



UNIVERSIDAD DE VALLADOLID

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE TELECOMUNICACIÓN

TRABAJO FIN DE MÁSTER

MÁSTER EN INGENIERÍA DE TELECOMUNICACIÓN

DETECTOR AUTOMÁTICO DE ARTEFACTOS EN SEÑALES NEURONALES BASADO EN TÉCNICAS DE AI E INCORPORACIÓN EN LA HERRAMIENTA MEDUSA[©]

AUTORA:

DÑA. AMALIA GIL CORREA

TUTORES:

DR. D. JESÚS POZA CRESPO

DR. D. CARLOS GÓMEZ PEÑA

TÍTULO: **Detector automático de artefactos en señales neuronales basado en técnicas de AI e incorporación en la herramienta MEDUSA[®]**

AUTOR: **Dña. Amalia Gil Correa**

TUTOR/ES: **Dr. D. Jesús Poza Crespo**
Dr. D. Carlos Gómez Peña

DEPARTAMENTO: **Teoría de la Señal y Comunicaciones e Ingeniería Telemática**

TRIBUNAL

PRESIDENTE: **Dr. D. Roberto Hornero Sánchez**

SECRETARIO: **Dr. Dña. Miriam Antón Rodríguez**

VOCAL: **Dr. Dña. María García Gadañón**

P. SUPLENTE: **Dr. D. Ignacio de Miguel Jiménez**

S. SUPLENTE: **Dr. D. Juan Pablo Casaseca de la Higuera**

V. SUPLENTE: **Dr. D. Luis Miguel San José Revuelta**

FECHA:

CALIFICACIÓN:

A mi familia

Agradecimientos

Me gustaría agradecer a Jesús Poza Crespo y Carlos Gómez Peña por volver a darme la oportunidad de realizar este proyecto con el Grupo de Ingeniería Biomédica de la Universidad de Valladolid, y por sus consejos y aportaciones, que han hecho posible que este trabajo haya podido completarse con éxito. Agradecer también a los chicos del Grupo de Ingeniería Biomédica por haberme resuelto las dudas que me han ido surgiendo.

Gracias a mi familia, pequeños y mayores, por haberme acompañado y guiado siempre durante mi etapa como estudiante.

Resumen

La información proporcionada por las distintas señales biológicas que genera el cuerpo humano puede ser muy valiosa para el diagnóstico precoz de enfermedades, así como para optimizar su tratamiento. En este sentido, las señales electromagnéticas generadas por el cerebro permiten identificar patrones neurológicos que reflejan ciertas enfermedades neuronales o que caracterizan el estado cognitivo del paciente. Técnicas como la electroencefalografía (EEG) y la magnetoencefalografía (MEG) han revolucionado la investigación en neurociencia al proporcionar información valiosa sobre la actividad cerebral en tiempo real de manera no invasiva. No obstante, estas técnicas no registran exclusivamente información neuronal, sino que también capturan información no deseada debido a múltiples fuentes de ruido del entorno, tanto ajenas a cada persona (ruido ambiental, por ejemplo, de la red eléctrica) como internas o fisiológicas (por ejemplo, de la actividad cardíaca u ocular). Estas señales espurias se denominan artefactos y alteran la señal neuronal genuina, comprometiendo la validez de los análisis posteriores y dificultando la extracción de información relevante. Para reducir el impacto de los artefactos en los registros EEG y MEG se han desarrollado diferentes métodos de preprocesamiento de señales, desde técnicas clásicas basadas en filtrado o transformadas hasta enfoques más modernos que emplean métodos estadísticos, como el análisis de componentes independientes (ICA). A pesar de la utilidad de estas técnicas, muchas de ellas requieren intervención manual o dependen de suposiciones sobre la naturaleza del ruido, por lo que su aplicación a conjuntos de datos masivos y la precisión de los resultados continúa viéndose comprometida. Por ello, surge la necesidad de desarrollar herramientas automáticas que permitan detectar y mitigar dichos artefactos, reduciendo la dependencia de procesos manuales y subjetivos.

En este Trabajo de Fin de Máster se ha trabajado con señales MEG reales procedentes de un conjunto de datos previamente etiquetado por expertos, así como con señales sintéticas generadas artificialmente a partir de componentes ICA neuronales combinadas con artefactos. Estas últimas han permitido evaluar el funcionamiento del sistema conociendo el contenido neuronal y el nivel de ruido añadido. La solución propuesta ha consistido en un detector automático de artefactos en señales MEG basado en *Deep Learning* (DL). Dicho detector se basa en el algoritmo ICA *Extended Infomax*, que permite separar las señales MEG en componentes; estas componentes son la entrada a una Red Neuronal Convolucional EEG-Inception. La red permite identificar componentes dudosas que, si bien no son clasificadas directamente como artefactos, presentan un bajo nivel de confianza para ser consideradas puramente neuronales. Además, como parte de este trabajo, se ha implementado un sistema de eliminación automática de segmentos

contaminados de la señal a partir de la probabilidad de clasificación ofrecida por la red. Para poder disponer de una representación visual de los resultados obtenidos, se ha diseñado e implementado una interfaz gráfica que permite representar los segmentos ruidosos de las señales neuronales y eliminarlos. Con esto, el usuario de la herramienta puede decidir conservar la señal bajo estudio en el conjunto de datos, descartarla o mitigar el ruido detectado a partir del *score* de clasificación calculado por la red y recuperar una versión limpia de la señal. La interfaz también permite la generación de señales sintéticas contaminadas con artefactos de origen cardíaco, ocular, de red eléctrica o de otro tipo. Todo el código desarrollado, implementado en Python, se ha integrado en el núcleo de MEDUSA[®], un entorno en Python orientado al análisis avanzado de señales neuronales. Se ha dividido en 5 partes que se han integrado en diferentes ficheros de Github: (i) carga de señales, (ii) estandarización de señales, (iii) CNN diseñada, (iv) modelo de inteligencia artificial para identificar y eliminar automáticamente artefactos, y (v) interfaz gráfica.

En cuanto a los resultados obtenidos, la red neuronal diseñada alcanzó una sensibilidad superior al 95 % en la detección de componentes artefactuadas sobre señales reales, así como una especificidad cercana al 99 % y una precisión de casi un 97 %. Asimismo, se aplicó un proceso de limpieza de artefactos sobre aquellas componentes clasificadas inicialmente como no artefacto pero con una probabilidad inferior al 90 % (señales denominadas dudosas). Tras esta limpieza, se consiguió reducir el conjunto dudoso hasta un 35,41 % de las señales iniciales. Por otro lado, en el caso de las señales sintéticas se analizó cómo influía el ajuste de potencia aplicado a los artefactos añadidos a señales neuronales limpias en la calidad de las detecciones. Los resultados muestran que el porcentaje de señal detectada como artefacto aumenta progresivamente a medida que se incrementa el ajuste de potencia aplicado, llegando a superar el 90 % cuando la potencia del artefacto alcanza el 200 % de la potencia de la señal neuronal.

La herramienta desarrollada permite mejorar la calidad de las señales MEG, facilitando el trabajo de los técnicos encargados de su preprocesado. Además, su incorporación en MEDUSA[®] permite a los investigadores acceder a herramientas de preprocesamiento dentro de una plataforma unificada, mejorando significativamente la eficiencia en la detección y eliminación de artefactos en registros neuronales.

Palabras clave

Artefactos, detección automática, EEG-Inception, Inteligencia Artificial, magnetoencefalografía, MEDUSA[®], modelo *Deep Learning*, *score*, segmentación.

Abstract

The information provided by the different biological signals generated by the human body can be extremely valuable for the early diagnosis of diseases, as well as for optimising their treatment. In this context, the electromagnetic signals produced by the brain make it possible to identify neurological patterns that reflect certain disorders or characterise a patient's cognitive state. Techniques such as electroencephalography (EEG) and magnetoencephalography (MEG) have transformed neuroscience research by offering real-time, non-invasive insights into brain activity. However, these techniques do not exclusively record neural information but also capture unwanted signals from multiple noise sources, both external (such as environmental noise from power lines) and internal or physiological (like heart or eye activity). These spurious signals, known as artefacts, distort genuine neural activity, compromising the validity of subsequent analyses and making it difficult to extract meaningful information. To reduce the impact of artefacts in EEG and MEG recordings, a variety of pre-processing methods have been developed, ranging from traditional approaches based on filtering or signal transforms to more modern statistical techniques. While useful, many of these methods require manual intervention or rely on assumptions about the nature of the noise, which limits their scalability to large datasets and affects the reliability of their results. This has created a growing need for automated tools capable of detecting and mitigating artefacts, reducing the dependence on manual and subjective processes.

In this Master's thesis we worked with real MEG signals from a dataset previously labelled by experts, and with synthetic signals artificially generated by combining neuronal ICA components with artefacts. The latter made it possible to evaluate the system performance with full knowledge of the neural content and the level of added noise. The proposed solution is an automatic artefact detector for MEG signals based on Deep Learning (DL). The detector relies on the ICA Extended Infomax algorithm, which separates MEG signals into components; these components are then fed into an EEG-Inception Convolutional Neural Network. The network identifies uncertain components that, while not directly classified as artefacts, show low confidence for being considered purely neuronal. In addition, as part of this work, we implemented an automatic system for removing contaminated signal segments based on the classification probabilities provided by the network. To provide a visual representation of the results, we designed and implemented a graphical interface that displays noisy segments of neural signals and removes them. This allows users to decide whether to keep the signal under study in the dataset, discard it, or mitigate the detected noise based on the classification score

computed by the network, recovering a cleaner version of the signal. The interface also supports generating synthetic signals contaminated with artefacts of cardiac, ocular, electrical, or other origins. All the developed code, implemented in Python, has been integrated into the core of MEDUSA[©], a Python-based framework for advanced neural signal analysis. The system is divided into five modules, organized into separate GitHub files: (i) signal loading, (ii) signal standardization, (iii) the designed CNN, (iv) the AI model for automatic artefact detection and removal, and (v) the graphical interface.

Regarding the results, the designed neural network achieved a sensitivity above 95 % in detecting artefact components in real signals, as well as a specificity close to 99 % and an accuracy of nearly 97 %. In addition, a reevaluation process was applied to components initially classified as non-artefacts but with a probability below 90 % (the so-called uncertain signals). After this cleaning step, the uncertain set was reduced to 35.41% of the initial signals. For the synthetic signals, we analyzed how the power adjustment applied to artefacts added to clean neural signals influenced detection quality. The results show that the percentage of signal detected as artefact increases progressively as the applied power adjustment grows, surpassing 90 % when the artefact power reaches 200 % of the neural signal power.

The developed tool enhances the quality of MEG signals, making the preprocessing work of technicians easier. Furthermore, its integration into MEDUSA[©] provides researchers with access to preprocessing tools within a unified platform, significantly improving efficiency in the detection and removal of artefacts from neural recordings.

Keywords

Artificial Intelligence, artefacts, automatic detection, Deep Learning model, EEG-Inception, magnetoencephalography, MEDUSA[©], score, segmentation.

Índice General

Capítulo 1. Introducción	18
1.1. Preprocesamiento de señales biológicas.....	19
1.2. Digitalización y análisis de señales mediante Inteligencia Artificial	20
1.3. Motivación	22
1.4. Hipótesis.....	23
1.5. Objetivos.....	23
1.6. Estructura del documento.....	24
Capítulo 2. Artefactos en señales electromagnéticas cerebrales y técnicas de preprocesado	27
2.1. Introducción	28
2.2. Artefactos y su clasificación	29
2.3. Técnicas tradicionales de reducción de ruido	33
2.4. Reducción de artefactos en señales neuronales	36
2.4.1. Descomposición de señales y clasificación manual	38
2.4.2. Tendencia actual: técnicas de IA.....	40
2.5. Señales sintéticas	46
Capítulo 3. La herramienta MEDUSA[®]	49
3.1. Introducción	50
3.2. Estructura de MEDUSA [®]	51
Capítulo 4. Materiales	54
4.1. Introducción	55
4.2. Base de datos	55
4.3. Conjunto de señales	55
4.3.1. Señales reales	56
4.3.2. Señales sintéticas.....	58

Capítulo 5. Desarrollo de la herramienta de detección y eliminación automática de artefactos en señales neuronales e integración en la herramienta MEDUSA[®]	63
5.1. Introducción	64
5.2. Creación de un sistema automático de detección y eliminación de artefactos basado en técnicas de IA.....	64
5.2.1. Etapa 1: Detección de artefactos	64
5.2.2. Etapa 2: Eliminación de artefactos	67
5.3. Integración del detector automático en la herramienta MEDUSA [®]	72
Capítulo 6. Resultados.....	77
6.1. Introducción	78
6.2. Desempeño del modelo de clasificación.....	78
6.3. Evaluación tras la eliminación de segmentos ruidosos	79
6.4. Evaluación sobre señales sintéticas	83
6.5. Interfaz gráfica en MEDUSA [®]	90
6.5.1. Ventana 1: Señales MEG reales	90
6.5.2. Ventana 2: Señales MEG sintéticas.....	92
Capítulo 7. Discusión y limitaciones	94
7.1. Introducción	95
7.2. Discusión.....	95
7.3. Limitaciones	98
Capítulo 8. Conclusión y líneas futuras	101
8.1. Introducción	102
8.2. Grado de consecución de los objetivos propuestos	102
8.3. Conclusiones y aportaciones originales.....	103
8.4. Líneas futuras.....	104
Bibliografía.....	106

Índice de Figuras

Figura 1. Artefacto cardíaco en señales MEG y EEG obtenidas en un niño con epilepsia. Primer conjunto de señales (en rojo): registros de magnetómetros. Segundo conjunto de señales (en salmón): registros de gradiómetros. Tercer conjunto de señales (en azul): registros de EEG. Cuarta señal (en negro): registro de ECG. Esquina superior derecha: localización del pico de artefacto de ECG en una topografía MEG. Esquina inferior derecha: localización del pico de artefacto de ECG en una topografía EEG (Papadelis, et al., 2018)	31
Figura 2. Artefacto ocular debido a parpadeos en señal MEG obtenida en un adulto de 24 años sano. Primer conjunto de señales (en rojo): registros de gradiómetros. Segundo conjunto de señales (en azul): registro vertical (VEOG) y horizontal (HEOG) de electrooculograma (EOG). A la derecha: localización del pico de artefacto de EOG en una topografía MEG (Papadelis, et al., 2018)	32
Figura 3. Artefacto muscular debido al músculo frontal en señal MEG obtenida en un niño con epilepsia. Primer conjunto de señales (en rojo): registros de gradiómetros. Segundo conjunto de señales (en salmón): registros de magnetómetros. Tercera señal (en azul): registro de EMG. A la derecha: localización del pico de artefacto de EMG en una topografía MEG (Papadelis, et al., 2018)	32
Figura 4. Artefacto no fisiológico debido a un implante coclear en un paciente adulto (Papadelis, et al., 2018)	33
Figura 5. Filtrado de una señal EEG. Gráfica superior: señal EEG original. Gráfica intermedia: señal EEG original tras haber aplicado un filtro de 0.1 a 60 Hz. Gráfica inferior: señal EEG filtrada tras haber suavizado la señal (Arslan, et al., 2015)	35
Figura 6. Conjunto de señales estimadas mediante ICA (izquierda) y mediante PCA (derecha) (Jafarabadi, et al., 2015)	39
Figura 7. Estructura general de una CNN (a) y de una RNN (b) (Zhang, et al., 2021). 43	43
Figura 8. Representación gráfica de las funciones de activación más comunes.....	46
Figura 9. Señales MEG simuladas obtenidas con el módulo SourceSimulator de MNE-Python. Se trata de señales provenientes de diferentes sensores, con una duración de 10 segundos (MNE, 2024).	48

Figura 10. Potenciales evocados generados a partir de señales EEG (primera gráfica) y MEG, con gradiómetros (segunda gráfica) y magnetómetros (tercera gráfica), haciendo uso del módulo SourceSimulator de MNE-Python (MNE, 2024).	48
Figura 11. Vista general de MEDUSA [®] . Se muestran tres tipos de registros, de EEG, ECG y EMG, cuyas señales adquiridas se introducen en la herramienta MEDUSA [®] mediante el protocolo LSL. Las señales pueden verse en las gráficas de tiempo real y pueden ser empleadas en las distintas aplicaciones dentro de la plataforma. En el ejemplo mostrado puede verse la aplicación RCP Speller. También pueden aplicarse distintos procesamientos a las señales dentro del núcleo de MEDUSA [®] . En este caso se muestra la detección de potenciales ERP en señales procedentes de registros EEG (Santamaría, et al., 2023).	53
Figura 12. Cuatro fragmentos de señales etiquetadas como cerebrales ("Label: 0").	57
Figura 13. Cuatro fragmentos de señales etiquetadas como cardíacas ("Label: 1").	57
Figura 14. Cuatro fragmentos de señales etiquetadas como red eléctrica ("Label: 2").	58
Figura 15. Cuatro fragmentos de señales etiquetadas como oculares ("Label: 3").	58
Figura 16. Selección y replicación del artefacto para la generación de señales sintéticas.	59
Figura 17. Máscara binaria de 5 minutos de duración.	60
Figura 18. Señal artefacto obtenida tras aplicar la máscara a la señal replicada.	60
Figura 19. Señal artefacto con diferentes ajustes de potencia respecto de la señal neuronal seleccionada.	61
Figura 20. Componente ICA número 24 etiquetada como neuronal empleada en la creación de una señal sintética con artefacto cardíaco.	61
Figura 21. Señales sintéticas generadas al añadir un artefacto con distintas proporciones de potencia respecto a la señal neuronal original.	62
Figura 22. Señales sintéticas generadas tras haber empleado diferentes artefactos y señales neuronales.	62
Figura 23. Red EEG-Inception adaptada al detector automático de artefactos de este TFM.	65
Figura 24. Señal neuronal con artefactos a la que se han aplicado las 4 segmentaciones posibles. Se muestran los artefactos marcados con colores tras la clasificación y los scores correspondientes: probabilidad de ser artefacto superior al 90 % en rojo,	

<i>probabilidad de ser artefacto entre un 75 y un 90 % en naranja, y probabilidad de ser artefacto entre un 50 y un 75 % en amarillo.</i>	70
Figura 25. <i>Suavización del score de una señal neuronal con artefactos calculado por la red (en azul) mediante la aplicación de un filtro gaussiano y un filtro de media móvil. El score suavizado se muestra en naranja.</i>	72
Figura 26. <i>Flujo de trabajo seguido en la ejecución del TFM.</i>	73
Figura 27. <i>Dos componentes etiquetadas por el técnico especialista como no artefacto pero que la red ha eliminado por considerarlas dudosas. (a) Componente etiquetada como no artefacto (0), y la red ha predicho artefacto (1); score del 53 %. (b) Componente etiquetada como no artefacto (0), y la red ha predicho no artefacto (0); score del 56 %.</i>	79
Figura 28. <i>Señal dudosa etiquetada por un técnico como "neuronal", clasificada por la red CNN como "neuronal", con un score en la clasificación del 87 %.</i>	81
Figura 29. <i>Señal dudosa etiquetada por un técnico como "neuronal", clasificada por la red CNN como "neuronal", con un score en la clasificación del 80 %.</i>	82
Figura 30. <i>Representación en el dominio del tiempo de la señal original (en azul) tras aplicarle diferentes niveles de segmentación y recuperarla (en rojo).</i>	82
Figura 31. <i>Representación del score obtenido tras clasificar la señal y aplicarle diferentes métodos de suavizado y representación en el dominio del tiempo de la señal original (en azul) tras aplicarle esos suavizados y recuperarla (en rojo).</i>	83
Figura 32. <i>Componente neuronal número 5 (fila izquierda) y zoom de la muestra 38000 a la 40000 (fila derecha), sobre la que se ha sumado un artefacto de tipo cardíaco para generar una señal sintética artefactuada, que se ha introducido en el método desarrollado para obtener la señal recuperada equivalente.</i>	85
Figura 33. <i>Componente neuronal número 14 (fila izquierda) y zoom de la muestra 14000 a la 16000 (fila derecha), sobre la que se ha sumado un artefacto de red eléctrica para generar una señal sintética artefactuada, que se ha introducido en el método desarrollado para obtener la señal recuperada equivalente.</i>	85
Figura 34. <i>Componente neuronal número 19 (fila izquierda) y zoom de la muestra 15000 a la 17000 (fila derecha), sobre la que se ha sumado un artefacto de tipo ocular para generar una señal sintética artefactuada, que se ha introducido en el método desarrollado para obtener la señal recuperada equivalente.</i>	86
Figura 35. <i>Componente neuronal número 30 (fila izquierda) y zoom de la muestra 46000 a la 48000 (fila derecha), sobre la que se ha sumado un artefacto de otro tipo para</i>	

<i>generar una señal sintética artefactuada, que se ha introducido en el método desarrollado para obtener la señal recuperada equivalente.</i>	86
Figura 36. <i>Porcentaje de señal detectado como no artefacto durante el primer minuto en función del ajuste aplicado a la señal habiendo realizado segmentaciones de 1 (en verde), 5 (en azul oscuro), 10 (en naranja) y 20 segundos (en azul claro).</i>	87
Figura 37. <i>Porcentaje de señal detectado como artefacto durante el segundo minuto en función del ajuste aplicado a la señal habiendo realizado segmentaciones de 1 (en verde), 5 (en azul oscuro), 10 (en naranja) y 20 segundos (en azul claro).</i>	88
Figura 38. <i>Porcentaje de señal detectado como artefacto durante el tercer minuto en función del ajuste aplicado a la señal habiendo realizado segmentaciones de 1 (en verde), 5 (en azul oscuro), 10 (en naranja) y 20 segundos (en azul claro).</i>	89
Figura 39. <i>Porcentaje de señal detectado como artefacto durante los dos últimos minutos en función del ajuste aplicado a la señal habiendo realizado segmentaciones de 1 (en verde), 5 (en azul oscuro), 10 (en naranja) y 20 segundos (en azul claro).</i>	89
Figura 40. <i>Primera ventana de la interfaz gráfica.</i>	90
Figura 41. <i>Botones empleados para clasificar los segmentos de las señales.</i>	91
Figura 42. <i>Botones empleados para suavizar el score y recuperar la señal neuronal.</i>	91
Figura 43. <i>Segunda ventana de la interfaz gráfica.</i>	92
Figura 44. <i>Botones empleados para seleccionar el tipo de artefacto.</i>	93
Figura 45. <i>Botones empleados para ajustar la potencia del artefacto respecto de la potencia de la señal neuronal.</i>	93

Índice de Tablas

Tabla 1. Tipos de artefactos y sus características principales (Team, 2023).....	30
Tabla 2. Entradas y salidas de la función de carga de señales <code>load_and_save_mat_files()</code>	73
Tabla 3. Entradas y salidas de la función de normalización de señales <code>zscore_normalization()</code>	74
Tabla 4. Entradas y salidas de la función de entrenamiento y <code>test</code> del modelo <code>EEG-Inception</code> modificado <code>train_meg_model()</code>	75
Tabla 5. Entradas y salidas de la función de interfaz gráfica <code>MEG_artifact_recovery_GUI()</code>	76
Tabla 6. Número de señales clasificadas como dudosas de nuevo tras eliminar los artefactos en las señales clasificadas originalmente como dudosas.....	83

Glosario de Acrónimos

Adam	<i>Adaptive moment estimation</i> , Estimación del Momento adaptativo.
AdaGrad	<i>Adaptive Gradient algorithm</i> , Algoritmo del Gradiente adaptativo.
AI	<i>Artificial Intelligence</i> , Inteligencia Artificial.
ANN	<i>Artificial Neural Networks</i> , Red Neuronal Artificial.
BCI	<i>Brain-Computer Interface</i> , Interfaz Cerebro-Máquina.
BSS	<i>Blind Source Separation</i> , Separación Ciega de Fuentes.
CNN	<i>Convolutional Neural Network</i> , Red Neuronal Convolucional.
c-VEP	<i>coded-modulated Visual Evoked Potential</i> , Potencial Evocado Visual modulado por código.
DL	<i>Deep Learning</i> , Aprendizaje Profundo.
ECG	ElectroCardioGram.
EEG	ElectroEncefaloGram.
EMG	ElectroMioGram.
EOG	ElectroOculoGram.
ERP	<i>Event-Related Potential</i> , Potencial Relacionado con Evento.
FFT	<i>Fast Fourier Transform</i> , Transformada Rápida de Fourier.
FIR	<i>Finite Impulse Response</i> , Respuesta al Impulso Finita.
GAN	<i>Generative Adversarial Network</i> , Red Generativa Adversativa.
GIB	Grupo de Ingeniería Biomédica.
HEOG	<i>Horizontal ElectroOculoGram</i> , Electrooculograma Horizontal.
IA	Inteligencia Artificial.
ICA	<i>Independent Component Analysis</i> , Análisis de Componentes Independientes.
LSL	<i>Lab Streaming Layer</i> , Capa de Transmisión de Laboratorio.
MEG	MagnetoEncefaloGram.
ML	<i>Machine Learning</i> , Aprendizaje Automático.
PCA	<i>Principal Component Analysis</i> , Análisis de Componentes Principales.
PSD	<i>Power Spectral Density</i> , Densidad Espectral de Potencia.
RCP	<i>Row-Column Paradigm</i> , Paradigma de Filas y Columnas.
RMSProp	<i>Root Mean Square Propagation</i> .
RNN	<i>Recurrent Neural Network</i> , Red Neuronal Recurrente.
SNR	<i>Signal to Noise Ratio</i> , Relación Señal a Ruido.
SQUID	<i>Superconducting Quantum Interference Device</i> , Dispositivo Superconductor de Interferencia Cuántica.
TFG	Trabajo Fin de Grado.
TFM	Trabajo Fin de Máster.
VEOG	<i>Vertical ElectroOculoGram</i> , Electrooculograma Vertical.
XAI	<i>eXplainable Artificial Intelligence</i> , Inteligencia Artificial Explicable.

Capítulo 1. Introducción

Índice

1.1.	Preprocesamiento de señales biológicas.....	19
1.2.	Digitalización y análisis de señales mediante Inteligencia Artificial	20
1.3.	Motivación	22
1.4.	Hipótesis	23
1.5.	Objetivos	23
1.6.	Estructura del documento.....	24

1.1. Preprocesamiento de señales biológicas

Las señales biológicas son fluctuaciones generadas por los diferentes sistemas fisiológicos del ser humano, y pueden tener origen eléctrico, magnético, químico o mecánico, entre otros (Kaplan, et al., 2005). Así, la señal biológica procedente de la actividad eléctrica del corazón se llama electrocardiograma (ECG), mientras que la generada a partir de la actividad muscular se denomina electromiograma (EMG). Las señales procedentes de la actividad neuronal reciben el nombre de electroencefalograma (EEG), si lo que se desea es registrar la actividad eléctrica del cerebro, o magnetoencefalograma (MEG), si es la actividad magnética la que se quiere medir (Kaplan, et al., 2005) (Reilly & Lee, 2010).

En la mayoría de los casos no basta con aplicar los sistemas de adquisición necesarios para obtener la señal biológica deseada. Preprocesarlas adecuadamente es de vital importancia en la investigación médica (Islam, et al., 2021). Por ejemplo, uno de los objetivos principales a la hora de limpiar estas señales puede ser aumentar la relación señal a ruido (SNR, *Signal to Noise Ratio*) mediante filtrado (Semmlow, 2004), o detectar la presencia de artefactos que contaminan la señal y dificultan su correcta interpretación (Islam, et al., 2021). En este Trabajo Fin de Máster (TFM) toma especial relevancia el segundo enfoque, aplicado específicamente a señales cerebrales. En concreto, el TFM se centra en la detección y posterior eliminación de artefactos en señales MEG.

Los artefactos, también denominados interferencias o ruido de forma genérica, provienen de factores externos al sistema de medición, alterando la señal adquirida e imposibilitando una correcta interpretación de la misma (Blum, et al., 2019). Esto tiene una implicación clínica directa, pues puede provocar que el diagnóstico de enfermedades neurológicas o la monitorización de funciones vitales se vean afectados por la imprecisión en las medidas (Hamal & Rehman, 2013). Los artefactos pueden ser: fisiológicos, procedentes de la actividad corporal del ser humano, como la actividad cardíaca, la respiratoria, la ocular o la muscular; o no fisiológicos, ajenos a la actividad corporal, como interferencias electromagnéticas, ruido instrumental o cambios en la posición de los sensores (Sweeney, 2013) (ICL, 2024). En ambos casos, su detección y eliminación toman especial relevancia para garantizar la calidad de los análisis posteriores, pues su presencia altera la SNR y puede resultar en conclusiones médicas erróneas (Islam, et al., 2012). En la práctica, se emplean diferentes estrategias para mitigar el efecto de estas interferencias, como el uso de cámaras de Faraday para blindar la sala de adquisición de interferencias electromagnéticas, medidas para limitar el movimiento del sujeto, o el procesado posterior efectuado sobre las señales, mediante filtros o técnicas de Inteligencia Artificial (IA) (Singh, 2014) (Hamdan, et al., 2023). En este TFM se va a analizar una metodología basada en técnicas de IA, y cómo se puede emplear para detectar los

artefactos presentes en señales MEG, así como su utilidad para eliminar las partes ruidosas de las señales neuronales.

1.2. Digitalización y análisis de señales mediante Inteligencia Artificial

La digitalización es el proceso mediante el cual se transforman datos de un formato físico a un formato digital (Semmlow, 2004). Esta técnica, aplicada a datos biológicos, ha conseguido transformar campos como la medicina o la neurociencia. La digitalización surge a partir de la necesidad de adquirir, procesar y almacenar grandes cantidades de datos en períodos cortos de tiempo (Garg, et al., 2021). Por ejemplo, las señales neuronales adquiridas a partir de un EEG o de un MEG se benefician de la capacidad de analizar grandes volúmenes de datos con precisión y rapidez.

Los métodos tradicionales de preprocesado y análisis de datos están basados en modelos matemáticos y estadísticos que tienen en cuenta características específicas de las señales, como su estacionariedad, su comportamiento determinista o su linealidad. En este ámbito se aplican técnicas como: la Transformada de Fourier, para obtener las componentes frecuenciales de las señales; la Transformada *Wavelet*, para el análisis de señales no estacionarias; o el filtrado, para eliminar ruido o frecuencias fuera del rango de interés (Shaker, 2006). Sin embargo, las señales neuronales están compuestas por datos complejos que no permiten hacer las suposiciones necesarias para poder aplicar algunos de los métodos clásicos comentados. Por ejemplo, para eliminar los artefactos presentes en señales EEG o MEG no basta con emplear un filtro de una determinada respuesta frecuencial debido a la heterogeneidad de las componentes que forman los registros neuronales (Besserve, et al., 2007).

Diversas técnicas de IA permiten identificar patrones y realizar tareas complejas como la eliminación de ruido imitando la inteligencia humana, algo que sería difícil de lograr con los métodos clásicos (Rouhiainen, 2018). Estas tecnologías han ofrecido nuevas posibilidades para comprender la actividad cerebral, identificar artefactos y mejorar la calidad de los datos para investigaciones o aplicaciones clínicas (Hamet & Tremblay, 2017). Las técnicas de IA se dividen en diversos tipos; así, el Aprendizaje Profundo o *Deep Learning* (DL) pertenece al subconjunto del Aprendizaje Automático o *Machine Learning* (ML), que a su vez está englobado en el ámbito general de la IA. El ML estudia diferentes algoritmos computacionales que permiten automatizar tareas a partir de la experiencia, mientras que el DL trata de imitar el comportamiento del cerebro humano elaborando algoritmos que asemejan el funcionamiento de las neuronas cerebrales. Estos

algoritmos consiguen adaptarse y aprender de forma autónoma a partir de grandes conjuntos de datos (Janiesch, et al., 2021).

A pesar de la versatilidad y eficiencia de las diversas técnicas de IA, es conveniente seguir considerando los métodos clásicos de análisis de datos, pues estos siguen ofreciendo soluciones versátiles para realizar un preprocesado inicial con el fin de mejorar la calidad del conjunto de datos. En este sentido, y aplicado al ámbito de las señales neuronales procedentes de registros EEG o MEG, se puede hacer uso del Análisis de Componentes Independientes (ICA, *Independent Component Analysis*) o del Análisis de Componentes Principales (PCA, *Principal Component Analysis*). El primero se utiliza para separar la mezcla de señales superpuestas que componen la señal neuronal global en sus fuentes independientes (Croce, et al., 2019). El segundo se emplea para reducir la dimensionalidad de los datos conservando la mayor cantidad posible de información relevante (Kaya, 2019). Ambas técnicas se pueden aplicar para eliminar componentes ruidosas o irrelevantes. Sin embargo, esta tarea discriminatoria se realiza habitualmente de forma manual por técnicos expertos en la materia, lo que consume mucho tiempo y conlleva un elevado grado de subjetividad (Croce, et al., 2019). Por eso, es en estos casos donde es razonable asumir que el uso de técnicas IA va a mejorar tanto la calidad de los resultados ofrecidos como el trabajo efectivo realizado por los técnicos (Yang, et al., 2018).

La implementación de las técnicas de IA se puede realizar en diferentes lenguajes de programación, como Matlab o Python. Estos son flexibles, intuitivos, y contienen una serie de librerías adaptadas a la IA. Además, facilitan el análisis de las señales, optimizan el flujo de trabajo y aumentan la reproducibilidad en los estudios (Mumtaz, et al., 2021). Sin embargo, han surgido softwares diseñados específicamente para el análisis de datos en neurociencia, como MEDUSA[®] (Santamaría, et al., 2024); se trata de una plataforma diseñada para el análisis y procesamiento de señales biomédicas, como las registradas mediante EEG o MEG, que consta de una interfaz preconfigurada que reduce la complejidad de uso (Santamaría, et al., 2023). Una de sus utilidades consiste en la integración de algoritmos de IA para su aplicación directa en el procesamiento de señales cerebrales.

Con el fin de eliminar la subjetividad presente en el etiquetado manual de artefactos que se encuentran en las señales neuronales y reducir el tiempo que se tarda en llevarlo a cabo, en este TFM se ha empleado un método basado en técnicas de IA capaz de detectar, clasificar y etiquetar dichos artefactos de manera automática. Además, permite eliminar las partes ruidosas de las señales que están completamente artefactuadas, sino que contienen intervalos neuronales que no conviene eliminar, debido a la información que puedan contener. Esto sería imposible de conseguir con herramientas convencionales o

mediante la actuación manual de un técnico especialista. La metodología diseñada está basada en DL debido a su capacidad para modelar relaciones complejas y no lineales en los datos, permitiendo identificar patrones ocultos que los métodos tradicionales no lograrían detectar (O'Shea & Nash, 2015). Además, se ha implementado la herramienta de detección y clasificación de artefactos en señales neuronales diseñada en el software MEDUSA[©] para favorecer el etiquetado de las componentes procedentes de registros MEG y poder eliminar, de forma completamente objetiva y automática, las partes ruidosas de las señales. Por último, para verificar que los resultados que ofrece la herramienta de IA son fiables y fundamentados, se ha adaptado el software diseñado para poder generar señales neuronales artificiales, que surgen de la combinación de segmentos neuronales a los que se les han añadido segmentos ruidosos procedentes de una fuente no neuronal, es decir, artefactos. Dichas señales, denominadas sintéticas, se han introducido en el modelo de identificación y limpieza de artefactos creado para comprobar que su funcionamiento es correcto partiendo del conocimiento que se tiene sobre la señal generada.

1.3. Motivación

Las técnicas empleadas para la detección y clasificación de artefactos en señales neuronales son técnicas rudimentarias basadas en la experiencia y conocimiento de los técnicos que se encargan de etiquetarlas y que, por lo tanto, no garantizan la correcta eliminación de los artefactos de forma objetiva y sin desechar gran parte de señal neuronal que puede ser relevante para el diagnóstico correspondiente. Además, la aplicación de otros métodos, como ICA o los filtros clásicos, no garantizan una precisión suficiente que cubra las necesidades para el diagnóstico médico. Así, las técnicas de DL, como las redes neuronales convolucionales (CNN, *Convolutional Neural Networks*) suponen un avance significativo en el análisis de las señales procedentes de registros MEG. Estas redes aprenden patrones complejos directamente de los datos sin necesidad de una extracción manual de características. La motivación de este trabajo radica en explorar el potencial de las CNN para resolver el problema de identificación y eliminación automática de los artefactos en señales neuronales, y poder generar una evidencia empírica mediante la generación de señales sintéticas que demuestren el correcto comportamiento de la herramienta desarrollada. Se espera que el enfoque planteado, junto con su integración en la plataforma MEDUSA[©], mejore la calidad de los datos procesados y facilite la extracción de información relevante para estudios clínicos y neurocientíficos.

