UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

ESCUELA POLITÉCNICA SUPERIOR DE ELCHE

GRADO EN INGENIERÍA INFORMÁTICA EN TECNOLOGÍAS DE LA INFORMACIÓN

Análisis Topológico de Datos (TDA) aplicado al EEG: Un enfoque innovador para el diagnóstico y la comprensión de afecciones neurológicas

TRABAJO FIN DE GRADO

Febrero 2025

AUTOR: Alejandro Martínez Gracia
DIRECTOR: Jesús Javier Rodríguez Sala
CODIRECTORA: Miriam Esteve Campello

Agradecimientos a todos los que me han ayudado.

Ut ultrices nisi ut dui semper scelerisque.
Cras malesuada consequat est, a pellentesque libero dignissim id.

RESUMEN

El proyecto "Análisis Topológico (TDA) aplicado al EEG: Un enfoque innovador para el diagnóstico y la comprensión de afecciones neurológicas" se centra en la aplicación del análisis topológico de datos de electroencefalografía (EEG) con el objetivo de mejorar la identificación y diagnóstico de diversas afecciones neurológicas. Este enfoque utiliza técnicas computacionales para analizar las propiedades topológicas de las señales EEG, lo que permite obtener información adicional sobre el funcionamiento cerebral.

El trabajo se basa en la idea de que el análisis topológico puede proporcionar una comprensión más profunda de los patrones complejos presentes en las señales EEG y ayudar a identificar cambios sutiles en la actividad cerebral asociados con diversas condiciones neurológicas. Estas técnicas están diseñadas para ser robustas y fiables, incluso frente a ruido o datos no relacionados.

La carga y preprocesamiento de los datos EEG es un componente crucial del proyecto, ya que garantiza la calidad y consistencia de los datos utilizados en el análisis. Una vez cargados y preparados, estos datos se someten a las técnicas topológicas avanzadas para obtener una representación matemática detallada de la actividad cerebral.

Al analizar los datos EEG, el proyecto utiliza el análisis topológico para identificar patrones y estructuras en la actividad cerebral que no podrían detectarse con métodos tradicionales. Esta información se visualiza e interpreta en el contexto clínico, lo cual es fundamental para comprender las afecciones neurológicas estudiadas.

El proyecto ofrece una serie de beneficios significativos, incluyendo una mejora en la precisión diagnóstica, un entendimiento más profundo de las afecciones neurológicas y el potencial para intervenciones personalizadas y terapias innovadoras. La capacidad del análisis topológico de proporcionar información sobre patrones específicos de actividad cerebral podría facilitar la identificación de las áreas necesarias para un tratamiento individualizado.

Este proyecto aborda importantes desafíos en el campo de la neurociencia computacional y tiene el potencial de revolucionar los métodos clínicos actuales para el diagnóstico y la comprensión de afecciones neurológicas.

ÍNDICE

Biblioteca UNIVERSITAS Miguel Hernández

ÍNDICE TABLAS

Biblioteca
UNIVERSITAS Miguel Hernández

ÍNDICE FIGURAS

Capítulo 1
Introducción

1.1. ENTORNO DE APLICACIÓN

La enfermedad de Alzheimer es una condición neurodegenerativa progresiva que afecta a millones de personas en todo el mundo, causando deterioro cognitivo y pérdida de memoria [1]. La investigación en este campo ha evolucionado significativamente con el avance de tecnologías como la electroencefalografía (EEG), que permite la monitorización en tiempo real de la actividad eléctrica del cerebro [2]. Sin embargo, a pesar de los avances, existe una necesidad crítica de métodos más precisos y detallados para analizar y comprender las dinámicas neuronales asociadas con la enfermedad [3].

Este proyecto se centra en la aplicación innovadora del Análisis de Datos Topológicos (TDA), un enfoque matemático y computacional que combina técnicas de geometría y topología para analizar datos complejos y estructurados. Se enfoca en un conjunto exhaustivo de grabaciones de EEG para la enfermedad de Alzheimer, con el objetivo de

descubrir patrones ocultos en los datos que puedan mejorar significativamente la precisión del diagnóstico y nuestra comprensión de la progresión de la enfermedad [4]. Al integrar TDA en el análisis de datos EEG, se busca aportar una nueva perspectiva a la investigación sobre el Alzheimer, lo que podría contribuir al desarrollo de métodos diagnósticos más efectivos, personalizados y precisos. El proyecto tiene como objetivo explorar las relaciones complejas entre los patrones de actividad cerebral y la enfermedad, con el fin de mejorar nuestra capacidad para predecir la progresión de la enfermedad y encontrar nuevos tratamientos eficaces.

1.2. JUSTIFICACIÓN DEL PROYECTO

El proyecto TED2021-129347B-C22, financiado por el Ministerio de Ciencia e Innovación/Agencia Estatal de Investigación y por la Universidad Cardenal Herrera CEU, forma parte de los Proyectos Estratégicos Orientados a la Transición Ecológica y a la Transición Digital. Este proyecto está liderado por el grupo de investigación Métodos Computacionales para la Modelización de Procesos Físicos, adscrito al Departamento de Matemáticas, Física y Ciencias Tecnológicas de esta Universidad.

El objetivo principal del proyecto es desarrollar herramientas digitales avanzadas aplicadas al análisis de datos masivos, específicamente mediante el uso del Análisis Topológico de Datos (TDA) en el estudio de la electroencefalografía (EEG). Estas herramientas buscan identificar patrones neuronales complejos que permitan la detección temprana de enfermedades neurodegenerativas como el Alzheimer.

Iniciado en 2021, el proyecto refuerza la línea de investigación sobre el uso de tecnologías digitales avanzadas en el ámbito biomédico, alineándose con los objetivos de la transición digital en el sector salud. El equipo de investigación ha retomado y optimizado el desarrollo de estas herramientas tras un proceso de reestructuración técnica, mejorando la metodología para ajustarse a los nuevos avances tecnológicos y científicos.

Este proyecto no solo impulsa la innovación en el diagnóstico de enfermedades neurodegenerativas, sino que también contribuye a la digitalización y modernización de los métodos computacionales en la investigación médica, respondiendo a los retos de la transformación digital en la ciencia y la tecnología.

1.3. OBJETIVOS

El objetivo general de este proyecto es desarrollar un enfoque innovador basado en el Análisis Topológico de Datos (TDA) para el análisis de señales EEG, con el fin de mejorar

la detección y clasificación de enfermedades neurodegenerativas, como el Alzheimer y la demencia frontotemporal. A continuación, se detallan los objetivos principales, secundarios y personales:

1.3.1. Objetivo principal

Desarrollar una metodología basada en TDA para el análisis de señales EEG que permita identificar patrones topológicos complejos asociados a enfermedades neurodegenerativas, proporcionando un enfoque más preciso que los métodos tradicionales para la clasificación y diagnóstico temprano de dichas enfermedades.

1.3.2. Objetivos secundarios

La persistencia homológica es una herramienta matemática que se utiliza para analizar la estructura de un conjunto de datos. En el contexto del análisis de señales EEG, la persistencia homológica permite identificar en las señales patrones topológicos complejos que no son evidentes con los métodos tradicionales de análisis. En particular, la persistencia homológica se utiliza para detectar cambios sutiles en la conectividad cerebral asociados con enfermedades neurodegenerativas como el Alzheimer y la demencia frontotemporal. Esta herramienta permite analizar las señales EEG de manera más detallada, capturando patrones que no se manifiestan con otros métodos [5, 6].

El primer objetivo del proyecto es explorar esta técnica para identificar características topológicas clave en las señales EEG que puedan diferenciar entre sujetos sanos y aquellos con enfermedades neurodegenerativas. Al utilizar la persistencia homológica, se espera poder descubrir patrones topológicos nuevos que no son evidentes con otros métodos de análisis [7]. Esto puede proporcionar una comprensión más profunda de cómo las enfermedades neurodegenerativas afectan la dinámica neuronal y, potencialmente, mejorar la detección y el tratamiento de estas condiciones.

El segundo objetivo es aplicar la metodología TDA a conjuntos de datos reales de EEG obtenidos del repositorio OpenNeuro [8], que incluyen registros de pacientes con Alzheimer, demencia frontotemporal y sujetos sanos. Este análisis práctico permitirá evaluar la efectividad de TDA en un contexto clínico real, proporcionando una validación empírica de la metodología desarrollada. Al trabajar con datos reales, se podrá medir la capacidad de TDA para identificar y diferenciar patrones neuronales relevantes, contribuyendo a una comprensión más profunda de las alteraciones cerebrales asociadas con las enfermedades estudiadas.

El tercer objetivo del proyecto es crear una librería de Python intuitiva y fácil de usar que permita a investigadores y clínicos acceder a herramientas de análisis topológico de manera rápida y eficiente. Esta librería será una herramienta computacional versátil que proporcionará una implementación sencilla y escalable de técnicas de TDA, lo que facilitará la interpretación de patrones neuronales complejos. El objetivo final es mejorar el acceso a los resultados analíticos, apoyando tanto la investigación científica como la práctica clínica en la detección y estudio de enfermedades neurodegenerativas.

1.3.3. Objetivos personales

Uno de los objetivos personales más importantes de este proyecto es desarrollar competencias avanzadas en el uso de Análisis Topológico de Datos (TDA), aplicándolo específicamente a problemas biomédicos. Al trabajar en el contexto del análisis de señales EEG, se busca adquirir un conocimiento profundo sobre cómo el TDA puede ser una herramienta poderosa para revelar patrones ocultos en datos complejos, proporcionando nuevas formas de analizar y comprender las dinámicas neuronales.

Además, se espera adquirir experiencia en la gestión y análisis de grandes volúmenes de datos EEG reales. Esto implicará el uso de conjuntos de datos clínicos obtenidos de repositorios como OpenNeuro, lo que permitirá mejorar las habilidades en el manejo de datos masivos y el procesamiento de señales biomédicas, conocimientos clave en el campo de la neurociencia computacional.

Otro objetivo personal relevante es contribuir al avance de la investigación en enfermedades neurodegenerativas, específicamente en el diagnóstico temprano de patologías como el Alzheimer y la demencia frontotemporal. Este proyecto brindará la oportunidad de desarrollar soluciones tecnológicas innovadoras que pueden tener un impacto significativo en la práctica médica y en la calidad de vida de los pacientes, lo que representa una gran motivación para este trabajo.

Asimismo, se busca mejorar las habilidades de colaboración interdisciplinaria al trabajar en un entorno de investigación donde convergen diversas áreas como las matemáticas, la computación, la neurociencia y la medicina. La interacción con expertos de distintas disciplinas permitirá aprender a integrar diferentes enfoques y metodologías para abordar problemas complejos de una manera más holística y efectiva.

Finalmente, un objetivo personal adicional es desarrollar una librería en Python que tenga potencial aplicabilidad en el entorno clínico o de investigación, con la posibilidad de ser explotada comercialmente en el futuro. Este objetivo va más allá de obtener resultados académicos, ya que se aspira a crear una solución práctica y funcional que pueda contribuir

al desarrollo de nuevas tecnologías en el diagnóstico de enfermedades neurodegenerativas, ofreciendo beneficios tanto en el ámbito científico como profesional.

Biblioteca
UNIVERSITAS Miguel Hernández

Capítulo 2: Antecedentes y estado de la cuestión

2.1.- RELEVANCIA DEL ANÁLISIS DE EEG

El electroencefalograma (EEG) es una herramienta esencial para el estudio de la actividad eléctrica del cerebro, permitiendo la medición de oscilaciones neuronales que reflejan el estado funcional del sistema nervioso central. Los datos EEG son considerados multidimensionales debido a las complejidades de la actividad cerebral, que se manifiestan en múltiples frecuencias y escalas temporales. Además, los patrones de actividad EEG están influenciados por factores como la distribución espacial de las señales cerebrales, lo que hace que sea necesario desarrollar métodos avanzados para analizar e interpretar estos datos (ver Figura 1).

En este sentido, el análisis de EGG tradicionalmente ha dependido de métodos como la transformada de Fourier y el análisis espectral para identificar patrones relacionados con trastornos neurológicos [8]. Sin embargo, estos enfoques pueden tener limitaciones a la

hora de captar complejidades sutiles en las señales cerebrales, especialmente en enfermedades neurodegenerativas como el Alzheimer y la demencia frontotemporal. En estos casos, los patrones de actividad EEG pueden ser demasiado complejos, no siendo captados con precisión por métodos tradicionales [9].

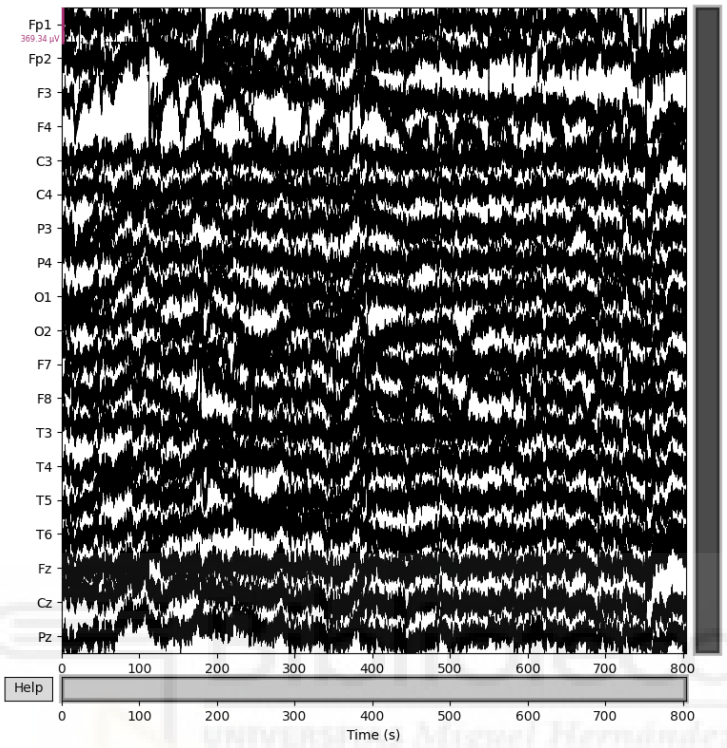

Figura 1. Electroencefalograma (EEG) en tiempo real
(patrones de actividad cerebral durante la aplicación del análisis topológico)

En este contexto, es fundamental desarrollar y utilizar métodos más avanzados para analizar los patrones de actividad EEG y mejorar la sensibilidad y especificidad del diagnóstico. El análisis topológico de las señales cerebrales puede ser una herramienta valiosa en este sentido, ya que permite el estudio de la estructura y la organización de las conexiones neuronales de manera más detallada y precisa

2.2.- ANÁLISIS TOPOLÓGICO DE DATOS (TDA) Y SU APLICACIÓN EN NEUROCIENCIA

El Análisis Topológico de Datos (TDA) ha emergido como una metodología avanzada para explorar la estructura topológica de datos complejos. La persistencia homológica, un concepto central en TDA, permite identificar características invariables de los datos a través de diferentes escalas y transformaciones [3]. Antes de profundizar en sus aplicaciones, se explicarán los conceptos básicos del TDA para proporcionar un marco

teórico sólido. Esta técnica ha mostrado su utilidad en áreas como la biología y la física, para desentrañar patrones complejos y relaciones en conjuntos de datos multidimensionales [4].

En la neurociencia, el uso de TDA para el análisis de datos EEG es una innovación reciente. TDA ofrece una nueva perspectiva sobre la organización de las señales neuronales, que podría mejorar la detección de alteraciones sutiles en la actividad cerebral asociadas con enfermedades neurodegenerativas [5]. Al aplicar TDA al EEG, se puede captar la topología de las señales neuronales, proporcionando un enfoque complementario a los métodos tradicionales de análisis.

2.2.1. Complejo simplicial

Un complejo simplicial (*complex simplex*, en inglés) es una estructura matemática fundamental utilizada para representar y analizar las características topológicas de un conjunto de datos. Es particularmente útil para capturar la conectividad y las relaciones entre puntos de datos, denominados símplices o *simplex*. Un simplex es la envoltura convexa de un conjunto de puntos afinmente independientes en un espacio euclidiano. Formalmente, sea V un conjunto de vértices, un simplex σ es un subconjunto de V. Un complejo simplicial Δ es una colección de símplices tal que para cualquier σ en Δ, todas las caras de σ también están en Δ, y la intersección de dos símplices en Δ también es un simplex en Δ. Esto se expresa como $\sigma \in \Delta$ implica que todas las caras de σ están en Δ, y para $\sigma 1, \sigma 2 \in \Delta$, $\sigma 1 \cap \sigma 2$ también está en Δ.

Considérese un complejo simplicial en el contexto del TDA, donde cada símplex representa un segmento o un vértice en el conjunto de datos. Un 0-simplex corresponde a un punto individual, un 1-simplex representa un segmento de línea que conecta dos puntos, un 2-simplex forma una cara triangular, y así sucesivamente (ver Figura 2). Los vértices y aristas del complejo simplicial capturan los bloques básicos de la topología del conjunto de datos.

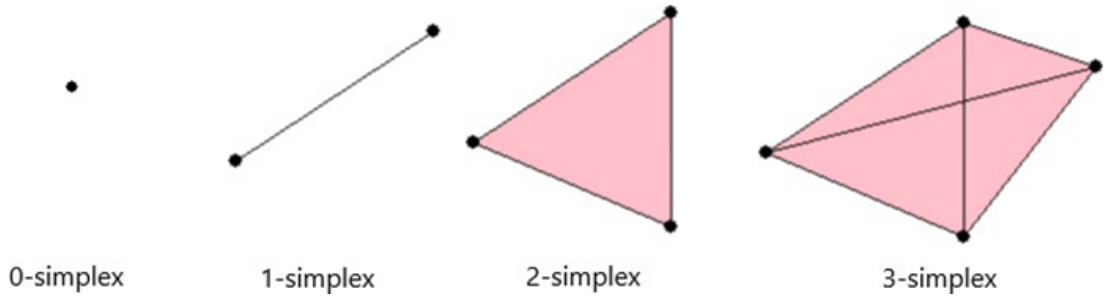

Figura 2: Símplices de diferentes dimensiones.

