ULPGC
Universidad de
Las Palmas de
Gran Canaria

ESCUELA DE INGENIERÍA
INFORMÁTICA

Grado en Ciencia e Ingeniería de Datos

Trabajo Fin de Grado

Aplicación de Redes Neuronales Multimodales para la Segmentación y Clasificación de Tumores en Imágenes Médicas

Laura Lasso García

Tutores

Javier Sánchez Pérez

Las Palmas de Gran Canaria
10 de julio de 2025

ULPGC Universidad de Las Palmas de Gran Canaria | EII ESCUELA DE INGENIERÍA INFORMÁTICA

TFT04

SOLICITUD DE DEFENSA DE TRABAJO DE FIN DE TÍTULO

D./Dª **Laura Lasso García**, autor del trabajo **Aplicación de Redes Neuronales Multimodales para la Segmentación y Clasificación de Tumores en Imágenes Médicas**, correspondiente a la titulación **Grado en Ciencia e Ingeniería de Datos**.

SOLICITA

que se inicie el procedimiento de defensa del mismo, para lo que se adjunta la documentación requerida, haciendo constar que

[X] se autoriza / [ ] no se autoriza la grabación en audio de la exposición y turno de preguntas.

Asimismo, con respecto al registro de la propiedad intelectual/industrial del TFT, declara que:
[ ] Se ha iniciado o hay intención de iniciarlo (defensa no pública).
[X] No está previsto.

Y para que así conste firma la presente. (fecha en firma electrónica)

El/La estudiante

Fdo. Laura Lasso García

A rellenar y firmar **obligatoriamente** por el/la/los/las tutores

En relación con la presente solicitud, se informa:(firmar donde corresponda)

Positivamente (en caso de detección de copia, esta firma quedará invalidada)	Negativamente (justificación en TFT05)
Fdo. Javier Sánchez Pérez,	Fdo. Javier Sánchez Pérez,

DIRECTORA DE LA ESCUELA DE INGENIERÍA INFORMÁTICA

Aplicación de Redes Neuronales Multimodales para la Segmentación y Clasificación de Tumores en Imágenes Médicas

Trabajo Fin de Grado

Laura Lasso García

Tutores

Javier Sánchez Pérez

Grado en Ciencia e Ingeniería de Datos

Escuela de Ingeniería Informática

Universidad de Las Palmas de Gran Canaria

Las Palmas de Gran Canaria

10 de julio de 2025

Índice general

Índice de figuras v

Índice de tablas vi

Agradecimientos vii

Resumen viii

Abstract ix

1. Introducción 1
- 1.1. Motivaciones . . . 3
- 1.2. Objetivos . . . 4
  - 1.2.1. Objetivos del proyecto . . . 4
  - 1.2.2. Objetivos académicos . . . 4
- 1.3. Organización del documento . . . 5

2. Estado del arte 7
- 2.1. Técnicas tradicionales en análisis de imágenes médicas . . . 8
- 2.2. Enfoques basados en aprendizaje profundo . . . 9
  - 2.2.1. Clasificación y detección con redes convolucionales . . . 9
  - 2.2.2. Segmentación médica con U-Net y variantes . . . 10
- 2.3. Transformers visuales en imagen médica . . . 11
- 2.4. Modelos multimodales . . . 12
  - 2.4.1. Integración de imagen médica y datos clínicos . . . 12
  - 2.4.2. Fusión de distintas modalidades de imagen médica . . . 13
- 2.5. Predicción clínica mediante regresión . . . 13

3. Metodología y planificación del trabajo 15
- 3.1. Metodología . . . 15
- 3.2. Planificación . . . 17
  - 3.2.1. Planificación inicial . . . 17
  - 3.2.2. Tareas realizadas . . . 18
- 3.3. Herramientas utilizadas . . . 19
  - 3.3.1. Entorno de hardware . . . 20
  - 3.3.2. Entorno de software . . . 20
  - 3.3.3. Librerías empleadas . . . 20
  - 3.3.4. Otras herramientas . . . 21

4. Conjuntos de datos **22**
4.1. Brain Tumor Segmentation Challenge 2020 . . . 22
4.1.1. Descripción del conjunto de datos . . . 23
4.1.2. Preprocesamiento de las imágenes médicas . . . 24
4.1.3. Análisis estadístico de variables clínicas . . . 26
4.2. RM cerebral metastásica con segmentación y radiómica . . . 27
4.2.1. Estructura del conjunto de datos . . . 29
4.2.2. Evaluación y justificación de la exclusión del conjunto de datos . . . 30

5. Redes neuronales empleadas **31**
5.1. Red neuronal 3D U-NET . . . 31
5.1.1. Arquitectura de la red neuronal . . . 31
5.1.2. Estructura general . . . 33
5.2. Red neuronal LViT . . . 33
5.2.1. Arquitectura de la red neuronal . . . 34
5.2.2. Estructura general . . . 34
5.2.3. Adaptación para la predicción de días de supervivencia . . . 36
5.3. Red neuronal SegResNet derivada de la red 3D U-Net . . . 38
5.3.1. Arquitectura de la red neuronal . . . 38
5.3.2. Estructura general . . . 39
5.4. Red neuronal 3D AutoEncoder . . . 40
5.4.1. Arquitectura de la red neuronal . . . 40
5.4.2. Estructura general . . . 41
5.5. Modelado basado en representaciones latentes . . . 42
5.5.1. Gradient boosting . . . 43
5.5.2. Regresor perceptrón multicapa (MLP) . . . 43
5.5.3. Regresión random forest . . . 43
5.5.4. Regresión ridge . . . 44
5.5.5. Regresión de vectores de soporte . . . 44
5.5.6. Categorical boosting . . . 45
5.6. Otras redes neuronales empleadas . . . 45
5.6.1. Red neuronal UNETR . . . 45
5.6.2. Red neuronal SAM2 . . . 46

6. Configuración de experimentos **48**
6.1. Estrategia de división de datos . . . 48
6.2. Preparación de datos por tipo de modelo . . . 49
6.2.1. Preprocesamiento de datos para modelos de segmentación . . . 49
6.2.2. Preprocesamiento de datos para modelos de regresión . . . 50
6.3. Configuración del entrenamiento . . . 52
6.3.1. Funciones de pérdida . . . 52
6.3.2. Métricas de evaluación . . . 55
6.3.3. Hiperparámetros de entrenamiento . . . 56
6.3.4. Optimizadores y técnicas de mejora . . . 58

7. Resultados experimentales **59**
7.1. Resultados de segmentación 59
7.1.1. Curvas de entrenamiento 59
7.1.2. Análisis de resultados 60
7.1.3. Discusión comparativa 63
7.2. Resultados de regresión 65
7.2.1. Curvas de entrenamiento y validación 67
7.2.2. Análisis de resultados 71
7.2.3. Comparación entre modalidades y fusiones 77
7.2.4. Discusión sobre viabilidad clínica 77
7.3. Comparación global de modelos y tareas 78
7.3.1. Segmentación vs regresión: modelos multitarea 78
7.3.2. Rendimiento de modelos multimodales 79
7.3.3. Consideraciones sobre eficiencia computacional 80

8. Conclusiones y trabajo futuro **81**
8.1. Síntesis de resultados y cumplimiento de objetivos 81
8.2. Impacto y relevancia práctica 82
8.3. Lecciones aprendidas 82
8.4. Limitaciones del trabajo actual 83
8.5. Trabajo futuro 84
8.6. Reflexiones finales 85

A. Competencias específicas cubiertas **86**

B. Objetivos de desarrollo sostenible **88**

Acrónimos **90**

Glosario **93**

Bibliografía **107**

Índice de figuras

3.1. Diagrama de la metodología CRISP-DM. Fuente: Decidata [Decidata, 2023] . . 16

4.1. Máscara de segmentación de un paciente del conjunto de datos BraTS. Se visualizan las principales subregiones del glioma: necrosis (amarillo), correspondiente al tejido tumoral muerto; edema peritumoral (morado), que representa la acumulación de líquido inflamatorio alrededor del tumor; y tumor realzado por contraste (azul), que indica las áreas más activas y agresivas del glioma tras la administración de gadolinio. 24
4.2. Estructura del conjunto de datos BraTS2020. 24
4.3. Visualización de un mismo corte axial cerebral en las cuatro modalidades mpMRI utilizadas en el conjunto de datos BraTS. 25
4.4. Distribución de edades redondeadas de los pacientes incluidos en el conjunto de datos BraTS. 27
4.5. Distribución de los días de supervivencia (redondeados a la decena más cercana) de los pacientes del conjunto de datos BraTS. 27
4.6. Radar de completitud de datos clínicos en el conjunto BraTS2020. Se observa que las variables Edad y Días de supervivencia presentan un 64 % de datos presentes, mientras que la variable Extensión de la resección solo alcanza un 35 %. 28
4.7. Diferentes vistas y segmentación del paciente 5. 29
4.8. Estructura del conjunto de datos BCBM-RadioGenomics. 30

5.1. Esquema de la arquitectura U-Net 3D. Fuente: [Çiçek et al., 2016] 32
5.2. Esquema general del modelo LViT. Fuente: [Wang et al., 2022] 35
5.3. Esquema general del modelo LViT. Fuente: [Wang et al., 2022], con modificaciones personalizadas. 37
5.4. Diagrama de la arquitectura de 3D AutoEncoder. Fuente: [Wang et al., 2023]. 41
5.5. Esquema del flujo de información, en donde, a partir del codificador de la red neuronal 3D AE o el de la LViT, se extraen las características latentes de las imágenes de entrada (y del texto en el caso de la red neuronal LViT), con lo que, posteriormente, se entrenan los modelos de regresión clásicos. . 42
5.6. Diagrama de la arquitectura de UNETR (UNEt TRansformers). Fuente: [Hatamizadeh et al., 2022a] 46
5.7. Diagrama de la arquitectura de SAM2. Fuente: [Zhang et al., 2024]. 47

6.1. Regresión modelo Random Forest entrenado con datos imputándoles la media a los valores faltantes. 51

6.2. Modelo SVR con kernel RBF sin aplicarle búsqueda de hiperparámetros, con aumento de datos. En la primera figura (a) se aplica ruido gaussiano a la salida, mientras que en la segunda (b) no se aplica ruido gaussiano a la salida. 52
6.3. Resultados de regresión para el modelo Ridge, aplicando MAE como función de pérdida en la primera figura (a), y aplicando MSE en la segunda (b), usando en ambos casos los datos sin aumento. 55

7.1. Curva de pérdida durante el entrenamiento y la validación del modelo U-Net 3D. 60
7.2. Evolución de las métricas de segmentación para el modelo 3D U-Net en entrenamiento y validación. 61
7.3. Evolución de las métricas de segmentación y pérdida para el modelo LViT en entrenamiento y validación. 62
7.4. Resultados de pérdida y métrica Dice durante el entrenamiento y validación usando SegResNet. 63
7.5. Imágenes T1 y T1ce del paciente 296 perteneciente a test. 63
7.6. Imágenes T2 y flair del paciente 296 perteneciente a test. 64
7.7. Valor real para cada tipo de segmentación del paciente 296 perteneciente a test. 64
7.8. Comparación entre el valor real y las predicciones de 3 modelos de segmentación en imágenes cerebrales, junto a sus mapas de error y métricas Dice por clase. El color verde corresponde a la métrica Dice en WT, el color rojo corresponde a la métrica Dice en TC, y el azul corresponde a la métrica Dice en ET. 65
7.9. Comparación tridimensional entre el valor real y las predicciones de 2 modelos de segmentación en imágenes cerebrales, que trabajan con volúmenes 3D. 66
7.10. Comparación de resultados para las distintas cabezas de regresión. 68
7.11. Comparación de resultados de los mejores modelos de regresión sobre SD, entrenados con las características latentes del 3D AutoEncoder y las estadísticas vóxel de las segmentaciones, sin aplicar aumento de datos. 72
7.12. Comparación de resultados de los mejores modelos de regresión sobre SD, entrenados con las características latentes del 3D AutoEncoder y las estadísticas vóxel de las segmentaciones, aplicando aumento de datos. 73
7.13. Comparación de resultados de los mejores modelos de regresión sobre SD, entrenados con el y4 del LViT sin aplicar aumento de datos. 74
7.14. Comparación de resultados de los mejores modelos de regresión sobre SD, entrenados con el y4 del LViT aplicando aumento de datos. 75

Índice de tablas

3.1. Resumen de fases y planificación inicial del proyecto 18
3.2. Resumen de tareas realizadas durante el proyecto 18

6.1. Asignación de tareas por modelo utilizado 48
6.2. Resumen de funciones de pérdida utilizadas en cada modelo de segmentación. 54
6.3. Resumen de funciones de pérdida utilizadas en los modelos de regresión. . 54
6.4. Resumen de hiperparámetros utilizados en modelos principales. 56

7.1. Coeficientes Dice e IoU (%) por clase y modelo en el conjunto de test. En negrita se destacan los mejores resultados por métrica. 61
7.2. Configuraciones de SVR Linear 69
7.3. Configuraciones de Ridge 69
7.4. Configuraciones de RandomForest 69
7.5. Configuraciones de CatBoost 70
7.6. Configuraciones de SVR RBF 70
7.7. Configuraciones de HistGradientBoosting 70
7.8. Configuraciones de Ridge Polynomial 70
7.9. Configuraciones de MLPRegressor 70
7.10. Errores MAE para la predicción de días de supervivencia (SD) en el conjunto de test, para los modelos de regresión entrenados con las características latentes sacadas del codificador del 3D AutoEncoder junto con estadísticas vóxel de las segmentaciones. 76
7.11. Errores MAE para la predicción de días de supervivencia (SD) en el conjunto de test, para los modelos de regresión entrenados con el y4 del LViT. 76
7.12. Errores MAE para la predicción de días de supervivencia (SD) en el conjunto de test, para las cabezas de regresión integradas en el modelo LViT. . 77

B.1. Tabla con la valoración del grado de relación del TFG con los ODS, según una escala de 0 a 3 (no procede, bajo, medio, alto). 89

Agradecimientos

Al llegar al final de este trabajo, quiero agradecer sinceramente a todas las personas que han estado a mi lado durante estos años y que han hecho posible que hoy esté cerrando esta etapa.

En primer lugar, gracias a mi tutor del TFG, Javier Sánchez Pérez, por su implicación, su tiempo y sus consejos. Su ayuda ha sido clave para que pudiera dar forma a este proyecto y llevarlo a buen puerto.

Gracias también a mis padres, Ramón y Cristina, por su apoyo constante, por estar siempre ahí, por preocuparse por mí y animarme incluso cuando las cosas se complicaban. Sin ustedes, nada de esto habría sido posible.

A mi hermano Yuri, por acompañarme siempre, por escucharme cuando necesitaba desahogarme, y por ser ese apoyo que muchas veces no se ve, pero siempre se siente.

A mi pareja, Juan Carlos, por su paciencia, por entenderme en los momentos de estrés, y por estar a mi lado en los días buenos y en los no tan buenos. Gracias por no soltarme de la mano mientras sacaba esto adelante.

A toda mi familia y amigos, por su cariño, su apoyo y sus palabras de ánimo en cada paso de este camino. En especial, a mi tío Berto, por su cercanía, por ayudarme en todo lo que pudo y por interesarse siempre por cómo iba todo.

Y, por supuesto, a mis compañeros de clase. Gracias por compartir este recorrido académico. En especial, a quienes han estado presentes desde el inicio hasta el final de la carrera, por el trabajo conjunto, el apoyo mutuo y la constancia durante todo el proceso. Ha sido un respaldo importante en esta etapa.

Gracias a todos por formar parte de este proceso.

Resumen

Detectar y clasificar tumores cerebrales es un reto fundamental dentro del ámbito médico. Un diagnóstico rápido y preciso es esencial para garantizar un tratamiento eficaz en pacientes afectados.

Este proyecto se enfoca en investigar y encontrar un sistema inteligente, basado en redes neuronales multimodales, que aborde el problema desde una perspectiva más amplia. El objetivo es ser capaces de analizar las imágenes médicas, extrayendo información relevante con mayor profundidad y combinando distintos tipos de información para mejorar el rendimiento y los resultados.

Para mejorar la segmentación, junto con la imagen, se integran datos clínicos del paciente, que también se usan para estimar el tiempo de vida, y por otro, se trabajan con diferentes tipos de imágenes médicas como entrada.

Abstract

Detecting and classifying brain tumors is a fundamental challenge in the medical field. A fast and accurate diagnosis is essential to ensure effective treatment for affected patients.

This project focuses on researching and finding an intelligent system based on multimodal neural networks that approaches the problem from a broader perspective. The goal is to analyze medical images by extracting relevant information in greater depth and combining different types of data to improve performance and outcomes.

To enhance segmentation, clinical data from the patient is integrated along with the medical images. These clinical features are also used to estimate survival time. Additionally, various types of medical images are used as input to enrich the analysis.

Capítulo 1

Introducción

Dentro del ámbito de la imagen médica, uno de los principales retos es el diagnóstico y seguimiento de tumores cerebrales. Estas patologías, como el glioma, presentan una elevada mortalidad y una progresión heterogénea, lo que hace esencial una caracterización precisa para tomar decisiones clínicas informadas. Sin embargo, este proceso requiere analizar imágenes complejas y de alta resolución, lo que a menudo demanda tiempo y experiencia. En particular, la segmentación de las regiones tumorales a partir de una Imagen de Resonancia Magnética (MRI, en sus siglas en inglés) (IRM) constituye una tarea crítica, ya que permite delimitar volúmenes, planificar intervenciones y monitorizar la evolución del paciente. Se trata, no obstante, de un procedimiento manual laborioso y subjetivo, con alta variabilidad entre observadores, incluso entre expertos [Menze et al., 2015, Bakas et al., 2017, Isensee et al., 2019].

En entornos hospitalarios con recursos limitados, la disponibilidad de tiempo por imagen es reducida, y la necesidad de realizar una evaluación pronóstica del paciente añade una capa adicional de complejidad. Esta predicción suele basarse en factores como la edad, el subtipo de glioma, el grado histológico o la respuesta al tratamiento, cuya integración cuantitativa y automatizada sigue siendo un desafío abierto.

Muchos enfoques actuales en segmentación automática se centran únicamente en la imagen, sin incorporar el contexto clínico del paciente, lo que limita su utilidad en situaciones reales [Zhou et al., 2021]. Además, la mayoría operan sobre una única modalidad de imagen, sin aprovechar la riqueza informativa que aportan diferentes secuencias de resonancia magnética (Secuencia de Resonancia Magnética T1 (T1), Secuencia de Resonancia Magnética T2 (T2), Fluid Attenuated Inversion Recovery (FLAIR), T1 con Contraste Gadolinio (T1c/T1Gd)), lo que puede afectar negativamente al rendimiento en tareas clínicas complejas [Valverde et al., 2017, Isensee et al., 2018b]. A esto se suma que los modelos de predicción, especialmente los que estiman la supervivencia, tienden a desarrollarse de forma independiente, sin conexión con las representaciones latentes generadas durante la segmentación, lo que puede provocar incoherencias o pérdida de información relevante [Mobadersany et al., 2018]. Frente a esta fragmentación, estudios recientes han demostrado que el aprendizaje conjunto y multimodal puede mejorar tanto la precisión de segmentación como la calidad de las predicciones clínicas [Chen et al., 2021b, Zhou et al., 2019, Esteva et al., 2021]. Esto se refuerza aún más cuando se integran características derivadas de las segmentaciones, como estadísticas vóxel o representaciones latentes más estructuradas [Lu et al., 2022, Kamnitsas et al., 2017, Chartsias et al., 2019].

Para responder a esta problemática, el trabajo plantea un enfoque integral basado en el uso de redes neuronales multimodales, que permitan combinar diferentes tipos de entradas

dentro de una misma arquitectura. Este enfoque busca mejorar la segmentación automática de los tumores mediante la incorporación de contexto clínico o diferentes tipos de imágenes de entrada, y al mismo tiempo, facilitar la generación de predicciones cuantitativas asociadas a cada caso, como la estimación de la supervivencia. Así, se exploran tanto estrategias de fusión multimodal para la segmentación, como el uso de representaciones aprendidas y características estadísticas derivadas de los datos como base para tareas de regresión. De esta manera, el sistema aspira a abordar de forma unificada dos objetivos complementarios y clínicamente relevantes: la interpretación precisa de las imágenes y la obtención de predicciones individualizadas.

El sistema se construirá utilizando el conjunto de datos Brain Tumor Segmentation Challenge (BraTS), que ofrece imágenes de resonancia magnética en múltiples modalidades junto con segmentaciones manuales y variables clínicas asociadas [Bakas et al., 2018]. Sobre esta base, se desarrollarán dos variantes principales de redes multimodales: una orientada a la fusión de diferentes modalidades de imagen, y otra centrada en la integración de información visual con datos clínicos. Este enfoque conjunto permite superar las limitaciones de los métodos tradicionales, que suelen tratar estas tareas por separado, y se fundamenta en estudios recientes que destacan los beneficios de combinar información heterogénea en aplicaciones médicas complejas [Valle et al., 2021, Chen et al., 2021b, Zhou et al., 2021].

Para abordar la tarea de segmentación multimodal mediante la fusión de diferentes modalidades de imagen, se implementan redes de convolución profundas de probada eficacia en imágenes médicas tridimensionales, como U-Net Tridimensional (3D) [Çiçek et al., 2016], una extensión de la Red neuronal tipo U para segmentación (U-Net) original adaptada a volúmenes, y Segmentation Residual Network (SegResNet) [Myronenko, 2018], que introduce conexiones residuales para mejorar la estabilidad del entrenamiento y la precisión de segmentación. Estas arquitecturas permiten capturar tanto detalles locales como estructuras globales del tumor.

Por otro lado, para integrar información clínica con imágenes y generar predicciones tanto de segmentación como cuantitativas (como la estimación de supervivencia), se exploraron modelos híbridos y enfoques multimodales. En este contexto, se utilizó la arquitectura Language meets Vision Transformer (LViT), que permite fusionar de manera efectiva entradas visuales y tabulares mediante mecanismos de atención inspirados en los transformers [Dosovitskiy et al., 2020]. Inicialmente, LViT se empleó para realizar segmentaciones [Chen et al., 2021b], pero, de forma original, en este trabajo se modificó la arquitectura añadiendo una cabeza de regresión, desarrollada específicamente para que la red pudiera aprender a partir de la combinación de ambas fuentes de información. Sin embargo, no se obtuvieron buenos resultados, ya que las cabezas de regresión no llegaban a ajustarse del todo a los datos.

Además, esa misma información multimodal (imagen + texto clínico) se utilizó como entrada para modelos clásicos [Huang et al., 2019], y, en paralelo, se emplearon también modelos tradicionales como el autoencoder, entrenando específicamente un 3D autoencoder para la reconstrucción de los volúmenes del conjunto de datos con el objetivo de extraer las características latentes más representativas desde su codificador [Chartsias et al., 2019]. Estas representaciones latentes, combinadas con estadísticas vóxel derivadas de las segmentaciones originales, se usaron para entrenar modelos de regresión que ofrecieron muy buenos resultados en la predicción.

Para optimizar el rendimiento, se realizó una búsqueda aleatoria junto con validación cruzada para seleccionar los hiperparámetros óptimos de cada modelo. Además, se aplicó

una estrategia de aumento de datos, añadiendo ruido gaussiano a los valores más altos de la variable objetivo, ya que los modelos inicialmente no se ajustaban bien a esos valores extremos. Esta técnica de aumento de datos mejoró notablemente la capacidad de ajuste de los modelos en los rangos altos.

Sin embargo, los modelos entrenados con los datos extraídos del codificador del LViT no lograron un rendimiento comparable en ningún caso, ya fuera sin aumento de datos o aplicando la estrategia de aumento, mostrando limitaciones en la calidad de las predicciones obtenidas.

Para validar la propuesta, cada variante se entrenó y evaluó usando métricas específicas tanto para segmentación (coeficiente de Dice, Índice de Superposición de Jaccard (IoU, en sus siglas en inglés)) como para regresión (Mean Absolute Error (MAE), Mean Squared Error (MSE)), además de aplicar técnicas de análisis de errores.

El desarrollo del sistema se ha llevado a cabo siguiendo el enfoque Cross-Industry Standard Process for Data Mining (CRISP-DM), un marco metodológico ampliamente adoptado en proyectos de ciencia de datos. A lo largo del proceso, se abordaron de forma estructurada las distintas etapas: desde la comprensión del problema y la preparación de los datos, hasta el diseño y entrenamiento de los modelos, su evaluación con métricas específicas y el análisis final de los resultados. En la práctica, esto implicó combinar entornos de prototipado como Google Colaboratory (COLAB) para las primeras pruebas, con estaciones de trabajo equipadas con Unidad de Procesamiento Gráfico (GPU, en sus siglas en inglés) para los entrenamientos más exigentes. Se prestó especial atención a la reproducibilidad y al rigor metodológico, empleando búsqueda aleatoria para la optimización de hiperparámetros en los modelos de regresión clásicos y manteniendo un análisis crítico en cada etapa, en línea con las buenas prácticas de la ciencia de datos aplicada a la medicina.

1.1. Motivaciones

Es innegable la gran influencia que la Inteligencia Artificial (IA) tiene ahora en la medicina, especialmente en el estudio de imágenes médicas. Se está observando un futuro alentador, donde identificar y segmentar tumores en resonancias magnéticas del cerebro es clave para mejorar la detección temprana y, por lo tanto, afinar las decisiones de los médicos. A pesar de esto, es crucial admitir que los datos que se obtienen solo de las imágenes, aunque son muy útiles, no bastan para entender por completo la salud del paciente. Más bien, son una parte importante de un diagnóstico mayor, que necesita combinar varias fuentes de información para entender mejor la situación.

Por eso, este estudio se enfoca en usar redes neuronales multimodales, que unan imágenes con datos clínicos, entre otras fuentes complementarias. Al integrar estos distintos tipos de datos, se busca mejorar la capacidad de los modelos para ofrecer herramientas más eficientes y adaptadas a las necesidades del entorno sanitario. Esta estrategia no solo persigue una mayor precisión en la segmentación de tumores y en la predicción de variables clínicas relevantes, sino también una utilidad práctica que facilite el trabajo diario de los profesionales.

1.2. Objetivos

Este estudio se centra principalmente en la concepción y la puesta en marcha de un sistema de aprendizaje profundo que pueda fusionar imágenes obtenidas por medios médicos y datos sobre el estado del paciente. La idea es que este sistema sirva de apoyo al diagnóstico y el estudio de tumores que se forman en el cerebro. Al fusionar distintos tipos de datos, se espera que la exactitud de las labores de segmentación y de pronóstico mejore notablemente, lo cual impulsaría el empleo de la IA en el sector de la medicina.

En el transcurso de la ejecución del proyecto, se han establecido varios propósitos, que se pueden dividir en dos grupos principales: primero, los propósitos técnicos, que son propios del proyecto en sí, y segundo, los propósitos de tipo académico, que están relacionados con la formación que implica el trabajo.

1.2.1. Objetivos del proyecto

El enfoque primordial desde el punto de vista técnico reside en crear un sistema fundamentado en redes neuronales multimodales. Este sistema deberá poder ejecutar segmentaciones de imágenes de resonancia magnética del cerebro, y a la vez, producir estimaciones sobre variables clínicas ligadas, como por ejemplo, los días que el paciente sobrevivirá.

Con este propósito en mente, se han definido las siguientes intenciones concretas:

- Poner en práctica y hacer pruebas con distintas estructuras que combinen datos visuales (imágenes médicas) e información organizada (edad, tiempo de vida en días) en ciertas situaciones, también estructuras que incluyan varios tipos de datos visuales.
- Comparar modelos unimodales y multimodales, evaluando su rendimiento en tareas de segmentación y regresión mediante métricas como el IoU o el MAE.
- Registrar de forma detallada todo el proceso, abarcando la obtención y el alistamiento de los datos, el diseño de los modelos, los ensayos hechos y una evaluación crítica de los resultados logrados.
- Desarrollar una infraestructura flexible que permita realizar entrenamientos tanto en entornos locales como en la nube, optimizando los recursos computacionales disponibles.

1.2.2. Objetivos académicos

Además de los objetivos técnicos, este proyecto tiene como propósito consolidar y aplicar los conocimientos adquiridos durante el Grado en Ciencia e Ingeniería de Datos, desarrollando habilidades fundamentales tanto a nivel técnico como metodológico. Los objetivos académicos se resumen en los siguientes puntos:

- Usar técnicas de aprendizaje automático, tratamiento de imágenes y análisis de datos en un caso práctico real en el ámbito de la salud.
- Fortalecer competencias prácticas en el uso de herramientas de desarrollo modernas como `PyTorch`, `TensorBoard`, `TensorFlow` y entornos de trabajo reproducibles.

- Aprender a planificar, documentar y llevar adelante proyectos técnicos de principio a fin.

- Impulsar el pensamiento crítico, la independencia a la hora de decidir y la capacidad de investigar aplicado a un trabajo que pueda tener un impacto en la sociedad.

1.3. Organización del documento

Este informe se ha organizado en ocho partes, donde cada una cubre un aspecto específico del proyecto, de manera gradual y exhaustiva.

El **capítulo 2** se enfoca en el estado actual del tema, considerando tres aspectos clave: los modelos básicos y avanzados para analizar imágenes médicas, los diseños de redes neuronales multimodales, y los métodos recientes de predicción médica usando aprendizaje automático. Se sitúan estos avances en el contexto de la IA aplicada a la medicina y se señalan los problemas actuales que este estudio busca resolver.

El **capítulo 3** está dedicado a la metodología y planificación del proyecto. Se explica el método general seguido, basado en los principios de la ciencia de datos y la evaluación experimental, así como la planificación inicial por etapas, las tareas hechas durante el desarrollo y los recursos usados. También se dan detalles del entorno de trabajo, incluido el hardware local para entrenar modelos grandes y el software utilizado.

En el **capítulo 4** se presenta el conjunto de datos empleado. Se describen en detalle las fuentes de información utilizadas, principalmente el conjunto de datos BraTS2020, que incluye imágenes multimodales de resonancia magnética junto a anotaciones manuales y datos clínicos estructurados. También se explica el formato de los datos, los pasos de preparación y la división de los conjuntos para entrenamiento, prueba y validación.

El **capítulo 5** describe la implementación de los modelos utilizados. Se detallan las arquitecturas seleccionadas, incluyendo las redes neuronales convolucionales para segmentación, las redes multimodales para integración de información y los modelos de regresión para la predicción. Se explica también la estructura de los modelos desarrollados, cómo se combinan los diferetes tipos de datos y otros aspectos.

En el **capítulo 6** se exponen las métricas empleadas, tanto para evaluación como para la pérdida en cada tarea (de segmentación y de regresión), el preprocesamiento de los datos según el modelo, la estrategia de división de datos que se siguió en cada modelo, las configuraciones de los hiperparámetros y las técnicas de entrenamiento, entre otros.

En el **capítulo 7** se presentan y examinan los resultados obtenidos. Se muestran las métricas usadas, se comparan los distintos enfoques implementados y se analiza la calidad de las segmentaciones y predicciones hechas por los modelos. Esta parte ofrece una visión crítica de los puntos fuertes y débiles de cada estrategia.

Finalmente, en el **capítulo 8** se presentan las conclusiones de este estudio, haciendo un recuento de los objetivos cumplidos y analizando los puntos susceptibles de mejora. Asimismo, se detallan algunas vías de investigación futuras que podrían surgir a raíz de este proyecto, como la aplicación de modelos más sofisticados de integración multimodal o la integración de datos genómicos.

Además de los capítulos principales, el documento incorpora al final un apartado dedicado a los Objetivos de Desarrollo Sostenible (ODS) que sigue el proyecto, destacando su alineación con iniciativas globales de impacto social y sanitario. Seguidamente, se incluye una lista de acrónimos para facilitar la lectura de términos técnicos abreviados, junto con un glosario que ofrece definiciones claras de los conceptos clave empleados en

el ámbito de la inteligencia artificial médica. Por último, se presenta la bibliografía, que recoge todas las fuentes consultadas y citadas a lo largo del trabajo.

Capítulo 2

Estado del arte

La IA ha transformado numerosos sectores en la última década, desde la industria y las finanzas hasta la educación o el entretenimiento. En el ámbito de la salud, esta revolución tecnológica ha cobrado un protagonismo particular gracias al avance de subcampos como el aprendizaje profundo y la visión por computador. Esta última se refiere a la capacidad de los algoritmos para interpretar y analizar automáticamente imágenes, simulando de alguna forma la percepción visual humana. Gracias a la combinación de grandes volúmenes de datos médicos y técnicas modernas de entrenamiento, la visión por computador se ha convertido en una herramienta fundamental para apoyar el diagnóstico, seguimiento y tratamiento de enfermedades.

Uno de los campos donde este avance ha sido más notorio es el análisis de imágenes médicas. Modalidades como la radiografía, la Tomografía Computarizada (TC) o IRM generan una cantidad masiva de datos visuales cada día. Estas imágenes contienen información crítica que puede guiar las decisiones clínicas, pero su análisis manual por parte de los profesionales sanitarios resulta cada vez más exigente, no solo por el volumen, sino también por la necesidad de precisión y rapidez. En este contexto, los algoritmos de visión por computador permiten automatizar tareas complejas, mejorando tanto la eficiencia como la fiabilidad del diagnóstico.

Entre los múltiples retos clínicos abordados con visión por computador, la detección y análisis de tumores cerebrales destaca por su complejidad y relevancia médica. Los tumores en el cerebro, en concreto, pueden adoptar formas y estructuras muy diversas, afectando regiones anatómicas sensibles y requiriendo una segmentación precisa para planificar tratamientos como la cirugía o la radioterapia. La resonancia magnética, por su capacidad para visualizar tejidos blandos en detalle, es la técnica más utilizada en este campo. Sin embargo, la interpretación de estas imágenes sigue siendo una tarea laboriosa que puede beneficiarse enormemente del soporte automatizado.

El presente capítulo analiza los enfoques tradicionales y modernos utilizados en tres tareas clave dentro del análisis visual médico: clasificación, detección y segmentación. A lo largo de la sección se pone especial énfasis en el caso de los tumores cerebrales, tanto por su relevancia clínica como por su papel central en el desarrollo de este trabajo. Se revisan las metodologías más utilizadas, los modelos de referencia y los avances recientes que permiten combinar imágenes multimodales con datos clínicos, abriendo la puerta a sistemas de diagnóstico más integrales y personalizados.

2.1. Técnicas tradicionales en análisis de imágenes médicas

Antes de la popularización de los enfoques basados en aprendizaje automático, el análisis de imágenes médicas se sustentaba en técnicas deterministas, estadísticas y heurísticas, muchas de ellas provenientes del campo clásico del procesamiento digital de imágenes. Estas metodologías buscaban extraer información relevante a partir de la intensidad de los píxeles, los bordes, la textura o la morfología de las estructuras anatómicas [Osher and Sethian, 1988, Gonzalez and Woods, 2006, Russ and Neal, 2016].

Uno de los métodos más empleados era la segmentación por umbralización, que permitía separar regiones de interés asignando un umbral fijo o adaptativo sobre el histograma de intensidades. Si bien es sencilla de implementar, esta técnica es sensible al ruido y suele fallar cuando las regiones objetivo presentan intensidades similares a las de los tejidos circundantes, algo frecuente en imágenes de resonancia magnética [Nicolas et al., 2000, Seymour et al., 2000].

Los métodos basados en contornos activos, también conocidos como *Snakes*, introdujeron un enfoque más flexible al permitir que una curva evolucione sobre la imagen bajo la influencia de fuerzas internas (suavidad) y externas (gradientes), buscando delinear estructuras anatómicas mediante la minimización de una función de energía [Kass et al., 1988]. Pese a su elegancia matemática, estos métodos dependen fuertemente de la inicialización y pueden quedar atrapados en mínimos locales, especialmente en imágenes con bordes débiles o ruido [Malladi et al., 1995, Zhu and Yuille, 1996].

Las operaciones morfológicas, como el erosionado, la dilatación, la apertura o el cierre, también fueron utilizadas para refinar regiones segmentadas, eliminar artefactos o rellenar huecos [Serra, 1983]. Estas técnicas se basan en operadores binarios aplicados sobre máscaras y resultan útiles como pasos intermedios, aunque rara vez ofrecen por sí solas una segmentación final precisa [Vincent, 1993].

En paralelo, los algoritmos de *clustering* no supervisado, como *k-means*, *fuzzy c-means* o *expectation-maximization*, permitieron agrupar píxeles en función de sus similitudes estadísticas. Estos enfoques han sido aplicados, por ejemplo, para segmentar tejidos cerebrales en IRM [Zhang et al., 2001, Pham et al., 2000]. Sin embargo, carecen de contexto espacial y pueden generar resultados inconsistentes en presencia de ruido o intensidades solapadas.

Por otro lado, la transformada de *watershed* se consolidó como una técnica robusta para la segmentación de estructuras adyacentes, modelando la imagen como una superficie topográfica donde las crestas y valles definen fronteras [Beucher, 1992]. Si bien eficaz, esta técnica tiende a la sobresegmentación si no se aplican previamente filtros o marcadores [Roerdink and Meijster, 2000].

En general, estas técnicas tradicionales fueron fundamentales en los primeros avances del análisis computacional de imágenes médicas. No obstante, sus limitaciones se hicieron evidentes ante la complejidad y heterogeneidad de patologías como los tumores cerebrales. Su dependencia de parámetros ajustados manualmente, la falta de capacidad para capturar relaciones espaciales complejas y la necesidad de una intervención humana constante motivaron la transición hacia métodos más flexibles y robustos, como los que ofrece el aprendizaje profundo [Littmann et al., 2021, Shen et al., 2017, Litjens et al., 2017].

2.2. Enfoques basados en aprendizaje profundo

Con el desarrollo del aprendizaje profundo, han surgido modelos que mejoran notablemente la precisión y automatización del análisis de imágenes médicas, superando las limitaciones de los métodos tradicionales.

El uso de redes neuronales profundas ha permitido incorporar un aprendizaje jerárquico de características, donde el modelo puede identificar desde patrones básicos de borde o textura hasta representaciones semánticas complejas asociadas a tejidos o patologías. Gracias a su capacidad de generalización, estos métodos han sido capaces de adaptarse a diferentes modalidades de imagen (como resonancia magnética, TC o ecografía) y a una amplia variedad de patologías, incluso cuando se dispone de volúmenes de datos moderadamente reducidos. Además, el desarrollo de técnicas como el aprendizaje multitarea, la transferencia de conocimiento y la integración multimodal ha favorecido la aparición de arquitecturas más robustas y clínicamente aplicables. En los apartados siguientes se detallan los principales enfoques utilizados en tareas específicas dentro del procesamiento médico, destacando sus avances y limitaciones actuales.

2.2.1. Clasificación y detección con redes convolucionales

La Red Neuronal Convolucional (CNN) ha transformado radicalmente el análisis de imágenes médicas, consolidándose como una herramienta esencial en tareas como la clasificación de patologías, la detección de lesiones y la segmentación precisa de estructuras anatómicas. Su capacidad para aprender representaciones jerárquicas de alto nivel directamente a partir de los datos ha permitido superar muchas de las limitaciones de los enfoques tradicionales basados en reglas manuales o características diseñadas por expertos.

Modelos generales como VGG, ResNet y DenseNet han sido adaptados con éxito a diversos entornos clínicos. Se han utilizado, por ejemplo, en la clasificación de tumores en mamografías con niveles de precisión comparables a los de radiólogos humanos [McKinney et al., 2020], en la detección de nódulos pulmonares en tomografías computerizadas de baja dosis [Ardila et al., 2019, Setio et al., 2016], y en la clasificación automática del tipo de glioma a partir de imágenes de resonancia magnética cerebral, donde se requiere una alta sensibilidad ante variaciones morfológicas sutiles [Havaei et al., 2017, Pereira et al., 2016].

En el contexto específico de los tumores cerebrales, se han desarrollado arquitecturas convolucionales especialmente diseñadas para segmentar regiones como el edema peritumoral, el núcleo tumoral y el tumor realzado por contraste. Una de las más influyentes es Deep Medical Image Computing Network (DeepMedic) [Kamnitsas et al., 2017], que emplea rutas paralelas a distintas escalas y un enfoque de post-procesamiento basado en Conditional Random Fields (CRF) para refinar los bordes segmentados. También destaca High-Resolution 3D Neural Network (HighRes3DNet) [Li et al., 2017], que introduce convoluciones dilatadas para mantener la resolución espacial sin necesidad de operaciones de pooling, lo que resulta útil en tareas donde la localización precisa es crítica. Otra propuesta relevante es Voxelwise Residual Network (VoxResNet) [Chen et al., 2018], que extiende los bloques residuales al dominio volumétrico, mejorando la capacidad del modelo para aprender representaciones profundas en imágenes tridimensionales.

Junto a estas, arquitecturas como Anisotropic Hybrid Network (AH-Net) [Xu et al., 2019] y Attention-Gated CNNs [Schlemper et al., 2019] han propuesto mecanismos avanzados para capturar información contextual en regiones relevantes de la ima-

gen, mejorando el rendimiento en tareas de segmentación en imágenes con estructuras complejas o poco diferenciadas, como el encéfalo o los tejidos blandos. Además, modelos como 3D CNN cascaded architectures [Zhao et al., 2018] han demostrado que dividir el problema en etapas progresivas (detección gruesa seguida de segmentación fina) puede ser una estrategia eficaz en entornos de diagnóstico asistido por computador.

En tareas de detección, se han adaptado redes originalmente diseñadas para visión por computador general, como Faster R-CNN, SSD o YOLO, a aplicaciones médicas. Estas redes han sido utilizadas para identificar regiones sospechosas en imágenes 2D, como radiografías de tórax o mamografías, con resultados prometedores en cuanto a sensibilidad y precisión [Rajpurkar et al., 2018, Mehta et al., 2021]. Aunque su extensión directa a imágenes 3D sigue siendo un reto computacional, han surgido variantes tridimensionales como You Only Look Once en 3D (3D-YOLO) [Oh et al., 2020], que han comenzado a explorar la detección en volúmenes de TC o Resonancia Magnética (RM) con cierto éxito en entornos experimentales.

Numerosos estudios han evaluado la eficacia de estas redes en condiciones clínicas reales. Algunos trabajos recientes [Reyes and Sánchez, 2024, Medina and Sánchez, 2023] realizan comparaciones sistemáticas de arquitecturas convolucionales aplicadas a la clasificación de tumores cerebrales, destacando el papel de las estrategias de normalización, el diseño de capas intermedias y el preprocesamiento multimodal en la mejora de la precisión diagnóstica, así como el compromiso entre eficiencia computacional y rendimiento predictivo. Además, se proponen criterios prácticos para seleccionar arquitecturas según la disponibilidad de recursos y el tipo de patología a analizar.