1.4. Hipótesis

Los artefactos presentes en las señales procedentes de registros MEG o EEG suponen un problema que, sumado a la complejidad de evaluar e interpretar señales neuronales, aumenta el tiempo necesario para llegar a un diagnóstico correcto y presenta un alto grado de subjetividad. Los métodos basados en CNN son capaces de identificar y eliminar artefactos en señales neuronales que no pueden ser detectados ni tratados con precisión por los métodos tradicionales que implican la intervención humana. La hipótesis de la que se parte en este TFM es que la herramienta desarrollada, basada en una CNN, es capaz de detectar componentes artefactuadas con mayor precisión y objetividad que la evaluación realizada por un técnico especialista. Esta capacidad se fundamenta en el hecho de que las CNN pueden aprender automáticamente características relevantes de los datos mediante un entrenamiento adecuado, reduciendo así el tiempo requerido para el etiquetado y eliminando la subjetividad inherente a la evaluación humana. Además, mediante la generación de señales sintéticas formadas por segmentos neuronales y artefactos se puede demostrar el correcto funcionamiento de la herramienta diseñada, y se pueden aprovechar las nuevas señales artificiales para entrenar otras redes al proporcionar un entorno controlado para simular diferentes condiciones y niveles de ruido. Todo esto implementado en el software MEDUSA[®] favorece la creación de aplicaciones en investigación y en entornos clínicos.

1.5. Objetivos

El objetivo principal de este TFM es **diseñar, implementar y evaluar una herramienta de detección, clasificación y eliminación automática de artefactos que contaminan las señales neuronales, e integrarla en el software MEDUSA[®]**. Además, se pretende adaptar la herramienta para que permita la generación de señales sintéticas mediante la combinación de partes neuronales y artefactos disponibles en la base de datos, así como la creación de métodos de eliminación del ruido en segmentos concretos de las señales. Para poder realizar estas tareas, se han empleado técnicas de IA que permiten identificar patrones complejos y solapados de ruido, automatizando el proceso de detección y clasificación de artefactos y seleccionando los segmentos contaminados con ruido para su posterior eliminación. El objetivo principal, a su vez, se ha descompuesto en los siguientes objetivos complementarios:

1. Estudiar los diferentes tipos de artefactos que afectan las señales electromagnéticas cerebrales y analizar las limitaciones de los métodos tradicionales para detectar y eliminar estos artefactos.

2. Familiarizarse con las principales técnicas utilizadas en el procesamiento de señales en neurociencia y destacar el papel de las herramientas modernas, como las basadas en IA, para automatizar y mejorar la precisión en el análisis de señales neuronales.
3. Familiarizarse con el software MEDUSA[©] e investigar para qué estudios de investigación neurocientífica ha sido empleada.
4. Diseñar e implementar una herramienta basada en CNN que detecte, clasifique y elimine artefactos en señales procedentes de registros MEG en estado de reposo, procedentes de sujetos del Hospital de Hokuto (Japón).
5. Generar señales sintéticas y evaluar el funcionamiento de la herramienta de identificación y eliminación automática de artefactos, analizando su rendimiento empleando señales MEG reales y dichas señales sintéticas generadas.
6. Diseñar e implementar métodos de eliminación de los segmentos contaminados por ruido detectados por la herramienta de IA.
7. Crear una interfaz gráfica que integre la herramienta desarrollada.
8. Integrar la herramienta en la plataforma MEDUSA[©].
9. Estudiar los resultados obtenidos a partir del software elaborado y comparar su rendimiento con el de otras investigaciones por parte de otros autores.
10. Obtener las conclusiones pertinentes y destacar las limitaciones y líneas futuras.

1.6. Estructura del documento

El documento está formado por 8 capítulos, cuya estructura es la siguiente:

- **Capítulo 1. Introducción.** En el primer capítulo se comienza introduciendo el preprocesamiento que se suele realizar a señales biológicas, en concreto a las de origen neural. Asimismo, se muestra el concepto de digitalización y su aportación en el campo de la neurociencia, y cómo su combinación con diferentes técnicas de IA puede mejorar el tiempo y calidad de los diagnósticos. Además, se expone la motivación para la realización de este trabajo, las hipótesis iniciales y los objetivos que se pretenden alcanzar.
- **Capítulo 2. Artefactos en señales electromagnéticas cerebrales y técnicas de preprocesado.** En este capítulo se describen los principales tipos de artefactos que suelen aparecer en los registros de señales MEG y se analiza cómo su

presencia afecta al proceso de estudio de las señales cerebrales. También se muestra la relevancia del tratamiento de datos en disciplinas biomédicas, desde la captura hasta la extracción de información significativa, y cómo esto se aplica a la neurociencia. También se introducen algunas de las técnicas tradicionales de reducción de ruido y se comparan con las más modernas, como las basadas en nuevos modelos de IA o la generación de señales sintéticas, y cómo estas pueden favorecer la eliminación de los artefactos en las señales neuronales.

- **Capítulo 3. La herramienta MEDUSA[©].** En este capítulo se detalla el funcionamiento de la herramienta MEDUSA[©], sus principales módulos y las funcionalidades que ofrece para el procesamiento y análisis de señales biomédicas. Se hace especial énfasis en su capacidad para integrar algoritmos de IA, como las CNN.
- **Capítulo 4. Materiales.** En el quinto capítulo se detallan las características de la base de datos empleada y el perfil de los sujetos involucrados en el estudio. A continuación, se describen en detalle las características y particularidades tanto de las señales reales como de las señales sintéticas generadas.
- **Capítulo 5. Desarrollo de la herramienta de detección y eliminación automática de artefactos en señales neuronales e integración en la herramienta MEDUSA[©].** En este capítulo se expone la herramienta desarrollada y la metodología implementada para la detección y clasificación de artefactos en señales neuronales. Se muestra el proceso de generación de señales sintéticas y la adaptación de todo el software para que sea compatible con MEDUSA[©].
- **Capítulo 6. Resultados.** El sexto capítulo se centra en describir los resultados obtenidos tras introducir varias señales neuronales a la herramienta generada, el proceso de eliminación de ruido en estas y su recomposición como señales limpias. Se evalúan distintos parámetros que determinan la calidad de la herramienta, incluyendo métricas de rendimiento y validación, con el objetivo de medir la efectividad de los métodos propuestos y su impacto en el preprocesamiento de señales neuronales.
- **Capítulo 7. Discusión y limitaciones.** En este capítulo se analizan los resultados obtenidos y el funcionamiento de la herramienta de identificación y eliminación automática de artefactos. También se comparan con los estudios realizados por otros investigadores en este campo. A su vez, se exponen las limitaciones que pueden suponer una barrera en el futuro desarrollo de este trabajo, y otros enfoques que contribuyan en el ámbito de la investigación neurocientífica.

- **Capítulo 8. Conclusiones y líneas futuras.** En el último capítulo se detallan las conclusiones obtenidas tras la ejecución de este TFM y las posibles líneas futuras en las que este proyecto pueda servir de apoyo.

Capítulo 2. Artefactos en señales electromagnéticas cerebrales y técnicas de preprocessado

Índice

2.1.	Introducción	28
2.2.	Artefactos y su clasificación.....	29
2.3.	Técnicas tradicionales de reducción de ruido.....	33
2.4.	Reducción de artefactos en señales neuronales.....	36
2.4.1.	Descomposición de señales y clasificación manual	38
2.4.2.	Tendencia actual: técnicas de IA	40
2.5.	Señales sintéticas	46

2.1. Introducción

El estudio del cerebro humano ha ido evolucionando a lo largo de los años. Gracias al psiquiatra alemán Hans Berger se pudo realizar en el año 1929 la primera medición de las señales emitidas por el cerebro de manera no invasiva. Así, Berger destacó principalmente por su contribución al desarrollo del EEG, que supuso un método revolucionario para registrar las señales eléctricas del cerebro (Berger, 1969). Además, identificó las ondas alfa, un patrón rítmico en el rango de 8-13 Hz que se asocia con estados de relajación, y las ondas beta, más rápidas, relacionadas con estados de alerta (Stevens, 1974). Gracias a estos descubrimientos, Berger pudo demostrar que el cerebro genera actividad eléctrica medible que está relacionada con los estados mentales.

Unos años más tarde, el físico y neurocientífico David Cohen realizó el primer registro exitoso de los campos magnéticos generados por el cerebro mediante el uso de un magnetómetro de bobina (Cohen, 1968). Posteriormente, en la década de los años 70 incorporó el uso del SQUID (*Superconducting Quantum Interference Device*), un sensor altamente sensible que se basa en el fenómeno cuántico de la superconducción (Vrba & Robinson, 2001). Su trabajo revolucionó el estudio del cerebro al demostrar que los campos magnéticos generados por la actividad neuronal se podían medir de manera fiable y complementaria a las mediciones proporcionadas por el EEG (Cohen, 1968). Así es como surge el MEG, que permite medir la actividad magnética cerebral mediante el uso de sensores extracraneales (Hämäläinen, et al., 1993).

Ambas técnicas facilitaron la comprensión de los procesos neuronales y asentaron las bases para el estudio moderno de la actividad neuronal. Sin embargo, a pesar de estos avances, tanto el EEG como el MEG se enfrentan a un mismo problema: las señales adquiridas están inevitablemente contaminadas por ruido y artefactos procedentes de diversas fuentes (Mahmud, et al., 2012). La interpretación de las señales cerebrales se ve comprometida por la presencia de estas interferencias, dificultando el diagnóstico de posibles enfermedades cerebrales (Urigüen & García-Zapirain, 2015). En este sentido, el preprocesamiento de señales se ha desarrollado a lo largo de los años según han ido avanzando las diferentes tecnologías en las que dicho preprocesado era necesario. Ya durante la era analógica las señales se manipulaban mediante circuitos eléctricos, filtros pasivos y técnicas como la transformada de Fourier analógica (Kwakernaak, et al., 1991). Con la llegada de la computación en la segunda mitad del siglo XX, el preprocesamiento digital de señales marcó un punto de inflexión. La transformación de señales continuas en datos discretos permitió la aplicación de algoritmos matemáticos más avanzados, como la transformada rápida de Fourier (FFT, *Fast Fourier Transform*) o el filtrado digital (Ambardar, 1995). En el ámbito del análisis de señales cerebrales, el preprocesamiento de señales ha sido fundamental para mejorar la calidad de los datos obtenidos mediante

EEG y MEG (Chawla, 2011). Desde métodos tradicionales como el filtrado e ICA hasta enfoques más modernos basados en técnicas de IA, el desarrollo de diferentes estrategias para el tratamiento de señales neurofisiológicas es necesario para garantizar la fiabilidad de los estudios en neurociencia, además de llegar a diagnósticos precisos de la forma más rápida posible.

En este capítulo se va a definir el concepto de artefacto referido al ámbito de las señales electromagnéticas cerebrales, y los tipos de artefactos que se pueden encontrar. Se va a explicar el impacto que tiene su presencia en el análisis de señales neuronales y lo importante que son su identificación y su eliminación, así como su repercusión directa en la calidad y fiabilidad de los estudios y diagnósticos médicos. También se comentarán algunos de los distintos métodos de preprocesado empleados hasta la fecha y cuáles son las tendencias futuras en esta área de investigación.

2.2. Artefactos y su clasificación

El ruido en cualquier sistema de comunicación es un fenómeno inevitable, ya que forma parte intrínseca de cualquier sistema de adquisición y transmisión de señales. En este sentido, cuando nos referimos al campo de la neurociencia, el ruido que está presente en las señales medidas en el cerebro se denomina artefacto (Burgess, 2020). Se trata de componentes no deseados, dado que no corresponden a la actividad neuronal genuina, que interfieren y corrompen la señal adquirida, dificultando su análisis e interpretación (Team, 2023). Estas fuentes de ruido surgen, en gran medida, debido a la alta sensibilidad de los sistemas de adquisición, como el EEG o el MEG. Los dispositivos empleados están diseñados para captar las débiles señales eléctricas y magnéticas, respectivamente, que son generadas por el cerebro (Colomer, et al., 2016). La amplitud de estas señales es tan baja que los sensores de medición son susceptibles a captar otras fuentes de señal que no son neuronales. Particularmente, los campos magnéticos cerebrales propios de las señales MEG son extremadamente débiles en comparación con las interferencias del entorno (Muthukumaraswamy, 2013). Además, el hecho de que estos sensores estén próximos a otras estructuras del cuerpo humano, como los ojos, los músculos faciales o algunas arterias, facilita la captación de actividad propia de sistemas fisiológicos diferentes al cerebral (MEG, 2019).

Los artefactos pueden ser fisiológicos o no fisiológicos. Los primeros derivan de actividades biológicas propias del ser humano, como parpadeos, ritmos cardíacos, actividad muscular o movimientos oculares. Los segundos provienen de fuentes externas ajenas al cuerpo humano, como interferencias electromagnéticas, movimientos de los sensores de medida o ruido ambiental (Fabietti, et al., 2020). En la Tabla 1 se recoge la

información más relevante de cada tipo de artefacto más común (Team, 2023). Además de los artefactos mostrados, cabe destacar que existen otros que son más difíciles de identificar o que se pueden clasificar como alguno de los ya mencionados, como el debido a la actividad respiratoria, que se puede englobar dentro de los artefactos musculares, dado que se debe al movimiento del tórax y de la cabeza al respirar (ICL, 2024).

Seguidamente, se muestran diferentes registros MEG que ilustran algunos de los artefactos comentados (Papadelis, et al., 2018). En primer lugar, en la Figura 1 puede verse un artefacto de tipo cardíaco, que refleja el comportamiento magnético del complejo QRS, es decir, la pulsación propia del ECG. Este se caracteriza por pulsaciones

Tipo de artefacto	Origen	Localización	Frecuencia	Amplitud	Manifestación
Fisiológicos					
Cardíaco	Actividad eléctrica del corazón.	Sensores cercanos al tórax	~ 1 Hz (ritmo cardíaco)	Moderada a alta	Pulsos periódicos en forma de ondas rítmicas.
Ocular	Movimientos oculares o parpadeos.	Sensores frontales y temporales	< 5 Hz (baja frecuencia)	Alta	Ondas de gran amplitud, lentas, con picos rítmicos.
Muscular	Movimientos musculares (cara, cuello, mandíbula).	Sensores temporales y occipitales	> 20 Hz (alta frecuencia)	Alta	Ruido de alta frecuencia, continuo o intermitente.
No fisiológicos					
Movimiento	Movimiento de electrodos o cables.	Electrodo afectado	Variable	Variable	Picos erráticos o cambios abruptos en la señal.
Magnético	Interferencias electromagnéticas externas.	Generalizado	Variable	Baja a moderada	Señales no correladas con la actividad neuronal.
Eléctrico	Interferencias de la red eléctrica o de dispositivos electrónicos.	Generalizado	50/60 Hz (frecuencia de red)	Variable	Señales sinusoidales repetitivas que contaminan los registros.

Tabla 1. Tipos de artefactos y sus características principales (Team, 2023).

relativamente breves que se producen al mismo tiempo que la frecuencia cardíaca del paciente. Sin embargo, no sólo el corazón es capaz de generar artefactos de origen cardíaco. Por ejemplo, si el paciente cuenta con prótesis coronarias, que son implantes colocados en el corazón para reemplazar las válvulas dañadas y corregir la insuficiencia que pueda presentar, también se pueden producir artefactos, pues estas prótesis están fabricadas con materiales paramagnéticos o ferromagnéticos (Papadelis, et al., 2018). La medición simultánea de las señales generadas tanto por el cerebro como por el corazón mediante MEG/EEG y ECG, respectivamente, permite identificar las componentes asociadas a cada uno de los órganos mencionados. Las señales neuronales mostradas se han adquirido mediante el uso de magnetómetros y gradiómetros. Los primeros miden la intensidad absoluta de los campos magnéticos en un punto del espacio, mientras que los segundos registran la diferencia de campo magnético entre dos puntos cercanos (Puce & Hämäläinen, 2017).

Por otro lado, en la Figura 2 se muestra un artefacto de tipo ocular debido a parpadeos, que contaminan las frecuencias más bajas (Papadelis, et al., 2018). También existen artefactos debidos a movimientos sacádicos de los ojos, es decir, desplazamientos rápidos oculares entre dos puntos, que suelen ser involuntarios cuando el paciente se encuentra con los ojos cerrados. Este tipo de artefactos pueden ser tratados como miogénicos debido a su comportamiento típico, pues suelen interferir con los rangos de frecuencia más altos.

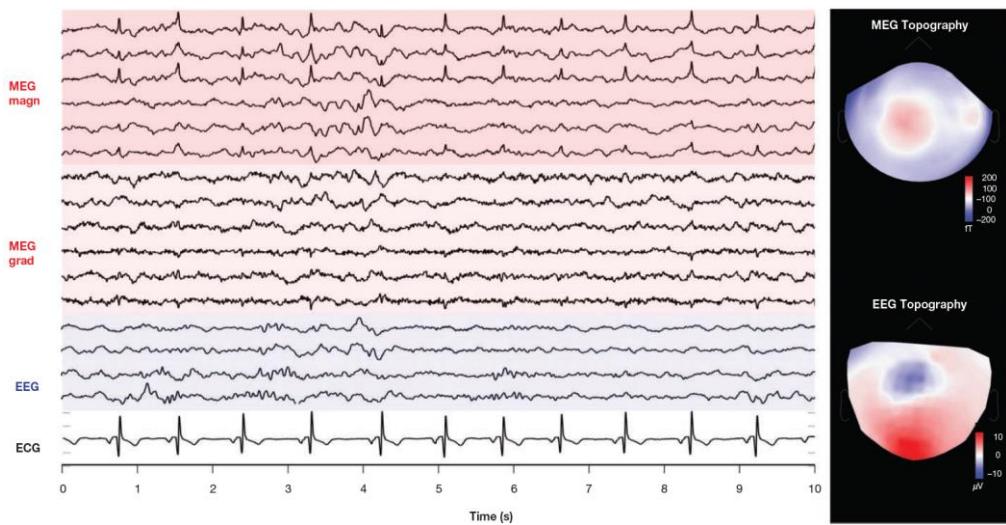


Figura 1. Artefacto cardíaco en señales MEG y EEG obtenidas en un niño con epilepsia. Primer conjunto de señales (en rojo): registros de magnetómetros. Segundo conjunto de señales (en salmon): registros de gradiómetros. Tercer conjunto de señales (en azul): registros de EEG. Cuarta señal (en negro): registro de ECG. Esquina superior derecha: localización del pico de artefacto de ECG en una topografía MEG. Esquina inferior derecha: localización del pico de artefacto de ECG en una topografía EEG (Papadelis, et al., 2018).

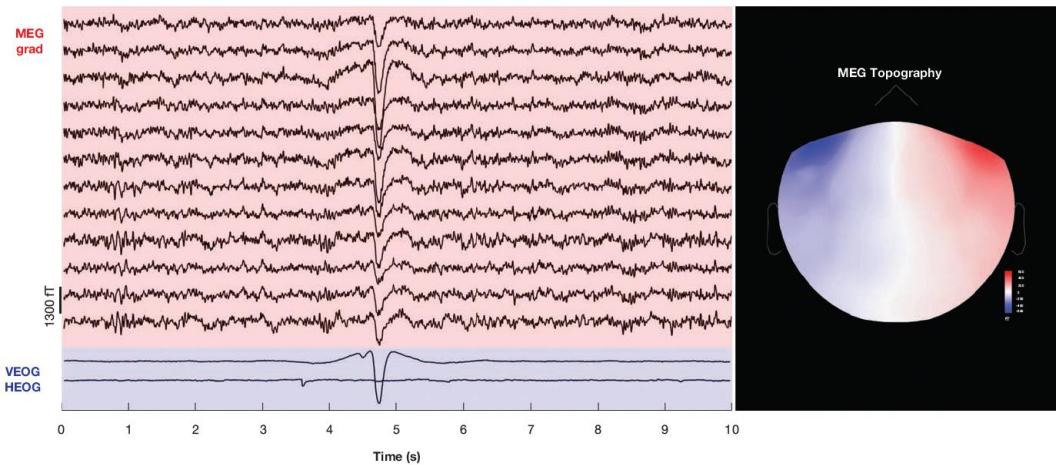


Figura 2. Artefacto ocular debido a parpadeos en señal MEG obtenida en un adulto de 24 años sano. Primer conjunto de señales (en rojo): registros de gradiómetros. Segundo conjunto de señales (en azul): registro vertical (VEOG) y horizontal (HEOG) de electrooculograma (EOG). A la derecha: localización del pico de artefacto de EOG en una topografía MEG (Papadelis, et al., 2018).

Finalmente, en la Figura 3 se puede ver un artefacto de tipo muscular generado por el músculo frontal al fruncir el ceño (Papadelis, et al., 2018). También generan artefactos que tienen un gran impacto en la señal neuronal los músculos maseteros, empleados al masticar o al imitar dicho movimiento. Tal y como se ha comentado previamente en la Tabla 1, la actividad muscular produce artefactos situados en alta frecuencia, entre 20 y 300 Hz. Así, puede solaparse por completo con el ancho de banda espectral de las oscilaciones de alta frecuencia generadas por la actividad neuronal, como las ondas gamma, lo que hace casi imposible distinguirlas de los artefactos.

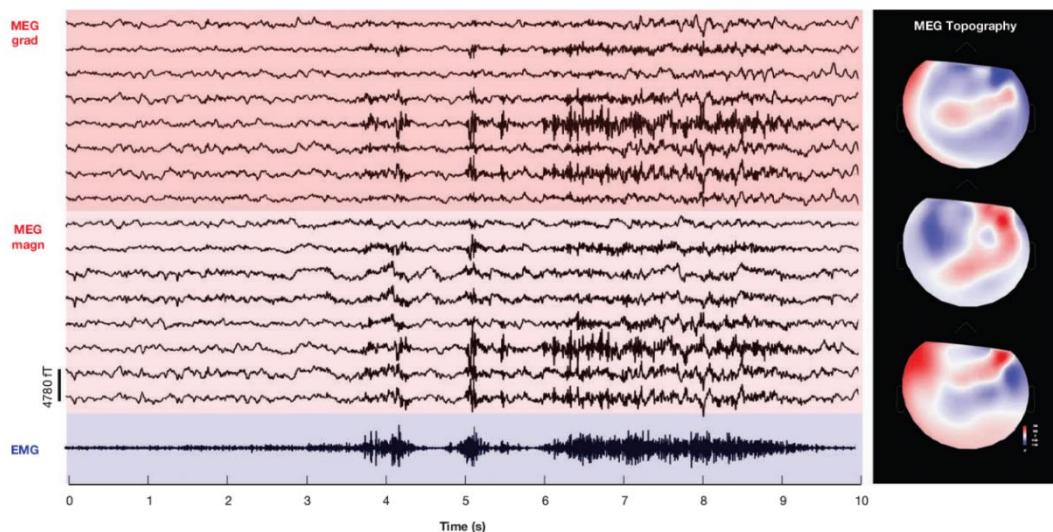


Figura 3. Artefacto muscular debido al músculo frontal en señal MEG obtenida en un niño con epilepsia. Primer conjunto de señales (en rojo): registros de gradiómetros. Segundo conjunto de señales (en salmón): registros de magnetómetros. Tercera señal (en azul): registro de EMG. A la derecha: localización del pico de artefacto de EMG en una topografía MEG (Papadelis, et al., 2018).

En cuanto a los artefactos no fisiológicos, no sólo pueden estar generados por elementos externos al cuerpo humano, sino que también pueden proceder de dispositivos médicos implantados en algunas partes del cuerpo del paciente (Papadelis, et al., 2018). Estos elementos pueden generar artefactos que provocan que la señal neuronal sea ininteligible. La Figura 4 ilustra un artefacto durante un registro de MEG en un paciente adulto con un implante coclear. Aquí puede verse que la mayoría de los magnetómetros se saturan por la perturbación del implante coclear, volviéndose incapaces de registrar cualquier actividad cerebral del paciente.

2.3. Técnicas tradicionales de reducción de ruido

Una vez adquirida la señal neuronal, antes de analizarla se debe llevar a cabo un acondicionamiento de la misma para mejorar su calidad y que su interpretación se pueda realizar de la forma más sencilla y rápida posible. A lo largo de la historia, se han desarrollado diferentes métodos para mejorar la calidad de las señales, desde sistemas mecánicos y analógicos hasta técnicas digitales basadas en modelos de IA. A finales del siglo XIX y a partir de la invención del telégrafo y el teléfono, surgió la necesidad de reducir el ruido en las señales eléctricas generadas en la comunicación (Van Bosse, 1998). En esta época, la atenuación del ruido dependía principalmente de filtros pasivos hechos con resistencias, inductancias y condensadores. Más adelante, a principios del siglo XX, con el avance de la radio y las telecomunicaciones, se popularizaron los filtros eléctricos analógicos, como los filtros paso-bajo para eliminar altas frecuencias no deseadas (Van Bosse, 1998).

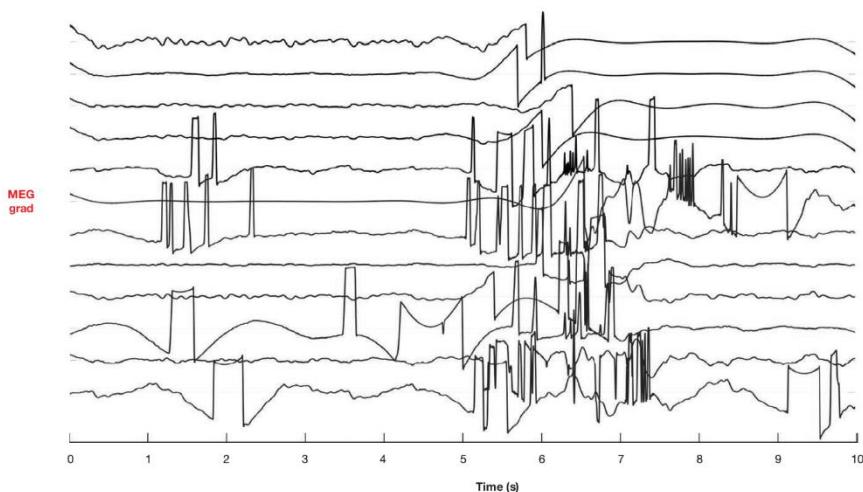


Figura 4. Artefacto no fisiológico debido a un implante coclear en un paciente adulto (Papadelis, et al., 2018).

Entre las técnicas más comunes de preprocesamiento se encuentra el filtrado digital, que se desarrolló en la década de 1960 (Proakis, 2001). Estos filtros son capaces de aplicar transformaciones matemáticas sobre la señal en tiempo real, e implican trabajar en el dominio de la frecuencia, por lo que se debe aplicar algún método basado en transformadas. Las transformadas descomponen las señales en sus componentes frecuenciales (Nitschke, et al., 1998). Esto puede ser beneficioso a la hora de detectar ruido en señales neuronales, pues muchos artefactos presentan características espectrales concretas y es más sencillo diferenciar patrones de ruido o cerebrales, pudiendo separar unos de otros. Para obtener el comportamiento frecuencial de una señal que originalmente está caracterizada de forma temporal se han desarrollado métodos basados en transformadas matemáticas, como la Transformada Rápida de Fourier (FFT, *Fast Fourier Transform*) o la Transformada *Wavelet*. La primera permite descomponer una señal en sus componentes espectrales, proporcionando información sobre qué frecuencias están presentes, pero no sobre cuándo ocurren (Ambardar, 1995). Esta limitación afecta especialmente a las señales de origen neuronal, donde los patrones de actividad varían con el tiempo. Para abordar este problema, la Transformada *Wavelet* introduce una representación en tiempo-frecuencia, ofreciendo una mejor resolución temporal para las componentes situadas en frecuencias altas y una mejor resolución frecuencial para las ubicadas en frecuencias bajas, permitiendo detectar cambios espectrales de manera dinámica (Azzerboni, et al., 2004). Esta flexibilidad hace que la Transformada *Wavelet* sea de gran utilidad para analizar eventos transitorios en EEG y MEG, como picos de actividad o respuestas ante estímulos, mientras que la FFT se emplea para examinar el contenido espectral global y eliminar artefactos de frecuencia fija, como el ruido eléctrico.

En cuanto a los filtros, se trata de dispositivos que permiten aislar un rango determinado de frecuencias dentro de una señal eléctrica, ajustando su amplitud y fase, si se desea (Semmlow, 2004). Su objetivo principal es diferenciar la información relevante de posibles interferencias, ruido o distorsiones no deseadas, mejorando así la calidad de la señal procesada. Existen diferentes tipos de filtros, y cada uno se aplica según las necesidades requeridas, pero los que más sentido tiene aplicar durante el procesamiento de señales neuronales son los filtros paso-bajo, paso-alto y de ranura. Por un lado, los filtros paso-bajo permiten el paso de las señales que los atraviesan desde la frecuencia cero hasta una frecuencia de interés, denominada frecuencia de corte, a partir de la cual la atenuación aplicada a la señal va siendo cada vez mayor (Sen, et al., 2023). Este tipo de filtros puede emplearse para reducir el ruido de alta frecuencia en las señales EEG o MEG, como artefactos musculares, interferencias electromagnéticas de alta frecuencia o ruido de los equipos de adquisición, que suelen estar en un rango de frecuencias superior al de las oscilaciones cerebrales relevantes. Así, es más fácil identificar las componentes de interés, como las ondas delta (0.5 – 4 Hz), zeta (4 – 8 Hz), alfa (8 – 13 Hz) y beta (13

– 30 Hz), que tienen frecuencias más bajas (Sanei & Chambers, 2013). Por otro lado, los filtros paso-alto permiten el paso de las frecuencias altas de las señales desde una frecuencia de corte superior a cero (Sen, et al., 2023). Artefactos como movimientos oculares o variaciones en la impedancia de los electrodos, que generan componentes de muy baja frecuencia, pueden eliminarse con este tipo de filtros y conservar componentes cerebrales de frecuencias altas, como las ondas gamma, cuyas oscilaciones son superiores a los 30 Hz (Sanei & Chambers, 2013). Finalmente, los filtros de ranura y los filtros elimina-banda permiten el paso de toda la señal salvo la parte comprendida entre dos frecuencias concretas (Sen, et al., 2023) (Hirano, et al., 1974). Con esto se pueden eliminar las interferencias debidas a la red eléctrica, es decir, las oscilaciones situadas en 50 o 60 Hz y sus armónicos, u otro tipo de ruido que corrompa la señal cerebral en bandas intermedias (Puce & Hämäläinen, 2017). La ventaja de este tipo de filtros respecto de los otros dos comentados es que eliminan frecuencias específicas de interferencia sin afectar a otras bandas. El principal problema de los filtros paso-bajo y paso-alto es que no son tan selectivos como para eliminar únicamente las componentes contaminadas con artefactos, sino que también rechazan componentes neuronales (Puce & Hämäläinen, 2017). Esto supone un grave problema para el diagnóstico de enfermedades, como la epilepsia o el Alzheimer, dado que se prescinde de información importante de la actividad cerebral del paciente. En la Figura 5 puede verse una señal EEG y su posterior filtrado tras haberse aplicado un filtro paso-banda de 0.1 a 60 Hz para reducir la presencia de artefactos y un filtro de media móvil de 0 a 55 Hz para eliminar variaciones rápidas no deseadas y hacer que la señal sea más estable (Arslan, et al., 2015).

A pesar de que el filtrado en el dominio frecuencial puede ser una herramienta de gran utilidad para reducir el ruido en señales neuronales, su aplicación puede conllevar ciertas

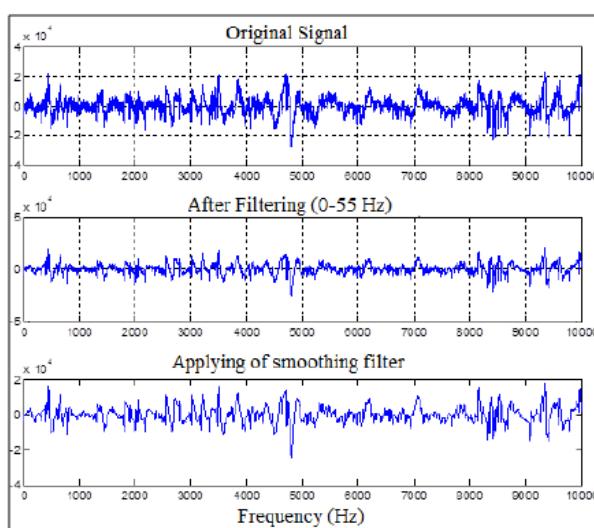


Figura 5. Filtrado de una señal EEG. Gráfica superior: señal EEG original. Gráfica intermedia: señal EEG original tras haber aplicado un filtro de 0.1 a 60 Hz. Gráfica inferior: señal EEG filtrada tras haber suavizado la señal (Arslan, et al., 2015).

desventajas, como la posible distorsión de la señal original o la eliminación de información neuronal relevante. Además, los filtros convencionales no siempre son eficaces para separar componentes solapadas en el dominio del tiempo o en las que su banda de frecuencias característica coincide con la del ruido, tal y como ya se ha comentado previamente. Por este motivo, surge la necesidad de aplicar otro tipo de técnicas que preserven la información neuronal eliminando la mayor parte posible de los artefactos que la contaminan.

2.4. Reducción de artefactos en señales neuronales

Reducir el impacto de los artefactos o eliminar su presencia en las señales electromagnéticas cerebrales es necesario para poder llevar a cabo una interpretación precisa de los eventos que codifican; además, garantiza una localización de fuentes exacta (Islam, 2015). Las fuentes se refieren a las regiones específicas del cerebro donde se origina la actividad neuronal que genera las señales registradas mediante técnicas como EEG o MEG (Hari, et al., 2000). Estos métodos no miden directamente la actividad cerebral en su origen, sino que capturan una representación externa de los campos eléctricos o magnéticos producidos por la suma de miles o millones de neuronas activas en una zona concreta. Para realizar una localización de fuentes, los algoritmos deben procesar estas señales registradas y estimar las ubicaciones cerebrales exactas de las fuentes a partir de los datos medidos en los sensores. Sin embargo, la presencia de artefactos, ya sean fisiológicos o no fisiológicos, introduce ruido que puede distorsionar los cálculos y generar resultados erróneos o imprecisos (Islam, 2015). La localización de fuentes se emplea para identificar qué áreas del cerebro están involucradas en funciones biológicas específicas, para localizar la región del cerebro donde se originan las crisis en pacientes con epilepsia o para determinar qué regiones deben evitarse para minimizar el impacto sobre funciones críticas en una cirugía (Michel & He, 2019). Por eso, la correcta identificación y eliminación de los artefactos es indispensable para que la señal residual refleje con mayor fidelidad la actividad cerebral de interés.

Algunos de los métodos que se emplean para mitigar la presencia de artefactos son en los siguientes momentos (Burgess, 2020):

- i. Durante la configuración y conexión al sistema de medida.
- ii. Durante la adquisición de la señal.
- iii. En el preprocesado de la señal ya adquirida, empleando un software de reducción de artefactos.

Para el primer caso y, por ejemplo, para MEG, se pueden emplear gradiómetros en lugar de magnetómetros, dado que son menos sensibles a las interferencias de fuentes de artefactos ambientales, ya que este ruido suele ser homogéneo en el espacio cercano y se cancela al calcular el gradiente (Vrba & Robinson, 2001). También se pueden incorporar al realizar la medición unos sensores de referencia situados a mayor distancia de la cabeza, que se utilizan para restar los artefactos externos (Puce & Hämäläinen, 2017). Además, se debe evitar cualquier material magnético en la ropa del paciente y, en el caso de que este tenga implantes, se pueden emplear equipos especiales que reduzcan las señales magnéticas producidas por los metales (Mosher & Funke, 2020).

Para el segundo método, el hecho de registrar señales libres de artefactos depende, en gran medida, del comportamiento del paciente. Factores como permanecer inmóvil, minimizar movimientos musculares (especialmente en la cabeza, ojos y mandíbula) y evitar actividades que puedan generar interferencias pueden reducir la probabilidad de que aparezcan componentes indeseados en las señales neuronales (Larson & Taulu, 2017). Por ejemplo, para minimizar los artefactos oculares se le puede pedir al paciente que fije la vista en un punto para restringir el movimiento ocular. Sin embargo, este método no es útil en niños de corta edad o pacientes con algún tipo de discapacidad neurológica (Puce & Hämäläinen, 2017). En ese caso, sería adecuado medir simultáneamente señales de EOG o ECG para poder identificar correctamente los artefactos (Muthukumaraswamy, 2013).

