más complicada de segmentar es el lumen de la cripta debido a las pequeñas dimensiones de las anotaciones. En conclusión, el mejor modelo conseguido por el modelo U-Net es el modelo 2.

Siguiendo con la investigación del mejor modelo, utilizaremos una librería de GitHub que contiene distintos modelos de segmentación. El objetivo es determinar cuál es la mejor arquitectura posible. Utilizaremos nuestro código de la U-Net como código principal y compararemos los resultados obtenidos con nuestro modelo principal.

## 5.4 Resultados otras arquitecturas

A continuación, hablaremos de los resultados obtenidos de las arquitecturas explicadas en el apartado de metodología. Se explicará los distintos experimentos y arquitecturas utilizadas y compararemos de entre los tres modelos el mejor resultado posible.

### 5.4.1 U-Net 2.0

En este apartado se explicarán los resultados obtenidos del modelo U-Net 2.0. Los experimentos realizados con sus resultados obtenidos se observan en la **Tabla 1.6**.

**Tabla 1.8:** Resultados de test de los modelos con sus mejores estructuras, evaluados mediante las métricas IOU y Categorical Dice

Modelos	Encoder	Cat Dice	IoU
Modelo1_U-Net	Resnet50	0,6029	0,3568
Modelo2_U-Net	Resnet18	0,5843	0,3453
<b>Modelo3_U-Net</b>	<b>Resnet34</b>	<b>0,6043</b>	<b>0,3644</b>
Modelo4_U-Net	Resnet101	0,59	0,3523
Modelo5_U-Net	Vgg11	0,5811	0,3403
Modelo6_U-Net	Vgg11_bn	0,5811	0,3085
Modelo7_U-Net	Vgg13	0,6003	0,3514
Modelo8_U-Net	Vgg13_bn	0,535	0,3002

Los resultados demuestran que con el encoder de tipo ResNet34 obtenemos los mejores resultados con una estructura de 3 bloques de encoders, un tamaño de entrada de 512x512 píxeles, 32 canales y una tasa de aprendizaje de  $1e-4$ . A pesar de tener un encoder preentrenado y pesos inicializados con ImageNet para mejorar el entrenamiento, obtenemos peores resultados que con nuestro modelo inicial, a pesar de que comparten una arquitectura similar. Esto demuestra que los modelos más complejos no siempre producen mejores resultados; es encontrar el modelo específico adecuado para cada problema en particular. Este resultado resalta que una mayor complejidad no necesariamente implica mejores resultados.

Más adelante se analizará la superposición de las máscaras generadas por la salida del Modelo3\_U-Net, comparando estos resultados con los mejores modelos obtenidos de cada estructura.

### 5.4.2 U-Net Plus Plus

Se han realizado diversos experimentos con diferentes combinaciones para obtener las mejores estructuras y parámetros posibles. A continuación, se presentarán los mejores resultados obtenidos utilizando la arquitectura U-Net++ y se analizarán los resultados.

**Tabla 1.9:** Resultados de test de los modelos U-Net++ con sus mejores estructuras, evaluados mediante las métricas IOU y Categorical Dice.

Modelos	Encoder	Cat Dice	IoU
Modelo1_U-Net++	Resnet50	0,6282	0,3612
Modelo2_U-Net++	Resnet50	0,6219	0,3538
Modelo3_U-Net++	Resnet34	0,598	0,3378
Modelo4_U-Net++	Resnet101	0,5446	0,2989
<b>Modelo5_U-Net++</b>	<b>Vgg11</b>	<b>0,6448</b>	<b>0,3729</b>
Modelo6_U-Net++	Vgg11_bn	0,6161	0,3509

A simple vista, se observa una mejora en comparación con los valores anteriores; sin embargo, no supera los resultados obtenidos por el modelo principal. El mejor modelo identificado utiliza una VGG11 como encoder, con una entrada de 512x512, 4 encoders de tamaño de canal 64 y un learning rate de 1e-4. Un aspecto notable de los resultados son los dos modelos iniciales, ambos utilizando ResNet50 con los mismos parámetros y estructura, diferenciándose únicamente en el tamaño de entrada. El Modelo2\_U-Net++, con una entrada de 1024x1024 píxeles, no muestra una mejora significativa e incluso el Modelo1\_U-Net++ con una entrada inferior (512x512) obtiene mejores resultados. Esto refuerza la idea de que un aumento en los parámetros no necesariamente mejora el modelo.