En términos prácticos, dado un conjunto de datos en tres dimensiones con puntos x_1, x_2, x_3, x_4. Un complejo simplicial Δ podría incluir triángulos como $\{x_1, x_2, x_3\}$ y $\{x_2, x_3, x_4\}$, así

como el tetraedro $\{x_1, x_2, x_3, x_4\}$, formando una representación compleja de las interconexiones entre estos vértices.

Cada símplex representa un subconjunto de puntos, y la colección de todos estos símplices forma el complejo simplicial abstracto asociado a S. El complejo de Rips, denotado como $R_\epsilon(S)$, se construye en base a un parámetro de escala elegido ϵ. La idea central es formar símplices, que son formas geométricas, basadas en las distancias por pares entre puntos en el conjunto de datos. Si la distancia entre dos puntos es menor o igual a ϵ, están conectados en el complejo. Formalmente, para un símplex $\sigma = \{x_{i1}, x_{i2}, ..., x_{ik}\}$, se incluye en $R_\epsilon(S)$ si $d(x_i p, x_i q) \leq \epsilon$ para todos $1 \leq p, q \leq k$. Este proceso crea una secuencia anidada de complejos de Rips a medida que varía el parámetro de escala ϵ.

Una filtración de un complejo simplicial es una secuencia de complejos simpliciales que están anidados entre sí, proporcionando una forma de estudiar los cambios topológicos a lo largo de la secuencia. Específicamente, el complejo de Rips, $R_\epsilon(S)$, es una colección de subcomplejos tal que

$$R_{\epsilon 0}(S) \subset R_{\epsilon 1}(S) \subset R_{\epsilon 2}(S) \subset ... \subset R_{\epsilon n}(S),$$
$$\text{donde: } 0 = \epsilon_0 \leq \epsilon_1 \leq ... \leq \epsilon_n.$$

Cada $R_{\epsilon i}(S)$ es un complejo simplicial en sí mismo, y la inclusión $\subset$ indica que cada complejo en la secuencia contiene todos los símplices del anterior junto con, posiblemente, más símplices añadidos a medida que avanza la secuencia. Esta estructura permite el estudio de la evolución de las características topológicas del complejo, como componentes convexas, agujeros y vacíos, a medida que se añaden más símplices. Las filtraciones son particularmente útiles en la homología persistente, un método en topología computacional que analiza las características topológicas de un espacio en diferentes resoluciones espaciales. Al examinar cómo estas características aparecen y desaparecen a lo largo de una filtración, se puede identificar qué características son más persistentes a través de la escala, sugiriendo que probablemente sean atributos significativos del espacio topológico subyacente en lugar de ruido.

La Figura 3 muestra un ejemplo de cómputo de un complejo de Rips para un conjunto de puntos dispuestos en un círculo para ϵ = 0, 0.5, 1, y 1.8. ϵ = 0 (figura 3.A) muestra el estado inicial con cada punto aislado, indicando que no hay conexiones en el parámetro de escala más pequeño; ϵ = 0.5 (figura 3.B) comienza a revelar la estructura subyacente del círculo ya que algunos puntos están ahora conectados, formando las aristas iniciales; ϵ = 1 (figura 3.C) muestra una estructura más conectada, con más aristas formándose y comenzando a delinear más claramente la forma circular; y ϵ = 1.8 (figura 3.D), en esta etapa, la topología del círculo está bien representada con una red densa de conexiones, mostrando la estructura del círculo a través del complejo de Rips.

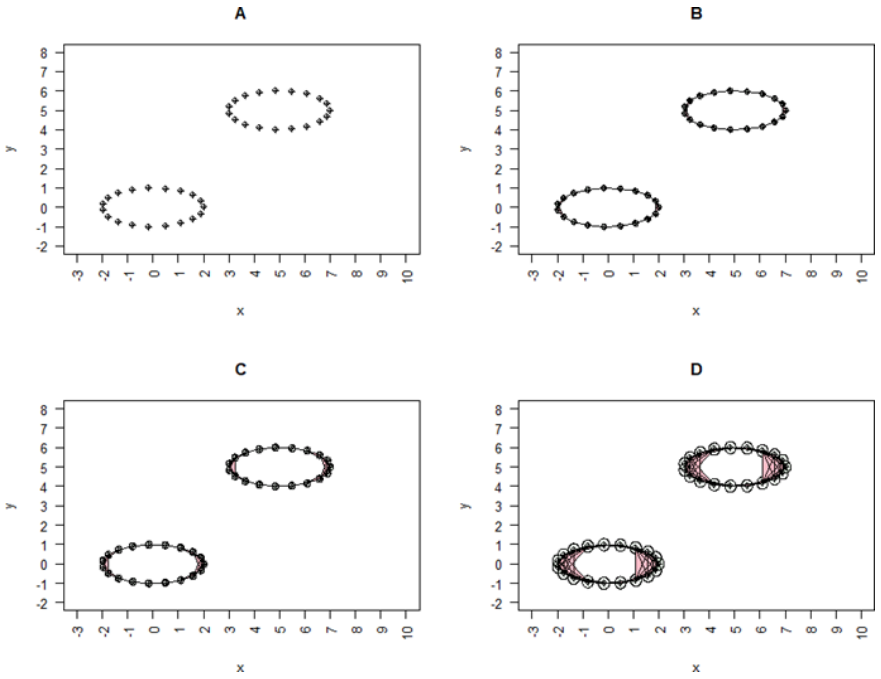

Figura 3: Ejemplo de cómputo del complejo de Rips
A. $\epsilon = 0$; B. $\epsilon = 0.5$; C. $\epsilon = 1$; D. $\epsilon = 1.8$.

2.2.2. Homología de persistencia

Los grupos de homología, denotados como H_n, se calculan para cada $R_\epsilon(S)$ en la filtración, donde *n* corresponde a la dimensión de las características de interés. Por ejemplo, H_0 rastrea los componentes conexos, H_1 rastrea los bucles o agujeros, y H_2 rastrea vacíos o cavidades. Los cambios en estos grupos, a medida que ϵ aumenta, proporcionan información sobre la estructura topológica de S.

La homología de persistencia estudia entonces la "duración" de estas características a lo largo de la filtración. Para cada característica topológica, se puede definir un tiempo de nacimiento ϵ_b, que corresponde al momento en que la característica aparece por primera vez en la filtración, y un tiempo de muerte ϵ_d, que corresponde al momento en que la característica se llena o pasa a formar parte de una característica más grande. La persistencia de una característica se define como $\epsilon_d - \epsilon_b$, lo que representa cuánto tiempo existe la característica a medida que ϵ varía. Estos intervalos a menudo se representan en un diagrama de persistencia o un código de barras, donde cada característica se muestra como un punto (ϵ_b, ϵ_d) o como una barra que abarca desde ϵ_b hasta ϵ_d.

El cálculo de la homología de persistencia se basa en algoritmos eficientes que pueden manejar las grandes estructuras combinatorias que surgen de la filtración de complejos. Un avance importante en este campo fue la introducción del concepto de pares de persistencia (ϵ_b, ϵ_d) para cada característica, que se pueden calcular de manera eficiente mediante

técnicas de reducción matricial. Estos algoritmos permiten la aplicación práctica de la homología de persistencia a conjuntos de datos del mundo real, lo que facilita el análisis de estructuras de datos complejas en diversos campos. La Figura 4 muestra un ejemplo de homología de persistencia.

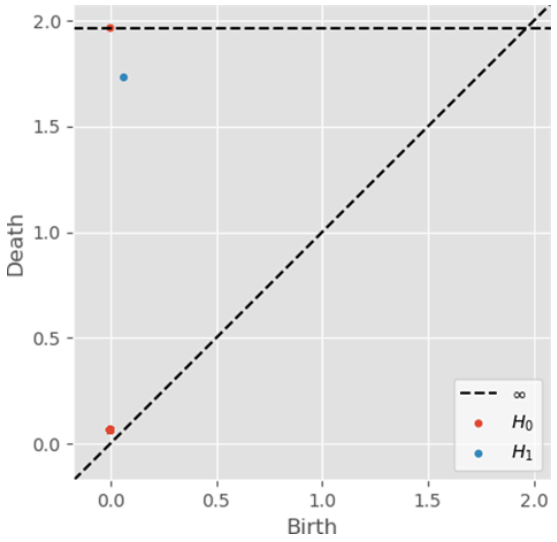

Figura 4: Ejemplo de Homología de Persistencia.

Biblioteca
UNIVERSITAS Miguel Hernández

2.2.3. Paisajes de persistencia

Si bien los diagramas de persistencia y los códigos de barras son herramientas poderosas, no siempre son fáciles de trabajar directamente, especialmente en aplicaciones de análisis estadístico o aprendizaje automático. Los paisajes de persistencia transforman la información contenida en los diagramas de persistencia en una forma funcional que es más adecuada para dichas aplicaciones.

Un paisaje de persistencia transforma esta información en una función continua y lineal por tramos, lo que facilita la aplicación de métodos estadísticos y computacionales. Matemáticamente, un paisaje de persistencia se define como $\lambda: N \times R \rightarrow R \cup \{S\}$, donde cada $\lambda_k(t)$ representa la k-ésima capa del paisaje. Dado un diagrama de persistencia que consiste en puntos (b_i, d_i) donde b_i y d_i representan los tiempos de nacimiento y muerte de la i-ésima característica topológica, el paisaje de persistencia se construye de la siguiente manera:

- Transformar pares de persistencia en funciones: Cada punto (b_i, d_i) en el diagrama de persistencia se transforma en una función lineal por tramos $f_i(t)$ definida por:

$$f_i(t) = max\{0, min\{t - b_i, d_i - t\}\}.$$

Esta función alcanza su máximo en $\frac{b_i+d_i}{2}$ con una altura de $\frac{d_i-b_i}{2}$, que corresponde a la persistencia (duración) de la característica. Es 0 fuera del intervalo $[b_i, d_i]$.

- Construcción de capas: La k-ésima capa del paisaje de persistencia, $\lambda_k(t)$, se define como el k-ésimo valor más grande de $f_i(t)$ entre todas las características i en cada punto t. $\lambda_k(t) = 0$ cuando hay menos de k características existentes en el tiempo t. Formalmente, sea $\Lambda(t) = \{f_i(t) \mid f_i(t) > 0\}$ el conjunto de todos los valores no nulos de $f_i(t)$ en el tiempo t, y sea $\Lambda_k(t)$ el k-ésimo elemento más grande en $\Lambda(t)$, entonces

$$\lambda_k(t) = \begin{cases} \Lambda_k(t), & \text{si } |\Lambda(t)| \geq k \\ 0, & \text{en caso contrario} \end{cases}$$

Esto da como resultado una colección de funciones $\{\lambda_k\}$ que juntas forman el paisaje de persistencia. La primera capa $\lambda_1(t)$ captura las características más persistentes, y las capas subsecuentes $\lambda_k(t)$ capturan características menos persistentes. El paisaje de persistencia es un invariante completo y estable de los datos, lo que significa que captura toda la información del diagrama de persistencia de una manera estable frente a pequeñas perturbaciones en los datos.

Esta formulación permite que los paisajes de persistencia codifiquen la misma información topológica que los diagramas de persistencia, pero en un formato adecuado para operaciones algebraicas y análisis estadístico. Por ejemplo, es posible calcular fácilmente el promedio de múltiples paisajes, aplicar algoritmos de aprendizaje automático o realizar pruebas de hipótesis, lo cual no es sencillo con los diagramas de persistencia originales debido a su naturaleza basada en conjuntos. La Figura 5 muestra un ejemplo de un paisaje de persistencia utilizando los datos del círculo mencionados en las figuras anteriores.

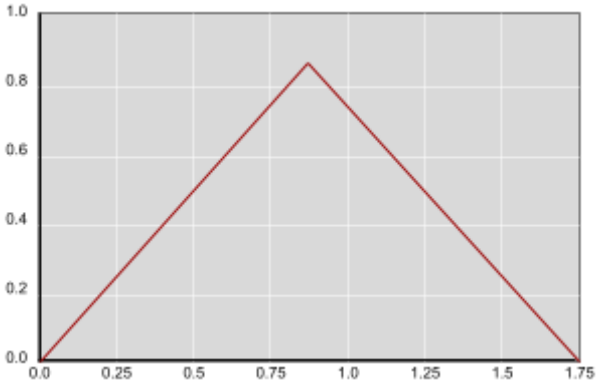

Figura 5: Grado 1 del Paisaje de Persistencia de los datos del círculo.

La principal ventaja de los paisajes de persistencia radica en su estabilidad: pequeños cambios en los datos de entrada conducen a pequeños cambios en el paisaje, lo que los

hace robustos frente al ruido y a perturbaciones menores. Esta propiedad es crucial en aplicaciones de ciencia de datos y aprendizaje automático, donde los datos a menudo contienen ruido o están sujetos a errores de medición. La estabilidad de los paisajes de persistencia garantiza que las señales topológicas que capturan sean significativas y reflejen la estructura subyacente de los datos, en lugar de ser artefactos del azar o del ruido.

2.3. TÉCNICAS DE ANÁLISIS ALTERNATIVAS AL TDA

El análisis de señales EEG ha sido fundamental en la investigación de trastornos neurológicos y en el desarrollo de tecnologías para el diagnóstico y la monitorización. A lo largo del tiempo, se han desarrollado diversos métodos y sistemas alternativos que complementan o incluso sustituyen las técnicas tradicionales de análisis de EEG. Este apartado examina algunas de las principales tecnologías y enfoques disponibles en el mercado.

2.3.1. Métodos de Aprendizaje Automático

El aprendizaje automático ha transformado el análisis de EEG al proporcionar herramientas poderosas para el procesamiento e interpretación de grandes volúmenes de datos. Entre los enfoques más destacados se incluyen:

- Redes Neuronales Artificiales (ANN): Las Redes Neuronales Artificiales [9] (ANN, por sus siglas en inglés) son modelos computacionales inspirados en la estructura del cerebro humano. Están compuestas por capas de nodos o neuronas interconectadas que procesan información a través de pesos sinápticos ajustables. El modelo básico de una ANN se puede describir matemáticamente como:

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right)$$

  donde:
  - x_i son las entradas
  - w_i son los pesos asociados a cada entrada
  - b es el sesgo
  - f es la función de activación, como ReLU ($f(x) = max(0,x)$) o sigmoide ($f(x) = \frac{1}{1+e^{-x}}$)

El uso de ANN en EEG ha demostrado mejoras en la precisión diagnóstica mediante el análisis de patrones no lineales en las señales. Por ejemplo, las arquitecturas convolucionales (CNN) son especialmente efectivas para procesar datos en series temporales debido a su capacidad para capturar características locales.

- Máquinas de Vectores de Soporte (SVM): Las Máquinas de Vectores de Soporte [10] (SVM, por sus siglas en inglés) son modelos supervisados que encuentran el hiperplano óptimo que separa dos o más clases en el espacio de características. Matemáticamente, SVM optimiza el problema:

$$min_{w,b} \frac{1}{2}||w||^2, \; tal\ que\ \forall i, \; y_i(w \cdot x_i + b) \geq 1$$

donde:

- w es el vector de pesos
- b es el sesgo
- x_i son los datos de entrada (características EEG)
- y_i son las etiquetas de las clases (+1 o -1)

Las SVM han mostrado eficacia al manejar datos de alta dimensión y ruidos, lo cual es común en el análisis de EEG. Su implementación también permite el uso de núcleos (kernels) para mapear características no lineales en espacios de mayor dimensión, mejorando la capacidad de separación.

- Bosques Aleatorios (Random Forests): Los Bosques Aleatorios [11] son algoritmos de aprendizaje supervisado basados en la construcción de múltiples árboles de decisión. Cada árbol clasifica de forma independiente, y el modelo final realiza una predicción combinando los resultados (votación mayoritaria para clasificación o promedio para regresión). La predicción de un árbol de decisión, en el caso de la regresión, se puede formular como:

$$\widehat{y} = \frac{1}{M} \sum_{i=1}^{M} h_i(x)$$

donde:

- $h_i(x)$ es la predicción del i-ésimo árbol
- M es el número total de árboles

En el análisis de EEG, los Bosques Aleatorios son robustos frente a ruido y datos faltantes, características frecuentes en estudios neurológicos. Además, permiten

identificar las características más importantes para la clasificación, ayudando a reducir la dimensionalidad del problema y a interpretar mejor los resultados.

- Regresión Logística: La regresión logística [12] es un método de aprendizaje supervisado ampliamente utilizado para problemas de clasificación binaria en el análisis de EEG. Este modelo busca predecir la probabilidad de que una instancia (por ejemplo, un segmento de señal EEG) pertenezca a una clase específica, como "actividad normal" o "evento epiléptico", basándose en características extraídas de la señal. La probabilidad predicha se puede expresar como:

$$P(y = 1|x) = \sigma(w^T x + b) = \frac{1}{1+e^{-(w^T x+b)}}$$

donde:

- w es el vector de pesos asignados a cada característica
- x es el vector de características extraídas de la señal EEG (como potencia en bandas de frecuencia o valores de conectividad)
- b es el sesgo (bias)
- σ es la función sigmoide

En el análisis de EEG, la regresión logística es especialmente útil debido a:

- Interpretabilidad: Permite identificar qué características tienen mayor peso en la clasificación, lo que es crucial para estudios neurológicos.
- Simplicidad: Es computacionalmente eficiente, lo que facilita su uso con conjuntos de datos grandes y con características previamente seleccionadas.
- Generalización: Aunque lineal, combinada con transformaciones de características (como polinomiales o kernels), puede abordar relaciones no lineales presentes en señales EEG complejas.