En la etapa del preprocesado de la señal se lleva a cabo un filtrado en el rango de frecuencias de interés como primera aproximación (Bashashati, et al., 2007). Sin embargo, el filtrado no es apropiado cuando el rango de frecuencias de los artefactos se solapa con el espectro de la actividad cerebral, tal y como se ha comentado previamente (Jiang, et al., 2019). Existen algunas técnicas que permiten excluir los sensores que están contaminados con artefactos, como técnicas basadas en métodos estadísticos, correlación o modelos de IA. Las dos primeras se basan en identificar sensores que muestran características anómalas en comparación con el resto (Larson & Taulu, 2017). La tercera emplea algoritmos que implementan procedimientos automáticos de clasificación para identificar y excluir sensores, ya sea mediante técnicas de ML o DL (Mumtaz, et al., 2021). Estos algoritmos se pueden complementar con algunos de los métodos estadísticos basados en la Separación Ciega de Fuentes (*Blind Source Separation*, BSS) o el Análisis de Componentes Independientes (*Independent Component Analysis*, ICA). Estas técnicas permiten identificar y aislar patrones específicos de artefactos en las señales neuronales, basándose en la idea de que las fuentes subyacentes (actividad cerebral y artefactos) son estadísticamente independientes (Yang, et al., 2021). El combinar ambos métodos permite que las componentes que sean altamente ruidosas sean descartadas automáticamente, mientras que las señales de los sensores restantes se puedan limpiar

eficazmente eliminando patrones específicos de artefactos (Vigario & Oja, 2008). A continuación, se exponen con más detalle estas y otras técnicas de preprocesado encargadas de detectar y eliminar la presencia de artefactos en señales neuronales.

2.4.1. Descomposición de señales y clasificación manual

La tecnología y la investigación en neurociencia avanzan a un ritmo acelerado, y con ellas los métodos más empleados hasta la fecha van quedando obsoletos. Así, surgen nuevas técnicas más precisas y fiables que mejoran la comprensión del funcionamiento del cerebro humano y consiguen acelerar el proceso de detección y diagnóstico de enfermedades neuronales. Respetando el objetivo principal del preprocesado de señales neuronales, esto es, la eliminación de información que no es de origen cerebral, han surgido dos técnicas estadísticas fundamentales que permiten separar los artefactos de las señales neuronales para, posteriormente, poder realizar una clasificación de las componentes generadas: PCA e ICA.

PCA se trata de una técnica de reducción de la dimensionalidad del conjunto de datos disponible, dado que permite disminuir el tamaño del vector de características sin perder información importante de la señal y, a su vez, revelar patrones que en la señal original podrían estar ocultos (Tibaduiza, et al., 2013). Su funcionamiento se basa en la aplicación de una transformación ortogonal que convierte un conjunto de observaciones de variables correlacionadas en un conjunto de variables linealmente no correlacionadas, denominadas componentes principales (Kaya, 2019). Esto se consigue a partir de la premisa de que la señal limpia tiende a concentrarse en las primeras componentes principales (las de mayor varianza), mientras que el ruido se localiza principalmente en las últimas componentes.

Por otro lado, ICA realiza una separación de las señales adquiridas en componentes independientes, que se corresponden con la estimación de las diferentes fuentes que han originado dichas señales (Croce, et al., 2019). Por este motivo se dice que ICA está basado en la Separación Ciega de Fuentes (BSS, *Blind Source Separation*), que es el proceso de separar señales que originalmente se encuentran mezcladas entre sí, aunque se disponga de muy poca información sobre las mismas (Pal, et al., 2013). Además, supone independencia estadística entre esas fuentes, y asume que no existe correlación temporal, frecuencial o espacial entre las distintas señales sobre las que se aplica (Urigüen & García-Zapirain, 2015). En cuanto a la primera condición, considerando dos variables escalares aleatorias, por ejemplo y_1 e y_2 , se dice que son independientes si la información en y_1 no aporta nada de información en y_2 , y viceversa. Por otro lado, dos variables aleatorias están incorreladas si su covarianza es igual a cero. A partir de estas premisas,

se puede asumir que, si las variables son independientes, están incorreladas. Sin embargo, en el caso contrario esta suposición no se cumple.

En la Figura 6 se muestran dos conjuntos de señales tras haber aplicado ICA y PCA sobre una base de datos procedente de un registro MEG realizado en una sala magnéticamente aislada empleando 122 canales (Jafarabadi, et al., 2015). Los datos incluyen señales procedentes del cerebro, artefactos fisiológicos y ruido externo. En la gráfica correspondiente a ICA se pueden distinguir nueve componentes independientes obtenidas. En ellas, es sencillo diferenciar los distintos artefactos: la primera (IC1), segunda (IC2) y cuarta (IC4) componente contienen artefactos musculares; la tercera (IC3) y la quinta (IC5) representan artefactos oculares debidos a movimientos de los ojos y parpadeos, respectivamente; la sexta (IC6) y la séptima (IC7) se deben al movimiento del cuerpo al respirar; la octava (IC8) es el artefacto cardíaco y, finalmente, la novena (IC9) se corresponde con el ruido procedente de los equipos de adquisición. En la gráfica obtenida tras aplicar PCA puede verse que las componentes de ruido no se han separado correctamente: la primera y segunda componentes (C1 y C2) son una mezcla de artefactos musculares y oculares, que equivalen a una combinación de las componentes independientes IC2 e IC3 obtenidas con ICA. Algo similar sucede con la tercera componente, que se trata de la combinación de IC2, IC3 e IC5. Como puede verse, para este caso ha funcionado mejor ICA que PCA y, generalmente, aplicado en el ámbito de señales neuronales, ese va a ser el comportamiento habitual. Esto se debe a que los artefactos fisiológicos suelen ser independientes de la actividad neuronal y, por lo tanto, las características del conjunto de datos se ajustan más a las condiciones necesarias para poder aplicar ICA (Jafarabadi, et al., 2015).

Sin embargo, ni PCA ni ICA llevan a cabo una clasificación de componentes, por lo que es necesario disponer de algún método adicional que permita clasificar las componentes generadas tras haber aplicado ICA o PCA, según sean de origen neuronal o

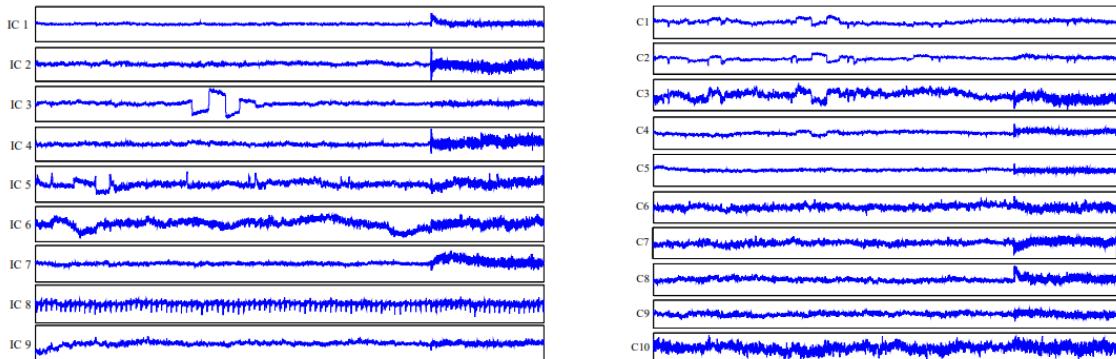


Figura 6. Conjunto de señales estimadas mediante ICA (izquierda) y mediante PCA (derecha) (Jafarabadi, et al., 2015).

artefactos. La forma habitual de realizar dicha clasificación es manualmente, y el proceso se conoce comúnmente como etiquetado (Croce, et al., 2019). Uno o varios técnicos especialistas tratan de buscar características distintivas de las señales y patrones que las caractericen. Tal y como se ha visto previamente, las formas de onda de los artefactos más comunes presentan rasgos característicos que hacen posible su identificación. Este proceso requiere experiencia y criterio de la persona encargada del etiquetado y, en muchas ocasiones, se complementa su análisis con el realizado por el de más expertos para contrastar la información y disminuir los sesgos individuales.

A pesar de su utilidad, la clasificación manual es una tarea ardua que requiere mucho tiempo para poder llevarla a cabo e implica un alto grado de subjetividad. Esto hace que los estudios se prolonguen durante períodos largos de tiempo y la fiabilidad de los diagnósticos sea baja, afectando directamente a los pacientes implicados y, en casos extremos, disminuyendo su tiempo de vida (Croce, et al., 2019). Además, las técnicas de descomposición de señales en componentes vistas presentan ciertas desventajas. Por un lado, PCA no es capaz de separar las componentes contaminadas con artefactos de las componentes neuronales cuando sus amplitudes son del mismo orden de magnitud (Jiang, et al., 2019). Así, PCA únicamente es útil cuando los niveles de ruido son bajos y, además, los subespacios de señal y de ruido son ortogonales entre sí. Esto implica que las direcciones principales de variabilidad de la señal no se mezclen con las del ruido, sino que estén en dimensiones separadas dentro del espacio de representación de los datos (Jafarabadi, et al., 2015). Dada la naturaleza dinámica y estocástica de las señales procedentes del cuerpo humano, suponer ortogonalidad en dichas señales sería la primera causa de fallo. Por otro lado, ICA no garantiza que las componentes independientes generadas contengan exclusivamente ruido y no información neuronal importante del paciente, dado que no siempre se puede suponer que existe independencia estadística entre las fuentes (Chawla, 2011). Con todo y con eso, con ICA se obtiene un mejor rendimiento que con PCA, tal y como se ha demostrado en el estudio mostrado previamente.

2.4.2. Tendencia actual: técnicas de IA

La contaminación de las señales neuronales con artefactos supone un problema crítico en el diagnóstico de enfermedades cerebrales, ya que existe la posibilidad de que esas señales contaminadas sean tratadas erróneamente como actividad neuronal genuina (Muthukumaraswamy, 2013). Este error puede llevar a interpretaciones incorrectas sobre el comportamiento cerebral subyacente, lo que compromete tanto los estudios clínicos como los experimentos científicos. También puede darse la situación contraria, es decir, que una señal de origen cerebral sea erróneamente descartada al ser interpretada como

ruido (Jiang, et al., 2019). Este y otros problemas, junto con la necesidad de los seres humanos por realizar tareas de manera automática y de la forma más rápida posible, ha llevado al desarrollo de herramientas que imiten el comportamiento humano y mejoren su rendimiento conforme vayan recopilando más información. La capacidad de estas herramientas que les permite realizar dichas tareas se denomina “inteligencia artificial” (Rouhiainen, 2018). Concretamente, su uso ha revolucionado el tratamiento de señales biomédicas y ha mejorado la calidad de los diagnósticos, ya que ha permitido prescindir parcialmente de la intervención humana, disminuyendo la variabilidad en los resultados (AlHinai, 2020).

Las diferencias entre los modelos tradicionales de procesado y el uso de modelos basados en IA se ven reflejadas en términos de automatización, precisión y escalabilidad (Jiang, et al., 2022). Las técnicas clásicas vistas anteriormente, como el filtrado o la clasificación basada en la descomposición de señales, han demostrado su eficacia en la eliminación de artefactos en señales neuronales. Sin embargo, en muchas ocasiones se hace inviable combinarlas todas debido al coste computacional y temporal que ello implica (Barbati, et al., 2004). Además, requieren un ajuste manual y dependen en gran medida de la experiencia del usuario, mientras que los algoritmos de IA pueden aprender de grandes volúmenes de datos y realizar tareas de forma autónoma, pues cuenta con la capacidad de encontrar patrones y correlaciones en los datos mucho más rápido que un humano (Callier & Sandel, 2021). A mayores, la escalabilidad de estos sistemas permite su aplicación en grandes volúmenes de datos, necesarios en estudios a gran escala y aplicaciones en tiempo real, como el monitoreo de pacientes en entornos clínicos (Hamet & Tremblay, 2017). Sin embargo, puede resultar conveniente en algunos casos no desechar por completo alguna de las técnicas convencionales y combinarla con los algoritmos de IA. En este sentido, tanto PCA como ICA pueden emplearse como un paso previo a la clasificación de señales neuronales con el fin de simplificar el conjunto de datos y extraer sus características más relevantes en el caso de PCA, o separarlos en componentes independientes según sus fuentes en el caso de ICA, tal y como se ha comentado previamente (Lakshmi, et al., 2017) (Urigüen & Garcia-Zapirain, 2015). Con esto se consigue optimizar la base de datos y, por lo tanto, mejorar la calidad de las detecciones por parte de la herramienta de IA empleada.

Dentro de la IA existen diversas técnicas que permiten realizar diferentes tareas y que, por lo tanto, están diseñados para aplicaciones específicas. En el ámbito de la neurociencia, los más empleados son las basadas en ML y, más recientemente, en DL. Por un lado, el ML permite a los sistemas aprender automáticamente a partir de los datos, sin necesidad de ser programados explícitamente (Vu, et al., 2018). Por otro lado, el DL es una rama avanzada del ML que emplea redes neuronales profundas para realizar análisis más complejos, siendo especialmente eficaz en el procesamiento de señales

biomédicas (Marblestone, et al., 2016). Los algoritmos de ML permiten clasificar segmentos de señales contaminadas con artefactos y mejorar la calidad de los datos. Sin embargo, el éxito del ML depende de características extraídas manualmente por los expertos, lo que puede introducir sesgos y limitar el rendimiento del modelo (Zheng & Casari, 2018). En contraposición, el DL elimina la necesidad de extraer características manualmente, ya que las redes neuronales son capaces de aprender representaciones directamente a partir de los datos originales. Además, el empleo de DL permite abordar la naturaleza no lineal de las señales cerebrales, lo que supone una complicación añadida para su análisis (Bitbrain, 2020).

Una vez elegido el modelo de IA que se va a emplear se debe configurar su arquitectura, que definirá el comportamiento del modelo en cuestión. Una de las más empleadas para el análisis de señales cerebrales son las Redes Neuronales Artificiales (ANN, *Artificial Neural Networks*). Estas se basan en el comportamiento de las neuronas y sus interconexiones construyendo arquitecturas de redes neuronales, lo que les permite aprender y tomar decisiones identificando patrones y clasificando datos (Walczak, 2019). Las ANN están formadas por muchas neuronas, llamadas perceptrones, cuya función es modificar las entradas que se les introducen multiplicándolas por unos pesos aleatorios y sumándolas entre sí con un valor de sesgo (Jain, et al., 1996). Tanto los pesos como el sesgo aplicado se pueden ajustar según se deseé y determinan el resultado concreto que va a proporcionar el perceptrón. A este resultado se le aplica una función de activación, que decide si el dato proporcionado se debe introducir por un nuevo perceptrón o no (Antona-Cortés, 2017).

Dentro de las ANN existen diferentes modelos de redes neuronales, como las CNN o las Recurrentes (RNN, *Recurrent Neural Networks*). Las primeras constan de varias capas de perceptrones y filtros convolucionales de diferentes dimensiones. Estos filtros se denominan *kernel* y, según la matemática convolucional implicada, la CNN realiza un producto escalar entre un filtro *kernel* y los datos de entrada con el fin de conseguir las singularidades más significativas de dichos datos. Además, contienen las capas de *pooling*, que reducen el tamaño de los datos a su entrada conservando las características más relevantes, las completamente conectadas, donde se refleja que cada entrada aplicada a cada neurona afecta a cada una de las salidas proporcionadas dado que tienen en cuenta toda la información extraída para llevar a cabo una clasificación, y las de salida (O'Shea & Nash, 2015). A su vez, las RNN están formadas por varias capas que toman la información de entradas anteriores que influyen en la entrada y salida actuales, es decir, se trata de un sistema que tiene memoria. Esto permite captar la evolución de los datos que se quieran analizar a lo largo del tiempo (Grossberg, 2013). Sin embargo, las CNN son más adecuadas para clasificar datos, como por ejemplo componentes ICA neuronales con artefactos, debido a su alta capacidad para reconocer patrones, clasificar datos y

detectar componentes dentro de un conjunto de valores (Junliang, 2022). Las RNN se emplean para tareas en las que el contexto y la secuencia de los datos son de vital importancia, como la generación de texto o la traducción automática (Guan, et al., 2017). Por estos motivos, en el presente TFM se ha optado por emplear una arquitectura de red convolucional. En la Figura 7 pueden verse las diferencias entre las distintas capas que forman las redes CNN y RNN. En las primeras puede verse cómo las neuronas se conectan exclusivamente con algunas de las neuronas adyacentes, mientras que en las segundas se ve que las neuronas pueden recibir información de todas las neuronas de la capa anterior (Zhang, et al., 2021).

Entre las herramientas más modernas destacan plataformas como EEGNet, una arquitectura de CNN diseñada específicamente para trabajar con señales electrofisiológicas, empleada para la identificación de artefactos oculares y musculares en señales provenientes de registros de EEG (Lawhern, et al., 2018). A su vez, existen otras herramientas adaptadas específicamente a señales adquiridas mediante MEG, como por ejemplo MEGnet (Garg, et al., 2021). En este caso, la tarea principal consiste en la identificación de artefactos cardíacos y oculares mediante la aplicación de una CNN tras haber aplicado ICA. Asimismo, se cuenta con softwares más generalistas, como TensorFlow o PyTorch, que permiten implementar modelos de eliminación de artefactos, pudiendo ajustar los hiperparámetros que caracterizan la red neuronal programada, como el número de neuronas o las capas ocultas que la forman (Novac, et al., 2022).

La elección del modelo y de la arquitectura de este que más se ajuste al problema que se desee resolver es sólo el paso inicial en todo el proceso de la aplicación de los modelos de IA. El entrenamiento de la red sería la siguiente etapa. Aquí se realiza un proceso iterativo de ajuste de pesos. Inicialmente, la red recibe un conjunto de datos etiquetados divididos en fragmentos que atraviesan múltiples capas convolucionales y de activación,

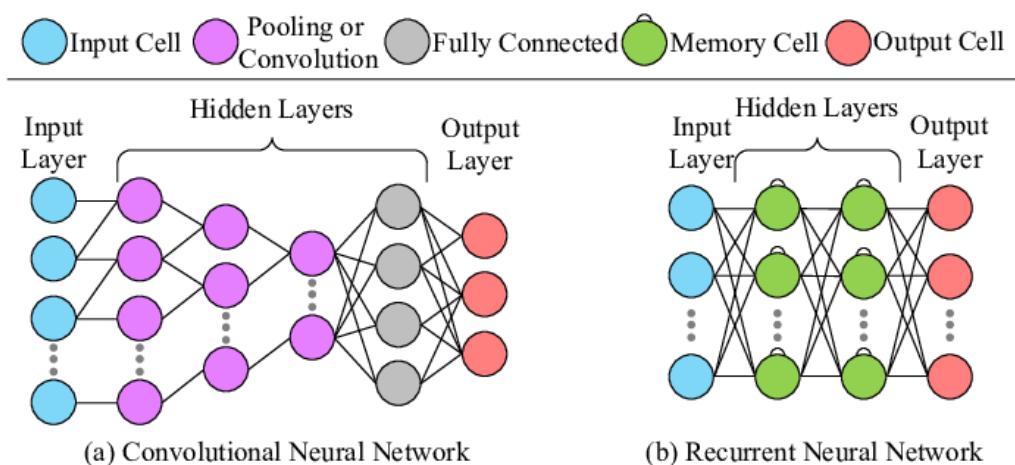


Figura 7. Estructura general de una CNN (a) y de una RNN (b) (Zhang, et al., 2021).

extrayendo características cada vez más abstractas. A cada uno de esos fragmentos se le aplica un peso (Ros-García, 2019). Por ejemplo, si el peso aplicado al primer fragmento es 1 y el peso aplicado al segundo fragmento es 10, entonces el segundo fragmento tendrá un efecto 10 veces superior al del primer fragmento. Luego, la salida se compara con la etiqueta real para calcular el error mediante una función de pérdidas (Janocha & Czarnecki, 2017). Este error se propaga hacia atrás (*backpropagation*) a través de la red utilizando algoritmos de optimización, como Adam (*Adaptive moment estimation*), para ajustar los pesos y mejorar la precisión del modelo (Taqi, et al., 2018). A lo largo de varias épocas, la CNN aprende patrones relevantes, reduciendo el error y mejorando su capacidad de generalización sobre datos no vistos. Sin embargo, no todo el conjunto de los datos se emplea para entrenar la red, sino que se reserva una parte para la etapa de validación y otra para la etapa de test (Lee, et al., 2019). La primera se emplea durante el entrenamiento y tiene como objetivo evaluar el desempeño del modelo y evitar el sobreajuste u *overfitting*, situación que se da cuando el modelo se ha ajustado demasiado a los datos de entrenamiento y no es capaz de generalizar a datos nuevos (O'Shea & Nash, 2015). El conjunto de validación permite ajustar los hiperparámetros de la red, como el número de neuronas y sus capas (Andonie, 2019). Una vez finalizado el entrenamiento, se evalúa el modelo con el conjunto de test, que contiene datos completamente nuevos para la red. Esta fase permite medir la capacidad real del modelo para generalizar a datos no vistos y evaluar su comportamiento global (Ma, et al., 2021).

Existen múltiples algoritmos de optimización, pero el que se ha empleado en este TFM es Adam, mencionado previamente. Está basado en el descenso del gradiente estocástico, y combina las técnicas AdaGrad (*Adaptive Gradient algorithm*) y RMSProp (*Root Mean Square Propagation*) (Kingma & Ba, 2014). En la primera se ajusta la tasa de aprendizaje de manera adaptativa, de modo que aquellos parámetros con gradientes grandes reciben actualizaciones más pequeñas, mientras que los que tienen gradientes más pequeños se ajustan en mayor medida. En la segunda se ajusta la tasa de aprendizaje de manera individual para cada peso evitando que se reduzca demasiado con el tiempo. Para lograrlo, en lugar de acumular todos los gradientes anteriores, calcula una media móvil del cuadrado de los gradientes usando un decaimiento exponencial y adapta el factor de entrenamiento dividiéndolo entre la media del declive exponencial del cuadrado de los gradientes (Taqi, et al., 2018).

Finalmente, para determinar el grado de activación que alcanza cada neurona de la red, es decir, si los datos que contienen son significativos, se aplica una función de activación específica. Las más comunes son la sigmoide, la tangente hiperbólica, ReLu y SoftMax (Sharma, et al., 2017):

- Por un lado, la función sigmoide se emplea en problemas de clasificación binaria dado que transforma la entrada real a un valor entre 0 y 1, es decir, asigna una probabilidad a cada dato, lo que hace que se pueda asignar a una clase específica. La gráfica que la caracteriza se muestra en la Figura 8 (a), que equivale a la siguiente expresión matemática:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

- Por otro lado, la función tangente hiperbólica (*tanh*) genera valores en un rango de -1 a 1, por lo que amplía el conjunto de valores que puede tomar la entrada en comparación con la función sigmoide. Además, está centrada en cero, lo que favorece que los pesos se actualicen de forma más eficiente durante el entrenamiento y el aprendizaje sea más rápido. Su gráfica se muestra en la Figura 8 (b) y su expresión matemática es la siguiente:

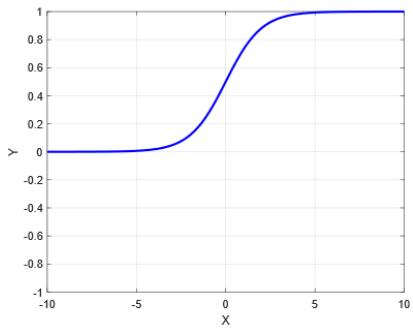
$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2)$$

- A su vez, ReLu devuelve 0 cuando la entrada es negativa y el propio valor de la entrada cuando esta es positiva, es decir, para valores positivos presenta un comportamiento lineal con un gradiente igual a 1, por lo que mejora la convergencia del entrenamiento. Esta función es menos costosa computacionalmente que la función sigmoide o la *tanh* debido a su sencillez matemática. Su gráfica puede verse en la Figura 8 (c), que se corresponde con la siguiente expresión:

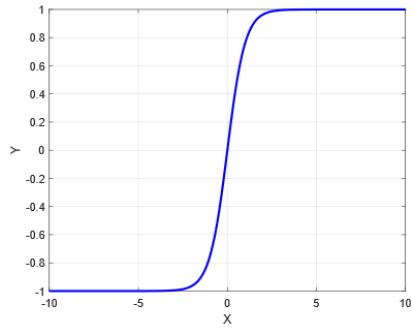
$$f(x) = \max(0, x) \quad (3)$$

- Finalmente, SoftMax se emplea para resolver problemas de clasificación multiclas, dado que la salida que ofrece representa la probabilidad de que la entrada pertenezca a una determinada clase. La gráfica correspondiente se muestra en la Figura 8 (d), que equivale a la siguiente expresión matemática:

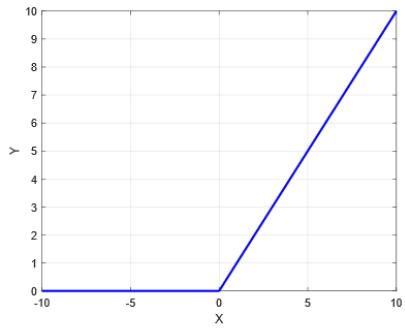
$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}} \quad (4)$$



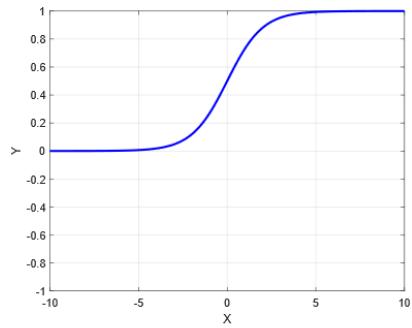
(a) Función sigmoide.



(b) Función tangente hiperbólica.



(c) Función ReLu.



(d) Función SoftMax.

Figura 8. Representación gráfica de las funciones de activación más comunes.

2.5. Señales sintéticas

Contar con la técnica más idónea de entrenamiento y optimización no sólo depende del problema específico que se quiera abordar, sino también del volumen de datos disponibles para el entrenamiento del modelo. En muchas ocasiones, disponer de la base de datos adecuada para un correcto entrenamiento de los algoritmos supone una dificultad añadida, no sólo por la calidad necesaria para obtener los resultados deseados, sino porque obtener grandes volúmenes de datos es muy costoso temporal y computacionalmente (Kaplan, et al., 2005) (Colomer, et al., 2016). Aquí es donde toma un papel importante la generación de señales sintéticas. Se entiende por señales sintéticas a datos generados de forma artificial que imitan las características de las señales neuronales reales, incluyendo tanto la actividad cerebral como los artefactos (Carre, et al., 2023). Como se trata de señales creadas para un fin y que no son generadas directamente por el cuerpo humano, se las puede configurar como se deseé, dando mayor relevancia a la parte neuronal o haciendo que sea la parte contaminada por artefactos la que predomine. El propósito de la generación de señales sintéticas radica en proporcionar un entorno controlado y conocido a priori, en el que se puedan desarrollar y evaluar de forma sistemática los métodos de

detección y eliminación de artefactos sin depender exclusivamente de datos reales (Laguna, et al., 2024). Además, permite llevar a cabo un análisis comparativo entre diferentes modelos de eliminación de artefactos para evaluar su comportamiento, ya que la proporción de ruido y señal introducida es conocida (Carrle, et al., 2023).

Las señales sintéticas se pueden generar de diferentes maneras, pero todas se basan en la combinación de señales neuronales simuladas con señales de artefactos, bien adquiridas con los sensores específicos o creadas con modelos matemáticos de artefactos (Betzel & Bassett, 2017). Para reproducir diferentes condiciones de contaminación se pueden modificar algunos de los parámetros que caracterizan a las señales, como la amplitud, la frecuencia o la SNR (Laguna, et al., 2024). Existen diversas herramientas que, aplicando distintos métodos, permiten generar señales sintéticas. El módulo *SourceSimulator* de MNE-Python es una de ellas. Este genera datos sintéticos simulando la actividad cerebral en las fuentes de la corteza y proyectándola hacia los sensores MEG/EEG, con la posibilidad de agregar ruido (MNE, 2024). En la Figura 9 puede verse el resultado de simular el módulo mencionado con unas condiciones de actividad cerebral y de ruido específicas. En primer lugar, se generaron señales simuladas desde fuentes corticales; a continuación, se proyectaron a los sensores MEG a través de un modelo de cabeza y, finalmente, se añadieron artefactos para simular unas condiciones realistas. Por otro lado, en la Figura 10 se muestran los potenciales evocados generados a partir de las señales sintéticas generadas con MNE-Python. Los potenciales evocados son el resultado del promedio de señales de EEG o MEG en múltiples repeticiones de un estímulo. La primera gráfica representa las señales adquiridas mediante 59 canales EEG, sensibles a la actividad eléctrica de la corteza cerebral, mientras que la segunda y la tercera se corresponden con registros MEG, de 203 y 102 canales, respectivamente. Las señales mostradas se calcularon promediando 49 ensayos del estímulo.

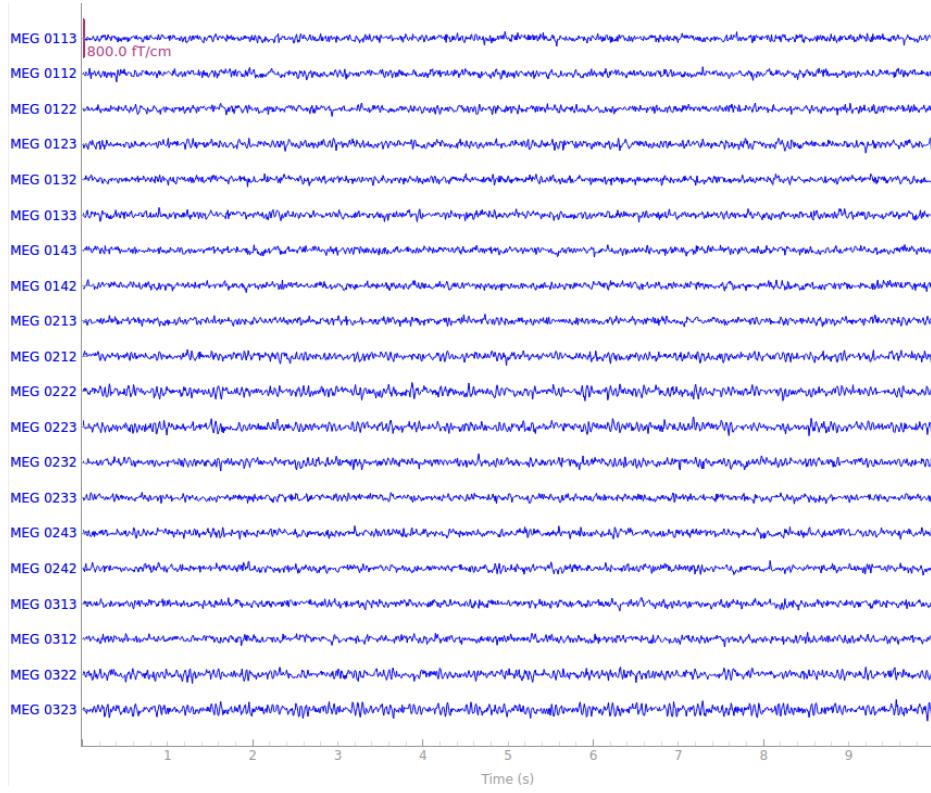


Figura 9. Señales MEG simuladas obtenidas con el módulo *SourceSimulator* de MNE-Python. Se trata de señales provenientes de diferentes sensores, con una duración de 10 segundos (MNE, 2024).

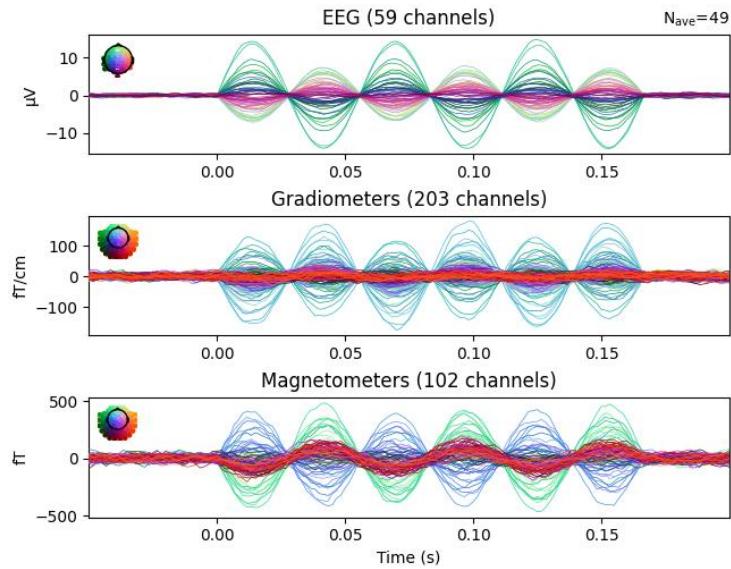


Figura 10. Potenciales evocados generados a partir de señales EEG (primera gráfica) y MEG, con gradiómetros (segunda gráfica) y magnetómetros (tercera gráfica), haciendo uso del módulo *SourceSimulator* de MNE-Python (MNE, 2024).

Capítulo 3. La herramienta MEDUSA[©]

Índice

3.1. Introducción	50
3.2. Estructura de MEDUSA[©]	51

3.1. Introducción

En las últimas décadas, el progreso científico ha sido notable en numerosos campos de la medicina, como el diagnóstico por imagen o la oncología. Sin embargo, existen otras áreas médicas, como el estudio del cerebro humano, que se encuentran en los primeros estadios de investigación, pues existen múltiples aspectos que aún no se han explorado (García-Albea, 2017). Por ello, se necesita seguir investigando para poder comprender el porqué de ciertos comportamientos humanos, la evolución de determinadas enfermedades, como las neuronales, o los tratamientos necesarios para tratarlas o evitar su avance.

En el campo de la neurociencia, muchos investigadores se encargan no sólo de estudiar la anatomía y la actividad cerebral, sino que además ofrecen sus conocimientos y permiten que otros neurocientíficos aporten valor a sus estudios. En este sentido, uno de los aspectos más críticos que supone una barrera en el avance científico es la realización de los experimentos pertinentes que, en muchas ocasiones, requieren de software específico para poder llevarse a cabo, como las interfaces cerebro-máquina (BCI, *Brain-Computer Interfaces*) (Wolpaw, 2013). Esta tecnología permite capturar en tiempo real información del cerebro en un dispositivo externo sin que sea necesaria la intervención de otras partes del cuerpo humano. Los sistemas BCI emplean técnicas de adquisición de señales como el EEG o el MEG que, posteriormente, son procesadas y analizadas por un software para extraer la información relevante (Wolpaw, et al., 2020). Hoy en día, existen múltiples plataformas que integran todas estas capas para proporcionar, en una única interfaz, un servicio de calidad, fiable y estable, como BCI2000 u OpenViBE (Schalk, et al., 2004) (Renard, et al., 2010). A pesar de que han sido las herramientas más usadas por los neurocientíficos para tratar problemas de BCI, cuentan con una serie de desventajas que limitan su funcionamiento: por un lado, no están preparadas para su aplicación en entornos cooperativos puesto que sólo son capaces de procesar unas pocas señales de entrada y no disponen de las herramientas necesarias para que los investigadores compartan sus experimentos o avances. Por otro lado, el lenguaje de programación que integra sus funcionalidades es C++ que, debido a su complejidad, no es apto para implementar los avances en BCI que van surgiendo, como los nuevos estudios basados en DL (Martínez, et al., 2021). Por este motivo, surge la necesidad de desarrollar nuevas herramientas que solucionen este tipo de problemas y mejoren la calidad de los estudios realizados. Desde el Grupo de Ingeniería Biomédica (GIB) de la Universidad de Valladolid se ha propuesto una alternativa a las plataformas BCI existentes, denominada MEDUSA[©] (MEDUSA, 2022). Se trata de un software basado en Python en el que conviven diferentes herramientas de procesado, análisis y modelado de señales cerebrales. Cuenta con una arquitectura modular y adaptable, fácil de usar y con una gran

potencia computacional que permite a los investigadores implementar sus propios algoritmos o utilizar herramientas avanzadas de IA y procesamiento de señales. En concreto, al estar basada en Python facilita la integración con librerías ampliamente utilizadas en el ámbito de la neurociencia, como *NumPy*, *SciPy*, *TensorFlow* o *PyTorch* (Santamaría, et al., 2023). A lo largo de este capítulo se describen sus características y funcionalidades principales.