Además, se han explorado diversas técnicas para mejorar la generalización y robustez de las CNNs en el ámbito médico. Entre ellas se incluyen el uso de transferencia de aprendizaje a partir de modelos preentrenados en conjuntos grandes, el aumento de datos mediante transformaciones espaciales y de intensidad, y la fusión multimodal de diferentes secuencias de imagen para capturar diferentes propiedades tisulares del tumor [Tajbakhsh et al., 2016, Litjens et al., 2017, Shen et al., 2017]. Esta última estrategia ha resultado particularmente útil en el estudio de gliomas, donde la integración de múltiples vistas del mismo tejido permite una caracterización más completa del proceso patológico.

2.2.2. Segmentación médica con U-Net y variantes

El modelo U-Net [Ronneberger et al., 2015] se ha consolidado como la arquitectura de referencia en tareas de segmentación médica, debido a su diseño eficaz y su capacidad para funcionar bien incluso con conjuntos de datos reducidos. Su estructura en forma de codificador-decodificador, complementada con conexiones de salto, permite preservar la información espacial durante todo el proceso de codificación y decodificación, facilitando la recuperación precisa de los contornos anatómicos.

A partir de esta base, han surgido múltiples variantes que adaptan U-Net a diferentes contextos clínicos. Por ejemplo, U-Net tridimensional (U-Net 3D) [Çiçek et al., 2016] extiende la arquitectura original para trabajar directamente sobre volúmenes tridimensionales, lo que resulta especialmente útil en el análisis de resonancias magnéticas, permitiendo capturar la continuidad espacial entre cortes axiales. U-Net con mecanismos de atención (Attention U-Net) [Oktay et al., 2018], por su parte, incorpora mecanismos de atención que aprenden a enfocar automáticamente las regiones más relevantes durante la decodificación, mejorando el rendimiento en escenarios donde los tumores pueden ser pequeños, irregulares o presentar bordes poco definidos. SegResNet [Myronenko, 2018], basado en

bloques residuales 3D, ha demostrado un rendimiento sobresaliente en segmentación de tumores cerebrales gracias a su estabilidad en el entrenamiento profundo y su capacidad de generalizar bien en datos clínicos heterogéneos.

Además de estas variantes, modelos como Volumetric Convolutional Neural Network (V-Net) [Milletari et al., 2016a] utilizan convoluciones volumétricas y funciones de pérdida específicas como *Dice loss* para mejorar la segmentación en estructuras desequilibradas, como ocurre con las regiones de Tumor Realzado (ET, en sus siglas en inglés) frente al Tumor Completo (WT, en sus siglas en inglés). También han surgido enfoques híbridos [Isensee et al., 2021] con el modelo no-new-U-Net (U-Net autoajustable) (nnU-Net), una versión autoajustable de U-Net que optimiza automáticamente hiperparámetros y arquitectura para cada tarea específica, obteniendo resultados punteros sin necesidad de intervención manual.

Estas variantes han sido ampliamente empleadas en competiciones como el BraTS Challenge [Bakas et al., 2018], centrado en la segmentación automática de tumores cerebrales en imágenes multimodales. En dicho entorno, los modelos deben diferenciar entre subregiones tumorales clave, como el tumor completo, el TC y la región de realce tumoral. Varios estudios [Isensee et al., 2018a, Myronenko, 2018, Wang et al., 2021a] han demostrado que pequeñas modificaciones en la arquitectura U-Net pueden tener un impacto significativo en el rendimiento, sobre todo cuando se combinan con técnicas de preprocesamiento, normalización y aumento de datos adaptadas al dominio biomédico.

2.3. Transformers visuales en imagen médica

Los Vision Transformer (ViT) han emergido como una alternativa poderosa a las CNNs en tareas de visión por computador, especialmente en el contexto de la imagen médica. A diferencia de las CNNs, que dependen de operaciones locales como las convoluciones, los ViT pueden modelar relaciones espaciales de largo alcance mediante mecanismos de atención, lo cual resulta crucial para detectar patrones globales en imágenes de alta resolución y estructuras anatómicas complejas.

Modelos como Transformer-based U-Net (TransUNet) [Chen et al., 2021a] han demostrado que combinar un codificador basado en ViT con un decodificador tipo U-Net permite capturar tanto el contexto global como los detalles locales, beneficiándose de las fortalezas de ambas arquitecturas. Este diseño ha sido validado en tareas como segmentación de órganos abdominales o de tumores cerebrales en imágenes MRI.

UNEt TRansformers (UNETR) [Hatamizadeh et al., 2022a], por su parte, es una de las primeras arquitecturas en aplicar directamente un Vision Transformer sobre volúmenes 3D, dividiendo el volumen en parches y codificándolos sin necesidad de convoluciones iniciales. Gracias a su estructura jerárquica, UNETR facilita la segmentación de estructuras internas con precisión y se ha posicionado como una referencia en segmentación médica tridimensional.

Otra propuesta destacada es Swin Transformer U-Net (SwinUNet), que introduce el concepto de atención con ventanas deslizantes jerárquicas [Cao et al., 2021]. Este enfoque reduce el coste computacional de la autoatención global y permite capturar relaciones espaciales a múltiples escalas, lo cual es especialmente útil en imágenes médicas con diferentes niveles de detalle.

Estudios comparativos [Hatamizadeh et al., 2022b, Martín and Sánchez, 2025] han evidenciado que las arquitecturas basadas en Transformers no solo igualan, sino que superan en muchos casos a las CNNs tradicionales en métricas como el coeficiente de *Dice*,

la precisión y la robustez frente a ruido o variaciones anatómicas. Estos modelos han demostrado su eficacia en múltiples modalidades (IRM, CT) y órganos (cerebro, pulmón, hígado), consolidándose como una opción de vanguardia en segmentación médica automatizada.

2.4. Modelos multimodales

La combinación de múltiples fuentes de información es una estrategia cada vez más presente en el desarrollo de modelos de IA para imagen médica. En contextos clínicos reales, las decisiones diagnósticas rara vez se basan únicamente en una imagen; suelen requerir también variables clínicas como edad, subtipo de tumor, estado funcional del paciente, o incluso descripciones textuales recogidas en informes radiológicos.

Por ello, los modelos multimodales surgen como una necesidad para generar predicciones más completas, personalizadas y coherentes con el contexto clínico. Esta integración puede centrarse tanto en la combinación de imágenes con datos clínicos estructurados o texto libre, como en la fusión de distintas modalidades de imagen médica obtenidas mediante diferentes secuencias o dispositivos.

2.4.1. Integración de imagen médica y datos clínicos

En este enfoque se combinan las representaciones extraídas de imágenes médicas con información clínica complementaria. En el caso de las variables estructuradas (como edad, sexo, tipo de glioma o días de supervivencia), existen diferentes formas de integrar estas fuentes: concatenando sus representaciones en una etapa temprana, introduciendo mecanismos de atención cruzada a nivel intermedio, o combinando predicciones al final del proceso. Modelos como LViT han implementado con éxito una integración intermedia, en la que se emplean codificadores compartidos para alinear los embeddings visuales con los vectores clínicos, logrando mejoras significativas en tareas como predicción de supervivencia, clasificación de tumores o análisis de progresión [Chen et al., 2021b, Valle et al., 2021, Baur et al., 2021]. Otros trabajos [He et al., 2021, Esteva et al., 2021] han explorado enfoques multimodales para mejorar la estratificación de riesgo y la toma de decisiones clínicas en distintos contextos oncológicos.

Por otro lado, cuando la información clínica se presenta en forma de texto libre (como informes radiológicos o descripciones de hallazgos), se recurre a arquitecturas basadas en aprendizaje contrastivo. Modelos como CLIP-Medical [Zhang et al., 2023b], MedKLIP [Zhang et al., 2023a], GLoRIA [Huang et al., 2021] o BioViL [Boecking et al., 2022] entrenan redes duales para proyectar imágenes y textos en un espacio latente común. Esta representación compartida permite realizar tareas como la recuperación de imágenes a partir de texto, la localización de patologías descritas o la generación automática de informes clínicos. Estas técnicas han sido especialmente útiles en campos como la radiología torácica y la histopatología digital, donde los textos pueden proporcionar detalles diagnósticos difíciles de captar solo a partir de la imagen. Además, han demostrado un buen rendimiento incluso cuando se dispone de bases de datos con anotaciones semiestructuradas, como MIMIC-CXR, CheXpert o PadChest [Irvin et al., 2019, Bustos et al., 2020].

En ambos casos, la integración clínica busca complementar el contenido puramente visual, aportando contexto semántico que puede ser crítico para mejorar la precisión diagnóstica y la utilidad clínica de los sistemas automáticos.

2.4.2. Fusión de distintas modalidades de imagen médica

La mayoría de los estudios clínicos avanzados, especialmente en tumores cerebrales, se apoyan en imágenes obtenidas a partir de múltiples secuencias de resonancia magnética, como T1, T2, FLAIR o T1c. Cada una de estas modalidades ofrece una perspectiva diferente sobre el tejido tumoral y su entorno, desde el edema hasta el realce por contraste. La integración de estas secuencias dentro del mismo modelo permite capturar características complementarias, lo que mejora significativamente el rendimiento en tareas como segmentación, clasificación y predicción de progresión tumoral [Bakas et al., 2018, Reyes and Sánchez, 2024].

Los enfoques tradicionales suelen realizar esta fusión en la etapa de entrada, apilando los canales de cada modalidad en un único tensor. Sin embargo, otros modelos como HeMIS [Havaei et al., 2016], MMFNet [Zhou et al., 2018], MD-GRU [Andermatt et al., 2016] o HyperDenseNet [Dolz et al., 2019] adoptan una arquitectura más modular, en la que cada modalidad se procesa por separado mediante codificadores individuales, que luego se fusionan a nivel intermedio mediante mecanismos como atención, pooling adaptativo o agregación estadística. Este tipo de diseños ha demostrado ser más robusto ante modalidades ausentes y diferencias de calidad entre secuencias, lo que resulta especialmente útil en entornos clínicos donde no siempre se dispone de todas las adquisiciones de forma uniforme.

Las redes más modernas, como UNETR [Hatamizadeh et al., 2022a], SwinUNet [Cao et al., 2023], TransBTS [Wang et al., 2021b] o MCTrans [Zhang et al., 2021b], combinan el procesamiento volumétrico 3D con mecanismos Transformer jerárquicos, permitiendo capturar relaciones espaciales de largo alcance entre modalidades y planos. Esta combinación ha mostrado resultados muy competitivos en segmentación de imágenes médicas, superando a arquitecturas convolucionales clásicas tanto en precisión como en coherencia estructural. Además, el uso de codificadores separados para cada modalidad junto a fusión atencional o contextual, como en el caso de FusionNet [Valverde et al., 2020], ha permitido mejorar la generalización en conjuntos de datos de múltiples fuentes o con variabilidad técnica considerable.

2.5. Predicción clínica mediante regresión

La estimación de variables clínicas, como la edad del paciente, los días de supervivencia o el subtipo tumoral, a partir de datos médicos, ha sido un objetivo recurrente tanto para modelos clásicos como para técnicas de aprendizaje profundo. Esta línea de trabajo resulta especialmente relevante en neuroimagen oncológica, donde generar predicciones pronósticas precisas puede ayudar a personalizar tratamientos y mejorar la toma de decisiones clínicas.

Una de las estrategias más efectivas ha sido el aprendizaje multitarea, que permite combinar la segmentación del tumor con la regresión de variables clínicas dentro de una misma red neuronal. Al compartir representaciones internas entre ambas tareas, se logra una mayor coherencia y generalización del modelo, especialmente cuando las predicciones clínicas están ligadas a las características morfológicas de la lesión. Este enfoque ha sido validado en competiciones como el BraTS Challenge [Bakas et al., 2018], donde se han desarrollado arquitecturas capaces de realizar segmentación y predicción de supervivencia de forma conjunta [Wang et al., 2021a, Baldassarre et al., 2019].

Además, se han explorado enfoques más clásicos basados en modelos de regresión

entrenados sobre representaciones derivadas de las imágenes. Entre las fuentes de información más utilizadas se encuentran los embeddings latentes extraídos de autoencoders tridimensionales entrenados para reconstruir los volúmenes del paciente, así como descriptores radiómicos relacionados con la intensidad, textura o forma de las lesiones, generados con herramientas como PyRadiomics [van Griethuysen et al., 2017]. También se ha recurrido a estadísticas derivadas de las máscaras de segmentación, como el volumen tumoral, la intensidad media por región o la variabilidad intra-lesional [Kickingereder et al., 2016].

Estos datos han alimentado diversos algoritmos, entre los que destacan Regresión de Vectores de Soporte (SVR, en sus siglas en inglés) con núcleos Función de Base Radial (RBF, en sus siglas en inglés), apropiado para capturar relaciones no lineales en conjuntos pequeños [Mobadersany et al., 2018]; redes neuronales densas como los multilayer perceptrons, capaces de modelar patrones complejos; y métodos de boosting como XGBoost o HistGradientBoosting, altamente competitivos en tareas de predicción médica sobre datos tabulares [Chen and Guestrin, 2016, Zhou et al., 2019]. También se ha evaluado el uso de regresión bayesiana, por su capacidad para estimar incertidumbre [Rasmussen and Williams, 2006], y de modelos lineales regularizados con características polinómicas, como Ridge o Lasso, que pueden ofrecer una mayor interpretabilidad clínica.

Estudios recientes han demostrado que la combinación de estas técnicas con representaciones visuales estructuradas permite generar estimaciones más robustas y personalizadas. Diversos trabajos [Chartsias et al., 2019, Mobadersany et al., 2018, Zhou et al., 2019] destacan cómo el uso de *embeddings* visuales junto a descriptores clínicos mejora la precisión de las predicciones y refuerza la aplicabilidad de estos sistemas en entornos hospitalarios reales.

Capítulo 3

Metodología y planificación del trabajo

El desarrollo de un proyecto de investigación no solo requiere conocimientos técnicos, sino también una metodología clara y una planificación bien estructurada que orienten el trabajo desde sus fases iniciales hasta su conclusión. Para alcanzar los objetivos definidos, es imprescindible establecer una hoja de ruta que guíe cada decisión, coordine las tareas de forma lógica y facilite la evaluación continua de los resultados.

En este capítulo se detallan, por un lado, los principios metodológicos que han guiado el desarrollo del proyecto, incluyendo el modelo adoptado y las fases que lo componen; y por otro, la organización temporal y operativa del trabajo, desde la definición de tareas hasta su distribución a lo largo del calendario previsto.

Asimismo, se exponen las herramientas empleadas para la planificación y seguimiento del proyecto, así como los criterios utilizados para priorizar tareas, asignar recursos y documentar el progreso.

3.1. Metodología

La metodología seguida en este trabajo se estructura en torno al modelo CRISP-DM [Shearer, 2000], un estándar consolidado en el ámbito de la ciencia de datos por su enfoque iterativo, su flexibilidad frente a diferentes tipos de problemas y su orientación práctica. Este modelo propone una estructura dividida en seis fases fundamentales: comprensión del negocio, comprensión de los datos, preparación de los datos, modelado, evaluación y despliegue, tal como se muestra en la figura 3.1.

En el contexto académico y científico de este proyecto, se ha adaptado este marco metodológico para ajustarse a los objetivos investigativos del trabajo. En concreto, la fase de despliegue se ha sustituido por una etapa centrada en la documentación técnica, el análisis detallado de los resultados obtenidos y una reflexión crítica sobre los hallazgos. A continuación, se exponen las fases seguidas en el proyecto con más detalle:

El desarrollo del proyecto siguió una metodología estructurada que comenzó con la comprensión del problema, donde se definió el objetivo general: desarrollar un sistema basado en redes neuronales profundas capaz de segmentar automáticamente tumores cerebrales a partir de imágenes de resonancia magnética y, en algunos casos, predecir variables clínicas relevantes como los días de supervivencia. Para ello, se realizó una revisión bibliográfica exhaustiva del estado del arte en segmentación médica, redes multimodales, aprendizaje multitarea y predicción clínica. Esto permitió no solo identificar los enfoques

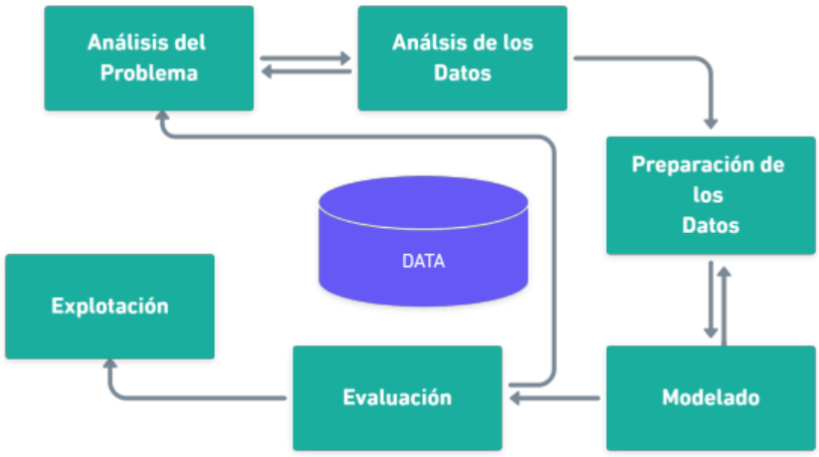


Figura 3.1: Diagrama de la metodología CRISP-DM. Fuente: Decidata [Decidata, 2023]

más prometedores (U-Net, UNETR, LViT, SVR, etc.), sino también delimitar los retos específicos del problema (combinación de modalidades, escasez de datos clínicos, necesidad de interpretabilidad, etc.).

La siguiente etapa se centró en la comprensión de los datos. Inicialmente se consideró el uso del conjunto Brain Cancer Brain Metastases (BCBM)-RadioGenomics, que combina imágenes médicas con máscaras de segmentación y variables clínicas. Sin embargo, se identificó que las segmentaciones no eran homogéneas entre pacientes y presentaban inconsistencias que dificultaban el entrenamiento de modelos robustos y la evaluación comparativa entre muestras. Por este motivo, se optó por utilizar finalmente el conjunto de datos BraTS2020 [Bakas et al., 2018] que incluye volúmenes 3D en formato Neuroimaging Informatics Technology Initiative (NIfTI) de imágenes cerebrales multimodales (T1, T1ce, T2, FLAIR), segmentaciones manuales y metadatos clínicos como edad, diagnóstico y supervivencia. Se analizó la estructura del conjunto de datos, su distribución, el número de pacientes, los formatos y las limitaciones asociadas (por ejemplo, presencia de valores ausentes o desequilibrio entre tipos de glioma).

La preparación de los datos implicó tareas clave de preprocesamiento: normalización de intensidades, alineación de las modalidades, recorte del cerebro, en el caso de que el modelo lo requiriese por las dimensiones que aceptaba como entrada, y reescalado espacial para uniformizar las dimensiones. También se generaron máscaras binarias y se separaron las particiones de entrenamiento, validación y test. En el caso de modelos multimodales, se integraron los datos clínicos estructurados y se adaptaron los formatos para ser compatibles con las arquitecturas seleccionadas. Para los modelos de regresión, se extrajeron descriptores latentes de las imágenes procesadas mediante codificadores.

Durante la fase de modelado se diseñaron e implementaron diversas arquitecturas de aprendizaje profundo. Inicialmente se exploraron modelos unimodales sencillos como ResNet34 y ResNet101, así como Segment Anything Model (SAM2) con pesos preentrenados, pero estos enfoques no lograron adaptarse adecuadamente a la segmentación de tumores cerebrales, produciendo predicciones incompletas o erróneas. Ante estos resultados insatisfactorios, se optó por implementar arquitecturas más especializadas para segmentación médica como U-Net, UNETR y SegResNet, que sí demostraron un ajuste adecuado a los datos y obtuvieron resultados satisfactorios. Para la parte de predicción clínica, se exploraron tanto modelos clásicos (SVR, Multi-Layer Perceptron - Perceptrón Multicapa (MLP), Gradient Boosting) como arquitecturas híbridas que integran cabezas de regresión

sobre embeddings visuales. En el caso de modelos multimodales, se implementaron variantes con fusión temprana, intermedia y tardía, integrando texto o variables estructuradas (como edad o diagnóstico). Además, se desarrolló un enfoque de aprendizaje multitarea que combina segmentación y regresión en una misma red.

La evaluación de los modelos se realizó utilizando métricas específicas para cada tarea: para segmentación se emplearon el coeficiente Dice, el IoU y la precisión pixel a pixel, mientras que para los modelos de regresión se recurrió al MAE, Root Mean Square Error (RMSE) y curvas de dispersión. Se realizaron múltiples experimentos variando arquitecturas, hiperparámetros y estrategias de entrenamiento para seleccionar las configuraciones más robustas. Además, se analizó el comportamiento por tipo de modalidad y combinación de fuentes de datos, lo que permitió validar la utilidad del enfoque multimodal frente al unimodal.

Finalmente, la documentación y análisis abarcó toda la experimentación mediante informes detallados, gráficos comparativos y registros de ejecución. Esta fase incluye la reflexión crítica sobre los resultados obtenidos, la identificación de limitaciones y posibles líneas de trabajo futuro.

3.2. Planificación

La planificación de un proyecto de investigación como este no solo sirve como guía temporal, sino que también permite organizar las tareas en fases lógicas y facilitar la toma de decisiones conforme se avanza. En este capítulo se expone, por un lado, la planificación inicial propuesta al inicio del trabajo, y por otro, una revisión de las tareas que se llevaron a cabo realmente, incluyendo los ajustes, dificultades y desviaciones detectadas respecto a la planificación original.

La planificación temporal del proyecto se estructuró inicialmente en cuatro grandes fases, con una estimación global de 600 horas de dedicación. Esta propuesta se recogió en el documento TFT02 presentado en la asignatura del proyecto, y sirvió como base para orientar la organización del tiempo, la distribución del esfuerzo y la asignación de prioridades. En la Tabla 3.1 se muestra dicha planificación dividida en fases y tareas previstas.

3.2.1. Planificación inicial

En la Tabla 3.1 se describen las fases propuestas, junto con las tareas inicialmente previstas en cada una. Estas incluían desde el análisis bibliográfico hasta el desarrollo e integración de modelos de segmentación y clasificación multimodal, pasando por la limpieza de datos, preprocesamiento y evaluación mediante métricas estándar. Esta estructura permitió abordar de forma ordenada las necesidades del proyecto, aunque, como es habitual en entornos reales de investigación, surgieron imprevistos que obligaron a redefinir parte del cronograma. Algunas fases se solaparon (como el modelado y la evaluación), y aparecieron tareas adicionales no contempladas inicialmente, como el análisis de múltiples configuraciones de fusión o la adaptación de modelos como LViT y SAM2.

Esta planificación inicial sirvió como base estructural, pero fue necesario adaptarla para incorporar modelos y técnicas no contempladas en la propuesta original, como la integración de SAM2 o el análisis detallado de errores en regresión. Estas adaptaciones se documentan en la sección de tareas realizadas (Tabla 3.2), donde se refleja la evolución real del proyecto frente a la planificación propuesta.

Tabla 3.1: Resumen de fases y planificación inicial del proyecto

Fase	**Tareas previstas**
Fase 1: Comprensión del problema y de los datos (90 h)	Definición del objetivo, revisión del estado del arte, análisis y exploración de los datos (TCIA, BraTS2020 u otros).
Fase 2: Preparación de datos y modelado (300 h)	Preprocesamiento y normalización de imágenes, codificación de datos clínicos, implementación y experimentación con modelos de segmentación y clasificación multimodal (U-Net, ViT, UNETR, LViT).
Fase 3: Evaluación y comparación de modelos (130 h)	Validación y comparación de modelos usando métricas como Dice, IoU o R^2, y evaluación del impacto de estrategias de fusión y datos clínicos.
Fase 4: Documentación y presentación (80 h)	Redacción del trabajo, preparación de la presentación oral y reflexión crítica sobre los resultados y mejoras futuras.

3.2.2. Tareas realizadas

En la Tabla 3.2 se recogen las tareas efectivamente ejecutadas durante el desarrollo del proyecto, distribuidas en las cuatro fases planificadas inicialmente. No obstante, muchas de estas tareas fueron ampliadas, modificadas o reestructuradas en función de la evolución real del trabajo, de los desafíos técnicos encontrados y de las decisiones que se tomaron tras los primeros resultados experimentales.

Tabla 3.2: Resumen de tareas realizadas durante el proyecto

Fase	**Tareas realizadas**
Estudio previo y análisis	Revisión de artículos sobre segmentación (U-Net, ViTs, SAM2, modelos multimodales), métodos de regresión (SVR, MLP, Random Forest, Gradient Boosting), y conjuntos de datos (BraTS2020 seleccionado). Análisis del conjunto de datos BraTS2020 y variables clínicas, técnicas de imputación y problemas en segmentación de clases minoritarias.
Diseño, desarrollo e implementación	Normalización y recorte de imágenes NIfTI, preparación de variables clínicas, implementación de redes 3D U-Net, SegResNet y LViT, desarrollo de cabezas de regresión, entrenamiento de modelos clásicos sobre representaciones latentes, integración de SAM2/MedSAM2, desarrollo de scripts de evaluación y visualización.
Evaluación, validación y prueba	Entrenamiento en GPU local, evaluación con Dice, IoU, MAE y MSE, comparación multimodal/unimodal, análisis de errores y sobreajuste, evaluación de la imputación de datos clínicos.
Documentación y presentación	Redacción, inclusión de figuras y gráficos, preparación de la defensa final y material complementario.

Si bien la planificación inicial resultó útil como hoja de ruta general, en la práctica se produjeron desviaciones importantes. La fase de análisis se desarrolló con normalidad,

permitiendo construir una base sólida de conocimiento sobre el estado del arte en segmentación y regresión clínica. Durante esta etapa también se reconsideró la base de datos utilizada: inicialmente se empleó el conjunto BCBM-RadioGenomics para el entrenamiento de redes unimodales y SAM2, pero debido a su inconsistencia en las segmentaciones y calidad de datos, se optó por BraTS2020, que presentaba mayor homogeneidad en sus segmentaciones y mejor calidad en sus datos clínicos estructurados. Con este nuevo conjunto se entrenaron tanto las redes neuronales de segmentación como los modelos de regresión.

Sin embargo, la fase de implementación se complicó debido a la necesidad de integrar modelos no contemplados inicialmente (como LViT y SAM2), lo que supuso una curva de aprendizaje y adaptación arquitectónica significativa.

Uno de los principales retos fue el tratamiento de las variables clínicas en los modelos de regresión. La imputación de valores ausentes con la media resultó en modelos poco expresivos y con baja varianza, ya que alteraba la distribución original y limitaba la capacidad predictiva. Para solucionar esto, se optó por eliminar los valores faltantes y aplicar una estrategia de aumento de datos centrada en los valores más altos y menos comunes de supervivencia, lo que permitió equilibrar la distribución y mejorar notablemente el rendimiento de los modelos. Este procedimiento se explicará con mayor detalle más adelante.

En la fase de evaluación, los experimentos con SAM2 también supusieron una desviación respecto a lo esperado. A pesar de utilizar pesos preentrenados de Meta y de entrenar distintas combinaciones de módulos (decodificador, codificador de memoria, codificador de prompts), el modelo no lograba segmentar con precisión. Las métricas iniciales parecían prometedoras, pero correspondían en muchos casos a falsos positivos o a predicciones sesgadas por el desequilibrio entre fondo y tumor.

También se detectaron problemas en modelos unimodales como ResNet34 y ResNet101, que no lograron adaptarse bien a la segmentación, produciendo predicciones incompletas o erróneas. Ante el fracaso de SAM2 y los modelos unimodales sencillos, se optó por implementar arquitecturas más especializadas para segmentación médica como 3D U-Net, SegResNet y LViT, que sí lograron ajustarse adecuadamente a los datos y obtuvieron resultados satisfactorios. Esto confirmó la hipótesis inicial de que la imagen por sí sola no era suficiente, reforzando la apuesta por enfoques multimodales que integraran información clínica estructurada.

La fase de documentación también se vio ampliada debido al número de modelos probados, configuraciones evaluadas y gráficas generadas. Se requirió un esfuerzo adicional para reflejar adecuadamente todos los resultados, especialmente aquellos que no fueron positivos, pero que aportaron información valiosa para la discusión.

3.3. Herramientas utilizadas

En esta sección se detallan los equipos empleados, los entornos de desarrollo configurados y las principales librerías utilizadas en cada fase del pipeline, abarcando desde la carga y preprocesamiento de imágenes médicas hasta la visualización y análisis de métricas de desempeño. Se describen también las herramientas auxiliares aplicadas a la documentación técnica, la anotación de datos clínicos y la gestión del control de versiones del código fuente.

La combinación de hardware adecuado, entornos virtuales bien gestionados y un ecosistema de librerías especializadas (como PyTorch, Medical Open Network for AI (MONAI), scikit-learn, nibabel, matplotlib, entre otras) ha permitido abordar de forma eficiente ta-

reas complejas de segmentación y regresión. Además, el uso de plataformas colaborativas, sistemas de control de versiones (Git/GitHub), y utilidades para la validación visual y la organización del trabajo, ha sido fundamental para garantizar una implementación funcional, reproducible y rigurosa del sistema diseñado.

3.3.1. Entorno de hardware

Durante el desarrollo del proyecto se utilizaron dos entornos computacionales complementarios. En una primera etapa, se trabajó con Google Colab, aprovechando su acceso gratuito a GPU NVIDIA Tesla T4, ideal para tareas iniciales como el preprocesamiento de datos, la exploración de modelos y la ejecución de pruebas rápidas. Su integración con Google Drive facilitó la gestión de conjuntos de datos y el almacenamiento temporal de resultados sin necesidad de configurar servidores locales.

Posteriormente, en las fases más exigentes del proceso, como el entrenamiento prolongado de modelos y la evaluación intensiva, se migró a un equipo local de alto rendimiento. Este equipo contaba con un procesador Intel Core i9-10900F con 10 núcleos y 20 hilos, 32 GB de memoria RAM DDR4, una tarjeta gráfica NVIDIA GeForce RTX 2060 SUPER con 8 GB de memoria GDDR6, y un disco SSD NVMe de 1 TB que aseguraba tiempos de lectura y escritura muy reducidos. El sistema operativo utilizado fue Windows 10 Pro de 64 bits, lo que permitió un entorno de trabajo estable y configurable para sesiones de entrenamiento más prolongadas y con mayor control sobre dependencias y versiones de librerías.

3.3.2. Entorno de software

La implementación del proyecto se realizó en un entorno de software moderno y orientado a ciencia de datos e IA, utilizando Windows 10 Pro de 64 bits como sistema operativo base. El desarrollo del código se llevó a cabo principalmente en Visual Studio Code, apoyado por extensiones para Python y Jupyter, y mediante el uso intensivo de Jupyter Notebooks para ejecución segmentada y documentación. Google Colab se empleó para prototipado rápido, entrenamiento preliminar y colaboración remota.

La gestión de entornos y dependencias se realizó con conda para garantizar reproducibilidad, y con venv y requirements.txt en entornos más ligeros o en Colab. El principal framework de IA fue PyTorch 2.0, utilizado para el diseño, entrenamiento y evaluación de modelos de segmentación y regresión. Sobre PyTorch se integró MONAI, un framework especializado en imagen médica que facilitó transformaciones volumétricas, manejo de datos médicos y arquitecturas predefinidas como SegResNet y UNETR, permitiendo una implementación eficiente y reproducible adaptada a datos cerebrales.

También se realizaron pruebas puntuales con TensorFlow 2.x para verificar funciones de pérdida personalizadas o métricas complementarias, aunque sin integrarlo en el pipeline principal. Este ecosistema proporcionó versatilidad, soporte específico para imagen médica y capacidad computacional, elementos clave para el desarrollo del proyecto.

3.3.3. Librerías empleadas

A lo largo del proyecto se emplearon librerías especializadas para todas las etapas del pipeline, desde el procesamiento hasta la visualización de resultados. Para manipulación y organización de datos se usaron `numpy`, `pandas` y `scipy`, junto con módulos estándar

de Python como `os, glob, json, pickle` e `itertools`, además de `tqdm`, `random`, `copy`, `shutil` y `time` para tareas auxiliares.

El tratamiento de imágenes médicas en formato NIfTI o DICOM se realizó con `nibabel`, `SimpleITK`, `torchio`, `nilearn` y `medpy`, permitiendo la lectura, transformación y visualización eficiente de datos tridimensionales cerebrales.

En el modelado y entrenamiento de redes neuronales se utilizaron `torch`, `torchvision` y `monai` (para arquitecturas como SegResNet o UNETR), `einops` para reestructuración de tensores, `segmentation_models.pytorch`, módulos de transformers (Hugging Face) y `albumentations` para aumentación de datos.

Para regresión y análisis estadístico se empleó `scikit-learn`, utilizando modelos como SVR, Random Forest, Gradient Boosting, HistGradientBoosting, MLP y modelos lineales con transformaciones polinómicas.

La visualización de resultados y métricas se realizó con `matplotlib`, `seaborn` y `plotly`, mientras que `tensorboard` se usó para monitorizar entrenamientos y `pydicom` para visualizar imágenes médicas en formato DICOM. Este conjunto de herramientas permitió construir un entorno robusto, reproducible y alineado con las mejores prácticas en investigación en imagen médica.

3.3.4. Otras herramientas

Además del ecosistema principal de librerías y frameworks, el proyecto utilizó herramientas auxiliares clave en distintas fases del desarrollo. Para integrar modelos preentrenados, se recurrió a Hugging Face Hub, facilitando la incorporación de arquitecturas como ViT, MedSAM2 o LViT. TensorBoard se empleó para monitorizar en tiempo real métricas de entrenamiento y comparar experimentos.

La documentación y redacción se realizó en LaTeX, tanto en Overleaf como en compiladores locales, permitiendo una organización estructurada, inclusión de figuras, tablas y gestión de bibliografía con BibTeX. La documentacióndel código se gestionó con Git y GitHub.

Para la validación visual y exploración tridimensional de segmentaciones se utilizó 3D Slicer, permitiendo inspeccionar resultados sobre cortes axiales, sagitales y coronales, y detectar errores anatómicos. Herramientas como Excel y Google Sheets se usaron para análisis exploratorio, visualización de distribuciones y gestión de muestras y planificación temporal.

En notebooks interactivos, se generaron animaciones 3D de segmentaciones con Matplotlib Animation e IPython.display para revisar secuencias de cortes, y se empleó Scikit-image (skimage.morphology) para procesar y refinar máscaras. Para la gestión de tareas y planificación personal, OneNote permitieron organizar tareas semanales, controlar errores y fijar objetivos a corto plazo, contribuyendo a una metodología de trabajo ordenada y adaptable.

Capítulo 4

Conjuntos de datos

El presente capítulo describe los conjuntos de datos utilizados durante el desarrollo del proyecto, detallando tanto su origen como sus características principales. La selección de estos conjuntos fue un aspecto fundamental para garantizar la validez de los experimentos, al proporcionar imágenes médicas realistas, segmentaciones manuales fiables y, en algunos casos, variables clínicas complementarias como edad o días de supervivencia. La calidad y diversidad de los datos constituyen elementos críticos en cualquier proyecto de aprendizaje automático aplicado al ámbito médico, donde la precisión diagnóstica y la robustez de los modelos dependen directamente de la representatividad y consistencia de las muestras utilizadas.

Durante el proceso de investigación se evaluaron múltiples conjuntos de datos públicos especializados en imagen médica cerebral, considerando criterios como la homogeneidad de las adquisiciones, la calidad de las anotaciones manuales, la disponibilidad de múltiples modalidades de resonancia magnética y la presencia de metadatos clínicos estructurados. Esta evaluación comparativa permitió identificar las fortalezas y limitaciones de cada conjunto, así como determinar cuáles resultaban más adecuados para los objetivos específicos del proyecto: segmentación automática de tumores cerebrales y predicción de variables clínicas relevantes. La documentación detallada de estas características facilita la reproducibilidad de los experimentos y proporciona una base sólida para futuras investigaciones en el área.

4.1. Brain Tumor Segmentation Challenge 2020

El conjunto de datos BraTS se ha consolidado a lo largo de los años como uno de los referentes internacionales más importantes en el ámbito de la imagen médica, especialmente en el estudio y segmentación automática de tumores cerebrales malignos. Este proyecto surge con el objetivo de abordar los principales retos clínicos asociados a la detección, caracterización y seguimiento de los gliomas, un tipo de tumor primario del sistema nervioso central que representa una de las formas más agresivas y frecuentes de cáncer cerebral.

Los gliomas presentan una gran heterogeneidad tanto entre pacientes como dentro de un mismo caso, lo que los convierte en una patología particularmente difícil de analizar. En una única imagen médica pueden observarse simultáneamente regiones necróticas, zonas del tumor ya muertas, edema y zonas activas que muestran realce tras la administración de contraste, lo que indica actividad tumoral agresiva. A ello se suman otras regiones que no presentan ningún tipo de realce, lo que complica aún más la interpretación visual.

Esta complejidad anatómica y funcional hace que la segmentación precisa del tumor sea un auténtico desafío. Incluso para radiólogos experimentados, delimitar de forma exacta los diferentes componentes de la lesión no siempre resulta evidente, debido a la falta de bordes definidos o a la superposición de características entre regiones. En este contexto, la ausencia de una segmentación clara puede dificultar tanto la planificación terapéutica como el seguimiento longitudinal del paciente, lo que refuerza la necesidad de sistemas automáticos fiables capaces de asistir en esta tarea.

Entonces, para responder a esta necesidad, BraTS propone un challenge anual que no solo facilita el acceso a un conjunto de datos curado y estandarizado, sino que también establece una línea base de referencia (benchmark) para evaluar modelos de IA en tareas críticas como la segmentación automática de tumores, la predicción de progresión y la estimación de la supervivencia del paciente.

Desde su lanzamiento en 2012, el conjunto de datos BraTS ha experimentado un crecimiento constante tanto en tamaño como en complejidad. Pasó de 50 casos iniciales a más de 500 en ediciones posteriores, incluyendo desde 2017 tareas adicionales como la predicción de supervivencia. Cada muestra incluye imágenes Multiparametric Magnetic Resonance Imaging (mpMRI), máscaras segmentadas y datos clínicos como edad, grado de resección y días de supervivencia.

A lo largo de los años, el reto BraTS ha incorporado desafíos más clínicos, como el análisis longitudinal para la progresión tumoral y la clasificación de pacientes según su esperanza de vida (corta, media o larga), basándose en características radiómicas derivadas de las segmentaciones.

Frente a otros conjuntos similares, BraTS destaca por la calidad de sus anotaciones expertas, la diversidad de origen clínico (19 centros), y su enfoque aplicado a la práctica médica real. Además, al ser una iniciativa pública, permite comparar modelos de forma estandarizada y reproducible mediante rankings oficiales.

4.1.1. Descripción del conjunto de datos

El conjunto de datos sigue el esquema de ficheros mostrado en la figura 4.2 en imágenes mpMRI preoperatorias, donde cada paciente cuenta con cuatro modalidades alineadas y registradas espacialmente. La estructura se organiza en dos carpetas principales: una contiene máscaras de segmentación y la otra no. En cada una de estas carpetas principales hay subcarpetas para cada paciente. Dentro de cada carpeta de paciente se encuentran la imagen anatómica estructural T1, la imagen T1 con contraste gadolinio (T1Gd), la imagen T2 sensible al contenido hídrico y la imagen T2-FLAIR, útil para detectar edema al suprimir el líquido cefalorraquídeo. Estas modalidades de imagen se pueden observar en la figura 4.3. En la carpeta principal de máscaras de segmentación, cada vóxel está etiquetado con un valor específico: 1 para el núcleo necrótico, 2 para el edema y 4 para el tumor realzado. Desde el año 2017, BraTS utiliza un esquema de anotación que agrupa estas etiquetas en tres grandes subregiones clave para el análisis automático. El tumor completo abarca todo el volumen tumoral, incluyendo el edema, por lo que esta máscara considera los vóxeles etiquetados como 1, 2 y 4. El núcleo tumoral comprende la necrosis, el tumor activo y las regiones no realzadas, teniendo en cuenta únicamente los vóxeles etiquetados como 1 y 4. Por último, el tumor realzado representa las áreas de realce activo más agresivas y solo incluye los vóxeles etiquetados como 4. A modo de ejemplo, la figura 4.1 muestra las diferentes segmentaciones en 2D de un paciente en concreto.

Además de las imágenes, cada carpeta principal del conjunto de datos está acompaña-

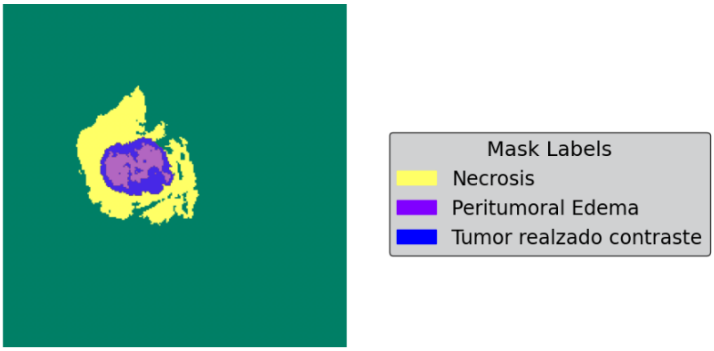


Figura 4.1: Máscara de segmentación de un paciente del conjunto de datos BraTS. Se visualizan las principales subregiones del glioma: necrosis (amarillo), correspondiente al tejido tumoral muerto; edema peritumoral (morado), que representa la acumulación de líquido inflamatorio alrededor del tumor; y tumor realzado por contraste (azul), que indica las áreas más activas y agresivas del glioma tras la administración de gadolinio.

da de archivos tabulares en formato `xlsx` que contienen información clínica relevante. En cada una de estas dos carpetas se encuentra un archivo de mapeo entre el identificador del sujeto y su grado tumoral (columna Grade), junto con los identificadores históricos (BraTS_2017_subject_ID, BraTS_2018_subject_ID, TCGA_TCIA_subject_ID, etc.) y el identificador actual BraTS_2020_subject_ID. También hay un segundo archivo que recoge variables clínicas clave como la edad del paciente, el número de días de supervivencia (Survival_days, si está disponible) y el grado de resección quirúrgica del tumor (Extent of Resection). Este último puede indicar resección completa del tumor visible (Resección Total Macroscópica (GTR, en sus siglas en inglés)) o resección parcial (Resección Subtotal (STR, en sus siglas en inglés)), información relevante para evaluar el pronóstico del paciente y su relación con la supervivencia.