### 5.4.3 Feature Pyramid Network (FPN)

Debido a la mayor complejidad estructural y el aumento de parámetros ajustables en esta arquitectura, se realizaron más pruebas y validaciones comparativas que en los modelos anteriores. A continuación, se presentan los resultados obtenidos:

**Tabla 1.10:** Resultados de test de los modelos FPN con sus mejores estructuras, evaluados mediante las métricas IOU y Categorical Dice.

Modelos	Encoder	Cat Dice	IoU
Modelo1_FPN	Resnet50	0,6435	0,4752
Modelo2_FPN	Resnet50	0,6554	0,4512
Modelo3_FPN	Resnet34	0,6466	0,4457
Modelo4_FPN	Resnet101	0,5994	0,4356
<b>Modelo5_FPN</b>	<b>Vgg11</b>	<b>0,6807</b>	<b>0,5171</b>
Modelo6_FPN	Vgg11_bn	0,6738	0,4851

Los resultados obtenidos muestran una notable mejora en comparación con todos los resultados anteriores del estudio. La mejor configuración se logró utilizando el encoder VGG11, con la siguiente estructura para este modelo:

**Tabla 1.11:** Configuración Modelo5\_FPN. TC: Tamaño del codificador, F: Numero de filtros inicial, NF: Numero de Filtros utilizados por cada capa, CD: Combinación de datos, D: Dropout, TE: Tamaño de entrada, Lr: Tasa de aprendizaje.

TC	F	NF	CD	D	TE	Lr
4	512	128	Concatenación	0.2	512x512	5e-4

El Modelo5\_FPN supera las métricas conseguidas por el primer modelo. Las métricas de evaluación ya están claras, pero para reforzar la fiabilidad de la decisión, se comparan las máscaras superpuestas. Esto nos permite tomar una decisión más fundamentada sobre el modelo óptimo en términos de precisión y rendimiento en la tarea de segmentación.

#### 5.4.4 Comparativa y Análisis de los Resultados

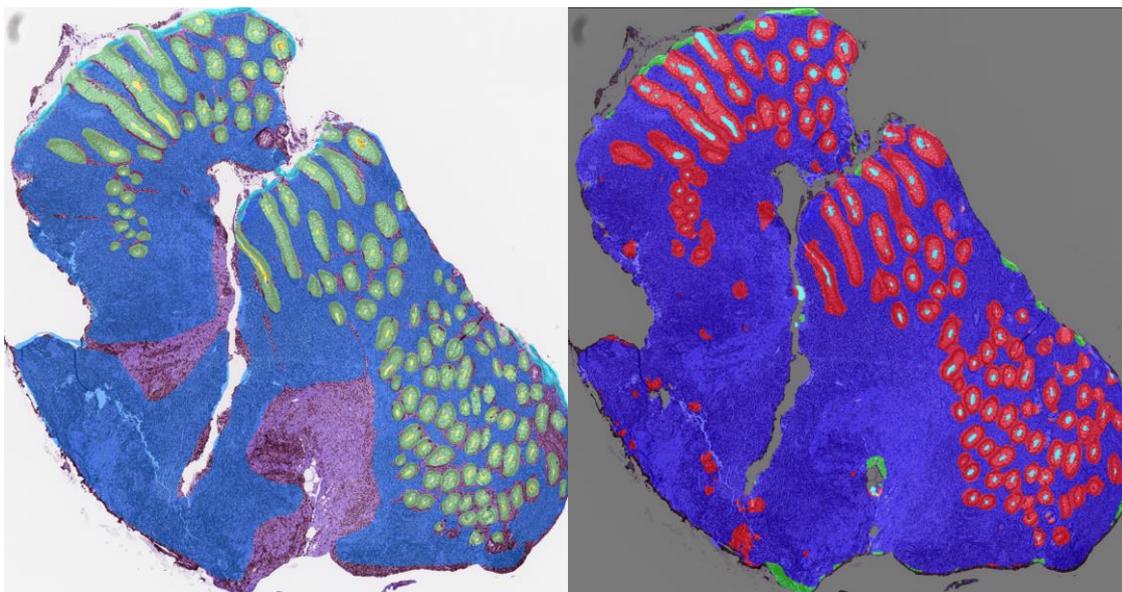
En este apartado, se presenta un análisis detallado de los resultados obtenidos mediante la evaluación de diferentes modelos de segmentación implementados. El objetivo principal es comparar y examinar críticamente el rendimiento de cada modelo con respecto a las métricas establecidas, así como mediante la comparación visual de las máscaras segmentadas. Esta evaluación no solo proporciona una visión clara de la efectividad de cada arquitectura y configuración probada, sino que también apunta a identificar el modelo más prometedor para el problema de segmentación. A continuación, se detallan los resultados de los mejores modelos de cada arquitectura en la siguiente tabla.