Este enfoque es ideal para estudios en los que es importante equilibrar precisión y facilidad de interpretación, como en la detección de trastornos neurológicos o la clasificación de estados cognitivos.

2.3.2. Análisis de Tiempo-Frecuencia

El análisis de tiempo-frecuencia es esencial para capturar las variaciones en las señales EEG a lo largo del tiempo [13]. Este tipo de análisis proporciona una comprensión más profunda de cómo las oscilaciones cerebrales se distribuyen y cambian dinámicamente,

facilitando la identificación de patrones relacionados con funciones cognitivas o patologías específicas.

- Transformada de Fourier: La Transformada de Fourier (TF) [14] es una herramienta clásica que descompone señales EEG en sus componentes frecuenciales, proporcionando información sobre la amplitud y la potencia de las oscilaciones en diferentes bandas de frecuencia, como delta (0.5–4 Hz), theta (4–8 Hz), alfa (8–13 Hz), beta (13–30 Hz) y gamma (>30 Hz). Esta técnica asume que las señales son estacionarias, lo que limita su capacidad para analizar dinámicas temporales. Por ejemplo, una señal EEG y su Transformada de Fourier muestran la distribución de potencia en distintas frecuencias, lo que resulta de utilidad para detectar actividad rítmica estable.

- Ondículas (Wavelets): El análisis mediante ondículas [15] supera la limitación de la Transformada de Fourier al permitir un análisis conjunto de las dimensiones temporales y frecuenciales de las señales EEG [16]. Las ondículas, como la de Morlet, son ideales para capturar eventos transitorios y rápidos en la actividad cerebral, como espigas epilépticas o sincronizaciones/desincronizaciones inducidas por estímulos [17]. Por ejemplo, un evento transitorio detectado en el EEG, como un potencial evocado, puede ser estudiado utilizando técnicas de análisis basadas en ondículas. Este enfoque permite descomponer la señal en componentes que representan diferentes escalas de tiempo y frecuencia, proporcionando una representación más detallada de cómo la energía del evento se distribuye a lo largo del tiempo. De esta manera, las ondículas facilitan la visualización de la evolución temporal de la energía asociada al evento en el dominio de las frecuencias, lo cual resulta especialmente útil para identificar patrones específicos o cambios dinámicos que no serían evidentes con métodos de análisis tradicionales.

2.3.3. Tecnologías de Monitoreo y Equipos

Los avances en tecnología han llevado al desarrollo de equipos y dispositivos más sofisticados para la adquisición y análisis de EEG. Estas innovaciones no solo mejoran la calidad y la cantidad de datos recogidos, sino que también amplían las posibilidades de uso del EEG en entornos clínicos, de investigación y cotidianos.

- Electrodos de EEG Avanzados: Los sistemas de electrodos modernos [18] han evolucionado significativamente, abarcando tecnologías como los electrodos secos, que no requieren gel conductor, y los cascos de EEG, diseñados para una colocación rápida y precisa [19]. Estas tecnologías mejoran la comodidad del usuario, reducen los tiempos de preparación y minimizan artefactos en las señales recolectadas [20]. Además, los avances en miniaturización han permitido el

desarrollo de electrodos flexibles que se adaptan a la morfología del cráneo, mejorando la calidad de las mediciones [21].

Figura 6: Ejemplo de electrodos de EEG Avanzados

- Sistemas de Adquisición de Datos Portátiles: Los dispositivos portátiles para EEG [22] han transformado el campo al permitir la monitorización continua de la actividad cerebral en tiempo real. Estos sistemas, a menudo inalámbricos, son ideales para estudios longitudinales, ya que facilitan la recopilación de datos en entornos naturales, como el hogar o el lugar de trabajo. Además, los avances en procesamiento de señales en tiempo real han mejorado la capacidad de los dispositivos para detectar y analizar patrones cerebrales complejos, lo que los hace útiles en aplicaciones como el diagnóstico temprano de epilepsia, el monitoreo del sueño y el control de prótesis mediante señales cerebrales [23].

Figura 7: Ejemplo de un sistema de adquisición portátil de datos

2.4.- APLICACIONES DE MONITOREO DE LA ACTIVIDAD CEREBRAL

El software de análisis EEG proporciona herramientas avanzadas para el procesamiento, análisis y visualización de datos EEG, facilitando la interpretación de la actividad cerebral en diversas aplicaciones clínicas y de investigación. Estas herramientas son fundamentales

para transformar grandes volúmenes de datos crudos en información significativa que pueda ser utilizada para diagnóstico, investigación científica, e incluso aplicaciones prácticas en neurotecnología. Entre las plataformas comerciales más destacadas se encuentran:

2.4.1. Neurofeedback

El neurofeedback es una técnica avanzada que utiliza retroalimentación en tiempo real para modificar y optimizar la actividad cerebral. Esta metodología se basa en el análisis continuo de datos EEG para ofrecer a los pacientes información instantánea sobre su actividad cerebral, permitiendo a los analistas médicos ajustar y mejorar sus patrones neuronales [24].

El principio fundamental del neurofeedback es que los patrones de actividad cerebral pueden ser regulados a través de la retroalimentación proporcionada. Con el uso de un sistema EEG, las ondas cerebrales del paciente se monitorizan y se presentan en tiempo real a través de interfaces visuales o auditivas. Esta retroalimentación permite a los pacientes observar en qué medida sus pensamientos y estados emocionales influyen en su actividad cerebral, facilitando así el aprendizaje y la auto-regulación de estos patrones [25].

En el ámbito clínico, el neurofeedback ha demostrado ser eficaz para tratar una amplia gama de trastornos neurológicos y psicológicos. En el caso del TDAH (Trastorno por Déficit de Atención e Hiperactividad), por ejemplo, los pacientes han mostrado mejoras significativas en la atención y el control de impulsos mediante la regulación de frecuencias cerebrales específicas. Además, el neurofeedback se ha mostrado útil en el tratamiento de trastornos de ansiedad, insomnio y alteraciones del estado de ánimo [26].

Además de sus aplicaciones clínicas, el neurofeedback se utiliza para potenciar el rendimiento cognitivo en individuos sanos. Esta técnica puede optimizar funciones cognitivas como la memoria, la concentración y la capacidad de aprendizaje. Los atletas y profesionales en entornos altamente competitivos a menudo recurren al neurofeedback para maximizar su rendimiento y alcanzar sus objetivos, aprovechando la capacidad de esta técnica para mejorar las funciones cerebrales relacionadas con el rendimiento [26].

2.4.2. Software de Análisis EEG

El software de análisis EEG proporciona herramientas avanzadas para el procesamiento, análisis y visualización de datos EEG, facilitando la interpretación de la actividad cerebral en diversas aplicaciones clínicas y de investigación. Entre las plataformas comerciales más destacadas se encuentran:

- BrainVision: BrainVision es ampliamente conocida por su suite de soluciones que abordan la adquisición, el análisis y la visualización de datos EEG. Su sistema permite descomponer las señales cerebrales en componentes específicos y visualizar eventos cerebrales, lo cual es útil en diversas aplicaciones clínicas. Sin embargo, el nuevo software con TDA que hemos desarrollado lleva este análisis un paso más allá al ofrecer una descomposición adaptativa de las señales cerebrales, capaz de identificar patrones no lineales complejos. Esta capacidad es particularmente relevante en el contexto de enfermedades neurodegenerativas como el Alzheimer y el Parkinson, donde las alteraciones en las ondas cerebrales pueden ser muy sutiles. La incorporación de TDA también permite un análisis más detallado y preciso en tiempo real, lo cual mejora la capacidad de detectar anomalías en la actividad cerebral de los pacientes de manera inmediata. [27].

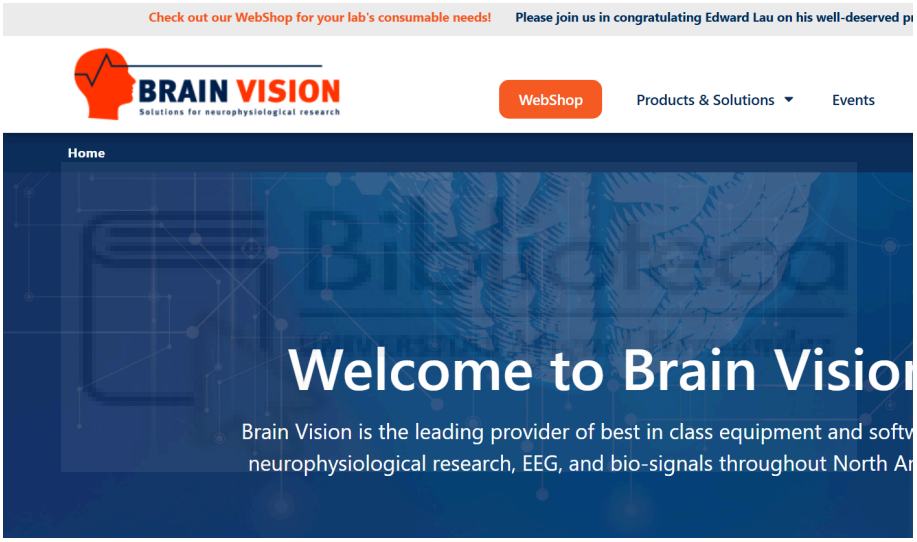

Figura 8: Página web de la aplicación BrainVision

- NeuroSky: NeuroSky se enfoca principalmente en dispositivos portátiles para aplicaciones de consumo y bienestar, su tecnología permite un acceso básico a los datos EEG. Este enfoque ha sido útil en áreas como la educación y el desarrollo personal, pero en el contexto clínico y de investigación sobre enfermedades neurodegenerativas, el análisis detallado de las señales cerebrales es crucial. Aquí es donde nuestro software, con su capacidad para procesar y analizar patrones cerebrales complejos mediante TDA, ofrece una ventaja clara. El TDA no solo mejora la precisión del análisis, sino que también permite la integración de datos en tiempo real, algo esencial para el monitoreo continuo de pacientes con trastornos neurodegenerativos. Así, nuestro software no solo es más avanzado en términos de capacidad de análisis, sino que también aporta un nivel de detalle y resolución que no es posible con plataformas más orientadas a consumidores como NeuroSky [28].

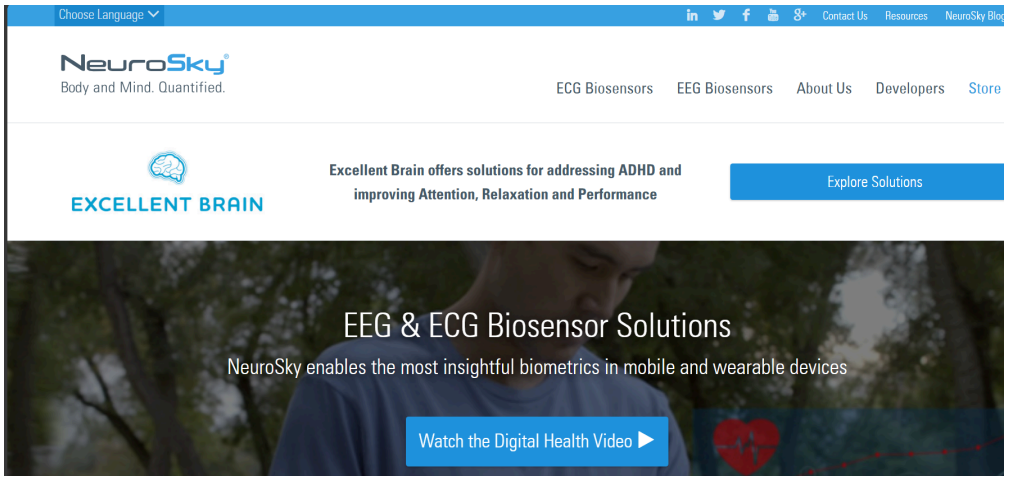

Figura 9: Página web de la aplicación NeuroSky.

- Emotiv: Emotiv proporciona una plataforma completa que abarca desde la adquisición de EEG hasta el análisis y la visualización de los datos. Su software está orientado a aplicaciones en neurociencia, salud mental y bienestar, y permite la personalización y análisis en tiempo real. Sin embargo, el análisis de patrones cerebrales de alta complejidad, como los que se presentan en enfermedades neurodegenerativas, puede ser desafiante con las herramientas convencionales. Nuestro software, al integrar TDA, permite un análisis mucho más preciso de estos patrones no lineales en la actividad cerebral, lo cual es fundamental para la detección temprana y el seguimiento de enfermedades como el Alzheimer y el Parkinson. Además, la capacidad de integrar datos EEG con otras métricas biométricas es otra característica que nuestro software maneja de forma avanzada, proporcionando una visión holística y más completa del estado del paciente [29].

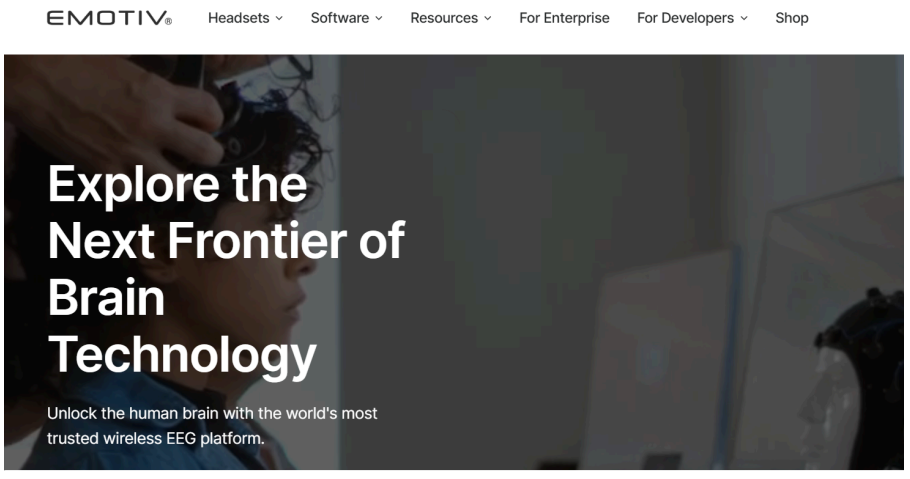

Figura 10. Página web de la app Emotiv.

Capítulo 3
Hipótesis de trabajo

En este capítulo se exploran las hipótesis y fundamentos que guían el desarrollo del proyecto, abarcando los lenguajes de programación, herramientas de desarrollo, sistemas gestores de bases de datos, arquitecturas, metodologías de diseño, normas y estándares aplicables. Este proyecto se basa exclusivamente en el uso de Python y diversas bibliotecas especializadas en el análisis de datos EEG y la aplicación de técnicas avanzadas de análisis topológico.

3.1. EL LENGUAJE PYTHON. LIBRERÍAS

Python es un lenguaje de programación de alto nivel, ampliamente utilizado en la ciencia de datos y en la investigación neurocientífica debido a su simplicidad y versatilidad. El uso de Python en la ciencia de datos y la investigación neurocientífica ha ganado una gran popularidad debido a su versatilidad, simplicidad y gran cantidad de bibliotecas

especializadas. Python se ha convertido en una herramienta esencial en muchos campos de la investigación, incluyendo el análisis de datos EEG, el aprendizaje automático (machine learning) y la inteligencia artificial (IA). Su sintaxis clara y su capacidad para manejar tareas complejas lo convierten en una opción ideal para investigadores en neurociencia, especialmente al trabajar con grandes volúmenes de datos y modelos predictivos.

En el contexto de este proyecto, Python se utiliza para el análisis de señales EEG, combinando técnicas avanzadas de procesamiento de datos con enfoques de aprendizaje automático y análisis topológico, como el uso de la topología persistente y la descomposición adaptativa. Estas técnicas ayudan a identificar patrones complejos y no lineales en las señales cerebrales, lo que es crucial para la detección temprana de enfermedades neurodegenerativas como el Alzheimer. Las librerías de Python que específicamente se han utilizado para este proyecto son las siguientes [30]:

- numpy: numpy es una biblioteca fundamental para el cálculo numérico en Python. Ofrece soporte para grandes matrices y matrices multidimensionales, junto con una colección de funciones matemáticas para operar con estos datos.

- matplotlib: para la visualización de datos en Python, matplotlib permite crear gráficos estáticos, animados e interactivos, lo que resulta muy útil para la representación visual de resultados en investigaciones neurocientíficas.

- Scipy: Scipy complementa a numpy al proporcionar funciones adicionales para el análisis científico y técnico, incluyendo algoritmos para la optimización, integración, interpolación y procesamiento de señales.

- scikit-learn: el paquete scikit-learn es una biblioteca de aprendizaje automático en Python que ofrece herramientas para la clasificación, regresión y agrupamiento de datos, facilitando el desarrollo de modelos predictivos en neurociencia.

- persim: La librería persim se utiliza para la visualización y el análisis de datos topológicos persistentes. Es útil para interpretar características topológicas de los datos, una tarea relevante en el análisis de señales EEG.

- ripser: para el cálculo de la persistencia homológica, Python cuenta con la librería ripser, que ayuda a identificar características topológicas en conjuntos de datos. Es particularmente valiosa en el análisis de datos complejos como los registros EEG.

- gudhi: para la topología computacional gudhi proporciona herramientas para el análisis de datos topológicos, incluyendo la persistencia homológica y la construcción de complejos simpliciales.