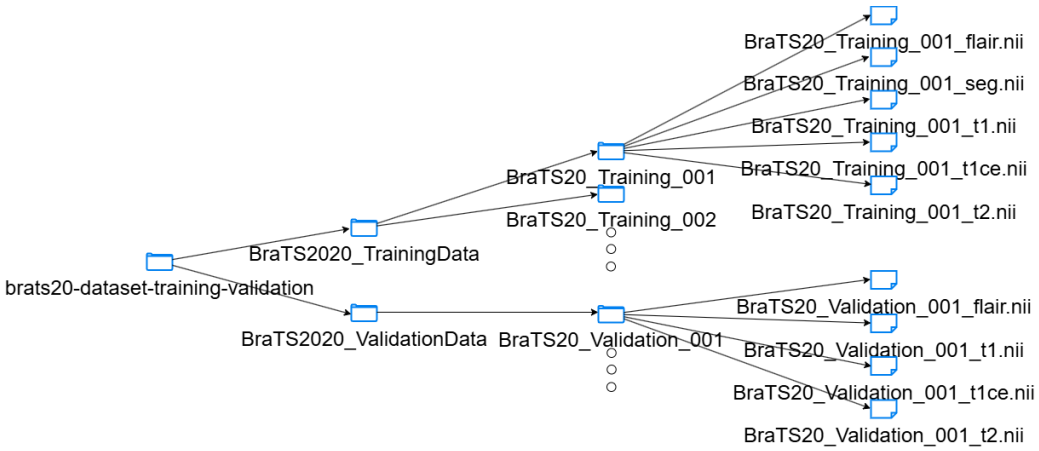
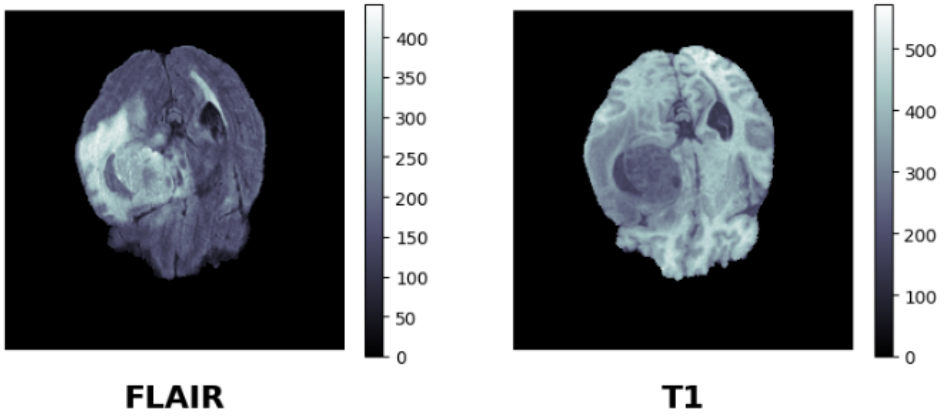


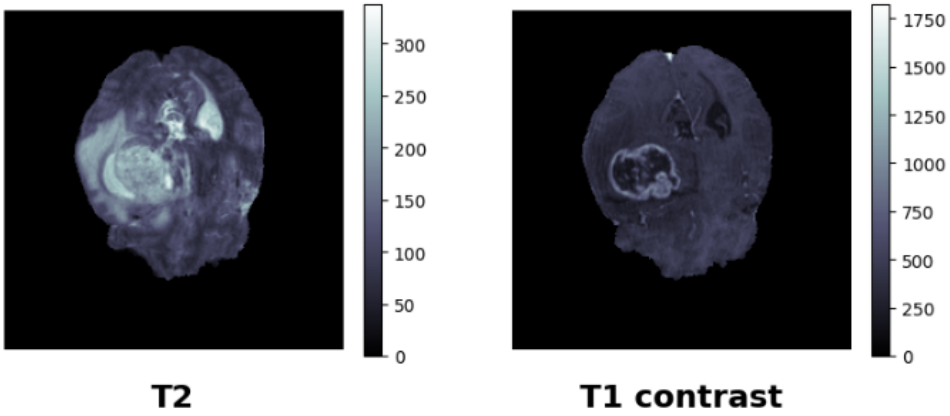
Figura 4.2: Estructura del conjunto de datos BraTS2020.

4.1.2. Preprocesamiento de las imágenes médicas

El conjunto de datos BraTS han sido cuidadosamente preprocesados siguiendo un protocolo estandarizado diseñado para garantizar la máxima uniformidad, consistencia y utilidad clínica a lo largo de todos los estudios incluidos. Este preprocesado es fundamental



(a) Modalidades FLAIR y T1. FLAIR permite observar el edema peritumoral, mientras que T1 proporciona la imagen anatómica estructural base.



(b) Modalidades T2 y T1 con contraste (T1Gd). T2 resalta zonas con contenido hídrico, y T1Gd revela las zonas más agresivas del tumor gracias al realce producido por el gadolinio.

Figura 4.3: Visualización de un mismo corte axial cerebral en las cuatro modalidades mpMRI utilizadas en el conjunto de datos BraTS.

para que los modelos de segmentación y predicción puedan generalizar correctamente, ya que en imágenes médicas pequeñas variaciones en la adquisición pueden suponer grandes diferencias en los resultados.

Uno de los pasos clave es alinear todas las imágenes a una plantilla anatómica común, lo que garantiza que las estructuras cerebrales se ubiquen en posiciones similares sin importar el hospital de origen. De este modo, se facilita tanto la comparación entre pacientes como el entrenamiento de modelos automáticos, al eliminar gran parte de la variabilidad interpaciente e interinstitucional.

Por otro lado, todas las imágenes han sido interpoladas a una resolución isotrópica de 1 milímetro cúbico por voxel, lo cual implica que cada pequeño “bloque” tridimensional (voxel) tiene la misma dimensión en los tres ejes espaciales. Esta regularidad en la resolución es especialmente importante cuando se trabaja con modelos 3D, ya que evita sesgos debidos a diferencias en la forma o tamaño de los vóxeles.

Otro paso importante es la eliminación del cráneo, también conocida como skull-stripping. Este procedimiento elimina estructuras externas al cerebro, como el hueso craneal, los ojos o el tejido extracerebral, que pueden confundir al modelo y que no aportan

información útil para la segmentación tumoral. De este modo, el resultado es una imagen centrada exclusivamente en el cerebro, limpia y más fácil de interpretar tanto para algoritmos como para humanos.

La estandarización de intensidades es también esencial, ya que las IRM no tienen un rango de intensidades fijo como ocurre en otras modalidades (por ejemplo, el TAC). Por eso, se aplica una normalización para que los valores de intensidad sean comparables entre estudios y entre modalidades (T1, T1Gd, T2, FLAIR), permitiendo así que un modelo aprenda patrones que no dependan del escáner o del hospital.

Uno de los aspectos más valiosos del BraTS es la calidad de sus anotaciones, ya que cada máscara de segmentación ha sido realizada manualmente por expertos clínicos con profundo conocimiento de la anatomía cerebral y los patrones de los gliomas. Estas segmentaciones, además, han sido revisadas y validadas por neurorradiólogos certificados con amplia experiencia en resonancias cerebrales. Este doble proceso de anotación experta y validación especializada garantiza etiquetas de altísima calidad, lo que resulta especialmente valioso para la comunidad científica y médica.

4.1.3. Análisis estadístico de variables clínicas

La figura 4.4 muestra la distribución de edades de los pacientes del conjunto BraTS2020, elaborada mediante un histograma en Excel a partir de la columna Age del archivo `survival_info.csv`. Las edades se agruparon en intervalos regulares y se contabilizó el número de pacientes en cada grupo, observándose una mayor concentración de casos entre los 50 y 70 años, lo que coincide con la epidemiología de los gliomas, más frecuentes en adultos de mediana y avanzada edad. Por su parte, la figura 4.5 representa la distribución de los días de supervivencia (Survival_days) desde el diagnóstico, también calculada en Excel mediante intervalos de 100 días. Esta distribución refleja una amplia variabilidad sin un patrón claramente definido y una mayor concentración en valores bajos, evidenciando la complejidad de predecir la evolución clínica en pacientes con gliomas agresivos.

El conjunto BraTS2020 está compuesto por 369 pacientes, de los cuales 259 incluyen información clínica estructurada completa con edad y días de supervivencia. Cada paciente en la carpeta de máscaras de segmentación dispone de cinco archivos en formato NIfTI, de dimensiones $240 \times 240 \times 155$ vóxeles, sumando un total de 1.845 volúmenes 3D. Este diseño cuidadoso facilita el estudio y la evaluación de modelos de segmentación, clasificación y predicción de supervivencia en tumores cerebrales.

Los resultados mostrados en la figura 4.6 evidencian un patrón preocupante en la completitud de los datos clínicos del conjunto. El gráfico de radar revela que las variables Age y Survival_days presentan exactamente el mismo porcentaje de completitud (64 % de datos presentes y 36 % de faltantes), lo que indica que ambos datos están vinculados en el archivo de información de supervivencia. En contraste, la variable Extent_of_Resection apenas alcanza un 35 % de completitud, dejando a dos tercios de los pacientes sin información sobre el tipo de resección quirúrgica realizada.

Esta correlación entre los datos faltantes de edad y supervivencia indica que BraTS2020 fue diseñado principalmente como un conjunto de imágenes médicas, relegando la información clínica a un segundo plano [Menze et al., 2014]. Esto es habitual en estudios retrospectivos, donde el seguimiento clínico puede perderse por transferencias de pacientes o diferencias en los protocolos hospitalarios.

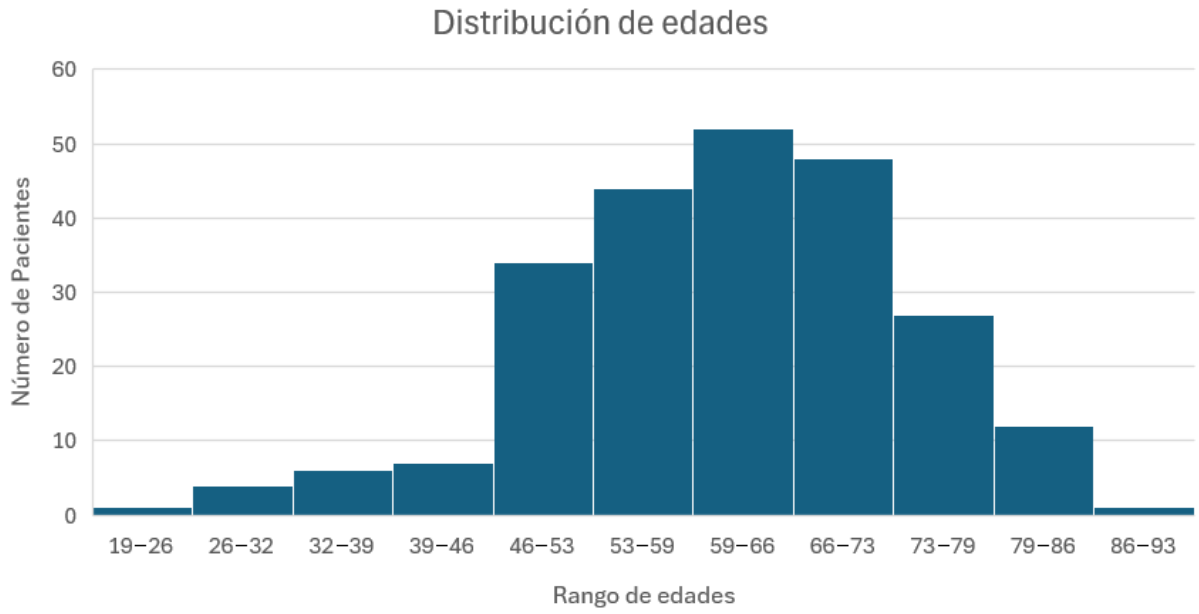


Figura 4.4: Distribución de edades redondeadas de los pacientes incluidos en el conjunto de datos BraTS.

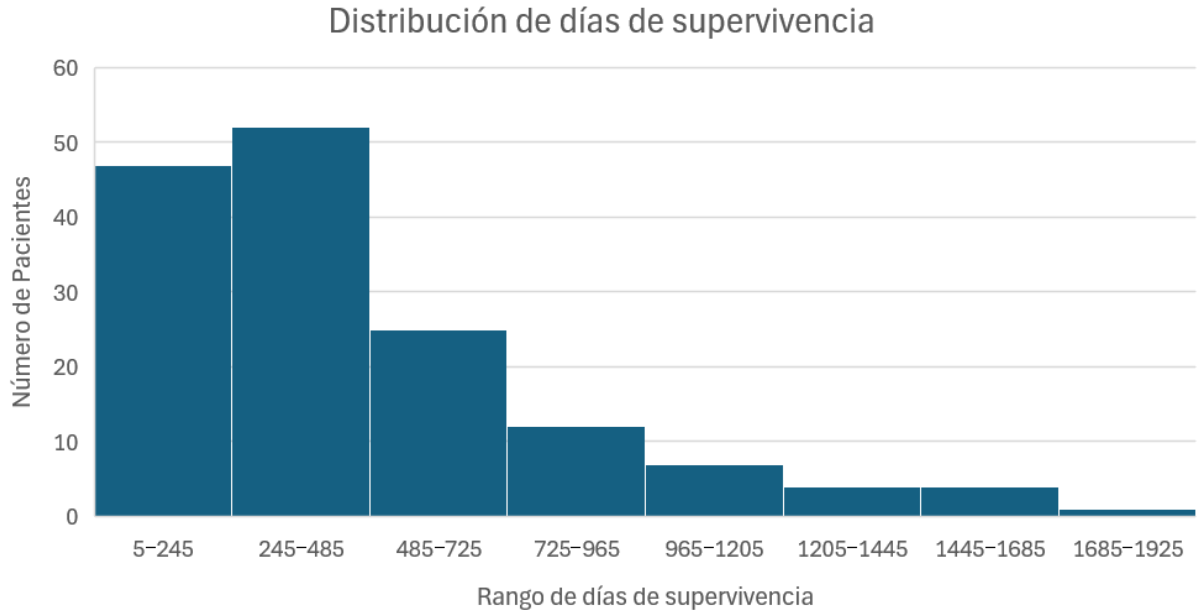


Figura 4.5: Distribución de los días de supervivencia (redondeados a la decena más cercana) de los pacientes del conjunto de datos BraTS.

4.2. RM cerebral metastásica con segmentación y radiómica

El conjunto de datos BCBM-RadioGenomics, disponible públicamente en TCIA, recoge el seguimiento de pacientes con metástasis cerebrales derivadas de cáncer de mama. Incluye imágenes de IRM en formato NIfTI comprimido, segmentaciones manuales 3D del tumor, la cavidad posoperatoria y estructuras vecinas de interés, todo ello ilustrado en la figura 4.7 con diferentes cortes y una segmentación tridimensional. Además, incorpora

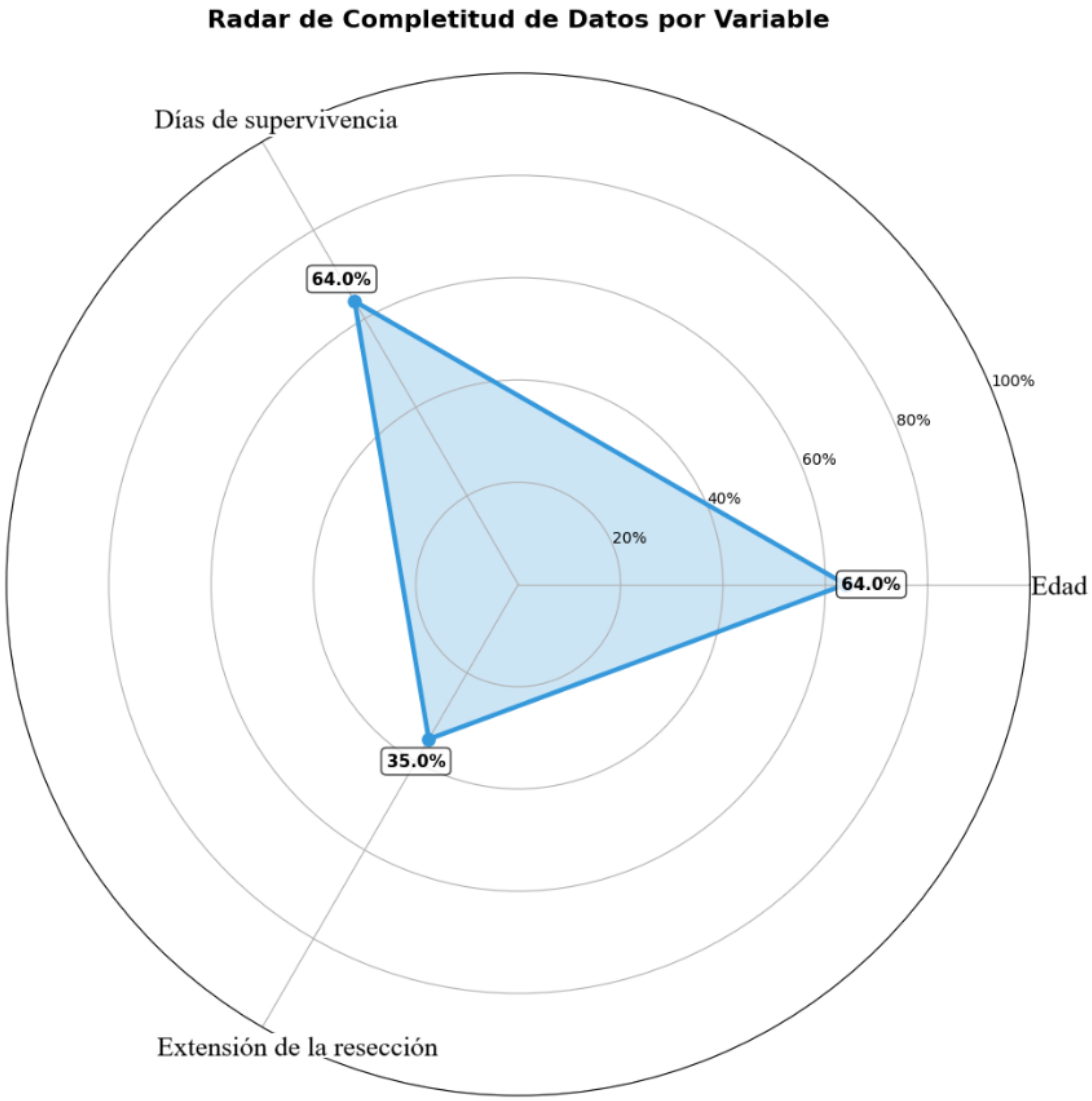
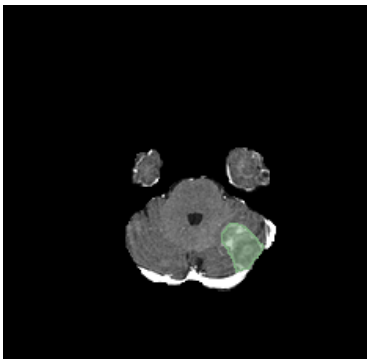


Figura 4.6: Radar de completitud de datos clínicos en el conjunto BraTS2020. Se observa que las variables Edad y Días de supervivencia presentan un 64 % de datos presentes, mientras que la variable Extensión de la resección solo alcanza un 35 %.

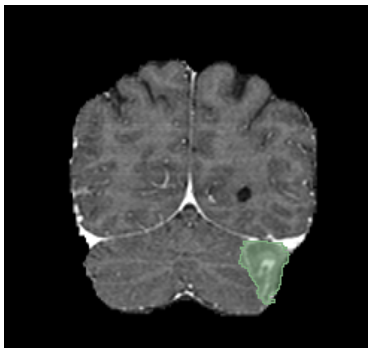
datos clínicos y genéticos como edad, sexo y el estado de los biomarcadores Receptor de Progesterona (PR, en sus siglas en inglés), Receptor de Estrógeno (ER, en sus siglas en inglés) y Factor Receptor 2 (HER2, en sus siglas en inglés), junto a 107 descriptores radiómicos extraídos mediante PyRadiomics. Estos descriptores cuantifican características como forma, textura, intensidad y patrones en las regiones de interés, permitiendo un análisis más profundo para mejorar el diagnóstico, el pronóstico y la personalización del tratamiento.

La metástasis cerebral en cáncer de mama constituye uno de los mayores retos en oncología. Las células tumorales, tras desprenderse del tejido mamario, atraviesan la barrera hematoencefálica utilizando mecanismos moleculares que degradan las paredes vasculares. Una vez en el cerebro, reprogramanan su metabolismo y manipulan células cerebrales como astrocitos y microglía para favorecer su supervivencia y crecimiento, formando colonias tumorales resistentes. Esta situación crea un círculo vicioso: los tumores dañan estructuras vitales y la barrera hematoencefálica dificulta el acceso de los tratamientos, lo que

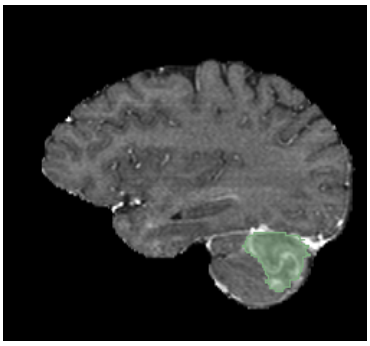
explica los pronósticos reservados y la necesidad de terapias innovadoras.



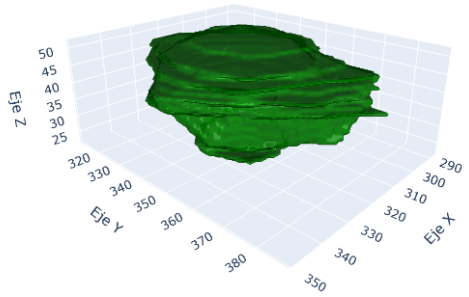
(a) Corte axial del paciente 5.



(b) Corte coronal del paciente 5.



(c) Corte sagital del paciente 5.



(d) Volumen segmentado de la cavidad cerebelar del hemisferio izquierdo.

Figura 4.7: Diferentes vistas y segmentación del paciente 5.

4.2.1. Estructura del conjunto de datos

El conjunto de datos está conformado por 3 elementos:

1. Una carpeta, compuesta a su vez por carpetas que corresponden al paciente y al número de seguimiento, donde se encuentra la imagen del cerebro junto con las segmentaciones correspondientes.

2. Una hoja de cálculo donde se incluyen los datos clínicos de cada paciente en el momento del seguimiento. Por ejemplo, a un paciente pueden haberle hecho 2 seguimientos, pues se incluirá en esta hoja de cálculo los datos clínicos del paciente en esos dos momentos.

3. Una hoja de cálculo donde se incluyen datos radiómicos de cada segmentación hecha a mano.

Esta estructura se puede ver esquematizada en la figura 4.8. El conjunto de datos estará compuesto por 3 elementos principales: (1) una carpeta con imágenes cerebrales y segmentaciones organizadas por paciente y número de seguimiento, (2) una hoja de cálculo con datos clínicos de cada paciente en distintos momentos del seguimiento, y (3) una

hoja de cálculo con datos radiómicos extraídos manualmente de las segmentaciones. Cada subcarpeta dentro de la carpeta de imágenes representa un paciente en un seguimiento específico, conteniendo archivos NIfTI (`.nii.gz`) con la imagen del cerebro y segmentaciones de tumores (que puede ser antes o después de su extracción) y regiones cercanas o de interés.

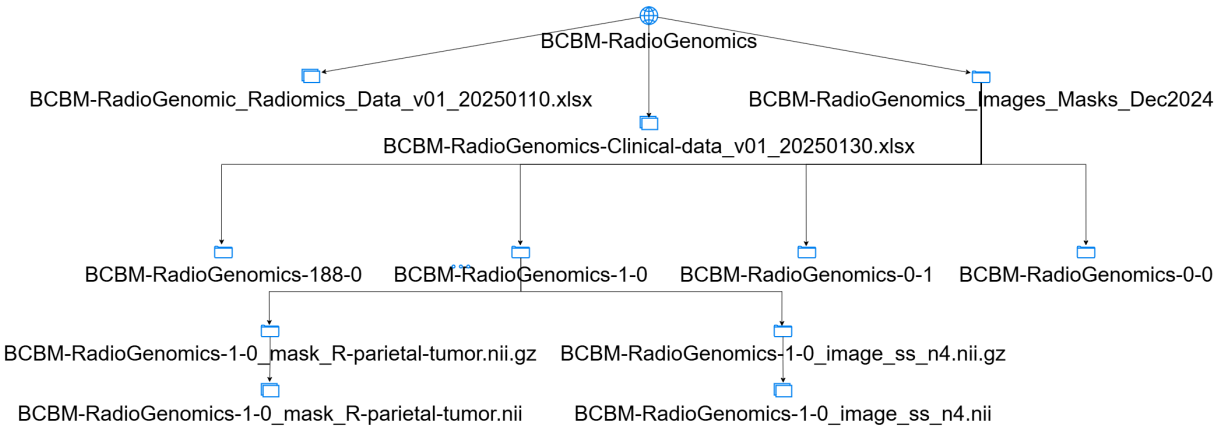


Figura 4.8: Estructura del conjunto de datos BCBM-RadioGenomics.

4.2.2. Evaluación y justificación de la exclusión del conjunto de datos

La exclusión del conjunto de datos BCBM-RadioGenomics de los resultados finales fue una decisión complicada, ya que este dataset fue fundamental en las primeras etapas del proyecto. Sin embargo, durante el desarrollo se identificaron importantes limitaciones metodológicas que afectaban la eficacia y reproducibilidad de los modelos de segmentación. El principal problema radicaba en la falta de homogeneidad entre las máscaras de segmentación: algunos pacientes contaban con hasta tres máscaras para diferentes regiones cerebrales, mientras que otros solo tenían una, lo que generaba un entorno de entrenamiento desigual y dificultaba el aprendizaje coherente de los algoritmos.

Adicionalmente, el propio conjunto de datos presentaba inconsistencias en las segmentaciones y en la nomenclatura de las etiquetas. Estas diferencias surgían porque varios técnicos participaron en el proceso de anotación, utilizando abreviaciones distintas, cometiendo errores ortográficos o empleando acrónimos variados. La falta de estandarización en la nomenclatura y en los criterios de segmentación complicaba aún más la generalización de los modelos.

Estas inconsistencias afectaron negativamente el rendimiento de los modelos, dificultando la obtención de resultados estables y fiables con cualquier arquitectura. La desigualdad en los datos y la variabilidad en el número de clases a segmentar complicaron la optimización y redujeron la capacidad de generalización. Como resultado, las métricas obtenidas fueron poco consistentes.

A pesar de estas limitaciones, la experiencia adquirida con BCBM-RadioGenomics fue valiosa y por ello se decidió incluirlo en la memoria. Trabajar con este conjunto permitió experimentar con redes neuronales para la segmentación de metástasis cerebrales y explorar distintas estrategias en el ámbito médico.

Capítulo 5

Redes neuronales empleadas

El presente capítulo tiene como objetivo describir en detalle las arquitecturas de redes neuronales profundas empleadas en el desarrollo del sistema propuesto, que constituyen el núcleo del proyecto al posibilitar tanto la segmentación de imágenes médicas como la predicción de variables clínicas relevantes, como la supervivencia de los pacientes.

Se presentan y analizan distintas arquitecturas implementadas, agrupadas según su propósito. Entre las redes de segmentación se incluyen 3D U-Net, SegResNet y UNETR, todas diseñadas para trabajar con volúmenes cerebrales en formato NIfTI, así como variantes avanzadas que incorporan mecanismos de atención o codificadores jerárquicos. Por otro lado, se abordan modelos de regresión basados en redes densas, transformadores y enfoques híbridos que integran visión por computadora y texto clínico.

Además de la descripción de las arquitecturas, el capítulo aborda las motivaciones de diseño, las adaptaciones realizadas sobre modelos existentes y otros aspectos clave que han guiado el desarrollo y la selección de las redes utilizadas.

5.1. Red neuronal 3D U-NET

El avance de las CNNs ha revolucionado el procesamiento de imágenes, pero las arquitecturas tradicionales no eran suficientes para la segmentación biomédica, que requiere una alta precisión espacial incluso con pocos datos. Para enfrentar este reto, en 2015 se introdujo la red U-Net, creada específicamente para lograr una segmentación médica eficiente y precisa.

La gran innovación de U-Net fue la adición de un camino de expansión simétrico al codificador y el uso de conexiones de salto entre capas que se corresponden, lo que permite recuperar detalles espaciales que se pierden durante el *downsampling*. Esta estructura la estableció como un modelo de referencia en el campo de la segmentación médica.

Con la creciente necesidad de segmentar volúmenes 3D como los de IRM o TC, en 2016 apareció 3D U-Net, una extensión que sustituye todas las operaciones 2D por sus equivalentes en 3D, permitiendo procesar volúmenes completos y capturar relaciones espaciales en las tres dimensiones.

5.1.1. Arquitectura de la red neuronal

La arquitectura implementada corresponde a una red de tipo 3D U-Net, diseñada específicamente para tareas de segmentación semántica en volúmenes médicos, como imágenes de resonancia magnética o tomografía computarizada. Esta red presenta una estructura

simétrica, reflejada en la figura 5.1, compuesta por una fase de codificación (izquierda), un cuello de botella (centro) y una fase de decodificación (derecha). Las flechas verdes representan conexiones de salto que concatenan mapas de características del codificador con los del decodificador, lo que facilita la recuperación de detalles espaciales perdidos durante la compresión. Cada bloque azul representa operaciones de convolución 3D seguidas de normalización y activación ReLU, permitiendo así segmentar volúmenes 3D con alta precisión espacial.

El funcionamiento de las convoluciones puede entenderse a partir de un ejemplo en el que un filtro se desplaza sobre una imagen de entrada, multiplicando y sumando los valores superpuestos para generar la matriz de salida. Este proceso permite extraer patrones locales, como bordes o texturas, en la imagen y constituye la base del procesamiento en redes convolucionales. En el caso de la 3D U-Net, este mismo principio se extiende a datos volumétricos, aplicando filtros 3D sobre bloques de datos en las tres dimensiones espaciales.

Desde un punto de vista matemático, las convoluciones 3D extienden el concepto de convolución bidimensional a volúmenes. Dado un volumen de entrada $\mathbf{X} \in \mathbb{R}^{D\times H\times W}$, donde D es la profundidad, H la altura y W la anchura, y un filtro 3D $\mathbf{K} \in \mathbb{R}^{d\times h\times w}$, la operación de convolución 3D en una posición (i, j, k) se define como:

$$Y(i,j,k) = \sum_{u=0}^{d-1}\sum_{v=0}^{h-1}\sum_{w=0}^{w-1} X(i+u, j+v, k+w)\cdot K(u,v,w),$$

donde Y es el mapa de activaciones generado por el filtro $\mathbf{K}$ al recorrer el volumen de entrada $\mathbf{X}$. Así, la red es capaz de aprender y combinar información contextual en todo el volumen, logrando segmentaciones más precisas y coherentes en comparación con métodos basados únicamente en convoluciones bidimensionales.

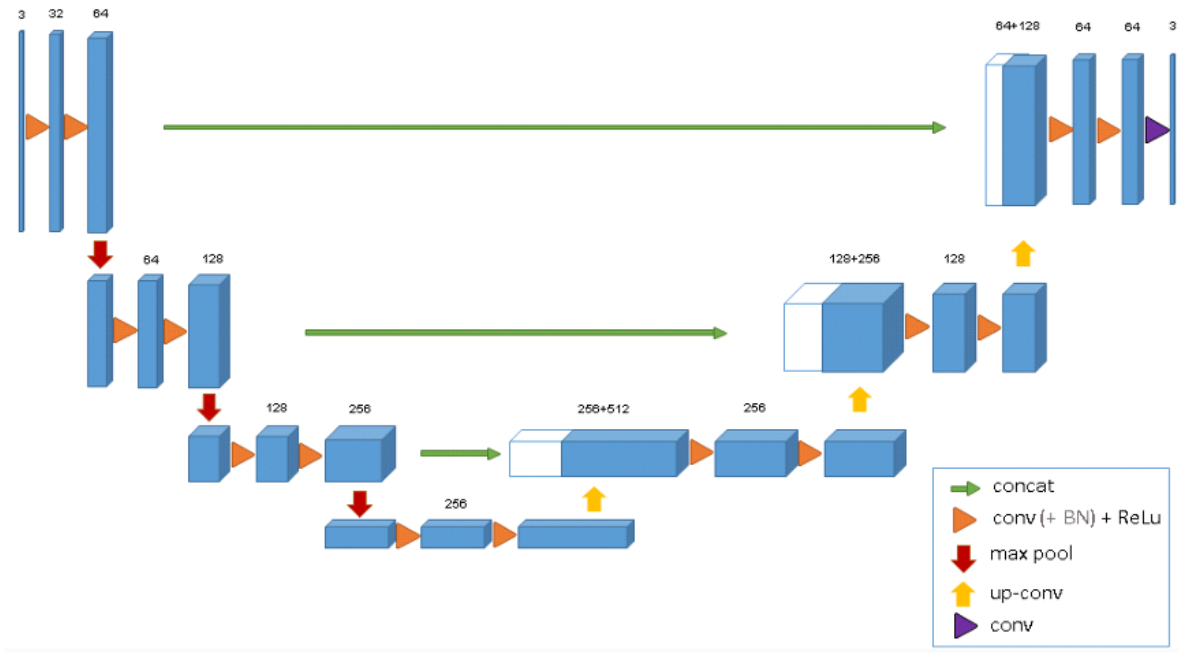


Figura 5.1: Esquema de la arquitectura U-Net 3D. Fuente: [Çiçek et al., 2016]

5.1.2. Estructura general

La red sigue la estructura en forma de 'U' característica de U-Net, compuesta por dos trayectorias principales: un codificador o camino de contracción, que reduce progresivamente las dimensiones espaciales del volumen de entrada mientras incrementa la profundidad de los mapas de características, y un decodificador o camino de expansión, encargado de recuperar las dimensiones espaciales originales mediante operaciones de *upsampling* y de refinar la segmentación utilizando información combinada del codificador a través de conexiones de salto.

En el núcleo de la arquitectura se encuentran los bloques `DoubleConv`, cada uno de los cuales está compuesto por dos capas convolucionales 3D (`Conv3d`) seguidas de normalización por grupos (`GroupNorm`) y una función de activación no lineal `ReLU`. Se emplean convoluciones con `kernel_size=3`, `stride=1` y `padding=1` para preservar las dimensiones espaciales. Este bloque realiza la siguiente operación dos veces de forma secuencial:

$$y = \text{ReLU}\left(\text{GroupNorm}\left(\text{Conv3D}(x)\right)\right).$$

El proceso de codificación se lleva a cabo a través de bloques denominados `Down`, que constan de una operación de `MaxPool3D` con factor de reducción $2 \times 2 \times 2$, seguida por un bloque `DoubleConv`. El objetivo de estos bloques es reducir el tamaño del volumen y aumentar el nivel de abstracción de las características aprendidas:

$$x_{\text{down}} = \text{DoubleConv}\left(\text{MaxPool3D}(x)\right).$$

En la etapa de decodificación, cada bloque `Up` realiza primero una operación de *upsampling*, mediante interpolación trilineal o `ConvTranspose3D`, para duplicar las dimensiones espaciales. Posteriormente, concatena el volumen reescalado con la correspondiente salida del codificador (con la misma forma) y aplica un bloque `DoubleConv`:

$$x_{\text{up}} = \text{DoubleConv}\left(\text{concat}\left(\text{Upsample}(x_1), x_2\right)\right).$$

Para asegurar la compatibilidad en las dimensiones antes de la concatenación, se aplica un *padding* que iguala las formas de los tensores implicados.

Finalmente, la última capa de la red es una convolución 3D con `kernel_size=1`, que proyecta el mapa de características resultante en un espacio de salida con tantas dimensiones como clases (número de etiquetas de segmentación):

$$\text{mask} = \text{Conv3D}_{1\times1}(x).$$

El resultado es un volumen 3D en el que cada *voxel* contiene las puntuaciones asociadas a cada clase.

5.2. Red neuronal LViT

En los últimos años, el aprendizaje profundo ha mejorado notablemente la segmentación médica, especialmente con modelos como U-Net y sus variantes. Estos enfoques han demostrado eficacia en la extracción de características locales, pero presentan limitaciones para capturar contexto global y dependen fuertemente de anotaciones manuales de alta calidad.

Paralelamente, los modelos multimodales que combinan imagen y texto, como CLIP o ViLT, han ganado relevancia, aunque su adaptación al entorno médico no es directa

debido a las particularidades de las imágenes clínicas (bajo contraste, alta variabilidad, estructuras poco diferenciadas).

En este contexto surge LViT, un modelo específicamente diseñado para segmentación médica multimodal. Integra texto clínico en el proceso de segmentación sin necesidad de modelos lingüísticos complejos como BERT, utilizando en su lugar convoluciones 1D para incorporar información textual de forma eficiente en distintos niveles jerárquicos de la red.

5.2.1. Arquitectura de la red neuronal

La arquitectura LViT es un modelo multimodal diseñado para la segmentación semántica de imágenes médicas, combinando como entradas imágenes (radiografías o TACs) y textos clínicos (informes radiológicos o datos estructurados). Presenta una estructura de doble U, con una rama CNN para preservar detalles locales y una rama basada en ViT que fusiona jerárquicamente la información textual, como se observa en la figura 5.2. El texto clínico se vectoriza mediante una capa BERT-Embed e ingresa en la rama Transformer, mientras que la imagen se procesa en la rama CNN, conectándose ambas mediante módulos de interacción CNN-ViT. Los módulos Pixel-Level Attention Module (PLAM), situados en los saltos de conexión, aplican atención local a nivel de píxel combinando Global Average Pooling (GAP) y Global Max Pooling (GMP), lo que permite refinar la máscara de segmentación tanto espacial como semánticamente.

LViT integra tres componentes clave: una rama CNN para capturar detalles espaciales, una rama Transformer jerárquica que modela relaciones semánticas globales informadas por texto, y el módulo PLAM que preserva la precisión en los bordes. En su versión semi-supervisada, incorpora el mecanismo Exponential Pseudo-label Iteration (EPI), que mejora la calidad de las pseudoetiquetas utilizando el texto clínico como guía, permitiendo refinar progresivamente las etiquetas generadas en escenarios con escasez de anotaciones.

Este diseño permite a LViT capturar simultáneamente el contexto semántico global del texto y la estructura espacial fina de la imagen, mejorando el rendimiento incluso con datos o anotaciones limitadas. Los resultados experimentales demuestran que LViT supera a modelos previos tanto en segmentación supervisada como semi-supervisada, manteniendo altos niveles de precisión incluso utilizando solo el 25 % de las anotaciones reales.

5.2.2. Estructura general

LViT se compone de dos bloques principales: un condificador, que reduce progresivamente la resolución espacial mientras fusiona información de imagen y texto, y un decodificador, encargado de reconstruir la segmentación espacial final combinando mecanismos de atención y operaciones convolucionales. El modelo recibe como entradas una imagen $\mathbf{X} \in \mathbb{R}^{B\times 3\times 224\times 224}$ y un texto $\mathbf{T} \in \mathbb{R}^{B\times L\times d_T}$, siendo $d_T = 768$.

En la etapa de preprocesamiento, la imagen pasa primero por un bloque inicial convolucional, generando un mapa de características con C canales (típicamente 64) mediante la operación $\mathbf{x}_1 = \text{ReLU}(\text{BN}(\text{Conv2D}(\mathbf{X})))$. Por su parte, el texto se adapta mediante una serie de convoluciones 1D, de modo que $\mathbf{T}_i = \text{Conv1D}(\mathbf{T}_{i-1})$ para $i = 1, 2, 3, 4$, obteniéndose embeddings text_1 a text_4 alineados jerárquicamente.

El codificador alterna bloques ViT y bloques DownBlock. Los DownBlock reducen la resolución espacial mediante CNNs, mientras que los bloques ViT aplican patch embedding, fusionan el texto embebido y emplean mecanismos MHSA y FFN. Matemáticamente, la autoatención se expresa como $\text{MHSA}(\mathbf{X}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V}$ y la red feed-forward

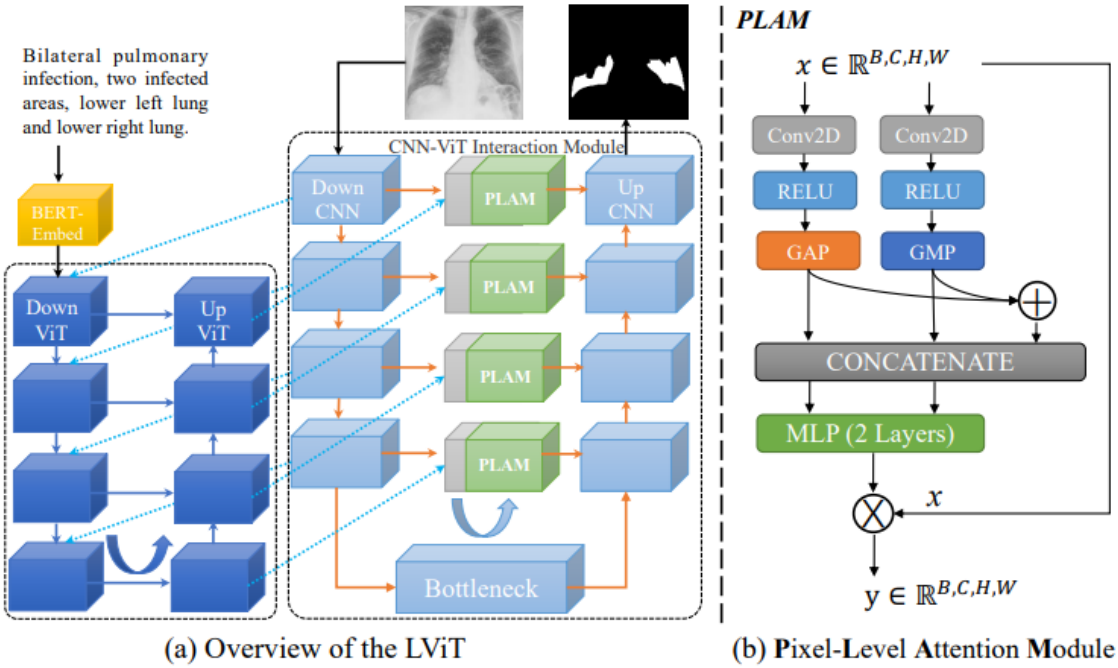


Figura 5.2: Esquema general del modelo LViT. Fuente: [Wang et al., 2022]

como $\mathrm{FFN}(\mathbf{X}) = \mathrm{MLP}(\mathrm{LayerNorm}(\mathbf{X}))$. El Vision Transformer convierte imágenes en secuencias de parches embebidos como vectores, enriquecidos con información posicional, y en variantes multimodales como LViT, estos vectores se combinan con texto clínico embebido. Así, los bloques Transformer capturan relaciones espaciales y semánticas, lo cual es especialmente útil en contextos médicos donde la combinación de imagen y texto mejora la interpretación.

