



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

Obtención de métricas sobre la actividad cerebral de usuarios a través de señales EEG y Machine Learning

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Herrero Arnedo, Marcos

Tutor/a: Insfrán Pelozo, César Emilio

Cotutor/a: Gaspar Figueiredo, Daniel

CURSO ACADÉMICO: 2022/2023

# Agradecimientos

Quiero aprovechar para agradecer en primer lugar a mi familia por todo el apoyo que me han dado y la paciencia que han tenido a lo largo de los momentos complicados.

Este trabajo no habría sido posible sin la ayuda de mi tutor Emilio y la gente del equipo de investigación, de los que me llevo un recuerdo muy especial. No puedo dejar sin una mención especial a Daniel, mi co-tutor, que me ha ayudado en todo lo que ha podido y más, dando siempre un punto de vista nuevo, aportándome nuevas ideas aun cuando pensaba que estaba en un callejón sin salida y aguantando mis quejas cuando nada parecía funcionar.

No me puedo olvidar de mis compañeros de piso, que han hecho de un piso de Valencia mi casa, siempre dispuestos a ayudar sacándome a cenar después de una tarde sin conseguir nada o con una cerveza en “el bar”.

Por último, quiero hacer una mención especial a Yael, Javi, Delia, Jordi, Carla y Estefy. Las primeras personas a las que salude un 2 de septiembre de 2019 en las jornadas de acogida de la ETSINF se han convertido en personas importantísimas en mi vida. Si nos hubieran dicho en ese momento todas las aventuras nacionales e internacionales que íbamos a vivir en solo 4 años no lo habiéramos creído. Tengo ganas de ver que nos depara el futuro.

# Resumen

El análisis del impacto de las aplicaciones de software en los usuarios es una parte fundamental del proceso de calidad. Con el fin de obtener resultados más precisos y objetivos, se ha introducido la monitorización de datos fisiológicos a los métodos de evaluación de software. En concreto, la monitorización de señales electroencefalográficas han demostrado un gran desempeño reconociendo el estado cognitivo – emocional de los usuarios. Empresas como BitBrain han invertido en estas tecnologías, desarrollando dispositivos capaces de capturar las señales y herramientas de software para su procesado y visualización. Este software permite conocer el estado del usuario a través de cuatro métricas: valencia o atracción, memorización, implicación y carga cognitiva. En este trabajo se pretende crear una herramienta de software que sea capaz de procesar los datos en crudo obtenidos por BitBrain y llevar a cabo el cálculo de las métricas de forma independiente. Esto permitiría crear la base de un proyecto *open-source* donde se pueda aumentar las funcionalidades e integrarlo en otras aplicaciones. Para ello, se ha llevado a cabo un estudio sobre el marco teórico, se han desarrollado procedimientos para el preprocesado, limpieza y disposición de los datos, y se han entrenado modelos de Machine Learning usando como referencia los datos de BitBrain. Como resultado, se ha generado un modelo para cada métrica usando redes neuronales de regresión. Se han probado diferentes configuraciones para el algoritmo, y se han encontrado similitudes en cuanto a precisión entre la mayoría de ellas. Los rangos de error de los modelos se encuentran dentro de un límite aceptable que permite identificar la tendencia en el estado del usuario.

**Palabras clave:** Métricas, EEG, Machine Learning

## **Abstract**

Analysing the impact of software applications in users plays a key role in the quality process. Aiming to obtain more accurate and objective results, monitoring physiological data is being introduced as an alternative to conventional evaluation techniques. Specifically, monitoring EEG signals has proved a great performance at evaluating users' cognitive – emotional state. Companies like BitBrain have invested in these technologies, developing devices capable of capturing raw EEG signals as well as software applications for processing and displaying data. This software provides four metrics about the user's state: valence, memorization, engagement and workload. The aim of this thesis is to create a software tool capable of processing the raw data obtained from BitBrain's software and then calculating the metrics independently. This would set the foundations of an open-source project where new functionalities could be implemented and integrations with other applications would be possible. To achieve this goal, this thesis contains a theoretical background, the procedures to pre-process, clean and lay out the data, and the training of Machine Learning models using BitBrain's data as reference. As a result, a model has been created for each metric. Different configurations for the algorithm have been applied and the accuracy results are similar for the majority of them. The model's errors are below an acceptable range that allows to identify the tendency in the variation of the user's state.

**Key words:** Metrics, EEG, Machine Learning

# Índice de contenidos

## Contenido

1. Introducción.....	2
1.1 Motivación .....	3
1.2 Objetivos.....	4
1.3 Contexto .....	5
1.4 Estructura del documento .....	5
2. Fundamentos teóricos.....	7
2.1 Dispositivos y software de BitBrain.....	7
2.1.1 Casco Diadem.....	7
2.1.2. Software.....	10
2.2 Datos en crudo .....	13
2.2.1 Origen y naturaleza de las ondas. ....	13
2.2.2 Propiedades de las ondas. ....	14
2.3 Métricas.....	16
2.3.1 Carga cognitiva .....	17
2.3.2 Memorización .....	19
2.3.3 Implicación .....	20
2.3.4 Valencia.....	21
2.4 Pre-procesamiento de los datos .....	22
2.3.1 Filtros de banda.....	23
2.3.2 Análisis de componentes independientes (ICA).....	23
2.5 Machine Learning.....	25
3. Diseño e implementación de la herramienta .....	30
3.1 Limpieza de datos.....	31
3.2 Disposición de los datos .....	34
3.3 Modelo de Machine Learning .....	35
4. Trabajos relacionados .....	37
5. Validación. ....	38
6. Conclusión .....	41
7. Trabajo futuro .....	42
8. Referencias .....	44

# 1. Introducción

El uso de aplicaciones de software se está integrando en cada vez más ámbitos de nuestra sociedad. Debido a la constante y creciente digitalización de la mayoría de los procesos, el uso de aplicaciones se ha visto incrementado en el ámbito personal a través del uso cotidiano de aplicaciones de diversa índole, como por ejemplo comercios online, entretenimiento, alimentación, viajes, servicios, etc.

Asimismo, el proceso de digitalización ha transformado profundamente la industria. Una gran parte de las actividades que se realizaban manualmente son ahora llevadas a cabo usando un software especializado. En este contexto donde el peso del software es cada vez mayor, la importancia de desarrollar aplicaciones de calidad y que se adapten a la variedad de necesidades que pueden tener los usuarios dependiendo de sus características.

Existe un gran abanico de métodos para evaluar la interacción entre un usuario y la aplicación. Estos métodos abarcan desde evaluaciones de la aplicación por parte de expertos, hasta evaluaciones que implican a los usuarios finales a través de encuestas subjetivas o a través de aspectos fisiológicos.

Por un lado, los resultados obtenidos mediante las evaluaciones de expertos y encuestas a usuarios finales pueden estar alterados por diferentes factores. Por ejemplo, la evaluación de una interfaz a través de un formulario puede verse afectada por el subconsciente del sujeto, recuerdos inexactos de la interacción con las interfaces, entre otros.

Con el fin de obtener mediciones más precisas sobre el estado del usuario, surge como alternativa la monitorización de datos fisiológicos del sujeto mientras hace uso de la aplicación para estudiar el impacto de diferentes eventos en el usuario.