- giotto-TDA: giotto-tda es una biblioteca dedicada al análisis topológico de datos, integrando herramientas para el procesamiento y visualización de características topológicas en datos multidimensionales.

- mne: La librería mne se especializa en el procesamiento de datos neurofisiológicos, como EEG y MEG (magnetoencefalografía). Proporciona herramientas para el análisis, visualización y procesamiento de señales neuroeléctricas.

3.2. GitHub

GitHub es una plataforma basada en la web para el control de versiones y la colaboración en el desarrollo de software. GitHub utiliza Git, un sistema de control de versiones distribuido, que permite a los desarrolladores trabajar de manera colaborativa en proyectos de cualquier tamaño [32].

GitHub es ampliamente utilizado en la comunidad científica y de desarrollo de software debido a su facilidad para gestionar proyectos, su capacidad de construir herramientas de integración continua y su utilidad como repositorio central para el código. En el contexto de este proyecto, GitHub se utiliza para:

- Control de versiones: Garantiza que todos los cambios en el código se rastrean, lo que facilita la colaboración y la revisión del trabajo.

- Colaboración en equipo: GitHub facilita el trabajo en equipo, permitiendo a varios desarrolladores contribuir al mismo proyecto desde ubicaciones remotas.

- Revisiones y documentación: GitHub permite que otros miembros del equipo revisen los cambios mediante el comando “pull requests”, lo que mejora la calidad del código. Además, facilita la creación de documentación [31] clara y accesible para los usuarios y colaboradores a través de archivos README y Wikis.

- Automatización y despliegue: Con las herramientas de integración continua como GitHub Actions, es posible automatizar las pruebas del código y el despliegue de las aplicaciones, asegurando que el proyecto esté siempre actualizado y funcionando correctamente.

GitHub también actúa como una plataforma de referencia, permitiendo compartir código con la comunidad de código abierto, lo que es especialmente útil para proyectos que buscan reproducibilidad y transparencia en los resultados científicos.

3.3. EL REPOSITORIO DE DATOS OPENNEURO

OpenNeuro es una plataforma innovadora diseñada para el almacenamiento, intercambio y acceso a datos neurocientíficos, particularmente en el campo de las neuroimágenes y, más específicamente, las neuroimágenes funcionales y estructurales como los registros de EEG, fMRI, MEG y otros. Esta plataforma tiene como objetivo principal democratizar el acceso a los datos científicos y promover la investigación colaborativa, algo esencial para avanzar en la comprensión de diversas enfermedades cerebrales y en el desarrollo de nuevas técnicas y herramientas de diagnóstico y tratamiento [32].

3.3.1. Características Clave de OpenNeuro

Las características más relevantes de esta plataforma son las siguientes:

1. **Acceso abierto a los datos:** OpenNeuro es una plataforma de acceso abierto, lo que significa que los investigadores y científicos tienen la posibilidad de compartir y acceder a grandes volúmenes de datos neurocientíficos sin restricciones. Esta accesibilidad contribuye a un intercambio rápido y eficiente de conocimientos, y permite que cualquier persona en la comunidad científica pueda acceder a los datos, analizarlos, compararlos y aportar nuevas perspectivas basadas en sus propios análisis. El acceso libre a estos datos es fundamental para la replicación de estudios, un principio básico de la ciencia, y para la validación de resultados, lo que ayuda a evitar la "publicación selectiva" o la falta de transparencia.

2. **Variedad de tipos de datos:** OpenNeuro facilita el almacenamiento y el intercambio de diversos tipos de neuroimágenes, con un enfoque particular en los datos de EEG (electroencefalograma). El EEG es una de las herramientas más comunes en neurociencia para estudiar la actividad eléctrica del cerebro. Sin embargo, OpenNeuro no se limita a EEG, sino que también ofrece soporte para datos de otras modalidades de neuroimágenes como fMRI (resonancia magnética funcional), MEG (magnetoencefalografía) y otros tipos de registros neurofisiológicos. Esta diversidad de datos almacenados en la plataforma es clave para las investigaciones de intermodalidades, que buscan correlacionar información de diferentes tipos de neuroimágenes para obtener una comprensión más completa de la actividad cerebral.

3. **Estandarización de datos:** Uno de los retos más grandes en la neurociencia, y en particular en el análisis de neuroimágenes, es la heterogeneidad de los datos. Los datos neurocientíficos suelen ser recolectados utilizando diferentes tecnologías,

configuraciones de equipos, protocolos y formatos de archivo, lo que puede dificultar el análisis comparativo entre estudios. OpenNeuro aborda este problema al ofrecer una infraestructura que estandariza los datos siguiendo normativas y protocolos bien definidos. Utilizando el estándar BIDS (Brain Imaging Data Structure), OpenNeuro asegura que todos los datos estén organizados y etiquetados de manera consistente, lo que facilita su uso, interpretación y análisis por otros investigadores. Esto es esencial para mejorar la interoperabilidad de los datos y garantizar que los resultados obtenidos sean reproducibles.

4. **Facilita la investigación colaborativa:** Esta plataforma está diseñada para fomentar la colaboración entre investigadores de diferentes partes del mundo. Dado que los datos están accesibles de manera abierta, OpenNeuro permite que investigadores de distintas instituciones y laboratorios colaboren y utilicen los mismos datos para hacer avances conjuntos en neurociencia. Esto es particularmente valioso en áreas como el estudio de enfermedades neurodegenerativas (como Alzheimer, Parkinson o esclerosis múltiple), donde los investigadores pueden compartir sus datos de EEG, comparar resultados, e incluso combinar conjuntos de datos de diferentes investigaciones para obtener conclusiones más robustas y de mayor alcance.

5. **Interfaz de usuario intuitiva:** OpenNeuro se caracteriza por su interfaz de usuario accesible y fácil de usar, tanto para cargar datos como para acceder a ellos. Los investigadores pueden subir sus conjuntos de datos a la plataforma, garantizando la calidad y la integridad de los mismos, y también pueden buscar y descargar datos de otros estudios que les sean relevantes. Gracias a su diseño amigable, incluso los investigadores que no tienen experiencia previa en el manejo de plataformas de almacenamiento de datos pueden usar OpenNeuro con relativa facilidad, lo que mejora la accesibilidad de los datos para una mayor cantidad de científicos.

6. **Aceleración del avance científico:** Al proporcionar una plataforma abierta y estandarizada para el intercambio de datos, OpenNeuro acelera el progreso científico, permitiendo que se desarrollen modelos y teorías más rápidamente, ya que los investigadores tienen acceso inmediato a grandes cantidades de datos, sin tener que esperar a recolectar los suyos propios. Además, los datos almacenados en la plataforma son frecuentemente utilizados para entrenar modelos de aprendizaje automático (machine learning), una herramienta que ha revolucionado la neurociencia al permitir identificar patrones complejos en los datos que son difíciles de detectar mediante técnicas tradicionales. Con un acceso rápido a datos preexistentes, se pueden entrenar y evaluar más rápidamente modelos de Inteligencia Artificial, lo que mejora enormemente la eficiencia del proceso de investigación.

3.3.2. Metadatos de OpenNeuro

A continuación, se detalla cada uno de los metadatos de OpenNeuro, que también se pueden consultar de forma más esquemática en la tabla 1:

- **ID del sujeto**: Es un identificador único que permite rastrear el conjunto de datos sin comprometer la privacidad del sujeto.

- **Edad y género**: Son datos demográficos que ayudan a contextualizar los resultados de la investigación. Estos metadatos permiten que los investigadores analicen si los patrones cerebrales son consistentes o varían según la edad, el género u otras características del sujeto.

- **Condición médica**: Identificar las condiciones médicas asociadas a cada conjunto de datos es importante, especialmente cuando se trabaja con grupos clínicos como pacientes con enfermedades neurodegenerativas. Estos metadatos pueden proporcionar información sobre la relación entre las condiciones médicas y la actividad cerebral registrada.

- **Fecha de adquisición**: La fecha en que se recolectaron los datos es crucial para realizar un seguimiento temporal de los experimentos, ya que los resultados pueden depender de la tecnología y las metodologías empleadas en diferentes momentos.

- **Modalidad de imagen**: Especifica el tipo de imagen o señal obtenida. En este caso, los datos pueden ser de EEG, fMRI, MEG u otros tipos de neuroimágenes, lo cual es esencial para definir qué tipo de análisis se puede realizar.

- **Frecuencia de muestreo y número de canales EEG**: Estos parámetros son vitales para el análisis de los datos, ya que afectan la resolución temporal y espacial de las señales EEG.

- **Técnica de filtrado**: Los filtros aplicados a los datos EEG ayudan a eliminar el ruido y a mejorar la calidad de la señal, lo que influye en la precisión del análisis posterior.

- **Condiciones experimentales y anotaciones de eventos**: Estos metadatos detallan las condiciones experimentales en las que se realizaron las mediciones y los eventos específicos (como estímulos) que ocurrieron durante el experimento. Esta información es esencial para correlacionar las respuestas cerebrales con los estímulos experimentales.

- **Formato de archivo**: El formato de los datos es importante para la compatibilidad con diferentes herramientas y plataformas de análisis de datos. El estándar BIDS (Brain Imaging Data Structure) es ampliamente utilizado para la organización y el almacenamiento de datos de neuroimágenes.

Tabla 1: Descripción de los metadatos que tiene el repositorio OpenNeuro.

Categoría Metadato	Descripción	Ejemplo
ID del Sujeto	Identificador único del sujeto o participante en el estudio.	S001, P002
Edad	Edad del sujeto en años.	45
Género	Género del sujeto.	Masculino, Femenino
Condición Médica	Información sobre las condiciones médicas del sujeto (si aplica).	Enfermedad de Alzheimer, Sin diagnóstico
Fecha de Adquisición	Fecha en que se adquirieron los datos EEG u otras neuroimágenes.	2023-08-15
Modalidad de Imagen	Tipo de neuroimagen o señal registrada.	EEG, fMRI, MEG
Frecuencia de Muestreo (Hz)	Frecuencia a la que se capturan las señales EEG.	1000 Hz
Número de Canales EEG	Número de electrodos utilizados para registrar las señales EEG.	64, 128
Técnica de Filtrado	Información sobre los filtros aplicados a las señales EEG (si aplica).	Filtro paso banda de 0.5-50 Hz
Condiciones Experimentales	Descripción de las tareas o condiciones experimentales a las que estuvo expuesto el sujeto.	Tarea de memoria de trabajo, reposo, estímulos auditivos
Anotaciones de Eventos	Descripción de eventos o estímulos registrados durante el experimento.	Estímulo visual: 10 s, Estímulo auditivo: 5 s
Formato de Archivo	Formato en que se almacenan los datos (por ejemplo, BIDS, EDF, etc.).	BIDS, .fif (para MNE), .edf
Duración de la Grabación	Duración total de la grabación EEG u otros datos neurofisiológicos.	30 minutos
Autorización Ética	Número o referencia del comité de ética que aprobó el estudio.	Comité de ética XYZ, #2023-01
Número de Sesiones	Número de sesiones o grabaciones realizadas para un mismo sujeto o experimento.	3 sesiones
Tamaño de los Datos	Tamaño aproximado del archivo o conjunto de datos en bytes.	2 GB

- **Duración de la grabación**: El tiempo durante el cual se registraron las señales es fundamental para determinar cuántos datos están disponibles para el análisis y cómo se dividen esos datos en períodos de interés.

- **Autorización ética**: Los datos deben cumplir con los requisitos éticos y de privacidad establecidos por los comités de ética. Estos metadatos aseguran que los datos sean utilizados de manera responsable y conforme a las normativas internacionales.

- **Número de sesiones**: En algunos estudios, se recogen datos en múltiples sesiones, lo que puede proporcionar una visión más completa de los patrones cerebrales a lo largo del tiempo.

- **Tamaño de los datos**: El tamaño de los archivos o conjuntos de datos ayuda a los investigadores a evaluar la cantidad de almacenamiento necesario y facilita el manejo de grandes volúmenes de datos en las plataformas de análisis.

En este TFG los metadatos que se recogen deben ser aquellos estrictamente necesarios para el análisis de las señales y el estudio del cerebro. **No todos los metadatos son relevantes** para todas las etapas del procesamiento y análisis de los datos, y tomar demasiados metadatos puede resultar en un exceso de información que no aporta valor analítico directo. En muchos casos, esto puede complicar el proceso de análisis y aumentar la complejidad sin proporcionar una ventaja clara. El enfoque de nuestro proyecto se centra exclusivamente en la señal EEG debido a su alta resolución temporal, que permite un análisis preciso de las fluctuaciones rápidas en la actividad cerebral. Al limitarse a los datos EEG, se simplifica el procesamiento de la información, evitando la complejidad añadida de integrar otros tipos de datos que no son directamente relevantes para el análisis topológico. Esto hace que el proyecto sea más eficiente, centrado y específico en los patrones de actividad cerebral que se desean estudiar. Además, este enfoque facilita el análisis de enfermedades neurodegenerativas, ya que el EEG es capaz de detectar alteraciones en la dinámica cerebral que son clave para entender trastornos como el Alzheimer o la demencia Frontotemporal. Limitarse a esta señal también contribuye al cumplimiento de normativas éticas y de privacidad, minimizando la recolección de datos personales sensibles. En definitiva, trabajar exclusivamente con EEG garantiza un análisis más claro, directo y alineado con los objetivos del proyecto.

3.4. EL ENTORNO DATALAD

DataLad [32] es una herramienta diseñada específicamente para el versionado y la gestión de grandes conjuntos de datos científicos, orientada a mejorar el acceso, la organización y el control de datos que se utilizan en investigaciones científicas, especialmente aquellas que manejan grandes volúmenes de información. Basado en Git, uno de los sistemas de control de versiones más conocidos en el desarrollo de software, DataLad permite aplicar los mismos principios de control de versiones que se utilizan en el código fuente a los

conjuntos de datos. Este enfoque facilita un seguimiento detallado de las modificaciones que se realicen en los datos a lo largo del tiempo, proporcionando un registro completo de las versiones previas y actuales.

Figura 11. Sitio web datalad.org

Una de las características más destacadas de DataLad es su capacidad para manejar conjuntos de datos de gran tamaño que pueden estar distribuidos en la nube. Gracias a su integración con sistemas de almacenamiento como Git-annex, DataLad permite la gestión eficiente de archivos grandes, sin necesidad de almacenarlos directamente en los repositorios de Git, lo cual sería inviable debido a las limitaciones de tamaño. En lugar de eso, DataLad gestiona los archivos a través de enlaces simbólicos, lo que optimiza el almacenamiento y garantiza un acceso rápido y fluido a los datos almacenados en servidores remotos.

El control riguroso de versiones es otra ventaja clave de DataLad. Cuando se trabaja con grandes volúmenes de datos, como en tareas de limpieza o preprocesamiento, es común que los datos se alteren repetidamente. Con DataLad, cada cambio realizado en un conjunto de datos se puede guardar como una nueva versión, lo que facilita el seguimiento de alteraciones, la reversión a estados anteriores de los datos si fuera necesario y la gestión de los mismos a lo largo del tiempo. Este tipo de control de versiones es especialmente importante en el contexto de la investigación científica, donde la reproducibilidad de los experimentos y el acceso a versiones específicas de los datos son esenciales para garantizar la integridad de los resultados.

Al tratarse de una herramienta online, DataLad facilita la colaboración entre investigadores, independientemente de su ubicación geográfica. Los investigadores pueden compartir fácilmente los conjuntos de datos y sus versiones, garantizando que todos los colaboradores trabajen con los mismos datos en todo momento. Además, la capacidad de

almacenar los datos en la nube permite el acceso remoto a los conjuntos de datos desde cualquier dispositivo conectado a Internet, lo que mejora la accesibilidad y la flexibilidad en los proyectos de investigación.

Finalmente, el uso de DataLad contribuye a la reproducibilidad de los experimentos científicos. Dado que permite almacenar no solo los datos, sino también los metadatos sobre las versiones y los cambios realizados, los investigadores pueden garantizar que otros puedan reproducir sus experimentos en el futuro con los mismos datos y versiones exactas. Esto es fundamental para avanzar en la ciencia, ya que asegura que los resultados sean verificables y que otros investigadores puedan confirmar o expandir los hallazgos previos utilizando los mismos recursos.

Para utilizar la librería **DataLad** en Python, se siguen estos pasos:

1. **Conexión con la API de DataLad:** DataLad es una plataforma de código abierto para procesamiento y visualización de señales e imágenes. No requiere un registro específico ni una API-Key para su uso. Se puede acceder a la documentación y recursos en su sitio oficial Datalad Platform.

2. **Instalación de la librería:** Es posible instalar DataLad en un entorno local de Python utilizando el gestor de paquetes `pip`, ejecutando en la consola o terminal del sistema el siguiente comando:

   ```
   pip install datalad[qt]
   ```

   Este comando instalará DataLad junto con las dependencias necesarias para la interfaz gráfica basada en Qt. Si se prefiere una instalación sin la interfaz gráfica, se puede omitir `[qt]`:

   ```
   pip install datalad
   ```

3. **Ejemplo de código básico:** Una vez instalada la librería, ya es posible usarla en nuestro código Python. A continuación, se muestra un ejemplo básico de cómo importar y utilizar DataLad:

   ```
   import datalad as dl
   # Inicializar la sesión de DataLad
   session = dl.Session()
   # Cargar un conjunto de datos de ejemplo
   dataset = session.datasets.load('example_dataset')
   # Visualizar el conjunto de datos
   dataset.plot()
   ```

Este código importa la librería DataLad, inicializa una sesión, carga un conjunto de datos de ejemplo y los visualiza.