A nivel de arquitectura, el codificador sigue la secuencia:

$$\begin{aligned} \mathbf{y}_1 &= \mathrm{ViT}(\mathbf{x}_1, \mathbf{x}_1, \mathrm{text}_1), & \mathbf{x}_2 &= \mathrm{DownBlock}(\mathbf{x}_1). \\ \mathbf{y}_2 &= \mathrm{ViT}(\mathbf{x}_2, \mathbf{y}_1, \mathrm{text}_2), & \mathbf{x}_3 &= \mathrm{DownBlock}(\mathbf{x}_2). \\ \mathbf{y}_3 &= \mathrm{ViT}(\mathbf{x}_3, \mathbf{y}_2, \mathrm{text}_3), & \mathbf{x}_4 &= \mathrm{DownBlock}(\mathbf{x}_3). \\ \mathbf{y}_4 &= \mathrm{ViT}(\mathbf{x}_4, \mathbf{y}_3, \mathrm{text}_4), & \mathbf{x}_5 &= \mathrm{DownBlock}(\mathbf{x}_4). \end{aligned}$$

Tras el paso por los bloques DownBlock y Vision Transformer, el modelo entra en la etapa de decodificación, cuyo objetivo es revertir progresivamente la compresión espacial y semántica aplicada durante el codificado y reconstruir un mapa de segmentación con la misma resolución que la imagen original. Se aplican bloques UpViT, que funcionan como transformadores inversos y reciben como entradas la salida del bloque ViT correspondiente a un nivel jerárquico ($\mathbf{y}_i$), la salida del nivel superior en el decodificador ($\hat{\mathbf{y}}_{i+1}$), que contiene información de resolución inferior pero semánticamente más rica, y el embedding textual adaptado a ese nivel (text_i). El bloque UpViT aplica una atención cruzada entre estas entradas para reconstruir una representación enriquecida en un nivel superior, siguiendo $\hat{\mathbf{y}}_i = \mathrm{UpViT}(\mathbf{y}_i, \hat{\mathbf{y}}_{i+1}, \mathrm{text}_i)$. Posteriormente, esta salida reconstruida se transforma y reescala para alinearse espacialmente con las salidas del codificador mediante el módulo Reconstruct, es decir, $\mathbf{x}_i = \mathbf{x}_i + \mathrm{Upsample}(\mathrm{Conv2D}(\hat{\mathbf{y}}_i))$. Este paso cumple dos funciones esenciales: reescalar la salida de ViT para que tenga la misma resolución que el mapa original del codificador (conexiones de salto) y fusionar información semántica

global (ViT) con información local (CNN), enriqueciendo las representaciones intermedias antes de la decodificación final.

A continuación, se aplica un decodificador convolucional que restituye la resolución espacial completa y permite obtener la segmentación final. Cada bloque UpCNN realiza un upsampling de la representación actual mediante interpolación bilineal o nearest neighbor, seguido de un módulo PLAM que resalta las regiones más relevantes de las representaciones extraídas del codificador, aplicando $\mathbf{s}' = \sigma(\text{MLP}([\text{GAP}(\mathbf{s}); \text{GMP}(\mathbf{s})])) \cdot \mathbf{s}$, donde GAP es global average pooling, GMP es global max pooling, MLP es un perceptrón multicapa y σ es la función sigmoide. Posteriormente, se fusiona el resultado *upsampled* con la conexión de salto mejorado por PLAM mediante una convolución: $\mathbf{x}_{\text{dec}} = \text{Conv2D}([\mathbf{x}^{\uparrow}, \mathbf{s}'])$. Este proceso se realiza jerárquicamente desde la salida más profunda hasta las características de resolución completa: $\text{UpCNN}(\mathbf{x}_5, \mathbf{x}_4) \rightarrow \text{UpCNN}(\cdot, \mathbf{x}_3) \rightarrow \text{UpCNN}(\cdot, \mathbf{x}_2) \rightarrow \text{UpCNN}(\cdot, \mathbf{x}_1)$.

Finalmente, una vez alcanzada la resolución original, se aplica una capa de salida para obtener el mapa de segmentación final: $\mathbf{S} = \sigma(\text{Conv2D}_{1\times1}(\mathbf{x}_{\text{final}}))$, donde $\sigma = \text{Sigmoid}$ para segmentación binaria y $\sigma = \text{Softmax}$ para segmentación multiclase. La salida final es $\mathbf{S} \in \mathbb{R}^{B\times n_{\text{clases}}\times 224\times 224}$, que representa el mapa de probabilidades de clase por píxel.

5.2.3. Adaptación para la predicción de días de supervivencia

En esta sección se detallan las diferentes variantes de la cabeza de regresión desarrolladas como parte de este trabajo, cuyo objetivo es predecir una variable escalar a partir de las representaciones profundas extraídas por el modelo. Estas cabezas no forman parte del diseño original de la arquitectura LViT, sino que constituyen una extensión específica diseñada para abordar el problema de regresión planteado en este proyecto. En este contexto, se emplean técnicas de regresión, de modo que el modelo genera una predicción continua en lugar de una clasificación por categorías. Para ello, se han diseñado cabezas de regresión que se conectan a partir de una capa intermedia de la red neuronal, tal y como se puede observar en la figura 5.3, recibiendo como entrada las representaciones de alto nivel generadas por el modelo principal y, mediante una o varias capas adicionales, transforman esa información en una predicción escalar continua.

Una vez extraídas las características profundas a partir del codificador de la red LViT, se introduce un módulo común de *pooling* global, seguido por distintas variantes de cabezas de regresión diseñadas para proyectar estas representaciones al espacio escalar. Como paso previo común a las tres primeras variantes, se aplica una operación de *pooling* promedio global adaptativo sobre el mapa de características de salida del codificador (habitualmente $\mathbf{y}_4$ o x_5), seguido de un aplanamiento:

$$\mathbf{z} = \frac{1}{H \cdot W} \sum_{i=1}^{H} \sum_{j=1}^{W} \mathbf{x}_{:,i,j} \in \mathbb{R}^C, \tag{5.1}$$

$$\mathbf{z}_{\text{flat}} = \text{Flatten}(\mathbf{z}) \in \mathbb{R}^C. \tag{5.2}$$

Este vector condensado $\mathbf{z}_{\text{flat}}$ sirve como entrada para las distintas variantes de la cabeza de regresión, cada una de las cuales sigue una estrategia diferente para transformar las representaciones extraídas en una predicción escalar de los días de supervivencia. A continuación se describen en detalle las cuatro variantes evaluadas.

La primera variante está formada por una capa lineal directa, que es la arquitectura más simple. Es una opción de baja complejidad computacional, pero con capacidad de

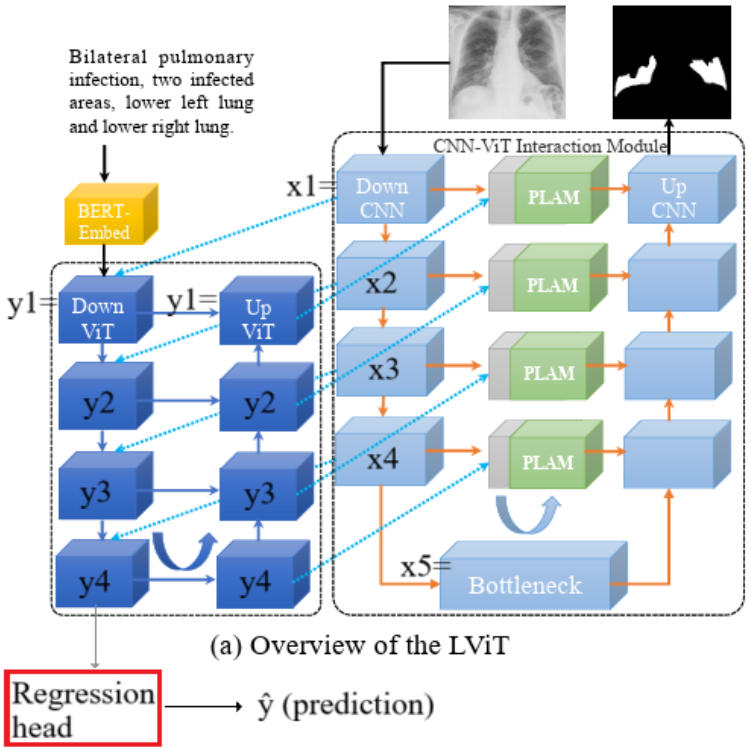


Figura 5.3: Esquema general del modelo LViT. Fuente: [Wang et al., 2022], con modificaciones personalizadas.

representación limitada. Consiste en conectar directamente el vector $\mathbf{z}_{\text{flat}}$ a una única capa lineal:

$$\hat{y} = \mathbf{w}^\top \mathbf{z}_{\text{flat}} + b \quad \text{con} \quad \mathbf{w} \in \mathbb{R}^C, \quad b \in \mathbb{R}. \tag{5.3}$$

En la segunda variante, se introduce una capa oculta (capa intermedia) completamente conectada con activación ReLU, lo que permite que pueda aprender funciones de regresión más complejas y no lineales. Por ello, este tipo de arquitectura mejora el poder de representación sin llegar a incrementar demasiado el coste:

$$\mathbf{h} = \text{ReLU}(\mathbf{W}_1 \mathbf{z} + \mathbf{b}_1), \quad \mathbf{W}_1 \in \mathbb{R}^{128 \times C}, \tag{5.4}$$

$$\hat{y} = \mathbf{w}_2^\top \mathbf{h} + b_2, \quad \mathbf{w}_2 \in \mathbb{R}^{128}. \tag{5.5}$$

Para la tercera variante, partiendo de la anterior, se añade una operación de *Dropout* al 50 % entre capas para reducir el sobreajuste, lo que fuerza a la red a no depender exclusivamente de ciertas activaciones durante el entrenamiento y mejora la generalización:

$$\mathbf{h} = \text{Dropout}(\text{ReLU}(\mathbf{W}_1 \mathbf{z} + \mathbf{b}_1)), \tag{5.6}$$

$$\hat{y} = \mathbf{w}_2^\top \mathbf{h} + b_2. \tag{5.7}$$

En la cuarta variante, se emplea una arquitectura más profunda, con el objetivo de capturar relaciones no lineales complejas entre las representaciones latentes y la variable objetivo (días de supervivencia), equilibrando capacidad y regularización.

La entrada proviene del bloque y4 del codificador ViT, tal y como se había nombrado antes, con forma $[B, 196, 512]$, que se reduce mediante media sobre los 196 tokens para obtener un vector de 512 dimensiones por muestra. Esta representación, luego se normaliza con LayerNorm antes de ser procesada.

La red consta de cuatro capas densas secuenciales de tamaño $512 \to 256 \to 128 \to 64$, cada una seguida de activación ReLU. Se añaden capas *Dropout* (30 % y 20 %) tras las dos primeras capas para prevenir sobreajuste. La última capa lineal genera la predicción escalar final. La formulación de la arquitectura es:

$$\mathbf{h}_0 = \text{LayerNorm}(\mathbf{z}), \tag{5.8}$$
$$\mathbf{h}_1 = \text{ReLU}(\mathbf{W}_1\mathbf{h}_0 + \mathbf{b}_1), \quad \mathbf{W}_1 \in \mathbb{R}^{256\times512} \tag{5.9}$$
$$\mathbf{h}'_1 = \text{Dropout}(\mathbf{h}_1, p = 0{,}3), \tag{5.10}$$
$$\mathbf{h}_2 = \text{ReLU}(\mathbf{W}_2\mathbf{h}'_1 + \mathbf{b}_2), \quad \mathbf{W}_2 \in \mathbb{R}^{128\times256} \tag{5.11}$$
$$\mathbf{h}'_2 = \text{Dropout}(\mathbf{h}_2, p = 0{,}2), \tag{5.12}$$
$$\mathbf{h}_3 = \text{ReLU}(\mathbf{W}_3\mathbf{h}'_2 + \mathbf{b}_3), \quad \mathbf{W}_3 \in \mathbb{R}^{64\times128} \tag{5.13}$$
$$\hat{y} = \mathbf{w}_4^\top \mathbf{h}_3 + b_4, \quad \mathbf{w}_4 \in \mathbb{R}^{64}. \tag{5.14}$$

5.3. Red neuronal SegResNet derivada de la red 3D U-Net

En este contexto, se presenta una versión modificada de la red SegResNet, diseñada como una extensión de la arquitectura 3D U-Net, ampliamente reconocida en tareas de segmentación volumétrica. Esta versión introduce bloques residuales como componente estructural clave en lugar de los bloques de convolución estándar. Los bloques residuales no solo permiten mejorar el flujo del gradiente en redes profundas, sino que también estabilizan el entrenamiento y facilitan la reutilización de características a través de conexiones de atajo.

A diferencia del diseño original de SegResNet en MONAI, que incluye una rama adicional tipo VAE usada como regularizador durante el entrenamiento, esta implementación está optimizada para entornos prácticos, prescindiendo de dicha rama para centrarse exclusivamente en la tarea de segmentación. Esta decisión reduce significativamente la complejidad computacional y acelera la inferencia, haciendo la red más adecuada para aplicaciones clínicas en tiempo real.

5.3.1. Arquitectura de la red neuronal

La arquitectura de SegResNet se basa en una estructura simétrica de tipo codificador-decodificador, heredada de la 3D U-Net, pero incorpora bloques residuales 3D como elemento principal en lugar de los bloques de convolución convencionales [Siddiquee et al., 2022, Zhang et al., 2021a]. Estos bloques residuales, inspirados en la filosofía de las ResNet, están formados por varias capas de convolución 3D (normalmente con kernel $3 \times 3 \times 3$), activaciones no lineales (como `LeakyReLU` o `PReLU`), y normalización, generalmente por lotes o por instancia, junto con conexiones de atajo que suman la entrada del bloque con su salida para facilitar el aprendizaje profundo y la estabilidad del entrenamiento.

La red está organizada en tres partes principales: un codificador, un cuello o puente y un decodificador. El codificador se encarga de extraer y comprimir la información relevante de la imagen de entrada, el cuello opera en la resolución espacial más baja y mayor profundidad semántica, y el decodificador reconstruye la segmentación a la resolución original, integrando tanto el contexto global como los detalles locales a través de conexiones de salto. Finalmente, una capa de convolución $1 \times 1 \times 1$ actúa como clasificador voxel a voxel. La ausencia de ramas auxiliares, como el VAE, permite que esta arquitectura mantenga un diseño más sencillo y eficiente, orientado a la segmentación directa y rápida en entornos clínicos.

5.3.2. Estructura general

El funcionamiento de la red SegResNet comienza con la entrada de un volumen 3D multimodal, compuesto por varias modalidades de resonancia magnética apiladas en el canal de entrada, con dimensiones típicas $(4, D, H, W)$. Este volumen es procesado inicialmente por una capa convolucional superficial que extrae las primeras características relevantes, sirviendo de base para el procesamiento posterior.

A continuación, el volumen atraviesa el codificador, una secuencia de bloques residuales. Cada bloque residual está formado por dos convoluciones 3D, normalización por lotes y activaciones `LeakyReLU`, junto con una conexión de atajo que suma la entrada y la salida del bloque. Matemáticamente, la salida de un bloque residual se expresa como:

$$\text{Output} = \text{LeakyReLU}(F(x) + x),$$

donde:

$$F(x) = \text{BN}(\text{Conv}_{3\times3\times3}(\text{LeakyReLU}(\text{BN}(\text{Conv}_{3\times3\times3}(x))))).$$

Si la dimensión de x no coincide con la de $F(x)$, el atajo se ajusta mediante una convolución $1 \times 1 \times 1$:

$$x' = \text{BN}(\text{Conv}_{1\times1\times1}(x)).$$

Tras cada bloque residual, se aplica un `MaxPooling3D` con stride $2 \times 2 \times 2$, lo que reduce la resolución espacial y permite al modelo captar patrones jerárquicos. El número de filtros se duplica en cada etapa, siguiendo la progresión $32 \to 64 \to 128 \to 256$.

En el cuello de la red, conocido como bridge o bottleneck, el volumen alcanza su menor resolución espacial y máxima profundidad semántica. Aquí se aplican dos bloques residuales consecutivos con 256 filtros, lo que permite capturar patrones globales y relaciones complejas en el volumen.

El decodificador reconstruye la segmentación espacial mediante operaciones de `UpSampling3D` con factor $2 \times 2 \times 2$. En cada etapa, la salida upsampleada se concatena con la activación correspondiente del codificador (conexiones de salto), lo que permite recuperar detalles espaciales que podrían haberse perdido durante el *downsampling*:

$$x_d = \text{Concat}(\text{UpSampling3D}(x), x_{\text{skip}}),$$

$$x'_d = \text{ResidualBlock}(x_d).$$

Cada bloque del decodificador también es residual, manteniendo la eficiencia y estabilidad del aprendizaje en toda la red.

Finalmente, una convolución $1 \times 1 \times 1$ actúa como capa clasificadora vóxel a vóxel, reduciendo la dimensión de canal al número de clases objetivo. Para convertir los logits en probabilidades, se aplica una función softmax:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}},$$

donde z_i es la activación del canal i en cada vóxel y K es el número total de clases.

Las operaciones matemáticas clave que sustentan la arquitectura incluyen la convolución 3D, que puede expresarse como:

$$Y(i,j,k) = \sum_{u=0}^{d-1} \sum_{v=0}^{h-1} \sum_{w=0}^{w-1} X(i+u, j+v, k+w) \cdot K(u,v,w),$$

y la normalización por lotes, que estabiliza el entrenamiento y se define como:

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}},$$

$$y_i = \gamma \hat{x}_i + \beta.$$

La activación `LeakyReLU` utilizada en los bloques residuales se define como:

$$\text{LeakyReLU}(x) = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases}$$

donde α suele ser 0.2.

5.4. Red neuronal 3D AutoEncoder

En el contexto de la segmentación y análisis de imágenes médicas 3D, los autoencoders 3D han demostrado ser herramientas eficaces para aprender representaciones compactas y significativas de los datos volumétricos. En este trabajo, se empleó una red 3D AutoEncoder como extractor de características latentes de las imágenes de resonancia magnética multimodal. Específicamente, se utilizó la salida final del codificador, es decir, el vector latente resultante tras la compresión del volumen de entrada, como un descriptor compacto de cada imagen.

Estos vectores latentes sirvieron como entrada para modelos de regresión orientados a predecir variables clínicas relevantes, como la supervivencia del paciente. Además, se combinaron con estadísticas voxel extraídas de las segmentaciones automáticas para enriquecer el conjunto de predictores y mejorar el rendimiento de los modelos de regresión.

5.4.1. Arquitectura de la red neuronal

El 3D AutoEncoder está diseñado como una red simétrica formada por dos bloques principales, reflejado en la figura 5.4: un codificador y un decodificador. El codificador se encarga de transformar un volumen 3D de entrada, que contiene múltiples modalidades de imagen (por ejemplo, T1, T1c, T2 y FLAIR), en una representación latente de baja dimensión [Wang et al., 2024, Rahman et al., 2022, Reve, 2024]. Para ello, el

codificador está compuesto por una secuencia de capas de convolución 3D, que incrementan progresivamente el número de canales y permiten capturar patrones espaciales jerárquicos. Estas capas están acompañadas de operaciones de `MaxPooling3D`, que reducen la resolución espacial y facilitan la extracción de características a diferentes escalas [Wang et al., 2024, Rahman et al., 2022]. Al final del bloque codificador, el volumen comprimido se aplana y pasa por una capa lineal, que genera un vector latente de tamaño fijo (512 dimensiones), actuando como un descriptor compacto y abstracto de la imagen original.

El decodificador, estructuralmente simétrico al codificador, está diseñado para revertir el proceso de compresión y reconstruir el volumen original a partir del vector latente [Wang et al., 2024, Rahman et al., 2022, Reve, 2024]. Comienza con una capa lineal que proyecta el vector latente de vuelta a una forma volumétrica. A continuación, emplea una serie de operaciones de `MaxUnpool3D`, que restauran las dimensiones espaciales originales utilizando los índices de pooling almacenados durante la codificación. Finalmente, el decodificador incorpora varias capas de convolución transpuesta 3D (`ConvTranspose3d`), que reducen el número de canales y refinan la reconstrucción espacial hasta alcanzar la forma y el número de canales de la imagen de entrada.

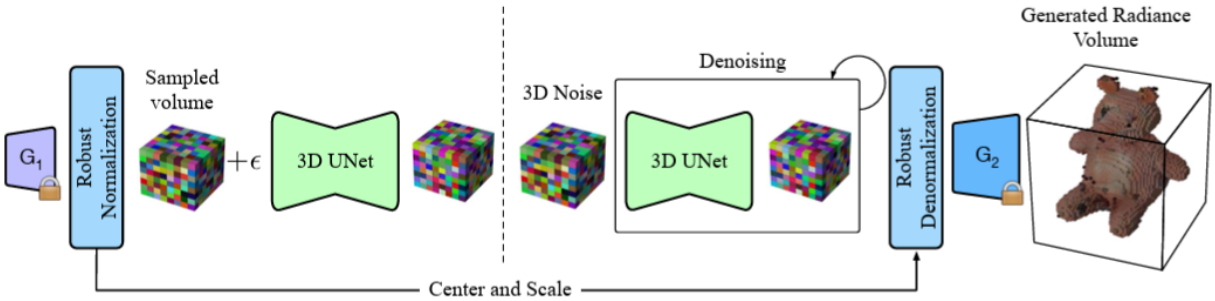


Figura 5.4: Diagrama de la arquitectura de 3D AutoEncoder. Fuente: [Wang et al., 2023].

5.4.2. Estructura general

El proceso general de la red comienza con la entrada de un volumen 3D multimodal, que atraviesa sucesivamente las capas de convolución 3D y pooling del codificador. Cada capa de convolución realiza la operación:

$$y_{i,j,k}^{(n)} = \sum_{c=1}^{C_{\text{in}}} \sum_{u,v,w} x_{i+u,j+v,k+w}^{(c)} \cdot w_{u,v,w}^{(c,n)} + b_n,$$

donde $x_{i+u,j+v,k+w}^{(c)}$ es el valor del voxel de entrada en el canal c, $w_{u,v,w}^{(c,n)}$ es el peso del filtro n en el canal c, y b_n es el sesgo correspondiente. Tras cada convolución, se aplica una operación de `MaxPooling3D`, que selecciona el valor máximo en cada subbloque espacial, reduciendo la resolución y guardando los índices de los máximos para su uso posterior en el decodificador.

Después de la última operación de pooling, el volumen comprimido se aplana y se transforma en un vector latente de 512 dimensiones mediante una capa lineal:

$$z = Wx + b,$$

donde x es el vector aplanado, W la matriz de pesos y b el sesgo. Este vector latente constituye la representación comprimida y abstracta de la imagen, que puede ser utilizada tanto para la reconstrucción como para tareas posteriores, como la predicción de variables clínicas a través de modelos de regresión.

En la fase de decodificación, el vector latente se reproyecta a un volumen 3D mediante una capa lineal inversa:

$$x = W'z + b'.$$

A continuación, se aplican operaciones de `MaxUnpool3D` utilizando los índices guardados durante el pooling, restaurando la estructura espacial original del volumen. Finalmente, se emplean tres capas de convolución transpuesta 3D (`ConvTranspose3d`), que realizan la operación inversa de la convolución tradicional, expandiendo espacialmente el tensor:

$$y = W^\top x + b.$$

Estas capas refinan la reconstrucción y restauran la resolución y el número de canales iniciales. El objetivo es que el volumen reconstruido sea lo más fiel posible al volumen de entrada original, minimizando la pérdida de información durante la compresión y descompresión.

5.5. Modelado basado en representaciones latentes

En esta sección se describen los modelos de regresión empleados para predecir datos clínicos relevantes, como el número de días de supervivencia de un paciente. La estrategia se basa en utilizar las características latentes extraídas por los codificadores de redes neuronales profundas, ya sean orientadas a segmentación o reconstrucción de imágenes, que condensan la información significativa en vectores manejables.

Como se ilustra en la figura 5.5, el proceso inicia con la extracción de estas características mediante un codificador, como un 3D Autoencoder o un LViT. Estas representaciones, junto con estadísticas voxel de la segmentación en algunos casos, se emplean como entrada para los modelos de regresión, permitiendo aprovechar los patrones aprendidos por las redes neuronales y combinarlos con la robustez de los métodos clásicos, mejorando así la precisión y adaptabilidad a nuevos casos.

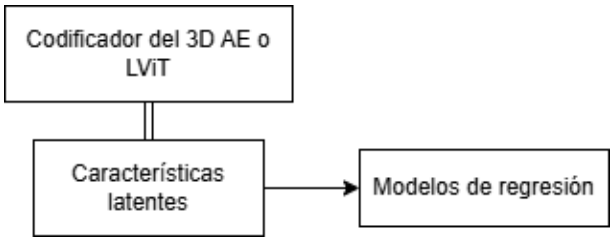


Figura 5.5: Esquema del flujo de información, en donde, a partir del codificador de la red neuronal 3D AE o el de la LViT, se extraen las características latentes de las imágenes de entrada (y del texto en el caso de la red neuronal LViT), con lo que, posteriormente, se entrenan los modelos de regresión clásicos.

5.5.1. Gradient boosting

El Gradient Boosting es un algoritmo de ensamble iterativo que construye nuevos aprendices débiles en cada paso con el objetivo de reducir la función de pérdida. Este modelo utiliza una estrategia de *boosting* donde se entrenan múltiples árboles de decisión secuencialmente, donde cada nuevo árbol se ajusta para corregir los errores del conjunto anterior. El proceso comienza con una predicción inicial, típicamente la media de los valores objetivo, y luego calcula los pseudo-residuos entre los valores observados y las predicciones actuales.

Matemáticamente, el algoritmo construye el modelo de forma aditiva:

$$F_m(x) = F_{m-1}(x) + \eta h_m(x),$$

donde η es la tasa de aprendizaje y $h_m(x)$ es el nuevo árbol que se ajusta a los residuos del modelo anterior. El proceso minimiza una función de pérdida utilizando el descenso por gradiente en el espacio de funciones, típicamente empleando el error cuadrático medio para tareas de regresión.

5.5.2. Regresor perceptrón multicapa (MLP)

El MLPRegressor implementa una red neuronal feedforward multicapa para tareas de regresión. Este modelo utiliza múltiples capas ocultas con neuronas interconectadas que pueden capturar relaciones complejas y no lineales entre las variables de entrada y salida. La arquitectura se basa en la propagación hacia adelante de la información a través de las capas, donde cada neurona aplica una transformación lineal seguida de una función de activación no lineal.

Para una red con una capa oculta, la función aprendida se expresa como:

$$f(x) = W_2 g(W_1^T x + b_1) + b_2,$$

donde W_1 y W_2 representan los pesos de la capa de entrada y oculta respectivamente, b_1 y b_2 son los sesgos, y $g(\cdot)$ es la función de activación. La arquitectura típica incluye capas ocultas con funciones de activación como ReLU, y utiliza algoritmos de optimización como Adam para el entrenamiento.

5.5.3. Regresión random forest

La regresión Random Forest es un método de aprendizaje automático ensamblado que utiliza una colección de árboles de decisión para hacer predicciones. Cada árbol en el bosque predice un valor, y la predicción final es el promedio de todas las predicciones de los árboles. Este enfoque ayuda a reducir el sobreajuste, aumenta la precisión y mejora la robustez en comparación con un solo árbol de decisión.

El algoritmo funciona mediante tres mecanismos principales: primero, el muestreo con reemplazo, donde se seleccionan subconjuntos aleatorios de datos (con reemplazo) para entrenar cada árbol; segundo, la selección aleatoria de características, en la cual en cada nodo se considera un subconjunto aleatorio de características para la división; y finalmente, la agregación de predicciones, donde para tareas de regresión las predicciones de todos los árboles individuales se promedian para obtener la predicción final.

La predicción final se expresa como:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^{T} h_t(x),$$

donde T es el número de árboles y $h_t(x)$ es la predicción del árbol t.

5.5.4. Regresión ridge

La regresión Ridge es un método de regularización que aplica penalización L2 para prevenir el sobreajuste mediante la penalización de coeficientes grandes. Este enfoque es particularmente útil cuando se trata de multicolinealidad en los datos, ya que reduce los errores estándar de las estimaciones y mejora la capacidad de generalización del modelo.

La regresión Ridge estándar aplica regularización L2 directamente sobre las características originales sin expansión polinómica. Este modelo es efectivo para problemas con relaciones lineales entre variables. La función objetivo a minimizar es:

$$\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2,$$

donde λ es el parámetro de regularización que controla la magnitud de la penalización aplicada a los coeficientes.

La regresión Ridge con características polinómicas combina la expansión polinómica de las variables originales con regularización L2. Esta aproximación permite capturar relaciones no lineales al expandir las variables con términos polinómicos como productos y potencias, mientras que Ridge aplica regularización para prevenir el sobreajuste. El término de regularización penaliza los coeficientes grandes, reduciendo así la complejidad del modelo. El objetivo es minimizar la función:

$$\min_{w} \|y - Xw\|^2 + \alpha \|w\|^2,$$

donde α controla el grado de regularización y X representa el espacio de características expandido con términos polinómicos. El parámetro α determina el equilibrio entre el ajuste a los datos y la simplicidad del modelo.

5.5.5. Regresión de vectores de soporte

SVR aplica el principio de las máquinas de vectores soporte adaptado para tareas de regresión. El modelo busca una función que prediga valores reales dentro de un margen de tolerancia ε, utilizando la función de pérdida ε-insensible:

$$|\xi|_\varepsilon = \begin{cases} 0 & \text{si } |\xi| \leq \varepsilon \\ |\xi| - \varepsilon & \text{en caso contrario} \end{cases}$$

La SVR lineal utiliza una formulación sin transformación no lineal del núcleo. Este modelo es apropiado cuando las relaciones entre variables son aproximadamente lineales y se busca un modelo más interpretable y computacionalmente eficiente. La SVR lineal busca encontrar un hiperplano que mejor se ajuste a los datos dentro de un margen de tolerancia especificado. La función objetivo a optimizar es:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2}||w||^2 + C \sum_{i=1}^{n} (\xi_i + \xi_i^*),$$

sujeto a las restricciones de que las predicciones estén dentro del margen ε o sean penalizadas por las variables de holgura ξ_i y ξ_i^*.

SVR con *kernel* RBF permite modelar relaciones no lineales transformando el espacio de características mediante una función gaussiana. Este kernel mide la similitud entre puntos basándose en su distancia euclidiana, donde puntos cercanos tienen mayor similitud. El kernel RBF se define como:

$$K(x, y) = \exp(-\gamma||x - y||^2),$$

donde γ determina el ancho del kernel.

5.5.6. Categorical boosting

Categorical Boosting es un algoritmo de Gradient Boosting desarrollado específicamente para manejar características categóricas de forma nativa. Utiliza técnicas avanzadas de *boosting* con árboles de decisión, incorporando métodos especiales para el procesamiento de variables categóricas sin necesidad de codificación manual previa. El algoritmo implementa técnicas como el ordenamiento objetivo y la codificación de características categóricas basada en estadísticas de destino.

La función objetivo que minimiza CatBoost incluye términos de regularización específicos:

$$\min_F \sum_{i=1}^{n} L(y_i, F(x_i)) + \Omega(F),$$

donde L es la función de pérdida, F es el modelo ensamblado y $\Omega(F)$ representa los términos de regularización que incluyen penalizaciones específicas para el manejo de características categóricas y prevención del sobreajuste.

5.6. Otras redes neuronales empleadas

La selección de arquitecturas alternativas se fundamentó en la literatura científica y en la necesidad de comparar diferentes paradigmas de aprendizaje automático, considerando la naturaleza 3D de las imágenes médicas, la complejidad tumoral y los requisitos computacionales. Sin embargo, no todas las redes evaluadas alcanzaron el rendimiento esperado. Factores como la complejidad del modelo, la disponibilidad de datos, los recursos computacionales y la especificidad del problema influyeron notablemente en los resultados. Algunas arquitecturas presentaron problemas de convergencia, sobreajuste o incapacidad para capturar adecuadamente las características relevantes, lo que se analiza en esta sección junto a las lecciones aprendidas durante su implementación.

5.6.1. Red neuronal UNETR

En los últimos años, los avances en redes neuronales profundas han transformado la segmentación médica, especialmente en imágenes 3D. Sin embargo, los modelos puramente convolucionales tienen dificultades para captar relaciones globales en datos volumétricos. Para superar estas limitaciones, surge UNETR, una arquitectura híbrida que combina la estructura en 'U' de U-Net con la capacidad de los transformers para modelar contexto global, aprovechando tanto la eficiencia de las CNN en detalles locales como la potencia de los transformers en relaciones de largo alcance.

UNETR emplea un codificador compuesto solo por bloques transformer, que aprenden representaciones locales y globales del volumen de entrada mediante mecanismos de

autoatención multi-cabeza y capas feed-forward. Extrae estados intermedios en distintas profundidades, los reorganiza como volúmenes 3D y los conecta al decodificador mediante conexiones de salto, lo que permite fusionar información global y detalles espaciales finos.

El proceso comienza dividiendo el volumen de entrada en parches cúbicos, que se aplanan y proyectan a *embeddings* con codificaciones posicionales 3D. Estos se procesan en el codificador transformer, calculando la atención multi-cabeza como:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V},$$

donde **Q**, **K** y **V** provienen de los *embeddings* de entrada. Los estados intermedios se remodelan como volúmenes 3D y, junto con el decodificador basado en *upsampling* y convoluciones 3D, permiten recuperar la resolución espacial y refinar la segmentación. La figura 5.6 muestra el flujo de datos y la estructura general de UNETR.

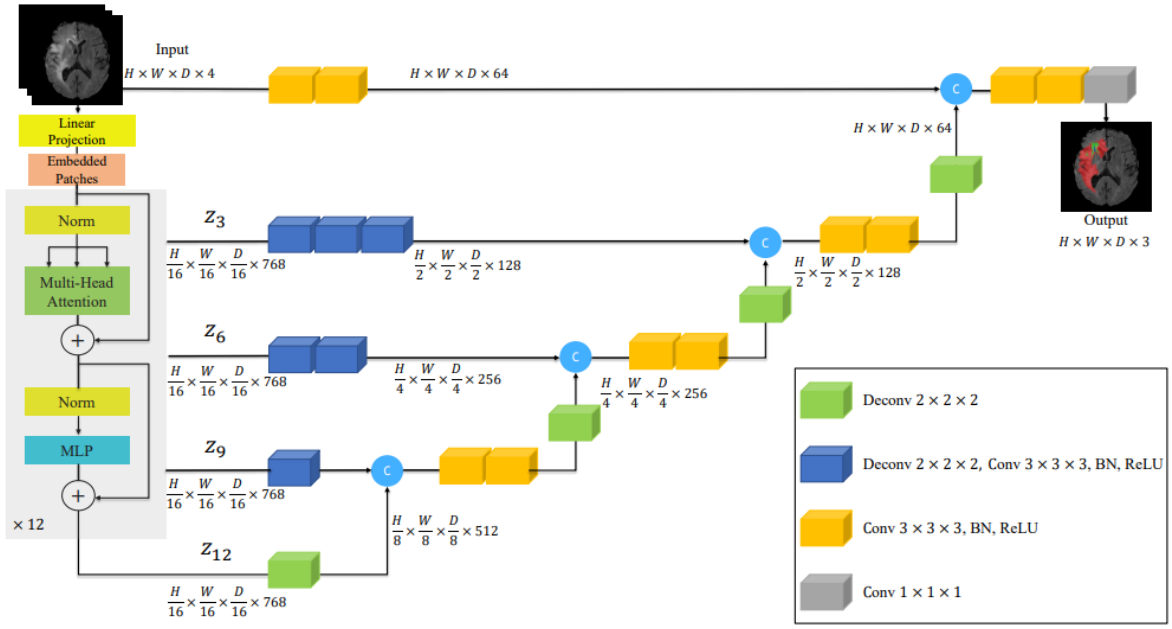


Figura 5.6: Diagrama de la arquitectura de UNETR (UNEt TRansformers). Fuente: [Hatamizadeh et al., 2022a]

5.6.2. Red neuronal SAM2

La segmentación automática ha avanzado notablemente con modelos fundacionales como SAM, desarrollado por Meta AI, que permite segmentar cualquier objeto en una imagen a partir de un clic, caja o máscara. El reto de mantener coherencia temporal y precisión en vídeo llevó al desarrollo de SAM 2 en 2024, que introduce un mecanismo de memoria para seguir objetos a lo largo de los fotogramas, reemplaza el codificador ViT por Hiera (más eficiente y escalable) y optimiza el flujo para aplicaciones en tiempo real [Meta AI, 2025, Ultralytics, 2025, Labellerr, 2024]. SAM 2.1 perfecciona aún más la precisión y robustez en escenas complejas, consolidando a SAM como referente en segmentación universal de imágenes y vídeos.

SAM 2 se basa en una arquitectura modular y jerárquica, cuyo núcleo es el codificador Hiera, capaz de extraer representaciones multiescala y profundas de cada fotograma. El

codificador de prompts transforma las entradas del usuario en vectores guía, mientras que el novedoso sistema de memoria, formado por un codificador, un banco y un módulo de atención de memoria, almacena y actualiza información sobre los objetos y las interacciones a lo largo de los fotogramas, manteniendo coherencia temporal y gestionando oclusiones o reapariciones. El decodificador de máscaras integra toda esta información mediante atención cruzada, generando segmentaciones precisas y refinadas tanto para objetos completos como para sus partes, como se ilustra en la figura 5.7.

El proceso de segmentación comienza con la extracción de características visuales por Hiera y la conversión de los prompts en vectores guía, integrados mediante atención cruzada. En vídeo, el sistema de memoria permite almacenar y recuperar información relevante sobre los objetos, asegurando segmentaciones consistentes incluso ante oclusiones o cambios de apariencia. El decodificador produce máscaras jerárquicas, un *Object Pointer*, un IoU Score y un *Occlusion Score*, permitiendo segmentación en tiempo real y adaptabilidad a escenas complejas. SAM 2 soporta tanto segmentación automática como guiada por prompts, facilitando aplicaciones desde edición de vídeo hasta investigación biomédica [Meta AI, 2025, Ultralytics, 2025].

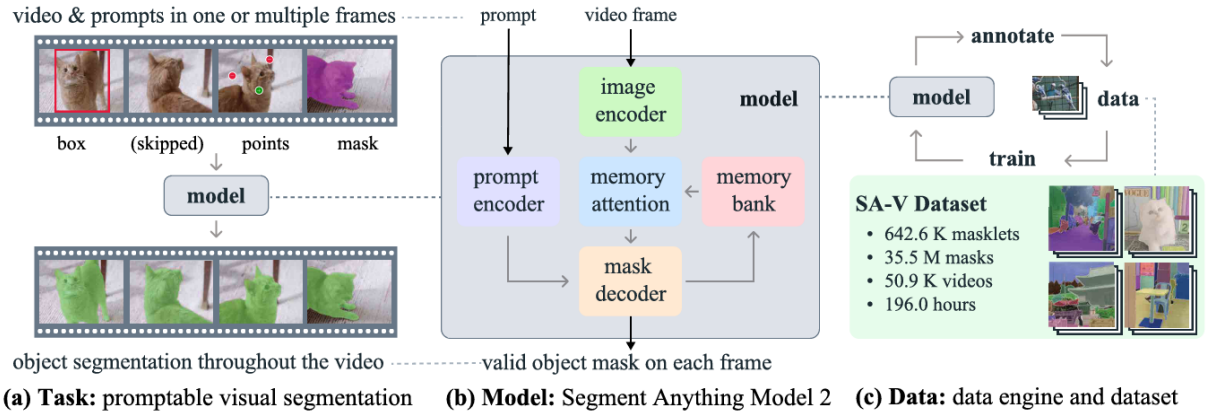


Figura 5.7: Diagrama de la arquitectura de SAM2. Fuente: [Zhang et al., 2024].

Capítulo 6

Configuración de experimentos

Este capítulo describe cómo se estructuraron y organizaron los diferentes experimentos realizados durante el desarrollo del proyecto. La configuración experimental es un aspecto clave para asegurar la validez de los resultados obtenidos, especialmente en estudios que combinan tareas de segmentación, extracción de características y predicción de variables clínicas a partir de imágenes médicas.

En este contexto, se han utilizado distintas arquitecturas de redes neuronales, tanto unimodales como multimodales, asignando a cada una un conjunto específico de tareas según sus capacidades. Algunas redes se emplearon únicamente para segmentación, otras se utilizaron para extraer representaciones latentes que fueron reutilizadas en tareas de regresión, y algunas arquitecturas fueron diseñadas directamente con una cabeza de regresión integrada. Esta variedad metodológica permitió comparar enfoques clásicos y modernos, así como evaluar el impacto de las representaciones visuales en el rendimiento predictivo.

La Tabla 6.1 resume la asignación de tareas realizada para cada modelo empleado, clasificando sus usos principales en función de si fueron utilizados para segmentar, extraer características o realizar regresión sobre variables como la supervivencia del paciente.

Tabla 6.1: Asignación de tareas por modelo utilizado

Modelo	**Segmentación**	**Extracción de características**	**Regresión**
3D U-Net	✓	–	–
UNETR	✓	–	–
SegResNet	✓	–	–
LViT	✓	✓	–
LViT + cabeza de regresión	✓	–	✓
Autoencoder 3D	–	✓	–
SAM2	✓	–	–

6.1. Estrategia de división de datos

Para asegurar una evaluación justa y evitar sobreajuste, se ha diseñado una estrategia de particionado consistente para dividir los datos en conjuntos de entrenamiento, validación y prueba.

El particionado se llevó a cabo respetando la integridad de los pacientes, asegurando que ningún paciente aparezca en más de un subconjunto, evitando así la fuga de información. Además, se intentó mantener una proporción equilibrada entre clases (cuando

correspondía) y un tamaño representativo para los conjuntos de validación y test, garantizando la posibilidad de evaluar de forma robusta tanto el ajuste durante el entrenamiento como el rendimiento general del modelo.

Para los modelos de segmentación, se filtraron inicialmente los pacientes para incluir solo aquellos con la columna de Edad no nula, lo que garantizaba también la disponibilidad de los Días de Supervivencia. Por esta razón, no se imputaron datos faltantes en ninguna de las redes de segmentación. A continuación, cada paciente fue asignado a un grupo de edad según su década, agrupando, por ejemplo, las edades de 30 a 39 bajo el valor 30. Para asegurar una distribución equilibrada y evitar sesgos, se aplicó una validación cruzada estratificada (Stratified K-Fold) con k=7 sobre estos grupos de edad, y la asignación obtenida se almacenó en un archivo csv para su uso posterior. En este esquema, los pacientes con un valor de subconjunto distinto de 0 se destinaron al entrenamiento, mientras que los de subconjunto igual a 0 se reservaron para validación. Los pacientes sin edad conocida, y por tanto ausentes en el csv, se emplearon como conjunto de test.

Para los modelos de regresión, la división del conjunto de datos se llevó a cabo utilizando la función `train_test_split()`, asignando un 80 % de los datos al conjunto de entrenamiento y distribuyendo el 20 % restante a partes iguales entre los conjuntos de validación y test, es decir, un 10 % para cada uno. Esta estrategia permitió disponer de suficientes datos para entrenar los modelos, validar su rendimiento durante el ajuste de hiperparámetros y evaluar su capacidad de generalización en un conjunto independiente.

6.2. Preparación de datos por tipo de modelo

Una de las etapas más críticas en el desarrollo de modelos de aprendizaje profundo es la preparación de los datos. Dependiendo del tipo de tarea se requieren preprocesamientos específicos que garanticen la compatibilidad con las arquitecturas utilizadas y favorezcan un aprendizaje efectivo. Esta sección detalla los pasos realizados para adaptar el conjunto de datos BraTS2020 y otros recursos complementarios según las necesidades de cada modelo.