**Tabla 1.2:** Resultados de métricas de segmentación para los mejores modelos obtenidos.

Modelos	Cat Dice	IoU
Modelo Inicial U-Net	0,6748	0,4648
Modelo U-Net	0,6043	0,3644
Modelo U-Net ++	0,6448	0,3728
Modelo FPN	0,6807	0,5171

Para evaluar visualmente los resultados, realizaremos la inferencia de los modelos y compararemos las máscaras de salida superpuestas con las imágenes originales, contrastándolas con las anotadas por los patólogos.

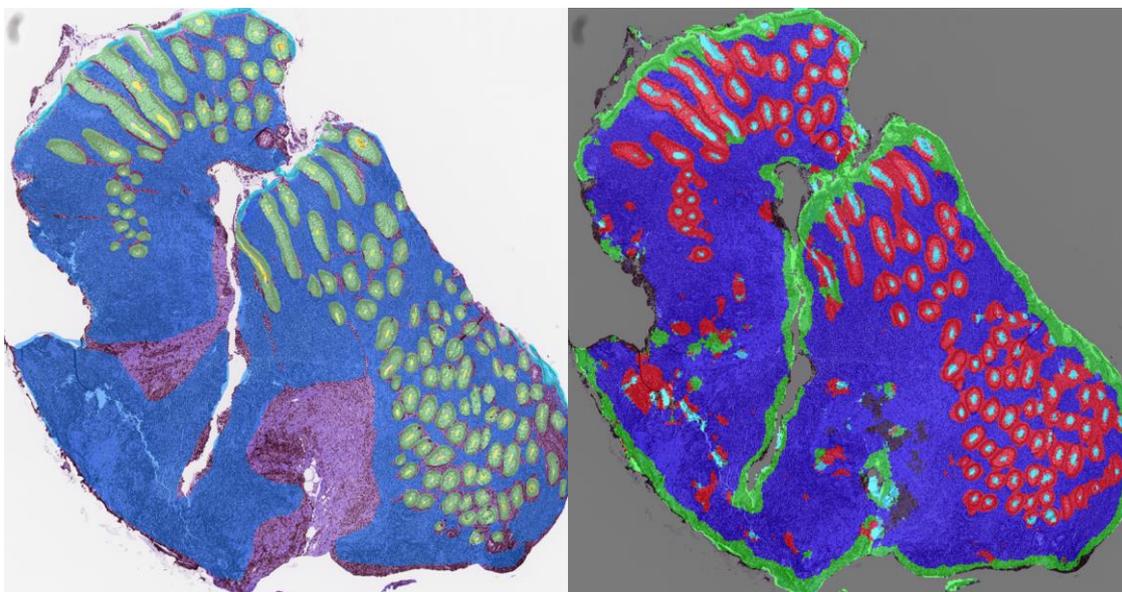
## 1. Modelo Inicial U-Net



**Figura 5.13:** Imagen WSIs del tejido. Derecha imagen anotada por el Modelo U-Net. Izquierda imagen anotada por patólogos.

El resultado obtenido ya ha sido analizado en el apartado 5.1.2, "*Evaluación y Ajuste de la Arquitectura del Modelo*", de este documento. Sin embargo, en esta imagen se observan errores en la detección del epitelio de las criptas (clase de color rojo) y del epitelio superficial (clase de color verde) en algunas partes.