A través de diversos estudios, se ha conseguido vincular determinadas actividades cerebrales en diferentes áreas del cerebro con un tipo de reacciones cognitivo – emocionales concretas. Esto ha permitido definir unas métricas que informan del estado cognitivo-emocional del usuario en cada momento, como pueden ser la atracción, el esfuerzo cognitivo, la memorización o la implicación.

De esta forma, empresas de neurotecnología como BitBrain [1] han desarrollado herramientas precisas pero costosas (con licencia anual de varios miles de euros) para medir las señales electroencefalográficas (EEG) haciendo uso de un casco neuronal. Además, estas herramientas tienen funcionalidades muy limitadas y se requiere hacer un post-procesamiento de datos para obtener una mejor interpretación de las distintas métricas.

En este trabajo se pretende proporcionar una alternativa *open source* al software creado por esta empresa, con el objetivo de reducir costes y permitir la implementación de funcionalidades adicionales que actualmente no son proporcionadas.

Para conseguirlo, se ha planteado el uso de Machine Learning (ML). Se pretende usar los datos en crudo obtenidos con el casco Diadem de BitBrain, y usar las métricas generadas por el software usando dichos datos para entrenar modelos para cada métrica.

## **1.1 Motivación**

Este proyecto se presenta como un reto en diferentes niveles. Por un lado, proporciona la oportunidad de trabajar en un ambiente multidisciplinar, ya que durante las diferentes fases del proyecto se abordan tareas tan dispares como es el preprocesado de datos, manipulación de señales, investigación en el campo de la biopsicología y neurociencia, entrenamiento de modelos de ML y el desarrollo de software con librerías *open source*.

Esto supone un desafío y al mismo tiempo una gran motivación, puesto que requiere que me introduzca en campos de los cuales no tengo conocimientos previos y, a través de la investigación y de forma autodidacta, encuentre soluciones para los objetivos marcados.

Otro aliciente de esta propuesta ha sido el uso del Machine Learning. Especialmente en los últimos años, la inteligencia artificial y el Machine Learning han experimentado un gran auge. El uso de estas herramientas se está extendiendo a cada vez más ámbitos y con nuevas aplicaciones. Con estas tecnologías consolidándose como una pieza clave en el futuro de la informática,

este trabajo de fin de grado se presenta como una oportunidad para iniciarme en el mundo del Machine Learning y adquirir experiencia en un entorno ideal para ello, puesto que su funcionamiento nos permite abordar todos los requerimientos del proyecto, uno de los cuales es tratar la ingente cantidad de datos de la que se posee.

Como conclusión, espero como resultado de este trabajo de fin de grado crear una aplicación que aborde los diferentes procesos de forma eficiente, con un código mantenible el cual permita añadir nuevas funcionalidades en un futuro de forma sencilla y que proporcione unos resultados precisos. A nivel personal, espero aprender nuevas tecnologías que me permitan ampliar mis conocimientos en diferentes campos, al mismo tiempo que espero desarrollar aptitudes como el aprendizaje autónomo, habilidades de investigación y resolución de problemas.

## **1.2 Objetivos**

Los objetivos principales de este proyecto son:

- Redactar documentación sobre el marco teórico que sirva como base para el desarrollo de futuros proyectos.
- Crear un proceso para la limpieza y disposición de los datos independiente y reutilizable.
- Generar un modelo de ML por cada métrica que realice predicciones según los datos en cada instante.

Para ello, se plantean una serie de tareas específicas:

1. Estudio del estado del arte para establecer la relación entre la actividad cerebral en diversas zonas y la magnitud de cada métrica.
2. Pre-procesado de los datos usando la librería MNE (paquete open source de Python para el análisis de datos neurofisiológicos) con el propósito de aislar las frecuencias de interés, así como para eliminar el ruido emitido por diferentes agentes (ruido ocular/parpadeo).
3. Filtrado y preparación de datos. Es necesario aislar los atributos que van a ser utilizados por cada modelo de ML siguiendo lo aprendido en el punto

- 1, por lo que se deberán filtrar según la métrica objetivo, así como por los sensores y frecuencias implicadas.
4. Entrenamiento de los modelos. Para cada métrica se diseñará un modelo que se centre en los datos relevantes, y que será entrenado utilizando tanto los datos de estudios previos como de posibles futuros experimentos.
5. Desarrollo de la aplicación que integre los modelos y permita al usuario subir datos en crudo para obtener las métricas, así como visualizar estas métricas a través de diferentes representaciones.
6. Diseño y ejecución de un estudio comparativo con datos de usuarios interactuando con la interfaz de usuario de una aplicación software.

### **1.3 Contexto**

Este trabajo de fin de grado se ha desarrollado en el contexto del grupo de investigación de Ingeniería del Software y Sistemas de Información (ISSI) del Instituto Universitario Mixto Tecnológico de Informática (ITI) de la Universitat Politècnica de València (UPV).

En particular, este trabajo ha contribuido a los resultados del Proyecto “AKILA: User Interface Adaptation through User-Experience-based Reinforcement Learning”, de la convocatoria de Subvenciones para grupos de investigación consolidados (CIAICO/2021/303) financiado por la Generalitat Valenciana y por el Fondo Social Europeo. IP: Silvia Abrahao. De enero del 2022 a diciembre de 2024.

### **1.4 Estructura del documento**

El contenido de este trabajo está distribuido según la siguiente estructura:

- Capítulo 2: Fundamentos teóricos. En este capítulo se explican los conceptos teóricos necesarios para la comprensión del trabajo.
- Capítulo 3: Diseño de la herramienta: En este capítulo se desarrolla la implementación de la herramienta y como se han aplicado los conceptos teóricos a lo largo del proceso.

- Capítulo 4: Trabajos relacionados. En este capítulo se analizan otros trabajos hechos en esta área y se comparan con la propuesta de este trabajo.
- Capítulo 5: Validación de los modelos. En este capítulo se comprueba la precisión de los modelos y se hace una valoración.
- Capítulo 6. Conclusión. En este capítulo se hace un análisis de lo obtenido en este trabajo y el cumplimiento de los objetivos iniciales.
- Capítulo 7. Trabajo futuro. En este capítulo se exponen las posibilidades para continuar con este trabajo y las herramientas que se podrían conseguir.

## **2. Fundamentos teóricos**

En esta sección se profundizará en los diferentes aspectos teóricos que son utilizados para el desarrollo del proyecto.

En primer lugar, se explicará el ecosistema de tecnologías de BitBrain, incluyéndose en ellas el casco neuronal Diadem, el cual es utilizado para la extracción de datos, y el software desarrollado por BitBrain para el tratamiento de los datos en crudo.

En segundo lugar, se desarrollará la naturaleza de los datos sobre los que se va a trabajar, procedentes de forma directa del casco Diadem y sin ningún tipo de procesado, a los que nos referiremos como datos en crudo.

Después, se describirán las métricas con las que se trabaja, definiéndolas y usando la literatura relacionada para explicar la relación de los eventos cerebrales con las variaciones en cada una.

Posteriormente, se tratará el proceso de limpieza de los datos en crudo, su importancia, los tipos de ruido y los pasos que se han llevado a cabo.