3.2. Estructura de MEDUSA[©]

MEDUSA[©] se caracteriza por tener un diseño modular dividido en diversos submódulos independientes que pueden ser modificados o incluidos en el sistema sin afectar al resto de la plataforma dado que están conectados por protocolos de comunicación sencillos (Santamaría, et al., 2023). Está implementada en Python, un lenguaje de alto nivel que simplifica el uso de la herramienta a los investigadores y es compatible con diversos sistemas operativos, como Windows, macOS o Linux (Santamaría, et al., 2018). MEDUSA[©] está dividida en dos partes bien diferenciadas: el núcleo MEDUSA[©] y la plataforma MEDUSA[©] (Santamaría, et al., 2023).

Por un lado, el núcleo MEDUSA[©] contiene las herramientas necesarias para analizar las señales cerebrales obtenidas mediante distintos métodos de adquisición, como EEG o MEG, y dispone de herramientas basadas en ML y DL para procesar las señales biológicas (Santamaría, et al., 2023). A su vez, puede dividirse en diferentes submódulos según el nivel de abstracción de las funcionalidades que estos implementan. Los módulos de más bajo nivel contienen los métodos de procesamiento comunes a las distintas señales que se pueden analizar, como filtros temporales y espaciales o métricas de activación y de conectividad. En un nivel de abstracción superior se encuentran las funciones específicas para extraer y analizar las características concretas de las señales de entrada (Santamaría, et al., 2023). Los algoritmos de procesamiento empleados en estos casos se basan en los potenciales evocados P300, imaginería motora, potenciales evocados visuales modulados (c-VEPs, *modulated Visual Evoked Potentials*) y neurorretroalimentación. El primero se trata de la respuesta cerebral generada aproximadamente 300 milisegundos después de la aparición de un estímulo inesperado, y se emplea para la detección de intención (Fazel-Rezai, et al., 2012). A su vez, la imaginación motora consiste en la activación de áreas motoras del cerebro al imaginar movimientos, útil en casos en los que el paciente tiene alguna parte del cuerpo inmovilizada debido a una condición fisiológica (Wierzgała, et al., 2018). c-VEPs son las señales cerebrales inducidas por estímulos visuales repetitivos y modulados, empleadas para detectar patrones de actividad específicos (Wang, et al., 2008). Finalmente, en la neurorretroalimentación los usuarios reciben información en

tiempo real sobre su actividad cerebral para aprender a autorregularla, aplicado en terapias y optimización cognitiva (Neuper & Pfurtscheller, 2010). El núcleo MEDUSA[®] permite combinar los estudios con otros lenguajes de programación o paquetes de Python, analizar señales almacenadas previamente en bases de datos o recién adquiridas en tiempo real y personalizar las condiciones de análisis según sea necesario.

Por otro lado, la plataforma MEDUSA[®] depende directamente del núcleo MEDUSA[®], y es la interfaz gráfica ofrecida a los usuarios del software que dispone de gráficos en tiempo real y herramientas con funciones de adquisición de señales (Santamaría, et al., 2023). Además, permite la creación e instalación de nuevas aplicaciones relacionadas con los sistemas BCI. Para garantizar un correcto funcionamiento, está dividida en tres partes: la encargada de adquirir las señales biológicas, las gráficas en tiempo real y las aplicaciones (Santamaría, et al., 2023). La primera hace uso de un protocolo de transmisión en tiempo real de datos biomédicos denominado LSL (*Lab Streaming Layer protocol*), que simplifica la sincronización y el intercambio de señales entre distintos dispositivos y programas asegurando una baja latencia. Además, este protocolo permite integrar diversas fuentes de datos en un flujo común de forma simultánea (LSL, 2019). En cuanto a las gráficas en tiempo real, LSL permite visualizar las señales capturadas de forma instantánea en gráficas temporales y frecuenciales que se actualizan en milésimas de segundo. Las temporales disponen de herramientas para modificar la representación de las señales, como escalados o factores de diezmado, mientras que las frecuenciales representan la densidad espectral de potencia (PSD, *Power Spectral Density*) haciendo uso del método Welch (Santamaría, et al., 2023). Este se basa en la Transformada de Fourier para obtener la representación frecuencial de las señales mejorando la precisión de la estimación al reducir la varianza en comparación con el método clásico de Fourier (Jwo, et al., 2021). Por último, la parte de mayor peso y que implica la participación directa de los usuarios es la que contiene las aplicaciones. Estas permiten llevar un control en tiempo real del estado cognitivo del paciente mientras este realiza diferentes actividades, organizadas en varias aplicaciones independientes ofrecidas como juegos interactivos. Algunas de ellas son *RCP Speller* o *c-VEP Speller*, que se emplean para que una persona con una discapacidad motora severa pueda comunicarse gracias a las señales emitidas por el cerebro al fijar la vista en unas imágenes (Santamaría, et al., 2023). La primera se basa en la detección de potenciales evocados P300 que se generan cuando el usuario fija su atención en un carácter deseado dentro de una matriz de caracteres que se van iluminando secuencialmente (Santamaría, et al., 2023). La respuesta puede ser detectada y procesada mediante algoritmos de DL integrados en MEDUSA[®]. Otras aplicaciones se emplean para desarrollar plasticidad neuronal, como *Motor imagery*, que se basa en la imaginación de movimientos sin necesidad de ejecutarlos físicamente (Santamaría, et al., 2023). Cuando una persona imagina que está moviendo una parte de

su cuerpo se activan patrones específicos en la corteza motora que pueden ser detectados con algoritmos de IA en señales EEG o MEG. De esta manera es posible interpretar la intención del usuario y traducirla en comandos, por ejemplo, para controlar un cursor o una prótesis (Ono, et al., 2014). También existen aplicaciones para informar al usuario de su estado neuronal concreto, como *Neurofeedback*. Esta permite entrenar a los usuarios en el control consciente de su propia actividad cerebral en tiempo real (Santamaría, et al., 2023). La información obtenida a partir de las señales neuronales adquiridas se devuelve a los usuarios en forma de estímulos visuales o auditivos, permitiéndoles ajustar su actividad neuronal voluntariamente. En la Figura 11 se muestra un esquema general de la herramienta MEDUSA[©] con los distintos componentes que se han mencionado.

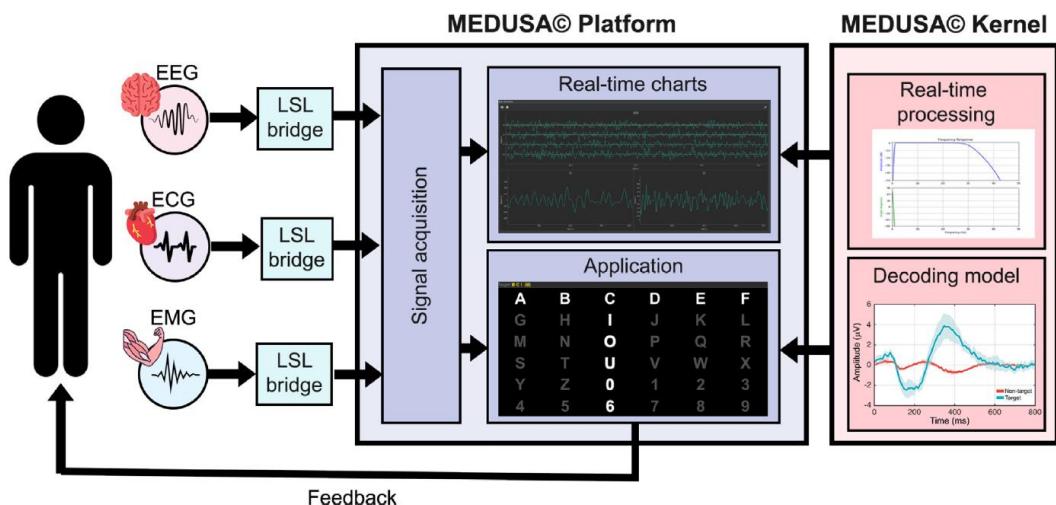


Figura 11. Vista general de MEDUSA[©]. Se muestran tres tipos de registros, de EEG, ECG y EMG, cuyas señales adquiridas se introducen en la herramienta MEDUSA[©] mediante el protocolo LSL. Las señales pueden verse en las gráficas de tiempo real y pueden ser empleadas en las distintas aplicaciones dentro de la plataforma. En el ejemplo mostrado puede verse la aplicación RCP Speller. También pueden aplicarse distintos procesamientos a las señales dentro del núcleo de MEDUSA[©]. En este caso se muestra la detección de potenciales ERP en señales procedentes de registros EEG (Santamaría, et al., 2023).

Capítulo 4. Materiales

Índice

4.1.	Introducción	55
4.2.	Base de datos	55
4.3.	Conjunto de señales	55
4.3.1.	Señales reales	56
4.3.2.	Señales sintéticas	58

4.1. Introducción

Este capítulo se centra en detallar los materiales empleados para la ejecución de este TFM, en concreto la base de datos con la que se ha contado. Se comentan las características de las señales reales utilizadas para el desarrollo de la herramienta, así como de las señales sintéticas generadas.

4.2. Base de datos

En este TFM se ha hecho uso de señales procedentes de registros MEG, correspondientes a 473 individuos que autorizaron la grabación de sus datos. Estos fueron tomados en el Hospital Hokuto, en la ciudad de Obihiro (Japón). Durante la adquisición de los datos los pacientes se encontraban tumbados y con los ojos cerrados pero despiertos, evitando cualquier movimiento que pudiera distorsionar la medida. Con el objetivo de garantizar la seguridad de los participantes y minimizar la aparición de somnolencia, los registros MEG fueron realizados bajo supervisión.

Se emplearon gradiómetros axiales para realizar las mediciones, y se hizo uso del sistema MEG Vision PQ1160C (Yokogawa Electric) con 160 canales. La frecuencia de muestreo utilizada fue de 1000 Hz y se aplicó un filtro paso-bajo con una frecuencia de corte de 200 Hz. El tiempo de registro por cada individuo tuvo una duración de cinco minutos. Con el fin de delimitar la posición de la cabeza de cada participante se utilizaron tres marcadores colocados de forma no arbitraria: uno a 5 mm por encima del nasión y los otros dos situados a 10 mm por delante del trago, a ambos lados de la cabeza. A continuación, las señales adquiridas fueron sometidas a un filtrado paso-banda entre 1 y 70 Hz mediante un filtro FIR (*Finite Impulse Response*) de orden 3000, lo que permitió restringir el espectro de frecuencias y reducir la presencia de ruido no deseado. Adicionalmente, se aplicó un filtro de ranura centrado en 50 Hz con el fin de eliminar la interferencia asociada al suministro eléctrico (Rodríguez-González, et al., 2021).

4.3. Conjunto de señales

Una vez descrita la base de datos empleada en este trabajo se debe analizar el conjunto de señales utilizadas para el entrenamiento, validación y evaluación del sistema desarrollado. Este apartado se divide en dos bloques principales: por un lado, las señales reales, obtenidas a partir de registros MEG adquiridos en condiciones controladas; y por

otro, las señales sintéticas, generadas artificialmente con el objetivo de disponer de un conjunto adicional de datos con características de ruido conocidas.

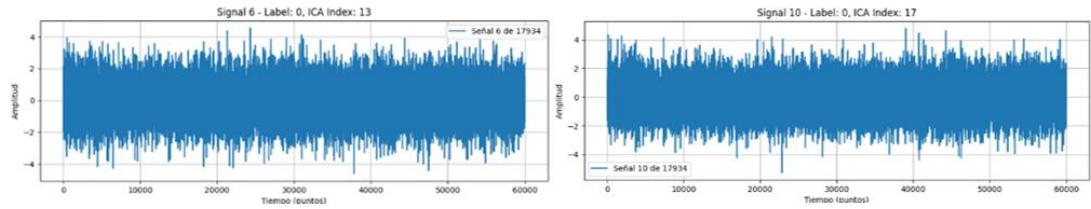
4.3.1. Señales reales

Tal y como se ha comentado previamente, una vez obtenidas las señales MEG procedentes de 160 canales por cada participante se debe realizar un preprocesado de las mismas. Para el caso bajo estudio, se hizo uso del algoritmo *Extended Infomax ICA* con el fin de obtener 160 componentes ICA por señal, una por cada canal. Este algoritmo se encarga de descomponer las señales adquiridas (mezclas de actividad neuronal y ruido) en un conjunto de componentes estadísticamente independientes. Cada componente puede interpretarse como una fuente que contribuye a la señal original (Lee, et al., 1999). Al aplicar esta técnica, es posible identificar y separar la actividad cerebral genuina de otros patrones no neuronales, como artefactos cardíacos, oculares o ruido ambiental. Además, ICA proporciona una matriz que determina cómo las fuentes independientes se combinan linealmente para dar lugar a las señales registradas en los sensores. Esta se conoce como la matriz de recomposición, y se emplea cuando se desea recomponer la señal limpia (por ejemplo, tras eliminar componentes artefactuadas), multiplicando la matriz de mezcla por las componentes que se quieren conservar (Hyvärinen, 2011). Así, se obtiene una versión filtrada de la señal original, donde la mayor parte del ruido ha sido eliminado, pero se preserva la actividad neuronal de interés.

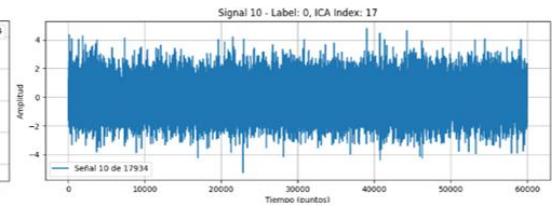
Tras haber aplicado ICA, varios técnicos expertos se encargaron de etiquetar cada una de las componentes generadas en varios grupos en función de si correspondían a actividad neuronal o a diferentes tipos de artefactos. El etiquetado asignado fue el siguiente: se empleó el valor 0 para las componentes que reflejaban actividad cerebral, el valor 1 para aquellas asociadas a señales cardíacas, el 2 para las interferencias procedentes de la red eléctrica, el 3 para las relacionadas con movimientos oculares y el valor 4 para otros tipos de artefactos.

Finalmente, se llevó a cabo un proceso de estandarización antes de introducir las señales en el algoritmo de detección y eliminación de artefactos diseñado. Este proceso consistió en normalizar cada componente, restando su media y dividiendo el resultado por su desviación estándar. Con esto se consigue que todas las señales tengan una media de cero y una varianza unitaria, lo que reduce diferencias numéricas entre componentes que podrían deberse únicamente a escalas distintas. Al normalizar las señales, se facilita que la red neuronal aprenda patrones relevantes en lugar de verse influida por diferencias arbitrarias en la magnitud de las señales, reduciendo así posibles sesgos durante la clasificación (Rodríguez-González, et al., 2021).

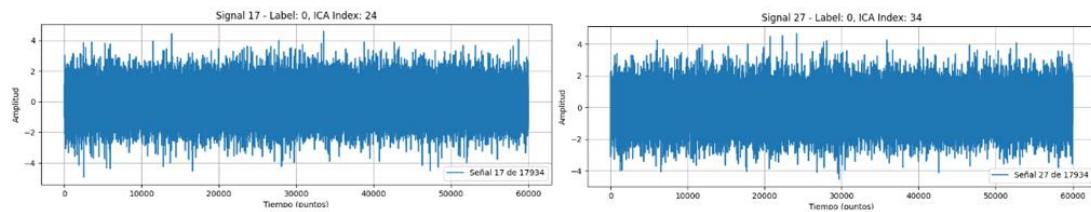
Las Figuras 12, 13, 14 y 15 muestran varios fragmentos de componentes neuronales, cardíacas, de la red eléctrica y oculares, respectivamente.



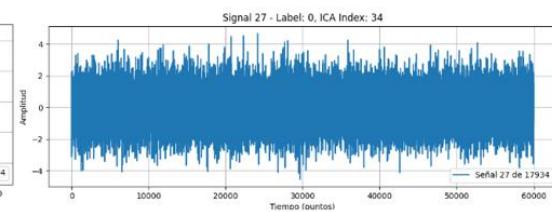
(a) Componente ICA número 13.



(b) Componente ICA número 17.

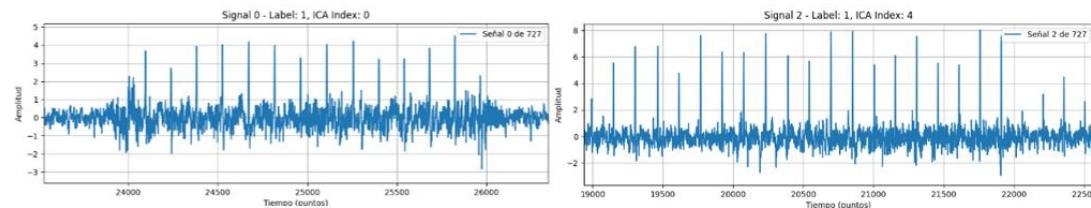


(c) Componente ICA número 24.

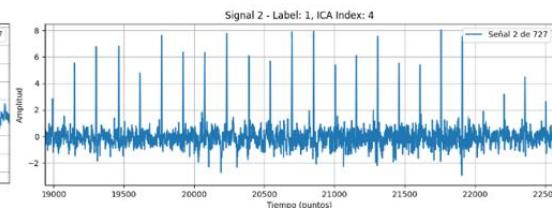


(d) Componente ICA número 34.

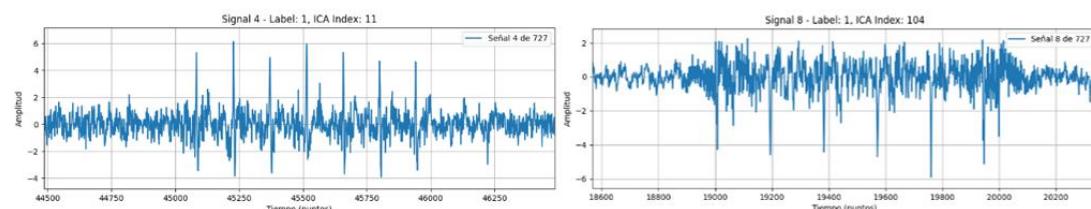
Figura 12. Cuatro fragmentos de señales etiquetadas como cerebrales ("Label: 0").



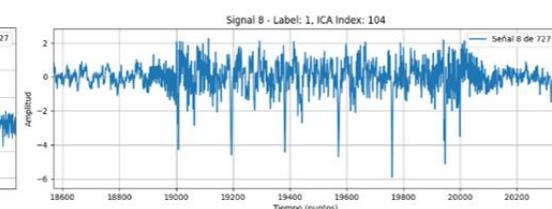
(a) Componente ICA número 0.



(b) Componente ICA número 4.



(c) Componente ICA número 11.



(d) Componente ICA número 104.

Figura 13. Cuatro fragmentos de señales etiquetadas como cardíacas ("Label: 1").

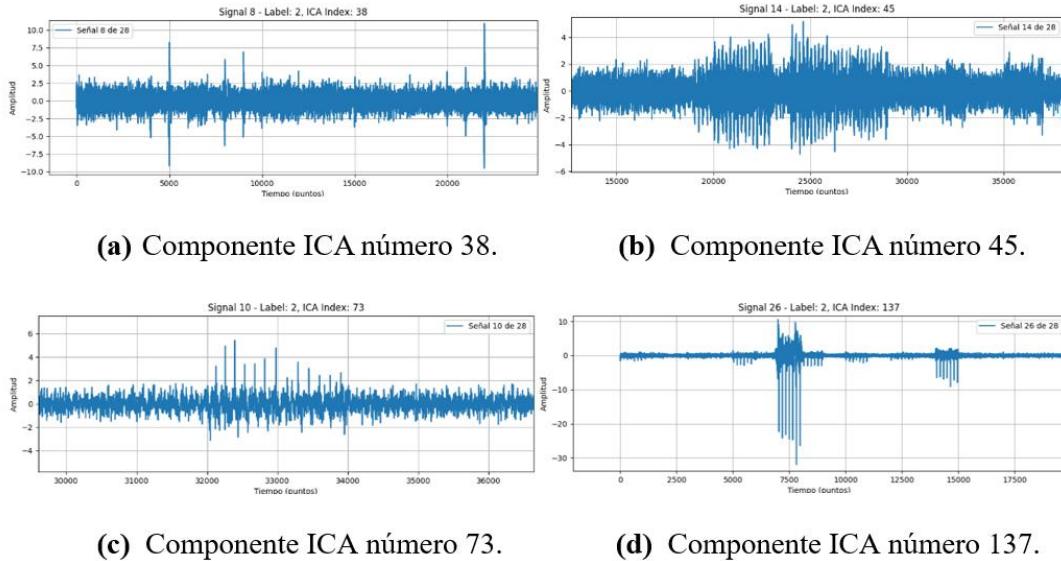


Figura 14. Cuatro fragmentos de señales etiquetadas como red eléctrica ("Label: 2").

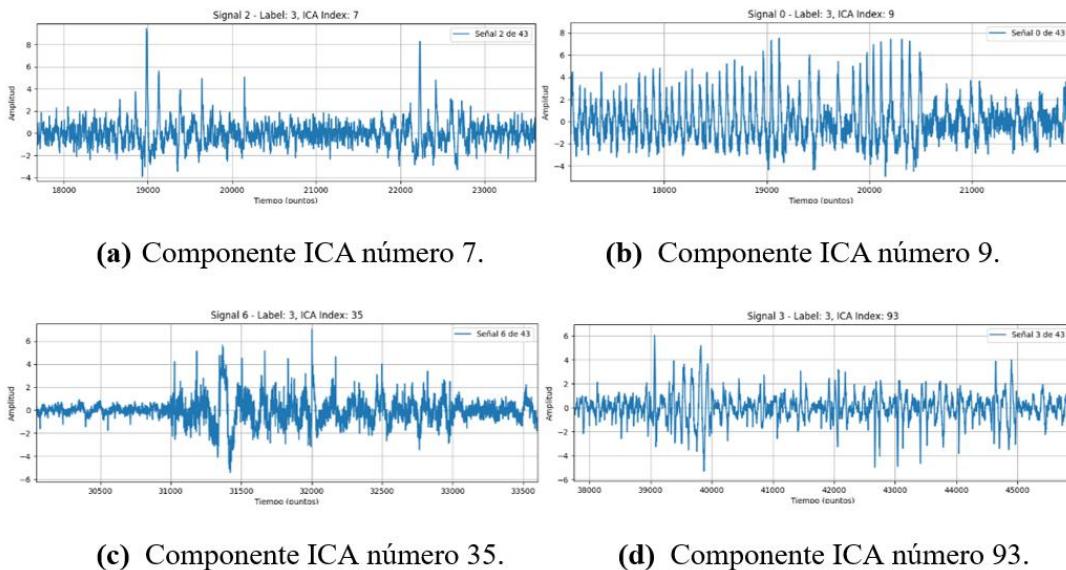


Figura 15. Cuatro fragmentos de señales etiquetadas como oculares ("Label: 3").

4.3.2. Señales sintéticas

Para complementar el conjunto de datos reales y disponer de señales con un *ground truth* controlado, se han generado señales sintéticas mediante la combinación de actividad neuronal y segmentos de ruido representativos de los artefactos disponibles. Este procedimiento permite crear ejemplos donde se conoce exactamente qué parte de la señal corresponde a actividad cerebral genuina y cuál ha sido añadida artificialmente como

contaminación, lo que resulta especialmente útil para evaluar el comportamiento de la herramienta de IA diseñada. El proceso de creación de estas señales sintéticas se llevó a cabo siguiendo los pasos que se describen a continuación:

- 1. Selección del artefacto.** En primer lugar, se identificaron manualmente segmentos de componentes ICA que representaban artefactos de interés, como actividad cardíaca, de red eléctrica, ocular o de otro tipo. A modo ilustrativo, para generar una señal sintética con artefacto cardíaco se seleccionó un tramo comprendido entre las muestras 22.000 y 24.000 de la componente ICA número 11, correspondiente a una clara manifestación de este tipo de ruido. Una vez identificado, el fragmento fue replicado de manera consecutiva hasta alcanzar una duración total de 5 minutos (equivalente a la duración de las señales reales del conjunto de datos). Esto puede verse en la Figura 16.
- 2. Aplicación de una máscara.** A la señal artefactuada replicada se le aplicó una máscara temporal diseñada específicamente para controlar la distribución del ruido a lo largo del tiempo. Esta máscara es una señal binaria de 5 minutos de duración que determina en qué momentos se introduce el artefacto en la señal sintética final:
 - Durante el primer minuto, la máscara toma el valor 0 en todos los puntos, por lo que no se añade artefacto.
 - En el segundo minuto, la máscara toma el valor 1 en intervalos de 2 segundos de duración, separados periódicamente.

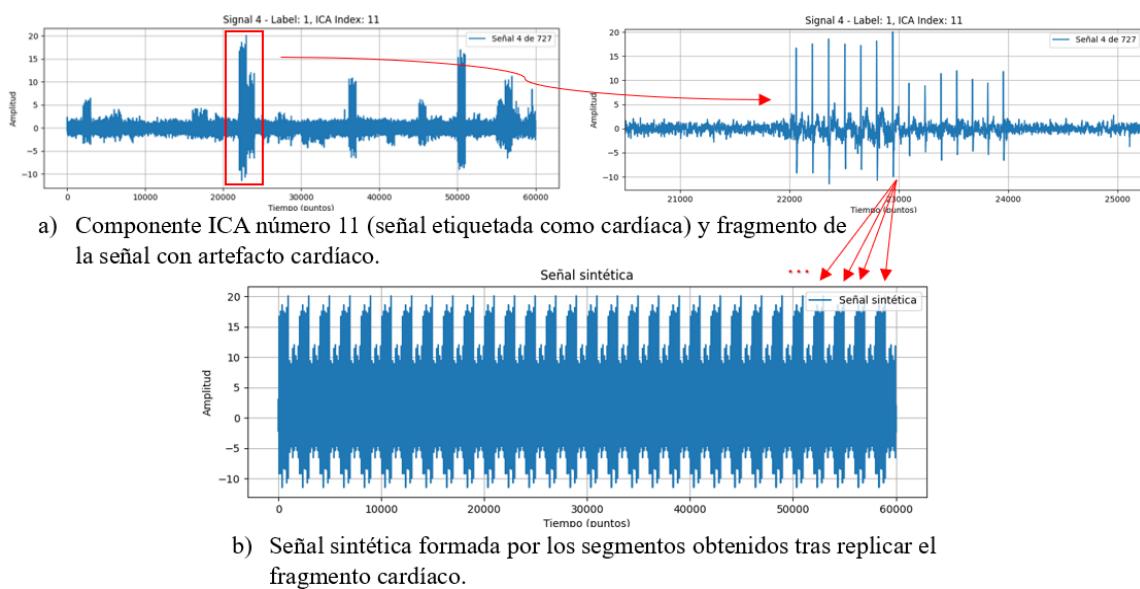


Figura 16. Selección y replicación del artefacto para la generación de señales sintéticas.

- Durante el tercer minuto, el patrón se repite con menor frecuencia: la máscara toma el valor 1 cada 5 segundos.
- A partir del cuarto minuto y hasta el final, la máscara toma el valor constante de 1, introduciendo de forma continua el artefacto.

La máscara puede verse representada en la Figura 17, y la aplicación de esta sobre la señal artefacto obtenida previamente se muestra en la Figura 18.

3. **Ajuste del artefacto.** El segmento aislado fue posteriormente modificado en su escala mediante un ajuste de potencia, lo que permitió simular diferentes intensidades de contaminación. Este paso permite ajustar el nivel de interferencia que se desea inyectar sobre la señal limpia al variar la potencia del artefacto respecto a la señal neuronal original. Para ello, se ha calculado la potencia de la señal neuronal y de la señal artefacto con la máscara. A continuación, se ha multiplicado la potencia de la señal neuronal obtenida por el porcentaje deseado y se ha dividido entre la potencia de la señal artefacto con la máscara. El valor obtenido es el que se ha multiplicado a la señal con la máscara para ajustar su potencia respecto a la de la señal neuronal. En la Figura 19 se puede ver la señal creada con diferentes ajustes de potencia.

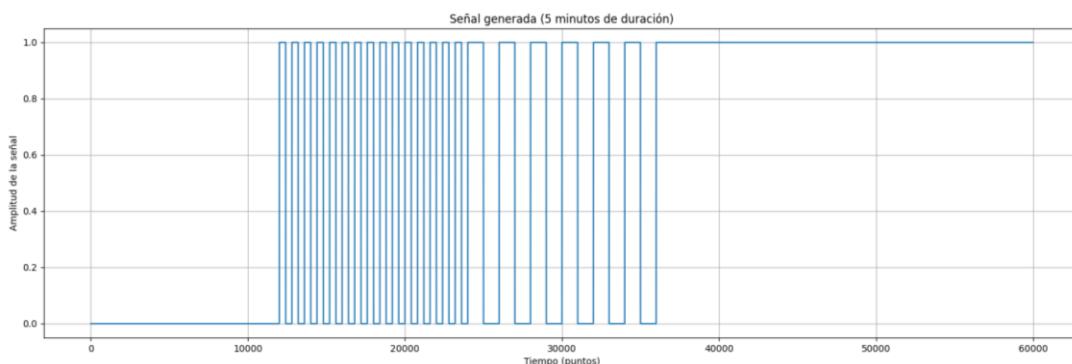


Figura 17. Máscara binaria de 5 minutos de duración.

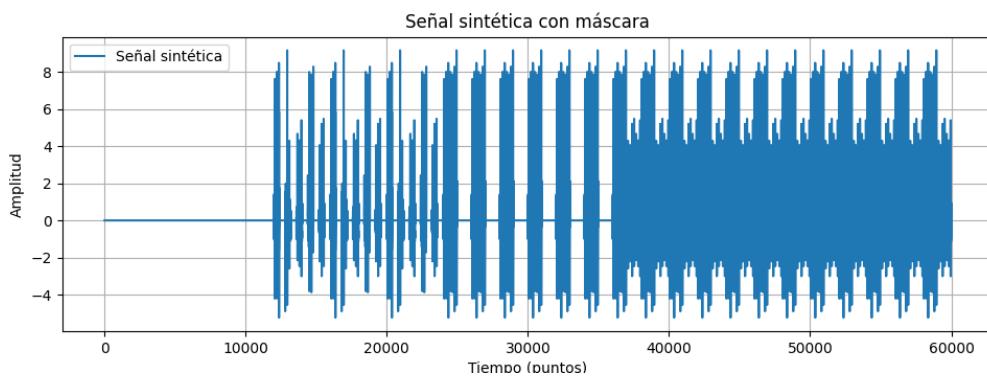


Figura 18. Señal artefacto obtenida tras aplicar la máscara a la señal replicada.

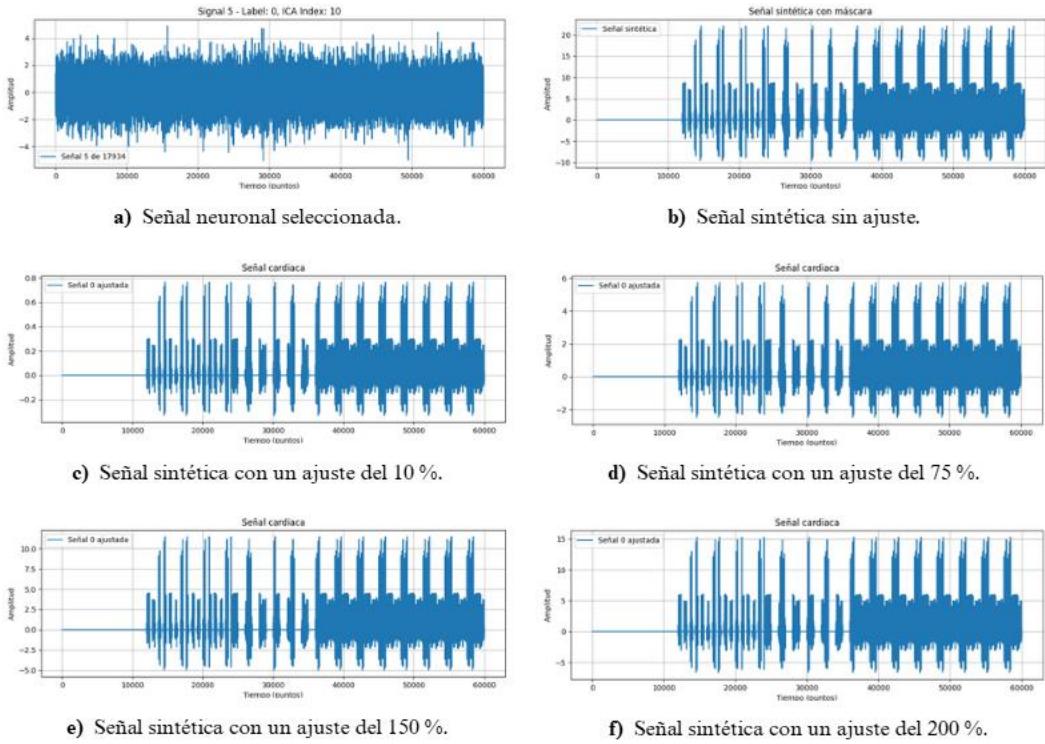


Figura 19. Señal artefacto con diferentes ajustes de potencia respecto de la señal neuronal seleccionada.

4. Composición con la señal neuronal. Finalmente, el artefacto procesado fue sumado a una señal neuronal limpia seleccionada previamente, generando así una nueva señal sintética con artefacto inyectado en condiciones conocidas. Este procedimiento puede repetirse con diferentes combinaciones de señales y tipos de artefacto. En la Figura 20 se puede ver la señal neuronal empleada en el ejemplo elaborado, y en la Figura 21 se muestra el resultado final con los distintos ajustes que se han aplicado previamente. Por último, en la Figura 22 pueden verse varias señales sintéticas generadas a partir de diferentes artefactos y señales neuronales seleccionadas.

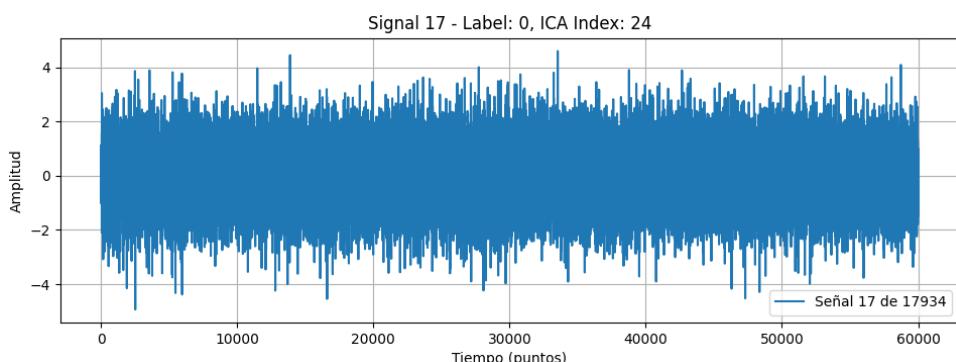


Figura 20. Componente ICA número 24 etiquetada como neuronal empleada en la creación de una señal sintética con artefacto cardíaco.