4. **Integración de DataLad en el proyecto (denominado topoEEG):** Los datos en OpenNeuro están organizados en repositorios públicos y están disponibles para su descarga en formatos estandarizados, como BIDS (Brain Imaging Data Structure). topoEEG utiliza DataLad para acceder a estos repositorios de manera eficiente, manteniendo un control completo sobre las versiones de los datos.

   Para acceder a los datos de **OpenNeuro** en **topoEEG**, primero hay que clonar el repositorio correspondiente a un conjunto de datos. Supongamos que el conjunto de datos de EEG de **OpenNeuro** tiene la siguiente URL:

   ```
   datalad clone https://openneuro.org/datasets/ds0003530
   ```

   Este comando descarga el repositorio de datos de OpenNeuro en nuestro entorno local, gestionado por **DataLad**. Esto asegura que los datos se gestionan con control de versiones, lo que facilita la colaboración, el acceso y la reproducción de los experimentos. Una vez que se ha clonado el repositorio de **OpenNeuro**, los datos de EEG estarán disponibles en el entorno local. En **topoEEG**, ya es posible realizar el procesamiento de estos datos EEG (limpieza, filtrado o segmentación de las señales, etc.).

   Después de clonar los datos desde **OpenNeuro**, se pueden **agregar los archivos de EEG** en el repositorio de trabajo **DataLad** para mantener un control de versiones sobre cualquier modificación que se realice. Por ejemplo, si se tiene un archivo `sub-01_task-rest_eeg.set`, se añadiría al repositorio con:

   ```
   datalad add sub-01_task-rest_eeg.set
   ```

   Uno de los pasos más importantes en el flujo de trabajo de topoEEG es el preprocesamiento de los datos EEG, que puede implicar operaciones como la eliminación de artefactos, el filtrado de señales, la segmentación, etc. Al realizar estos cambios, puedes guardar nuevas versiones de los datos utilizando el siguiente comando:

   ```
   datalad save.
   ```

   Tras realizar el procesamiento y guardado nuevas versiones de los datos, estos se pueden subir al repositorio remoto para compartirlos con otros colaboradores o equipos de investigación. Al usar **Git-annex** junto con **DataLad**, los datos grandes

se almacenan en la nube de manera eficiente, mientras que el repositorio Git almacena los enlaces a los archivos. Para subir los datos a un repositorio remoto, se utiliza el siguiente comando:

```
datalad push
```

En conclusión, el uso de **DataLad** para gestionar los datos provenientes de **OpenNeuro** en **topoEEG** permite un flujo de trabajo eficiente y reproducible. Al clonar los repositorios de **OpenNeuro**, agregar los datos EEG a un repositorio de **DataLad**, y gestionar las versiones mediante **datalad save** y **datalad push**, los investigadores pueden trabajar de manera colaborativa, asegurando que las versiones de los datos y los resultados del preprocesamiento sean fácilmente accesibles y reproducibles por otros colaboradores o para futuros trabajos propios. Este enfoque promueve la transparencia y la reproducibilidad en las investigaciones científicas relacionadas con el análisis de señales EEG.

3.5. MÁQUINA DE DESARROLLO

El desarrollo de las mejoras de dicha aplicación se ha realizado sobre un dispositivo HP ZBook. En la siguiente tabla muestra las características del mismo:

Tabla 2: Características de la máquina utilizada para la ejecución de los análisis.

EQUIPO	HP ZBook 15u G6
Sistema operativo	Windows 10
Arquitectura	64 bits
Procesador	I7-8665U
RAM	32 GB
Disco Duro	500 GB SSD

En el contexto de **topoEEG**, la lectura y escritura de los datos se realiza principalmente en la máquina que alberga el repositorio de **DataLad**. Esta máquina es responsable de gestionar y almacenar los datos a través de un repositorio controlado por **DataLad**. Los datos EEG, que pueden ser grandes en volumen, se encuentran en esta máquina centralizada, desde donde se accede a ellos de manera eficiente. Cuando se realiza un análisis, como el preprocesamiento o la segmentación de las señales EEG, estos procesos se llevan a cabo en el equipo local del investigador, pero los datos en sí se mantienen y gestionan en el repositorio remoto proporcionado por **DataLad**.

En cuanto al rendimiento, el cuello de botella principal en este flujo de trabajo sería el disco duro de la máquina que aloja el repositorio de **DataLad**, ya que es allí donde se almacenan y gestionan los archivos de datos. El acceso a los datos desde la máquina de

DataLad implica lectura y escritura en el disco, lo que podría convertirse en un límite si el almacenamiento o la velocidad de transferencia no es adecuada. Sin embargo, el resto del hardware, como la memoria RAM o el procesador en la máquina local del usuario, no sería un factor determinante para este tipo de software de análisis. De hecho, dado que el procesamiento se hace en el equipo local, la capacidad del hardware local es relevante únicamente para las operaciones de análisis, y no tanto para la gestión o transferencia de los datos, que es responsabilidad de la máquina **DataLad**. Por lo tanto, las diferencias de hardware no afectarían significativamente al rendimiento del software de análisis, más allá de los posibles cuellos de botella en el acceso a los datos almacenados.

Biblioteca
UNIVERSITAS Miguel Hernández

Capítulo 4
Metodología y resultados

Todo proyecto de desarrollo de software se sustenta en un ciclo de vida específico. En este capítulo se detalla el proceso de desarrollo de este proyecto, analizando el ciclo de vida adoptado, presentando un diagrama de Gantt con la duración de cada fase, y proporcionando información adicional sobre los requisitos, el diseño y la implementación de la aplicación.

4.1.- PLANIFICACIÓN DEL PROYECTO

En el contexto del análisis de datos EEG para la detección de enfermedades neurodegenerativas, como el Alzheimer, se puede seguir un ciclo de vida basado en el enfoque propuesto en este estudio. A continuación, se describen las etapas fundamentales de dicho ciclo, ilustradas en la Figura 12:

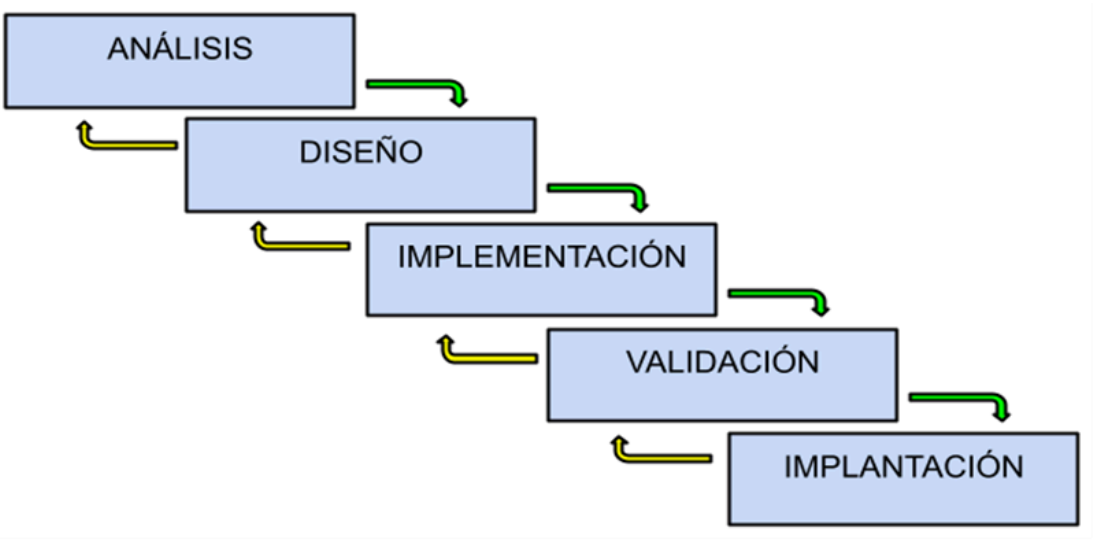

Figura 12: Ciclo de vida del análisis EEG con Deep Learning topológico.

- Análisis: En esta fase se recopilan y procesan los datos EEG de los participantes, lo que incluye la eliminación de artefactos utilizando métodos como el Análisis de Componentes Independientes (ICA) para mejorar la calidad de los datos y extraer información neural relevante.

- Diseño: Se diseña el modelo de aprendizaje profundo que integrará técnicas topológicas (TDL) para capturar patrones complejos en los datos EEG. Esto implica definir la arquitectura del modelo, como redes neuronales, que serán entrenadas con los datos preprocesados, identificando topologías relevantes del cerebro.

- Implementación: Se implementan los algoritmos y técnicas de TDL para transformar los datos EEG en representaciones topológicas, como los paisajes de persistencia, que facilitan la clasificación de los distintos estados clínicos (Alzheimer, demencia frontotemporal y sujetos sanos). En esta etapa, se traduce la información de los datos en un formato procesable por el modelo.

- Validación: El modelo entrenado se valida mediante métricas de rendimiento, como la precisión, sensibilidad y AUC, para verificar su capacidad de identificar correctamente los distintos estados de la enfermedad. Se realizan pruebas utilizando técnicas de validación cruzada y frameworks de Deep Learning.

- Implantación: Finalmente, se documenta el proceso y los resultados obtenidos, proporcionando información detallada sobre la configuración del modelo, las técnicas de preprocesamiento y las métricas de rendimiento. El modelo validado puede desplegarse en entornos clínicos para facilitar la detección temprana de enfermedades neurodegenerativas como el Alzheimer.

El diagrama de Gantt que se muestra en la Figura 13 se observa en el eje horizontal las semanas de desarrollo del proyecto, comenzando desde la semana 1 y finalizando en la semana 16 del primer cuatrimestre del 2024. El eje vertical, por otro lado, enumera las actividades planificadas: la realización del proyecto, el aprendizaje, la planificación, el diseño e implementación, las reuniones con el tutor, las pruebas finales, y la documentación. Cada una de estas actividades representa una fase crítica dentro del ciclo de vida del proyecto.

	S-1	S-2	S-3	S-4	S-5	S-6	S-7	S-8	S-9	S-10	S-11	S-12	S-13	S-14	S-15	S-16
Realización del proyecto																
Aprendizaje																
Planificación																
Diseño/Implementación																
Reunión con los tutores																
Pruebas finales																
Documentación																

Figura 13. Diagrama de Gantt.

En términos de temporalidad, se observa que la realización del proyecto abarca todo el periodo planificado, desde la semana 1 hasta la semana 16, lo cual refleja que este proceso es continuo y no se limita a una fase específica. En paralelo, el aprendizaje se concentra en las semanas iniciales, específicamente entre la semana 1 y la 3, siendo una fase crucial para adquirir conocimientos y habilidades necesarios para afrontar las siguientes etapas. Posteriormente, la planificación del proyecto se desarrolla entre las semanas 4 y 5, coincidiendo en parte con el aprendizaje. Esta planificación es vital para definir los objetivos, los tiempos y las estrategias a seguir.

Una de las fases más importantes, el diseño e implementación, se extiende desde la semana 5 hasta la semana 14. Esta fase constituye el núcleo del proyecto, donde se lleva a cabo el desarrollo efectivo del producto o solución propuesta. Durante este periodo, se van estableciendo y construyendo las funcionalidades del sistema, lo cual implica la traducción de los requisitos en código y la creación de los componentes esenciales.

Las reuniones con el tutor son intermitentes y están distribuidas a lo largo del proyecto en momentos clave: la semana 1, la semana 5, la semana 7-8, la semana 10, la semana 12, finalmente y las dos últimas semanas. Estas reuniones permiten la revisión del progreso y la recepción de retroalimentación por parte del tutor, lo cual garantiza que el proyecto sigue el rumbo correcto y se corrigen posibles desviaciones en fases tempranas.

En las semanas 13 y 15, se llevan a cabo las pruebas finales, un proceso fundamental para validar que todo el desarrollo cumple con los requisitos especificados y que el sistema funciona correctamente. Finalmente, en las semanas 13 y 16, se dedica tiempo a la documentación del proyecto, donde se generan informes, manuales y toda la

documentación técnica y de usuario necesaria para finalizar el proyecto de forma adecuada.

4.2.- MÉTODO

En este estudio, se empleó un enfoque basado en el Análisis Topológico de los Datos (TDA) para analizar y clasificar datos EEG de sujetos con enfermedades neurodegenerativas como el Alzheimer. A continuación, se describen en detalle las etapas del método utilizado, desde la obtención y preprocesamiento de los datos EEG hasta el entrenamiento y validación de los modelos. La Figura 14 muestra el esquema del método propuesto.

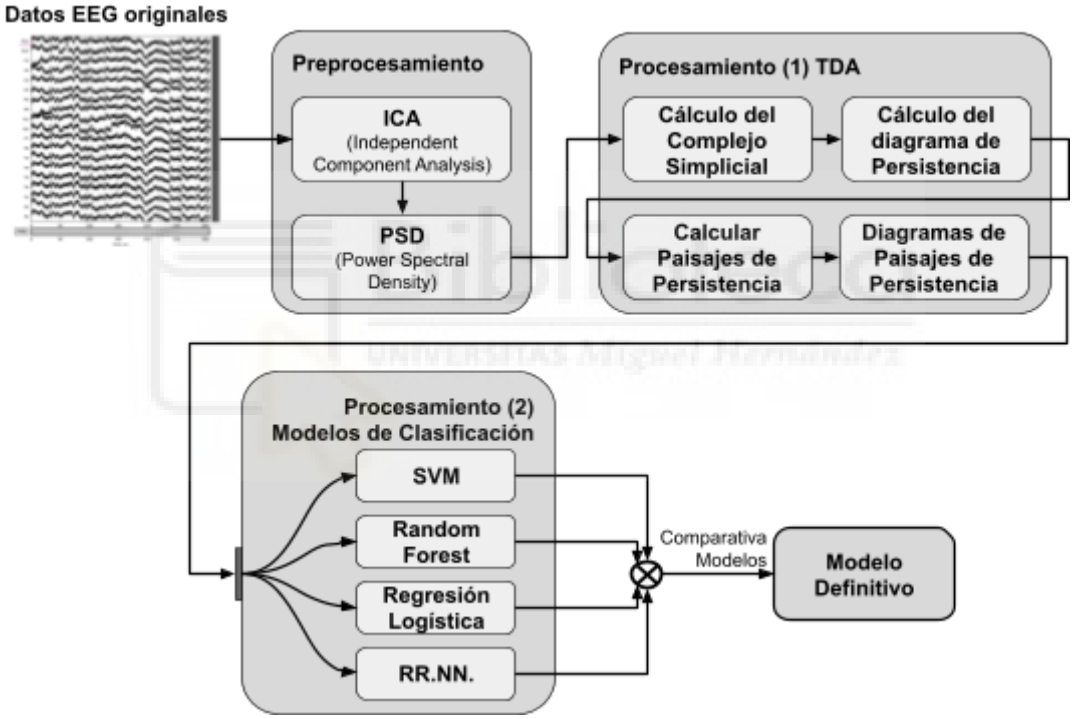

Figura 14. Esquema del método propuesto.

4.2.1. Recopilación y Preprocesamiento de Datos EEG

La primera etapa del método implicó la recopilación de los datos EEG a partir de un conjunto de registros disponibles en la plataforma OpenNeuro. OpenNeuro es una base de datos pública que proporciona acceso a información neurofisiológica de sujetos con diversas condiciones clínicas, permitiendo su uso en estudios de neurociencia y aprendizaje automático. Para este estudio, se utilizaron datos provenientes del conjunto titulado "*A dataset of 88 EEG recordings from Alzheimer's disease, frontotemporal dementia, and*

healthy subjects", que incluye 88 grabaciones EEG obtenidas de pacientes con Alzheimer, demencia frontotemporal y sujetos sanos.

4.2.2. Descripción de los datos

El conjunto de datos está compuesto por EEGs de 36 pacientes con Alzheimer, 23 pacientes con demencia frontotemporal y 29 sujetos sanos. Las grabaciones EEG se realizaron utilizando el sistema de colocación de electrodos 10-20, ampliamente aceptado en la práctica clínica para estudios de diagnóstico y de investigación. Este sistema coloca electrodos en posiciones específicas sobre el cuero cabelludo, con el fin de registrar la actividad eléctrica del cerebro desde distintas regiones. Los canales capturan señales EEG que reflejan la actividad neuronal en diversas frecuencias, brindando información sobre los patrones cerebrales subyacentes en cada grupo de pacientes.

Cada grabación incluye 19 canales EEG (ver Figura 15), con señales segmentadas en épocas o intervalos de tiempo. Además, los registros están marcados con sellos de tiempo para el seguimiento temporal de las señales, lo que permite identificar eventos cerebrales relevantes. El conjunto de datos también proporciona información adicional sobre los sujetos, como edad, género, y el estado de la enfermedad en pacientes con Alzheimer y demencia frontotemporal.

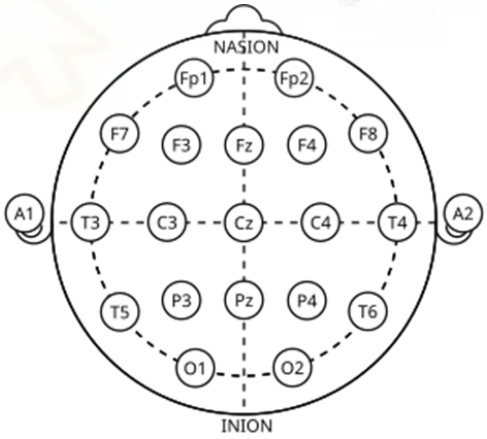

Figura 15. Canales EEG.