Finalmente, se explicarán los fundamentos del Machine Learning, su funcionamiento y los algoritmos utilizados en este trabajo.

### **2.1 Dispositivos y software de BitBrain.**

#### **2.1.1 Casco Diadem**

Los datos usados en esta tesis han sido obtenidos usando el casco de electroencefalograma diseñado por la empresa BitBrain. Este es un casco de sensores secos, debido a que no necesita de la aplicación de geles para su funcionamiento, y consta de 12 sensores, los cuales están ubicados de forma simétrica en las áreas pre-frontal, frontal, parietal y occipital.

Los sensores están ubicados siguiendo el sistema internacional 10-10 para la localización de sensores EEG [2], el cual se usa para estandarizar la posición de los sensores y así poder universalizar los resultados de los experimentos.



Figura 1 Casco Diadem

Los sensores utilizados son el AF7, Fp1, Fp2, AF8, F3, F4, P3, P4, PO7, O1, O2 y PO8. Gráficamente, se puede ver la localización en el cuero cabelludo de estos sensores en la Figura 2 y Figura 3. Cada sensor recoge los datos de un punto concreto midiendo el potencial eléctrico y los digitaliza a través de su canal asignado.

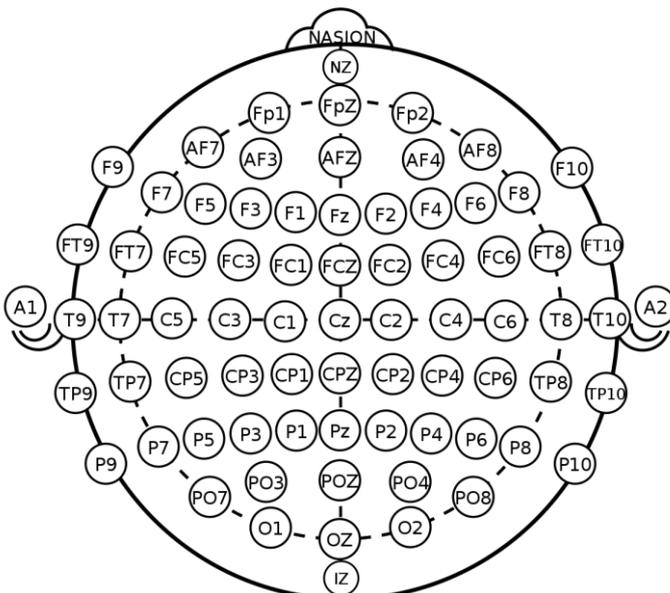


Figura 2. Sistema 10-10 para la localización de electrodos EEG

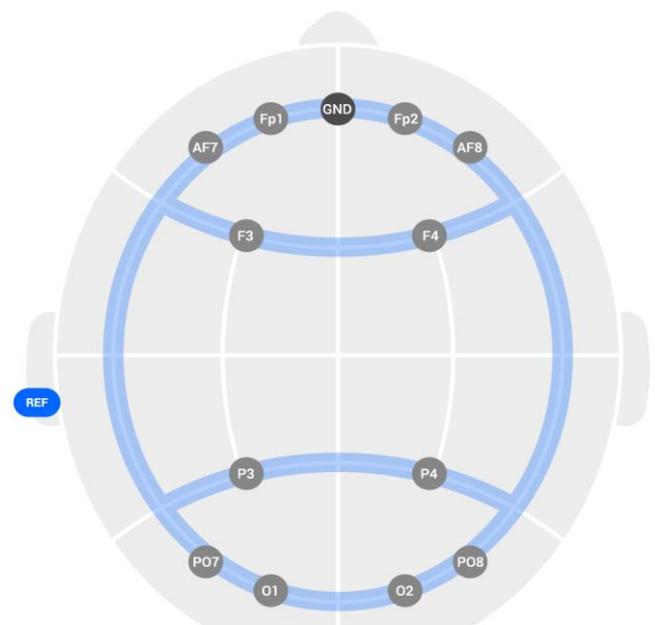


Figura 3. Sensores ubicados en el casco Diadem

El casco utiliza la técnica del referenciado para obtener el potencial eléctrico de los sensores. Esta técnica consiste en determinar un punto de la cabeza como referencia y extraer el potencial eléctrico en comparación a dicho punto. En este caso, se utiliza como punto de referencia el lóbulo del oído izquierdo, al cual se conecta una pinza (punto REF en la Figura 3). Usando librerías especializadas como MNE, la que se explicará más adelante, es posible re-referenciar los potenciales, de forma que se selecciona un nuevo punto de referencia y se computan los potenciales respecto a él.

Los datos extraídos por el casco a través de un amplificador EEG. El amplificador posee una resolución de 24 bits y una frecuencia de muestreo de 256Hz. Respecto al almacenamiento, permite introducir una microSD para guardar los datos en crudo, o bien permite la opción de conectar un dispositivo Bluetooth, permitiendo hasta 8 horas de grabación de forma continua. El uso de este casco con un amplificador Bluetooth permite que se hagan experimentos con libertad de movimiento y con un nivel de intrusividad bajo para el usuario final.

Estos datos en crudo se pueden exportar y leer en archivos CSV, los cuales constan de 15 columnas y tantas filas como registros se hayan guardado en una sesión. Las 12 primeras columnas hacen referencia a los sensores. Por defecto, cada sensor lleva asignado arbitrariamente un canal. A estos doce sensores se le suman otras tres columnas: la decimotercera contiene el número de secuencia, la decimocuarta corresponde a la marca temporal y la última hace referencia a la marca temporal corregida usando el cómputo de tiempo inicial. Concretamente, los valores asociados a sensores son números decimales que representan los microvoltios medidos en cada sensor, mientras que los tres últimos son números enteros que sirven de apoyo para el manejo secuencial de los datos.

Otra funcionalidad que también está disponible con el casco Diadem es la obtención en tiempo real de los datos en crudo. BitBrain proporciona un *Software Development Kit* (SDK) que permite recibir los datos en una aplicación directamente desde el casco al tiempo que son obtenidos, lo cual abre la puerta

a poder desarrollar aplicaciones que puedan dar *feedback* sobre el usuario mientras se lleva a cabo el experimento.

### **2.1.2. Software**

BitBrain pone a nuestra disposición dos aplicaciones de software y un servicio de tratamiento de datos para llevar a cabo al completo todo el proceso. En primer lugar, proporcionan una aplicación para la definición de experimentos y captura de datos en crudo. En segundo lugar proporcionan un servicio que usará los datos capturados por la aplicación anterior y los procesará para obtener métricas. Finalmente, proporciona otra aplicación para visualizar dichas métricas. Vemos a continuación cada una de las aplicaciones y servicio que nos ofrece.

La primera de ellas es SennsLab. SennsLab es la herramienta utilizada en el laboratorio a la hora de realizar los experimentos, y que posteriormente proporcionará los datos en crudo. SennsLab no solo proporciona un software para la recogida de datos, si no que permite diseñar y ejecutar experimentos personalizados. Los experimentos poseen un alto grado de configurabilidad que facilita posteriormente el análisis de los datos. De esta forma, nos permite crear perfiles para los participantes y asignarles variables de segmentación (género, edad, país de origen, etc.).