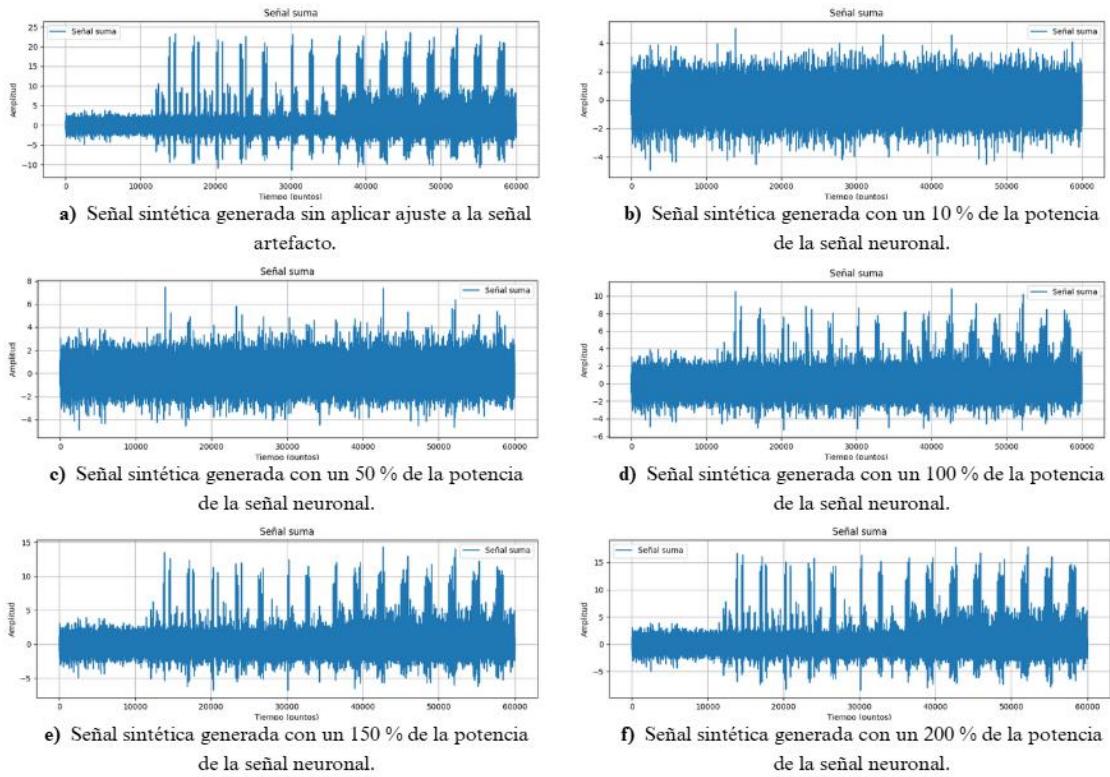


Figura 21. Señales sintéticas generadas al añadir un artefacto con distintas proporciones de potencia respecto a la señal neuronal original.

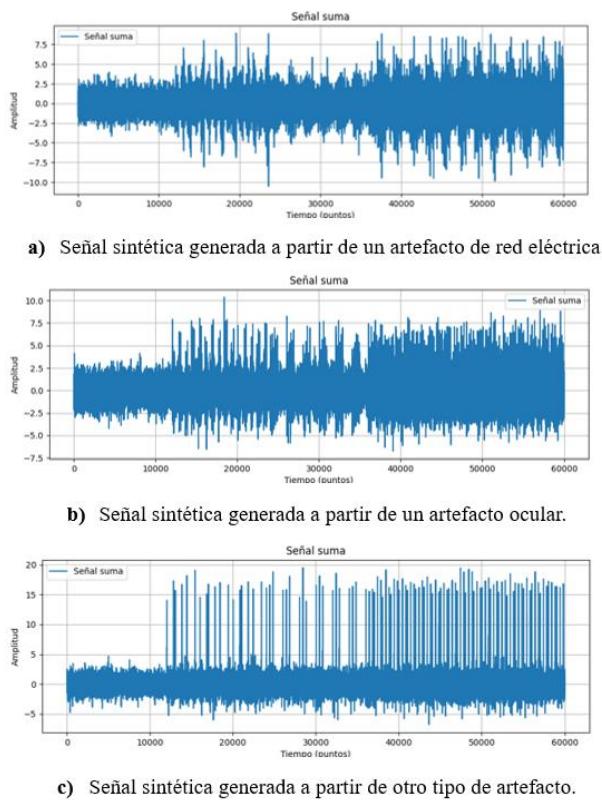


Figura 22. Señales sintéticas generadas tras haber empleado diferentes artefactos y señales neuronales.

Capítulo 5. Desarrollo de la herramienta de detección y eliminación automática de artefactos en señales neuronales e integración en la herramienta MEDUSA[©]

Índice

5.1. Introducción	64
5.2. Creación de un sistema automático de detección y eliminación de artefactos basado en técnicas de IA.....	64
5.2.1. Etapa 1: Detección de artefactos.....	64
5.2.2. Etapa 2: Eliminación de artefactos	67
5.3. Integración del detector automático en la herramienta MEDUSA[©].....	72

5.1. Introducción

En el presente capítulo se introduce la herramienta basada en DL empleada, junto con las mejoras incorporadas para visualizar y analizar las componentes cerebrales afectadas por artefactos, y cómo se ha llevado a cabo su integración en la herramienta MEDUSA[©].

5.2. Creación de un sistema automático de detección y eliminación de artefactos basado en técnicas de IA

Para la elaboración del detector automático se ha partido de la herramienta realizada en el TFG del Grado en Tecnologías de Telecomunicación de la Universidad de Valladolid, un modelo de DL basado en la CNN EEG-Inception (Gil Correa, 2023). Esta CNN se diseñó específicamente para procesar señales EEG en interfaces cerebro-computador basadas en potenciales evocados (ERP, *Event-Related Potentials*). Presenta una arquitectura modular donde se analizan las señales a múltiples escalas temporales de forma simultánea y se extraen las características más relevantes a distintas resoluciones temporales (Santamaría, et al., 2020).

5.2.1. Etapa 1: Detección de artefactos

Tal y como se realizó en el TFG, en este TFM la red se modificó respecto a la CCN EEG-Inception original con el fin de adaptarla a los datos de los que se disponía, ya que la arquitectura inicial está pensada para el tratamiento de señales EEG. El modelo final puede verse en la Figura 23. Este cuenta con dos entradas: las características o *features* extraídas de la base de datos y una matriz que asigna un canal del registro a cada componente. A la primera entrada se le aplican dos convoluciones de tamaño 25x1 que, a una frecuencia de muestreo de 200 Hz, permiten obtener dos ventanas temporales de 125 milisegundos. Las salidas se concatenan y se reducen las dimensiones tras aplicar un *Average Pooling*. Esta capa calcula el valor medio de los valores de una ventana de su matriz de entrada, y la salida que ofrece es una nueva matriz formada por dichos valores medios calculados (Ertam & Aydin, 2017). A continuación, se vuelven a realizar dos convoluciones de tamaño 6x1 donde las características extraídas presentan un nivel de abstracción mayor. De nuevo, las salidas vuelven a concatenarse y se aplica otro *Average Pooling*. Tras esto, se realiza una convolución de 8x1 y otra de 4x1, junto con sus *Average Pooling*. Por último, se aplica un *Global Average Pooling*, que devuelve la media de los valores de una ventana de la matriz sobre la que se aplica, y se concatena con el *Global Max Pooling* efectuado sobre la segunda entrada de la red, que devuelve el valor máximo

de la misma. La salida final se obtiene tras aplicar la capa *Dense*. Cada convolución, tal y como se detalla en la parte inferior derecha de la Figura 23, está formada por una normalización *Batch Normalization*, una activación y el *Dropout*. Cabe destacar que en el TFG se implementó la arquitectura EEG-Inception utilizando la librería TensorFlow de Python, mientras que en el TFM se ha optado por una implementación en PyTorch. Esto se ha debido a que las versiones más modernas de MEDUSA[©] han dejado de dar soporte a TensorFlow, considerándolo obsoleto dentro de su ecosistema.

En cuanto al conjunto de datos empleado en el modelo diseñado, este se ha dividido en dos, uno para entrenar la red y otro para testearla. Para ello, y tal y como se realizó también en el TFG, se ha hecho uso de los métodos *k-fold* y *earlystopping*. El primero toma el conjunto de datos disponible y lo divide en *k* partes iguales. A continuación, entrena el modelo *k* veces, y emplea *k* – 1 partes para entrenar y la parte restante para validar. Al final, calcula el promedio de las métricas obtenidas en cada una de las *k* validaciones (Colomer, et al., 2016). En este caso se ha tomado un valor de *k* igual a 5 para conseguir un buen equilibrio entre coste computacional y variedad en la evaluación. Además, se reduce el riesgo de que el rendimiento del modelo esté sesgado a una sola partición de datos. Por otro lado, *earlystopping* detiene el entrenamiento automáticamente cuando el modelo deja de mejorar en el conjunto de validación, evitando que se produzca sobreajuste u *overfitting*. Para ello, se debe indicar el valor del parámetro *patience*, que define cuántas épocas consecutivas se permiten sin mejora antes de detener el entrenamiento (Hussein & Shareef, 2024). En este caso se ha escogido un valor de 10 para evitar parar el entrenamiento demasiado pronto y tolerar posibles oscilaciones durante el ajuste.

Para la etapa de entrenamiento se deben escoger los hiperparámetros que permitan una buena capacidad de generalización. En primer lugar, se ha escogido el número de épocas de la red, que se corresponde con el número de veces que el modelo recorre completamente el conjunto de datos de entrenamiento durante el proceso de aprendizaje.

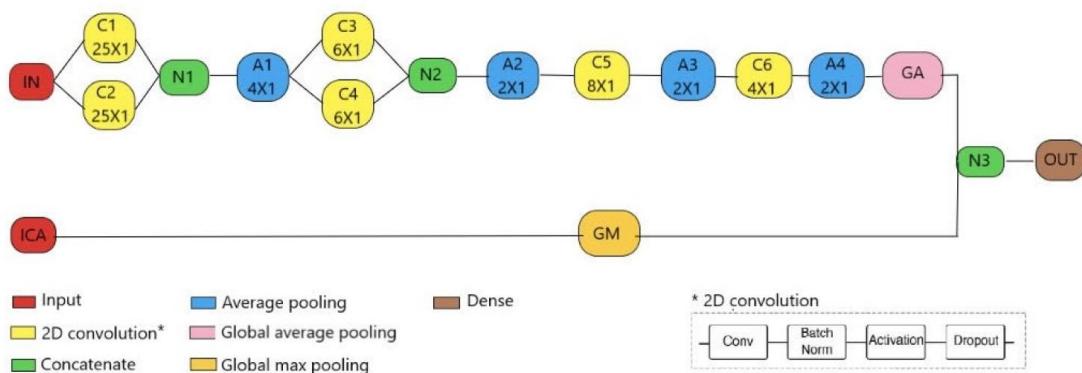


Figura 23. Red EEG-Inception adaptada al detector automático de artefactos de este TFM.

Este parámetro influye directamente en la capacidad del modelo para ajustar sus pesos, ya que un número bajo de épocas puede provocar que el modelo no aprenda lo suficiente, mientras que un número excesivo puede llevar al sobreajuste. En este trabajo se ha decidido establecer el número de épocas en 100, un valor que, combinado con *early stopping*, permite al modelo aprender de forma adecuada sin prolongar innecesariamente el proceso de entrenamiento ni comprometer su rendimiento. Por otro lado, se debe escoger la función de pérdidas, que cuantifica la diferencia entre las predicciones del modelo y las etiquetas reales. Cuanto mayor sea la diferencia, mayor será la corrección que se aplica a los pesos de la red en la siguiente iteración (O'Shea & Nash, 2015). En este TFM se ha utilizado la función *Categorical Crossentropy*, útil tanto para clasificación binaria como multiclase, ya que mide la distancia entre la distribución de probabilidades predicha por la red y la distribución real de las clases, penalizando mayoritariamente aquellas predicciones alejadas de la clase correcta. También debe escogerse el *batch size*, que se corresponde con el número de muestras que se procesan simultáneamente antes de actualizar los pesos del modelo durante el entrenamiento. En lugar de ajustar los pesos tras cada muestra individual o después de todo el conjunto de entrenamiento se hace en pequeños bloques denominados *batches*. En este caso, la elección del *batch size* ha sido condicionada por el tipo de entrada seleccionada: si se hace uso de la entrada correspondiente a la matriz que codifica la posición del canal dentro del registro el *batch size* toma un valor de 4, mientras que si no se usa dicha entrada su valor es de 128. Esto se debe a que, al incluir la matriz, el modelo requiere más memoria para procesar cada lote por lo que un valor mayor sobrecargaría los recursos computacionales disponibles.

Además, para evaluar la herramienta de clasificación se ha hecho uso de diferentes métricas que permiten valorar su rendimiento (Khanna, 2025):

- La sensibilidad mide la proporción de verdaderos positivos detectados correctamente, es decir, la capacidad del modelo para identificar correctamente los casos positivos.
- La especificidad evalúa la proporción de verdaderos negativos correctamente clasificados, lo que indica cómo de bien el modelo evita falsos positivos.
- La precisión representa el porcentaje de predicciones que realmente son correctas, reflejando la fiabilidad del modelo al etiquetar un caso como positivo o negativo.
- La matriz de confusión ofrece una representación que compara las clases reales con las predichas, permitiendo observar detalladamente los aciertos y errores de clasificación. En las filas de la matriz se representa el etiquetado real, mientras que en las columnas se muestra el etiquetado predicho por la red.

- El coeficiente *kappa* cuantifica el grado de acuerdo entre las predicciones del modelo y las etiquetas reales; toma valores entre -1 y 1, donde un valor cercano a 1 indica un alto nivel de concordancia.

En este trabajo se han seleccionado las 40 primeras componentes ICA de cada sujeto para entrenar la red, ya que, debido al funcionamiento del algoritmo ICA, las primeras componentes son las que mayor amplitud presentan y, por tanto, las que más contribuyen a la reconstrucción de la señal original. Sin embargo, para equilibrar la representación de las clases, especialmente la de los artefactos que aparece con menor frecuencia en todo el conjunto de datos, se han añadido además aquellas componentes etiquetadas como artefacto dentro del rango de las componentes 41 a 160, es decir, se han añadido todas las componentes etiquetadas como artefacto del conjunto de datos. Esta decisión se justifica por el hecho de que la base de datos original presenta un desbalanceo de clases, con una mayoría de componentes cerebrales respecto a las ruidosas, lo que podría afectar negativamente al rendimiento del modelo si no se lleva a cabo un ajuste. Además, el entrenamiento se realiza en múltiples iteraciones donde en la primera se entrena con todas las componentes seleccionadas, mientras que en el resto se van eliminando las componentes clasificadas por la red como “dudosas”. El criterio seguido para clasificar una señal como “dudosa” es el siguiente: se identifican aquellas componentes que han sido etiquetadas como no artefacto (es decir, con clase real igual a 0), pero que el modelo clasifica con una probabilidad inferior a 0.9 de pertenecer a dicha clase. Estas componentes, al no ser clasificadas como limpias por la red, se consideran potencialmente contaminadas o ambiguas. Por tanto, en la siguiente iteración de entrenamiento, se eliminan del conjunto de datos. El bucle se detiene cuando se cumple alguna de las siguientes condiciones: en primer lugar, si la sensibilidad acumulada supera el 95 %, lo que indica que el modelo ha alcanzado un rendimiento suficientemente alto en la detección de componentes ruidosas; y, en segundo lugar, si se detecta un empeoramiento de la sensibilidad durante dos iteraciones consecutivas, lo que sugiere que continuar eliminando componentes podría perjudicar la capacidad del modelo para generalizar correctamente. Cabe destacar que todo el sistema de detección y clasificación de artefactos en señales neuronales ha sido diseñado para funcionar de forma modular, permitiendo su uso tanto en señales reales como sintéticas.

5.2.2. Etapa 2: Eliminación de artefactos

Una vez completado el proceso de clasificación iterativa, el siguiente paso, y una de las principales aportaciones del presente TFM, consiste en analizar por qué el modelo clasifica ciertas componentes como dudosas. Para ello, se ha desarrollado un sistema que resalta visualmente las regiones que la red considera ruidosas, proporcionando al usuario una primera explicación de los motivos por los que una componente, inicialmente

etiquetada como limpia, podría estar contaminada. Además, el objetivo principal de este TFM es disponer de una herramienta que permita limpiar una base de datos de señales neuronales contaminadas con artefactos. La primera opción que cabría plantearse para limpiar la base de datos de la que se dispone es eliminar por completo las señales “dudosas”. Sin embargo, la pérdida de información relevante en dichas señales conllevaría graves problemas de identificación de posibles enfermedades o condiciones neuronales en los pacientes bajo estudio. La opción que se ha planteado y estudiado en este trabajo consiste en detectar el tramo específico de señal neuronal contaminado con ruido procedente de otras fuentes que no son neuronales y disminuir su efecto o eliminarlo por completo sin que afecte a las partes que son exclusivamente neuronales. La reconstrucción posterior de la señal cerebral estará limpia de artefactos, que son ajenos a las señales generadas en el cerebro. La idea de la que se parte es la siguiente: aprovechando la herramienta de IA diseñada explicada en el anterior subapartado, es posible calcular el *score* de probabilidad de artefacto de cada uno de los fragmentos que componen cada señal; esta probabilidad permite aplicar un proceso de limpieza automática, eliminando únicamente aquellos tramos cuya probabilidad de ser ruido supera un determinado umbral. Este enfoque permite conservar la mayor parte de la información neuronal útil, evitando la necesidad de descartar componentes completas.

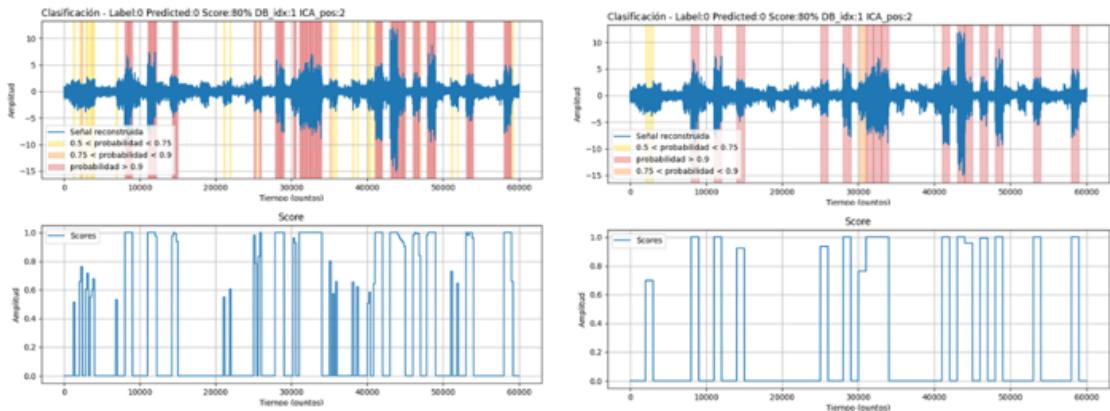
Con el fin de detectar esas partes ruidosas se han dividido las señales en segmentos. La duración de dichos segmentos queda a elección del usuario, obteniendo una clasificación más precisa pero más lenta cuanto menor es esta. Posteriormente, cada segmento se replica hasta volver a obtener una señal de 5 minutos de duración y poder introducir las nuevas señales generadas (réplicas de segmentos) por el algoritmo de DL. Se ha probado a generar segmentos de 1, 5, 10 y 20 segundos de duración y, sabiendo que la frecuencia de muestreo empleada es de 200 Hz, se tiene que:

- Para segmentar la señal en épocas de 1 segundo, la señal tendrá 200 muestras. Como la señal completa tiene una duración de 5 minutos, o 300 segundos (lo que equivale a 60000 muestras), habrá un total de 300 épocas de 1 segundo de duración, lo que equivale a la introducción de 300 señales en el algoritmo de IA.
- Para segmentar la señal en trozos de 5 segundos, la señal tendrá 1000 muestras. Esto resulta en 60 épocas de 5 segundos de duración, o 60 señales generadas.
- Para segmentar la señal en épocas de 10 segundos, la señal tendrá 2000 muestras, con lo que se obtienen 30 épocas de 10 segundos de duración (30 señales).
- Para segmentar la señal en épocas de 20 segundos, la señal tendrá 4000 muestras, y habrá 15 épocas de 20 segundos de duración (15 señales).

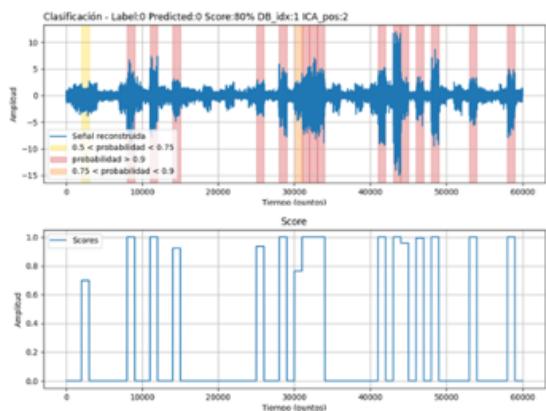
En las gráficas de clasificación, el parámetro “*Label*” corresponde al etiquetado realizado previamente por un experto, antes de cargar las señales MEG en la herramienta de clasificación. Por otro lado, el parámetro “*Predicted*” hace referencia a la predicción hecha por el algoritmo de clasificación. Finalmente, el parámetro “*Score*” se corresponde con el valor numérico que especifica la probabilidad con la que el algoritmo obtiene la predicción realizada. Así, en la tarea previa de detección de artefactos se clasificaron como “dudosas” aquellas componentes etiquetadas como no artefacto; es decir, su parámetro “*Label*” era igual a cero, cuyo *score* para dicha categoría era menor que 0.9, o lo que es lo mismo, su probabilidad estaba por debajo del 90 % para la categoría de no artefacto. Con el objetivo de proporcionar los resultados obtenidos de forma visual, la clasificación indica qué partes de la señal ha considerado ruidosas, y el *score* correspondiente para la clase artefacto: en rojo marca aquellos segmentos cuyo *score* supera el 90 %, en naranja destaca aquellos cuyo *score* se encuentra entre el 75 y el 90 %, y en amarillo aquellos cuyo *score* está entre el 50 y el 75 %. Además, se ha obtenido una gráfica con los *scores* obtenidos a lo largo de la señal completa. Dicha gráfica adopta una forma de pulsos rectangulares, cuyo ancho es acorde a la duración de los segmentos generados. En la Figura 24 se muestra la misma señal en la que se han aplicado diferentes niveles de segmentación, junto con la clasificación obtenida y el *score* correspondiente.

Para tratar de eliminar el ruido detectado en la señal original se ha decidido multiplicar esta última por el complemento del *score* obtenido, es decir, el 100 % menos el *score*, dado que el objetivo es eliminar o amortiguar la influencia de las partes ruidosas, y el *score* es mayor cuanto más ruidoso considere el algoritmo el segmento correspondiente. Sin embargo, multiplicar directamente por una forma de onda rectangular implicaría transiciones muy abruptas en la señal resultante en las zonas de cambio de amplitudes. Por eso, se ha decidido aplicar un suavizado a la gráfica del *score*. Se han propuesto dos alternativas diferentes, a elección del usuario: (i) suavizado mediante la aplicación de un filtro gaussiano o (ii) mediante la aplicación de un filtro de media móvil. El primero aplica una suavización basada en una distribución normal, lo que implica que los valores centrales del *score* tienen más peso y los extremos menos, generando una transición suave entre las zonas ruidosas y las no ruidosas. Este tipo de suavizado es útil para evitar cortes bruscos en la señal y proporciona una delimitación progresiva de los artefactos, respetando mejor la continuidad temporal. En términos generales, la función gaussiana se expresa de la siguiente manera (Nixon & Aguado, 2019):

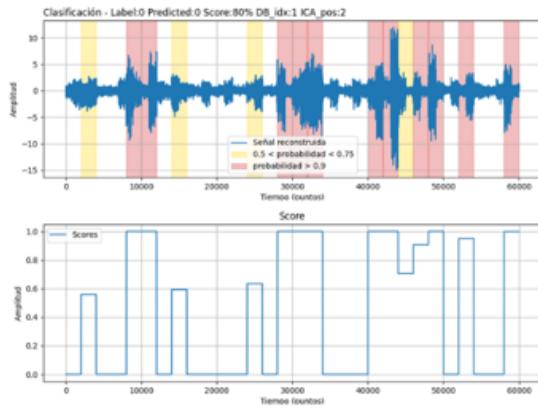
$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (5)$$



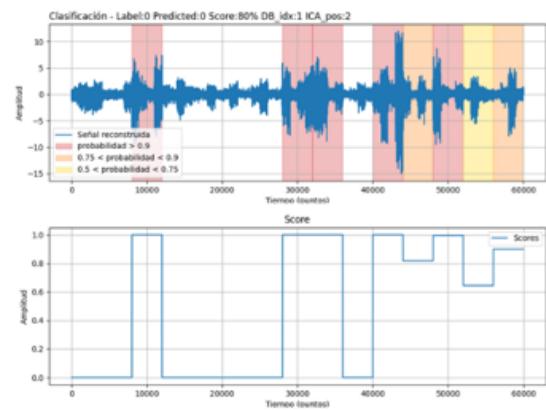
a) Clasificación y score de la señal tras segmentarla en segmentos de 1 segundo.



b) Clasificación y score de la señal tras segmentarla en segmentos de 5 segundos.



c) Clasificación y score de la señal tras segmentarla en segmentos de 10 segundos.



d) Clasificación y score de la señal tras segmentarla en segmentos de 20 segundos.

Figura 24. Señal neuronal con artefactos a la que se han aplicado las 4 segmentaciones posibles. Se muestran los artefactos marcados con colores tras la clasificación y los scores correspondientes: probabilidad de ser artefacto superior al 90 % en rojo, probabilidad de ser artefacto entre un 75 y un 90 % en naranja, y probabilidad de ser artefacto entre un 50 y un 75 % en amarillo.

El valor del parámetro *sigma*, σ , en un filtro gaussiano define el ancho de la campana de la función gaussiana, y afecta directamente a la cantidad de suavizado aplicado a los datos. Así, si *sigma* es pequeño la campana es más estrecha, lo que hace que los valores más próximos al valor central tengan un peso significativo. En este caso, el suavizado aplicado a los datos es menor, y permite conservar más los detalles de la señal. Por otro lado, para un valor mayor de *sigma* la campana es más ancha, haciendo que valores más alejados del centro tengan influencia en el suavizado de los datos. Este último es mayor en este caso, lo que provoca que se pierdan ciertos detalles de la señal. La función gaussiana viene caracterizada por la propiedad de decaimiento, que establece que la mayor parte de la energía se concentra en una zona alrededor del centro. Así, se tiene que:

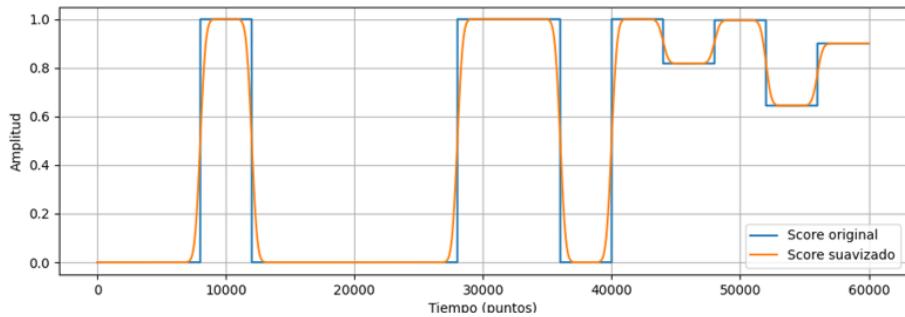
- 1σ : contiene aproximadamente el 68% del área.
- 2σ : contiene aproximadamente el 95% del área.
- 3σ : contiene más del 99% del área.

Por esta razón, en aplicaciones prácticas, se considera que un radio de 3σ desde el centro es suficiente para capturar casi toda la información relevante de la función gaussiana. Así, en este TFM se ha considerado que con un valor de sigma igual a 500 se aplica un suavizado donde cada punto del *score* no depende solo de su valor puntual, sino también de la influencia de los valores vecinos contenidos dentro de un rango de aproximadamente ± 1500 muestras. De este modo, el suavizado resultante es lo suficientemente amplio como para eliminar picos aislados en el *score*, generando una transición progresiva entre las zonas clasificadas como ruidosas y las que no lo son.

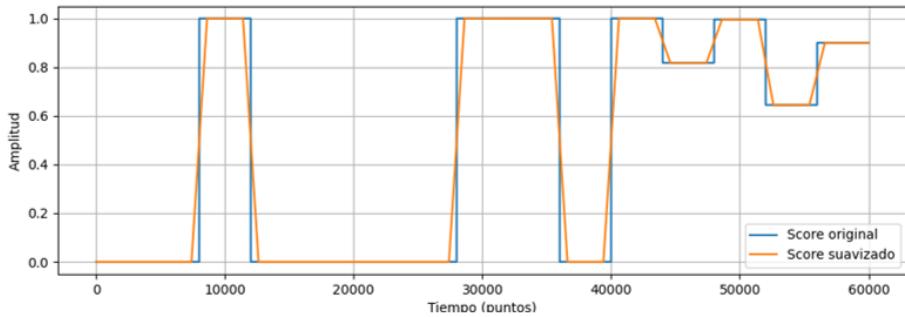
Por otro lado, el filtro de media móvil es muy similar al filtro gaussiano ya que ambos reemplazan cada punto por un promedio ponderado de sus valores vecinos. Sin embargo, la diferencia fundamental radica en la forma en que asignan los pesos a los puntos dentro de la ventana. En el filtro de media móvil, todos los puntos dentro de la ventana tienen el mismo peso, es decir, la contribución de cada uno es constante e independiente de su distancia al punto central. Esto puede provocar que los bordes de la ventana tengan tanta influencia como el propio centro, lo que puede dar lugar a transiciones algo más abruptas. La expresión que caracteriza al filtro de media móvil es la siguiente (Smith, 1999):

$$y[n] = \frac{1}{W} * \sum_{k=-D}^A x[n+k], \quad (6)$$

donde D es el número de muestras que coge por detrás de la muestra n , A es el número de muestras que coge por delante de la muestra n , y W , que es el tamaño de la ventana del filtro, es el sumatorio de A y D más 1. Esta ventana deslizante, que es de tamaño fijo, define el número de puntos que se tienen en cuenta en cada cálculo y, dentro de ella, todos los valores reciben el mismo peso. Es decir, la ventana funciona como una caja de tamaño constante que se desplaza a lo largo de la señal, y para cada posición el valor resultante es la media aritmética de los puntos contenidos en esa ventana. Al igual que sucedía con el filtro gaussiano, cuanto mayor es la ventana mayor es el suavizado, pero también se pierde más detalle y se pueden suavizar excesivamente las transiciones, mientras que si la ventana es pequeña el efecto de suavizado es más ligero, conservando más las fluctuaciones rápidas de la señal. Para que su efecto sea similar al causado por el filtro gaussiano se ha escogido un valor de ventana igual a 1201 muestras dado que permite alcanzar un compromiso adecuado entre el nivel de suavizado y la preservación de la estructura de la señal. En la Figura 25 puede verse el resultado de suavizar el *score* de una señal con los dos filtros mencionados.



a) Score suavizado con un filtro gaussiano.



b) Score suavizado con un filtro de media móvil.

Figura 25. Suavización del score de una señal neuronal con artefactos calculado por la red (en azul) mediante la aplicación de un filtro gaussiano y un filtro de media móvil. El score suavizado se muestra en naranja.

5.3. Integración del detector automático en la herramienta MEDUSA[©]

El código desarrollado se ha adaptado para integrarse dentro de MEDUSA[©]. Esta integración ha requerido reorganizar y modularizar dicho código, así como ajustar su estructura para hacerlo compatible con las herramientas y convenciones internas de MEDUSA[©]. Gracias a ello, se ha podido implementar una interfaz visual usando Python que permite al usuario cargar señales, aplicar la detección de artefactos y visualizar los tramos ruidosos resaltados con colores, facilitando así tanto la inspección manual como la limpieza automática asistida.

El flujo de trabajo empleado se resume en la Figura 26, donde se muestran las primeras etapas de adquisición de señales MEG, su paso por el algoritmo de ICA y el etiquetado de las componentes obtenidas por parte de los técnicos especialistas, hasta la aplicación del algoritmo de IA. Este último ha seguido una estructura modular en Python, organizada en bloques funcionales que reflejan cada una de las etapas del flujo de trabajo. Cabe destacar que los ficheros de código ejecutable se encuentran integrados en el núcleo MEDUSA[©], lo que implica que para ejecutarlos es necesario clonar el repositorio y

trabajar directamente sobre el código. A continuación, se detalla la estructura de funciones y dónde se pueden encontrar dentro de MEDUSA®.

1. **Carga de señales.** Las señales neuronales se encuentran en formato *.mat* y son cargadas mediante un script que las convierte a formato *.h5*.

Clase: Recording

Ubicación: medusabci / medusa-kernel / medusa / components.py

Estructura: la función empleada es la siguiente:

- `load_and_save_mat_files()`: carga todos los archivos *.mat* de una carpeta, aplica normalización Z-Score a las matrices de características e ICA (función definida en otro archivo) y guarda el resultado en formato HDF5. Las entradas que se deben especificar y las salidas que devuelve se recogen en la Tabla 2.

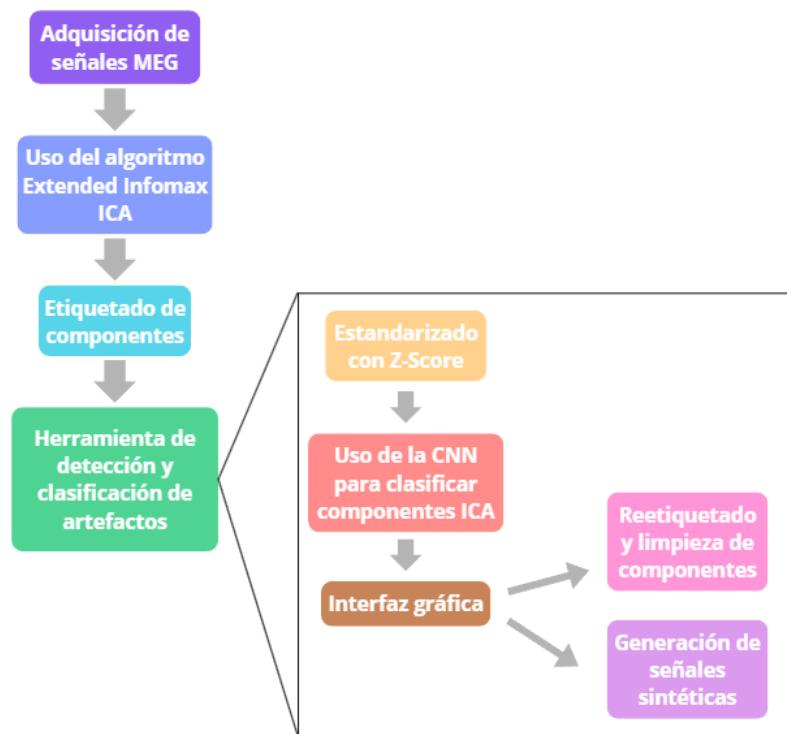


Figura 26. Flujo de trabajo seguido en la ejecución del TFM.

Entradas	input_path	Ruta a la carpeta que contiene los archivos <i>.mat</i> .
	output_path	Ruta a la carpeta donde se guardará el archivo <i>.h5</i> .
Salidas	No retorna valores. Guarda un archivo HDF5 con los datos preprocesados.	

Tabla 2. Entradas y salidas de la función de carga de señales `load_and_save_mat_files()`.

2. Estandarización de señales. Las señales extraídas son estandarizadas para mejorar el rendimiento del modelo. Se aplica una transformación Z-Score, normalizando cada componente respecto a su media y desviación típica.

Ubicación: medusabci / medusa-kernel / medusa / transforms.py

Estructura: la función que se ha creado es la siguiente:

- `zscore_normalization ()`: aplica normalización Z-Score por canal a las matrices de características y de ICA. Las entradas y salidas de esta función son las que se muestran en la Tabla 3.

3. Modelo CNN EEG-Inception. Aquí se detalla el modelo CNN implementado, la variante modificada de la arquitectura EEG-Inception.

Clase: EEGInceptionV1ICA

Ubicación: medusabci / medusa-kernel / medusa / deep_learning_models.py

Estructura: El modelo cuenta con la clase PtModel, cuya estructura de funciones principales es la siguiente:

- En primer lugar, se definen las capas con las que va a contar el modelo, divididas en 4 bloques (análisis de un solo canal, filtrado espacial, análisis de múltiples canales y bloque de salida).
- `forward ()`: se hace uso de los bloques definidos anteriormente, especificando las entradas y salidas de cada uno de ellos.
- `transform_data ()`: comprueba que los datos están en formato NumPy y los convierte a un tensor de PyTorch para poder entrenar el modelo de DL con esa librería.
- `fit ()`: entrena el modelo.
- `predict_proba ()`: predice las probabilidades de cada clase para los datos de entrada.

4. Clasificación y detección de componentes ICA. En esta sección se hace uso del modelo de CNN comentado con anterioridad, pudiendo así entrenarlo con los datos previamente cargados y llevar a cabo la etapa de test.

Entradas	<code>temp_features</code>	Matriz de características.
	<code>temp_ica</code>	Matriz de componentes ICA.
Salidas	<code>temp_features_norm</code>	Matriz de características normalizada.
	<code>temp_ica_norm</code>	Matriz ICA normalizada.