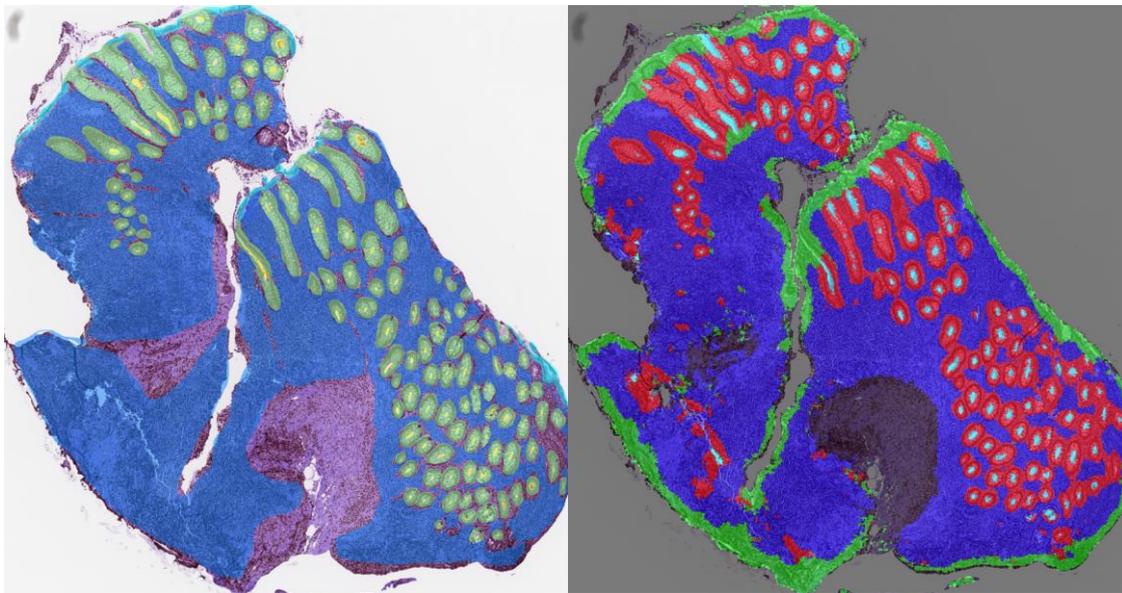
## 2. Modelo U-Net 2.0



**Figura 5.14:** Imagen WSIs del tejido. Derecha imagen anotada por el Modelo U-Net 2.0. Izquierda imagen anotada por patólogos.

Comparando las dos imágenes, **Figura 5.13**, observamos que los resultados de las métricas de análisis del modelo de segmentación coinciden con la máscara superpuesta. Se observa claramente que la segmentación no es tan precisa y hay errores evidentes en varias áreas de la imagen.

### 3. Modelo U-Net Plus Plus

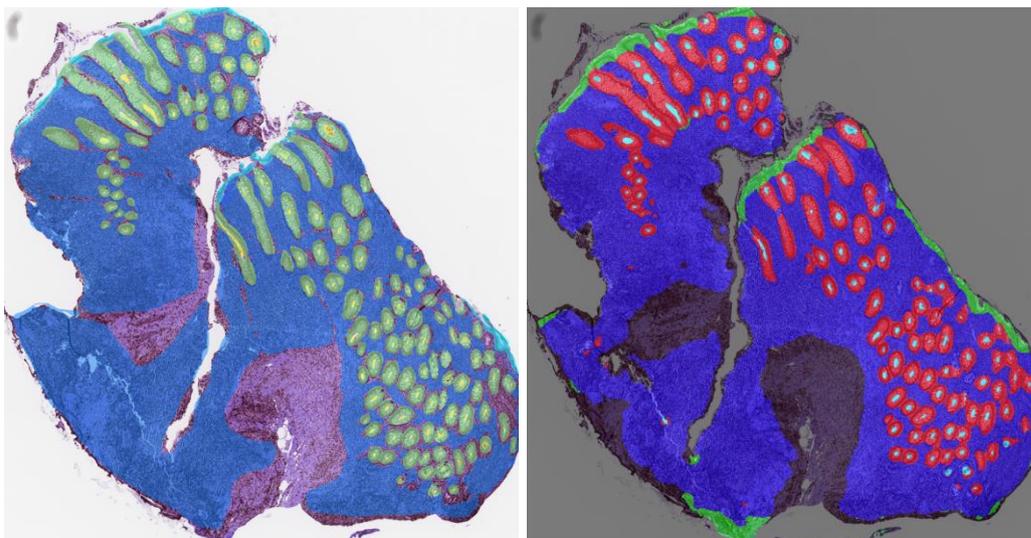


**Figura 5.15:** Imagen WSIs del tejido. Derecha imagen anotada por el Modelo U-Net PlusPlus librería. Izquierda imagen anotada por patólogos.

En la **Figura 5.15**, la mejora en comparación con el resultado anterior es evidente. El modelo U-Net++ muestra una segmentación superior de las diferentes clases y evita segmentar las zonas que no son de interés. Esto es crucial, ya que la capacidad del modelo para distinguir entre las zonas de interés y las de no interés es fundamental para nuestro problema. No se trata solo de segmentar la imagen por clases, sino de realizar una separación precisa entre las áreas relevantes y las irrelevantes.

Las métricas obtenidas, como el IoU (Intersección sobre Unión) y el Cat Dice (Categorical Dice), han mostrado un aumento significativo en comparación con el modelo U-Net básico, indicando una segmentación más precisa y consistente. Sin embargo, aún se pueden observar algunos errores en las imágenes, donde el modelo confunde ciertas clases en áreas específicas. Estos errores sugieren que, aunque el rendimiento ha mejorado, todavía existe margen para optimizar la segmentación y reducir la confusión entre clases similares.