A un experimento se le pueden asignar los componentes de hardware que se van a utilizar para la recogida de datos. Dentro de la gama de componentes de compatibles encontramos los de Biosensing, como el Ring de BitBrain; los de EEG, como el Diadem o los cascos genéricos configurables y los de Eye-tracking, como las Tobii Pro Glasses. SennsLab sincronizará y capturará datos de los dispositivos configurados de forma simultánea todos los dispositivos asociados.

Respecto a la estructura del experimento, SennsLab ofrece tipos de plantilla variados. Las plantillas pueden presentar imágenes o contenido audiovisual, y estos pueden ser pasivos, si no se espera una respuesta por parte del usuario, o activos, donde el sujeto debe introducir algún tipo de respuesta o *feedback*. Asimismo, es posible crear experimentos de interacción libre, en la que

podremos personalizar con total libertad si usar videos, imágenes o dejar que los participantes de los experimentos interactúen libremente con el dispositivo (ej., consola, ordenador).

SennsLab incorpora un proceso de calibración cuya importancia será detallada en el apartado 2.4. Este proceso consta de varias fases:

- Familiarización: Consta de una tarea de 30 segundos de duración en la que se pide al sujeto que realice restas de 7 en 7 a raíz de un número inicial. El objetivo de esta fase es familiarizar al usuario con el formato de la fase de calibración, por lo que no se registra ningún dato.
- Lavado 1: Durante esta fase el sujeto mantiene los ojos cerrados durante dos minutos. Esto permite eliminar rastros de cualquier respuesta emocional de fases previas para evitar que interfieran en mediciones posteriores. Además, proporciona una referencia de la actividad del sujeto durante un estado de reposo.
- Ojos abiertos: El participante permanece dos minutos con los ojos abiertos. Esta fase proporciona una medición de un estado de reposo con mayor actividad que el lavado.
- Calibración: El objetivo de esta fase es inducir en el sujeto un estado de estrés controlado leve a través de una tarea con carga cognitiva moderada. Al igual que en la fase de familiarización, se le pedirá al participante que realice restas de 7 en 7 consecutivas partiendo de un número generado por SennsLab. En este caso, el número será más complejo que en la fase de familiarización para aumentar la dificultad.
- A continuación, tendría lugar el experimento definido anteriormente.

Al finalizar el experimento, SennsLab permite descargar los datos en crudo en formato CSV para que se pueda tratar este tipo de datos de forma independiente.

Para procesar estos datos, Bitbrain proporciona un servicio llamado SennsCloud, una herramienta en la nube donde se pueden enviar los datos en crudo recogidos para su procesado y cómputo de distintas métricas.

Finalmente, para poder visualizar y analizar las métricas obtenidas, Bitbrain proporciona la aplicación de SennsMetrics. Esta aplicación muestra las métricas obtenidas a través de SennsCloud en gráficos temporales, que permiten analizar

las variaciones a lo largo del experimento. Ofrece a su vez la opción de visualizar las métricas en base a una serie de eventos definidos durante el experimento, que permite diferenciar como un mismo evento ha impactado en varios participantes. Asimismo, permite llevar a cabo un análisis más profundo de la variación de las métricas agrupando a los participantes según los rasgos que se han especificado durante la recogida de datos (sexo, edad, país de origen, etc.).

SennsMetrics permite exportar los datos obtenidos en formato de hoja de cálculo, donde se especifican los valores de las métricas en orden temporal, así como los eventos que han tenido lugar en cada instante. Los datos exportados de SennsMetrics tienen una frecuencia de 256Hz, es decir, las métricas tienen 256 mediciones en cada segundo.

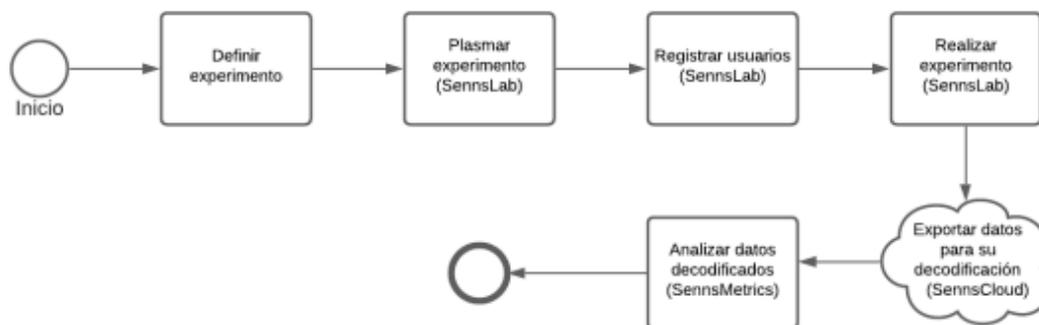


Figura 4. Flujo de trabajo del software de BitBrain

El formato de las métricas tiene una peculiaridad, y es que las muestras se agrupan en grupos de ocho valores. Es decir, ocho valores consecutivos tienen el mismo valor para las métricas. Esto se debe a que BitBrain utiliza los valores obtenidos durante esas ocho muestras para computar el valor de las métricas, ya que es necesaria cierta continuidad para poder obtener los valores

## 2.2 Datos en crudo

En este apartado se va a tratar las propiedades de los datos en crudo recogidos por el casco Diadem.

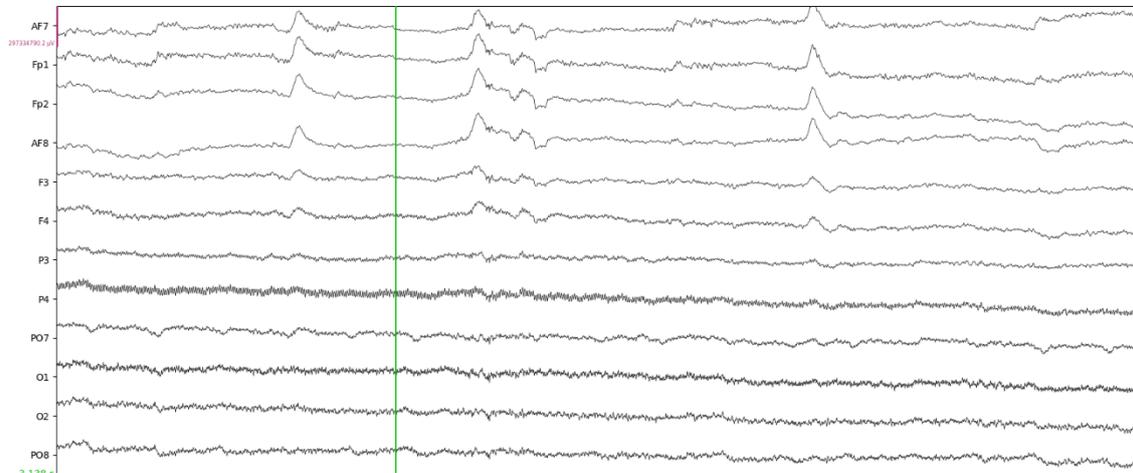


Figura 5. Datos en bruto obtenidos del casco Diadem

### 2.2.1 Origen y naturaleza de las ondas.

Las neuronas se comunican entre sí a través de impulsos eléctricos creando una red. La activación simultánea de conjuntos grandes de neuronas puede dar lugar a patrones neuronales. Estos patrones reflejan la actividad cerebral y pueden servir para interpretar el estado cognitivo, emocional y fisiológico de un individuo.