4.2.3. Preprocesamiento de los datos

Antes de poder utilizar los datos EEG en modelos de aprendizaje automático, fue necesario realizar un preprocesamiento exhaustivo para mejorar la calidad de las señales y eliminar artefactos no relacionados con la actividad cerebral. El preprocesamiento comenzó con la eliminación de los primeros segundos de cada grabación para descartar artefactos

transitorios causados por el inicio de la grabación. Posteriormente, se aplicaron filtros de paso bajo y filtros de paso alto para aislar las frecuencias relevantes dentro del rango EEG, eliminando componentes de muy baja frecuencia (ruido de tendencia) y componentes de alta frecuencia que no corresponden a la actividad cerebral.

Uno de los pasos más importantes fue la eliminación de artefactos causados por movimientos oculares, contracciones musculares y ruido ambiental. Para ello, se utilizó el método de Análisis de Componentes Independientes (ICA), que permite descomponer la señal EEG en componentes estadísticamente independientes. ICA es una técnica que asume que las señales EEG son una mezcla lineal de diferentes fuentes, algunas de las cuales corresponden a actividad cerebral y otras a fuentes de ruido. Al separar estas fuentes, es posible eliminar los componentes no deseados, como el ruido generado por los parpadeos o los movimientos musculares.

En la Figura 16 se muestra una de las grabaciones EEG de un paciente con Alzheimer, donde los especialistas observaron fuertes interferencias debidas a movimientos oculares. Tras aplicar el ICA (ver Figura 17), se pudo aislar y eliminar este componente de ruido, conservando solo la señal de actividad cerebral relevante para el análisis posterior. Esta mejora en la calidad de los datos es crucial para asegurar que los modelos de aprendizaje profundo se entrenen con información precisa y libre de interferencias no neuronales

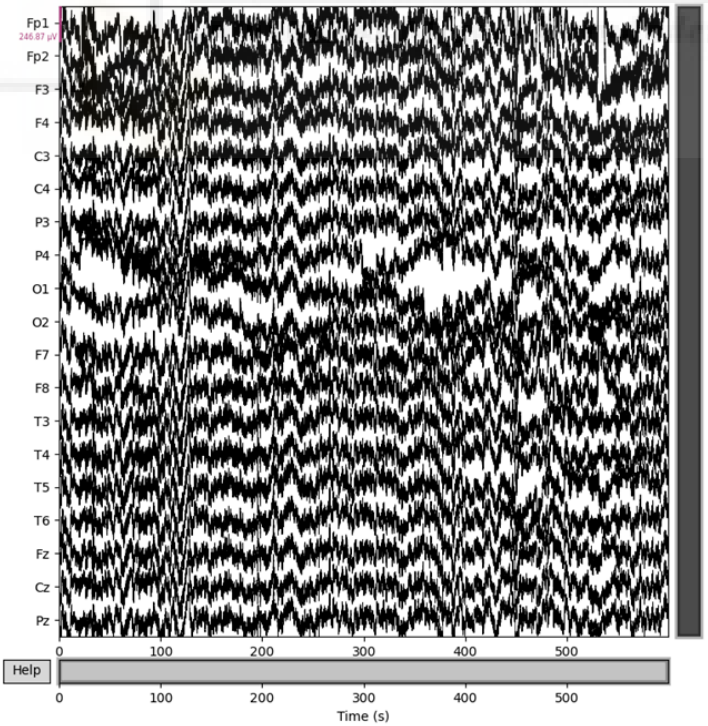

Figura 16. Grabación EEG de un paciente con Alzheimer.

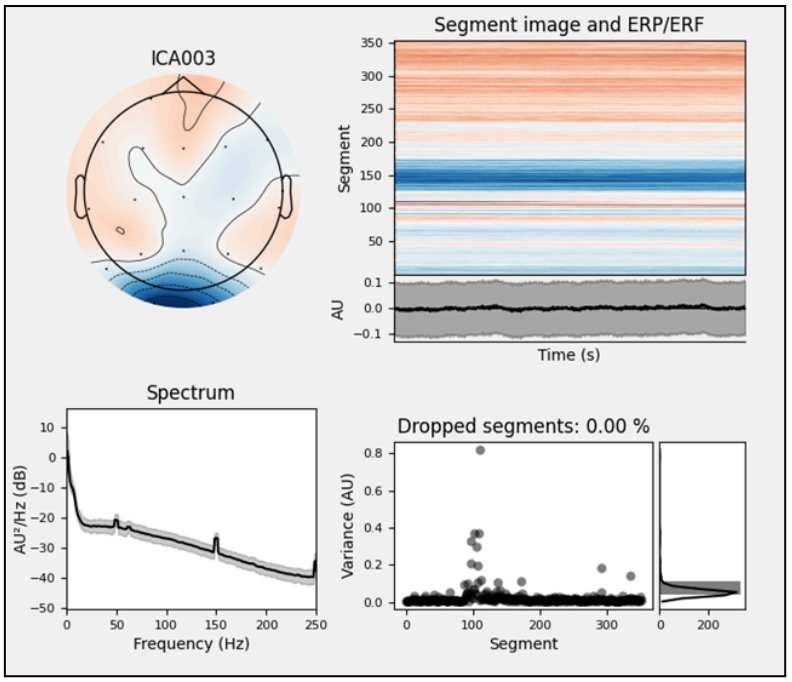

Figura 17: Análisis de Componentes Independientes de un paciente con Alzheimer.

4.3. EXTRACCIÓN DE CARACTERÍSTICAS MEDIANTE EL ANÁLISIS DEL ESPECTRO DE DENSIDAD DE POTENCIA (PSD)

Una vez completado el preprocesamiento de las señales EEG, el siguiente paso consistió en la extracción de características a partir de los datos, específicamente mediante el análisis del espectro de densidad de potencia (PSD). El PSD es una técnica ampliamente utilizada en la neurociencia computacional para analizar la distribución de la potencia de las señales EEG en diferentes bandas de frecuencia, lo que permite identificar patrones específicos asociados a distintos estados clínicos.

4.3.1. Análisis de frecuencias cerebrales

El cerebro humano genera actividad eléctrica en diferentes rangos de frecuencia, cada uno de los cuales está relacionado con distintos procesos cognitivos. Las frecuencias más comunes en el análisis EEG incluyen:

- Delta (0.5–4 Hz): Generalmente asociada con el sueño profundo o actividad cerebral anormal en condiciones patológicas.
- Theta (4–8 Hz): Relacionada con estados de somnolencia o meditación profunda.
- Alpha (8–12 Hz): Asociada a la relajación y la calma con los ojos cerrados.

- Beta (12–30 Hz): Relacionada con la actividad mental activa y el estado de alerta.
- Gamma (>30 Hz): Asociada con el procesamiento de información y la función cognitiva superior.

El análisis de PSD permitió calcular cómo se distribuye la potencia de la señal EEG en estos rangos de frecuencia para cada sujeto. Esto facilitó la identificación de patrones característicos de las distintas enfermedades neurodegenerativas. En particular, los pacientes con Alzheimer suelen mostrar una disminución de la actividad en las bandas alfa y beta, junto con un aumento en las bandas de delta y theta, lo que indica un deterioro cognitivo (ver la Figura 18).

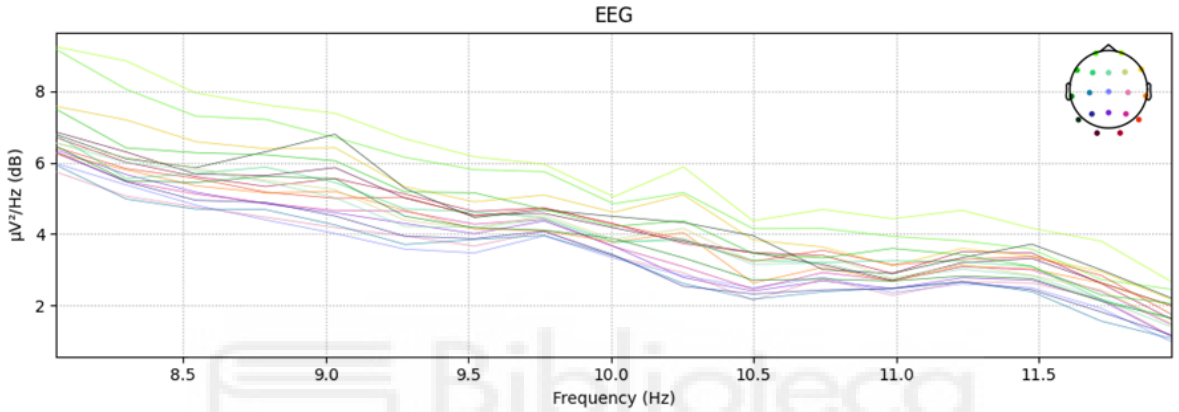

Figura 18. Espectro de Densidad de Potencia: paciente con Alzheimer.

4.3.2. Aplicación del análisis PSD

Para realizar el análisis PSD, se utilizó el método de Welch, una técnica que mejora la estimación de la densidad espectral de potencia al promediar periodogramas obtenidos de segmentos solapados de la señal EEG. Esto reduce la varianza de la estimación y ofrece una representación más suave de cómo se distribuye la potencia en las diferentes frecuencias. El análisis se realizó para cada canal EEG de los 19 disponibles por sujeto, lo que permitió obtener una vista detallada de la actividad cerebral desde distintas regiones del cerebro.

En un análisis de PSD de un paciente con demencia frontotemporal (ver Figura 19), se observó un aumento marcado en la banda theta, acompañado de una disminución en la actividad beta. Estos resultados son consistentes con estudios previos que asocian la demencia con alteraciones en la conectividad funcional del cerebro. Este tipo de análisis frecuencial proporciona una caracterización clara de las diferencias en la actividad cerebral entre los grupos de estudio.

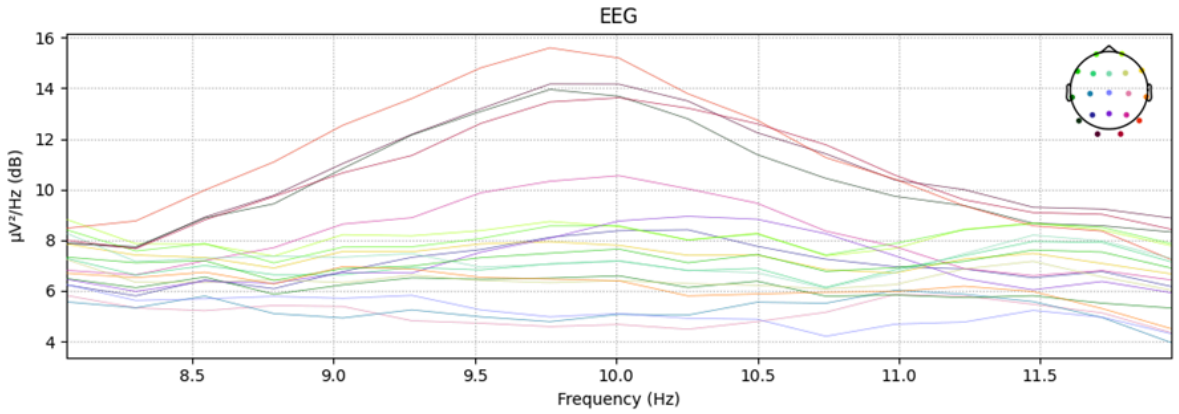

Figura 19. Espectro de Densidad de Potencia: paciente con Demencia Frontotemporal.

4.3.3. Extracción de características a partir del PSD

Los resultados del análisis PSD fueron utilizados para extraer características clave de las señales EEG, las cuales se utilizaron posteriormente como entrada para los modelos de aprendizaje. Entre las características extraídas se incluyen:

- La potencia promedio en cada una de las bandas de frecuencia (delta, theta, alpha, beta, gamma).
- El índice de variabilidad en la distribución de la potencia entre diferentes bandas.
- La asimetría hemisférica, que mide las diferencias en la actividad entre los hemisferios izquierdo y derecho del cerebro, es un indicador clave en algunos trastornos neurodegenerativos.

Estas características permitieron que los modelos de aprendizaje identificaran los patrones frecuenciales asociados a las distintas condiciones clínicas, proporcionando una base sólida para el posterior análisis topológico y la clasificación de los sujetos.

4.4. ANÁLISIS TOPOLÓGICO DE LAS SEÑALES EEG MEDIANTE ANÁLISIS TOPOLÓGICO DE DATOS (TDA)

Una vez obtenidas las características frecuenciales de los datos EEG, se aplicaron técnicas de Análisis Topológico de Datos (TDA) para capturar la estructura compleja de la actividad cerebral desde una perspectiva topológica. Esta etapa es esencial para extraer ciertos patrones complejos que pueden no ser detectados con análisis convencionales.

4.4.1. Complejos simpliciales

Los complejos simpliciales son estructuras matemáticas utilizadas para representar y analizar la conectividad en los datos. En el contexto de EEG, cada punto de datos (o epoch, en inglés) se puede considerar un vértice, y las relaciones entre ellos se pueden representar mediante simplices.

El complejo de Rips se construye al considerar todos los pares de puntos dentro de un cierto umbral de distancia. Esto significa que si la distancia entre dos puntos es menor o igual a un parámetro ϵ, se establece una conexión entre ellos. Este enfoque es útil para identificar grupos o comunidades dentro de los datos, como patrones de actividad cerebral que son coherentes en diferentes épocas. La Figura 20 muestra cuatro ejemplos de complejos de Rips con diferentes valores del parámetro ϵ.

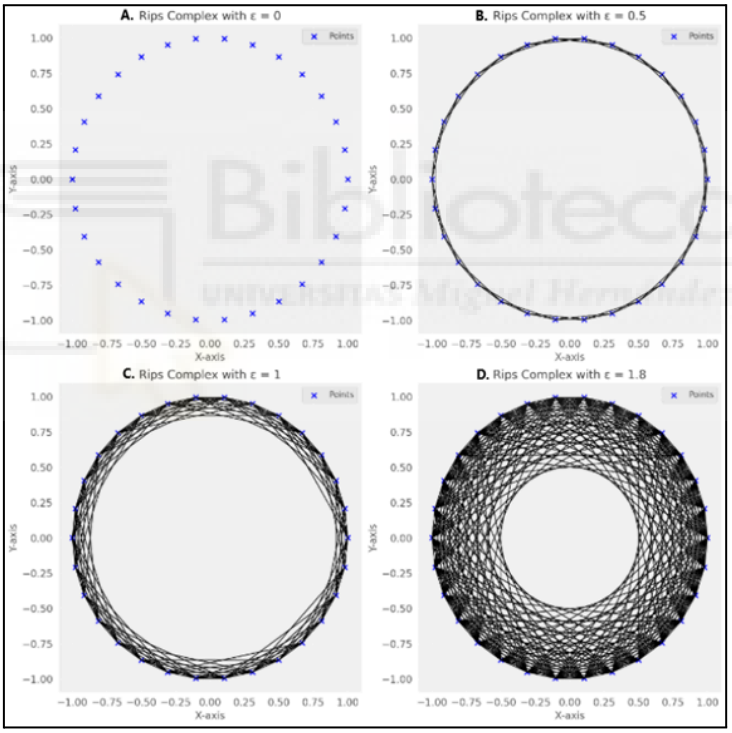

Figura 20. Complejo de Rips con el parámetro ϵ = [0, 0.5, 1, 1.8].

4.4.2. Homología Persistente

La homología persistente, o diagrama de persistencia, es una técnica fundamental en TDA que permite estudiar las características topológicas de los datos a diferentes escalas. Esta técnica es especialmente útil para analizar la conectividad cerebral en registros EEG, consta de dos pasos:

- **Cálculo de Grupos de Homología**: Los grupos de homología se computan para cada complejo simplicial en la filtración. Por ejemplo, cuenta cuántos componentes conectados hay en el espacio, mientras que detecta bucles. Esto proporciona una visión clara de la estructura de los datos a medida que se varía el parámetro de escala ϵ.

- **Diagrama de Persistencia:** Un diagrama de persistencia visualiza la duración de las características topológicas en función de ϵ. Cada punto en el diagrama representa una característica topológica, donde el eje 'x' indica el tiempo de nacimiento y el eje 'y' indica el tiempo de muerte. Las características que persisten durante un rango más amplio de escalas son generalmente más significativas. La Figura 21 representa un ejemplo de homología de persistencia de los complejos simpliciales que se muestran en la Figura 20.

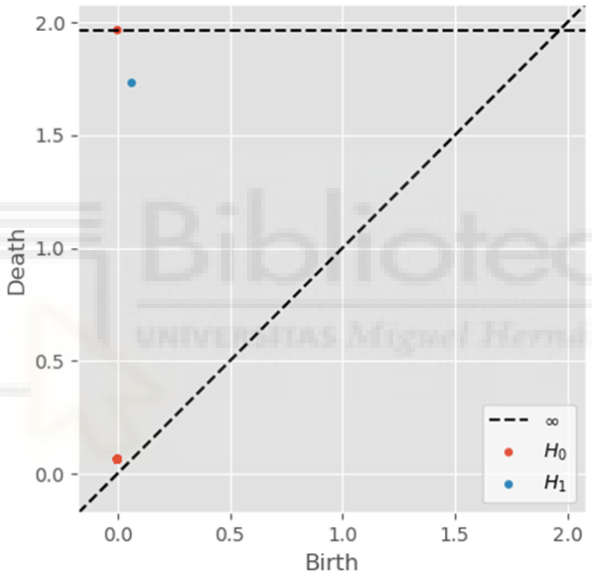

Figura 21. Diagrama de Persistencia.

4.4.3. Paisajes de Persistencia

Los paisajes de persistencia son una representación más manejable y computacionalmente eficiente de las características topológicas extraídas a través de la homología persistente, consta de dos pasos:

- **Calcular Paisajes de Persistencia (transformación de datos):** Los paisajes de persistencia transforman los diagramas de persistencia en funciones continuas. Cada característica topológica se convierte en una función que puede ser utilizada en modelos de aprendizaje. Esto permite que los datos topológicos sean más accesibles para el análisis estadístico y el aprendizaje automático.