#### 4. Modelo FPN



**Figura 5.16:** Imagen WSIs del tejido. Derecha imagen anotada por el Modelo FPN librería. Izquierda imagen anotada por patólogos.

El modelo FPN se destaca como el mejor en comparación con todos los anteriores en términos métricos. En las imágenes, se observa claramente que la segmentación es la más precisa de todas, logrando una segmentación más detallada de las áreas de interés y abarcando mayoritariamente todas las clases. Aunque se nota que el epitelio superficial (marcado en azul) no segmenta completamente el área de interés, este pequeño error puede ser aceptable, como se mencionó anteriormente. Esto sugiere que el modelo FPN ofrece una mejora significativa en la precisión y consistencia de la segmentación, destacándose por su capacidad para capturar con mayor claridad los detalles y estructuras relevantes en las imágenes histológicas analizadas.

Como análisis final de los resultados, podemos destacar que nuestro modelo inicial es bastante sólido, proporcionando valores precisos. Sin embargo, siempre es importante explorar más técnicas y modelos de segmentación para encontrar la solución más óptima. Considerando la limitación de una base de datos pequeña, se aplicaron técnicas de aumento de datos (data augmentation) en todas las estructuras mencionadas anteriormente, pero no se lograron mejoras significativas en los resultados. La estructura FPN, sin embargo, mejora considerablemente nuestra segmentación y obtiene resultados mucho mejores, además de tener un menor número de parámetros, lo que implica un menor tiempo de procesamiento computacional.



## Capítulo 6. Conclusiones y futuras líneas

### 6.1 Conclusiones

En este trabajo, hemos explorado la aplicación de la inteligencia artificial (IA) en la detección de colitis ulcerosa (CU) mediante el análisis de imágenes histológicas completas (Whole Slide Images, WSIs). La CU es una enfermedad inflamatoria crónica del intestino que afecta a millones de personas en todo el mundo, causando graves problemas de salud. La identificación precisa y temprana de los signos de CU en muestras histológicas es crucial para un tratamiento efectivo y para evitar complicaciones severas a largo plazo.

La implementación de la IA y los sistemas de diagnóstico asistido por computadora (CAD) pueden mejorar significativamente la precisión y eficiencia del diagnóstico, reducir la carga de trabajo de los patólogos y proporcionar una segunda opinión objetiva y reproducible. Uno de los principales desafíos a superar es la falta de índices de evaluación específicos para la CU que faciliten su implementación en estos sistemas.

El desarrollo del índice PHRI ha sido fundamental para este trabajo, ya que presenta características simples que permiten su fácil integración en sistemas CAD. Este índice se basa en la detección de neutrófilos en zonas específicas de interés, lo cual es crucial para un diagnóstico preciso de la colitis ulcerosa. La segmentación precisa de estas zonas es esencial debido a la dificultad en la detección de neutrófilos. Por esta razón, este trabajo se enfoca en segmentar las áreas de interés, facilitando así la posterior identificación de neutrófilos en dichas áreas.

Gracias al índice PHRI, es posible desarrollar modelos de segmentación que faciliten la identificación y análisis de las áreas afectadas por CU, mejorando así la precisión del diagnóstico y el tratamiento de esta enfermedad.

Durante el desarrollo de nuestro modelo de segmentación, se han utilizado diversas arquitecturas de redes neuronales convolucionales (CNN), ya que estas son las más utilizadas actualmente en el campo de la segmentación de imágenes. Inicialmente, se exploró y optimizó una red basada en la arquitectura U-Net, debido a su amplia implementación en aplicaciones de inteligencia artificial basadas en imágenes. A través de numerosos experimentos y mejoras, logramos obtener resultados sólidos. Sin embargo, dado el amplio abanico de arquitecturas disponibles y la necesidad de seguir mejorando, investigamos otros modelos de segmentación. Esto incluyó el uso de redes preentrenadas y estructuras más novedosas. A partir de esta investigación, se encontró que la arquitectura Feature Pyramid Network (FPN) es la óptima para nuestro problema. La FPN demostró ser capaz de extraer información precisa de las imágenes con un alto nivel de detalle y exactitud, superando los resultados obtenidos con otras arquitecturas.