En concreto, se exponen las transformaciones (normalización, recortes, etc.) aplicadas a las imágenes volumétricas en formato NIfTI, el tratamiento de las máscaras de segmentación, la extracción de representaciones latentes y la combinación con datos clínicos. Además, se abordará específicamente cómo se gestionó el problema de los valores faltantes en la parte de regresión, dada su importancia para la calidad y el ajuste de los modelos predictivos.

6.2.1. Preprocesamiento de datos para modelos de segmentación

Para los modelos de segmentación se aplicaron estrategias de preprocesamiento y adaptación específicas según la arquitectura de red neuronal utilizada. En la red 3D U-Net, cada paciente dispone de las cuatro modalidades de imagen (Flair, T1, T1ce, T2) en archivos .nii tridimensionales, que se cargan y normalizan al rango de forma independiente. Estas modalidades se apilan formando un tensor multicanal de dimensiones $(4, D, H, W)$. La segmentación, también en 3D, etiqueta cada vóxel como 1 (núcleo necrótico), 2 (edema) o 4 (tumor realzado), a partir de lo cual se generan tres máscaras binarias: WT (etiquetas 1, 2 y 4), TC (1 y 4) y ET (4), que constituyen la salida deseada del modelo en forma $(3, D, H, W)$.

En el caso de la red LViT, que solo admite imágenes 2D, el conjunto de datos se adaptó seleccionando la modalidad T2 y extrayendo el corte axial con mayor contenido tumoral (o el central si no hay tumor). Este slice se normaliza y redimensiona a 224x224 píxeles. Para la segmentación, el mismo corte se convierte en una máscara binaria (0) y se escala a 255, considerando la máscara WT. La imagen se interpola por área y la segmentación por vecino más cercano. Los datos clínicos se extraen del archivo survival_info.csv, omitiendo los valores faltantes en la descripción.

Para SegResNet, que requiere un tamaño de entrada de (128, 128, 96), las imágenes de T1, T2, T1ce y Flair se normalizan al rango y se ajustan centralmente a las dimensiones requeridas mediante padding o crop según sea necesario. Las cuatro modalidades se agrupan en una matriz de forma (128, 128, 96, 4). La segmentación se ajusta del mismo modo, convirtiendo las etiquetas a 0, 1, 2, 3 (fondo, tumor no realzado, edema, tumor realzado) y aplicando codificación one-hot, obteniendo una máscara final de (128, 128, 95, 4). En el conjunto de test, se realiza un zoom out del 65 % y, si es necesario, crop centrado o padding para asegurar el tamaño final, permitiendo comparar la predicción de este modelo en igualdad de condiciones con los otros.

6.2.2. Preprocesamiento de datos para modelos de regresión

Para los modelos de regresión, se emplearon distintos procedimientos según la arquitectura y el tipo de información utilizada. En el caso de la red neuronal LViT con cabeza de regresión, el preprocesamiento de imágenes y la gestión de descripciones siguieron el mismo protocolo que en la LViT de segmentación. Las muestras que no disponían de valores esenciales, como la edad o el tipo de glioma, se descartaban de los conjuntos de entrenamiento, validación y prueba.

Por otro lado, se entrenó un autoencoder 3D para obtener representaciones latentes de volúmenes de resonancia magnética a partir de la última capa del codificador. El entrenamiento se realizó sobre el conjunto de datos de BraTS2020, utilizando particiones estratificadas por edad, y cada volumen 3D servía como entrada y objetivo, ya que el autoencoder busca reconstruir el propio volumen.

A partir de estas representaciones, se entrenaron modelos de regresión utilizando dos tipos de entrada: características latentes extraídas del autoencoder 3D, que contienen información visual de la IRM y se complementan con estadísticas vóxel (medias, varianzas, percentiles de intensidad) calculadas sobre las máscaras de segmentación; y características latentes generados por la última capa del bloque ViT del codificador LViT, que integran información visual y semántica derivada del texto clínico (edad, tipo de tumor, grado de resección, según la variable a predecir).

En el entrenamiento de modelos clásicos de regresión con estas características del 3D AutoEncoder y estadísticas vóxel, se buscaba predecir variables como los días de supervivencia. Las características latentes del autoencoder 3D actuaban como descriptores compactos de la imagen, mientras que las estadísticas vóxel resumían la distribución de intensidades y variaciones locales del volumen. Para cada modalidad (T1, T2, FLAIR, T1c), se extraían métricas globales y regionales, enriqueciendo la entrada a los modelos de regresión.

En el caso de los modelos de regresión con características latentes extraídas del bloque ViT (y4) de LViT, primero se obtenían las representaciones latentes fusionadas de imágenes y descripciones clínicas. Estas se condensaban mediante pooling promedio, generando un vector latente de 512 características por muestra, y se almacenaban junto

a las variables objetivo en archivos `csv`. Dependiendo de la variable clínica a predecir, la red LViT se entrenaba con diferentes descripciones para evitar fuga de información. Una vez generados estos conjuntos, se entrenaban modelos de regresión clásicos con las características latentes como entrada y variables clínicas como salida.

Para abordar el desbalance en la variable de días de supervivencia, especialmente la escasez de pacientes con supervivencias elevadas, se emplearon dos estrategias: entrenar modelos con los datos originales y aplicar una técnica de aumento de datos. Inicialmente, se probó imputar la media a los valores faltantes, pero esta aproximación modificaba la distribución original de las variables, provocando que los modelos solo predijeran un rango limitado de valores y generando sesgos en las predicciones, tal y como se puede observar en la figura 6.1. Por ello, se optó por una estrategia de aumento de datos, que consistió en añadir ruido gaussiano a las características latentes de los pacientes con supervivencias altas, tanto si esas características provenían del autoencoder 3D como del bloque y4 de la red LViT. Adicionalmente, se añadió ruido gaussiano a la salida del modelo para evitar que hubiesen diferentes valores predichos para un mismo paciente, es decir, evitar la aparición de líneas de puntos verticales en las gráficas de dispersión que indicarían predicciones idénticas para casos distintos, como se puede observar en la figura 6.2. El objetivo era generar nuevas muestras sintéticas que, aunque derivadas de los casos reales, presentaran pequeñas variaciones controladas. De este modo, se simula la presencia de más ejemplos reales en los extremos del rango de supervivencia, donde los datos originales son escasos [Stocksieker et al., 2023, Wen and Angryk, 2024, Science, 2025]. Gracias a este aumento de la diversidad en los datos, el modelo tuvo la oportunidad de aprender patrones más sólidos y generalizables en los casos de supervivencia elevada, lo que redujo el riesgo de sobreajuste y ayudó a que las predicciones sean más fiables y precisas incluso en los escenarios menos frecuentes. Así, se logra que el modelo no solo funcione bien en los casos más comunes, sino que también sea capaz de enfrentarse con garantías a situaciones poco habituales.

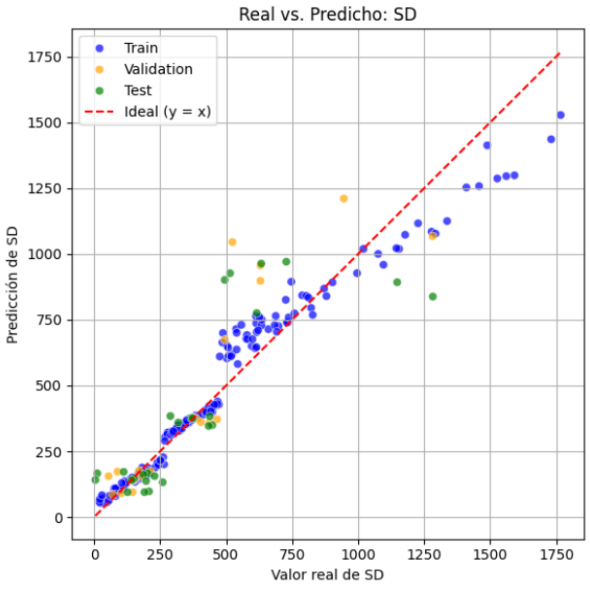
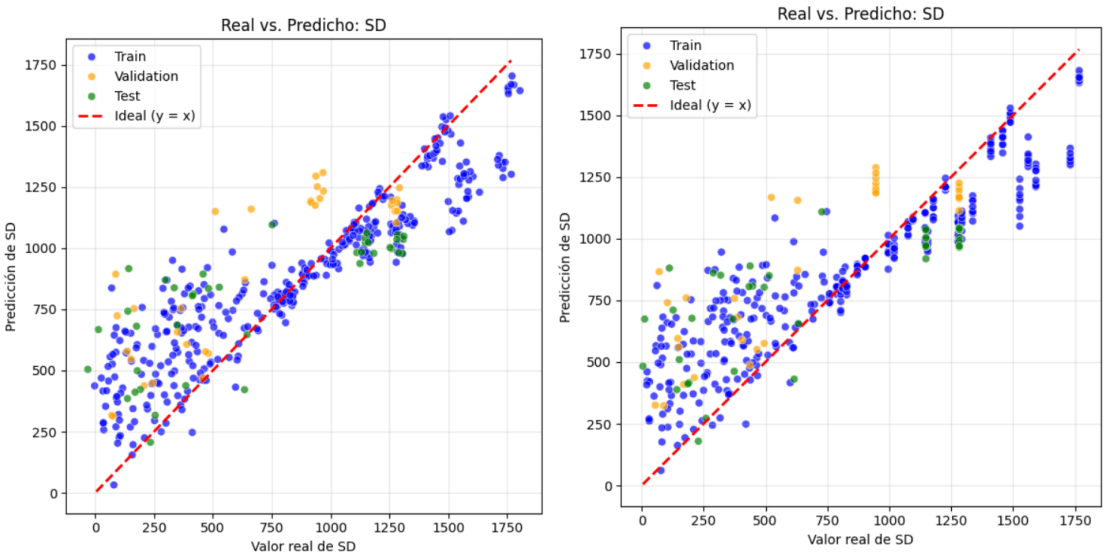


Figura 6.1: Regresión modelo Random Forest entrenado con datos imputándoles la media a los valores faltantes.



(a) (b)

Figura 6.2: Modelo SVR con kernel RBF sin aplicarle búsqueda de hiperparámetros, con aumento de datos. En la primera figura (a) se aplica ruido gaussiano a la salida, mientras que en la segunda (b) no se aplica ruido gaussiano a la salida.

6.3. Configuración del entrenamiento

Una vez preprocesados los datos y seleccionadas las arquitecturas, el siguiente paso clave fue configurar los experimentos de entrenamiento. Esta sección describe en detalle los aspectos técnicos del entrenamiento de los modelos, incluyendo las funciones de pérdida utilizadas, los optimizadores, el número de épocas, los criterios de parada temprana, y el seguimiento de métricas durante las distintas fases.

Dado que se abordaron tanto tareas de segmentación como de regresión, las configuraciones fueron adaptadas a cada tipo de modelo y objetivo. Los modelos de segmentación requirieron configuraciones específicas para manejar el desequilibrio entre clases, mientras que los modelos de regresión necesitaron ajustes particulares para predecir valores continuos de supervivencia. Además, los modelos multimodales y multitarea presentaron desafíos adicionales en términos de balanceo entre las diferentes funciones de pérdida y la sincronización del aprendizaje entre tareas.

6.3.1. Funciones de pérdida

Las funciones de pérdida utilizadas variaron según el modelo y el objetivo del entrenamiento. Para los modelos de segmentación, se emplearon combinaciones de funciones tradicionales como Binary Cross Entropy (BCE) y métricas específicas como Dice Loss, tal y como se resume en la Tabla 6.2.

Las funciones de entropía cruzada son fundamentales en aprendizaje automático y se adaptan a distintos problemas de clasificación. La BCE se usa en clasificación binaria y multi-label, penalizando las diferencias entre las probabilidades predichas y las etiquetas reales [Goodfellow et al., 2016, Murphy, 2012]:

$$\mathrm{BCE}(p, y) = -\left[y \cdot \log(p) + (1 - y) \cdot \log(1 - p)\right], \tag{6.1}$$

donde p es la probabilidad predicha (tras la sigmoide) y y la etiqueta real (0 o 1).

La variante BCEWithLogitsLoss combina la función sigmoide y BCE en una sola operación, mejorando la eficiencia y la estabilidad numérica durante el entrenamiento [Paszke et al., 2019]:

$$\mathcal{L}(x, y) = -\left[y \cdot \log(\sigma(x)) + (1 - y) \cdot \log(1 - \sigma(x))\right], \tag{6.2}$$

donde x es el logit del modelo, y la etiqueta real, y $\sigma(x)$ la función sigmoide.

La Categorical Cross Entropy (CCE) se utiliza en clasificación multi-clase con clases mutuamente excluyentes y suele combinarse con softmax, que asegura que las probabilidades sumen 1:

$$\text{CCE} = -\sum_{i=1}^{C} y_i \log(p_i), \tag{6.3}$$

donde C es el número de clases, y_i la etiqueta verdadera (one-hot) y p_i la probabilidad predicha para la clase i. BCE es adecuada para problemas multi-label y CCE para problemas multi-clase.

En cuanto a la Dice Loss, evalúa la superposición entre la máscara predicha y la real, siendo especialmente útil en segmentación médica con fuerte desbalance entre clases [Milletari et al., 2016b, Sudre et al., 2017]:

$$\text{Dice Loss} = 1 - \frac{2 \cdot |P \cap G| + \epsilon}{|P| + |G| + \epsilon}, \tag{6.4}$$

donde P es la máscara predicha, G la real, y ϵ un valor pequeño para evitar divisiones por cero [Milletari et al., 2016b].

En concreto, el modelo 3D U-Net, utilizó BCEWithLogitsLoss junto con DiceLoss para abordar el problema multi-label, permitiendo que las tres clases (ET, TC, WT) puedan solaparse en el mismo píxel. Este modelo devuelve logits sin activación final, por lo que requiere aplicar BCEWithLogitsLoss y, posteriormente, la función sigmoide para la evaluación y umbralización, generando tres máscaras binarias independientes.

El modelo LViT usa BCE normal, ya que integra la activación sigmoide en su arquitectura y devuelve probabilidades directamente. Aquí, la combinación DiceLoss + BCE se pondera al 50 % mediante WeightedDiceBCE para equilibrar ambos objetivos.

Por su parte, SegResNet aborda un problema multi-clase con cuatro clases excluyentes (fondo, WT, ET, TC), empleando DiceLoss combinada con Categorical Cross Entropy (CCE). Este modelo utiliza softmax como activación final, lo que produce una única máscara con clases exclusivas.

La combinación de Dice Loss con BCE o CCE aprovecha la complementariedad de enfoques: BCE y CCE aportan optimización pixel-wise con gradientes estables y clasificación local precisa, mientras que Dice Loss ofrece optimización regional, manejo efectivo del desequilibrio de clases y coherencia espacial. Esta sinergia estabiliza el entrenamiento inicial y optimiza el solapamiento cuando el modelo mejora, logrando mayor robustez y precisión en la segmentación final.

En las tareas de regresión de este proyecto, centradas principalmente en predecir los días de supervivencia a partir de representaciones latentes, se emplearon funciones de pérdida específicas adaptadas a la naturaleza continua de la variable objetivo. El MSE es una función de pérdida fundamental en regresión, ya que cuantifica el error promedio al elevar al cuadrado la diferencia entre los valores predichos y los reales [Goodfellow et al., 2016, Murphy, 2012]. Penaliza más fuertemente los errores grandes e incentiva al modelo a realizar predicciones cercanas a los valores verdaderos. Su formulación es:

Tabla 6.2: Resumen de funciones de pérdida utilizadas en cada modelo de segmentación.

Modelo	**Función de pérdida utilizada**	**Tipo de problema**
3D U-Net	BCEWithLogitsLoss + DiceLoss	Multi-label
LViT	DiceLoss + BCE	Multi-label
SegResNet	DiceLoss + CCE	Multi-clase

$$\mathcal{L}\text{MSE} = \frac{1}{N}\sum i = 1^{N}(x_i - \hat{x}_i)^2, \tag{6.5}$$

donde x_i es el valor real y $\hat{x}_i$ el valor predicho. El MSE se utilizó como función de pérdida en el entrenamiento del autoencoder, dada su simplicidad y eficacia para medir el desempeño en problemas de predicción continua [Goodfellow et al., 2016].

Por otro lado, el MAE Normalizado cuantifica el error promedio absoluto entre las predicciones y los valores reales, ajustando este error según la escala de los datos [Willmott and Matsuura, 2005, Chai and Draxler, 2014]. Esto permite comparar el desempeño del modelo en diferentes contextos, independientemente de la magnitud de las variables. Su expresión es:

$$\text{Score} = \frac{\sum |y - \hat{y}|}{\sum y}, \tag{6.6}$$

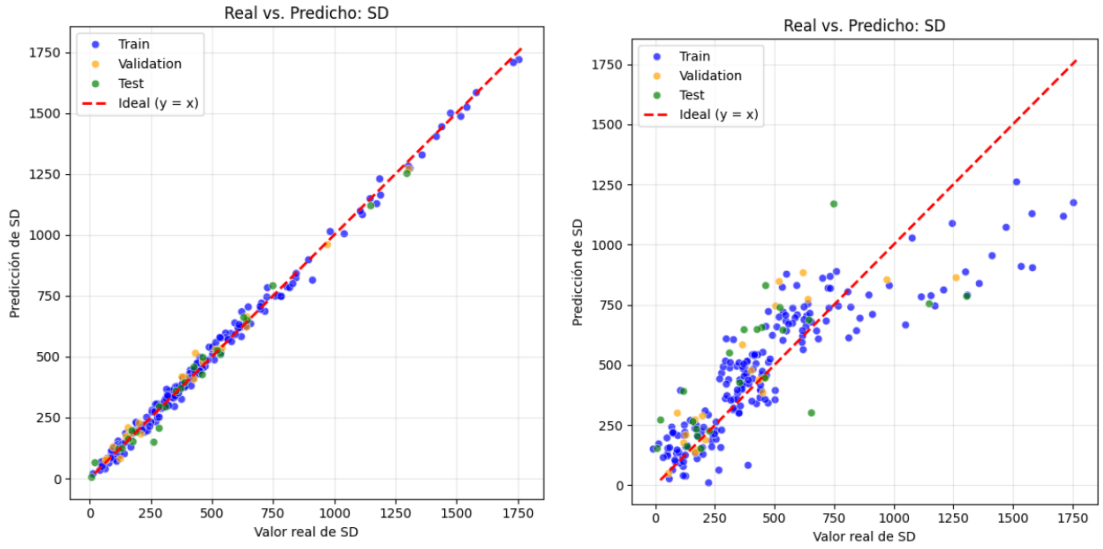
donde y son los valores reales y $\hat{y}$ las predicciones. Al normalizar el error absoluto medio por la suma de los valores reales, esta métrica facilita la interpretación del error relativo y resulta especialmente útil cuando la escala de los datos varía significativamente [Willmott and Matsuura, 2005].

Para el autoencoder, se utilizó el MSE. Por su parte, para los modelos de regresión se probó a usar como función de pérdida el MSE y el MAE normalizado, pero finalmente se optó por el MAE normalizado, puesto que, al ser más robusta daba lugar a modelos de regresión que se ajustaban mucho mejor a los datos sin aumento, tal y como se puede observar en la figura 6.3. Los modelos de regresión clásicos, como SVR o Random Forest, utilizaron como función de pérdida el MAE normalizado, mientras que, en el modelo multimodal LViT con cabeza de regresión, se combinó una función de pérdida de segmentación con una pérdida de regresión ponderada, lo cual permitió entrenar de forma conjunta ambas tareas (multitarea).

La Tabla 6.3 presenta un resumen de las funciones de pérdida aplicadas en los modelos empleados para tareas de regresión.

Tabla 6.3: Resumen de funciones de pérdida utilizadas en los modelos de regresión.

Modelo	**Función de pérdida utilizada**
3D AutoEncoder	MSE
Modelos de regresión clásicos	MAE normalizado y MSE
LViT + cabeza de regresión	Pérdida en segmentación + $\alpha\cdot$ MSE, con $\alpha = 0{,}5$



(a) (b)

Figura 6.3: Resultados de regresión para el modelo Ridge, aplicando MAE como función de pérdida en la primera figura (a), y aplicando MSE en la segunda (b), usando en ambos casos los datos sin aumento.

6.3.2. Métricas de evaluación

Para comparar de forma objetiva el rendimiento de los modelos entrenados, es fundamental definir métricas específicas que se ajusten a la naturaleza de cada tarea, ya sea segmentación o regresión.

Para segmentación se usó el *Dice Score*, que es una métrica esencial para evaluar el solapamiento entre la predicción binarizada de un modelo y la máscara de referencia en tareas de segmentación, especialmente en el ámbito médico [Milletari et al., 2016b, Sudre et al., 2017]. Se define matemáticamente como

$$\text{Dice} = \frac{2 \cdot |P \cap G| + \varepsilon}{|P| + |G| + \varepsilon}, \tag{6.7}$$

donde P es la predicción binarizada, G la máscara de referencia (valor real) y ε una constante de estabilización para evitar divisiones por cero. El *Dice Score* equivale al *F1-score* en segmentación binaria y su valor varía entre 0 (sin solapamiento) y 1 (coincidencia perfecta), proporcionando una medida directa de la calidad de la segmentación [Milletari et al., 2016b].

Por otro lado, IoU es otra métrica ampliamente utilizada para cuantificar la proporción de solapamiento respecto a la unión entre la predicción y la referencia [Jaccard, 1901, Rahman and Wang, 2016]. Su fórmula es

$$\text{IoU} = \frac{|P \cap G| + \varepsilon}{|P \cup G| + \varepsilon} = \frac{|P \cap G| + \varepsilon}{|P| + |G| - |P \cap G| + \varepsilon}, \tag{6.8}$$

donde P es la predicción, G la máscara de referencia y ε una constante de estabilización. El IoU penaliza más los errores que el *Dice Score* al considerar explícitamente la unión, y su valor también oscila entre 0 y 1, siendo 1 la coincidencia perfecta [Rahman and Wang, 2016].

En cuanto a las tareas de regresión, en el caso del 3D AutoEncoder, se optó por utilizar el MSE tanto como función de pérdida durante el entrenamiento como métrica de evaluación final. Esto es una práctica muy extendida porque el MSE es fácil de interpretar, penaliza fuertemente los errores grandes y ofrece una visión clara del ajuste global del modelo.

Por otro lado, para los modelos de regresión clásicos y la cabeza de regresión del modelo LViT, se empleó el MAE como métrica principal de evaluación, que resulta especialmente útil, ya que es más robusto ante valores atípicos y proporciona una interpretación directa del error promedio en las mismas unidades que los datos originales. A diferencia del MSE, el MAE no penaliza desproporcionadamente los errores grandes, lo que permite una evaluación más equilibrada del rendimiento del modelo cuando pueden existir outliers en los datos de supervivencia. Así, se obtiene una evaluación mucho más justa y objetiva del desempeño predictivo en el contexto clínico.

6.3.3. Hiperparámetros de entrenamiento

La elección adecuada de los hiperparámetros de entrenamiento es un aspecto esencial para lograr que las redes neuronales profundas y modelos de regresión clásicos aprendan de manera eficiente y generalicen correctamente a nuevos datos. Estos parámetros, que deben definirse antes de iniciar el entrenamiento, influyen de forma directa en la calidad del aprendizaje y en la estabilidad del proceso de optimización. Si no se seleccionan cuidadosamente, pueden conducir tanto a un ajuste excesivo al conjunto de entrenamiento como a una incapacidad del modelo para aprender patrones útiles.

En este trabajo, la selección de los valores de hiperparámetros se abordó combinando experimentación práctica y estrategias de optimización, adaptando cada ajuste a las particularidades del conjunto de datos BraTS2020 y a las arquitecturas de red empleadas. Además, se prestó especial atención a mantener un equilibrio entre el rendimiento obtenido y las limitaciones computacionales, algo especialmente relevante en modelos tridimensionales que requieren un uso intensivo de memoria.

La Tabla 6.4 recoge los hiperparámetros más relevantes de las redes neuronales principales que sí se utilizaron finalmente en la evaluación comparativa.

Tabla 6.4: Resumen de hiperparámetros utilizados en modelos principales.

Modelo	**Batch Size**	**LR**	**scheduler**	**Epochs**	**Acc. Grad.**
3D U-Net	1	5e-4	ReduceLROnPlateau	50	4
LViT	2	2e-4	CosineAnnealing	300/100	–
SegResNet	1	5e-4	ReduceLROnPlateau	50	4
AutoEncoder 3D	1	5e-4	ReduceLROnPlateau	50	4

En los modelos de regresión clásicos, cada algoritmo fue configurado con parámetros cuidadosamente seleccionados para optimizar su desempeño según las características del problema y los datos. Para comparar de forma rigurosa los distintos algoritmos implementados, se desarrolló un sistema de optimización de hiperparámetros basado en búsqueda aleatoria con validación cruzada de tres folds.

La búsqueda de hiperparámetros se realizó en cuatro escenarios experimentales: modelos entrenados con características latentes del 3D AutoEncoder sin aumento de datos,

modelos con características del 3D AutoEncoder con aumento de datos, modelos entrenados con características del bloque y4 de LViT sin aumento de datos, y modelos con características de LViT con aumento de datos. En cada uno de estos escenarios se ejecutó una búsqueda independiente de hiperparámetros, asegurando que las configuraciones óptimas fueran específicas para cada tipo de representación y estrategia de aumento de datos.

El espacio de búsqueda fue diseñado de forma informada, centrando los rangos de exploración en configuraciones previamente identificadas como efectivas. Por ejemplo, para Random Forest, la búsqueda se centró en torno a cien estimadores y una profundidad máxima de veinte niveles, mientras que para SVR con kernel RBF, los parámetros se centraron en $C = 2{,}0$, $\epsilon = 0{,}01$ y $\gamma = 0{,}00022$. Esta estrategia redujo significativamente el espacio de exploración y aumentó la probabilidad de encontrar configuraciones óptimas.

Los hiperparámetros óptimos para cada modelo, determinados por búsqueda aleatoria con validación cruzada de tres folds, se detallan a continuación según el escenario experimental.

Para la predicción de días de supervivencia normalizados utilizando características latentes del 3D AutoEncoder sin aumento de datos, RandomForest mostró el mejor desempeño con profundidad máxima de 21, características máximas de 0.919, 3 muestras mínimas por hoja, 5 para división y 140 estimadores. Ridge utilizó $\alpha = 2{,}068$, CatBoost empleó profundidad 3, regularización L2 de 10.965 y tasa de aprendizaje de 0.137. SVR RBF se configuró con $C = 2{,}826$, $\varepsilon = 0{,}008$ y $\gamma = 0{,}00023$. HistGradientBoosting utilizó regularización L2 de 0.089, tasa de aprendizaje de 0.065, profundidad máxima de 10, 366 iteraciones y 18 muestras mínimas por hoja. SVR lineal empleó $C = 61{,}285$ y $\varepsilon = 0{,}140$. Ridge Polynomial utilizó $\alpha = 37{,}554$. MLPRegressor se configuró con $\alpha = 0{,}016$ y tasa de aprendizaje inicial de 0.017.

Con aumento de datos aplicado a las características del 3D AutoEncoder, RandomForest mantuvo su superioridad con la misma configuración. Ridge utilizó $\alpha = 37{,}464$, CatBoost empleó profundidad 3, regularización L2 de 10.965 y tasa de aprendizaje de 0.137. SVR RBF se configuró con $C = 2{,}796$, $\varepsilon = 0{,}008$ y $\gamma = 0{,}0004$. HistGradientBoosting utilizó regularización L2 de 0.066, tasa de aprendizaje de 0.060, profundidad máxima de 9, 311 iteraciones y 19 muestras mínimas por hoja. SVR lineal empleó $C = 15{,}7$ y $\varepsilon = 0{,}157$. Ridge Polynomial mantuvo $\alpha = 37{,}554$. MLPRegressor se configuró con $\alpha = 0{,}083$ y tasa de aprendizaje inicial de 0.022.

Para los modelos entrenados con características del bloque y4 de LViT sin aumento de datos, CatBoost mostró el mejor desempeño con profundidad 3, regularización L2 de 10.965 y tasa de aprendizaje de 0.137. Ridge utilizó $\alpha = 37{,}464$, RandomForest se configuró con profundidad máxima de 18, características máximas de 0.837, 3 muestras mínimas por hoja, 7 para división y 130 estimadores. SVR RBF empleó $C = 2{,}124$, $\varepsilon = 0{,}024$ y $\gamma = 0{,}00156$. HistGradientBoosting utilizó regularización L2 de 0.066, tasa de aprendizaje de 0.060, profundidad máxima de 9, 311 iteraciones y 19 muestras mínimas por hoja. SVR lineal se configuró con $C = 15{,}7$ y $\varepsilon = 0{,}157$. Ridge Polynomial utilizó $\alpha = 97{,}091$. MLPRegressor empleó $\alpha = 0{,}083$ y tasa de aprendizaje inicial de 0.022.

Finalmente, con aumento de datos aplicado a las características del bloque y4 de LViT, SVR RBF obtuvo el mejor rendimiento con $C = 2{,}124$, $\varepsilon = 0{,}024$ y $\gamma = 0{,}00156$. CatBoost se configuró con profundidad 4, regularización L2 de 8.987 y tasa de aprendizaje de 0.131. RandomForest utilizó profundidad máxima de 21, características máximas de 0.919, 3 muestras mínimas por hoja, 5 para división y 140 estimadores. Ridge empleó $\alpha = 37{,}464$. HistGradientBoosting se configuró con regularización L2 de 0.034, tasa de

aprendizaje de 0.109, profundidad máxima de 4, 353 iteraciones y 18 muestras mínimas por hoja. SVR lineal utilizó $C = 37{,}554$ y $\varepsilon = 0{,}952$. Ridge Polynomial empleó $\alpha = 97{,}091$. MLPRegressor se configuró con $\alpha = 0{,}083$ y tasa de aprendizaje inicial de 0.022.

6.3.4. Optimizadores y técnicas de mejora

Durante el entrenamiento, la selección de optimizadores y técnicas de mejora se adaptó tanto a los modelos de regresión clásicos como a las redes neuronales profundas, considerando las particularidades de cada enfoque.

En los modelos de regresión clásicos, como SVR, Random Forest, HistGradientBoostingRegressor o Ridge, la optimización se realiza principalmente mediante algoritmos internos específicos de cada modelo, como la minimización de funciones de coste o el ajuste de árboles y kernels. En este contexto, la configuración relevante suele centrarse en parámetros propios de cada algoritmo, como el número de árboles, la regularización, el tipo de kernel o el grado del polinomio, y no en la elección explícita de optimizadores genéricos o técnicas de ajuste dinámico del aprendizaje.

Por otro lado, en las redes neuronales profundas y modelos de segmentación, la elección del optimizador y las técnicas de mejora resulta crucial para la estabilidad y eficiencia del entrenamiento. El optimizador Adam fue empleado de manera generalizada en modelos como 3D U-Net, LViT, SegResNet y el AutoEncoder 3D, gracias a su capacidad para adaptar dinámicamente la tasa de aprendizaje y acelerar la convergencia, combinando las ventajas del descenso por gradiente con momento y la adaptación RMSprop [Kingma and Ba, 2015, BytePlus, 2025]. Adam ajusta los pasos de actualización utilizando tanto la media del gradiente como la media de sus cuadrados, lo que permite un entrenamiento más estable y eficiente.

Entre las técnicas de mejora, destaca el uso de *early stopping* en LViT, que monitoriza la métrica de validación y detiene el entrenamiento si no se observan mejoras tras un número determinado de épocas, lo que previene el sobreajuste y reduce el tiempo computacional [Prechelt, 1998, Zilliz, 2023]. Además, en los modelos tridimensionales se aplicó acumulación de gradientes, una estrategia que permite simular un mayor tamaño de lote efectivo acumulando gradientes a lo largo de varias iteraciones antes de actualizar los pesos, lo que resulta especialmente útil cuando existen limitaciones de memoria en GPU [Labs, 2024].

Capítulo 7

Resultados experimentales

En este capítulo se presentan los resultados obtenidos durante la fase de pruebas del proyecto, analizando tanto los datos numéricos como las observaciones derivadas de la evaluación de los distintos modelos desarrollados. El objetivo es ofrecer una visión clara del rendimiento de las soluciones propuestas para la identificación automática de tumores cerebrales y la predicción de factores clínicos relevantes, señalando aciertos, limitaciones y posibles mejoras. Esta valoración experimental se fundamenta en un análisis detallado de técnicas de aprendizaje profundo aplicadas a imágenes cerebrales en oncología, examinando el desempeño individual y combinado de los modelos, y proporcionando información relevante sobre la efectividad real de las estrategias implementadas, así como patrones que pueden orientar futuras investigaciones.

Los resultados se organizan en tres apartados principales: análisis de los modelos de segmentación (U-Net 3D, SegResNet y LViT), evaluación de los modelos de regresión (tanto convencionales como integrados con LViT) y una comparativa global que integra los hallazgos de ambas tareas. Cada sección incluye curvas de aprendizaje, datos cuantitativos, visualizaciones y discusiones para facilitar la interpretación de los resultados y las limitaciones de cada enfoque.

7.1. Resultados de segmentación

Esta sección presenta un análisis detallado del rendimiento de los modelos de segmentación implementados (U-Net 3D, SegResNet y LViT) para la segmentación automática de tumores cerebrales, evaluando su capacidad para delimitar las distintas regiones tumorales en resonancias magnéticas.

La evaluación se estructura en cuatro partes complementarias: análisis de curvas de entrenamiento para detectar convergencia y sobreajuste, evaluación cuantitativa mediante métricas Dice e IoU por subregión tumoral, evaluación cualitativa con visualizaciones de segmentaciones y errores, y una discusión comparativa de fortalezas y debilidades considerando robustez y generalización.

7.1.1. Curvas de entrenamiento

Se presentan las trayectorias de la función de pérdida y las métricas clave de segmentación (Dice e IoU/Jaccard) para cada modelo, lo que proporciona una visión detallada del desempeño y la estabilidad del aprendizaje durante el entrenamiento. Estas curvas

permiten comparar de forma directa no solo la convergencia de cada arquitectura, sino también su capacidad de generalización y robustez frente a los datos de validación.

Para la red 3D U-Net, la figura 7.1 ilustra la evolución de la función de pérdida en entrenamiento y validación, mostrando una convergencia estable a partir de la época 20, con valores finales de 0.1469 (entrenamiento) y 0.1567 (validación). Las figuras 7.2a y 7.2b presentan el incremento progresivo y estable de los coeficientes Dice y Jaccard, alcanzando máximos de 0.8702 y 0.7751 (entrenamiento), y 0.8592 y 0.7589 (validación), respectivamente. La diferencia inferior al 2 % entre métricas de entrenamiento y validación indica buena generalización y ausencia de sobreajuste, consolidando a U-Net 3D como una arquitectura robusta para segmentación de tumores cerebrales. Además, la estabilidad de las curvas sugiere que el modelo logra un aprendizaje consistente a lo largo del proceso.

En la red neuronal LViT, la figura 7.3a muestra un descenso constante de la función de pérdida y un aumento firme de IoU y Dice en ambos conjuntos. Aunque se observan fluctuaciones en las métricas de validación a mitad del entrenamiento, estas reflejan la exploración de parámetros típica de los transformers y no indican inestabilidad significativa. Los resultados finales de LViT son sólidos, con un IoU de validación cercano a 0.7 y un Dice de validación alrededor de 0.8, cifras comparables a U-Net 3D, y sin indicios de sobreajuste, lo que demuestra que LViT es capaz de aprender representaciones complejas y mantener un rendimiento competitivo en tareas de segmentación médica.

Para la red neuronal SegResNet, el entrenamiento se adaptó dinámicamente tras observar en las curvas de pérdida y precisión que el modelo seguía mejorando tras 15 épocas, sin señales de sobreajuste. Por ello, se amplió el número de épocas, lo que resultó en mayor estabilidad y mejor generalización. Los resultados de esta estrategia extendida se muestran en la figura 7.4, evidenciando la evolución positiva de las métricas principales y confirmando que el ajuste dinámico de los hiperparámetros puede ser clave para optimizar el rendimiento de arquitecturas profundas en segmentación.

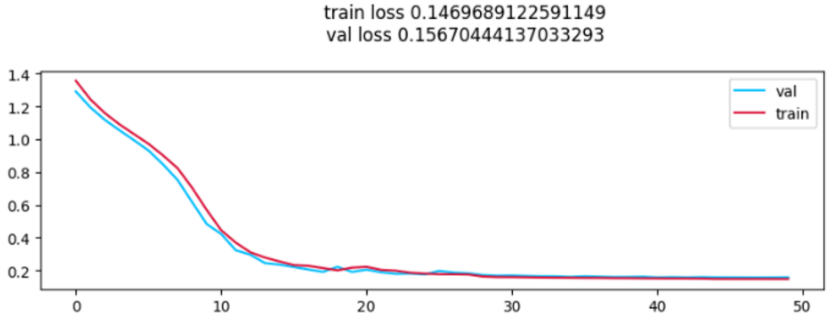
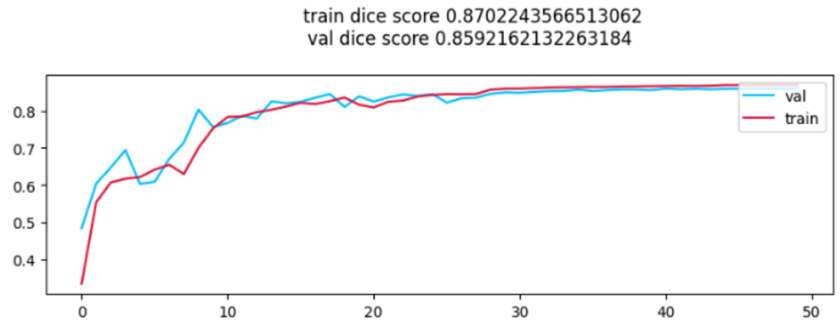


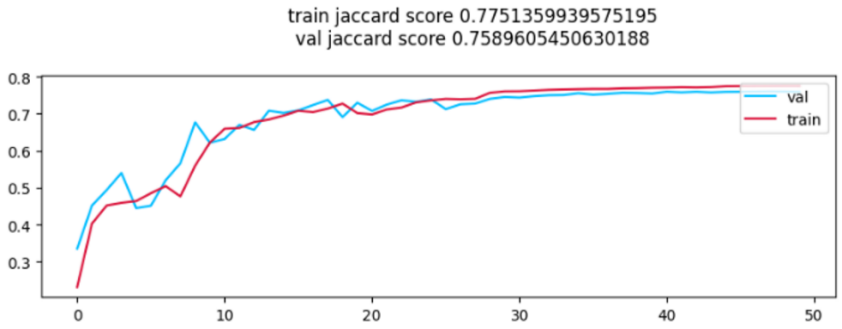
Figura 7.1: Curva de pérdida durante el entrenamiento y la validación del modelo U-Net 3D.

7.1.2. Análisis de resultados

La evaluación cuantitativa es fundamental para comparar objetivamente el rendimiento de los modelos de segmentación desarrollados, utilizando un conjunto de pruebas independiente para asegurar la validez de los resultados y la capacidad de generalización. El protocolo de evaluación calcula métricas estándar de segmentación médica, como los coeficientes Dice e IoU, aplicadas a las tres subregiones tumorales principales del protocolo



(a) Coeficiente de Dice



(b) Coeficiente de Jaccard (IoU)

Figura 7.2: Evolución de las métricas de segmentación para el modelo 3D U-Net en entrenamiento y validación.

BraTS, permitiendo identificar tanto el rendimiento global como las fortalezas específicas de cada modelo.

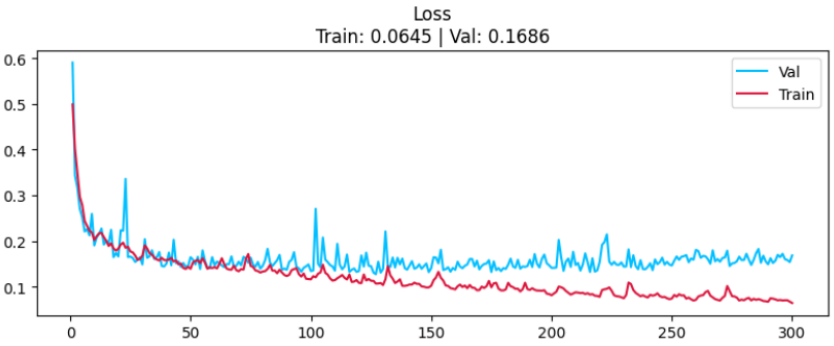
Los resultados detallados se presentan en la tabla 7.1, donde se destacan en negrita los valores más altos por clase. El modelo 3D U-Net demuestra un rendimiento superior en la segmentación de la región WT, mientras que SegResNet presenta mejores resultados para las regiones TC y ET, aunque con diferencias relativamente pequeñas entre ambas arquitecturas. Estos hallazgos indican que cada modelo presenta fortalezas específicas dependiendo de la subregión tumoral analizada.

Tabla 7.1: Coeficientes Dice e IoU (%) por clase y modelo en el conjunto de test. En negrita se destacan los mejores resultados por métrica.

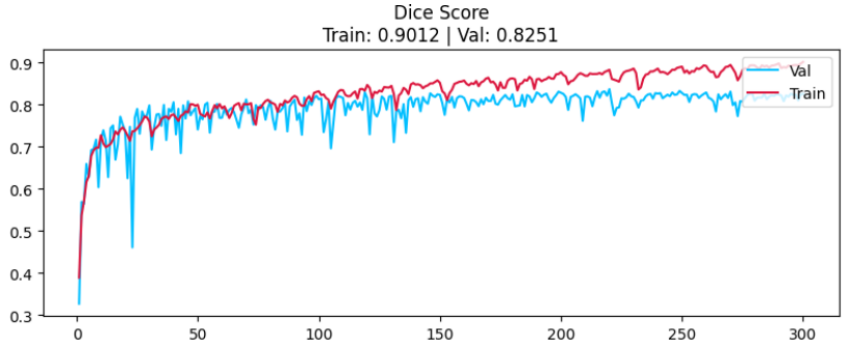
Modelo	WT Dice	WT IoU	TC Dice	TC IoU	ET Dice	ET IoU
LViT	72.8 %	62.1 %	–	–	–	–
3D U-Net	**84.2 %**	**74.8 %**	46.2 %	36.3 %	50.0 %	41.0 %
SegResNet	28.1 %	20.9 %	**52.2 %**	**40.3 %**	**51.0 %**	**42.6 %**

Para complementar el análisis cuantitativo, se seleccionó un caso representativo del conjunto de test (Paciente 296), cuyas imágenes multimodales se muestran en las figuras 7.5 y 7.6, y el valor real en la figura 7.7. La figura 7.8 compara visualmente las segmentaciones generadas por cada modelo (3D U-Net, LViT y SegResNet) con la máscara real, mostrando tanto las áreas correctamente segmentadas como los errores a través de mapas de diferencia.