Esta actividad cerebral es medible gracias a través de las señales electroencefalográficas. Las señales EEG son la representación gráfica de las ondas cerebrales, cuya amplitud refleja el potencial eléctrico en un punto del cerebro.

Las señales electroencefalográficas son la representación gráfica de las ondas cerebrales. Son obtenidas usando las ondas cerebrales que emite constantemente el cerebro. Las ondas cerebrales son ondas cuya amplitud reflejan el potencial eléctrico en un punto del cerebro debido a la comunicación interneuronal.

Las neuronas se comunican entre sí a través de impulsos eléctricos creando una red. La activación simultánea de conjuntos grandes de neuronas puede dar lugar

a patrones neuronales. Estos patrones reflejan la actividad cerebral y pueden servir para interpretar el estado cognitivo, emocional y fisiológico de un individuo.

### 2.2.2 Propiedades de las ondas.

Las ondas cerebrales, como cualquier tipo de onda, son determinadas por una serie de valores físicos (amplitud, frecuencia, etc), los cuales en este caso son de gran relevancia debido a que proporcionan información específica sobre la actividad cerebral.

La amplitud de la onda representa el valor máximo que va a alcanzar, y por lo tanto restringe el rango en el que se van a encontrar los valores. La amplitud de las ondas cerebrales se mide en micro voltios ( $\mu\text{V}$ ).

La frecuencia es la variable que más relevancia tiene a la hora de trabajar con ondas cerebrales [3]. Se define como la inversa del periodo (T). El periodo es el tiempo que tarda una onda en completar una oscilación, mientras que la frecuencia es las oscilaciones que completa la onda por unidad de tiempo.

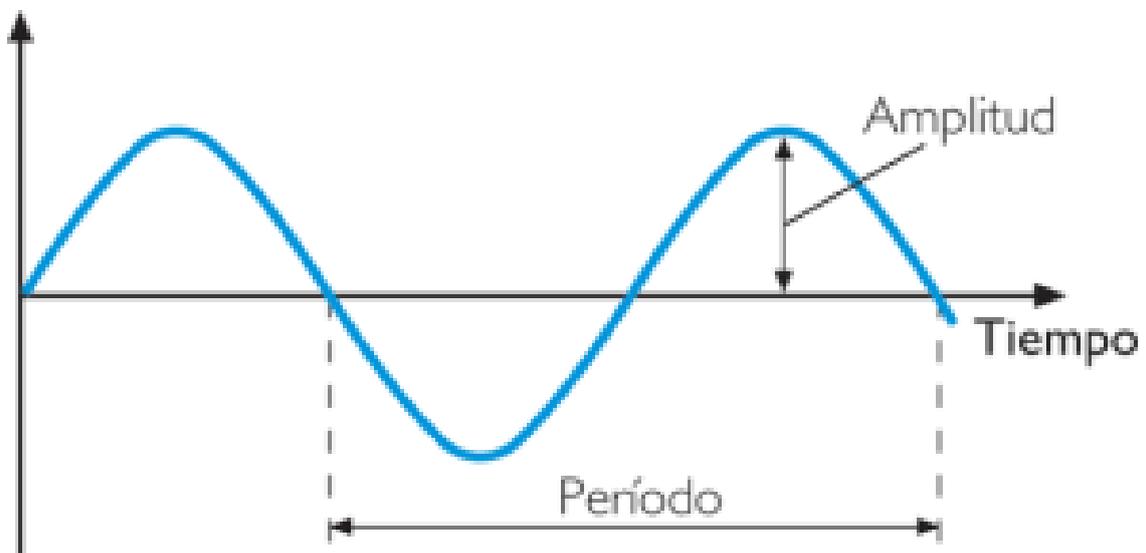


Figura 6. Propiedades de una onda

En el análisis de señales EEG, se hace una división del espectro de la onda en rangos. Los rangos más comunes son nombrados usando una letra griega, entre ellos encontramos: la **delta** (0.5 a 4Hz), la **theta** (4 a 8Hz), la **alpha** (8 a 12Hz) y

la **beta** (12 a 30Hz) [Figura 7]. En los rangos superiores a la beta encontramos frecuencias en las cuales diversos estudios han encontrado relación con fenómenos como la epilepsia [3], pero que no tienen una relevancia significativa con este estudio, por lo que serán eliminadas. Por otra parte, las frecuencias menores a 1 Hz son eliminadas utilizando un filtro de paso alto debido a que están contaminadas por el ruido generado por el *offset* o desplazamiento.

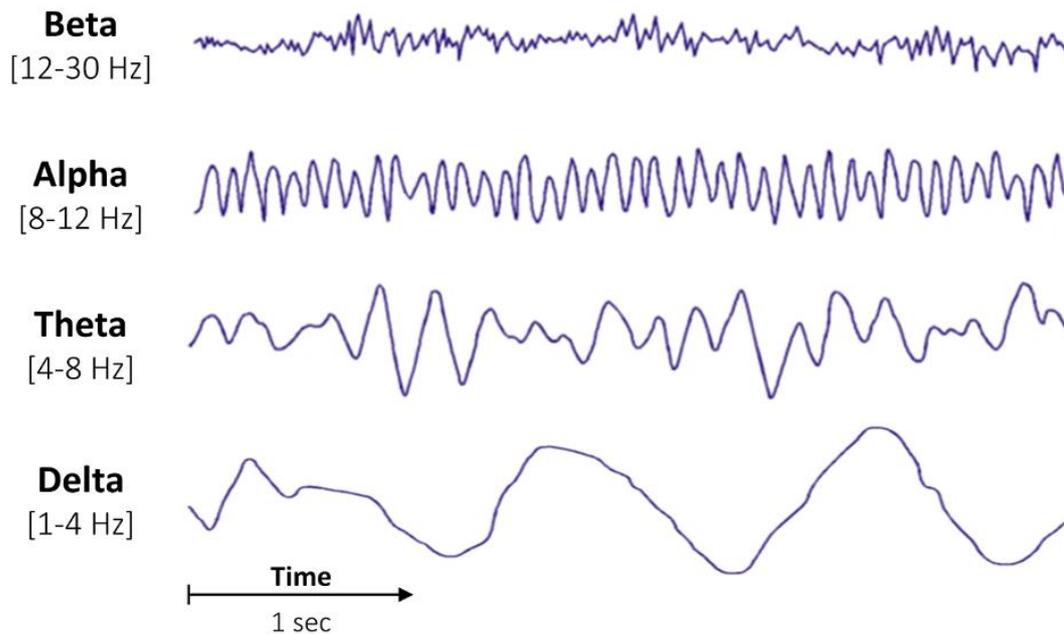


Figura 7 Frecuencias en señales EEG

Estos rangos están asociados con una actividad cerebral en específico, por lo que normalmente son separadas y tratadas de forma particular dependiendo del objetivo del estudio.

Es necesario remarcar que los rangos de frecuencias son relativos, y que pueden verse afectados por diferentes factores como la edad, actividad física y mental, o variables biológicas y relacionadas con el entorno.