A pesar de trabajar con una base de datos relativamente pequeña, el ajuste de hiperparámetros, estructuras y tipos de modelos ha sido crucial para mejorar la robustez y precisión de nuestros modelos. Gracias a la cuidadosa optimización de estos elementos y al exhaustivo nivel de pruebas realizadas, hemos logrado alcanzar resultados destacados. Estas técnicas permiten que el modelo generalice de manera más efectiva y sea menos propenso a sobreajustarse a los datos de entrenamiento, lo cual es esencial para su aplicación en un entorno clínico real. Es importante destacar que, como se mencionó anteriormente, incrementar la complejidad del modelo y aumentar la cantidad de datos no siempre garantiza mejoras. Encontrar el modelo más preciso que se ajuste al problema específico ha sido clave para lograr un rendimiento óptimo.

En conclusión, hemos desarrollado un modelo de segmentación basado en deep learning y redes neuronales convolucionales que segmenta con alta precisión las zonas de interés necesarias para la detección de neutrófilos. Esto contribuye significativamente en el proyecto PICaSSO en la mejora en la detección y diagnóstico de la CU utilizando el índice PHRI.

## 6.2 Futuras líneas

El trabajo continuará con la exploración de técnicas avanzadas de segmentación de las áreas de imágenes histológicas relacionadas con la colitis ulcerosa y la detección de neutrófilos en estas áreas específicas. Al utilizar este modelo como base para la segmentación de imágenes, se puede construir un sistema de diagnóstico altamente preciso, basado en la detección de neutrófilos.

Es fundamental ampliar y mejorar las bases de datos existentes mediante la colaboración con centros de investigación, para asegurar una mayor diversidad y representatividad de las muestras. La ampliación de las bases de datos permitirá entrenar modelos más robustos y precisos. Para alcanzar este objetivo, es esencial que las anotaciones de las imágenes sean precisas y estandarizadas, utilizando el índice PHRI para asegurar la consistencia en los datos y facilitar la continuación del proyecto PICaSSO. Dado que el proceso de anotación es costoso, otra forma efectiva de realizar estas anotaciones es mediante la técnica de *Active Learning*.

*Active Learning* es un enfoque en el que el modelo selecciona de manera inteligente las muestras más informativas del conjunto de datos no etiquetado para ser anotadas por un experto. En lugar de etiquetar un gran volumen de datos de manera aleatoria, el modelo identifica aquellos ejemplos para los cuales tiene mayor incertidumbre o que proporcionarán el mayor valor informativo. Esto permite reducir la cantidad de datos etiquetados necesarios y optimizar el uso de recursos, mejorando la eficiencia del proceso de anotación.

El modelo de segmentación se centrará en las áreas de interés para su posterior aplicación en la detección de neutrófilos en las imágenes histológicas, aprovechando las capacidades de arquitecturas de redes neuronales como U-Net y FPN para lograr una segmentación precisa y eficiente. Además, el proyecto busca fusionar esta técnica con el diagnóstico mediante endoscopias. Al combinar la detección de neutrófilos en imágenes histológicas con los resultados obtenidos a través de endoscopias, se podrán utilizar ambos valores de manera complementaria para diagnosticar y detectar la colitis ulcerosa con mayor precisión y fiabilidad.

Esta fusión de técnicas permitirá un enfoque diagnóstico más integral, aprovechando la capacidad de las endoscopias para proporcionar una visión detallada del estado del tracto gastrointestinal y la precisión de la segmentación de imágenes histológicas para identificar la presencia de neutrófilos. Esta combinación no solo mejorará la precisión del diagnóstico de la CU, sino que también facilitará una mejor personalización de los tratamientos, adaptándolos a las necesidades específicas de cada paciente basándose en un análisis más completo y detallado de su condición.

En definitiva, el avance en las técnicas de segmentación de imágenes histológicas y la integración con diagnósticos endoscópicos representa un paso significativo hacia una detección más precisa y rápida de la colitis ulcerosa. Esto contribuirá al avance del campo de la medicina, permitiendo diagnósticos más consistentes y personalizados, mejorando así los resultados clínicos y la calidad de vida de los pacientes.