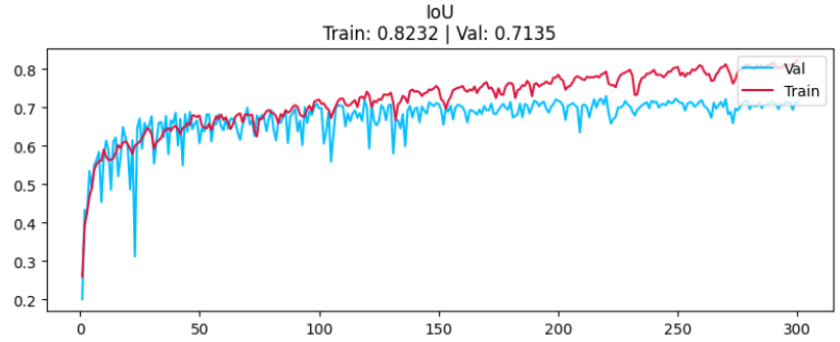
En la parte superior de la figura 7.8, se observan las segmentaciones producidas por cada modelo, mientras que la fila intermedia resalta las discrepancias respecto al valor real,



(a) Pérdida



(b) Coeficiente de Dice

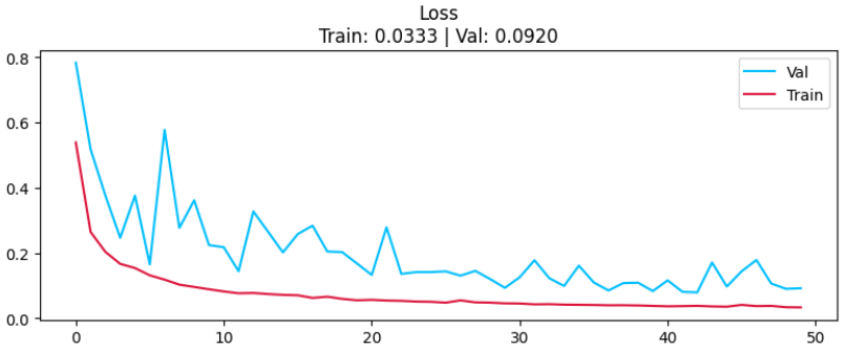


(c) Coeficiente de Jaccard (IoU)

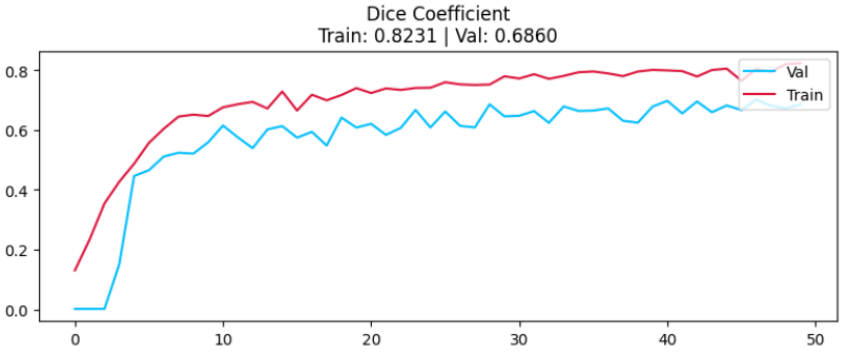
Figura 7.3: Evolución de las métricas de segmentación y pérdida para el modelo LViT en entrenamiento y validación.

permitiendo identificar visualmente las regiones donde cada arquitectura falla o acierta. En la parte inferior, las gráficas de barras resumen las métricas de desempeño por clase (en este caso el Dice), facilitando la comparación cuantitativa entre modelos.

Para los modelos que operan de manera natural en el ámbito tridimensional, como 3D U-Net y SegResNet, se han creado representaciones volumétricas adicionales que reflejan la esencia tridimensional de sus predicciones. El modelo 3D U-Net demuestra un rendimiento superior en todas las regiones tumorales (WT, TC y ET), mientras que SegResNet presenta resultados competitivos especialmente en las regiones TC y ET. Estas visualizaciones en 3D, presentadas como renderizados volumétricos en la figura 7.9, permiten apreciar la coherencia espacial y la continuidad anatómica que estos modelos logran mantener a lo largo de todo el volumen.



(a) Pérdida



(b) Coeficiente de Dice Global

Figura 7.4: Resultados de pérdida y métrica Dice durante el entrenamiento y validación usando SegResNet.

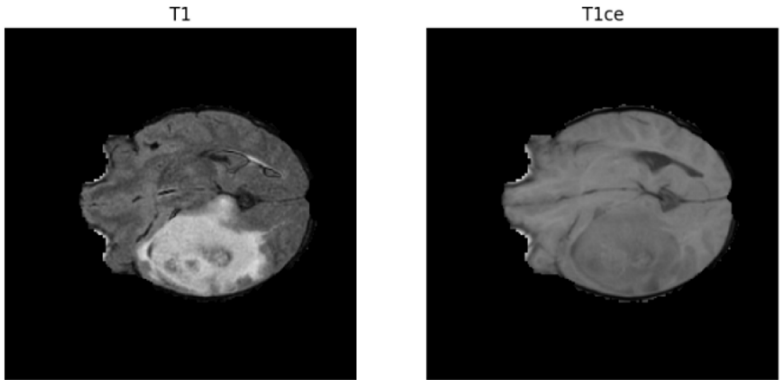


Figura 7.5: Imágenes T1 y T1ce del paciente 296 perteneciente a test.

7.1.3. Discusión comparativa

El análisis comparativo de los modelos evaluados pone de manifiesto patrones de rendimiento únicos que reflejan las características arquitectónicas de cada enfoque.

El modelo LViT muestra un rendimiento sólido en la segmentación de WT (Whole Tumor), alcanzando un Dice del 72,8 % y un IoU del 62,1 %, lo que lo posiciona como una opción competitiva dentro de los métodos basados en transformers para segmentación médica. Sin embargo, su arquitectura original presenta limitaciones importantes: solo permite realizar un tipo de segmentación a la vez y está diseñada para trabajar con imágenes 2D, lo que restringe su aplicabilidad en tareas multiclase o en contextos donde la información volumétrica 3D es clave. Por tanto, aunque LViT obtiene buenos resultados

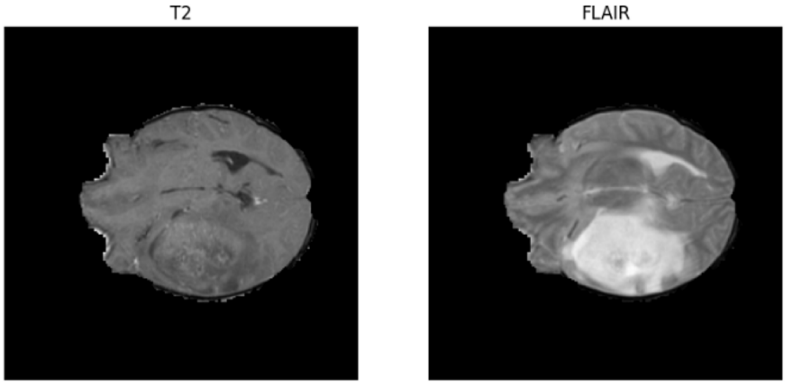


Figura 7.6: Imágenes T2 y flair del paciente 296 perteneciente a test.

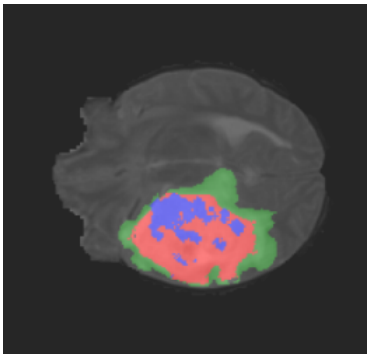


Figura 7.7: Valor real para cada tipo de segmentación del paciente 296 perteneciente a test.

en la detección de la región tumoral completa, no puede abordar de manera integral la segmentación simultánea de múltiples subregiones tumorales, a diferencia de otros modelos como 3D U-Net o SegResNet.

En cambio, la red 3D U-Net demuestra superioridad en la segmentación de la región WT gracias a su capacidad para procesar información volumétrica completa y mantener coherencia espacial tridimensional. Sus convoluciones 3D capturan patrones anatómicos complejos a través de múltiples cortes, mientras que las conexiones de salto preservan información de alta resolución durante la codificación-decodificación, facilitando la reconstrucción precisa de estructuras tumorales extensas y heterogéneas.

Por su parte, SegResNet presenta rendimiento aceptable en las regiones TC y ET, donde su arquitectura residual captura adecuadamente características localizadas. Sin embargo, su desempeño se compromete en la segmentación de WT, ya que la complejidad volumétrica y la necesidad de integrar información contextual tridimensional exceden las capacidades de su diseño residual, limitando su eficiencia en el procesamiento de información volumétrica extensa requerida para la segmentación completa del tumor.

El análisis global indica que los modelos 3D, como 3D U-Net y SegResNet, junto con LViT, ofrecen mayor robustez y coherencia anatómica. 3D U-Net sobresale particularmente en la segmentación de WT, mientras que LViT también demuestra un rendimiento destacado en esta región. SegResNet presenta un desempeño ligeramente superior a 3D U-Net en las regiones ET y TC, aunque su rendimiento se ve considerablemente comprometido en WT. A pesar de las ventajas de 3D U-Net, su mayor demanda de recursos computacionales puede limitar su uso en entornos con hardware restringido.

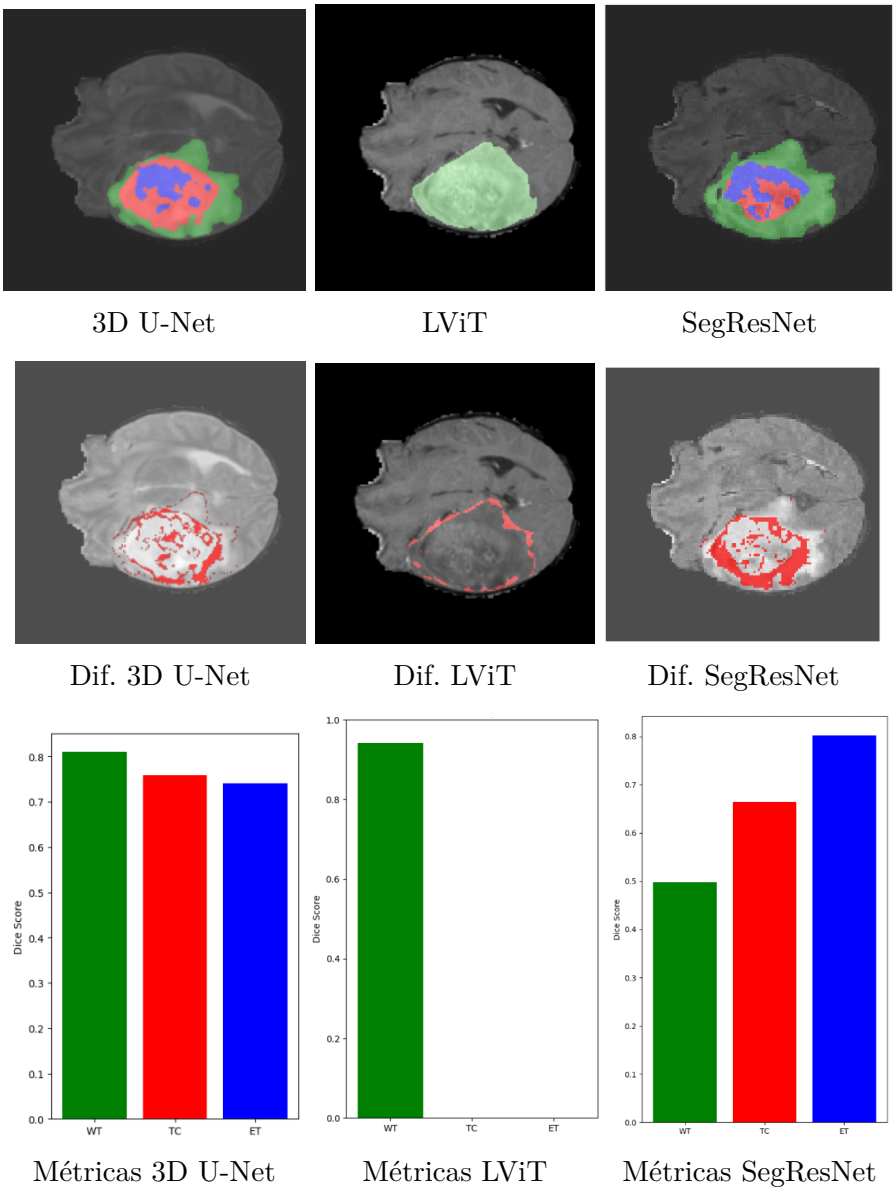


Figura 7.8: Comparación entre el valor real y las predicciones de 3 modelos de segmentación en imágenes cerebrales, junto a sus mapas de error y métricas Dice por clase. El color verde corresponde a la métrica Dice en WT, el color rojo corresponde a la métrica Dice en TC, y el azul corresponde a la métrica Dice en ET.

7.2. Resultados de regresión

Esta sección se centra en los resultados de la predicción de variables clínicas continuas, como los días de supervivencia, mediante modelos de regresión que conectan características extraídas de resonancias cerebrales con datos clínicos. Se evaluaron tanto modelos tradicionales de aprendizaje automático como arquitecturas multimodales avanzadas, incluyendo LViT con cabezas de regresión especializadas.

La estrategia incluye modelos que trabajan directamente con características extraídas de imágenes y sistemas híbridos que combinan representaciones latentes con información clínica estructurada, tal y como se ha nombrado anteriormente. Esto permite analizar por separado la calidad de las representaciones latentes y la capacidad predictiva de los

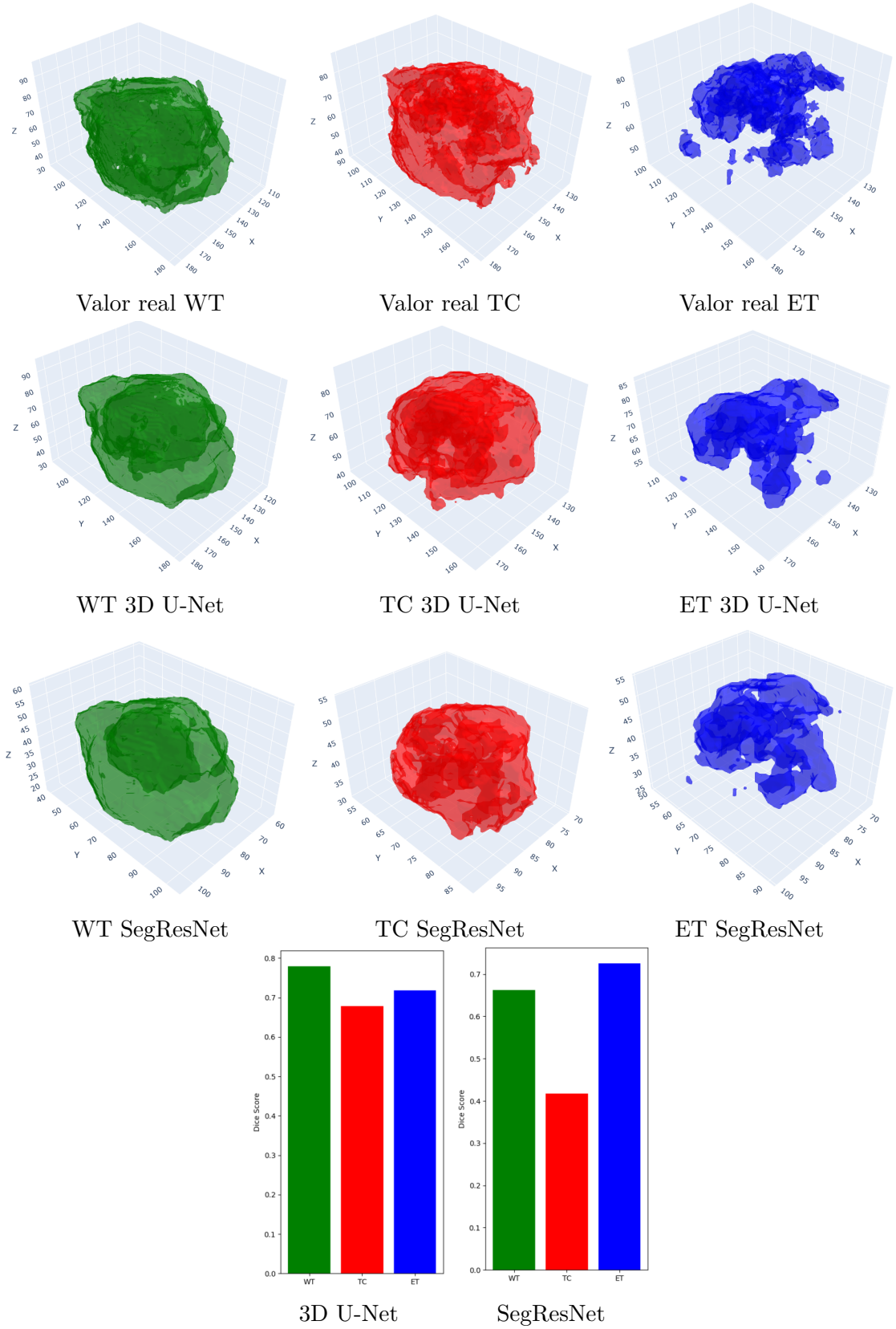


Figura 7.9: Comparación tridimensional entre el valor real y las predicciones de 2 modelos de segmentación en imágenes cerebrales, que trabajan con volúmenes 3D.

algoritmos de regresión, facilitando una comprensión detallada de los factores que influyen en el rendimiento del sistema.

7.2.1. Curvas de entrenamiento y validación

En este apartado, se estudia cómo evolucionan los errores en varios modelos, entre ellos, SVR, MLP y la arquitectura híbrida LViT con componentes de regresión, entre otros, lo que permite comparar directamente cómo aprende cada uno y determinar cuáles son más estables y eficaces para minimizar los errores de predicción.

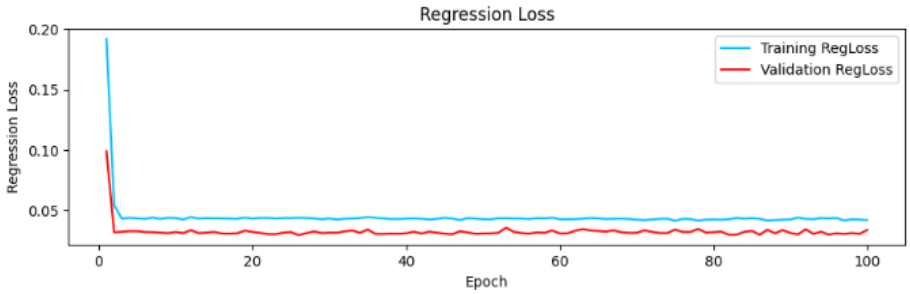
El modelo 3D AutoEncoder fue entrenado para reconstruir volúmenes de resonancia magnética y extraer características latentes útiles para predicción clínica. Durante el entrenamiento, la pérdida disminuyó rápidamente en las primeras épocas, pasando de valores iniciales de aproximadamente 0.013 a estabilizarse en torno a 0.002 tras las primeras 10 épocas, lo que indica una rápida adaptación y convergencia eficiente del modelo. Tanto la pérdida de entrenamiento como la de validación evolucionaron de forma paralela y se mantuvieron próximas durante todo el proceso, lo que sugiere una buena generalización y ausencia de sobreajuste.

En cuanto al tiempo computacional, se observó una reducción progresiva desde 4.5 minutos por época al inicio hasta estabilizarse en 3.9 minutos después de la época 15. Un pico puntual de 4.1 minutos cerca de la época 45 se atribuye a operaciones adicionales como validaciones o guardado de checkpoints. Estos resultados reflejan un entrenamiento eficiente y estable, con buen equilibrio entre aprendizaje y generalización.

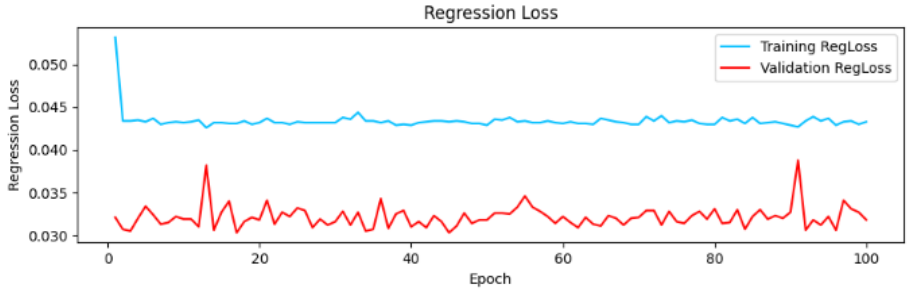
En cuanto a la red neuronal LViT con cabeza de regresión, que integra imagen y texto clínico para segmentación y predicción de supervivencia, permite evaluar cómo diferentes configuraciones de la cabeza de regresión, con variaciones en arquitectura y regularización, afectan el aprendizaje y la capacidad predictiva del modelo, como se observa en las gráficas de regresión, en la figura 7.10, donde se aprecia que cada cabeza tiene su propio ritmo, unas más estables y otras que luchan por converger, lo que deja claro que el modelo es bastante susceptible a los hiperparámetros y a cómo está construida cada cabeza. En dicha figura 7.10, las cabezas 1 y 3 van reduciendo la pérdida tanto al entrenar como al validar, lo que parece indicar que aprenden mejor y de forma más estable con el tiempo. En cambio, las cabezas 2 y 4 oscilan mças y la diferencia entre las curvas es mayor, lo que podría significar que les cuesta adaptarse a la validación, quizás por sobreajuste o por no tener los hiperparámetros bien ajustados. Esta diferencia en el comportamiento también podría deberse a lo complicado que es la función de pérdida combinada y a lo difícil que es optimizar tareas tan distintas a la vez.

Con estos primeros resultados, es posible comparar el rendimiento de cada diseño de cabeza de regresión y orientar la selección de configuraciones para futuras versiones del modelo. Sin embargo, en el ámbito de la regresión, las cabezas muestran una clara dificultad para ajustarse a los datos: aunque la pérdida disminuye a lo largo de las épocas, las predicciones no logran capturar adecuadamente la variabilidad real de las variables clínicas, lo que evidencia una limitada capacidad de ajuste y subraya la necesidad de mejorar la arquitectura o los hiperparámetros para obtener resultados más precisos.

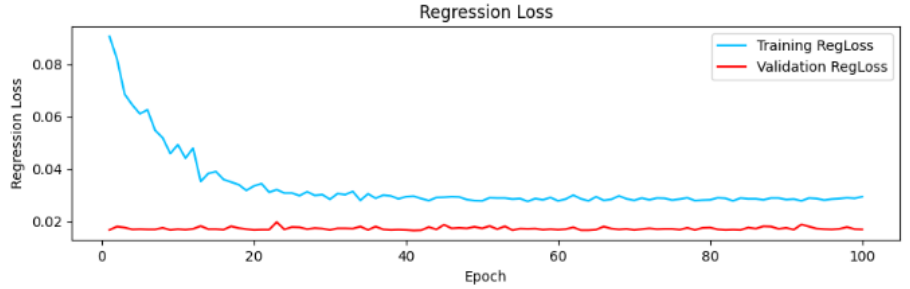
Además de las redes neuronales, se entrenaron diversos modelos de regresión clásicos como SVR, Random Forest, Gradient Boosting, Ridge con características polinómicas, MLPRegressor y redes con Keras, para predecir los días de supervivencia (SD) utilizando dos tipos de entrada: por un lado, las representaciones latentes extraídas del codificador del 3D AutoEncoder combinadas con estadísticas vóxel de las segmentaciones; y por otro,



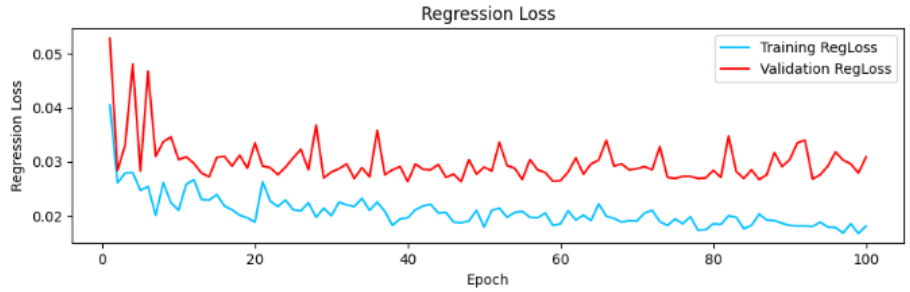
(a) 1ª cabeza de regresión



(b) 2ª cabeza de regresión



(c) 3ª cabeza de regresión



(d) 4ª cabeza de regresión

Figura 7.10: Comparación de resultados para las distintas cabezas de regresión.

los vectores obtenidos del bloque `y4` de la red LviT, que integran información visual y textual.

Para evaluar de manera rigurosa el rendimiento de los diferentes algoritmos de regresión implementados, se desarrolló un sistema de optimización de hiperparámetros que permite identificar las configuraciones óptimas para cada modelo, tal y como se explicó anteriormente.

A modo de ejemplo, las Tablas 7.2 y 7.3 muestran los resultados de optimización de hiperparámetros para los algoritmos SVR Linear y Ridge, comparando el rendimiento con

diferentes configuraciones y tipos de características. La Tabla 7.2 presenta las métricas de validación para múltiples configuraciones evaluadas, mostrando cómo diferentes valores de C y epsilon afectan el rendimiento según el tipo de características utilizadas (3D AutoEncoder vs LViT) y la aplicación de aumento de datos. La Tabla 7.3 muestra los resultados correspondientes para la regresión Ridge, evidenciando variaciones significativas en las métricas de validación según el valor del parámetro de regularización alpha.

De manera similar, las Tablas 7.4 y 7.5 ilustran el proceso de optimización para los algoritmos Random Forest y CatBoost respectivamente, mostrando las configuraciones de hiperparámetros evaluadas y sus correspondientes métricas de rendimiento. La Tabla 7.4 presenta resultados para diferentes combinaciones de profundidad máxima, número de características, muestras mínimas por hoja y estimadores, mientras que la Tabla 7.5 muestra las configuraciones óptimas de profundidad, regularización L2 y tasa de aprendizaje.

Las Tablas 7.6 y 7.7 presentan los resultados para SVR con kernel RBF y Histogram Gradient Boosting respectivamente, evidenciando la complejidad del espacio de hiperparámetros y la importancia de la optimización sistemática. Finalmente, las Tablas 7.8 y 7.9 muestran las configuraciones encontradas para Ridge Polynomial y MLP Regressor, completando el análisis comparativo de todos los algoritmos evaluados en el estudio.

Tabla 7.2: Configuraciones de SVR Linear

Parámetros	**DA**	**Características**	**CV Val MAE**
C: 59.34, ε: 0.047	–	3D AutoEncoder	0.0582
C: 15.70, ε: 0.157	✓	3D AutoEncoder	0.1414
C: 37.55, ε: 0.952	✓	3D AutoEncoder	0.2420
C: 15.70, ε: 0.157	–	LViT	0.2006
C: 29.31, ε: 0.367	✓	LViT	0.2256
C: 37.55, ε: 0.952	✓	LViT	0.2373

Tabla 7.3: Configuraciones de Ridge

Parámetros	**DA**	**Características**	**CV Val MAE**
α: 2.068	–	3D AutoEncoder	0.0228
α: 37.464	✓	3D AutoEncoder	0.0990
α: 97.001	–	3D AutoEncoder	0.1327
α: 37.464	✓	LViT	0.2341
α: 2.068	✓	LViT	0.3113
α: 37.464	–	LViT	0.0607

Tabla 7.4: Configuraciones de RandomForest

Parámetros	**DA**	**Características**	**CV Val MAE**
depth: 23, features: 0.986, leaf: 2, split: 4, est: 88	–	3D AutoEncoder	0.0127
depth: 19, features: 0.662, leaf: 3, split: 5, est: 103	✓	3D AutoEncoder	0.0194
depth: 20, features: 0.600, leaf: 1, split: 3, est: 137	–	3D AutoEncoder	0.0239
depth: 22, features: 0.937, leaf: 1, split: 4, est: 132	–	LViT	0.1774
depth: 21, features: 0.919, leaf: 3, split: 5, est: 140	✓	LViT	0.1819
depth: 24, features: 0.639, leaf: 3, split: 6, est: 123	✓	LViT	0.1849

Tabla 7.5: Configuraciones de CatBoost

Parámetros	DA	Características	CV Val MAE
depth: 3, L2: 9.803, lr: 0.190	–	3D AutoEncoder	0.0337
depth: 3, L2: 10.965, lr: 0.137	✓	3D AutoEncoder	0.0406
depth: 4, L2: 3.564, lr: 0.244	✓	3D AutoEncoder	0.0652
depth: 4, L2: 3.564, lr: 0.244	✓	LViT	0.1989
depth: 3, L2: 10.965, lr: 0.137	✓	LViT	0.2140
depth: 3, L2: 9.175, lr: 0.222	✓	LViT	0.2144

Tabla 7.6: Configuraciones de SVR RBF

Parámetros	DA	Características	CV Val MAE
C: 2.796, ε: 0.008, γ: 0.0004	–	3D AutoEncoder	0.0841
C: 2.826, ε: 0.008, γ: 0.00023	✓	3D AutoEncoder	0.1254
C: 1.103, ε: 0.023, γ: 0.0006	–	3D AutoEncoder	0.1717
C: 2.124, ε: 0.024, γ: 0.0016	✓	LViT	0.1431
C: 3.847, ε: 0.024, γ: 0.0017	✓	LViT	0.1340
C: 2.794, ε: 0.023, γ: 0.0003	✓	LViT	0.1735

Tabla 7.7: Configuraciones de HistGradientBoosting

Parámetros	DA	Características	CV Val MAE
L2: 0.034, lr: 0.109, depth: 4, iter: 353, leaf: 18	–	3D AutoEncoder	0.0148
L2: 0.089, lr: 0.065, depth: 10, iter: 366, leaf: 18	✓	3D AutoEncoder	0.0473
L2: 0.005, lr: 0.129, depth: 5, iter: 357, leaf: 34	–	3D AutoEncoder	0.0896
L2: 0.343, lr: 0.059, depth: 9, iter: 311, leaf: 19	–	LViT	0.1712
L2: 0.089, lr: 0.065, depth: 10, iter: 366, leaf: 18	✓	LViT	0.1703
L2: 0.149, lr: 0.131, depth: 10, iter: 373, leaf: 27	✓	LViT	0.1865

Tabla 7.8: Configuraciones de Ridge Polynomial

Parámetros	DA	Características	CV Val MAE
α: 97.091	–	3D AutoEncoder	0.1690
α: 37.554	✓	3D AutoEncoder	0.1692
α: 2.158	–	3D AutoEncoder	0.2453
α: 97.091	✓	LViT	0.1934
α: 37.554	✓	LViT	0.1965
α: 2.158	✓	LViT	0.1988

Tabla 7.9: Configuraciones de MLPRegressor

Parámetros	DA	Características	CV Val MAE
α: 0.061, lr: 0.015	–	3D AutoEncoder	0.2080
α: 0.083, lr: 0.022	✓	3D AutoEncoder	0.2185
α: 0.046, lr: 0.080	–	3D AutoEncoder	1.0653
α: 0.083, lr: 0.022	✓	LViT	0.2343
α: 0.061, lr: 0.015	✓	LViT	0.2627
α: 0.038, lr: 0.096	✓	LViT	4.4640

Luego, en cuanto a las visualizaciones de regresión, proporcionan una perspectiva cualitativa del rendimiento de los modelos optimizados. La figura 7.11 presenta los gráficos de valores reales versus predichos para los mejores modelos de regresión entrenados con características latentes del 3D AutoEncoder sin aplicar un aumento de datos. Estos gráficos revelan que modelos como Random Forest y Ridge sin características polinómicas logran una correlación estrecha entre valores reales y predichos, evidenciada por la concentración de puntos cerca de la línea diagonal ideal, sin embargo, los demás modelos no llegan a predecir bien los valores más altos. En contraste, la figura 7.12 muestra los resultados de los mismos modelos tras aplicar una estrategia de aumento de datos mediante la incorporación de ruido gaussiano en las observaciones asociadas a los valores superiores de la variable objetivo. Esta intervención permite mejorar la capacidad predictiva del modelo, especialmente en los rangos donde inicialmente presentaba mayores dificultades de ajuste. Se observa una mejora general en la precisión de las predicciones y, de forma particularmente destacada en modelos como Gradient Boosting, una reducción significativa en la dispersión de los valores predichos.

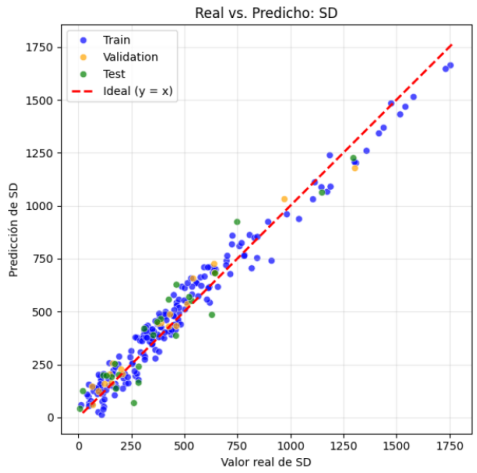
Las figuras 7.13 y 7.14 presentan los resultados de los modelos entrenados con representaciones y4 del LViT sin y con aumento de datos, mostrando un comportamiento considerablemente diferente. Los gráficos de dispersión revelan una correlación notablemente más débil entre los valores predichos y reales en comparación con los resultados del autoencoder 3D. En la figura 7.13, correspondiente a los modelos sin aumento de datos, se observa una dispersión significativa de los puntos alrededor de la línea de identidad ideal, con valores de correlación que oscilan entre 0.4 y 0.6 para la mayoría de los modelos. Los puntos se distribuyen de manera más errática, especialmente en los rangos de supervivencia más altos, donde las predicciones tienden a subestimar sistemáticamente los valores reales.

La figura 7.14, que muestra los resultados con aumento de datos, presenta una mejora marginal en la correlación, con valores que alcanzan aproximadamente 0.7 en los mejores casos, particularmente para los modelos CatBoost y SVR con kernel RBF. Sin embargo, la dispersión sigue siendo considerable, y se mantiene la tendencia de subestimación en los valores de supervivencia más elevados. Los modelos Ridge y Random Forest muestran un comportamiento particularmente inconsistente, con correlaciones que no superan el 0.5, evidenciando dificultades para capturar patrones complejos en las representaciones del LViT.

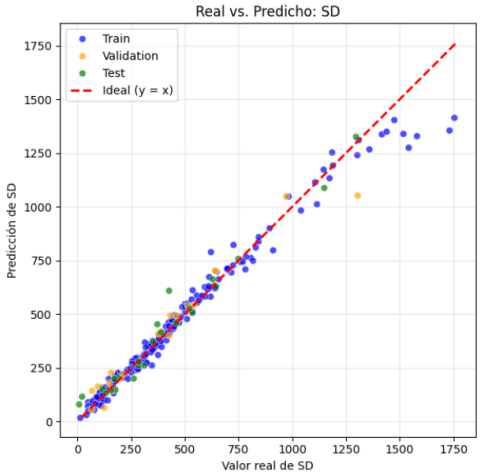
7.2.2. Análisis de resultados

En cuanto a la tarea de regresión, se evaluaron las predicciones sobre los días de supervivencia (SD) utilizando diferentes tipos de representaciones de entrada y estrategias de entrenamiento. Los resultados se presentan en las Tablas 7.10, 7.11 y 7.12, donde se agrupan los distintos modelos según el tipo de representación de entrada: características latentes del AutoEncoder 3D junto con estadísticas vóxel de las segmentaciones, representaciones del codificador ViT y salidas de la cabeza de regresión de LViT.

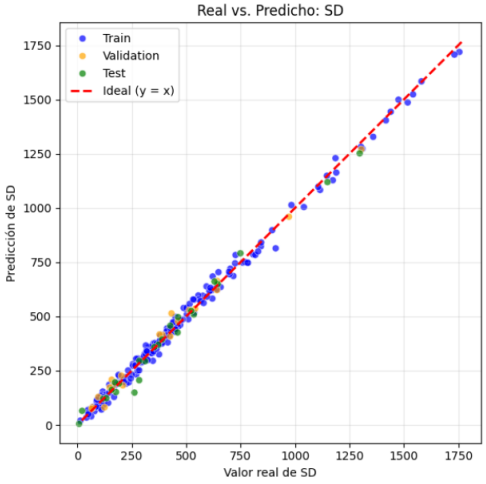
La tabla 7.10 presenta los resultados de los modelos de regresión entrenados utilizando las características latentes extraídas del codificador del AutoEncoder 3D junto con estadísticas vóxel de las segmentaciones. Para la predicción de días de supervivencia (SD), la regresión Random Forest sin un aumento de datos logra el mejor rendimiento, seguido Gradient Boosting con aumento de datos y la regresión Random Forest con aumento de datos.



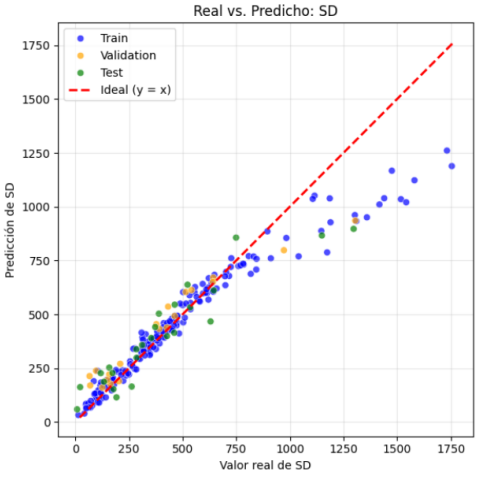
(a) Modelo SVR lineal sin DA.



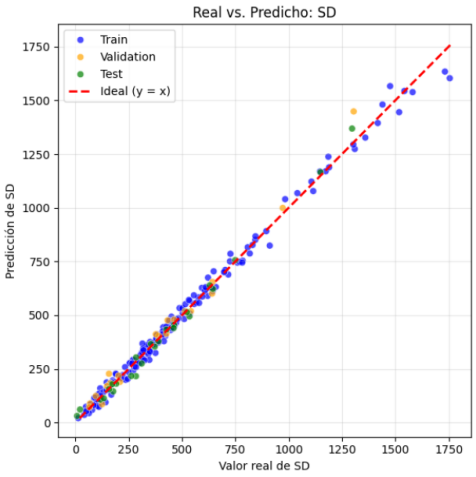
(b) Modelo CatBoost sin DA.



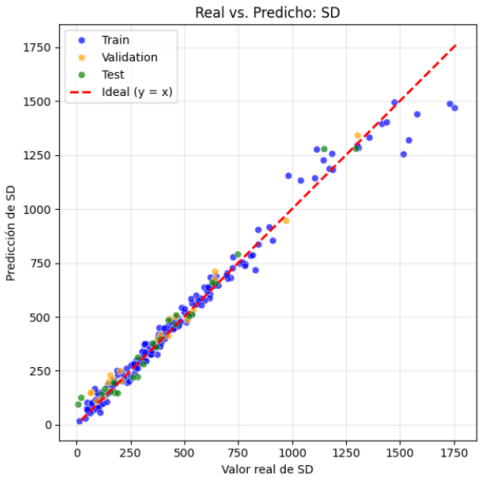
(c) Modelo Ridge sin características polinómicas sin DA.



(d) Modelo SVR-RBF sin DA.

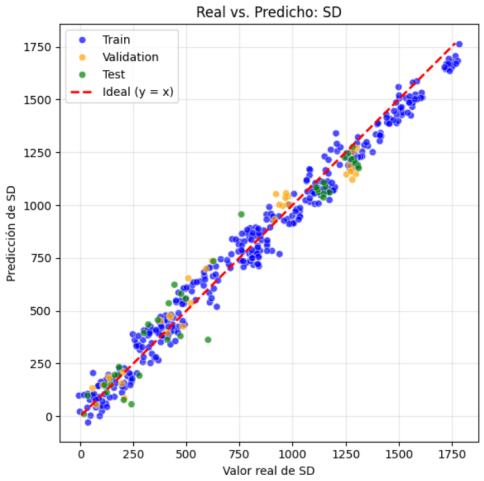


(e) Modelo Random Forest sin DA.

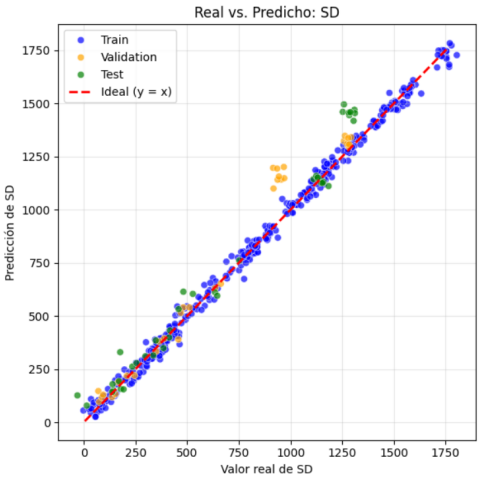


(f) Modelo Gradient Boosting sin DA.

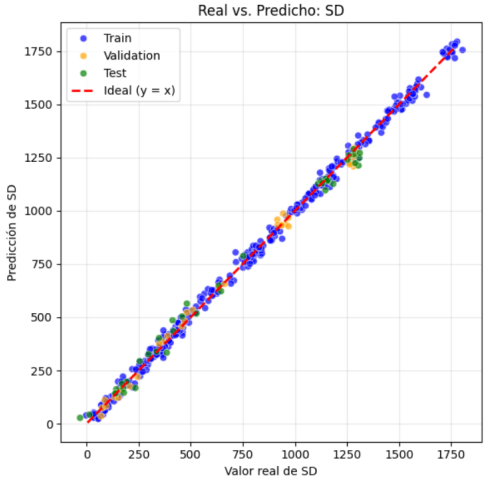
Figura 7.11: Comparación de resultados de los mejores modelos de regresión sobre SD, entrenados con las características latentes del 3D AutoEncoder y las estadísticas vóxel de las segmentaciones, sin aplicar aumento de datos.



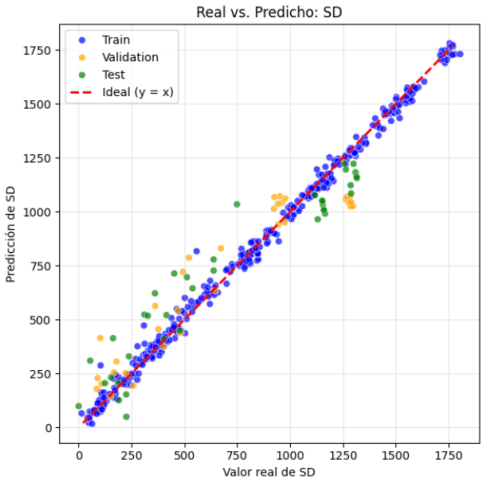
(a) Modelo SVR lineal con DA.