Los campos con los que está asociada cada frecuencia son:

- **Delta:** Las ondas delta son las de menor frecuencia, pero de mayor amplitud de onda, y por lo tanto las que presentan los mayores rangos de valores. Están relacionadas con estados de profunda relajación en el individuo, como la fase 3 NREM del sueño. La mayor concentración de actividad delta se encuentra en las zonas frontales.

- **Theta:** Las ondas theta están asociadas a un estado de relajación más activa que la delta, como en la meditación o sueño ligero. A su vez, están asociadas con procesos creativos.
- **Alpha:** Las ondas alpha están presentes sobretodo en la zona del córtex visual. Están relacionadas con estados de calma, y se ven incrementadas cuando el sujeto tiene los ojos cerrados. Asimismo, su actividad aumenta cuando el sujeto es expuesto a estímulos considerados placenteros.
- **Beta:** Las ondas beta están presentes principalmente en las zonas frontal y central del cerebro. Están asociadas con los procesos activos del cerebro, los que requieren de cierta carga cognitiva y lógica, y con los estados de alerta y vigilia.

Es importante mencionar que la interacción entre los diferentes tipos de ondas es altamente compleja, y no puede ser reducida únicamente a lo explicado en este apartado. Para ello, se hace uso de los estudios llevados a cabo en este campo con el objetivo de establecer la relación entre estas ondas y las reacciones cognitivo – emocionales de un usuario, los cuales serán explicados en próximos apartados.

## 2.3 Métricas

Las métricas son indicadores que expresan numéricamente cualidades del estado cognitivo–emocional de un individuo. Están formadas por una escala de valores que normalmente se encuentra entre 0 y 100. Dado que los indicadores cognitivo-emocionales pueden llegar a ser tan abstractos, relativos y personales, se necesita una serie de mediciones que establezcan una referencia para cada persona. Para ello, se utiliza el proceso de calibración detallado en la Sección 2.1.2.

La interacción entre los procesos neuronales es extremadamente compleja, además es muy particular para cada ser humano. De esta forma, un mismo evento puede provocar una actividad muy intensa en una zona del cerebro de un determinado individuo, y una más débil en esa misma zona de otro sujeto. Ambos pueden estar experimentando la misma reacción a nivel cognitivo-emocional,

manifestando mismos valores de una misma métrica, pero debido a las particularidades de su cerebro serían detectadas como diferentes por una aplicación sin calibrar.

El proceso de calibración nos permite establecer unos umbrales que son relativos a cada sujeto. Así se puede recoger los valores de actividad cerebral para una situación de relajación absoluta y establecerlos como el valor mínimo relativo para esa persona, al cual se le otorgará una puntuación de 0 en la métrica. De la misma forma, al someter al sujeto a una situación de estrés mental, se pueden obtener los valores en un estado de alta carga cognitiva y asignarlos al 100 relativo.

Gracias a este proceso, las métricas se encuentran en unos valores relativos personalizados para cada usuario que oscilan entre el 0 y el 100. Sin embargo, es posible encontrar valores menores que 0 y mayores que 100. Esto se debe a que, al ser valores relativos, es posible llegar a situaciones más y menos estimulantes que las propuestas en el proceso de calibración. Por lo que si el usuario es sometido a una carga cognitiva mayor que la del proceso de calibración, los valores superaran el 100 y viceversa.

Las métricas que se van a utilizar son: valencia, carga cognitiva, implicación y memorización. Existen más métricas en el estudio de la actividad cerebral, como el impacto o la excitación, pero no están incluidos en el software de BitBrain por lo que no serán tenidos en cuenta.

### **2.3.1 Carga cognitiva**

La carga cognitiva se define como el esfuerzo cognitivo que tiene que realizar un usuario para resolver o completar una tarea. La carga cognitiva depende en gran medida de la capacidad cognitiva de cada persona. Una persona con una gran capacidad cognitiva experimentará una carga cognitiva baja realizando la misma tarea con la que una persona de menor capacidad cognitiva experimentará una gran carga cognitiva.

La escala de esta métrica está comprendida entre los valores 0 y 100, siendo 0 el valor para carga cognitiva nula y 100 el valor para una carga cognitiva máxima.

Una carga cognitiva demasiado alta puede dar lugar a que el usuario se bloquee realizando la tarea, o incluso llegue a abandonarla. Asimismo, una carga cognitiva baja puede permitir al usuario destinar recursos a realizar otras tareas en segundo plano, pero también puede causarle aburrimiento y desinterés.

Existen diferentes tipos de carga cognitiva:

- Carga intrínseca: carga que está directamente relacionada con la tarea que se está ejecutando. Viene determinada por los recursos necesarios de forma inherente de la tarea.
- Carga extrínseca: está asociada a los elementos externos a la tarea que pueden incrementar la dificultad de la misma añadiendo una carga adicional.
- Carga relacionada: carga relacionada con el propio esfuerzo del aprendizaje, es decir, asimilar los conceptos que se están aplicando para resolver la tarea y trasladarlos a la memoria a largo plazo.

Una situación con una gran carga cognitiva intrínseca sería una tarea de gran dificultad, como cálculos matemáticos complejos. Se podría aumentar la carga extrínseca de esta situación añadiendo ruido a la forma en la que se presentan estos números, incrementando el esfuerzo que el usuario tiene que hacer para reconocerlos. La carga relacionada de esta tarea dependerá de la familiaridad del sujeto con estos cálculos, puesto que si el sujeto ha automatizado el proceso de cálculo la carga será baja, mientras que si está aprendiendo a realizarlos la carga será mayor.

Es importante tener en cuenta los distintos tipos de carga a la hora de elegir una tarea y evaluar los resultados obtenidos; ya que, dependiendo de la tarea, las condiciones en las que se plantea y el usuario; se verá favorecido un tipo de carga u otro.

Situaciones que podrían suponer una gran carga cognitiva son realizar varias tareas a la vez, la resolución de problemas, cálculos, puzles, etc.

La relación entre la carga cognitiva y la actividad cerebral se ve reflejada en las ondas alfa y theta [4][5][6][7]. Además, se manifiesta a través de dos fenómenos: el incremento puntual del potencial de ondas alfa acompañado del decremento de potencial en las ondas theta, y un decremento de fase de las ondas alfa e incremento de las ondas theta.

### **2.3.2 Memorización**

La memorización es el proceso de codificar, almacenar y recuperar la información. Esta métrica pretende medir el esfuerzo que se está realizando a raíz del proceso de memorización.

Esta métrica usará una escala comprendida entre los valores 0 y 100. El valor 0 representará el estado en el que el individuo no se encuentre memorizando ninguna información, y el valor 100 representará un alto requerimiento de memorización.

Se distinguen dos tipos de memoria. La memoria a corto plazo y la memoria a largo plazo. La memoria a largo plazo es aquella que almacena información de forma persistente y es casi ilimitada, lo que nos permite almacenar recuerdos y conocimientos resistentes al olvido.

Sin embargo, la memoria que va a medir esta métrica es la memoria a corto plazo o memoria de trabajo. Esta memoria permite retener información recientemente adquirida durante periodos cortos de tiempo. La capacidad es más limitada que la de largo plazo, y se ve afectada por interferencias causadas por nueva información o distracciones. Es posible transferir la información de la memoria de corto plazo a la de largo plazo a través de la repetición o técnicas mnemotécnicas.