Tabla 3. Entradas y salidas de la función de normalización de señales `zscore_normalization ()`.

Ubicación: medusabci / medusa-kernel / medusa / artifact_removal.py

Estructura: la función empleada se detalla a continuación:

- `train_meg_model ()`: ejecuta el train y el test del modelo EEG-Inception modificado para clasificar señales MEG de forma binaria o discriminando entre clases. Las entradas que se deben indicar y las salidas que devuelve la función se recogen en la Tabla 4.

5. Interfaz gráfica. La interfaz gráfica permite visualizar las señales clasificadas como dudosas, junto con sus predicciones y *scores*. El usuario puede eliminar las partes contaminadas y descartar o recuperar la señal correspondiente. También desde la interfaz se permite generar señales sintéticas aplicando artefactos sobre señales neuronales limpias.

Ubicación: medusabci / medusa-kernel / medusa / artifact_removal.py

Estructura: esta es la función que se ha utilizado:

- `MEG_artifact_recovery_GUI ()`: muestra las señales clasificadas como dudosas tras haber ejecutado el anterior bloque de código y la eliminación de los artefactos presentes en estas. También permite la generación de señales sintéticas y su limpieza para evaluar el rendimiento del modelo. Las entradas que necesita la función y la salida que devuelve se muestran en la Tabla 5.

Entradas	<code>ica_power</code>	Determina si se utilizan las componentes ICA como entrada adicional al modelo.
	<code>ica_chosen_components</code>	Número de componentes ICA a utilizar por sujeto.
	<code>multi</code>	Determina si se realiza clasificación binaria o multiclas.
	<code>input_path</code>	Ruta del archivo .h5 con los datos de entrada.
	<code>n_filas</code>	Características totales (número de sujetos x número de canales).
	<code>n_columnas</code>	Número de muestras de cada señal.
	<code>n_columnas_ica</code>	Número de componentes IC totales.
	<code>confianza</code>	Umbral de probabilidad por debajo del cual se eliminan componentes dudosas.
	<code>model_path</code>	Ruta donde se guardará el modelo entrenado.
	<code>output_path</code>	Carpeta donde se guardarán los resultados del experimento.
Salidas	Modelo entrenado.	
	Archivo de resultados en formato .h5	

Tabla 4. Entradas y salidas de la función de entrenamiento y test del modelo EEG-Inception modificado `train_meg_model ()`.

Entradas	input_path	Ruta al archivo .h5 que contiene las señales.
	block_size	Tamaño de los bloques que se cargan en memoria de forma incremental para evitar problemas de memoria.
	total_rows	Características totales (número de sujetos x número de canales).
	total_cols	Número de muestras de cada señal.
	results_path	Ruta al archivo .h5 con los resultados generados por el proceso anterior.
	confidence	Umbral de probabilidad por debajo del cual se detectan artefactos.
	ica_chosen_components	Número de componentes ICA a utilizar por sujeto.
Salida	Interfaz gráfica.	

Tabla 5. Entradas y salidas de la función de interfaz gráfica *MEG_artifact_recovery_GUI()*.

Capítulo 6. Resultados

Índice

6.1.	Introducción	78
6.2.	Desempeño del modelo de clasificación.....	78
6.3.	Evaluación tras la eliminación de segmentos ruidosos.....	79
6.4.	Evaluación sobre señales sintéticas	83
6.5.	Interfaz gráfica en MEDUSA[©].....	90
6.5.1.	Ventana 1: Señales MEG reales.....	90
6.5.2.	Ventana 2: Señales MEG sintéticas.....	92

6.1. Introducción

En este capítulo se exponen los resultados obtenidos tras aplicar los métodos desarrollados para la detección automática de artefactos en señales neuronales, haciendo uso tanto de señales reales como sintéticas. Se describen las pruebas realizadas con los modelos de clasificación diseñados y se analizan los resultados finales obtenidos tras el entrenamiento del sistema. Finalmente, se presentan los resultados obtenidos tras la integración del sistema en la herramienta MEDUSA[©], evaluando su funcionamiento sobre datos preprocesados y simulados.

6.2. Desempeño del modelo de clasificación

En este apartado se presentan los resultados obtenidos durante la evaluación del modelo de clasificación diseñado para la detección y eliminación automática de artefactos en componentes ICA de registros MEG. El modelo ha sido evaluado en dos versiones: (i) *clasificación binaria*, destinada a diferenciar entre componentes cerebrales y componentes artefactuadas, y (ii) *clasificación multiclase*, que permite identificar el origen específico del artefacto (cardíaco, red eléctrica, ocular u otros), tal y como se realizó en el TFG. Sin embargo, el modelo integrado en la herramienta MEDUSA[©], que permite resaltar y eliminar las partes ruidosas de las señales, ha sido el binario debido a su mejor funcionamiento. Esto se detalla a continuación.

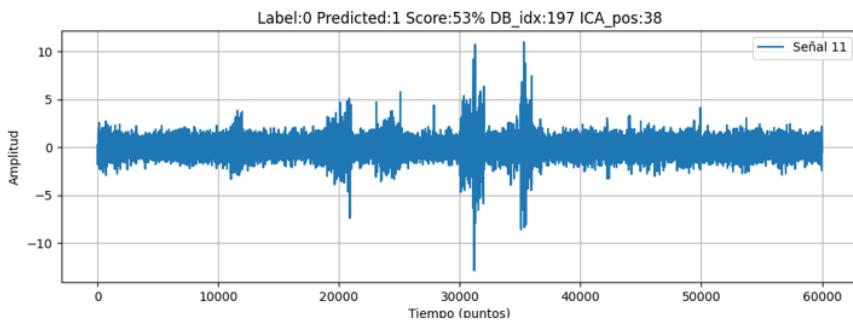
Los resultados obtenidos en la clasificación binaria consiguen una precisión del 96.92%, una sensibilidad del 67.14% y una especificidad del 98.56%. En el caso de la clasificación multiclase, el modelo ha logrado una precisión global del 86.90%, con una especificidad promedio del 88.13%. La sensibilidad, sin embargo, varía en función de la clase, siendo más elevada en componentes oculares (81.39%) y cardíacas (74.41%), y más baja en la detección de artefactos asociados a la red eléctrica (42.85%) y otros tipos menos frecuentes (25.00%). Además de las métricas clásicas, se ha calculado el coeficiente *kappa*, obteniendo un valor de 0.6824 en clasificación binaria, lo que indica un nivel de concordancia moderado con respecto al etiquetado manual, mientras que en la clasificación multiclase el valor es de 0.3450, lo que indica que existe mayor dificultad a la hora de distinguir entre tipos específicos de artefacto.

Durante el proceso de entrenamiento para el caso binario, el modelo también ha identificado un conjunto de componentes que, pese a estar etiquetadas como limpias por el técnico especialista, presentan características que el sistema considera dudosas, tal y como se ha comentado previamente. En la Figura 27 se muestran algunos ejemplos de estas componentes etiquetadas como dudosas. Este tipo de resultados pone de manifiesto

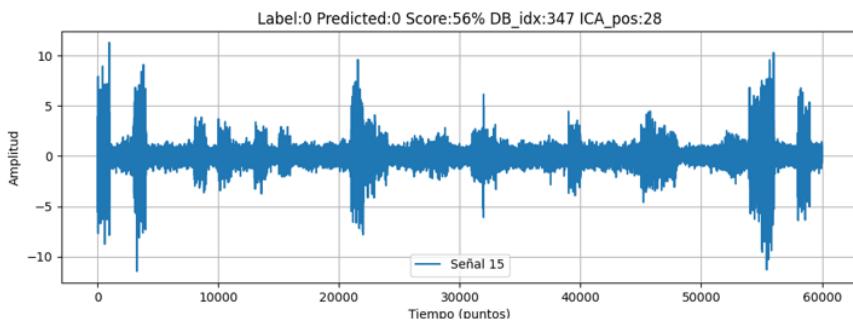
uno de los principales problemas del etiquetado manual: su dependencia de la interpretación subjetiva del experto y la posibilidad de que ciertos artefactos, especialmente los de baja intensidad o duración, pasen desapercibidos. Estas componentes dudosas han sido eliminadas progresivamente en un proceso iterativo, aplicando el criterio basado en la probabilidad asignada por el modelo (considerando dudosas aquellas componentes con probabilidad de ser de origen neuronal inferior a 0,9). Como consecuencia, se ha observado una mejora progresiva en las métricas, especialmente en la sensibilidad, llegando a obtener valores de esta del 93.68 %, 94.53 % y 95.47 % en algunas iteraciones, lo que confirma que la eliminación de muestras ambiguas favorece el rendimiento general del sistema.

6.3. Evaluación tras la eliminación de segmentos ruidosos

A continuación, se presentan los resultados obtenidos tras aplicar el proceso de eliminación de artefactos sobre una componente etiquetada como dudosa por la red. En



a) Componente número 11 etiquetada como dudosa eliminada.



b) Componente número 15 etiquetada como dudosa eliminada.

Figura 27. Dos componentes etiquetadas por el técnico especialista como no artefacto pero que la red ha eliminado por considerarlas dudosas. (a) Componente etiquetada como no artefacto (0), y la red ha predicho artefacto (1); score del 53 %. (b) Componente etiquetada como no artefacto (0), y la red ha predicho no artefacto (0); score del 56 %.

la Figura 28(a) se muestra una de esas señales en el dominio temporal, donde puede observarse cómo la señal original (en azul) presenta una serie de picos y oscilaciones de gran amplitud, distribuidos en distintos momentos, los cuales indican la presencia de artefactos. Tras aplicar la eliminación de los segmentos considerados contaminados según el *score* calculado por el modelo, se obtiene la señal recuperada (en rojo), que presenta una forma mucho más estable, con una reducción de las oscilaciones anómalas y manteniendo la estructura subyacente de la señal neuronal. Para analizar cómo este proceso afecta a la señal en el dominio frecuencial, en la Figura 28(b) y Figura 28(c) se representan los espectrogramas de la señal original y de la recuperada, respectivamente. En el primero se observa una mayor concentración de potencia en los puntos donde hay mayor contenido de ruido, lo que refleja la presencia de componentes de alta intensidad asociadas a artefactos. En cambio, en la señal recuperada la potencia disminuye de forma significativa, lo que sugiere que el proceso aplicado ha sido eficaz en la reducción de ruido y en la preservación de la actividad neuronal relevante. En la Figura 29 se muestra otra señal dudosa y su correspondiente señal recuperada, tanto en el dominio temporal como en el frecuencial. En la gráfica correspondiente al espectrograma puede verse que se reduce significativamente la potencia asociada a artefactos. No obstante, el contenido espectral subyacente se mantiene, lo que sugiere que la información neuronal no se ha visto comprometida tras la eliminación de los artefactos.

Por otro lado, en la Figura 30 se muestra este proceso de eliminación de artefactos tras aplicar diferentes niveles de segmentación a la misma señal. Se ha observado que, cuanto menor es la longitud de las segmentaciones para la clasificación, mayor es la capacidad del sistema para eliminar el ruido. Con segmentaciones de 1 segundo, la detección es mucho más localizada, permitiendo identificar y suprimir fragmentos concretos de la señal contaminados por artefactos, lo que se traduce en una señal recuperada con menos ruido. Por el contrario, al utilizar segmentaciones más largas, la detección es más general, lo que provoca que el modelo tenga menos capacidad para identificar picos puntuales de ruido, conservando parte de la potencia no deseada en la señal final.

A su vez, en la Figura 31 se muestran los resultados obtenidos tras aplicar los distintos métodos de suavizado sobre el *score*. En primer lugar, se observa que el suavizado mediante un filtro gaussiano permite una reducción mayor de la potencia. Por otro lado, el filtro de media móvil, aunque también logra reducir la presencia de artefactos, presenta algunas limitaciones derivadas de su propia naturaleza. Al aplicar una ventana con pesos uniformes y bordes abruptos, este filtro genera cortes sobre el *score*, lo que provoca que en ocasiones se eliminen pequeños fragmentos de la señal, que podrían corresponder a actividad neuronal genuina. Sin embargo, el comportamiento de ambos filtros es muy similar y las diferencias se aprecian generalmente en fragmentos concretos observados en detalle.

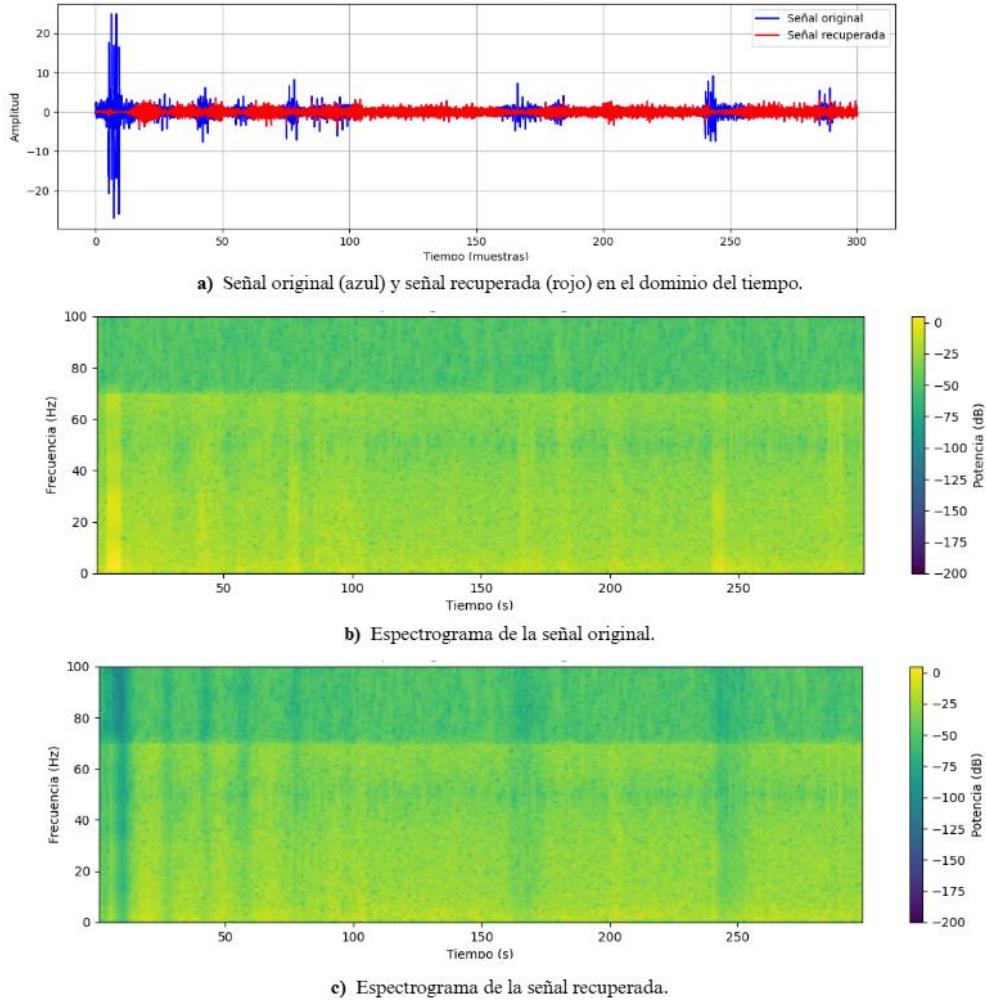


Figura 28. Señal dudosa etiquetada por un técnico como "neuronal", clasificada por la red CNN como "neuronal", con un score en la clasificación del 87 %.

Adicionalmente, se ha llevado a cabo una reevaluación de componentes dudosas tras haber eliminado el ruido que contenían con la herramienta desarrollada. Para ello, se ha obtenido el número de componentes clasificadas originalmente como dudosas y el número de componentes que detecta el algoritmo clasificadas como dudosas nuevamente tras la limpieza de artefactos en todas ellas. Esto se ha evaluado aplicando las distintas segmentaciones posibles. En la Tabla 6 se recogen los resultados obtenidos. Puede verse que, de las 14469 señales inicialmente clasificadas como dudosas, únicamente el 35.41 % se mantuvo como dudoso con segmentaciones de 1 segundo, mientras que con segmentaciones de 20 segundos este porcentaje aumentó al 43.46 %. Estos resultados dan una idea del funcionamiento de la herramienta: aplicando segmentaciones de corta duración se obtienen mejores resultados al capturar detalles concretos de las señales que con segmentaciones más largas pasarían desapercibidos. Sin embargo, cabe destacar que con segmentaciones más pequeñas se tiene un número superior de señales. Por ello, existe

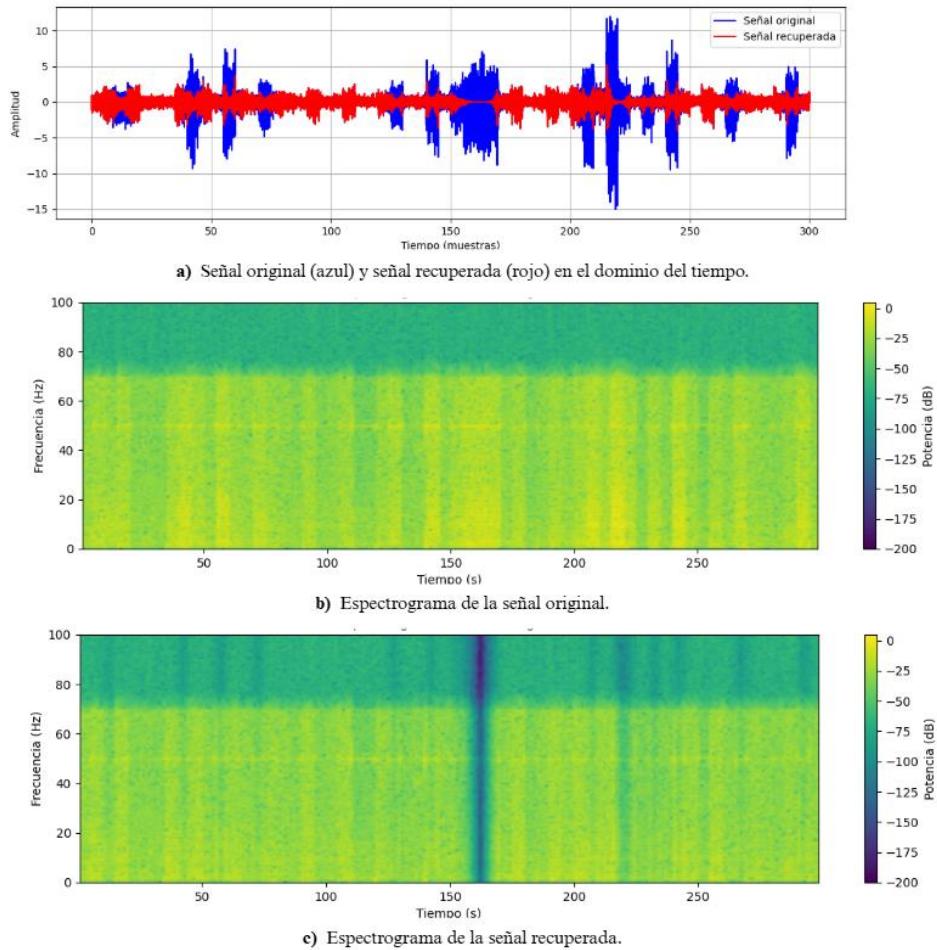


Figura 29. Señal dudosa etiquetada por un técnico como "neuronal", clasificada por la red CNN como "neuronal", con un score en la clasificación del 80 %.

un compromiso entre el tamaño de los segmentos y el tiempo de procesamiento: segmentaciones más cortas permiten una clasificación más precisa al ofrecer mayor resolución temporal en la detección de artefactos, pero incrementan de forma significativa el tiempo necesario para procesar todo el conjunto de señales.

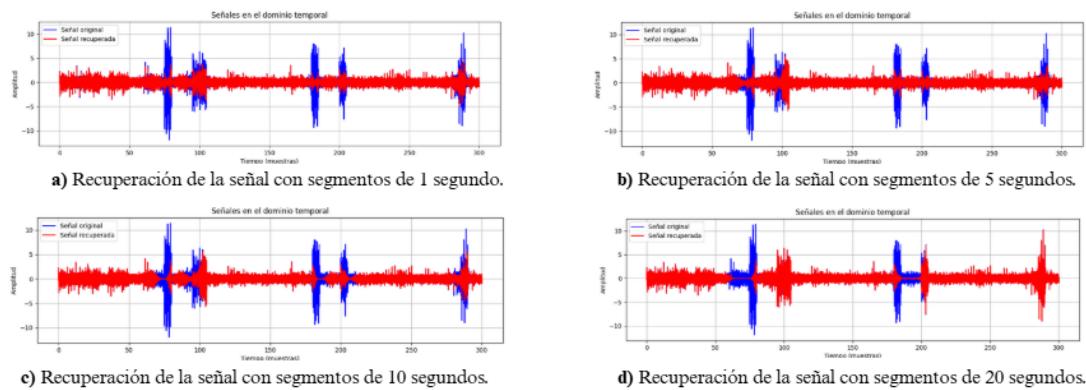
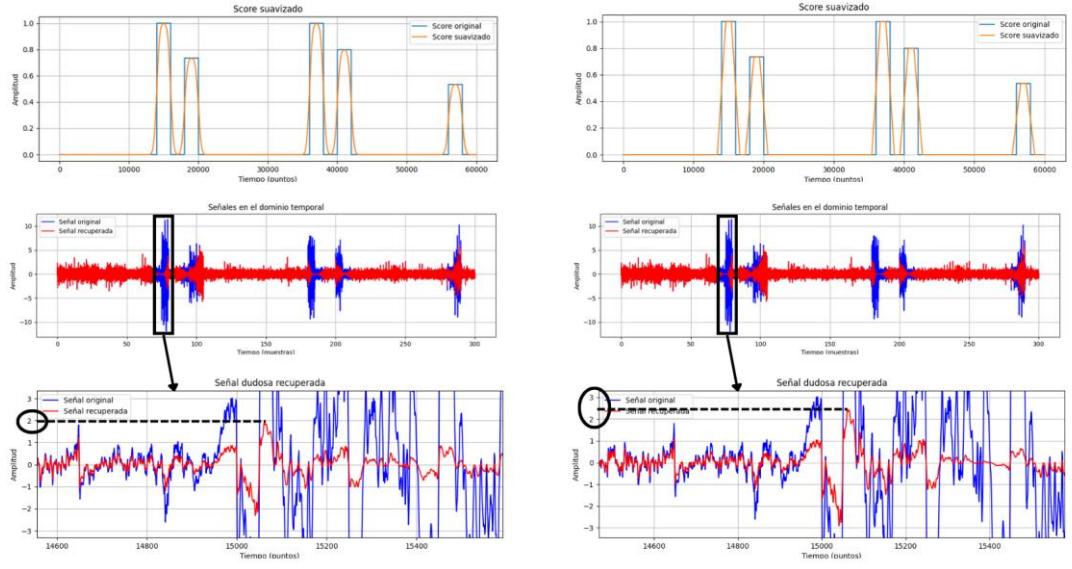


Figura 30. Representación en el dominio del tiempo de la señal original (en azul) tras aplicarle diferentes niveles de segmentación y recuperarla (en rojo).



a) Recuperación de la señal tras aplicar un filtro gaussiano al score.

b) Recuperación de la señal tras aplicar un filtro de media móvil al score.

Figura 31. Representación del score obtenido tras clasificar la señal y aplicarle diferentes métodos de suavizado y representación en el dominio del tiempo de la señal original (en azul) tras aplicarle esos suavizados y recuperarla (en rojo).

	Segmentos de 1 segundo	Segmentos de 5 segundos	Segmentos de 10 segundos	Segmentos de 20 segundos
Número original de componentes dudosas	14469 señales			
Número de componentes dudosas tras eliminar los artefactos	5123 señales (35.41 %)	5492 señales (37.96 %)	5788 señales (40 %)	6289 señales (43.46 %)

Tabla 6. Número de señales clasificadas como dudosas de nuevo tras eliminar los artefactos en las señales clasificadas originalmente como dudosas.

6.4. Evaluación sobre señales sintéticas

Con el objetivo de evaluar la eficacia del sistema desarrollado, se ha llevado a cabo una serie de pruebas utilizando las señales sintéticas generadas. Estas señales han sido creadas combinando componentes neuronales limpias con segmentos específicos de artefacto a los que se ha aplicado una máscara temporal; esta controla la aparición intermitente de los artefactos a lo largo de los cinco minutos de duración de cada señal, tal y como se comentó previamente. Gracias a este proceso es posible disponer de un *ground truth*, ya

que se conoce con exactitud qué partes de la señal corresponden a actividad cerebral y cuáles han sido contaminadas de forma artificial. Para ello, se ha aplicado el mismo detector de artefactos que en las señales reales. El modelo ha generado un *score* de probabilidad sobre cada fragmento temporal, a partir del cual se ha aplicado un proceso de suavizado y eliminación de los tramos identificados como ruidosos.

Los resultados obtenidos muestran que la herramienta es capaz de detectar con alta precisión los segmentos contaminados, y es capaz de eliminar los artefactos sintéticos introducidos. Esto se observa, por ejemplo, en la Figura 32 para señales contaminadas con artefactos cardíacos, en la Figura 33 para artefactos de red eléctrica, en la Figura 34 para oculares y en la Figura 35 para otro tipo de artefactos, todos ellos habiendo elegido un tamaño de segmento de 5 segundos de duración. En estos ejemplos se ve que, para los segmentos no contaminados, es decir, los correspondientes al primer minuto de señal, la herramienta no elimina nada y la señal neuronal permanece intacta. Por otro lado, para los dos últimos minutos, donde la señal está completamente contaminada con artefactos, la herramienta elimina la mayor parte del tramo al quedar inservible tras la contaminación. Además, en las gráficas con *zoom* se aprecia mejor la reducción del artefacto en cuestión. Por ejemplo, para el caso en el que se añade señal cardíaca a la componente neuronal, el complejo QRS se ve reducido en gran medida. Lo mismo sucede para los picos propios de la componente de red eléctrica y los oculares. En estos últimos es posible ver cómo la potencia del artefacto propio de pestaños se reduce significativamente tras aplicar el proceso de eliminación de artefactos sobre las señales neuronales contaminadas de forma artificial.

Adicionalmente, se ha analizado la influencia del ajuste en la capacidad de detección de la herramienta, es decir, cómo varía el número de detecciones correctas para cada uno de los minutos de señal sintética según la SNR aplicada. Para ello, se ha escogido aleatoriamente un conjunto elevado de señales neuronales y artefactos de los distintos tipos disponibles (cardíacas, de red eléctrica, oculares y de otro tipo) con el fin de capturar la mayor variabilidad posible de señales. A continuación, en lugar de seleccionar manualmente el tramo de artefacto deseado para sumárselo a la señal neuronal, se ha segmentado la señal artefacto en segmentos de 5 segundos, los cuales se han introducido en la herramienta de IA desarrollada. Seguidamente, se ha obtenido el *score* de clasificación de cada uno de los segmentos, y se ha escogido el tramo de señal artefacto cuyo *score* era superior a 0.9 durante el mayor intervalo de tiempo continuo. Así se garantiza que se selecciona el artefacto más relevante y de mayor duración de forma automática de cada una de las señales. Posteriormente, se ha replicado el tramo seleccionado hasta obtener una señal de 5 minutos de duración, y se ha aplicado la máscara creada. El siguiente paso ha consistido en modificar la potencia de la señal artefacto creada respecto de la señal neuronal. Para ello, los porcentajes de variación de

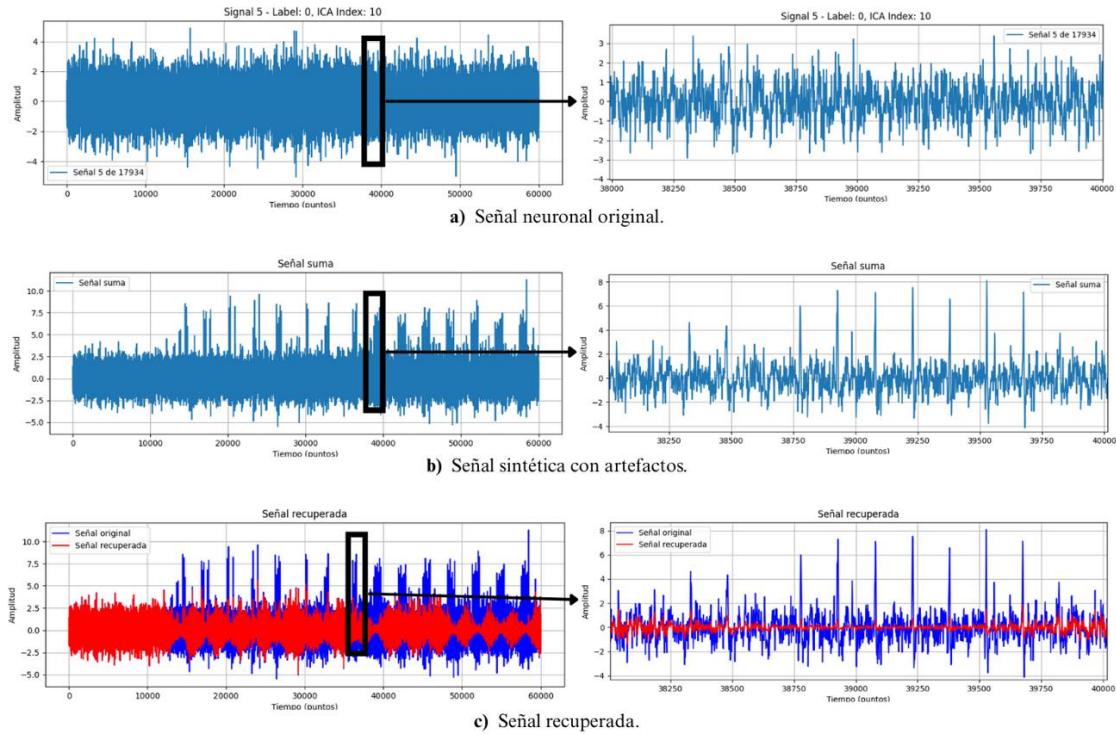


Figura 32. Componente neuronal número 5 (fila izquierda) y zoom de la muestra 38000 a la 40000 (fila derecha), sobre la que se ha sumado un artefacto de tipo cardíaco para generar una señal sintética artefactuada, que se ha introducido en el método desarrollado para obtener la señal recuperada equivalente.

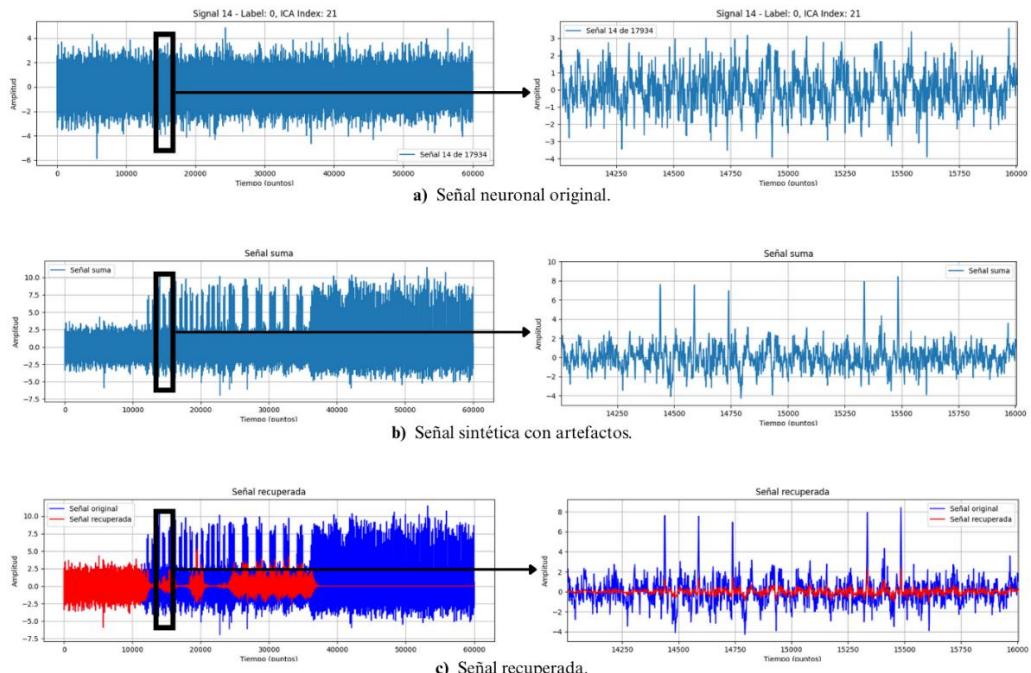


Figura 33. Componente neuronal número 14 (fila izquierda) y zoom de la muestra 14000 a la 16000 (fila derecha), sobre la que se ha sumado un artefacto de red eléctrica para generar una señal sintética artefactuada, que se ha introducido en el método desarrollado para obtener la señal recuperada equivalente.

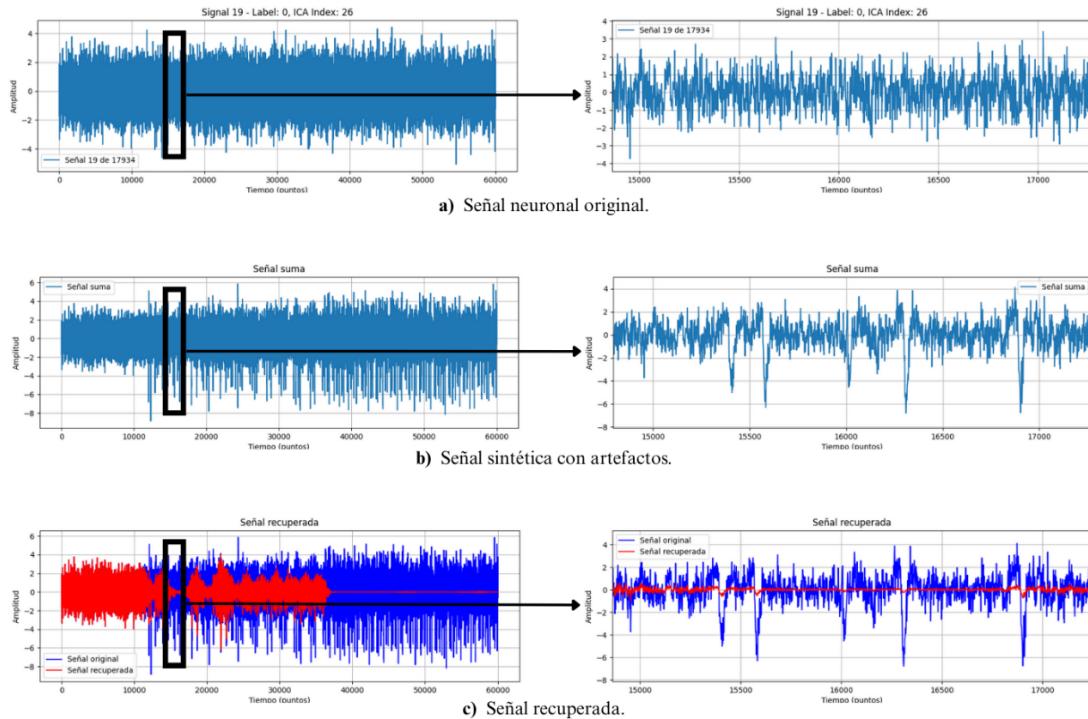


Figura 34. Componente neuronal número 19 (fila izquierda) y zoom de la muestra 15000 a la 17000 (fila derecha), sobre la que se ha sumado un artefacto de tipo ocular para generar una señal sintética artefactuada, que se ha introducido en el método desarrollado para obtener la señal recuperada equivalente.