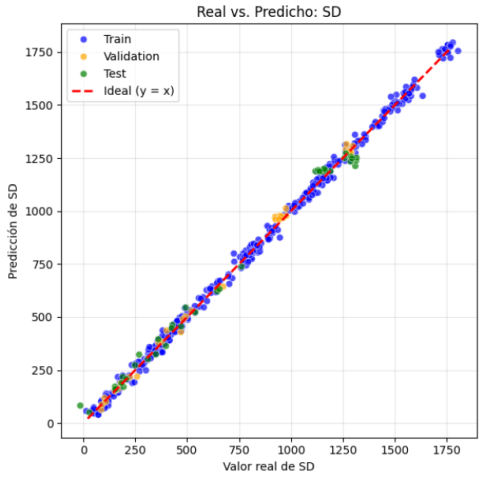
(b) Modelo CatBoost con DA.



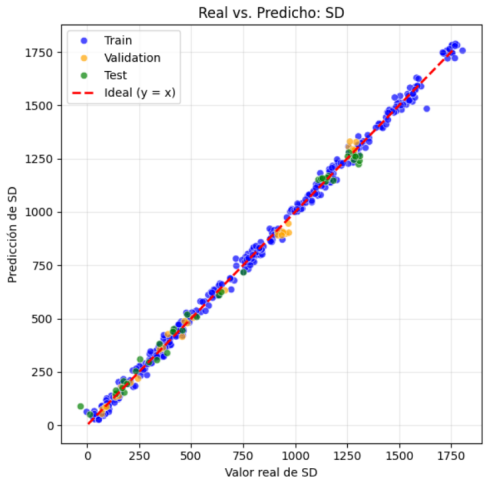
(c) Modelo Ridge sin características polinómicas con DA.



(d) Modelo SVR-RBF con DA.

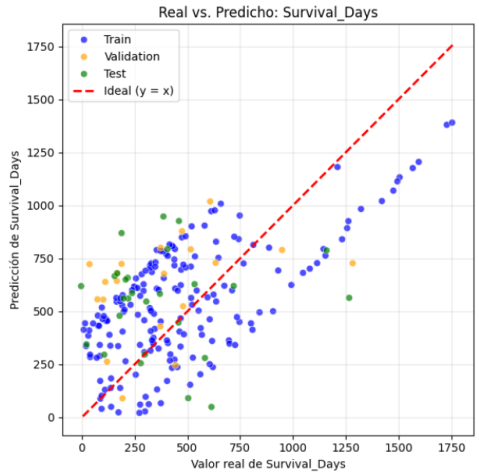


(e) Modelo Random Forest con DA.

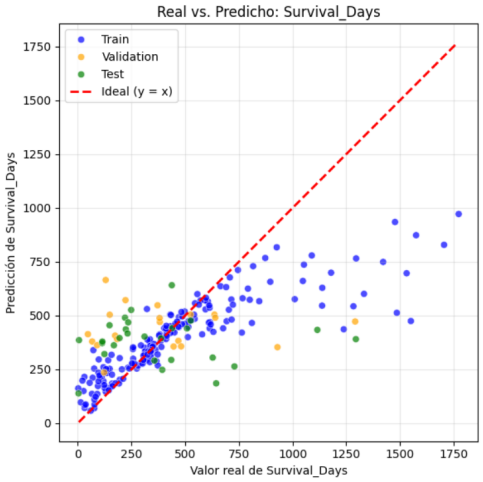


(f) Modelo Gradient Boosting con DA.

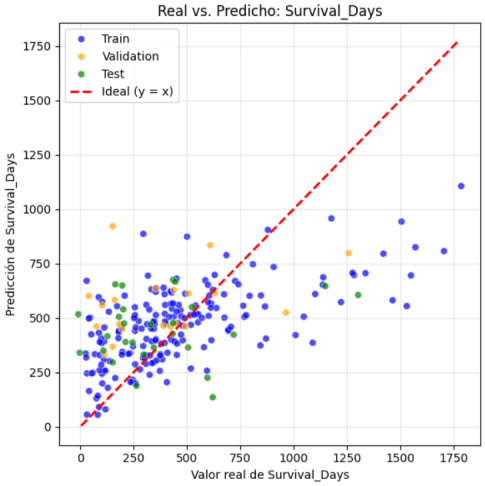
Figura 7.12: Comparación de resultados de los mejores modelos de regresión sobre SD, entrenados con las características latentes del 3D AutoEncoder y las estadísticas vóxel de las segmentaciones, aplicando aumento de datos.



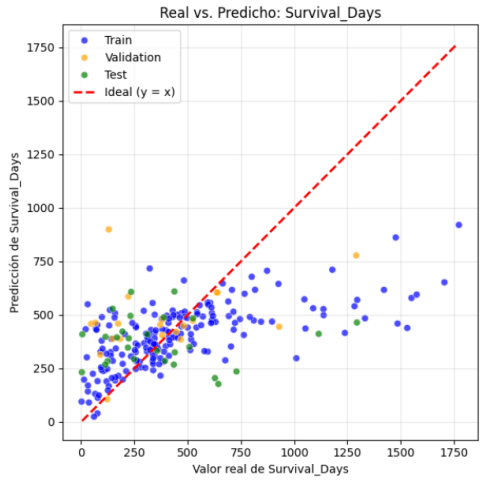
(a) Modelo SVR lineal sin DA.



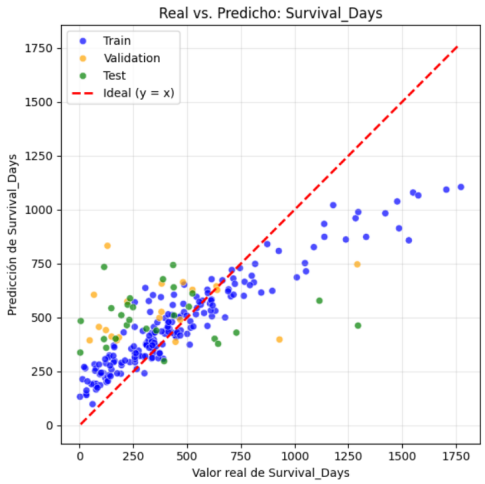
(b) Modelo CatBoost sin DA.



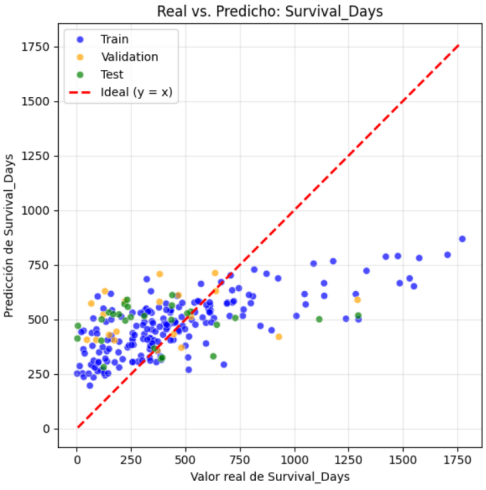
(c) Modelo Ridge sin DA.



(d) Modelo SVR-kernel RBF sin DA.



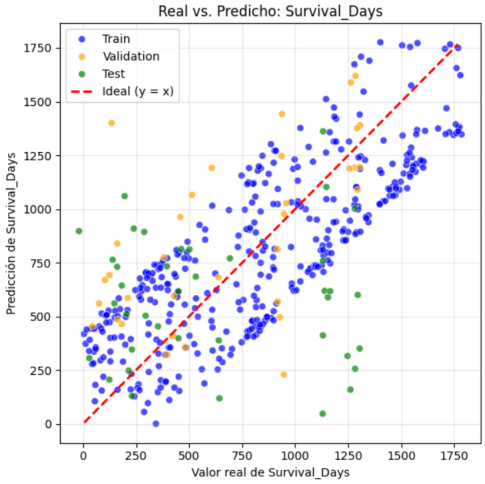
(e) Modelo Random Forest sin DA.



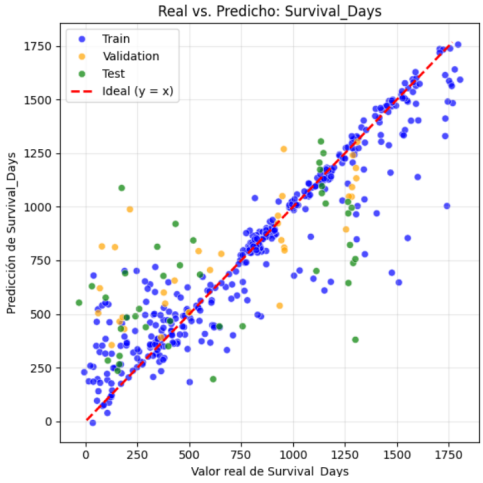
(f) Modelo Gradient Boosting sin DA.

Figura 7.13: Comparación de resultados de los mejores modelos de regresión sobre SD, entrenados con el y4 del LViT sin aplicar aumento de datos.

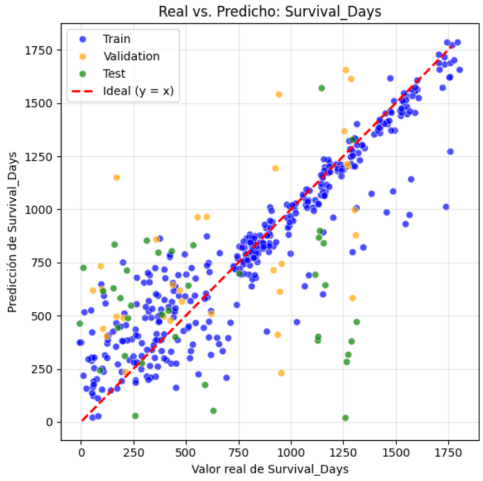
La tabla 7.11 muestra los resultados para modelos entrenados con las representaciones y4 del LViT, revelando un rendimiento considerablemente inferior comparado con las



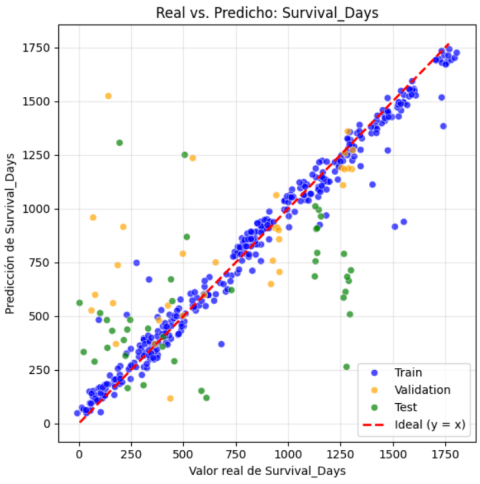
(a) Modelo SVR lineal con DA.



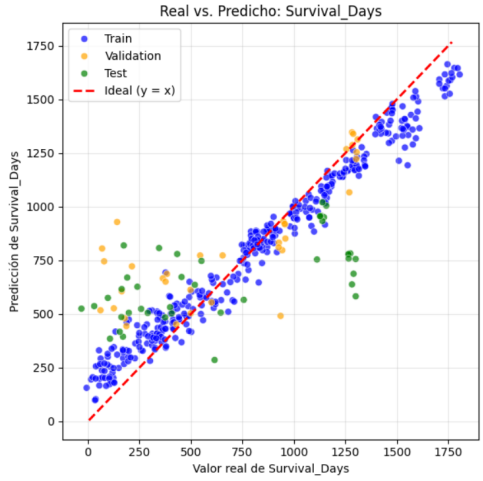
(b) Modelo CatBoost con DA.



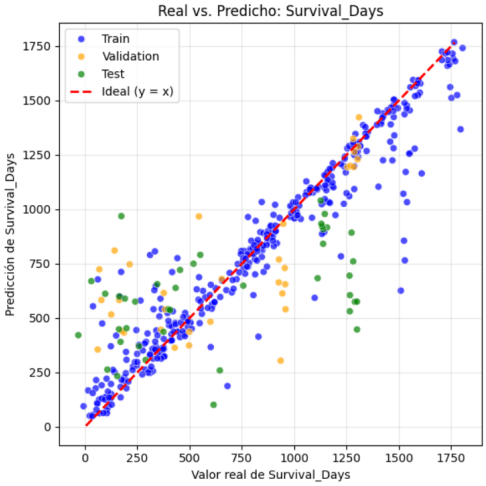
(c) Modelo Ridge con DA.



(d) Modelo SVR-RBF con DA.



(e) Modelo Random Forest con DA.



(f) Modelo Gradient Boosting con DA.

Figura 7.14: Comparación de resultados de los mejores modelos de regresión sobre SD, entrenados con el y4 del LViT aplicando aumento de datos.

características del AutoEncoder 3D. Catboost sin aumento de datos consigue las mejores predicciones, seguido del SVR con kernel RBF sin aumento de datos. En la tabla se puede

observar que aplicar aumento de datos empeora el rendimiento de los modelos. Luego, para la tabla 7.12, presenta los resultados de las cabezas de regresión integradas directamente en el modelo LViT, mostrando el rendimiento más variable, con LViT Head 2 obteniendo el mejor resultado.

Tabla 7.10: Errores MAE para la predicción de días de supervivencia (SD) en el conjunto de test, para los modelos de regresión entrenados con las características latentes sacadas del codificador del 3D AutoEncoder junto con estadísticas vóxel de las segmentaciones.

Modelo	**DA**	**MAE SD**
Gradient Boosting	–	0.0165
MLP Regressor	–	0.1770
Random Forest Regression	–	0.0060
Ridge+Características polinómicas	–	0.2116
Ridge sin características polinómicas	–	0.0108
SVR con Kernel RBF	–	0.0537
SVR lineal	–	0.0461
CatBoost	–	0.0192
Gradient Boosting	✓	0.0079
MLP Regressor	✓	0.1620
Random Forest Regression	✓	0.0081
Ridge+Características polinómicas	✓	0.1814
Ridge sin características polinómicas	✓	0.0174
SVR con Kernel RBF	✓	0.0762
SVR lineal	✓	0.0495
CatBoost	✓	0.0354

Tabla 7.11: Errores MAE para la predicción de días de supervivencia (SD) en el conjunto de test, para los modelos de regresión entrenados con el y4 del LViT.

Modelo	**DA**	**MAE SD**
Gradient Boosting	–	0.1499
MLP Regressor	–	0.3433
Random Forest Regression	–	0.1528
Ridge+Características polinómicas	–	0.1978
Ridge sin características polinómicas	–	0.1723
SVR con Kernel RBF	–	0.1343
SVR lineal	–	0.2091
CatBoost	–	0.1270
Gradient Boosting	✓	0.1967
MLP Regressor	✓	0.2564
Random Forest Regression	✓	0.1889
Ridge+Características polinómicas	✓	0.2707
Ridge sin características polinómicas	✓	0.2717
SVR con Kernel RBF	✓	0.2043
SVR lineal	✓	0.2874
CatBoost	✓	0.1725

Tabla 7.12: Errores MAE para la predicción de días de supervivencia (SD) en el conjunto de test, para las cabezas de regresión integradas en el modelo LViT.

Modelo	**Capas Ocultas**	**Dropout**	**MAE SD**
LViT Head 1	0	–	0.289
LViT Head 2	1 (128 neuronas)	–	0.265
LViT Head 3	1 (128 neuronas)	50 %	1.452
LViT Head 4	3 (256, 128, 64 neuronas)	30 %, 20 %	0.559

7.2.3. Comparación entre modalidades y fusiones

El análisis comparativo entre las diferentes representaciones revela que las características latentes del AutoEncoder 3D proporcionan consistentemente el mejor rendimiento predictivo, superando significativamente tanto a las representaciones del codificador ViT como a las cabezas de regresión integradas del LViT. Esta superioridad sugiere que las representaciones aprendidas por el AutoEncoder 3D capturan información más relevante para las tareas de regresión evaluadas, posiblemente debido a su entrenamiento específico en la reconstrucción de volúmenes médicos tridimensionales.

En términos de modalidades unimodales versus multimodales, los modelos entrenados con características del AutoEncoder 3D combinadas con estadísticas vóxel de segmentaciones demuestran que la fusión de información multimodal es beneficiosa, puesto que mejora la capacidad predictiva comparada con el uso exclusivo de características de imagen. Sin embargo, la fusión no siempre garantiza mejores resultados, como evidencian los rendimientos inferiores de las representaciones del LViT, sugiriendo que la calidad de las representaciones individuales es más crítica que la simple agregación de modalidades.

El impacto del aumento de datos muestra patrones variables según el algoritmo y el tipo de representación. Para las características del AutoEncoder 3D, algunos modelos como Gradient Boosting mejoran significativamente con aumento de datos, mientras que Random Forest muestra resultados mixtos. En contraste, para las representaciones del LViT, el aumento de datos muestra un efecto predominantemente negativo, sugiriendo que estas representaciones son más sensibles a las transformaciones aplicadas durante el aumento de datos.

7.2.4. Discusión sobre viabilidad clínica

Desde una perspectiva de viabilidad clínica, los resultados obtenidos con las características del AutoEncoder 3D presentan niveles de precisión prometedores para su potencial implementación en entornos clínicos. Los errores MAE de 0.0060 para días de supervivencia, cuando se normalizan, representan desviaciones relativamente pequeñas que podrían ser clínicamente aceptables para tareas de apoyo al diagnóstico y planificación terapéutica. Sin embargo, la traducción de estos valores normalizados a escalas clínicas reales requiere una evaluación cuidadosa de su significado práctico.

La fiabilidad de las predicciones varía considerablemente según el tipo de modelo y representación utilizada. Los modelos basados en características del AutoEncoder 3D demuestran mayor consistencia y estabilidad en sus predicciones, lo que es crucial para aplicaciones clínicas donde la confiabilidad es crucial. En contraste, las representaciones del LViT muestran mayor variabilidad, con algunos modelos presentando errores significativamente altos que limitarían su utilidad clínica inmediata.

La implementación práctica de estos sistemas requiere consideraciones adicionales sobre la interpretabilidad y explicabilidad de las predicciones. Los modelos de Random Forest y Gradient Boosting ofrecen mayor transparencia en sus decisiones comparados con las representaciones de aprendizaje profundo, lo que podría facilitar su aceptación en entornos clínicos. Además, la robustez ante variaciones en los datos de entrada, evidenciada por los diferentes comportamientos con aumento de datos, sugiere la necesidad de validación exhaustiva en poblaciones diversas antes de su despliegue clínico.

7.3. Comparación global de modelos y tareas

El análisis integral de los resultados obtenidos en este trabajo permite establecer una perspectiva comparativa entre las diferentes aproximaciones implementadas, abarcando tanto los modelos de segmentación como los de regresión. Se consideran el rendimiento individual de cada arquitectura, su capacidad de integración en sistemas multitarea y su aplicabilidad en escenarios clínicos reales. Esta comparación global es esencial para comprender las fortalezas y limitaciones de cada enfoque, así como para identificar las configuraciones más prometedoras en la práctica médica. La evaluación incluye desde modelos especializados en tareas únicas (como la segmentación precisa de tumores o la predicción de supervivencia) hasta arquitecturas híbridas capaces de abordar simultáneamente ambos objetivos.

Se implementaron varios algoritmos diferentes de aprendizaje automático, que abarcan tanto modelos de segmentación (por ejemplo, 3D U-Net, SegResNet, LViT) como de regresión (incluyendo métodos convencionales y redes integradas con visión y texto clínico). Estas arquitecturas se combinaron con diversas estrategias de fusión multimodal y técnicas de optimización de hiperparámetros, generando un ecosistema experimental robusto y variado, lo que permite comparar de manera sistemática el comportamiento de cada modelo en tareas de segmentación y regresión, así como su interacción en sistemas multitarea.

7.3.1. Segmentación vs regresión: modelos multitarea

Los resultados experimentales muestran diferencias claras entre los modelos especializados y los multitarea, especialmente al analizar el rendimiento en segmentación y regresión clínica.

En el caso de la segmentación, modelos dedicados como U-Net 3D y SegResNet alcanzaron métricas superiores (Dice hasta 0.85 para WT y 0.78 para TC), logrando una segmentación precisa de los detalles anatómicos y las fronteras tumorales, incluso en regiones complejas como la transición entre edema y tejido sano. Este enfoque especializado permite que toda la capacidad de aprendizaje se centre en optimizar la segmentación, lo que se traduce en resultados más robustos y coherentes.

Por otro lado, los modelos multitarea como LViT, que integran segmentación y predicción de supervivencia, ofrecen una aproximación más equilibrada pero con compromisos. Aunque la segmentación se mantiene en niveles competitivos (Dice promedio de 0.76 para WT), la integración de una cabeza de regresión en la arquitectura original de LViT no produjo buenos resultados en la predicción de supervivencia. Los errores MAE obtenidos por la cabeza de regresión integrada oscilaron entre 0.265 y 0.289, pero estos valores no superaron a los alcanzados por modelos de regresión clásicos entrenados con características latentes del codificador del autoencoder 3D y estadísticas voxel de las segmentaciones.

Asimismo, los modelos de regresión entrenados con las características del bloque y4 del ViT de LViT tampoco lograron igualar el desempeño de los enfoques clásicos. En contraste, los modelos de regresión tradicionales, que combinan las representaciones latentes del autoencoder 3D con estadísticas voxel de las segmentaciones, ofrecieron una mayor precisión y una mejor capacidad de generalización en la predicción de días de supervivencia. Esto sugiere que, aunque la LViT original es efectiva en segmentación, su extensión directa a la regresión clínica (ya sea mediante una cabeza de regresión o usando embeddings intermedios del ViT) no resulta tan competitiva como las estrategias basadas en codificadores diseñados específicamente para extracción de características latentes relevantes.

7.3.2. Rendimiento de modelos multimodales

La integración de información multimodal fue un factor clave para mejorar el rendimiento tanto en segmentación como en regresión clínica. En la etapa de segmentación, los modelos capaces de procesar múltiples modalidades de imagen (como T1, T1ce, T2 y FLAIR) mostraron una ventaja significativa frente a los enfoques unimodales. Modelos como 3D U-Net y SegResNet aprovecharon la riqueza de información que aporta cada modalidad, logrando una mayor coherencia morfológica y precisión anatómica en todas las clases tumorales. En particular, 3D U-Net destacó globalmente por su robustez, mientras que SegResNet obtuvo resultados especialmente buenos en la segmentación del tumor realzado (ET). Por su parte, la arquitectura original de LViT demostró un rendimiento competitivo en la segmentación del tumor completo (WT), aunque su diseño 2D y enfoque unimodal limitaron su aplicabilidad en tareas multiclase o volumétricas.

En el ámbito de la regresión clínica, la estrategia más efectiva consistió en entrenar modelos clásicos (como regresión lineal, Random Forest o Gradient Boosting) utilizando una fusión de características latentes extraídas del codificador del AutoEncoder 3D junto con estadísticas vóxel derivadas de las segmentaciones. Esta combinación permitió a los modelos captar tanto patrones complejos presentes en las imágenes como información morfológica y volumétrica relevante para el pronóstico clínico, reflejándose en mejoras claras en la predicción de supervivencia. Es importante destacar que, aunque estos modelos no son multimodales en sentido estricto, sí aprovechan la diversidad de información generada por arquitecturas multimodales de segmentación.

No obstante, no todas las aproximaciones multimodales ofrecieron el mismo nivel de rendimiento en regresión. Los modelos basados en representaciones del codificador ViT de LViT, aunque útiles para segmentación, no lograron capturar de forma adecuada los patrones necesarios para la predicción clínica, mostrando un desempeño inferior frente a los modelos que combinaban AutoEncoder 3D y estadísticas vóxel. Además, la variante de LViT con cabeza de regresión integrada tampoco obtuvo buenos resultados en la predicción clínica, confirmando que, pese a su eficacia en segmentación, esta arquitectura en concreto no es tan competitiva para tareas de regresión como los enfoques clásicos apoyados en representaciones latentes especializadas y estadísticas morfológicas, tal y como se evidenció en las secciones anteriores.

El impacto de las estrategias de aumento de datos también fue relevante, pero dependió de la arquitectura y la variable objetivo. Algoritmos como Gradient Boosting se beneficiaron de manera consistente del aumento de datos, mejorando la precisión en la predicción de supervivencia, mientras que otros, como Random Forest, mostraron una respuesta más variable. Esto resalta la importancia de adaptar las estrategias de aumento

de datos y la integración multimodal a las características específicas de cada modelo y tarea para maximizar su rendimiento.

7.3.3. Consideraciones sobre eficiencia computacional

El análisis de eficiencia computacional pone de manifiesto diferencias notables entre las arquitecturas evaluadas, lo que repercute directamente en su viabilidad para entornos clínicos reales. Los modelos de segmentación 3D, como U-Net y SegResNet, requieren recursos computacionales elevados debido al procesamiento volumétrico completo. En una GPU RTX 2060 SUPER, estos modelos presentan tiempos de entrenamiento de 2 a 4 horas por época, siendo SegResNet especialmente exigente, con un tiempo de entrenamiento significativamente mayor que el resto, lo que limita su aplicabilidad en escenarios con recursos restringidos.

Por otro lado, los modelos de regresión basados en características extraídas destacan por su eficiencia: completan el entrenamiento en cuestión de minutos y permiten una experimentación iterativa ágil, aspecto crucial cuando se dispone de hardware limitado o se requiere ajustar modelos rápidamente. Durante la inferencia, los modelos volumétricos necesitan entre 6 y 8 GB de VRAM para procesar volúmenes completos, aunque este requisito puede reducirse mediante estrategias como el procesamiento por parches o la disminución de la resolución. En cambio, los modelos de regresión, al trabajar sobre características de baja dimensionalidad, apenas requieren memoria (< 1 GB), lo que facilita su integración en sistemas con recursos limitados y permite adaptar fácilmente la arquitectura al entorno de despliegue.

Los modelos multimodales, como LViT, representan un compromiso entre rendimiento y eficiencia. Su entrenamiento demanda aproximadamente un 50 % más de tiempo que U-Net 3D, debido a la complejidad añadida por el procesamiento de texto y los mecanismos de atención. Sin embargo, su capacidad multitarea puede compensar este coste, ya que evita la necesidad de entrenar modelos independientes para cada tarea. Por su parte, el AutoEncoder 3D requiere un entrenamiento inicial más largo (6-8 horas), pero una vez entrenado, la extracción de características es muy eficiente, permitiendo entrenar múltiples modelos de regresión con un coste computacional mínimo.

En cuanto a la optimización de hiperparámetros mediante búsqueda aleatoria con validación cruzada, resultó ser una estrategia eficiente en todos los modelos de regresión, completándose en pocos minutos con 15-20 iteraciones por modelo y proporcionando configuraciones competitivas sin el coste de búsquedas exhaustivas.

Capítulo 8

Conclusiones y trabajo futuro

Elaborar este trabajo resultó ser una ocasión atractiva para estudiar, proyectar y poner en marcha un sistema de aprendizaje profundo orientado al análisis multimodal de imágenes del ámbito médico. Durante el desarrollo, se logró implementar una solución capaz de segmentar automáticamente las zonas tumorales en resonancias magnéticas cerebrales, integrando además información clínica organizada. Cabe destacar que, en el caso de los modelos de regresión, se exploró la fusión de datos multimodales a nivel de entrada, lo que permitió combinar distintas fuentes de información y, en algunos casos, mejorar el rendimiento predictivo. Esta perspectiva multimodal constituye un progreso considerable en el terreno de la IA aplicada a la medicina [Song, 2025], ya que el ensamblaje de diversas fuentes informativas puede facilitar una comprensión más global y exacta de los casos clínicos.

La dificultad propia del manejo de datos médicos multimodales ha necesitado la formulación de metodologías concretas que traten tanto los elementos técnicos como los clínicos del problema [Wang et al., 2020]. A lo largo de este proceso, se han examinado numerosas arquitecturas de redes neuronales profundas, desde modelos clásicos de segmentación hasta planteamientos más novedosos que añaden sistemas de atención y fusión de modalidades [Xu et al., 2024]. La práctica acumulada no solo ha hecho posible lograr las metas técnicas propuestas, sino que también ha brindado valiosas enseñanzas sobre las restricciones y posibilidades que muestra la aplicación de IA en entornos médicos verdaderos.

8.1. Síntesis de resultados y cumplimiento de objetivos

En líneas generales, los propósitos que se marcaron al comienzo del proyecto se han alcanzado de forma bastante exitosa. Se pudo crear un sistema multimodal que puede analizar imágenes de resonancia magnética del cerebro para delimitar los tumores cerebrales y, en el caso de la LViT, junto con datos clínicos organizados. Las arquitecturas que mejor rendimiento obtuvieron en la tarea de segmentación multimodal (diferentes modalidades de entrada) fueron 3D U-Net [Golts et al., 2021] y SegResNet [Zhu et al., 2023], mientras que la red LViT [Li et al., 2022] ofreció los mejores resultados en el enfoque de fusión de imágenes con texto clínico estructurado.

En el ámbito de la regresión para predicción de supervivencia, se identificaron los modelos más efectivos: Gradient Boosting, Random Forest Regressor y SVR con kernel

RBF, entrenados sobre representaciones latentes extraídas del modelo LViT y del 3D AutoEncoder junto con estadísticas vóxel [Soltaninejad et al., 2021]. Los mejores resultados se obtuvieron entrenando estos modelos con representaciones del 3D AutoEncoder, especialmente cuando se aplicó aumento de datos mediante ruido gaussiano sobre las características latentes de pacientes con días de supervivencia elevados. Esta técnica de aumento de datos demostró ser particularmente efectiva para mejorar la capacidad predictiva en el rango de supervivencia más alto, donde tradicionalmente los modelos presentan mayor dificultad debido a la menor representación de estos casos en el conjunto de datos.

Por otro lado, el análisis comparativo de diferentes arquitecturas permitió identificar patrones específicos de rendimiento según el tipo de tarea y modalidad de datos, proporcionando una base sólida para la selección de arquitecturas en futuros proyectos de análisis multimodal en el dominio médico.

8.2. Impacto y relevancia práctica

El sistema que se ha creado prueba que es factible fusionar distintas fuentes de información médica para refinar el análisis computarizado de tumores en el cerebro [Song, 2025]. El poder descubrir propiedades ocultas desde las partes más internas del modelo y luego usarlas para alimentar modelos de regresión de siempre es un aporte metódico importante, ya que permite ver la parte visual y entender mejor el peso de cada fuente. Esta manera, mezcla lo bueno del aprendizaje profundo al tratar imágenes, con lo fácil de entender y lo fuerte que son los métodos clásicos de aprendizaje automático [Xu et al., 2024].

Que se hayan podido usar correctamente las técnicas para mezclar información de varias fuentes en medicina abre puertas a usar este sistema en hospitales, donde fusionar datos visuales y clínicos puede dar diagnósticos más afinados y pronósticos más seguros. Lo que se hizo podría ser el cimiento de herramientas para ayudar a diagnosticar en hospitales, haciendo más fácil decidir al automatizar tareas repetitivas y apoyando el creciente mundo de la medicina hecha a la medida. En concreto, el sistema podría ser como una segunda opinión para radiólogos y oncólogos, dando análisis numéricos objetivos que ayuden a la evaluación clínica tradicional. Poder analizar muchos datos rápidamente, podría reducir bastante los tiempos de diagnóstico, sobre todo en hospitales muy llenos, y así dar más acceso a análisis especializados en zonas con pocos recursos. Además, la metodología desarrollada podría adaptarse fácilmente a otros tipos de tumores y patologías, expandiendo su impacto potencial en múltiples especialidades médicas y contribuyendo a la estandarización de protocolos de diagnóstico por imagen a nivel internacional.

8.3. Lecciones aprendidas

Durante el desarrollo del proyecto, se evidenció la complejidad inherente al manejo de información médica procedente de múltiples fuentes, lo que exige una atención especial a la calidad y completitud de los datos, en particular de la información clínica relevante[Wang et al., 2020]. Se comprendió que los datos médicos requieren un tratamiento diferenciado, ya que suelen presentar valores faltantes y provienen de contextos muy diversos, lo que implica no solo diseñar estrategias robustas para la gestión y preprocesamiento de la información, sino también garantizar que los sistemas de IA sean transparentes y fiables, de modo que los profesionales sanitarios puedan interpretar y confiar en sus resultados.

Se observó que las estrategias habituales para rellenar datos faltantes, como la imputación por la media, pueden inducir errores considerables si se aplican a variables clínicas relevantes, o que siguen una distribución en concreto, por lo que es necesario emplear métodos más robustos y eficientes para gestionar la información incompleta. En este sentido, el aumento de datos mediante la aplicación de ruido gaussiano a las características latentes demostró ser especialmente eficaz para mejorar la robustez de los modelos de regresión, sobre todo en los casos de supervivencia elevada, donde la escasez de datos es más acusada.

No obstante, se evidenció que no todas las entradas que integran varios tipos de datos son igualmente útiles para entrenar modelos de regresión. Por ejemplo, el uso de características latentes como el y4 del codificador ViT de LViT no ofreció buenos resultados en este contexto, lo que pone de manifiesto que la calidad y relevancia de las características extraídas es tan importante como la cantidad o diversidad de datos integrados. Por tanto, la selección cuidadosa de las representaciones y fuentes de información resulta clave para el éxito de los modelos predictivos en el ámbito médico.

Por otro lado, se observó que es muy importante que trabajen juntas personas de diferentes áreas, combinando lo que se sabe de tecnología con la experiencia de los médicos para crear soluciones que realmente sirvan y se puedan usar en el mundo real. Para saber si algo funciona, no basta con mirar números; también hay que preguntar a los médicos si lo que predice el sistema es útil en la práctica, resaltando la necesidad de tener a los profesionales médicos participando en todo el proceso de creación del sistema.

8.4. Limitaciones del trabajo actual

A pesar de los logros alcanzados, el proyecto presenta limitaciones que deben ser reconocidas. Las cabezas de regresión, aunque ofrecieron pérdidas bajas durante el entrenamiento y métricas agregadas satisfactorias, mostraron una correspondencia deficiente entre valores reales y predichos en las gráficas de dispersión, tendiendo a predecir valores centrados en la media del conjunto. Esta limitación evidenció que las métricas tradicionales de evaluación pueden ser engañosas cuando se trata de problemas de regresión con alta variabilidad, donde la capacidad del modelo para capturar la dispersión real de los datos es tan importante como minimizar el error promedio.

La inclusión de muestras con datos clínicos incompletos durante el entrenamiento, donde los valores ausentes se imputaron con la media, afectó negativamente la distribución de los datos y condujo a una función de predicción excesivamente conservadora. Este enfoque de imputación, aunque común en la práctica, demostró ser inadecuado para variables clínicas críticas como los días de supervivencia, donde la variabilidad natural es un componente esencial de la información. Aunque el aumento de datos mediante ruido gaussiano aplicado a las características latentes mostró mejoras en casos específicos de supervivencia elevada, esta técnica no logró resolver completamente el problema de la tendencia del modelo hacia predicciones conservadoras en algunos casos, como es el caso de los modelos de regresión entrenados con el y4 del codificador ViT de la red LViT.

Los experimentos iniciales con redes unimodales y el modelo SAM2 no lograron resultados satisfactorios, mientras que la red UNETR mostró sobreajuste significativo, adaptándose bien al conjunto de entrenamiento pero fallando en la generalización al conjunto de validación. El modelo SAM2, a pesar de utilizar tanto los pesos preentrenados originales como los especializados de MedSAM2, presentó dificultades para aprender efectivamente durante el ajuste fino, con métricas inestables que no reflejaban mejoras reales en la seg-

mentación. Otra limitación significativa fue la falta de diversidad en términos del propio conjunto de datos, también sobre protocolos de imagen y poblaciones de pacientes, lo que puede haber restringido la capacidad de generalización de los modelos desarrollados [Xu et al., 2024].

8.5. Trabajo futuro

En cuanto a las líneas de investigación futuras, una dirección prometedora sería adaptar la arquitectura LViT para que pueda abordar de manera simultánea los tres tipos de segmentación (WT, TC y ET), superando así la limitación actual de segmentar solo una clase por ejecución. Esto permitiría comparar su rendimiento con arquitecturas ya consolidadas en segmentación multiclase y aprovechar mejor su capacidad multimodal.

Asimismo, sería interesante explorar la incorporación de una cabeza de regresión directamente en arquitecturas como 3D U-Net o SegResNet, lo que permitiría evaluar si estos modelos, además de su eficacia en segmentación, pueden predecir variables clínicas relevantes como la supervivencia, lo que podría abrir nuevas vías para el desarrollo de modelos multitarea que integren segmentación y predicción clínica en una sola arquitectura, facilitando así el flujo de trabajo y la interpretación de resultados, puesto que, en este caso se intentó realizar, pero no dió buenos resultados.

Por otro lado, enriquecer los datos disponibles para cada paciente mediante la incorporación de información sobre tratamientos previos, datos genéticos y variables fisiológicas ampliaría el contexto clínico y podría mejorar la precisión de las predicciones. El uso de mapas derivados de radiografías o la integración directa de descripciones textuales de los informes médicos también aportaría valor añadido al análisis clínico, siguiendo las tendencias actuales de fusión multimodal en el ámbito médico[Li et al., 2022, Adeel et al., 2022].

En el campo de la regresión para predicción de supervivencia, es fundamental avanzar hacia arquitecturas más sofisticadas que superen las limitaciones de las cabezas de regresión actuales. La investigación en modelos probabilísticos que reflejen mejor la incertidumbre inherente a la predicción clínica, junto con técnicas avanzadas de regularización, puede ayudar a evitar la tendencia hacia predicciones conservadoras. Además, refinar las estrategias de aumento de datos más allá del ruido gaussiano, incorporando técnicas de síntesis de datos sintéticos y métodos de balanceado para casos extremos, podría mejorar significativamente el rendimiento predictivo.

El uso de modelos de mayor tamaño entrenados con grandes volúmenes de datos médicos se perfila como una vía para incrementar la precisión y utilidad de los sistemas, especialmente a la luz de los avances recientes que han demostrado mejoras sustanciales en tareas médicas[Rane, 2023]. Sin embargo, resulta imprescindible validar estos modelos en entornos clínicos reales, asegurando que su utilidad trascienda el rendimiento técnico y se adapte a las necesidades diarias de los profesionales sanitarios. En este sentido, la adopción de técnicas de aprendizaje federado permitiría entrenar modelos con datos de múltiples hospitales sin comprometer la privacidad de los pacientes, ampliando la diversidad y la robustez de los datos empleados[Guan et al., 2024].

La exploración de arquitecturas emergentes, como las redes neuronales de grafos (GNNs) [D2iCE, 2025] y Transformers especializados para el dominio médico, podría aportar nuevas capacidades para modelar relaciones complejas entre modalidades y mejorar la integración de información heterogénea. El desarrollo de modelos más compactos y eficientes, mediante técnicas de destilación de conocimiento y compresión[Qin et al., 2021], facilitaría su implantación en entornos clínicos con recursos limitados. Además, avan-

zar en métodos de explicabilidad específicos para modelos multimodales médicos será esencial para aumentar la confianza clínica y cumplir con los requisitos regulatorios. Finalmente, la incorporación de técnicas de imagen molecular avanzada, como PET/TC digital[of Geneva, 2024], podría enriquecer el análisis multimodal al aportar información morfológica, funcional y molecular simultáneamente.

8.6. Reflexiones finales

Este trabajo proporcionó una experiencia completa al crear sistemas de IA aplicados en el campo médico. Los desafíos encontrados y las soluciones implementadas han contribuido significativamente al conocimiento sobre el manejo multimodal de datos médicos, estableciendo bases sólidas para futuras investigaciones en este ámbito. La experiencia demostró que crear sistemas de IA médica requiere no solo competencia técnica, sino también comprender profundamente las necesidades clínicas y las limitaciones prácticas del entorno sanitario. La investigación reveló la complejidad inherente de trasladar avances teóricos en IA a aplicaciones prácticas en medicina, donde aspectos como la variabilidad de los datos, la interpretabilidad de los resultados y la integración en flujos de trabajo existentes adquieren importancia crítica.

El proyecto demostró que, aunque existen limitaciones técnicas y metodológicas, el enfoque multimodal para analizar imágenes médicas presenta un potencial considerable para mejorar el diagnóstico y pronóstico en oncología cerebral [Song, 2025]. La experiencia adquirida subraya la importancia de un enfoque que combine conocimientos técnicos con comprensión clínica para desarrollar soluciones verdaderamente efectivas en la salud, confirmando el enorme potencial que existe para transformar la práctica clínica mediante estas tecnologías emergentes aplicadas con cuidado y rigor. La investigación evidenció que el éxito de estos sistemas no se mide únicamente por métricas técnicas, sino por su capacidad real de asistir a los profesionales médicos en la toma de decisiones clínicas y, finalmente, mejorar los resultados para los pacientes.

La reflexión sobre este desarrollo tecnológico en el contexto médico también evidenció la necesidad de considerar aspectos éticos y regulatorios desde las primeras fases. La responsabilidad de desarrollar herramientas que influyan en decisiones médicas críticas requiere un compromiso continuo con la transparencia, la validación rigurosa y la colaboración estrecha con la comunidad médica. El trabajo realizado constituye un paso importante hacia la implementación de herramientas de IA que asistan de manera efectiva y segura a los profesionales sanitarios, contribuyendo a la evolución hacia una medicina más precisa, personalizada y accesible. Esta experiencia confirma que el futuro de la IA en medicina depende no solo de los avances tecnológicos, sino también de la capacidad para integrar estas tecnologías de forma responsable y eficaz en el complejo ecosistema de la atención sanitaria.

Apéndice A

Competencias específicas cubiertas

El desarrollo de este trabajo ha permitido aplicar de forma práctica diversas competencias específicas definidas en la Memoria del Plan de Estudios del Grado en Ciencia e Ingeniería de Datos de la Universidad de Las Palmas de Gran Canaria. A continuación, se detallan aquellas competencias directamente relacionadas con las actividades realizadas durante el proyecto:

- **CE1: Capacidad para adquirir, integrar y procesar datos procedentes de distintas fuentes.**
  Esta competencia ha sido esencial durante la fase de comprensión y preparación de los datos, en la que se integraron imágenes médicas en formato NIfTI y datos clínicos estructurados. También se aplicaron técnicas de normalización y organización de los datos para su uso en modelos de aprendizaje profundo.

- **CE3: Capacidad para aplicar técnicas de análisis exploratorio, preprocesamiento y transformación de datos.**
  Se realizaron tareas de exploración, visualización y transformación de datos clínicos y radiómicos, incluyendo la conversión de formatos, la detección de valores atípicos, la normalización de escalas y el manejo de datos faltantes.

- **CE6: Capacidad para seleccionar e implementar modelos y algoritmos adecuados para resolver problemas concretos de ciencia de datos.**
  A lo largo del proyecto se compararon distintos enfoques de modelado (segmentación y regresión), evaluando arquitecturas unimodales y multimodales, y adaptando configuraciones de entrenamiento para obtener resultados óptimos en el contexto médico.