## Capítulo 7. Bibliografía

- [1] Santiago García López. "Crohn y Colitis Ulcerosa". En: *Confederación ACCU Crohn y Colitis Ulcerosa* (2019). URL: <https://accuesp.com/crohn-y-colitis/la-enfermedad/sintomas>.
- [2] Ryan Ungaro y otros. "Colitis ulcerosa". En: *The Lancet* 389.10080 (2017), pp. 1756-1770. URL: <https://www.sciencedirect.com/science/article/pii/S0140673616321262>.
- [3] Biblioteca Nacional de Medicina de los EE.UU (2023). MedlinePlus. En: Intestino grueso. URL: [https://medlineplus.gov/spanish/ency/esp\\_imagepages/19220.htm](https://medlineplus.gov/spanish/ency/esp_imagepages/19220.htm).
- [4] Virginia Ollero Pena, Complejo Hospitalario Universitario de Ferrol. En: *II.1.2.2 Endoscopia en el seguimiento de la colitis ulcerosa* (2023). URL: <https://endi.geteccu.org/endoscopia-en-el-seguimiento-de-la-colitis-ulcerosa/>.
- [5] Sandhya Pruthi. Mayo Clinic. En: *Biopsia: Algunos tipos de biopsias que se utilizan para diagnosticar el cáncer* (2023). URL: <https://www.mayoclinic.org/es/diseases-conditions/cancer/in-depth/biopsy/art-20043922>.
- [6] Paredes Méndez, Juan Eloy, Alosilla Sandoval, Paulo Anibal, Quiroga Purizaca, Wilmer Gustavo, Mestanza Rivasplata, Ana Lucía, Cieza Mestanza, Victor, Vargas Marcacuzco, Henry, Lazo Molina, Luis, & Arenas Gamio, Jose Luis. (2019). Evaluación de actividad endoscópica e histológica como predictores de recaída clínica en colitis ulcerativa. *Revista de Gastroenterología del Perú*, 39(3), 222-228. Recuperado en 15 de julio de 2024, de [http://www.scielo.org.pe/scielo.php?script=sci\\_arttext&pid=S102251292019000300004&lng=e&tlng=es](http://www.scielo.org.pe/scielo.php?script=sci_arttext&pid=S102251292019000300004&lng=e&tlng=es).
- [7] Dr. Ivan Rivas, Medico Internista y Gastroenterólogo. En: *Gastroenterología. Índices endoscópicos en enfermedad inflamatoria intestinal* (2021). URL: <http://www.ivanrivasmd.com/indices-endoscopicos-en-enfermedad-inflamatoria-intestinal/>
- [8] Mosli MH, Feagan BG, Zou G, *et al* Development and validation of a histological index for UC. *Gut* 2017;66:50-58. URL: <https://gut.bmj.com/content/66/1/50>.
- [9] Gui X, Bazarova A, Del Amor R, Vieth M, de Hertogh G, Villanacci V, Zardo D, Parigi TL, Røyset ES, Shivaji UN, Monica MAT, Mandelli G, Bhandari P, Danese S, Ferraz JG, Hayee B, Lazarev M, Parra-Blanco A, Pastorelli L, Panaccione R, Rath T, Tontini GE, Kiesslich R, Bisschops R, Grisan E, Naranjo V, Ghosh S, Iacucci M. PICaSSO Histologic Remission Index (PHRI) in ulcerative colitis: development of a novel simplified histological score for monitoring mucosal healing and predicting clinical outcomes and its applicability in an artificial intelligence system. *Gut*. 2022 May;71(5):889-898. doi: 10.1136/gutjnl-2021-326376. Epub 2022 Feb 16. PMID: 35173041; PMCID: PMC8995819. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8995819/>
- [10] Instituto Nacional del Cáncer. Sitio oficial del Gobierno de los Estados Unidos. En: *Diccionarios del NCI. Diccionario del cáncer NCI. Glóbulo Blanco*. URL: <https://www.cancer.gov/espanol/publicaciones/diccionarios/diccionario-cancer/def/globulo-blanco>
- [11] Dr. Santiago Nova. KenHub. En: *Células Sanguíneas* (2023). URL: <https://www.kenhub.com/es/library/anatomia-es/celulas-sanguineas>.
- [12] Mariano Rivera, Aprendizaje Automático. En: *Perceptrón Multicapa en Tensorflow-Keras* (2023). URL: [https://www.cimat.mx/~mrivera/cursos/aprendizaje\\_profundo/mlp/mlp.html](https://www.cimat.mx/~mrivera/cursos/aprendizaje_profundo/mlp/mlp.html).
- [13] Valery Naranjo, D. (2023). Introducción al aprendizaje automático en aplicaciones de telecomunicación, UPV (14705). En: *Deep Learning*.
- [14] IBM. (2023). *Re: ¿Qué son las redes neuronales convolucionales?* URL: <https://normas-apa.org/referencias/ejemplos/>.