De esta forma, las tareas que van a incrementar la memorización son aquellas en las que se requiera recordar información para la realización de la misma. Por ejemplo, recordar una combinación numérica o una serie de pasos a seguir.

Diversos estudios han tratado de encontrar patrones en la actividad cerebral que se relacionen con el proceso de memorización [8][9][10][11][12]. Pese a los diferentes enfoques, la mayoría coinciden en que el incremento en la actividad

de ondas theta está estrechamente ligada a un aumento de los recursos destinados a la retención de información. Esta actividad se concentra en las zonas del córtex frontal y el lóbulo temporal derecho.

### **2.3.3 Implicación**

La implicación representa el interés o entusiasmo que un individuo experimenta durante una situación o tarea. Es un estado mental altamente deseable en un usuario que se encuentra realizando una tarea, ya que una alta implicación aumentará el nivel de concentración y por lo tanto la productividad así como la capacidad de completarla exitosamente.

La implicación puede verse incrementada por diversos factores. La motivación intrínseca de la actividad es un claro aliciente, dado que cuanto más estimulante y atractiva sea para el sujeto, más interés mostrará en completarla independientemente de cualquier incentivo externo. Otra forma de incrementar la implicación es a través de la motivación externa, que puede ser introducida con recompensas por completar la tarea u otros métodos.

Ámbitos como la educación han probado la importancia de esta métrica. Principalmente en edades tempranas, la implicación ha demostrado estar muy ligada a la capacidad de aprendizaje y ser necesaria para mantener la atención. Ser capaces de conocer el grado de implicación que una tarea es capaz de generar en un niño resulta clave a la hora de diseñar un plan de aprendizaje que resulte atractivo y eficiente.

Esta métrica usará una escala comprendida entre los valores 0 y 100. El valor 0 representará el estado en el que el individuo no esté prestando atención a la tarea, y el valor 100 representará el estado en el que el individuo está completamente involucrado en la tarea.

Un estudio consiguió relacionar el índice de implicación con la actividad cerebral localizada en la parte posterior del cerebro [13][14][15][16]. En él se establece que las ondas que están implicadas en la implicación son las alfa, beta y theta .

Atendiendo a la definición de las métricas y a las zonas del cerebro que son participes en cada respuesta cognitivo – emocional, surgen relaciones entre las métricas. Algunas interpretaciones relacionan a la memorización con la carga cognitiva, ya que el proceso de la memorización tiene una carga cognitiva asociada, por lo que podría parte de la carga cognitiva mostrada podría deberse a él.

### **2.3.4 Valencia**

La valencia es definida como la cualidad afectiva intrínseca de un evento, objeto o situación. Puede ser valencia positiva (atracción) o valencia negativa (aversión). Por ejemplo, emociones asociadas a la valencia positiva son la alegría o la satisfacción, mientras que asociadas a la valencia negativa están el miedo o la ira. Es común encontrar situaciones o eventos que presenten una ambivalencia, es decir, que se exprese una mezcla de ambas.

Algunas de las situaciones que podrían provocar valencia positiva en un usuario son tareas que le resulten placenteras como jugar a un juego que le agrade, completar una tarea satisfactoriamente o ganar una partida. Por otra parte, algunas situaciones que podían provocar una reacción negativa son tareas frustrantes o tediosas.

La valencia es una métrica especial, puesto que los valores entre los que oscila se miden de -100 a 100 en lugar de 0 a 100. Esto se debe a que el valor mínimo de la escala no corresponde al valor menos intenso de la métrica. Por ejemplo, el valor mínimo (cero), en el resto métricas se corresponde a la ausencia de carga cognitiva, implicación o memorización. Sin embargo, el valor mínimo en la valencia expresa el valor máximo de valencia negativa (aversión), mientras que el máximo (cien) expresa la mayor intensidad de valencia positiva y el cero expresa una respuesta afectiva neutra.

Se han realizado diversos estudios con el objetivo de identificar con que fenómenos cerebrales se asocia la valencia [5]. Desde el comienzo de la investigación en los años 80, se ha puesto el foco en el análisis de la asimetría en el córtex frontal como principal indicador de esta métrica.

El córtex frontal se divide en la parte izquierda y derecha, y su actividad está fuertemente reaccionada. En este caso nos referimos a asimetría con la diferencia de actividad en estas dos partes. Se ha conseguido vincular la presencia de una fuerte asimetría entre las dos facciones del córtex con una mayor valencia experimentada por el usuario. La polaridad de la valencia está determinada por el hemisferio que experimenta mayor activación: una mayor actividad en el hemisferio derecho implica una reacción de valencia negativa, mientras que una mayor actividad en el hemisferio izquierdo indica que la respuesta relacionada con la atracción o la valencia positiva.

Por lo tanto, cuando la actividad en un hemisferio sea considerablemente mayor que en el otro, los valores de la valencia se encontraran en los límites, mientras que si no hay una diferencia pronunciada esta se encontrara entorno al 0.

En cuanto a las ondas emitidas por el cerebro, se ha establecido que la actividad cerebral en esta zona es inversamente proporcional a las ondas alfa [17][18][19]. Por ello, cuanto menor sea la presencia de ondas alfa en un hemisferio, mayor será la actividad en él.

## **2.4 Pre-procesamiento de los datos**

Una vez son adquiridos los datos en crudo, es necesario llevar a cabo una serie de pasos previos a la obtención de las métricas con el objetivo de alcanzar unos resultados más precisos.

A estos pasos se les conoce como pre-procesamiento, y son utilizados para eliminar el ruido que diferentes agentes pueden haber introducido en los datos.

Las técnicas a utilizar en el pre-procesamiento variarán dependiendo tanto de la naturaleza de los datos como del objetivo del proyecto. Algunos de los procedimientos empleados en esta tesis son los filtros de banda y el análisis de componentes independientes (ICA)

### 2.3.1 Filtros de banda

Los filtros de banda son una técnica que nos permite aislar las frecuencias deseadas, eliminando o atenuando las que no se encuentren en el rango establecido.

Existen varios tipos de filtros: los de paso bajo, que permiten el paso a las frecuencias menores que el umbral; los de paso alto, que permiten el paso a las frecuencias mayores que el umbral y los filtros de paso de banda, los cuales permiten seleccionar un rango entre dos frecuencias límite y elimina las frecuencias fuera de ese rango.

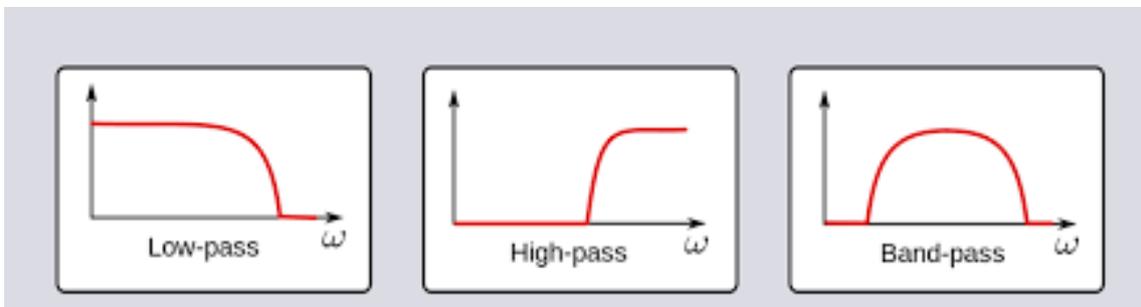


Figura 8 Filtros de paso bajo, alto y de banda

En este estudio se utilizan diferentes tipos de filtros de banda. En primer lugar, se utiliza un filtro de paso alto sobre la frecuencia de 1 Hz con el fin de eliminar el ruido producido por el *offset*. El ruido de *offset* está producido por desviaciones en el voltaje causadas por componentes como los electrodos o el amplificador del casco.