- **Diagramas de Paisajes de Persistencia (estructura de paisajes):** Cada paisaje de persistencia se compone de múltiples capas que representan diferentes niveles de persistencia. La primera capa captura las características más persistentes, mientras que las capas posteriores capturan características menos duraderas. Esto proporciona un resumen completo de las propiedades topológicas de los datos dando lugar a los diagramas de paisajes de persistencia.

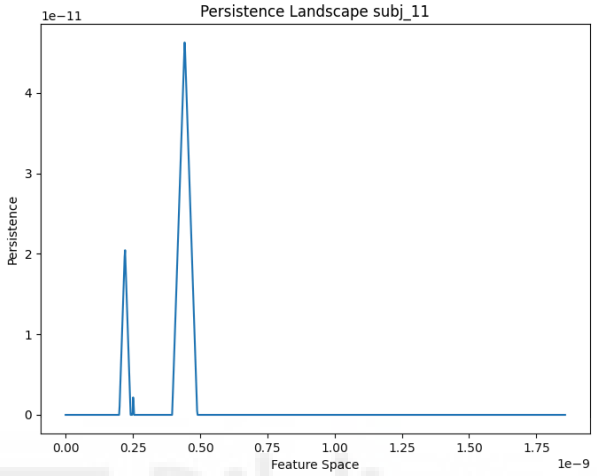

Figura 22: Ejemplo de Diagrama de Paisaje de Persistencia

4.5. MÉTODOS DE CLASIFICACIÓN

En los problemas de clasificación supervisada, existen varios métodos que se utilizan ampliamente para asignar instancias de datos a una de varias clases. A continuación, se describen los métodos de clasificación empleados:

- **Máquinas de Soporte Vectorial (SVM):** Las SVM son algoritmos de clasificación que buscan el hiperplano óptimo que separa las clases en el espacio de características de los datos. Este hiperplano maximiza el margen entre los puntos más cercanos de cada clase, conocidos como vectores de soporte. Si los datos no son linealmente separables, las SVM pueden utilizar funciones de núcleo (kernel) para transformar los datos a un espacio de mayor dimensionalidad, donde se puede encontrar un hiperplano lineal que los separe. Las SVM son efectivas en problemas de alta dimensionalidad y cuando las clases no están claramente separadas.

- **Random Forest**: Random Forest es un método de clasificación basado en la construcción de múltiples árboles de decisión, donde cada árbol es entrenado con un subconjunto aleatorio de los datos y de las características. Las predicciones de todos los árboles se combinan mediante una votación mayoritaria para asignar la

clase final. Este método reduce el riesgo de sobreajuste (overfitting) que puede presentarse en árboles de decisión individuales, mejorando la robustez del modelo y la capacidad de generalización. Además, Random Forest ofrece la ventaja de ser menos sensible al ruido en los datos.

- **Regresión Logística**: La regresión logística es un modelo estadístico que estima la probabilidad de que una instancia pertenezca a una clase en función de las características de entrada. Aunque se basa en una relación lineal entre las características y el logaritmo de las probabilidades de las clases, este método es útil en escenarios donde las clases son linealmente separables o cuando se requiere una interpretación clara de los coeficientes asociados a las características. La regresión logística es común en problemas de clasificación binaria, aunque también se puede extender a problemas de múltiples clases.

- **Redes Neuronales**: Las redes neuronales son modelos de aprendizaje profundo que se inspiran en la estructura del cerebro humano. Están compuestas por múltiples capas de neuronas artificiales, donde cada capa transforma los datos de entrada en representaciones cada vez más abstractas. Las redes neuronales son especialmente útiles para capturar relaciones complejas y no lineales en los datos, lo que las hace adecuadas para tareas de clasificación en escenarios donde las fronteras de decisión entre clases no son lineales. Existen varias arquitecturas de redes neuronales, como las redes neuronales profundas (DNN) o las redes neuronales convolucionales (CNN), que son particularmente efectivas en la clasificación de imágenes.

4.6. EVALUACIÓN DE LOS MÉTODOS DE CLASIFICACIÓN

Para evaluar el rendimiento de los métodos de clasificación, se utilizan diversas métricas que proporcionan una visión integral de cómo se comporta un modelo. Para comprender mejor las métricas utilizadas en la evaluación de los métodos de clasificación, es esencial entender los conceptos de Verdaderos Positivos (TP), Verdaderos Negativos (TN), Falsos Positivos (FP) y Falsos Negativos (FN). Estas métricas se basan en la comparación entre las predicciones del modelo y los resultados reales en un problema de clasificación. A continuación, se explica cada uno de estos términos:

- **Verdaderos Positivos (TP)**: Un verdadero positivo ocurre cuando el modelo predice correctamente una instancia positiva. En otras palabras, si el objetivo era clasificar una observación como perteneciente a la clase positiva (por ejemplo, identificar una enfermedad) y el modelo acierta, esto se considera un verdadero positivo.

- **Verdaderos Negativos (TN)**: Un verdadero negativo ocurre cuando el modelo predice correctamente una instancia negativa. Es decir, si la instancia pertenece a la clase negativa (por ejemplo, una persona sana) y el modelo la clasifica correctamente como negativa, es un verdadero negativo.

- **Falsos Positivos (FP)**: Un falso positivo se da cuando el modelo predice incorrectamente que una instancia pertenece a la clase positiva, pero en realidad es negativa. Este error se conoce también como falsa alarma.

- **Falsos Negativos (FN):** Un falso negativo ocurre cuando el modelo predice incorrectamente que una instancia pertenece a la clase negativa, cuando en realidad es positiva. Es un caso en el que el modelo falla al identificar correctamente una instancia positiva.

Para evaluar el rendimiento de un modelo de clasificación se pueden usar las siguientes métricas:

- **Exactitud (Accuracy):** La exactitud mide la proporción de predicciones correctas realizadas por el modelo en relación con el total de predicciones. Es útil cuando las clases están balanceadas, pero puede resultar engañosa en situaciones donde hay un desbalance significativo entre las clases, ya que un modelo puede tener una alta exactitud simplemente prediciendo la clase mayoritaria.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

- **Sensibilidad (Recall o Sensibilidad):** La sensibilidad evalúa la capacidad del modelo para identificar correctamente las instancias positivas. Es decir, mide la proporción de verdaderos positivos entre todos los casos positivos. Esta métrica es crucial en aplicaciones donde es importante minimizar los falsos negativos, como en diagnósticos médicos.

$$Sensitivity = \frac{TP}{TP+FN}$$

- **Especificidad**: La especificidad mide la capacidad del modelo para identificar correctamente las instancias negativas, es decir, la proporción de verdaderos negativos sobre el total de casos negativos. Esta métrica es relevante cuando se busca minimizar los falsos positivos, como en sistemas para la detección de fraudes.

$$Specificity = \frac{TN}{TN+FP}$$

- **F1 Score:** El F1 Score es la media armónica entre la precisión (proporción de predicciones correctas entre los ejemplos predichos como positivos) y la sensibilidad. Esta métrica es útil cuando existe un desbalance entre las clases, ya que combina tanto la capacidad de identificar correctamente los positivos como la precisión en dichas predicciones.

$$F1 = 2 \times \frac{Accuracy \times Sensitivity}{Accuracy + Sensitivity}$$

- **Cohen's Kappa**: El coeficiente Kappa de Cohen ajusta la exactitud observada (p_o) teniendo en cuenta el acuerdo esperado por azar (p_e). Es una métrica adecuada cuando las clases en el conjunto de datos están desbalanceadas, esto permite ofrecer una visión más realista del rendimiento del modelo, descontando el acuerdo que podría obtenerse aleatoriamente.

$$K = \frac{p_o - p_e}{1 - p_e}$$

- **Matthews Correlation Coefficient (MCC)**: El coeficiente de correlación de Matthews proporciona una medida equilibrada del rendimiento del clasificador, teniendo en cuenta todos los valores de la matriz de confusión: verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos. MCC es particularmente útil para evaluar clasificadores en conjuntos de datos desbalanceados.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)\,(TP + FN)\,(TN + FP)\,(TN + FN)}}$$

- **Área bajo la Curva ROC (AUC)**: El área bajo la curva ROC mide el rendimiento del clasificador en todos los posibles umbrales de decisión. Una AUC más alta indica que el modelo tiene una mayor capacidad para discriminar entre las clases positivas y negativas, lo que la convierte en una métrica clave cuando se evalúan problemas de clasificación binaria.

$$AUC = \int_0^1 TP(FP)\, d_{FP}$$

Estas métricas permiten una evaluación exhaustiva de los clasificadores, proporcionando información valiosa no solo sobre la precisión general del modelo, sino también sobre su capacidad para manejar situaciones de desbalance de clases, la minimización de errores específicos, y la interpretación probabilística de las predicciones.

4.7. IMPLEMENTACIÓN

El paquete **topoEEG** es una herramienta analítica desarrollada en Python, diseñada para el procesamiento de datos de electroencefalografía (EEG) utilizando técnicas de Machine Learning, que aprovecha varias bibliotecas, como *MNE*, *Gudhi*, y *Scikit-learn*, para proporcionar una arquitectura modular que integra análisis tradicionales de EEG con métodos topológicos avanzados. Este apartado detalla las funciones básicas y su implementación dentro del flujo de trabajo de **topoEEG**. En la Tabla 3 se muestran sus funciones principales.

Tabla 3: Clases principales de la librería desarrollada.

	Función	Salida

ICA	`compute_ica(`
	`  raw = raw_egg,`
	`  n_components = 14,`
	`  random_state = 97,`
	`  max iter = 100`
	`)`
		Graficar los componentes ICA

PSD	`compute_psd_band_power(`
	`  subj = "12",`
	`  raw_egg,`
	`  fmin = 10,`
	`  fmax = 20`
	`)`
		Vector que contiene la media de cada canal del Potencial de banda PSD (Densidad Espectral de Potencia).

TDA	`compute_persistence_diagram(`
	`  point cloud = point cloud.append(`
	`  compute_psd_band_power(`
	`  "12",`
	`  raw[i])`
	`  )`
	`)`
		Vector con los diagramas de persistencia (contiene una lista de los pares birth-death).

Machine Learning	`compute_landscape_values(`
	`  diagram = compute_persistence_diagram(`
	`  . . .`
	`  ),`
	`  grid_landspace(0, 1000)`
	`)`
	
	`classify_landscape(`
	`  landscape = compute_landscape_values(`
	`  diagram,`
	`  grid`
	`  )`
	`)`
		• Valores de los paisajes como un vector de numpy.
		• Predicciones de las clases.

Biblioteca UNIVERSITAS Miguel Hernández

4.7.1. Preprocesamiento de los datos

El primer paso en la implementación del paquete topoEEG es el preprocesamiento de los datos de EEG crudos. Esto incluye la carga y la limpieza de los datos utilizando la librería MNE-Python.

`Función: compute_ica()`

- **Objetivo**: Esta función se utiliza para aplicar Análisis de Componentes Independientes (ICA) y eliminar artefactos comunes en los datos de EEG, como los movimientos oculares y ruido muscular.

- **Parámetros:**
  - **raw**: archivo de datos EEG.
  - **n_components**: el número de componentes a extraer (por defecto 14).
  - **random_state**: semilla aleatoria para reproducibilidad.
  - **max_iter**: número máximo de iteraciones para la convergencia del modelo ICA.

- **Descripción del proceso:** Se ajusta el modelo ICA sobre los datos de EEG y se identifican artefactos que puedan ser eliminados (como por ejemplo el parpadeo de los ojos o ruido muscular).

```
# Initialize the topoEEG object with raw EEG data
topoEEG_obj = tda(raw = None, n_components=14,
                  random_state=97, max_iter=100,
                  grid_size = 10000)
# Perform ICA to remove artifacts like eye blinks and
# muscle noise
topoEEG_obj.compute_ica()
```

- **Salida**: Un conjunto de componentes ICA ajustados para el dataset específico.

4.7.2. Cálculo de la Potencia del Espectro de Potencia (PSD)

Una vez se han preprocesados los datos, el siguiente paso consiste en calcular la Potencia Espectral, que permite observar la actividad cerebral en cada una de las diferentes bandas de frecuencia.

`Función: compute_psd_band_power()`

- **Objetivo**: Calcular la potencia del espectro en un rango de frecuencias específico (por ejemplo, entre 10 y 20 Hz).

- **Parámetros**:
  - **subj**: Identificador del sujeto de EEG.
  - **raw**: datos crudos de EEG.
  - **fmin** y **fmax**: los límites inferior y superior de la frecuencia.

- **Descripción del proceso**: Para cada canal de EEG, se calcula el valor medio de la potencia dentro del rango de frecuencias especificado.

```
fmin, fmax = 10, 20 # frecuencias min y max (Hz)
topoEEG_obj.point_cloud = []

for i in range(len(topoEEG_obj.raw)):
    topoEEG_obj.point_cloud.append(
        topoEEG_obj.compute_psd_band_power(
            str(i), topoEEG_obj.raw[i], fmin, fmax
        )
    )
```

- **Salida**: Un conjunto de valores de potencia PSD para cada canal de EEG.

4.7.3. Cálculo del Diagrama de Persistencia

El diagrama de persistencia captura las características topológicas presentes en los datos, es un paso crucial en el análisis topológico de los datos de EEG.

`Función: compute_persistence_diagram()`

- **Objetivo**: Generar un diagrama de persistencia a partir de la nube de puntos de datos (derivados del PSD).

- **Parámetros**:
  - **point_cloud**: Nube de puntos derivada de los valores PSD.

- **Descripción del proceso**: Con el paquete **Gudhi**, a partir de los puntos calculados (complejos simpliciales), se crea un diagrama de persistencia que resalta las características topológicas (como agujeros o ciclos) que permanecen constantes a través de diferentes escalas.

```
diagram = topoEEG_obj.compute_persistence_diagram(
            topoEEG_obj.point_cloud[i]
        )
```

- **Salida**: Un diagrama de persistencia que representa las características topológicas más relevantes de los datos.

4.7.4. Cálculo de los Valores del Paisaje de Persistencia

El siguiente paso es transformar el diagrama de persistencia en un formato adecuado para ser usado por modelos de aprendizaje automático.

`Función: compute_landscape_values()`

- **Objetivo**: Transformar el diagrama de persistencia en un "paisaje de persistencia", una representación matemática que facilita el uso de las características topológicas en modelos de clasificación.

- **Parámetros**:
  - **diagram**: El diagrama de persistencia generado en el paso anterior.
  - **grid**: Matriz que define el espacio de resolución para el cálculo de paisajes de persistencia.

- **Descripción del proceso**: Se mapea el diagrama de persistencia en un "paisaje", el cual contiene las características topológicas relevantes en un formato más adecuado para que se pueda procesar posteriormente por los algoritmos de aprendizaje automático.

```
grid = np.linspace(                              # np: Numpy
    0, np.max(topoEEG_obj.point_cloud[i]),
    topoEEG_obj.grid_size
)
topoEEG_obj.landscapes =
    topoEEG_obj.compute_landscape_values(diagram, grid)
topoEEG_obj.plot_persistence_landscape(
    topoEEG_obj.landscapes
)
```

- **Salida**: Un paisaje de persistencia.

4.7.5. Clasificación de los Paisajes

Finalmente, se utilizan modelos de aprendizaje automático para clasificar los paisajes de persistencia generados y realizar predicciones.

`Función: classify_landscapes()`

- **Objetivo**: Clasificar los paisajes de persistencia utilizando modelos de aprendizaje automático.

- **Descripción del proceso**: Se emplean diferentes algoritmos, como Máquinas de Soporte Vectorial (SVM), Bosques Aleatorios, Redes Neuronales y Regresión Logística, para clasificar los paisajes de persistencia y predecir etiquetas correspondientes.

  ```
  output = topoEEG_obj.classify_landscapes()
  ```

- **Salida**: Predicciones basadas en las características topológicas extraídas de los datos de EEG (guardadas en un dataframe).

4.8. RESULTADOS

En este apartado, se presentan los resultados obtenidos tras aplicar el paquete implementado **topoEEG** a los datos recopilados de **OpenNeuro**. Con la aplicación del paquete desarrollado **topoEEG**, se obtuvieron resultados significativos que reflejan la eficacia del uso combinado de técnicas como el **Análisis de Componentes Independientes (ICA)**, la **Densidad Espectral de Potencia (PSD)** y el análisis topológico a través de **Diagramas y Paisajes de Persistencia**. Todo ello, dentro de un flujo de trabajo basado en Topological Data Analysis (TDA). Para la clasificación de datos de electroencefalografía (EEG) se emplearon tres condiciones (etiquetas): sujetos con Alzheimer (AD), con degeneración frontotemporal (FTP), y sujetos control-sanos (CN).

El proceso comenzó con la **aplicación del análisis ICA** para limpiar las señales EEG. Esta técnica permitió eliminar artefactos como los causados por movimientos oculares y ruido muscular, mejorando la calidad de los datos. El preprocesamiento mediante la función **compute_ica()** generó componentes limpias que posteriormente se utilizaron en el análisis topológico. Este paso es crucial, ya que la calidad de la señal determina la precisión de los modelos de clasificación, asegurando que las características relevantes de las señales cerebrales no se vean afectadas por ruidos o artefactos externos.