- [15] Iakubovskii, Pavel. (2019). En: *Segmentation Models Pytorch*, *GitHub*. URL: [https://github.com/qubvel/segmentation\\_models.pytorch](https://github.com/qubvel/segmentation_models.pytorch).
- [16] Luft, Harrison & Schillaci, Calogero & Ceccherini, Guido & Vieira, Diana & Lipani, Aldo. (2022). Deep Learning Based Burnt Area Mapping Using Sentinel 1 for the Santa Cruz Mountains Lightning Complex (CZU) and Creek Fires 2020. *Fire*. 5. 163. 10.3390/fire5050163. URL: [https://www.researchgate.net/publication/364497833\\_Deep\\_Learning\\_Based\\_Burnt\\_Area\\_Mapping\\_Using\\_Sentinel\\_1\\_for\\_the\\_Santa\\_Cruz\\_Mountains\\_Lightning\\_Complex\\_CZU\\_and\\_Creek\\_Fires\\_2020](https://www.researchgate.net/publication/364497833_Deep_Learning_Based_Burnt_Area_Mapping_Using_Sentinel_1_for_the_Santa_Cruz_Mountains_Lightning_Complex_CZU_and_Creek_Fires_2020).
- [17] DataScientest (2024). En: *U-Net: todo lo que tienes que saber sobre la red neuronal de Computer Vision*. URL: <https://datascientest.com/es/u-net-lo-que-tienes-que-saber>.
- [18] Hesaraki, Saba. Medium (2023). En: *U-NET++*. URL: <https://medium.com/@saba99/U-Net-443b429ae0be>.
- [19] Tsung-Yi Lin and Piotr Dollár and Ross Girshick and Kaiming He and Bharath Hariharan and Serge Belongie (2017). En: *Feature Pyramid Networks for Object Detection*. URL: <https://arxiv.org/abs/1612.03144>.
- [20] Ordas Jiménez, I. (2020). *Enfermedad Inflamatoria Intestinal*. Clínic de Barcelona. URL: [https://www.clinicbarcelona.org/asistencia/enfermedades/enfermedad-inflamatoria-intestinal?\\_gl=1\\*16wfy9e\\*\\_up\\*MQ..\\*\\_ga\\*MTM3NzY5NDcyMS4xNzE4MjcyMzQ5\\*\\_ga\\_R4X4292K2Y\\*MTcxODI3MjM0OC4xLjEuMTcxODI3MjM4OC4wLjAuMA](https://www.clinicbarcelona.org/asistencia/enfermedades/enfermedad-inflamatoria-intestinal?_gl=1*16wfy9e*_up*MQ..*_ga*MTM3NzY5NDcyMS4xNzE4MjcyMzQ5*_ga_R4X4292K2Y*MTcxODI3MjM0OC4xLjEuMTcxODI3MjM4OC4wLjAuMA).
- [21] InteractiveChaos. *Introducción al Machine Learning, Adam*. InteractiveChaos making things simple URL: <https://interactivechaos.com/es/manual/tutorial-de-machine-learning/adam>

## ANEXOS

### ANEXO A. Impacto del proyecto en los objetivos de la Agenda 2030

#### Relación trabajo fin de grado con los Objetivos de Desarrollo Sostenible 2030

Objetivos de Desarrollo Sostenibles ODS	No aplicable	Bajo	Medio	Alto
1. Fin de la pobreza	✓			
2. Hambre cero	✓			
3. Salud y bienestar				✓
4. Educación de calidad	✓			
5. Igualdad de género	✓			
6. Agua limpia y saneamiento	✓			
7. Energía asequible y no contaminante	✓			
8. Trabajo decente y crecimiento económico	✓			
9. Industria. Innovación e infraestructura				✓
10. Reducción de las desigualdades	✓			
11. Ciudades y comunidades sostenibles	✓			
12. Producción y consumo responsable	✓			
13. Acción por el clima	✓			
14. Vida submarina	✓			
15. Vida de ecosistemas terrestres	✓			
16. Paz, justicia e instituciones sólidas	✓			
17. Alianzas para lograr los objetivos	✓			

En este trabajo de fin de grado, he desarrollado un modelo de segmentación para ayudar en la detección de la colitis ulcerosa, contribuyendo al **Objetivo de Desarrollo Sostenible 3: Salud y Bienestar** de la Agenda 2030. Gracias a esta segmentación de tejidos, es posible agilizar el proceso de detección de la colitis ulcerosa, permitiendo que el diagnóstico sea más rápido y preciso. Esta técnica también tiene el potencial de mejorar, en el futuro, la detección de neutrófilos en el trabajo de PiCaSSO avanzando así en el diagnóstico y tratamiento de la CU. Además, este proyecto se alinea con el **Objetivo de Desarrollo Sostenible 9: Industria, Innovación e Infraestructura**, ya que el uso de inteligencia artificial y modelos de segmentación avanzados promueve la innovación tecnológica en el campo médico.