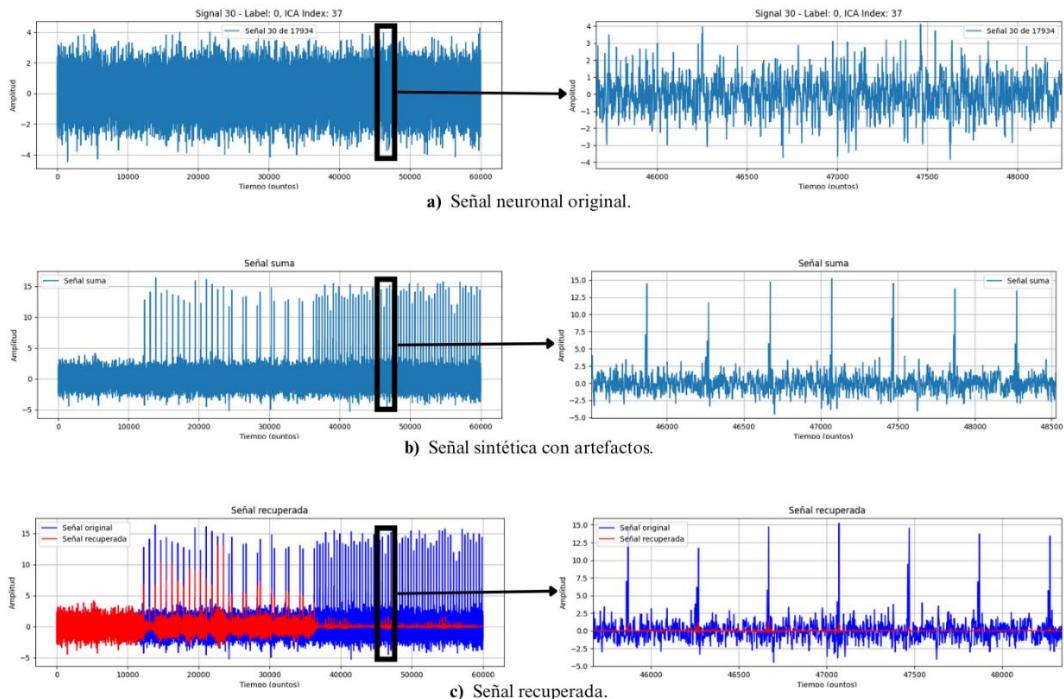


Figura 35. Componente neuronal número 30 (fila izquierda) y zoom de la muestra 46000 a la 48000 (fila derecha), sobre la que se ha sumado un artefacto de otro tipo para generar una señal sintética artefactuada, que se ha introducido en el método desarrollado para obtener la señal recuperada equivalente.

señal escogidos han variado del 10 % al 400 %. Los últimos pasos han sido los explicados en capítulos anteriores: suma del artefacto con ajuste a la señal neuronal, paso de la señal resultante por el algoritmo de IA con diferentes segmentaciones y obtención del *score* de clasificación para cada segmento. A partir de los resultados obtenidos, se ha analizado el porcentaje de detecciones correctas para cada minuto de señal. Esto se ha hecho así porque el porcentaje de ruido añadido en cada minuto es distinto. Por ejemplo, el primer minuto de señal, al cual no se le añade ruido al valer la máscara cero, no depende del ajuste aplicado en ninguno de los casos.

En la Figura 36 se muestra la gráfica obtenida para el primer minuto, habiendo realizado segmentaciones de 1, 5, 10 y 20 segundos. Puede verse que como este primer tramo es independiente del ajuste aplicado al no añadirle ruido, las gráficas obtenidas son muy similares. El *score* obtenido en este caso se refiere al porcentaje de señal detectado como no artefacto que, como puede observarse, es bastante elevado, superior en todos los casos al 90 %.

En la Figura 37 se muestra la gráfica correspondiente al segundo minuto y en la Figura 38 la correspondiente al tercer minuto. La primera corresponde a intervalos de ruido de 2 segundos de duración y luego 2 segundos sin ruido, mientras que a la segunda se le añade ruido cada 10 segundos, con una duración del tramo ruidoso de 5 segundos. Por eso, en este caso el *score* obtenido se refiere al porcentaje de señal detectado como artefacto. En ambas situaciones el 50 % del tiempo la señal está libre de artefactos. Sin embargo, en el primer caso, con segmentaciones de 1 segundo la herramienta es capaz de discernir bien qué es artefacto y qué no lo es al haber introducido tramos de ruido de 2 segundos (es

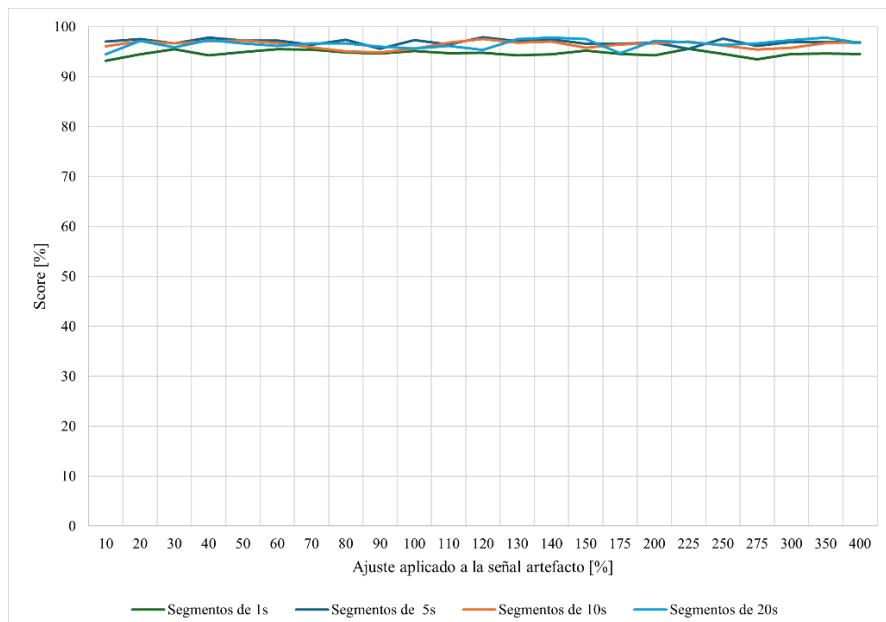


Figura 36. Porcentaje de señal detectado como no artefacto durante el primer minuto en función del ajuste aplicado a la señal habiendo realizado segmentaciones de 1 (en verde), 5 (en azul oscuro), 10 (en naranja) y 20 segundos (en azul claro).

decir, los segmentos son de menor tamaño que los tramos de ruido), y así alcanzar la máxima precisión, que en este caso es del 50 %. A partir de segmentaciones de 5 segundos el porcentaje de detección aumenta porque detecta todo el tramo como ruidoso al estar mayoritariamente contaminado. Lo mismo sucede en la segunda situación, solo que la máxima precisión se puede alcanzar también con segmentaciones de 5 segundos al durar los tramos ruidosos dicho tiempo.

Finalmente, en la Figura 39 puede verse la gráfica para los dos últimos minutos, a los cuales se les añade el ruido generado de forma continua. En este caso el porcentaje de señal detectado como artefacto aumenta a medida que se incrementa el ajuste de potencia aplicado a la señal, llegando a alcanzar porcentajes superiores al 90 % a partir de un ajuste de potencia del artefacto del 200 % de la potencia de la señal neuronal.

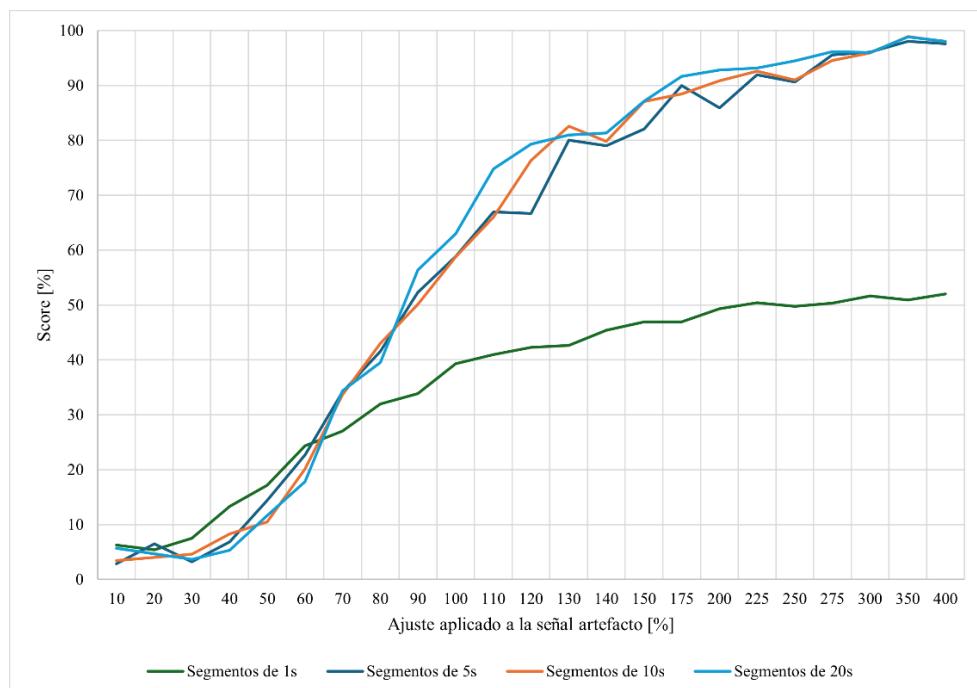


Figura 37. Porcentaje de señal detectado como artefacto durante el segundo minuto en función del ajuste aplicado a la señal habiendo realizado segmentaciones de 1 (en verde), 5 (en azul oscuro), 10 (en naranja) y 20 segundos (en azul claro).

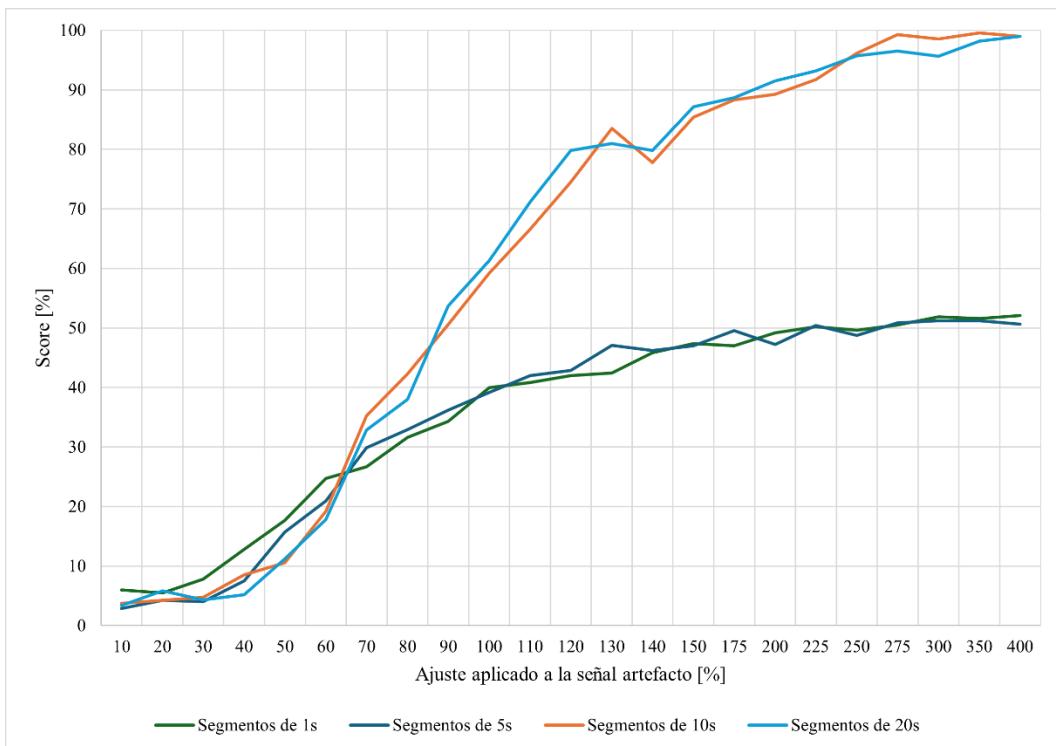


Figura 38. Porcentaje de señal detectado como artefacto durante el tercer minuto en función del ajuste aplicado a la señal habiendo realizado segmentaciones de 1 (en verde), 5 (en azul oscuro), 10 (en naranja) y 20 segundos (en azul claro).

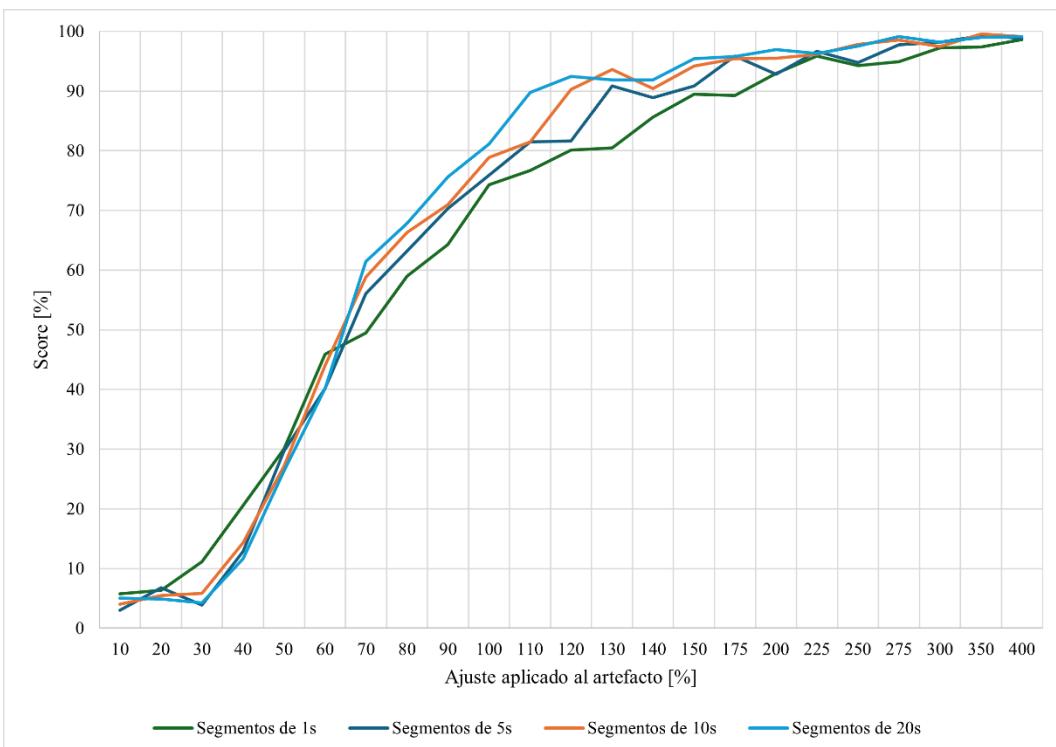


Figura 39. Porcentaje de señal detectado como artefacto durante los dos últimos minutos en función del ajuste aplicado a la señal habiendo realizado segmentaciones de 1 (en verde), 5 (en azul oscuro), 10 (en naranja) y 20 segundos (en azul claro).

6.5. Interfaz gráfica en MEDUSA[©]

En este apartado se presenta la interfaz gráfica desarrollada como parte de la herramienta de análisis de artefactos. Esta interfaz ha sido diseñada con el objetivo de facilitar la interacción con los resultados obtenidos por la red neuronal, así como permitir un control para los usuarios sobre el proceso de eliminación de artefactos. La interfaz cuenta con dos ventanas principales: la primera está dedicada a la visualización de las señales clasificadas como dudosas, donde se pueden observar los tramos contaminados identificados automáticamente y llevar a cabo su eliminación; la segunda está enfocada en la generación de señales sintéticas, y permite comprobar cómo actúa el sistema ante diferentes niveles de ruido. A continuación, se describen ambas ventanas y las funcionalidades que ofrecen.

6.5.1. Ventana 1: Señales MEG reales

La primera ventana se muestra en la Figura 40 y se denomina *Doubt Signals* ya que en ella se representan las señales clasificadas como dudosas por el algoritmo de IA. Cuenta con dos gráficas donde se pueden ver las distintas señales generadas, desde las dudosas hasta las recuperadas limpias de artefactos, así como el resultado de los segmentos ruidosos o la gráfica del *score* correspondiente. Junto a ellas hay dos paneles de información que describen las características más relevantes de las señales que se estén representando en ese momento; estos son la media, la desviación estándar, los valores máximo y mínimo, la mediana, la potencia, el valor cuadrático medio, el número de cruces por cero y la frecuencia dominante.

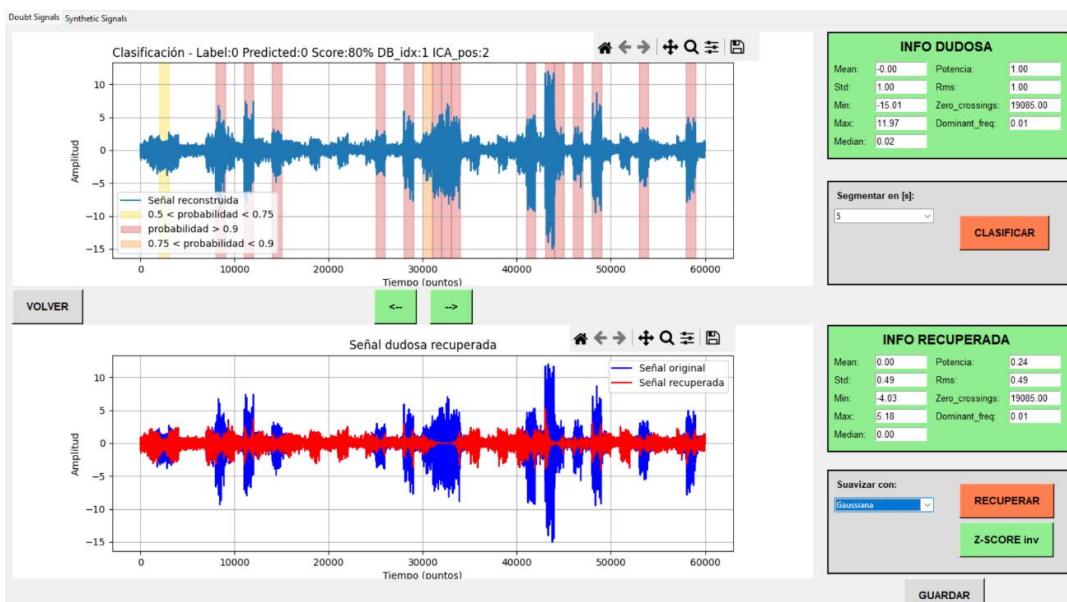
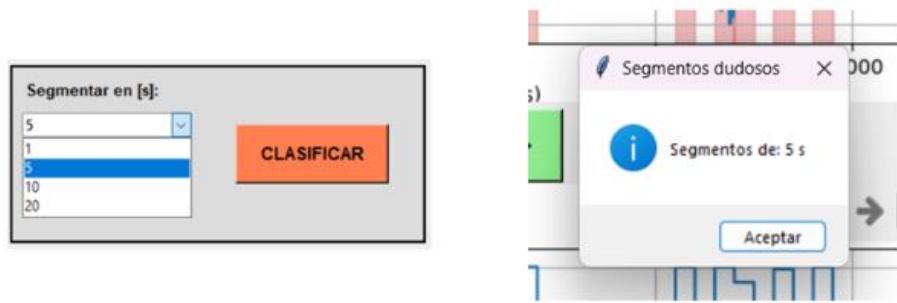


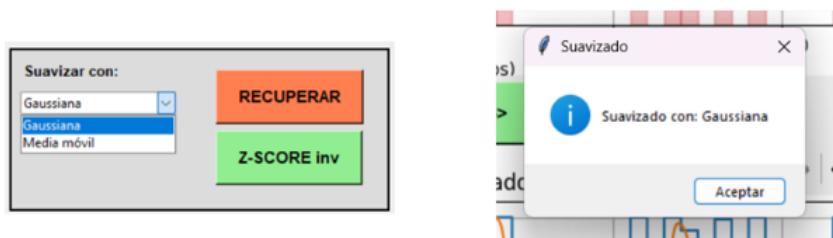
Figura 40. Primera ventana de la interfaz gráfica.

Para segmentar las señales se dispone de un desplegable que permite seleccionar la duración de los segmentos. Esto es lo que se muestra en la Figura 41 junto con la ventana emergente que aparece al seleccionar una de las opciones. Se obtienen los segmentos clasificados pulsando sobre el botón de *CLASIFICAR*, para a continuación poder ver los segmentos considerados ruidosos en la primera gráfica y el *score* correspondiente en la segunda. A continuación, este último se puede suavizar seleccionando una de las dos opciones del segundo desplegable, tal y como se puede ver en la Figura 42. De nuevo, al seleccionar una de ellas aparece un cuadro informativo. Para limpiar la señal empleando el *score* suavizado no hay más que clicar sobre el botón de *RECUPERAR*, y para obtener su potencia original se debe pulsar sobre el botón de *Z-SCORE inv*. Finalmente, se puede guardar la señal recuperada seleccionando el botón de *GUARDAR* si se considera que la eliminación de los artefactos ha sido eficaz y la señal puede contribuir a la reconstrucción de la señal original, en este caso, libre de artefactos.



a) Desplegable para elegir la duración de los segmentos y botón de *clasificar*.
b) Ventana informativa de confirmación.

Figura 41. Botones empleados para clasificar los segmentos de las señales.



a) Desplegable para elegir el suavizado del *score* y botones de *recuperar* y de *Z-Score inverso*.
b) Ventana informativa de confirmación.

Figura 42. Botones empleados para suavizar el *score* y recuperar la señal neuronal.

6.5.2. Ventana 2: Señales MEG sintéticas

La segunda ventana se muestra en la Figura 43 y se denomina *Synthetic Signals* ya que en ella se representan las señales sintéticas generadas a partir de la suma de una señal neuronal y un artefacto. De nuevo, cuenta con dos gráficas donde se pueden ver las señales que contienen exclusivamente componente neuronal y aquellas correspondientes a artefactos. Para estas últimas, es posible elegir el tipo de artefacto gracias al desplegable que se muestra en la Figura 44. También dispone de dos paneles de información que describen las características más relevantes de las señales que se estén mostrando.

Una vez elegidas las señales con las que se desea trabajar, tanto la neuronal como el artefacto, se debe especificar el segmento concreto de artefacto que se desea añadir a la señal neuronal. Para ello, se debe indicar la muestra de inicio y de fin de dicho segmento, y clicar sobre el botón de *IR*. A continuación, se clicará sobre el botón de *MÁSCARA* para aplicar el filtro creado al artefacto, y se variará su potencia respecto a la de la señal neuronal según se desee. Para ello, la interfaz cuenta con otro desplegable que permite seleccionar el porcentaje de ajuste, tal y como se puede ver en la Figura 45. Para añadir el ruido creado a partir de un artefacto a la señal neuronal no hay más que clicar sobre el botón de *SUMAR*. Tras esto es posible ver la señal sintética generada en la primera gráfica. Finalmente, se puede limpiar dicha señal siguiendo los mismos pasos que los explicados en la primera ventana: elegir la duración de los segmentos y su posterior clasificación, y suavizado del *score* y recuperación de la señal limpia.

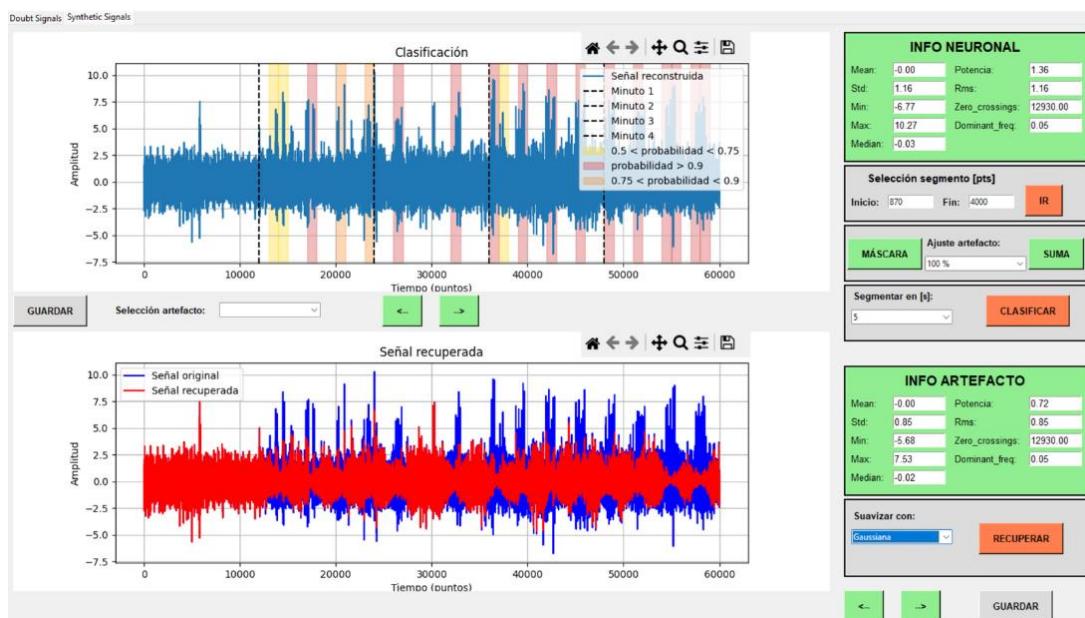


Figura 43. Segunda ventana de la interfaz gráfica.



a) Desplegable para elegir el artefacto para generar la señal sintética. b) Ventana informativa de confirmación.

Figura 44. Botones empleados para seleccionar el tipo de artefacto.



a) Desplegable para elegir el ajuste aplicado al artefacto. b) Ventana informativa de confirmación.

Figura 45. Botones empleados para ajustar la potencia del artefacto respecto de la potencia de la señal neuronal.

Capítulo 7. Discusión y limitaciones

Índice

7.1.	Introducción	95
7.2.	Discusión	95
7.3.	Limitaciones	98

7.1. Introducción

En este TFM se ha implementado un detector automático de artefactos en señales provenientes de registros MEG que permite reducir la presencia de dichos artefactos gracias al *score* calculado por la propia red. Asimismo, se han realizado diversas pruebas a partir de señales sintéticas, generadas mediante la combinación de señales neuronales y componentes artefactuadas, para evaluar la capacidad del método desarrollado para llevar a cabo una eliminación de los artefactos. Todo esto ha sido integrado en el software MEDUSA[©]. En este capítulo se analizan los resultados obtenidos, se contrastan con investigaciones previas en el mismo ámbito y se exponen las principales limitaciones detectadas a lo largo del proceso de diseño y desarrollo de la herramienta.

7.2. Discusión

En el presente TFM se ha desarrollado un sistema basado en técnicas de IA, concretamente una CNN, para la detección y eliminación automática de artefactos en señales MEG a partir de componentes ICA. La CNN empleada se basa en una arquitectura similar a EEG-Inception y ha sido reimplementada en PyTorch, para optimizar y ampliar con nuevas funcionalidades la versión desarrollada previamente en el TFG. En este nuevo sistema destaca su capacidad de actuar directamente sobre el ruido de las señales contaminadas. La herramienta desarrollada ha puesto de manifiesto que la clasificación realizada manualmente por técnicos especialistas, al basarse en criterios subjetivos, puede conllevar errores significativos. En varios casos, se identificaron componentes que, pese a haber sido etiquetadas como de origen neuronal, presentaban fragmentos contaminados por artefactos. Del mismo modo, algunas componentes marcadas como ruidosas correspondían mayoritariamente a actividad cerebral válida. Además, el sistema posteriormente justifica visualmente su decisión, resaltando en la señal los tramos que considera artefactuados, lo que facilita la interpretación del resultado y refuerza la utilidad de la herramienta como apoyo al técnico. Con esto lo que se consigue es que la herramienta informe al usuario por qué una señal dice que es artefacto mediante el resultado de las partes que considera ruidosas, asemejándose al comportamiento típico de una IA explicativa (XAI, *eXplainable Artificial Intelligence*). Otro aspecto destacable ha sido la utilización de señales sintéticas, generadas a partir de la combinación de componentes cerebrales con segmentos replicados de artefactos reales, modulados mediante una máscara temporal. Esto ha permitido disponer de un conjunto de datos con un *ground truth* conocido, que se ha empleado para validar el comportamiento del modelo. Otra novedad que cabe mencionar respecto al TFG previo ha sido el uso del *score* de salida del modelo, posteriormente suavizado mediante técnicas como el enventanado

o el filtrado gaussiano, para determinar los tramos concretos de las señales contaminados por artefactos. Esto ha permitido evolucionar la herramienta de clasificación, aplicando una eliminación parcial y localizada del ruido, sin necesidad de descartar la componente completa. Finalmente, el sistema ha sido integrado en la plataforma MEDUSA[®], lo que permite su uso en diversos análisis del ámbito de la neurociencia y de las neurotecnologías. Su funcionalidad permite la carga de señales, la aplicación del modelo y la visualización de los tramos ruidosos, así como la generación de señales sintéticas.

Otros estudios realizados previamente emplean también métodos de IA para detectar artefactos en señales provenientes de registros EEG. Así, en ICLabel se emplea una red neuronal para etiquetar automáticamente componentes ICA en EEG, consiguiendo altas tasas de precisión en los resultados (Pion-Tonachini, et al., 2019). El clasificador automático que emplean clasifica las señales en diferentes categorías o clases, y cuenta con más de 200.000 componentes independientes, provenientes de más de 6.000 registros de EEG. Al disponer de tantos datos, el clasificador consigue generalizar muy bien en diferentes condiciones de registro. Además, en ICLabel se realiza un etiquetado colaborativo o *crowdsourcing*; en este, múltiples expertos se encargan de etiquetar las componentes para, posteriormente, combinar todas las etiquetas disponibles y solventar los desacuerdos asignando más peso a aquellos expertos cuyas decisiones tienden a coincidir con las de los demás, es decir, que muestran una mayor consistencia con el grupo. Así, se obtiene una etiqueta final más representativa y fiable. También hacen uso de una CNN, aplicando el optimizador Adam, y emplean diferentes técnicas para mejorar el entrenamiento: como el balanceo de clases, para abordar el desequilibrio que pueda existir entre las distintas clases que forman la base de datos, o *early stopping*, para mitigar el sobreajuste. A su vez, Garg *et al.* (2017, 2021) propusieron dos estudios empleando una CNN con distinta estructura para identificar artefactos en MEG. En el primero, realizado en el año 2017, la CNN era de una única dimensión, y los datos provenían de registros MEG realizados a 49 sujetos en estado de reposo, cuyo etiquetado fue realizado por un único experto. En este caso el modelo identificaba artefactos oculares, y la precisión alcanzada fue de 95.86 % (Garg, et al., 2017). En el segundo estudio, realizado en el año 2021 y donde se desarrolló la herramienta MEGnet, comentada previamente, se amplió la base de datos hasta contar con un total de 217 sujetos para realizar los registros, y la CNN integraba tanto datos temporales como espaciales. En este estudio se identificaron artefactos oculares, cardíacos y de otro tipo mientras los pacientes se encontraban realizando tareas y en estado de reposo con los ojos cerrados. El etiquetado lo realizaron 4 expertos mediante una inspección visual independiente, y se alcanzó hasta un 98.95 % de precisión, un 96.74 % de sensibilidad y un 99.34 % de especificidad (Garg, et al., 2021). Estos resultados son comparables a los obtenidos en este TFM, donde se ha contado con más del doble de sujetos que se encontraban en estado de reposo con los ojos

cerrados, y donde se aplicó ICA junto con una inspección visual. A pesar de que los resultados ofrecidos por la CNN para una clasificación binaria son peores que los que obtuvieron con MEGnet (precisión del 96.92 %, sensibilidad del 67.14 % y especificidad del 98.56 %), se ha podido demostrar que el proceso de reevaluación de componentes mejora los resultados, alcanzando una sensibilidad de hasta el 95.47 %.

En cuanto a la arquitectura empleada en este TFM, esta podría compararse con la que se desarrolló en EEGNet, una red basada en un modelo CNN aplicable a tareas de clasificación e identificación de artefactos en señales EEG (Lawhern, et al., 2018). La adaptación de la red EEG-Inception es más compleja que EEGNet, dado que esta utiliza una única entrada con una estructura secuencial de convoluciones temporales y espaciales, mientras que la arquitectura propuesta en este TFM cuenta con dos entradas distintas: una que contiene las características temporales de cada componente y otra que asocia cada una con su canal correspondiente. Aunque el diseño resultante es más complejo que el de EEGNet y, por tanto, más exigente a nivel computacional, ofrece una mayor flexibilidad y capacidad para abordar el problema específico de detección de artefactos en MEG, donde los patrones pueden ser sutiles, variables y difíciles de distinguir con arquitecturas más simples.

A diferencia de métodos que sólo clasifican componentes, en este TFM se aplica un preprocesamiento mediante *score*, con capacidad de eliminación de los artefactos en MEG. Esta aproximación local de eliminación de ruido coincidiría con estrategias similares que evitan eliminar canales enteros o señales completas, tal y como se planteó con *Autoreject* (Jas, et al., 2017). Aquí, se emplea una técnica que elimina automáticamente segmentos ruidosos de EEG y MEG combinando regresión y heurísticas. Hace uso de la técnica de estimación de umbral óptimo de pico a pico, donde se estima un umbral para cada sensor por separado. Los sensores se consideran defectuosos si la amplitud de pico a pico de su señal excede dicho umbral y, si el número de sensores defectuosos en un ensayo es bajo, los datos corruptos de esos sensores se reemplazan mediante interpolación a partir de las señales de los sensores cercanos. Por otro lado, si la mayoría de los sensores en un ensayo están defectuosos, este se considera irreparable por interpolación y se excluye del análisis posterior (Jas, et al., 2017). Esto imita cómo un experto humano marcaría un ensayo como defectuoso durante la inspección visual. Aunque *Autoreject* no usa DL, ha demostrado reducir la presencia de artefactos sin la necesidad de descartar canales completos. Sin embargo, no distingue el tipo de artefacto presente ni ofrece una clasificación específica sobre su origen. Además, el método de eliminación de artefactos que emplea mediante la exclusión de sensores puede provocar la pérdida de información neuronal importante si dichos sensores contienen, además de ruido, componentes neuronales.

Finalmente, la generación de señales sintéticas ha permitido evaluar el comportamiento de la herramienta diseñada al conocer la cantidad de ruido que se introducía a las señales neuronales limpias. Otros autores han desarrollado técnicas para la generación de señales cerebrales de forma artificial con el fin de ampliar la base de datos de la que disponían y poder evaluar sus herramientas de análisis. Esto es lo que realizaron, por ejemplo, Vaziri *et al.* (2023) con la herramienta EEGg (*EEG generator*) (Vaziri, et al., 2023). Esta permite generar señales EEG sintéticas permitiendo al usuario, entre otras cosas, elegir el número de fuentes, añadir ruido blanco a las señales, o seleccionar el número de electrodos para simular el registro. Para ello, hicieron uso de la plataforma Matlab, y emplearon imágenes de resonancia magnética para obtener una matriz que simula la propagación de las señales desde las fuentes neuronales hasta los sensores. Con esto, se ofrece a los usuarios una herramienta que permite simular escenarios realistas y probar algoritmos de localización de fuentes u otros análisis con la ventaja de conocer el *ground truth* subyacente.

7.3. Limitaciones

En este TFM se ha tratado de subsanar las limitaciones encontradas en el TFG en cuanto a usabilidad de la herramienta de detección de artefactos se refiere, de forma que se facilite el acceso a sus funciones, aumentando la transparencia del software implementado mediante la creación de la interfaz gráfica y su implementación en MEDUSA[©]. Aun así, existen algunas limitaciones que se deben tener en cuenta.

Por un lado, la cantidad y la calidad de los datos de los que se dispone para entrenar la herramienta de identificación y eliminación de artefactos influyen directamente en los resultados obtenidos. Como el número de sujetos que ha participado para formar la base de datos ha sido escaso, debido a la complejidad y alto coste característicos de MEG, el modelo puede aprender patrones específicos de ese grupo limitado, pero puede fallar al enfrentarse a nuevos sujetos con características fisiológicas distintas. Esto es lo que se conoce como limitada capacidad de generalización. Además, al haber pocos sujetos, algunas clases, como por ejemplo los artefactos cardíacos u oculares, están infrarepresentadas o no tienen suficiente variabilidad. Esto puede provocar un desbalanceo que afecta al rendimiento del modelo, reduciendo su sensibilidad frente a esas clases menos frecuentes. En cuanto a la calidad de los datos, como estos han sido etiquetados por un único técnico especialista existe cierto grado de error o subjetividad que hace que la clasificación efectuada por la red pueda ser incorrecta. Asimismo, un solo experto no tiene mecanismos de corrección ni contraste, por lo que, si comete un error, ese error se traslada al modelo, lo que puede afectar negativamente al entrenamiento y generar modelos con aprendizaje equivocado. Con esto se destaca la importancia de

contar con la mayor cantidad posible de sujetos, tanto para construir la base de datos como para etiquetar las señales que la componen.