- **CE8: Capacidad para diseñar y evaluar experimentos de validación y rendimiento de modelos predictivos.**
  Los modelos fueron evaluados utilizando métricas específicas como IoU, Dice, MAE y R^2. Se establecieron experimentos controlados para comparar el rendimiento de distintas arquitecturas y técnicas de fusión de datos.

- **CE9: Capacidad para utilizar herramientas de desarrollo y gestión de proyectos de ciencia de datos.**
  Durante el trabajo se utilizó un entorno completo de desarrollo basado en `Visual Studio Code`, `Jupyter Notebooks`, `Git`, y entornos virtuales gestionados con `Conda`. Se mantuvo además un repositorio con control de versiones que permitió gestionar los avances del proyecto.

- **CE10: Capacidad para comunicar de forma efectiva resultados y metodologías empleadas en proyectos de ciencia de datos.**
  La redacción de este documento, la preparación de visualizaciones interpretables, y la elaboración de una presentación final del trabajo han contribuido al desarrollo de esta competencia.

- **CE11: Capacidad para trabajar de forma autónoma y afrontar proyectos de carácter abierto o de investigación aplicada.**
  Dado que este proyecto se desarrolló de manera individual y con un elevado componente investigativo, se ha ejercitado de forma intensiva esta competencia, desde la toma de decisiones técnicas hasta la planificación y documentación.

Estas competencias han sido fundamentales para la consecución de los objetivos del proyecto, y su desarrollo ha contribuido a consolidar el perfil profesional en el ámbito de la ciencia de datos aplicada a la salud.

Apéndice B

Objetivos de desarrollo sostenible

Este Trabajo de Fin de Grado se alinea estrechamente con el ODS 3 (Salud y Bienestar), ya que tiene como objetivo mejorar el diagnóstico y seguimiento de tumores cerebrales mediante el desarrollo de sistemas basados en inteligencia artificial. También, se relaciona con el ODS 9 (Industria, Innovación e Infraestructuras), al fomentar el uso de tecnologías avanzadas como el aprendizaje profundo y su integración en entornos clínicos reales. En cuanto a los ODS con relación baja, el ODS 4 (Educación de calidad) se ve reflejado de forma indirecta al tratarse de un proyecto formativo que fomenta la aplicación práctica de conocimientos avanzados en ciencia de datos. El ODS 8 (Trabajo decente y crecimiento económico) se relaciona con el proyecto ya que impulsa la innovación tecnológica en el sector sanitario. Por último, el ODS 10 (Reducción de las desigualdades) también se relaciona de forma parcial con el proyecto, ya que propone usar la tecnología de manera que, en un futuro, podría ayudar a que más personas, sin importar su situación, tengan acceso a un diagnóstico médico más justo y accesible dentro del sistema público de salud. La valoración completa del grado de relación del TFG con todos los ODS se presenta en la Tabla B.1.

Tabla B.1: Tabla con la valoración del grado de relación del TFG con los ODS, según una escala de 0 a 3 (no procede, bajo, medio, alto).

ODS	**Nivel de relación**			
	0	**1**	**2**	**3**
1. Fin de la Pobreza	**X**			
2. Hambre cero	**X**			
3. Salud y Bienestar				**X**
4. Educación de calidad		**X**		
5. Igualdad de género	**X**			
6. Agua limpia y saneamiento	**X**			
7. Energía asequible y no contaminante	**X**			
8. Trabajo decente y crecimiento económico		**X**		
9. Industria, Innovación e Infraestructuras				**X**
10. Reducción de las desigualdades		**X**		
11. Ciudades y comunidades sostenibles	**X**			
12. Producción y consumo sostenibles	**X**			
13. Acción por el clima	**X**			
14. Vida submarina	**X**			
15. Vida de ecosistemas terrestres	**X**			
16. Paz, justicia e instituciones sólidas	**X**			
17. Alianzas para lograr objetivos	**X**			

Acrónimos

3D Tridimensional. 2

3D-YOLO You Only Look Once en 3D. 10

AH-Net Anisotropic Hybrid Network. 9

Attention U-Net U-Net con mecanismos de atención. 10

BCBM Brain Cancer Brain Metastases. 16

BCE Binary Cross Entropy. 52

BraTS Brain Tumor Segmentation Challenge. 2

CCE Categorical Cross Entropy. 53

CNN Red Neuronal Convolucional. 9

COLAB Google Colaboratory. 3

CRF Conditional Random Fields. 9

CRISP-DM Cross-Industry Standard Process for Data Mining. 3

DeepMedic Deep Medical Image Computing Network. 9

EPI Exponential Pseudo-label Iteration. 34

ER, en sus siglas en inglés Receptor de Estrógeno. 28

ET, en sus siglas en inglés Tumor Realzado. 11

FFN Feed-Forward Network. 34

FLAIR Fluid Attenuated Inversion Recovery. 1

GAP Global Average Pooling. 34

GMP Global Max Pooling. 34

GPU, en sus siglas en inglés Unidad de Procesamiento Gráfico. 3

GTR, en sus siglas en inglés Resección Total Macroscópica. 24

HER2, en sus siglas en inglés Factor Receptor 2. 28

HighRes3DNet High-Resolution 3D Neural Network. 9

IA Inteligencia Artificial. 3

IoU, en sus siglas en inglés Índice de Superposición de Jaccard. 3

IRM Imagen de Resonancia Magnética (MRI, en sus siglas en inglés). 1

LViT Language meets Vision Transformer. 2

MAE Mean Absolute Error. 3

MHSA Multi-Head Self-Attention. 34

MLP Multi-Layer Perceptron - Perceptrón Multicapa. 16

MONAI Medical Open Network for AI. 19

mpMRI Multiparametric Magnetic Resonance Imaging. 23

MSE Mean Squared Error. 3

NIfTI Neuroimaging Informatics Technology Initiative. 16

nnU-Net no-new-U-Net (U-Net autoajustable). 11

ODS Objetivos de Desarrollo Sostenible. 5, 88

PLAM Pixel-Level Attention Module. 34

PR, en sus siglas en inglés Receptor de Progesterona. 28

RBF, en sus siglas en inglés Función de Base Radial. 14

ReLU Rectified Linear Unit. 32

RM Resonancia Magnética. 10

RMSE Root Mean Square Error. 17

SAM2 Segment Anything Model. 16

SegResNet Segmentation Residual Network. 2

STR, en sus siglas en inglés Resección Subtotal. 24

SVR, en sus siglas en inglés Regresión de Vectores de Soporte. 14

SwinUNet Swin Transformer U-Net. 11

T1 Secuencia de Resonancia Magnética T1. 1

T1c/T1Gd T1 con Contraste Gadolinio. 1

T2 Secuencia de Resonancia Magnética T2. 1

TC Tomografía Computarizada. 7

TransUNet Transformer-based U-Net. 11

U-Net Red neuronal tipo U para segmentación. 2

U-Net 3D U-Net tridimensional. 10

UNETR UNEt TRansformers. 11

V-Net Volumetric Convolutional Neural Network. 11

VAE Variational AutoEncoder. 38

ViT Vision Transformer. 11

VoxResNet Voxelwise Residual Network. 9

WT, en sus siglas en inglés Tumor Completo. 11

Glosario

acumulación de gradientes Técnica que permite simular un mayor tamaño de lote efectivo acumulando gradientes a lo largo de varias iteraciones antes de actualizar los pesos del modelo. 58

análisis de errores Proceso sistemático de examinar y categorizar los errores cometidos por un modelo para identificar patrones y áreas de mejora. 3

aprendizaje conjunto Técnica de machine learning donde múltiples tareas relacionadas se entrenan simultáneamente para mejorar el rendimiento general del sistema mediante el intercambio de información entre tareas. 1

aprendizaje multitarea Técnica de machine learning donde un modelo se entrena simultáneamente para realizar múltiples tareas relacionadas, compartiendo representaciones internas para mejorar la generalización. 9

aumento de datos Técnica que incrementa artificialmente el tamaño del conjunto de entrenamiento aplicando transformaciones como rotaciones, escalados o cambios de intensidad a las imágenes originales. 10

autoencoder Tipo de red neuronal que aprende a comprimir datos de entrada en una representación latente de menor dimensión y luego reconstruir la entrada original. 2

boosting Técnica de ensamble que combina múltiples modelos débiles de forma secuencial, donde cada nuevo modelo corrige los errores del anterior. 14

búsqueda aleatoria con validación cruzada de tres folds Técnica de optimización de hiperparámetros en la que se seleccionan combinaciones aleatorias de parámetros y se evalúan mediante validación cruzada con tres particiones (folds) del conjunto de datos. En este procedimiento, los datos se dividen en tres subconjuntos del mismo tamaño: en cada iteración, dos de ellos se usan para entrenar el modelo y el tercero para validarlo, rotando el papel de cada subconjunto hasta que todos hayan sido usados como validación. Finalmente, se promedia el rendimiento obtenido en las tres iteraciones para estimar la calidad de cada configuración y seleccionar la más adecuada de forma eficiente.. 56

cabeza de regresión Componente final de una red neuronal diseñado específicamente para realizar tareas de regresión, prediciendo valores numéricos continuos. 2

caracterización Proceso de identificación y descripción detallada de las propiedades y características específicas de una patología o estructura anatómica. 1

clasificación Tarea en el análisis de imágenes médicas que consiste en asignar una o varias etiquetas clínicas a una imagen completa o a una región específica, permitiendo identificar la presencia o ausencia de patologías o características relevantes. 7

coeficiente de Dice Métrica de similitud que mide el solapamiento entre dos conjuntos, comúnmente utilizada para evaluar la calidad de segmentaciones médicas. 3

conexiones de salto Enlaces directos entre capas no consecutivas en una red neuronal que permiten preservar información espacial durante el procesamiento. 10

contexto clínico Información médica relevante del paciente que incluye historial, síntomas, tratamientos previos y otros factores que influyen en el diagnóstico y tratamiento. 1

contornos activos Método de segmentación que utiliza curvas deformables que evolucionan bajo la influencia de fuerzas internas y externas para delimitar objetos. 8

convolución Operación matemática fundamental en redes neuronales que aplica filtros sobre datos para extraer características locales. 2

descenso por gradiente Algoritmo de optimización que minimiza una función ajustando iterativamente los parámetros en la dirección opuesta al gradiente. 43

detección Proceso mediante el cual se localizan estructuras, lesiones o regiones de interés específicas dentro de una imagen médica, generalmente mediante algoritmos que identifican la posición y extensión de estos elementos. 7

downsampling Proceso de reducir la resolución espacial de un mapa de características, generalmente mediante técnicas como el muestreo por pasos, el uso de convoluciones con stride mayor que uno o el pooling, lo que disminuye el tamaño de la representación y puede implicar pérdida de información. 31

early stopping Técnica de regularización que detiene el entrenamiento cuando una métrica de validación deja de mejorar durante un número determinado de épocas. 58

edema Acumulación anormal de líquido en los tejidos, que provoca hinchazón. En el contexto de tumores cerebrales, suele referirse al edema peritumoral visible en imágenes de resonancia magnética. 22

edema peritumoral Acumulación de líquido inflamatorio en el tejido cerebral que rodea un tumor, visible en imágenes de resonancia magnética. 9

embeddings Representaciones vectoriales de alta dimensión que codifican información semántica de datos como texto o imágenes. 12

estaciones de trabajo Computadoras de alto rendimiento equipadas con hardware especializado como GPUs, diseñadas para tareas computacionalmente intensivas. 3

estadísticas vóxel Medidas cuantitativas calculadas a partir de los valores de intensidad de los vóxeles en una región específica de una imagen médica tridimensional. 2

evaluación pronóstica Proceso de predicción del curso probable de una enfermedad y sus resultados esperados basándose en factores clínicos y biomarcadores. 1

fusión multimodal Técnica que combina información proveniente de diferentes fuentes o modalidades de datos para mejorar el análisis y las predicciones. 2

glioma Tipo de tumor cerebral primario que se origina en las células gliales del sistema nervioso central, caracterizado por su agresividad y heterogeneidad. 1

grado histológico Clasificación que describe el grado de diferenciación celular y agresividad de un tumor basándose en características microscópicas. 1

hiperparámetros Parámetros de configuración del modelo que se establecen antes del entrenamiento y controlan el proceso de aprendizaje. 5

integración cuantitativa Proceso de combinar múltiples medidas numéricas y datos cuantitativos de forma sistemática para obtener una evaluación unificada. 1

logits Valores de salida sin procesar de una red neuronal antes de aplicar funciones de activación como softmax o sigmoide. 40

mecanismos de atención Técnica que permite al modelo enfocar selectivamente en partes relevantes de la entrada durante el procesamiento. 2

metástasis cerebrales Tumor secundario en el cerebro originado por células cancerosas que se han diseminado desde otro órgano. 27

modalidad En imagen médica, se refiere a diferentes tipos de secuencias o técnicas de adquisición (T1, T2, FLAIR, etc.). 1

modelo multimodal Sistema que integra y procesa simultáneamente diferentes tipos de datos, como imágenes médicas y texto clínico. 34

modelos clásicos Algoritmos de machine learning tradicionales como SVM, Random Forest o regresión lineal, en contraposición a los métodos de aprendizaje profundo. 2

multimodal Técnica que combina información proveniente de diferentes fuentes o modalidades de datos para mejorar el análisis y las predicciones. 1

necrosis Muerte del tejido tumoral, visible como regiones hipointensas en imágenes de resonancia magnética. IV, 24

normalización Proceso de ajustar los valores de intensidad de las imágenes a un rango estándar para mejorar el entrenamiento del modelo. 10

padding Técnica que añade valores (típicamente ceros) de manera simétrica alrededor de los bordes de una imagen o tensor para alcanzar dimensiones específicas. 33

patch Pequeña región rectangular extraída de una imagen, utilizada como unidad básica de procesamiento en Vision Transformers. 34

pooling Operación que reduce las dimensiones espaciales de los mapas de características, típicamente mediante máximo o promedio. 9

preprocesamiento Conjunto de transformaciones aplicadas a los datos antes del entrenamiento para mejorar la calidad y consistencia de la información. 5

progresión heterogénea Desarrollo variable e impredecible de una enfermedad que puede manifestarse de diferentes formas entre pacientes. 1

prototipado Proceso de desarrollo rápido de versiones preliminares de un sistema para probar conceptos y funcionalidades antes de la implementación final. 3

pseudoetiquetas Etiquetas generadas automáticamente por un modelo para datos no anotados, utilizadas en aprendizaje semi-supervisado. 34

radiología Especialidad médica que utiliza técnicas de imagen para diagnosticar y tratar enfermedades. 12

realce por contraste Aumento de intensidad en ciertas regiones de la imagen tras la administración de agente de contraste, indicativo de actividad tumoral. 13

red neuronal feedforward Tipo de red neuronal donde la información fluye en una sola dirección, desde la entrada hacia la salida. 43

regularización Técnica para prevenir el sobreajuste añadiendo penalizaciones a la función de pérdida del modelo. 37

representaciones latentes Codificaciones de alta dimensión aprendidas por modelos de aprendizaje profundo que capturan características esenciales de los datos de entrada. 1

resonancia magnética Técnica de imagen médica que utiliza campos magnéticos y ondas de radio para crear imágenes detalladas de órganos y tejidos. 1

ruido gaussiano Perturbación aleatoria que sigue una distribución normal, comúnmente utilizada en técnicas de aumento de datos para mejorar la robustez del modelo. 51

scheduler Componente que ajusta dinámicamente la tasa de aprendizaje durante el entrenamiento según criterios predefinidos como el número de épocas o el rendimiento del modelo. 56

segmentación Procedimiento que consiste en delimitar de forma precisa regiones anatómicas o patológicas dentro de una imagen médica, dividiéndola en partes significativas para facilitar el análisis, diagnóstico o planificación de tratamientos. 1

skull-stripping Proceso de eliminación del cráneo y tejidos extracraneales de imágenes cerebrales para enfocar el análisis en el parénquima cerebral. 25

sobreajuste Fenómeno donde un modelo aprende demasiado específicamente los datos de entrenamiento, perdiendo capacidad de generalización. 18

subtipo de glioma Clasificación específica de gliomas basada en características histológicas, genéticas y moleculares que determina el pronóstico y tratamiento. 1

supervivencia En oncología, tiempo transcurrido desde el diagnóstico hasta un evento específico como la muerte o recurrencia. 1

tensor Estructura matemática multidimensional utilizada para representar datos en redes neuronales. 13

transferencia de aprendizaje Técnica que utiliza conocimiento aprendido en una tarea para mejorar el rendimiento en una tarea relacionada. 10

transformers Arquitectura de red neuronal basada en mecanismos de atención, originalmente desarrollada para procesamiento de lenguaje natural. 2

umbralización Técnica de segmentación que separa regiones de una imagen basándose en valores de intensidad. 8

upsampling Proceso de aumentar la resolución espacial de un mapa de características mediante interpolación o convoluciones transpuestas. 33

variabilidad entre observadores Diferencias en la interpretación o medición de los mismos datos entre diferentes profesionales o evaluadores. 1

voxel Elemento volumétrico tridimensional, equivalente al píxel en imágenes 3D. 25

watershed Algoritmo de segmentación que modela la imagen como una superficie topográfica para identificar regiones. 8

Bibliografía

[Adeel et al., 2022] Adeel, A. M., Khan, K. B., Salahuddin, S., Rehman, E., Khan, S. A., Khan, M. A., Kadry, S., and Gandomi, A. H. (2022). A review on multimodal medical image fusion: Compendious analysis of medical modalities, multimodal databases, fusion techniques and quality metrics. *Computers in Biology and Medicine*, 144:105253.

[Andermatt et al., 2016] Andermatt, S., Pezold, S., and Cattin, P. C. (2016). Multi-dimensional gated recurrent units for the segmentation of biomedical 3d-data. In *International Workshop on Large-Scale Annotation of Biomedical Data and Expert Label Synthesis*, pages 142–151. Springer.

[Ardila et al., 2019] Ardila, D., Kiraly, A. P., Bharadwaj, S., Choi, B., Reicher, J. J., Peng, L., Tse, D., Etemadi, M., Ye, W., Corrado, G., et al. (2019). End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography. *Nature Medicine*, 25(6):954–961.

[Bakas et al., 2017] Bakas, S., Akbari, H., Sotiras, A., et al. (2017). Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features. *Scientific Data*, 4:170117.

[Bakas et al., 2018] Bakas, S., Reyes, M., Jakab, A., et al. (2018). Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. *arXiv preprint arXiv:1811.02629*.

[Baldassarre et al., 2019] Baldassarre, L. et al. (2019). Deep learning for segmentation and overall survival prediction of brain tumors from mri. *Journal of Neuroscience Methods*.

[Baur et al., 2021] Baur, C., Denner, S., Wiestler, B., Navab, N., and Albarqouni, S. (2021). Autoencoders for unsupervised anomaly segmentation in brain mr images: A comparative study. *Medical Image Analysis*, 69:101952.

[Beucher, 1992] Beucher, S. (1992). The watershed transformation applied to image segmentation. In *Proceedings of the 10th Pfefferkorn Conference on Signal and Image Processing in Microscopy and Microanalysis*, pages 299–314.

[Boecking et al., 2022] Boecking, B., Usuyama, N., Tsvetkov, Y., et al. (2022). Making the most of text semantics to improve biomedical vision–language processing. In Avidan, S. et al., editors, *Computer Vision – ECCV 2022*, volume 13696 of *Lecture Notes in Computer Science*, pages 1–21. Springer Nature.

[Bustos et al., 2020] Bustos, A., Pertusa, A., Salinas, J. M., and de la Iglesia-Vayá, M. (2020). Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical Image Analysis*, 66:101797.

[BytePlus, 2025] BytePlus (2025). Understanding adam optimizer: A deep dive into machine learning's adaptive optimization technique. `https://www.byteplus.com/en/topic/483788?title=understanding-adam-optimizer-a-deep-dive-into-machine-learning-s-adaptive-optimization-technique`. Accessed: Jun. 14, 2025.

[Cao et al., 2023] Cao, H., Wang, Y., Chen, J., et al. (2023). High-resolution swin transformer for automatic medical image segmentation. *Sensors*, 23(7):3420.

[Cao et al., 2021] Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., and Wang, M. (2021). Swin-unet: Unet-like pure transformer for medical image segmentation.

[Chai and Draxler, 2014] Chai, T. and Draxler, R. R. (2014). Root mean square error (rmse) or mean absolute error (mae)? – arguments against avoiding rmse in the literature. *Geoscientific Model Development*, 7(3):1247–1250.

[Chartsias et al., 2019] Chartsias, A., Joyce, T., Giuffrida, M. V., and Tsaftaris, S. A. (2019). Disentangled representation learning in cardiac image analysis. *Medical Image Analysis*, 58:101535.

[Chen et al., 2018] Chen, H., Dou, Q., Yu, L., Qin, J., and Heng, P.-A. (2018). Voxresnet: Deep voxelwise residual networks for brain segmentation from 3d mr images. *NeuroImage*, 170:446–455.

[Chen et al., 2021a] Chen, J. et al. (2021a). Transunet: Transformers make strong encoders for medical image segmentation. *arXiv preprint arXiv:2102.04306*.

[Chen et al., 2021b] Chen, R. J., Lu, M., Chen, T. Y., et al. (2021b). Multimodal coattention transformer for survival prediction in glioma patients. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 50–60. Springer.

[Chen and Guestrin, 2016] Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 785–794. ACM.

[Çiçek et al., 2016] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., et al. (2016). 3d u-net: Learning dense volumetric segmentation from sparse annotation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 424–432. Springer.

[D2iCE, 2025] D2iCE (2025). Applying graph neural networks for medical image analysis. `https://www.d2ice.ie/projects/imaging/gnn_medical`.

[Decidata, 2023] Decidata (2023). Evolución de la metodología de ciencia de datos. Accedido: [fecha de acceso].

[Dolz et al., 2019] Dolz, J., Gopinath, K., Yuan, J., Desrosiers, C., Ben Ayed, I., and Cheriet, F. (2019). Hyperdense-net: A hyper-densely connected cnn for multi-modal image segmentation. *IEEE Transactions on Medical Imaging*, 38(5):1116–1126.

[Dosovitskiy et al., 2020] Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

[Esteva et al., 2021] Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K. K., Cui, C., Corrado, G., Thrun, S., and Dean, J. (2021). A guide to deep learning in healthcare. *Nature Medicine*, 25(1):24–29.

[Golts et al., 2021] Golts, A., Khapun, D., Shats, D., Shoshan, Y., and Gilboa-Solomon, F. (2021). An ensemble of 3d u-net based models for segmentation of kidney and masses in ct scans. In *OpenReview*.

[Gonzalez and Woods, 2006] Gonzalez, R. C. and Woods, R. E. (2006). *Digital Image Processing*. Pearson Education, 3rd edition.

[Goodfellow et al., 2016] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

[Guan et al., 2024] Guan, H., Yap, P.-T., Bozoki, A., and Liu, M. (2024). Federated learning for medical image analysis: A survey. *Pattern Recognition*, 151.

[Hatamizadeh et al., 2022a] Hatamizadeh, A. et al. (2022a). Unetr: Transformers for 3d medical image segmentation. *Proceedings of CVPR*.

[Hatamizadeh et al., 2022b] Hatamizadeh, A., Nath, V., Tang, Y., Yang, D., Roth, H., and Xu, D. (2022b). Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. In *BrainLes (MICCAI Workshop)*, pages 284–296.

[Havaei et al., 2017] Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.-M., and Larochelle, H. (2017). Brain tumor segmentation with deep neural networks. *Medical Image Analysis*, 35:18–31.

[Havaei et al., 2016] Havaei, M., Guizard, N., Chapados, N., and Bengio, Y. (2016). Hemis: Hetero-modal image segmentation. *arXiv preprint arXiv:1607.05194*.

[He et al., 2021] He, Y., Zhao, X., Xu, M., Fan, Y., Zhao, Y., and Zhang, L. (2021). Multimodal transformer for survival prediction in glioma patients. *Medical Image Analysis*, 73:102170.

[Huang et al., 2021] Huang, X., Liu, Q., Shao, Y., Wang, D., Li, Q., Yao, J., and Xu, D. (2021). Gloria: A multimodal global-local representation learning framework for label-efficient medical image recognition. *IEEE Transactions on Medical Imaging*, 40(12):3413–3423.

[Huang et al., 2019] Huang, Y., Lin, Q., Xu, Z., et al. (2019). Fusion of medical imaging and electronic health records using deep learning: A systematic review and implementation guidelines. *NPJ Digital Medicine*, 3(1):1–9.

[Irvin et al., 2019] Irvin, J., Rajpurkar, P., Ko, M., Yu, Y., Ciurea-Ilcus, S., Chute, C., Marklund, H., Haghgoo, B., Ball, R., Shpanskaya, K., et al. (2019). Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):590–597.

[Isensee et al., 2021] Isensee, F., Jaeger, P. F., Kohl, S. A. A., Petersen, J., and Maier-Hein, K. H. (2021). nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, 18(2):203–211.

[Isensee et al., 2019] Isensee, F., Kickingereder, P., Wick, W., et al. (2019). Automated brain tumor segmentation using cascaded anisotropic convolutional neural networks. In *International MICCAI Brainlesion Workshop*, pages 419–427. Springer.

[Isensee et al., 2018a] Isensee, F., Petersen, J., Klein, A., Zimmerer, D., Jaeger, P. F., Kohl, S. A., Wasserthal, J., Koehler, G., Norajitra, T., Wirkert, S., and Maier-Hein, K. H. (2018a). No new-net. *arXiv preprint arXiv:1809.10486*.

[Isensee et al., 2018b] Isensee, F., Petersen, J., Kohl, S. A. A., Jäger, P. F., and Maier-Hein, K. H. (2018b). nnu-net: Self-adapting framework for u-net-based medical image segmentation. *arXiv preprint arXiv:1809.10486*.

[Jaccard, 1901] Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, 37:547–579.

[Kamnitsas et al., 2017] Kamnitsas, K., Ledig, C., Newcombe, V. F. J., Simpson, J. P., Kane, A. D., Menon, D. K., Rueckert, D., and Glocker, B. (2017). Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer.

[Kass et al., 1988] Kass, M., Witkin, A., and Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331.

[Kickingereder et al., 2016] Kickingereder, P. et al. (2016). Radiomic profiling of glioblastoma: Identifying mri predictors of patient outcome. *Neurology*.

[Kingma and Ba, 2015] Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. *International Conference on Learning Representations (ICLR)*.

[Labellerr, 2024] Labellerr (2024). Segmentation Simplified: A Deep Dive into Meta's SAM 2. `https://www.labellerr.com/blog/sam-2/`. Accessed: 2025-06-13.

[Labs, 2024] Labs, C. (2024). Gradient accumulation | continuum labs. `https://training.continuumlabs.ai/training/the-fine-tuning-process/hyperparameters/gradient-accumulation`. Accessed: Jun. 14, 2025.

[Li et al., 2022] Li, H. et al. (2022). Lvit: Language meets vision transformer in medical image segmentation. *arXiv preprint arXiv:2206.14718*.

[Li et al., 2017] Li, W., Wang, G., Fidon, L., Ourselin, S., Cardoso, M. J., and Vercauteren, T. (2017). Compactness prior based 3d segmentation with high-level prior for prostate mr images. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 371–379. Springer.

[Litjens et al., 2017] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A., van Ginneken, B., and Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88.

[Littmann et al., 2021] Littmann, G., Zuo, X., and Kainz, B. (2021). Validity of classical image processing techniques in the age of deep learning: a critical review. *Medical Image Analysis*, 70:101993.

[Lu et al., 2022] Lu, Y., Bian, W., Cheng, M.-M., Chen, Q., and Luo, P. (2022). Multimodal medical image segmentation with self-supervised fusion learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6997–7007.

[Malladi et al., 1995] Malladi, R., Sethian, J. A., and Vemuri, B. C. (1995). Shape modeling with front propagation: A level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):158–175.

[Martín and Sánchez, 2025] Martín, O. A. and Sánchez, J. (2025). Evaluation of vision transformers for multimodal image classification: A case study on brain, lung, and kidney tumors. Technical report, Cornell University.

[McKinney et al., 2020] McKinney, S. M., Sieniek, M., Godbole, V., Godwin, J., Antropova, N., Ashrafian, H., Back, T., Chesus, M., Corrado, G., Darzi, A., et al. (2020). International evaluation of an ai system for breast cancer screening. *Nature*, 577(7788):89–94.

[Medina and Sánchez, 2023] Medina, J. and Sánchez, J. (2023). High accuracy brain tumor classification with efficientnet and magnetic resonance images. In *Proceedings of the 5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI 2023)*, pages 55–63, Tenerife, Canary Islands, Spain. International Frequency Sensor Association (IFSA) Publishing, SL.

[Mehta et al., 2021] Mehta, D., Srivastava, A., et al. (2021). You only look one-level feature. *arXiv preprint arXiv:2101.05027*.

[Menze et al., 2015] Menze, B. H., Jakab, A., Bauer, S., et al. (2015). The multimodal brain tumor image segmentation benchmark (brats). *IEEE Transactions on Medical Imaging*, 34(10):1993–2024.

[Menze et al., 2014] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., et al. (2014). The multimodal brain tumor image segmentation benchmark (brats). *IEEE Transactions on Medical Imaging*, 34(10):1993–2024.

[Meta AI, 2025] Meta AI (2025). Segment Anything Model 2 (SAM 2). `https://ai.meta.com/sam2/`. Accessed: 2025-06-13.

[Milletari et al., 2016a] Milletari, F., Navab, N., and Ahmadi, S.-A. (2016a). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *Fourth International Conference on 3D Vision (3DV)*, pages 565–571. IEEE.

[Milletari et al., 2016b] Milletari, F., Navab, N., and Ahmadi, S.-A. (2016b). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *Proceedings of 2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571.

[Mobadersany et al., 2018] Mobadersany, P., Yousefi, S., Amgad, M., Gutman, D. A., Barnholtz-Sloan, J. S., Velázquez Vega, J. E., Brat, D. J., and Cooper, L. A. D. (2018). Predicting cancer outcomes from histology and genomics using convolutional networks. *Proceedings of the National Academy of Sciences*, 115(13):E2970–E2979.

[Murphy, 2012] Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. MIT Press.

[Myronenko, 2018] Myronenko, A. (2018). 3d mri brain tumor segmentation using autoencoder regularization. In *International MICCAI Brainlesion Workshop*, pages 311–320. Springer.

[Nicolas et al., 2000] Nicolas, J. et al. (2000). A comparison of automatic and semi-automatic thresholding techniques for mri lesion volume quantification in multiple sclerosis. *Magnetic Resonance Imaging*, 18(3):279–288.

[of Geneva, 2024] of Geneva, U. (2024). Molecular imaging: Pet/ct. `https://www.unige.ch/medecine/pippa/molecular-imaging-petct`.

[Oh et al., 2020] Oh, T. S., Mummadi, C. K., Krähenbühl, P., Keutzer, K., and Darrell, T. (2020). 3d-yolo: End-to-end real-time 3d object detection on point clouds. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5263–5270. IEEE.

[Oktay et al., 2018] Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N., Kainz, B., et al. (2018). Attention u-net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*.

[Osher and Sethian, 1988] Osher, S. and Sethian, J. A. (1988). Fronts propagating with curvature-dependent speed: algorithms based on hamilton–jacobi formulations. *Journal of Computational Physics*, 79(1):12–49.

[Paszke et al., 2019] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035.

[Pereira et al., 2016] Pereira, S., Pinto, A., Alves, V., and Silva, C. A. (2016). Brain tumor segmentation using convolutional neural networks in mri images. *IEEE Transactions on Medical Imaging*, 35(5):1240–1251.

[Pham et al., 2000] Pham, D. L., Xu, C., and Prince, J. L. (2000). Current methods in medical image segmentation. *Annual Review of Biomedical Engineering*, 2(1):315–337.

[Prechelt, 1998] Prechelt, L. (1998). Early stopping - but when? In *Neural Networks: Tricks of the Trade*, pages 55–69. Springer.

[Qin et al., 2021] Qin, D., Bu, J., Liu, Z., Shen, X., Zhou, S., Gu, J., Wang, Z., Wu, L., and Dai, H. (2021). Efficient medical image segmentation based on knowledge distillation. *IEEE Transactions on Medical Imaging*, 40(12):3820–3831.

[Rahman et al., 2022] Rahman, M. M., Islam, S. S., Rahman, M. A., and Islam, S. M. S. (2022). Spatial feature fusion in 3d convolutional autoencoders for lung tumor segmentation from 3d ct images. *SSRN Electronic Journal*.

[Rahman and Wang, 2016] Rahman, M. M. and Wang, Y. (2016). Optimizing intersection-over-union in deep neural networks for image segmentation. *arXiv preprint arXiv:1608.01471*.

[Rajpurkar et al., 2018] Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., et al. (2018). Deep learning for chest radiograph diagnosis: A retrospective comparison of the chexnext algorithm to practicing radiologists. *PLoS Medicine*, 15(11):e1002686.

[Rane, 2023] Rane, N. (2023). Transformers for medical image analysis: Applications, challenges, and future scope. *SSRN Electronic Journal*.

[Rasmussen and Williams, 2006] Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. MIT Press.

[Reve, 2024] Reve, A. (2024). Autoencoders for detecting anomalies in medical imaging. Bachelor's thesis, University of Twente.

[Reyes and Sánchez, 2024] Reyes, D. and Sánchez, J. (2024). Performance of convolutional neural networks for the classification of brain tumors using magnetic resonance imaging. *Heliyon*, 10(3).

[Roerdink and Meijster, 2000] Roerdink, J. B. and Meijster, A. (2000). The watershed transform: Definitions, algorithms and parallelization strategies. *Fundamenta Informaticae*, 41(1–2):187–228.

[Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241.

[Russ and Neal, 2016] Russ, J. C. and Neal, J. C. (2016). *The Image Processing Handbook*. CRC Press, 7th edition.

[Schlemper et al., 2019] Schlemper, J., Oktay, O., Schaap, M., Heinrich, M., Kainz, B., Glocker, B., and Rueckert, D. (2019). Attention gated networks: Learning to leverage salient regions in medical images. *Medical Image Analysis*, 53:197–207.

[Science, 2025] Science, T. D. (2025). Time series augmentations. `https://towardsdatascience.com/time-series-augmentations-16237134b29b/`. Accessed: Jun. 14, 2025.

[Serra, 1983] Serra, J. (1983). *Image Analysis and Mathematical Morphology*. Academic Press.

[Setio et al., 2016] Setio, A. A. A., Traverso, A., de Bel, T., Berens, M. S., van den Bogaard, C., Cerello, P., Ciompi, F., et al. (2016). Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: The luna16 challenge. *Medical Image Analysis*, 42:1–13.

[Seymour et al., 2000] Seymour, A. M., Worsley, K. J., and Evans, A. C. (2000). An intensity-based medical image registration algorithm using robust statistics. *NeuroImage*, 12(1):30–41.

[Shearer, 2000] Shearer, C. (2000). The crisp-dm model: The new blueprint for data mining. `https://www.the-modeling-agency.com/crisp-dm.pdf`. The Modeling Agency.

[Shen et al., 2017] Shen, D., Wu, G., and Suk, H.-I. (2017). Deep learning in medical image analysis. *Annual review of biomedical engineering*, 19:221–248.

[Siddiquee et al., 2022] Siddiquee, M. M. R., Yang, D., He, Y., Xu, D., and Myronenko, A. (2022). Automated segmentation of intracranial hemorrhages from 3d ct: Instance 2022 challenge report. *arXiv preprint arXiv:2209.10648*.

[Soltaninejad et al., 2021] Soltaninejad, M. et al. (2021). Predicting overall survival time in glioblastoma patients using gradient boosting machines algorithm and recursive feature elimination technique. *Computers in Biology and Medicine*, 139:104914.

[Song, 2025] Song, J. (2025). A review of multimodal medical image classification based on deep learning. *Mathematical Modeling and Algorithm Application*, 4(2):39–47.

[Stocksieker et al., 2023] Stocksieker, S., Pommeret, D., and Charpentier, A. (2023). Data augmentation for imbalanced regression. In *Proceedings of the 40th International Conference on Machine Learning*, volume 206 of *Proceedings of Machine Learning Research*, pages 33638–33660.

[Sudre et al., 2017] Sudre, C. H., Li, W., Vercauteren, T., Ourselin, S., and Cardoso, M. J. (2017). Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, 10553:240–248.

[Tajbakhsh et al., 2016] Tajbakhsh, N., Shin, J. Y., Gurudu, S. R., Hurst, R. T., Kendall, C. S., Gotway, M. B., and Liang, J. (2016). Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE Transactions on Medical Imaging*, 35(5):1299–1312.

[Ultralytics, 2025] Ultralytics (2025). SAM 2: Segment Anything Model 2 - Ultralytics YOLO Docs. `https://docs.ultralytics.com/models/sam-2/`. Accessed: 2025-06-13.

[Valle et al., 2021] Valle, E., Ribeiro, F., Fornaciali, M., et al. (2021). Deep learning for breast cancer histopathological image classification: A review. *Computers in Biology and Medicine*, 131:104248.

[Valverde et al., 2017] Valverde, S., Cabezas, M., Roura, E., Gonzalez-Villa, S., Pareto, D., Vilanova, J. C., Ramio-Torrenta, L., Rovira, A., Oliver, A., and Llado, X. (2017). Improving automated multiple sclerosis lesion segmentation with a cascaded 3d convolutional neural network approach. *NeuroImage: Clinical*, 17:1006–1015.

[Valverde et al., 2020] Valverde, S., Cabezas, M., Roura, E., Gonzalez-Villa, S., Pareto, D., Vilanova, J. C., Ramio-Torrenta, L., Rovira, A., Oliver, A., and Llado, X. (2020). Fusionnet: A deep fully residual convolutional neural network for multi-modal brain tumor image segmentation. *Computer Methods and Programs in Biomedicine*, 197:105620.

[van Griethuysen et al., 2017] van Griethuysen, J. J. et al. (2017). Computational radiomics system to decode the radiographic phenotype. *Cancer Research*, 77(21):e104.

[Vincent, 1993] Vincent, L. (1993). Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. *IEEE Transactions on Image Processing*, 2(2):176–201.

[Wang et al., 2021a] Wang, F. et al. (2021a). A multi-task deep learning framework for brain tumor segmentation and prognosis. *Medical Image Analysis*.

[Wang et al., 2022] Wang, L., Wang, Y., Wang, Y., et al. (2022). Lvit: Language meets vision transformer. *arXiv preprint arXiv:2206.14718*.

[Wang et al., 2020] Wang, R., Lei, T., Cui, R., Zhang, B., Meng, H., and Nandi, A. K. (2020). Medical image segmentation using deep learning: A survey. *arXiv preprint arXiv:2009.13120*.

[Wang et al., 2021b] Wang, Y., Chen, J., Zhong, Z., Li, Y., Huang, Z., and Chen, H. (2021b). Transbts: Multimodal brain tumor segmentation using transformer. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 109–119. Springer.

[Wang et al., 2024] Wang, Y., Li, H., Zhang, X., et al. (2024). 3d medical image analysis with autoencoder-based feature extraction and shallow ensemble learning. In *Proc. SPIE 12998, Medical Imaging 2024: Computer-Aided Diagnosis*, page 129981E.

[Wang et al., 2023] Wang, Z., Zhang, Y., et al. (2023). 3d autoencoder for unsupervised medical image analysis. *arXiv preprint arXiv:2307.05445*.

[Wen and Angryk, 2024] Wen, J. and Angryk, R. A. (2024). Class-based time series data augmentation to mitigate extreme class imbalance for solar flare prediction. *arXiv preprint arXiv:2405.20590*.

[Willmott and Matsuura, 2005] Willmott, C. J. and Matsuura, K. (2005). Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Research*, 30(1):79–82.

[Xu et al., 2019] Xu, D., Zhou, S., Yuan, Q., Li, H., Ma, J., Chen, Y., and Heng, P.-A. (2019). Efficient deep network architecture for 3d medical image segmentation. *Medical Image Analysis*, 54:1–13.

[Xu et al., 2024] Xu, Y., Quan, R., Xu, W., Huang, Y., Chen, X., and Liu, F. (2024). Advances in medical image segmentation: A comprehensive review of traditional, deep learning and hybrid approaches. *Bioengineering*, 11(10):1034.

[Zhang et al., 2024] Zhang, B. et al. (2024). Segment anything model 2 (sam-2). *arXiv preprint arXiv:2408.00714*.

[Zhang et al., 2023a] Zhang, H., Chen, Y., Yang, Y., et al. (2023a). Medklip: Medical knowledge enhanced language-image pre-training. *arXiv preprint arXiv:2303.09053*.

[Zhang et al., 2023b] Zhang, H., Yang, Y., Chen, Y., et al. (2023b). Clip-medical: Contrastive pretraining of medical image-text pairs with lightweight transformers. *arXiv preprint arXiv:2304.11048*.

[Zhang et al., 2021a] Zhang, X., Zhang, Y., Wang, Y., and Liu, Q. (2021a). Blood vessel segmentation based on the 3d residual u-net. *International Journal of Pattern Recognition and Artificial Intelligence*, 35(13):2157007.

[Zhang et al., 2001] Zhang, Y., Brady, M., and Smith, S. (2001). Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging*, 20(1):45–57.

[Zhang et al., 2021b] Zhang, Y., Liu, H., Hu, Q., Sun, J., and Zheng, Y. (2021b). Multimodal fusion transformer for brain tumor segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 347–356. Springer.

[Zhao et al., 2018] Zhao, Z., Li, S., Liu, Z., Li, F., and Benediktsson, J. A. (2018). 3d deep learning based classification of hyperspectral images. In *2018 IEEE International Conference on Image Processing (ICIP)*, pages 2326–2330. IEEE.

[Zhou et al., 2018] Zhou, H., Qi, Y., Yin, Y., Li, T., Liu, X., Li, X., Gong, G., and Wang, L. (2018). Mmfnet: Multimodality mri fusion network for nasopharyngeal carcinoma segmentation. *arXiv preprint arXiv:1812.10033*.

[Zhou et al., 2021] Zhou, Z., Gu, Y., Dong, Z., and Gao, Z. (2021). A review of multimodal medical image fusion techniques. *Computer Methods and Programs in Biomedicine*, 207:106228.

[Zhou et al., 2019] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., and Liang, J. (2019). Deep learning for medical image segmentation: A survey. *arXiv preprint arXiv:1904.08128*.

[Zhu and Yuille, 1996] Zhu, S.-C. and Yuille, A. L. (1996). Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):884–900.

[Zhu et al., 2023] Zhu, Z., Gao, X., Li, Y., Hao, L., Yang, G., and Wang, H. (2023). Ca-segresnet: A context-aware segmentation residual network for vertebral bone segmentation in ct spine images. *SSRN Electronic Journal*.

[Zilliz, 2023] Zilliz (2023). How does early stopping prevent overfitting in deep learning? `https://zilliz.com/ai-faq/how-does-early-stopping-prevent-overfitting-in-deep-learning`. Accessed: Jun. 14, 2025.

ULPGC
Universidad de
Las Palmas de
Gran Canaria

ESCUELA DE INGENIERÍA
INFORMÁTICA