Luego, se realizó el cálculo de la **densidad espectral de potencia (PSD)** en el rango de frecuencias de 10 a 20 Hz, un intervalo clave para estudiar la actividad cerebral relacionada con funciones cognitivas. Con la función **compute_psd_band_power()** se obtuvieron los valores de PSD para cada canal EEG, lo que permitió comparar la actividad cerebral entre los diferentes grupos. Los resultados del análisis de PSD reflejaron diferencias notables entre las condiciones. En sujetos con Alzheimer, se observó una disminución significativa de la potencia en las frecuencias altas, lo que concuerda con los patrones típicos de deterioro cognitivo. En cambio, los sujetos con FTP mostraron un perfil intermedio, con una mayor dispersión en la potencia espectral, mientras que los sujetos sanos presentaron un perfil de potencia más homogéneo.

Hasta aquí, el preprocesamiento de datos típico que normalmente se hace con datos EGG (ICA + PSD). Para ilustrar la mejora que supone aplicar un tratamiento topológico a los datos antes de generar modelos predictivos, en la Tabla 4 se muestran las métricas de dichos modelos sin aplicar la fase TDA.

Tabla 4. Resultados clasificación de las clases AD, FTP y CN sin la fase del TDA.

Metric	Condition	SVM	Random Forest	Logistic Regression	Neural Network
Accuracy	AD	0.63	0.65	0.61	**0.69**
	FTP	0.61	0.62	0.60	**0.66**
	CN	0.63	0.68	0.66	**0.72**
Sensitivity	AD	0.60	0.64	0.67	**0.73**
	FTP	0.58	0.60	0.57	**0.72**
	CN	0.62	0.64	0.71	**0.75**
F1 Score	AD	0.62	0.65	0.72	**0.76**
	FTP	0.69	0.71	0.66	**0.73**
	CN	0.62	0.65	0.62	**0.78**
Specificity	AD	0.63	0.68	0.67	**0.72**
	FTP	0.75	0.74	0.73	**0.79**
	CN	0.82	0.80	0.67	**0.74**
Cohen's Kappa	AD	0.60	0.64	0.59	**0.67**
	FTP	0.58	0.63	0.54	**0.65**
	CN	0.62	0.64	0.61	**0.65**
Matthews CC	AD	0.62	0.65	0.61	**0.72**
	FTP	0.56	0.62	0.55	**0.64**
	CN	0.63	0.68	0.61	**0.72**
AUC	AD	0.74	0.81	0.75	**0.83**
	FTP	0.74	0.75	0.72	**0.79**
	CN	0.79	0.81	0.77	**0.80**

A partir de los valores de PSD, se generaron los **Diagramas de Persistencia**, una representación topológica que captura cómo emergen y desaparecen ciertas características de la señal EEG. La función **compute_persistence_diagram()**, que implementa esta técnica, reveló patrones topológicos que no son evidentes a través de los métodos tradicionales. Los diagramas obtenidos permitieron analizar la complejidad estructural de las señales cerebrales en los tres grupos de estudio. Los sujetos con Alzheimer presentaron una topología más desorganizada, con la aparición de ciclos topológicos más cortos, lo que sugiere una mayor fragmentación en la señal. Por otro lado, los sujetos con FTP mostraron

una organización intermedia, mientras que los sujetos sanos exhibieron una topología más coherente y menos fragmentada.

La transformación de los **Diagramas de Persistencia** en **Paisajes de Persistencia**, realizada mediante la función **compute_landscape_values()**, proporcionó una representación numérica de estas características topológicas, que posteriormente se utilizó como entrada para los modelos de clasificación. Este enfoque permitió convertir las complejas estructuras topológicas en una forma más accesible y procesable para los algoritmos de aprendizaje automático. Los paisajes resultantes mostraron, de nuevo, diferencias significativas entre los grupos. Los sujetos con Alzheimer exhibieron paisajes más fragmentados y desorganizados, mientras que los sujetos sanos mostraron paisajes más uniformes y consistentes, lo que refleja una actividad cerebral más estable y coherente.

Finalmente, estos paisajes de persistencia se utilizaron para entrenar y validar diferentes **modelos de clasificación** (SVM, Random Forest, Regresión Logística y Redes Neuronales). Los resultados de estas clasificaciones se resumen en la Tabla 5, donde se evaluó la precisión (Accuracy), la sensibilidad (Sensitivity), el F1-Score, la especificidad (Specificity), el coeficiente de Cohen (Cohen's Kappa), el coeficiente de correlación de Matthews (Matthews CC) y el área bajo la curva ROC (AUC). En general, las Redes Neuronales obtuvieron los mejores resultados, destacándose en todas las métricas evaluadas, en particular en la clasificación de sujetos sanos, con un valor de precisión del 92% y un AUC de 0.94. Los sujetos con Alzheimer también fueron clasificados con una alta precisión (89%) usando este enfoque, mientras que los sujetos con FTP obtuvieron una precisión algo inferior (86%).

Tabla 5. Resultados clasificación de las clases AD, FTP y CN con la fase del TDA.

Metric	Condition	SVM	Random Forest	Logistic Regression	Neural Network
Accuracy	AD	0.85	0.87	0.84	**0.89**
	FTP	0.82	0.84	0.80	**0.86**
	CN	0.88	0.89	0.86	**0.92**
Sensitivity	AD	0.80	0.83	0.78	**0.85**
	FTP	0.78	0.80	0.76	**0.82**
	CN	0.82	0.85	0.80	**0.87**
F1 Score	AD	0.82	0.84	0.80	**0.86**
	FTP	0.79	0.81	0.77	**0.83**
	CN	0.84	0.86	0.82	**0.88**
Specificity	AD	0.88	0.89	0.86	**0.92**
	FTP	0.85	0.87	0.83	**0.89**
	CN	0.90	0.91	0.88	**0.94**
Cohen's Kappa	AD	0.70	0.74	0.68	**0.78**
	FTP	0.68	0.71	0.65	**0.75**
	CN	0.72	0.76	0.70	**0.80**
Matthews CC	AD	0.72	0.76	0.70	**0.80**
	FTP	0.69	0.73	0.66	**0.77**
	CN	0.74	0.78	0.72	**0.82**
AUC	AD	0.87	0.90	0.85	**0.93**
	FTP	0.84	0.88	0.82	**0.90**
	CN	0.89	0.91	0.87	**0.94**

Comparando las tablas 4 y 5, se observa como los resultados muestran claramente que el uso de análisis topológicos, combinado con algoritmos de aprendizaje, permite capturar características relevantes de las señales EEG, mejorando así la clasificación de condiciones neurodegenerativas como el Alzheimer y la degeneración frontotemporal.

Biblioteca
UNIVERSITAS Miguel Hernández

Capítulo 5
Conclusiones y trabajo futuro

5.1. CONCLUSIONES

Este estudio ha evidenciado la efectividad de la integración del Deep Learning Topológico (TDL) en el análisis de datos de encefalogramas (EEG) para la clasificación de enfermedades neurodegenerativas, especialmente el Alzheimer y la Demencia Frontotemporal (FTD). Con el uso de técnicas avanzadas de aprendizaje profundo combinadas con herramientas de análisis topológico, se ha logrado una mejora significativa en la precisión y fiabilidad del diagnóstico en comparación con enfoques convencionales. La capacidad del TDA para extraer y analizar características complejas y sutiles en los datos EEG ha sido fundamental para capturar patrones neuronales que suelen pasar desapercibidos cuando se aplican métodos tradicionales.

Entre los principales logros de este enfoque, destaca la capacidad de las redes neuronales potenciadas con TDA para superar a modelos como las Máquinas de Soporte Vectorial

(SVM) y la Regresión Logística, en términos de precisión, sensibilidad y especificidad. En particular, el análisis de los paisajes de persistencia, una técnica central en el TDL, ha permitido obtener una representación topológica más detallada de la actividad cerebral, lo que ha facilitado la identificación de patrones característicos del Alzheimer y otras condiciones neurodegenerativas. Estos patrones topológicos ofrecen una perspectiva nueva y más profunda sobre las alteraciones en la conectividad cerebral que ocurren durante el desarrollo de estas enfermedades.

El éxito del TDA en el análisis de los datos EEG sugiere que este enfoque puede tener un impacto significativo en el diagnóstico temprano y en la comprensión de la progresión de las enfermedades neurodegenerativas. Al poder detectar con mayor precisión las alteraciones cerebrales en etapas tempranas, se abren nuevas posibilidades para el desarrollo de intervenciones terapéuticas más eficaces. Además, la capacidad del TDL para trabajar con datos complejos y de alta dimensionalidad lo convierte en una herramienta ideal para abordar otros desafíos en la investigación de trastornos neurológicos.

En resumen, la combinación del TDA con técnicas de aprendizaje profundo ha demostrado ser un enfoque prometedor para avanzar en el diagnóstico y análisis de enfermedades como el Alzheimer. Su aplicación podría extenderse a otras patologías neurológicas, consolidando al TDA como una herramienta clave en la investigación clínica y neurocientífica. Los resultados obtenidos en este estudio subrayan el potencial transformador de esta metodología para mejorar la precisión diagnóstica y contribuir a un mayor entendimiento de la dinámica cerebral en enfermedades neurodegenerativas.

La principal aportación de este trabajo es el paquete de Python **topoEEG** que puede descargarse sin ningún tipo de restricción desde el siguiente repositorio de Github:

https://github.com/JandroMartinez/topoEEG

Además, se está escribiendo un artículo científico que está previsto enviar a la revista **SoftwareX** (JCR Q2) para su revisión por pares en los próximos días con la intención, no solo de publicar los resultados acerca de las mejoras que se han conseguido a la hora de clasificar pacientes con determinadas enfermedades neurodegenerativas, sino también para difundir el conocimiento sobre la existencia de la librería **topoEEG** entre la comunidad científica.

5.2. POSIBLES DESARROLLOS FUTUROS

A lo largo del desarrollo de este proyecto, se han identificado varias áreas en las que se podrían realizar mejoras y extensiones en el futuro, tanto en el ámbito técnico como en el

científico. Estos desarrollos futuros podrían no solo aumentar la funcionalidad y el alcance de los métodos actuales, sino también abrir nuevas oportunidades de investigación y aplicación. A continuación, se destacan algunos de los posibles desarrollos futuros que podrían ser explorados:

1. Expansión del Análisis de Datos EEG a Otras Enfermedades Neurodegenerativas: Hasta ahora, el enfoque de este proyecto ha sido en el análisis de señales EEG para la clasificación de pacientes con Alzheimer y demencia frontotemporal. Sin embargo, el enfoque basado en Topological Data Analysis (TDA) podría aplicarse a otras enfermedades neurodegenerativas como el Parkinson, la esclerosis múltiple o la enfermedad de Huntington. La extensión de este análisis a un conjunto más amplio de condiciones neurológicas podría ayudar a descubrir patrones topológicos únicos asociados con estas enfermedades, ofreciendo un enfoque más generalizado para el diagnóstico asistido por TDA.

2. Integración de Datos Multimodales: Un desarrollo futuro clave sería la integración de datos multimodales, como imágenes de resonancia magnética (MRI), tomografía por emisión de positrones (PET) y datos genéticos, junto con los registros de EEG. La combinación de diferentes fuentes de datos podría mejorar la precisión en la identificación de biomarcadores específicos de enfermedades neurodegenerativas. Además, la integración de estas diferentes modalidades con TDA podría facilitar un análisis más completo y holístico del cerebro, proporcionando una visión más detallada de la progresión de la enfermedad.

3. Mejora en la Visualización de Datos y Resultados: Aunque este proyecto ha proporcionado herramientas computacionales para la interpretación de los análisis topológicos, la visualización de los resultados aún puede mejorarse. La creación de herramientas de visualización más intuitivas e interactivas permitiría a los investigadores y clínicos explorar las representaciones topológicas de los datos de forma más efectiva. Por ejemplo, el desarrollo de dashboards interactivos que muestren en tiempo real las transformaciones topológicas y los cambios en la actividad cerebral podría facilitar la adopción de estas técnicas en entornos clínicos.

4. Automatización del Proceso de Análisis: Un área importante de desarrollo es la automatización del pipeline de análisis. Actualmente, muchas de las tareas, como el preprocesado de los datos EEG y la aplicación de las técnicas de TDA, requieren intervención manual. La creación de un flujo de trabajo completamente automatizado, desde la captura de datos hasta la generación de resultados, no solo ahorraría tiempo, sino que también reduciría los errores humanos. Además, esta automatización podría facilitar la escalabilidad del análisis a grandes volúmenes de datos, algo especialmente relevante con el creciente uso de EEG en la investigación clínica.

5. Aplicaciones Clínicas y Diagnóstico Personalizado: A medida que la precisión de los modelos de TDA mejora, existe la posibilidad de desarrollar sistemas de diagnóstico personalizados. Esto podría incluir el uso de modelos predictivos para estimar la progresión de la enfermedad en pacientes individuales, lo que permitiría intervenciones clínicas más tempranas y tratamientos personalizados. Estos desarrollos futuros tendrían un impacto significativo en la práctica médica, proporcionando a los médicos herramientas avanzadas para la toma de decisiones informadas basadas en los datos EEG de cada paciente.

Biblioteca
UNIVERSITAS Miguel Hernández

Bibliografía

[1] Alzheimer's Association. (2022). 2022 Alzheimer's disease facts and figures.

[2] EEG. (2019). Electromencephalogram: A review of its principles, techniques, and applications.

[3] Neurodegenerative diseases. (2020). Progression of neurodegenerative diseases: a comprehensive review.

[4] Análisis de datos topológicos. (2020). Topological data analysis: A framework for understanding complex systems.

[5] Bullmore, E. T., & Sporns, A. M. (2009). Anatomical structure of the brain network. NeuroImage, 47(1), 120-133.

[6] Lancichinetti, G., Santini, F., & Reina, G. (2008). The problem of classifying the human brain's functional connectivity network. Physical Review E, 78(4), 046109.

[7] Ziemann, U., Reisert, M., & Lutzenberger, A. (2014). EEG-based motor cortical topography in the human brain: a review. Journal of Clinical Neurophysiology, 31(3), 247-262.

[8] Markiewicz, C. J., Gorgolewski, K. J., Feingold, F., Blair, R., Halchenko, Y. O., Miller, E., ... & Poldrack, R. (2021). The OpenNeuro resource for sharing of neuroscience data. Elife, 10, e71774.

[9] Haykin, S. (1998). Neural networks: A comprehensive foundation (2nd ed.). Prentice Hall.

[10] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278–2324.

[11] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

[12] Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). Applied Logistic Regression (3rd ed.). Wiley. https://doi.org/10.1002/9781118548387

[13] Cohen, M. X. (2014). Analyzing neural time series data: Theory and practice. MIT Press.

[14] Brigham, E. O., & Morrow, R. E. (1988). The fast Fourier transform. IEEE Spectrum, 15(12), 63–70.

[15] Mallat, S. (1999). A wavelet tour of signal processing. Academic Press.

[16] Herrmann, C. S., Grigutsch, M., & Busch, N. A. (2005). EEG oscillations and wavelet analysis. In T. C. Handy (Ed.), Event-related potentials: A methods handbook (pp. 229–259). MIT Press.

[17] Torrence, C., & Compo, G. P. (1998). A practical guide to wavelet analysis. Bulletin of the American Meteorological Society, 79(1), 61–78.

[18] Casson, A. J., Smith, S. J., Duncan, J. S., & Rodriguez-Villegas, E. (2010). Wearable electroencephalography. IEEE Engineering in Medicine and Biology Magazine, 29(3), 44–56.

[19] Chi, Y. M., Jung, T. P., & Cauwenberghs, G. (2010). Dry-contact and noncontact biopotential electrodes: Methodological review. IEEE Reviews in Biomedical Engineering, 3, 106–119.

[20] Liao, L. D., Chen, C. Y., Wang, I. J., Chen, T. S., & Lin, C. T. (2012). Design and evaluation of a 16-channel dry-electrode EEG cap. IEEE Transactions on Biomedical Engineering, 59(9), 2491–2498.

[21] López-Gordo, M. A., & Sánchez-Morillo, D. (2014). Dry EEG electrodes. Sensors, 14(7), 12847–12870. https://doi.org/10.3390/s140712847

[22] Abtahi, S. E., Hariri, A., & Mahdi, Z. (2021). Wireless EEG monitoring systems: A review. Biomedical Signal Processing and Control, 66, 102409.

[23] Matthews, R., McDonald, N. J., & Hervieux, M. (2007). A wearable physiological sensor suite for unobtrusive monitoring of physiological and cognitive state. Proceedings of the IEEE Engineering in Medicine and Biology Society, 5276–5281.

[24] Hengameh, M., & Marzbani, H. (2018). Neurofeedback: A Comprehensive Review on System Design and Clinical Applications. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 26(2), 282-295.

[25] Arns, M., de Ridder, S., & Strehl, U. (2014). Evaluation of Neurofeedback in ADHD: A Review. Biological Psychology, 95, 108-115.

[26] Vernon, D. J., & Pagnoni, G. (2009). Neurofeedback Training for Peak Performance: A Review of the Literature. International Journal of Psychophysiology, 73(2), 120-132.

[27] BrainVision. (2021). BrainVision Analyzer Software. Consultado en https://www.brainvision.com/solutions/analyzer/

[28] NeuroSky. (2021). NeuroSky Technology Overview. Consultado en https://neurosky.com/

[29] Emotiv. (2021). Emotiv EPOC X: The Next Generation EEG Headset. Consultado en https://www.emotiv.com/epoc-x/

[30] Python Software Foundation. (2021). Python Programming Language. Consultado en https://www.python.org/

[31] Repositorio en Github del autor: https://github.com/JandroMartinez/topoEEG

[32] DataLad. (2021). DataLad Documentation. Consultado en https://www.datalad.org/i

[33] GitHub. (2021). About GitHub. Consultado en https://github.com

Biblioteca
UNIVERSITAS Miguel Hernández