Por otro lado, las señales sintéticas generadas para evaluar el comportamiento de la herramienta y proporcionar un *ground truth* fiable no reproducen toda la complejidad de una señal neuronal real. El proceso de replicar artefactos puede introducir patrones demasiado regulares al construirse mediante la combinación de fragmentos reales de artefactos replicados y sumados a señales neuronales limpias, lo que facilita la tarea del modelo y no representa la variabilidad real. En condiciones naturales, los artefactos pueden superponerse, variar en intensidad, duración y morfología de forma no predecible, mientras que las señales generadas artificialmente tienden a presentar patrones más regulares. Además, el método de generación está limitado a artefactos simples o conocidos obtenidos únicamente a partir de señales MEG, y no incluye interacciones ni artefactos compuestos. Sería interesante incorporar señales complementarias como ECG o EOG en los análisis, ya que permitirían identificar de forma más precisa el origen de ciertos artefactos fisiológicos, como los de tipo cardíaco u ocular, respectivamente.

En cuanto al proceso de eliminación de artefactos mediante el *score* calculado por el modelo, se debe destacar que su aplicación requiere de ajustes minuciosos para garantizar que se eliminan únicamente los segmentos realmente contaminados sin afectar a la actividad neuronal válida. El uso de técnicas de suavizado, como el filtrado gaussiano o el enventanado, si bien ayuda a evitar cortes abruptos en la señal, puede provocar la difuminación de los límites reales entre segmentos ruidosos y limpios, dificultando la identificación del inicio y fin de los artefactos. Esto puede conllevar a la eliminación parcial de actividad neuronal válida o, por el contrario, a la preservación de fragmentos contaminados con bajo *score*. Además, la elección del umbral a partir del cual se elimina la señal no es trivial y puede afectar significativamente al resultado, especialmente en señales con niveles intermedios de contaminación.

Otra limitación a considerar es la integración con MEDUSA[©] y la carga computacional que supone el uso de modelos de DL como EEG-Inception, especialmente durante la fase de entrenamiento. Aunque la herramienta funciona correctamente, la falta de aceleración por GPU en algunos entornos o equipos limita su rendimiento, lo que puede traducirse en tiempos de procesamiento elevados. Esta restricción puede dificultar su uso en tiempo real o en sistemas con recursos limitados, por lo que sería recomendable optimizar el modelo o implementar versiones más ligeras para entornos con menor capacidad de cómputo. Esto podría conseguirse mediante la aplicación de técnicas que permitieran reducir la base de datos, como PCA o *Autoencoders*, que reducen la dimensionalidad y aceleran el procesamiento del modelo.

Por último, cabe destacar que la funcionalidad desarrollada se encuentra integrada en el núcleo MEDUSA[©], lo que implica que para ejecutarla es necesario clonar el repositorio y trabajar directamente sobre el código, generalmente desde entornos como PyCharm. Esta dependencia técnica limita su accesibilidad a usuarios sin conocimientos de programación. Lo ideal sería empaquetar la herramienta como una aplicación descargable dentro de la interfaz de la plataforma MEDUSA[©], permitiendo que cualquier usuario, incluso sin experiencia en desarrollo, pueda utilizarla de forma intuitiva y autónoma. Esta limitación podría solventarse en futuras versiones para así mejorar la usabilidad del sistema.

Capítulo 8. Conclusión y líneas futuras

Índice

8.1. Introducción	102
8.2. Grado de consecución de los objetivos propuestos.....	102
8.3. Conclusiones y aportaciones originales.....	103
8.4. Líneas futuras	104

8.1. Introducción

En este último capítulo se reflexiona sobre el trabajo realizado, evaluando en qué medida se han alcanzado los objetivos planteados al inicio del proyecto. A partir de los resultados obtenidos y tras la ejecución del TFM, se extraen una serie de conclusiones y se destacan las principales contribuciones aportadas. Además, se plantean futuras líneas de desarrollo que podrían ampliar el alcance de esta investigación, así como los posibles escenarios de aplicación de la herramienta desarrollada en contextos clínicos o experimentales.

8.2. Grado de consecución de los objetivos propuestos

En base a los objetivos propuestos en el primer capítulo de este TFM, en este apartado se revisa el grado de consecución de cada uno de ellos y su contribución para la elaboración de la herramienta de detección y eliminación de artefactos en señales neuronales diseñada:

1. A partir de diversos libros, artículos de revista y otros estudios se ha entendido el comportamiento de los diferentes tipos de artefactos que afectan las señales electromagnéticas cerebrales y se han analizado las limitaciones de los métodos tradicionales para detectar y eliminar estos artefactos.
2. Mediante una búsqueda de diferentes fuentes en Internet, se han podido estudiar las principales técnicas utilizadas en el procesamiento de señales en neurociencia. Se ha ahondado específicamente en el papel de las herramientas modernas, como las basadas en IA, para automatizar y mejorar la precisión en el análisis de señales neuronales.
3. Se ha llevado a cabo un análisis exhaustivo del software MEDUSA© y se ha examinado para qué estudios de investigación neurocientífica ha sido empleada.
4. Se ha diseñado una herramienta basada en CNN que permite detectar, clasificar y eliminar artefactos en señales procedentes de registros MEG en estado de reposo.
5. A partir de la técnica generada, se han creado señales sintéticas y se ha podido demostrar que el funcionamiento de la herramienta de IA es correcto. También se ha podido analizar su rendimiento al comparar los resultados empleando señales MEG reales y las señales sintéticas generadas.
6. Se han implementado métodos de eliminación de los segmentos contaminados por ruido detectados por la herramienta de IA diseñada.

7. Con el fin de ofrecer al usuario una herramienta visual e intuitiva se ha creado una interfaz gráfica que integra las funcionalidades mencionadas.
8. Se ha llevado a cabo la integración del software en la herramienta MEDUSA[©].
9. Se han estudiado los resultados obtenidos a partir del software elaborado y se ha comparado su rendimiento con el de otras investigaciones.
10. Por último, se han sintetizado las conclusiones principales y se han identificado las posibles limitaciones del trabajo realizado, junto con una propuesta de futuras mejoras que podrían ampliar su utilidad y rendimiento en contextos reales.

8.3. Conclusiones y aportaciones originales

Tras la ejecución de este TFM y el estudio de cada uno de sus apartados se han extraído las siguientes conclusiones:

1. Las señales neuronales registradas mediante MEG están expuestas a la presencia de artefactos de origen no cerebral, como los de tipo cardíaco, ocular o debidos a la red eléctrica, los cuales pueden interferir en el análisis e interpretación de la actividad cerebral real. Esta contaminación limita la calidad de los datos y puede condicionar negativamente cualquier estudio clínico o investigación neurocientífica que se base en dichas señales.
2. La aplicación de modelos de DL, concretamente una CNN como EEGInception, ha demostrado su utilidad para clasificar componentes ICA contaminadas por artefactos, superando las limitaciones de las técnicas manuales o tradicionales. El método desarrollado no sólo alcanza altos niveles de precisión, sino que además permite detectar errores de etiquetado humano que en condiciones habituales se pasarían por alto.
3. El rendimiento del sistema de clasificación depende en gran medida de varios aspectos técnicos, como la estandarización previa de las señales, el diseño de la arquitectura específica de la CNN y una correcta selección de hiperparámetros durante el entrenamiento.
4. La generación de señales sintéticas ha permitido evaluar la herramienta de forma controlada y ha facilitado la incorporación de artefactos en distintas condiciones y niveles de intensidad sobre actividad neuronal real. Este proceso ha facilitado disponer de un *ground truth* del modelo de IA, posibilitando una evaluación más rigurosa del mismo. Además, estas señales permiten simular diferentes escenarios clínicos o experimentales sin necesidad de nuevas adquisiciones.

5. El software desarrollado ha sido implementado dentro del entorno de análisis neuronal MEDUSA[©], ofreciendo una nueva interfaz para procesar señales MEG contaminadas. Esta permite visualizar tanto las componentes clasificadas como las regiones señaladas como ruidosas, así como recuperar la señal limpia a partir del *score* calculado por el modelo.

Además, en este trabajo se han generado una serie de aportaciones personales que añaden valor al proyecto desarrollado. A continuación, se describen las más relevantes:

1. Se ha desarrollado un sistema basado en IA capaz de detectar, clasificar y eliminar artefactos en registros MEG a partir de componentes ICA, adaptado específicamente a la arquitectura EEG-Inception y ajustado a las necesidades de este trabajo. El modelo alcanzó una sensibilidad superior al 95 % en la detección de componentes artefactuadas en señales reales, lo que permitió mejorar la calidad de la base de datos.
2. Además de la detección, se ha integrado un mecanismo que permite eliminar los tramos ruidosos detectados, aportando una mejora práctica respecto a otros trabajos que se centran solo en la clasificación. Tras aplicar este procedimiento, se consiguió que una parte relevante de las señales inicialmente clasificadas como dudosas pasaran a ser consideradas limpias.
3. Se ha implementado un procedimiento para generar señales sintéticas que permiten simular artefactos de distinta naturaleza y potencia, lo que ha facilitado la evaluación objetiva y controlada del rendimiento del modelo.
4. Se ha diseñado una interfaz visual, intuitiva y funcional que permite tanto la visualización y limpieza de señales dudosas como la generación de señales sintéticas.
5. La herramienta diseñada no queda como un proyecto aislado, sino que se ha incorporado en MEDUSA[©], contribuyendo a la ampliación de esta plataforma de análisis de señales cerebrales e incrementando su accesibilidad para investigadores neurocientíficos.

8.4. Líneas futuras

A continuación, se presentan varias líneas de trabajo futuro de este TFM que podrían servir como base para ampliar y mejorar su utilidad en el futuro, en base a las limitaciones encontradas tras su ejecución.

Por un lado, la mejora en la calidad de la base de datos utilizada se ve influenciada por la diversidad de sujetos y las condiciones de la adquisición de datos. Para mejorar la generalización del modelo y su robustez, sería conveniente incorporar nuevos registros MEG procedentes de distintos entornos clínicos y con mayor variabilidad. Asimismo, aumentar la representación de clases poco frecuentes, como ciertos tipos de artefactos, permitiría un entrenamiento más equilibrado.

Por otro lado, para disminuir la subjetividad y los posibles errores que surgen durante el proceso de etiquetado de las señales neuronales, se podría integrar un sistema de consenso entre varios expertos o emplearse técnicas de etiquetado asistido por IA para mejorar la consistencia. Además, una revisión automatizada del etiquetado permitiría detectar incongruencias y mejorar la calidad del conjunto de datos.

Para poder extender la herramienta diseñada a pacientes con distintas patologías neurológicas, se podría aplicar a otras modalidades de neuroimagen, como EEG o electrocorticografía. Asimismo, sería interesante integrar módulos de diagnóstico asistido o predicción temprana de alteraciones cognitivas para ampliar su utilidad.

En cuanto al modelo de IA empleado, sería interesante comparar su rendimiento con otras arquitecturas avanzadas, como las redes híbridas CNN-RNN. También podrían explorarse métodos no supervisados para detectar artefactos sin necesidad de un etiquetado previo y, a pesar de haber incorporado al sistema ciertos elementos de explicabilidad al mostrar los tramos considerados como artefacto, la transparencia del sistema mejoraría si se empleara alguna técnica formal de explicabilidad de modelos (XAI) (Jiang, et al., 2022).

Además, la generación de señales sintéticas abre un abanico de posibilidades para validar y robustecer modelos de IA en condiciones controladas. En este sentido, se podrían crear diferentes tipos de señales con niveles variables de ruido, o incluso simular patologías específicas. También se podría desarrollar un generador de datos basado en GANs (*Generative Adversarial Networks*) para crear señales más realistas, lo que aumentaría la calidad del entrenamiento y validación (Carle, et al., 2023).

Por último, sería útil implementar en MEDUSA© una interfaz gráfica más completa que permita al usuario visualizar los resultados en tiempo real, editar etiquetas o ajustar algunos parámetros del modelo. Asimismo, resultaría de interés integrar los módulos de preprocesamiento y análisis de la plataforma para conseguir un flujo de trabajo continuo desde la adquisición de datos hasta la interpretación final.

Bibliografía

AlHinai, N. (2020). Introduction to biomedical signal processing and artificial intelligence. In W. Zgallai (Ed.), *Biomedical signal processing and artificial intelligence in healthcare* (pp. 1-28). Academic Press.

Ambardar, A. (1995). *Analog and digital signal processing* (2nd ed.). Boston, MA, USA: PWS Publishing.

Andonie, R. (2019). Hyperparameter optimization in learning systems. *Journal of Membrane Computing*, 1(4), 279-291.

Antona-Cortés, C. (2017). *Herramientas modernas en redes neuronales: la librería Keras*. Trabajo de Fin de Grado, Universidad Autónoma de Madrid, Madrid.

Arslan, A., Şen, B., Çelebi, F. V., Peker, M., & But, A. (2015). A comparison of different classification algorithms for determining the depth of anesthesia level on a new set of attributes. In *2015 International Symposium on Innovations in Intelligent SysTems and Applications (INISTA)* (pp. 1-7). IEEE.

Azzerboni, B., Carpentieri, M., La-Forest, F., & Morabito, F. C. (2004). Neural-ICA and wavelet transform for artifacts removal in surface EMG. *Proceedings of the 2004 IEEE International Joint Conference on Neural Networks*, 4, 3223-3228.

Barbati, G., Porcaro, C., Zappasodi, F., Rossini, P. M., & Tecchio, F. (2004). Optimization of an independent component analysis approach for artifact identification and removal in magnetoencephalographic signals. *Clinical Neurophysiology*, 115 (5), 1220-1232.

Bashashati, A., Fatourechi, M., Ward, R. K., & Birch, G. E. (2007). A survey of signal processing algorithms in brain-computer interfaces based on electrical brain signals. *Journal of Neural Engineering*, 4(2), R32–R57.

Berger, H. (1969). On the electroencephalogram of man. In *The fourteen original reports on the human electroencephalogram* (Journal of the Neurological, 13 ed., Vol. 13). *Electroencephalography and Clinical Neurophysiology*.

Besserve, M., Jerbi, K., Laurent, F., Baillet, S., Martinerie, J., & Garnero, L. (2007). *Classification methods for ongoing EEG and MEG signals*. Biological research.

Betzel, R. F., & Bassett, D. S. (2017). Generative models for network neuroscience: prospects and promise. *Journal of The Royal Society Interface*, 14(136), 20170623. doi:10.1098/rsif.2017.0623

Bitbrain. (2020). *How Deep Learning is changing machine learning AI in EEG data processing*. Retrieved Enero 22, 2025, from <https://www.bitbrain.com/blog/ai-eeg-data-processing>

Blum, S., Jacobsen, N. S., Bleichner, M. G., & Debener, S. (2019). A Riemannian Modification of Artifact Subspace Reconstruction for EEG Artifact Handling. *Frontiers in Human Neuroscience*, 13, 141.

Burgess, R. C. (2020). Recognizing and Correcting MEG Artifacts. *Journal of Clinical Neurophysiology*, 37 (6), 508-517.

Callier, P., & Sandel, O. (2021). Introduction to artificial intelligence. *Actualites Pharmaceutiques*, 60, 18-20.

Carlle, F. P., Hollenbenders, Y., & Reichenbach, A. (2023). Generation of synthetic EEG data for training algorithms supporting the diagnosis of major depressive disorder. *Frontiers in Neuroscience*, 17, 1219133.

Chawla, M. P. (2011). PCA and ICA processing methods for removal of artifacts and noise in electrocardiograms: A survey and comparison. *Applied Soft Computing*, 11 (2), 2216-2226.

Cohen, D. (1968). Magnetoencephalography: evidence of magnetic fields produced by alpharhythm currents. *Science*, 161(3843), 784-786.

Colomer, A., Fuentes, F., Naranjo, V., Guixeres, J. A., & Alcañiz, M. (2016). A Comparison of Physiological Signal Analysis Techniques and Classifiers for Automatic Emotional Evaluation of Audiovisual Contents. *Frontiers in Computational Neuroscience*, 10, 74.

Croce, P., Zappasodi, F., Marzetti, L., Merla, A., Pizzella, V., & Chiarelli, A. M. (2019). Deep convolutional neural networks for feature-less automatic classification of independent components in multi-channel electrophysiological brain recordings. *IEEE Transactions on Biomedical Engineering*, 66(8), 2372-2380.

Ertam, F., & Aydin, G. (2017). Data classification with deep learning using TensorFlow. In *2017 International Conference on Computer Science and Engineering (UBMK)* (pp. 755-758). IEEE.

Fabietti, M., Mahmud, M., Lotfi, A., Averna, A., Gugganmos, D., Nudo, R., & Chiappalone, M. (2020). Neural network-based artifact detection in local field potentials recorded from chronically implanted neural probes. In *International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). IEEE.

Fazel-Rezai, R., Allison, B. Z., Guger, C., Sellers, E. W., Kleih, S. C., & Kübler, A. (2012). P300 brain computer interface: current challenges and emerging trends. *Frontiers in Neuroengineering*, 5, 14.

García-Albea, E. (2017). *Su majestad el cerebro*. Madrid: La Esfera de los Libros.

Garg, A., Popli, R., & Sarao, B. S. (2021). Growth of digitization and its impact on big data analytics. *IOP Conference Series: Materials Science and Engineering*, 1022, 012083.

Garg, P., Davenport, E., Murugesan, G., Wagner, B., Whitlow, C., Maldjian, J., & Montillo, A. (2017). Automatic 1D convolutional neural network-based detection of artifacts in MEG acquired without electrooculography or electrocardiography. In *2017 International Workshop on Pattern Recognition in Neuroimaging (PRNI)* (pp. 1-4). IEEE.

Garg, P., Treacher, A. H., Davenport, E., Godwin, R., Proskovec, A., Bezerra, L. G., & Montillo, A. A. (2021). MEGnet: Automatic ICA-based artifact removal for MEG using spatiotemporal convolutional neural networks. *NeuroImage*, 241, 118402.

Gil Correa, A. (2023). *Detector Automático de Artefactos en Señales Neuronales basado en Técnicas de Inteligencia Artificial*. Trabajo de Fin de Grado, Universidad de Valladolid, Valladolid.

Grossberg, S. (2013). Recurrent neural networks. *Scholarpedia*, 8(2), 1888.

Guan, Y., Yuan, Z., Sun, G., & Cong, J. (2017). FPGA-based accelerator for long short-term memory recurrent neural networks. In *22nd Asia and South Pacific Design Automation Conference (ASP-DAC)* (pp. 629-634). IEEE.

Hamal, A. Q., & Rehman, A. W. (2013). Artifact Processing of Epileptic EEG Signals: An Overview of Different Types of Artifacts. In *International Conference on Advanced Computer Science Applications and Technologies* (pp. 358-361). IEEE.

Hämäläinen, M., Hari, R., Ilmoniemi, R. J., Knuutila, J., & Lounasmaa, O. V. (1993). Magnetoencephalography-theory, instrumentation, and applications to noninvasive studies of the working human brain. *Reviews of Modern Physics*, 65(2), 413.

Hamdan, S., DuBray, K., Treutel, J., Paudyal, R., & Poudel, K. (2023). Reducing MEG interference using machine learning. *Machine Learning with Applications*, 12, 100462.

Hamet, P., & Tremblay, J. (2017). Artificial intelligence in medicine. *Metabolism*, 69, 36-40.

Hari, R., Levänen, S., & Raij, T. (2000). Timing of human cortical functions during cognition: role of MEG. *Trends in Cognitive Sciences*, 4(12), 455-462.

Hirano, K., Nishimura, S., & Mitra, S. (1974). Design of Digital Notch Filters. *IEEE Transactions on Communications*, 22, 964-970.

Hussein, B. M., & Shareef, S. M. (2024). An empirical study on the correlation between early stopping patience and epochs in deep learning. (E. Sciences, Ed.) *ITM Web of Conferences*, 64, 01003.

Hyvärinen, A. (2011). Testing the ICA mixing matrix based on inter-subject or inter-session consistency. *NeuroImage*, 58 (1), 122-136.

ICL. (2024). *Iclabel Tutorial: EEG Independent Component Labeling*. Retrieved Octubre 26, 2024, from <https://labeling.ucsd.edu/tutorial/labels>

Islam, M. (2015). *Artifact characterization, detection and removal from neural signals*. National University of Singapore.

Islam, M. K., Nguyen, A. T., Zhou, Y., & Yang, Z. (2012). Analysis and Processing of In-Vivo Neural Signal for Artifact Detection and Removal. In *2012 5th International Conference on BioMedical Engineering and Informatics* (pp. 437-442). IEEE.

Islam, M., Rastegarnia, A., & Sanei, S. (2021). Signal Artifacts and Techniques for Artifacts and Noise Removal. In M. Ahad, & M. Ahmed, *Signal Processing Techniques for Computational Health Informatics* (pp. 23-80). Cham: Springer International Publishing.

Jafarabadi, M. A., Hajizadeh, E., & Anoshiravan, K. (2015). *ICA or PCA? Which One is More Efficient Method for Artifact Identification in MEG Recordings*. ResearchGate.

Jain, A. K., Mao, J., & Mohiuddin, K. M. (1996). Artificial neural networks: A tutorial. *Computer*, 29(3), 31-44.

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic markets*, 31(3), 685-695.

Janocha, K., & Czarnecki, W. M. (2017). *On loss functions for deep neural networks in classification*. Faculty of Mathematics and Computer Science. Jagiellonian University, Krakow, Poland: arXiv:1702.05659.

Jas, M., Engemann, D. A., Bekhti, Y., Raimondo, F., & Gramfort, A. (2017). Autoreject: Automated artifact rejection for MEG and EEG data. *NeuroImage*, 159, 417-429.

Jiang, X., Bian, G.-B., & Tian, Z. (2019). Removal of Artifacts from EEG Signals: A Review. *Sensors*, 19, 987.

Jiang, Y., Li, X., Luo, H., Yin, S., & Kaynak, O. (2022). Quo vadis artificial intelligence? *Discover Artificial Intelligence*, 2(1), 4.

Junliang, C. (2022). CNN or RNN: Review and experimental comparison on image classification. In *Proceedings of the IEEE 8th International Conference on Computer and Communications (ICCC)* (pp. 1939-1944). IEEE.

Jwo, D. J., Chang, W. Y., & Wu, I. H. (2021). Windowing Techniques, the Welch method for improvement of Power Spectrum Estimation. *Computers, materials & continua*, 67, 3.

Kaplan, A. Y., Fingelkurts, A. A., Fingelkurts, A. A., Borisov, S. V., & Darkhovsky, B. S. (2005). Nonstationary nature of the brain activity as revealed by EEG/MEG: Methodological, practical and conceptual challenges. *Signal Processing*, 85(11), 2190-2212.

Kaya, I. (2019). A brief summary of EEG artifact handling. *Brain-computer interface*, 9.

Khanna, M. (2025). *Classification Problem: Relation between Sensitivity, Specificity and Accuracy*. Retrieved Junio 5, 2025, from <https://www.analyticsvidhya.com/blog/2021/06/classification-problem-relation-between-sensitivity-specificity-and-accuracy/>

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv*, 1412, 6.

Kwakernaak, H., Sivan, R., & Strijbos, R. C. (1991). Modern signals and systems. *NASA STI/Recon Technical Report A*, 91, 11586.

Laguna, S., Gopinath, K., Hoopes, A., Alexander, D. C., Arnold, S. E., Balbastre, Y., & Edlow, B. L. (2024). Synthetic data in generalizable, learning-based neuroimaging. *Imaging Neuroscience*, 2, 1-22.

Lakshmi, K. A., Surling, S. N., & Sheeba, O. (2017). A novel approach for the removal of artifacts in EEG signals. In *International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)* (pp. 2595-2599). IEEE.

Larson, E., & Taulu, S. (2017). Reducing sensor noise in MEG and EEG recordings using oversampled temporal projection. *IEEE Transactions on Biomedical Engineering*, 65 (5), 1002-1013.

Larson, E., & Taulu, S. (2017). The Importance of Properly Compensating for Head Movements During MEG Acquisition Across Different Age Groups. *Brain topography*, 30(2), 172–181.

Lawhern, V. J., Solon, A. J., Waytowich, N. R., Gordon, S. M., Hung, C. P., & Lance, B. J. (2018). EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces. *Journal of Neural Engineering*, 15(5), 056013.

Lee, S. B., Gui, X., Manquen, M., & Hamilton, E. R. (2019). Use of training, validation, and test sets for developing automated classifiers in quantitative ethnography. In *International Conference on Quantitative Ethnography* (Vol. Proceedings 1, pp. 117-127). Cham: Springer International Publishing.

Lee, T. W., Girolami, M., & Sejnowski, T. J. (1999). Independent component analysis using an extended infomax algorithm for mixed subgaussian and supergaussian sources. *Neural computation*, 11 (2), 417-441.

LSL. (2019). *Lab Streaming Layer*. Retrieved Febrero 4, 2025, from <https://labstreaminglayer.org/#/>

Ma, W., Papadakis, M., Tsakmalis, A., Cordy, M., & Traon, Y. L. (2021). Test selection for deep learning systems. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 30 (2), 1-22.

Mahmud, M., Travalin, D., Bertoldo, A., Girardi, S., Maschietto, M., & Vassanelli, S. (2012). An automated classification method for single sweep local field potentials recorded from rat barrel cortex under mechanical whisker stimulation. *Journal of Medical and Biological Engineering*, 32(6), 397-404.

Marblestone, A. H., Wayne, G., & Kording, K. P. (2016). Toward an Integration of Deep Learning and Neuroscience. *Frontiers in Computational Neuroscience*, 10, 215943.

Martínez, V., Thielen, J., Santamaria, E., Pérez, S., Desain, P., & Hornero, R. (2021). Brain–computer interfaces based on code-modulated visual evoked potentials (c-VEP): A literature review. *Journal of Neural Engineering*, 18(6), 061002.

MEDUSA. (2022). *This is MEDUSA®*. Retrieved Febrero 2, 2025, from <https://www.medusabci.com/home/>

MEG, W. (2019). *Artifacts in MEG data*. Retrieved Enero 17, 2025, from https://www.neuro.mcw.edu/meg/index.php/Artifacts_in_MEG_data#:~:text=Artifact%20Handling,What%20is%20an%20artifact%3F,help%20you%20remove%20the%20artifact.

Michel, C. M., & He, B. (2019). EEG source localization. *Handbook of Clinical Neurology*, 160, 85-101.

MNE. (2024). *Generate simulated source data*. Retrieved 01 28, 2025, from https://mne.tools/1.8/auto_examples/simulation/source_simulator.html

Mosher, J. C., & Funke, M. E. (2020). Towards best practices in clinical magnetoencephalography: patient preparation and data acquisition. *Journal of Clinical Neurophysiology*, 37 (6), 498-507.

Mumtaz, W., Rasheed, S., & Irfan, A. (2021). Review of challenges associated with the EEG artifact removal methods. *Biomedical Signal Processing and Control*, 68, 102741.

Muthukumaraswamy, S. D. (2013). High-frequency brain activity and muscle artifacts in MEG/EEG: a review and recommendations. *Frontiers in Human Neuroscience*, 7, 138.

Neuper, C., & Pfurtscheller, G. (2010). Neurofeedback training for BCI control. In *Brain-Computer Interfaces: Revolutionizing Human-Computer Interaction* (pp. 65-78). Springer.

Nitschke, J. B., Miller, G. A., & Cook, E. W. (1998). Digital filtering in EEG/ERP analysis: Some technical and empirical comparisons. *Behavior Research Methods, Instruments, & Computers*, 30, 54-67.

Nixon, M., & Aguado, A. (2019). *Feature extraction and image processing for computer vision* (4th ed.). Academic press.

Novac, O. C., Chirodea, M. C., Novac, C. M., Bizon, N., Oproescu, M., Stan, O. P., & Gordan, C. E. (2022). Analysis of the application efficiency of TensorFlow and PyTorch in convolutional neural network. *Sensors*, 22(22), 8872.

Ono, T., Shindo, K., Kawashima, K., Ota, N., Ito, M., Ota, T., . . . Ushiba, J. (2014). Brain-computer interface with somatosensory feedback improves functional recovery from severe hemiplegia due to chronic stroke. *Frontiers in Neuroengineering*, 7, 19.

O'Shea, K., & Nash, R. (2015). *An introduction to Convolutional Neural Networks*. arXiv:1511.08458.

Pal, M., Roy, R., Basu, J., & Bepari, M. S. (2013). Blind source separation: A review and analysis. In *International Conference Oriental COCOSDA held jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE)* (pp. 1-5). IEEE.

Papadelis, C., AlHilani, M., & Pearl, P. L. (2018). Artifacts in Pediatric and Adult Magnetoencephalography. In W. O. Tatum (Ed.), *Atlas of Artifacts in Clinical Neurophysiology* (pp. 183-204). Springer Publishing Company.

Pion-Tonachini, L., Kreutz-Delgado, K., & Makeig, S. (2019). ICLabel: An automated electroencephalographic independent component classifier, dataset, and website. *NeuroImage*, 198, 181-197.

Proakis, J. G. (2001). *Digital signal processing: principles algorithms and applications* (4th ed.). Pearson Prentice Hall.

Puce, A., & Hämäläinen, M. (2017). A Review of Issues Related to Data Acquisition and Analysis in EEG/MEG Studies. *Brain Sciences*, 7(6), 58.

Reilly, R. B., & Lee, T. C. (2010). Electrograms (ECG, EEG, EMG, EOG). *Technology and Health Care, IOS Press*, 18(6), 443–458.

Renard, Y., Lotte, F., Gibert, G., Congedo, M., Maby, E., Delannoy, V., & Lécuyer, A. (2010). Openvibe: An open-source software platform to design, test, and use brain–computer interfaces in real and virtual environments. *Presence*, 19 (1), 35-53.

Rodríguez-González, V., Gómez, C., Hoshi, H., Shigihara, Y., Hornero, R., & Poza, J. (2021). Exploring the interactions between neurophysiology and cognitive and behavioral changes induced by a non-pharmacological treatment: A network approach. *Frontiers in Aging Neuroscience*, 13, 696174.

Ros-García, A. (2019). *Sistema de percepción de elementos viarios usando técnicas de visión por computador para aplicación en conducción autónoma*. Trabajo de Fin de Máster, Universidad Politécnica de Cartagena, Murcia.

Rouhiainen, L. (2018). *Inteligencia artificial*. Alienta Editorial.

Sanei, S., & Chambers, J. A. (2013). *EEG signal processing*. John Wiley & Sons.

Santamaría, E., Martínez, V., & Hornero, R. (2018). MEDUSA: una nueva herramienta para el desarrollo de sistemas Brain-Computer Interface basada en Python. In *10º Simposio CEA de Bioingeniería, Interfaces Cerebro-Máquina*. UVaDOC.

Santamaría, E., Martínez, V., Marcos, D., Rodríguez, V., Pérez, S., Moreno, S., & Hornero, R. (2023). MEDUSA©: A novel Python-based software ecosystem to accelerate brain-computer interface and cognitive neuroscience research. *Computer Methods and Programs in Biomedicine*, 230, 107357.

Santamaría, E., Martínez, V., Marcos, D., Rodríguez, V., Pérez, S., Moreno, S., & Hornero, R. (2024). *MEDUSA©*. Retrieved Noviembre 1, 2024, from <https://www.medusabci.com/>

Santamaría, E., Martínez, V., Vaquerizo, F., & Hornero, R. (2020). EEG-Inception: A novel deep convolutional neural network for assistive erp-based brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28 (12), 2773-2782.

Schalk, G., McFarland, D. J., Hinterberger, T., Birbaumer, N., & Wolpaw, J. R. (2004). BCI2000: a general-purpose brain-computer interface (BCI) system. *IEEE Transactions on Biomedical Engineering, 51* (6), 1034-1043.

Semmlow, J. (2004). *Biosignal and Medical Image Processing* (2nd ed.). CRC Press.

Sen, D., Mishra, B. B., & Pattnaik, P. K. (2023). A review of the filtering techniques used in EEG signal processing. In *2023 7th International Conference on Trends in Electronics and Informatics (ICOEI)* (pp. 270-277). IEEE.

Shaker, M. M. (2006). EEG waves classifier using wavelet transform and Fourier transform. *Brain, 2*(3), 169-174.

Sharma, S., Sharma, S., & Athaiya, A. (2017). Activation functions in neural networks. *Towards Data Science, 6*(12), 310-316.

Singh, S. P. (2014). Magnetoencephalography: Basic principles. *Annals of Indian Academy of Neurology, 17*(1), S107-S112.

Smith, S. W. (1999). *The scientist and engineer's guide to digital signal processing* (2nd ed.). California Technical Publishing.

Stevens, L. A. (1974). *Exploradores del cerebro*. Barral.

Sweeney, K. (2013). *Motion artifact processing techniques for physiological signals*. Ph. D. Thesis, National University of Ireland Maynooth, Ireland.

Taqi, A. M., Awad, A., Al-Azzo, F., & Milanova, M. (2018). The impact of multi-optimizers and data augmentation on TensorFlow convolutional neural network performance. In *IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)* (pp. 140-145). IEEE.

Team, T. B. (2023). *All about eeg artifacts and filtering tools*. Retrieved Enero 17, 2025, from <https://www.bitbrain.com/blog/eeg-artifacts>

Tibaduiza, D. A., Mujica, L. E., Anaya, M., Rodellar, J., & Güemes, A. (2013). Principal component analysis vs. independent component analysis for damage detection. In *Proceedings 6th European Workshop on Structural Health Monitoring & 1st European Conference On Prognostics and Health Management* (Vol. 2, pp. 3-6).

Urigüen, J. A., & Garcia-Zapirain, B. (2015). EEG artifact removal-state-of-the-art and guidelines. *Journal of Neural Engineering, 12*(3), 031001.

Van Bosse, J. G. (1998). *Signaling in telecommunication networks* (36 ed.). John Wiley & Sons.

Vaziri, A. Y., Makkiabadi, B., & Samadzadehaghdam, N. (2023). EEGg: generating synthetic EEG signals in Matlab environment. *Frontiers in Biomedical Technologies, 10*(3), 370-381.

Vigario, R., & Oja, E. (2008). BSS and ICA in neuroinformatics: from current practices to open challenges. *IEEE reviews in biomedical engineering, 1*, 50-61.

Vrba, J., & Robinson, S. (2001). The effect of environmental noise on magnetometer-and gradiometer-based MEG systems. *Signal*, 1, 0-5.

Vrba, J., & Robinson, S. E. (2001). Signal processing in magnetoencephalography. *Methods*, 25(2), 249-271.

Vu, M. A., Adali, T., Ba, D., Buzsáki, G., Carlson, D., Heller, K., & Dzirasa, K. (2018). A shared vision for machine learning in neuroscience. *Journal of Neuroscience*, 38 (7), 1601-1607.

Walczak, S. (2019). Artificial neural networks. In *Advanced methodologies and technologies in artificial intelligence, computer simulation, and human-computer interaction* (pp. 40-53). IGI Global Scientific Publishing.

Wang, Y., Gao, X., Hong, B., Jia, C., & Gao, S. (2008). Brain-computer interfaces based on visual evoked potentials. *IEEE Engineering in medicine and biology magazine*, 27 (5), 64-71.

Wierzgała, P., Zapała, D., Wojcik, G. M., & Masiak, J. (2018). Most popular signal processing methods in motor-imagery BCI: a review and meta-analysis. *Frontiers in neuroinformatics*, 12, 78.

Wolpaw, J. R. (2013). Brain-computer interfaces. *Handbook of clinical neurology*, 110, 67-74.

Wolpaw, J. R., Millan, J. D., & Ramsey, N. F. (2020). Brain-computer interfaces: definitions and principles. *Handbook of clinical neurology*, 168, 15-23.

Yang, B., Duan, K., Fan, C., Hu, C., & Wang, J. (2018). Automatic ocular artifacts removal in EEG using deep learning. *Biomedical Signal Processing and Control*, 43, 148-158.

Yang, W., Pilozzi, A., & Huang, X. (2021). An Overview of ICA/BSS-Based Application to Alzheimer's Brain Signal Processing. *Biomedicines*, 9, 386.

Zhang, K., Ying, H., Dai, H. N., Li, L., Peng, Y., Guo, K., & Yu, H. (2021). Compacting deep neural networks for Internet of Things: Methods and applications. *IEEE Internet of Things Journal*, 8 (15), 11935-11959.

Zheng, A., & Casari, A. (2018). *Feature engineering for machine learning: principles and techniques for data scientists* (1st ed.). O'Reilly Media, Inc.