A su vez, se utiliza un filtro de paso bajo para eliminar las frecuencias superiores a 30 Hz, ya que no se consideran relevantes para la obtención de las métricas.

Finalmente, se aplican filtros de paso de banda para obtener los diferentes tipos de onda.

### 2.3.2 Análisis de componentes independientes (ICA)

El término artefacto se refiere a cualquier ruido en los datos que pueda ser atribuido a una fuente concreta [20]. Los dispositivos de medición de señales EEG son susceptibles a cualquier tipo de estímulo nervioso que pueda ser

producido en áreas cercanas. Algunos ejemplos son el ruido generado por el parpadeo, por los movimientos oculares o por contracciones de los músculos del cuello y mandíbula.

Todos estos artefactos generan ruido siguiendo unos patrones muy concretos. El ruido ocular es el más relevante debido a la frecuencia con la que ocurre. Los parpadeos son la principal fuente dentro del ruido ocular, debido a que una persona adulta parpadea de 15 a 20 veces por minuto y duran de 250 a 300 milisegundos. Los sensores que se encuentran en la parte frontal son altamente sensibles a este ruido, ya que suele ser de una magnitud diez veces mayor a los datos EEG normales.

De la misma forma, los sensores frontales también detectan los movimientos oculares. Las neuronas de la retina se organizan formando dipolos, es decir, son una fuente con un polo negativo y uno positivo. Mientras no hay movimiento ocular, estos dipolos están en equilibrio, por lo que no se aprecia en los electroencefalogramas. Cuando se produce un movimiento, la posición de los dipolos cambia y aumenta en un lado de la cabeza mientras que disminuye en el otro, reflejándose así en las ondas.

Otro tipo de ruido es el producido por los músculos cercanos. Las contracciones en la zona del cuello o de la mandíbula se manifiestan en los sensores cercanos. Habitualmente, el ruido muscular no es problemático, ya que sus apariciones son puntuales y no suelen impactar en gran medida en los datos. Sin embargo, si el sujeto está excesivamente tenso durante el experimento puede dificultar la recogida de datos.

Con el objetivo de eliminar el ruido generado por artefactos ajenos a la actividad cerebral, se aplica el algoritmo ICA, análisis de componentes independientes o *Independent Component Analysis* en inglés. Este algoritmo permite separar las fuentes de información ciegamente, es decir, sin previo conocimiento de cuales o cuantas son las fuentes.

Cada electrodo del casco detecta de forma simultánea datos provenientes de todas las partes del cerebro, y no solo de la más próxima como intuitivamente se podría pensar, al igual que ruido de las fuentes mencionadas anteriormente. El ICA nos permite separar las señales EEG de los artefactos ya que las señales

tienen diferentes propiedades espacio-temporales dependiendo de la fuente de la que provengan.

Un buen ejemplo para ilustrar este procedimiento es imaginar que en una misma habitación hay diferentes fuentes de sonido: persona A, persona B y un ventilador. Si se coloca un micrófono en la habitación, el sonido todos los orígenes será captado por el micrófono mezclado y no será posible separarlos. Si añadimos otros dos micrófonos en posiciones diferentes a la habitación, el sonido seguirá mezclado en cada uno de los micrófonos, pero debido a que cada micrófono capta de forma diferente el sonido de cada fuente, será posible diferenciar a que agente corresponde cada señal.

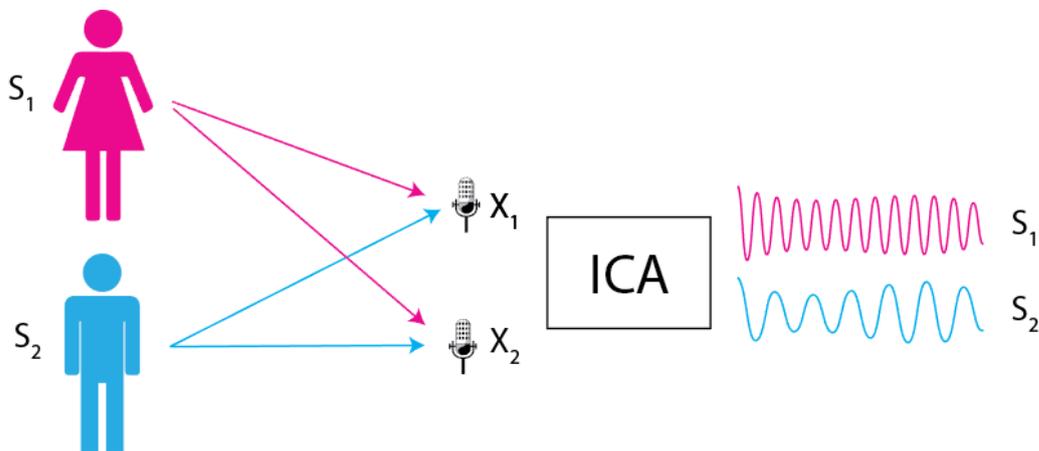


Figura 9. Funcionamiento del algoritmo ICA

Aplicando este principio a las señales EEG, usando múltiples sensores es posible detectar todos los artefactos cuyas ondas se mezclan con las relacionadas con la actividad cerebral, teniendo en cuenta la limitación de que el número máximo de componentes está limitado por el número de sensores.

Esto resulta de gran utilidad dado que permite seleccionar los componentes que se consideren emisores de ruido y eliminar las interferencias que hayan generado en la señal.

## 2.5 Machine Learning

El Aprendizaje Automático o *Machine Learning* es una rama de la Inteligencia Artificial. Consiste en una serie de técnicas que otorgan a los computadores la capacidad de aprender. Se entiende “aprender” como la capacidad de mejorar el

rendimiento y los resultados obtenidos a través de la experiencia y el procesamiento de los datos. Esto permite que las máquinas puedan tomar decisiones sobre nuevos casos sin tener que ser programados explícitamente para ello, extrapolando lo aprendido en los casos previos.

El *Machine Learning* está estrechamente ligado con el reconocimiento de patrones. En otros campos previos de la Inteligencia Artificial, como los sistemas expertos, conseguían llegar a conclusiones aplicando una serie de reglas establecidas previamente sobre un conjunto de datos, aplicando un razonamiento deductivo. Por otra parte, los modelos de Machine Learning funcionan a través del razonamiento inductivo. Tan solo disponen de los datos de entrada y la conclusión a la que se pretende llegar, por lo que deben inferir una función que sustituya a las reglas, para después aplicarla sobre nuevos datos de entrada y hacer predicciones.

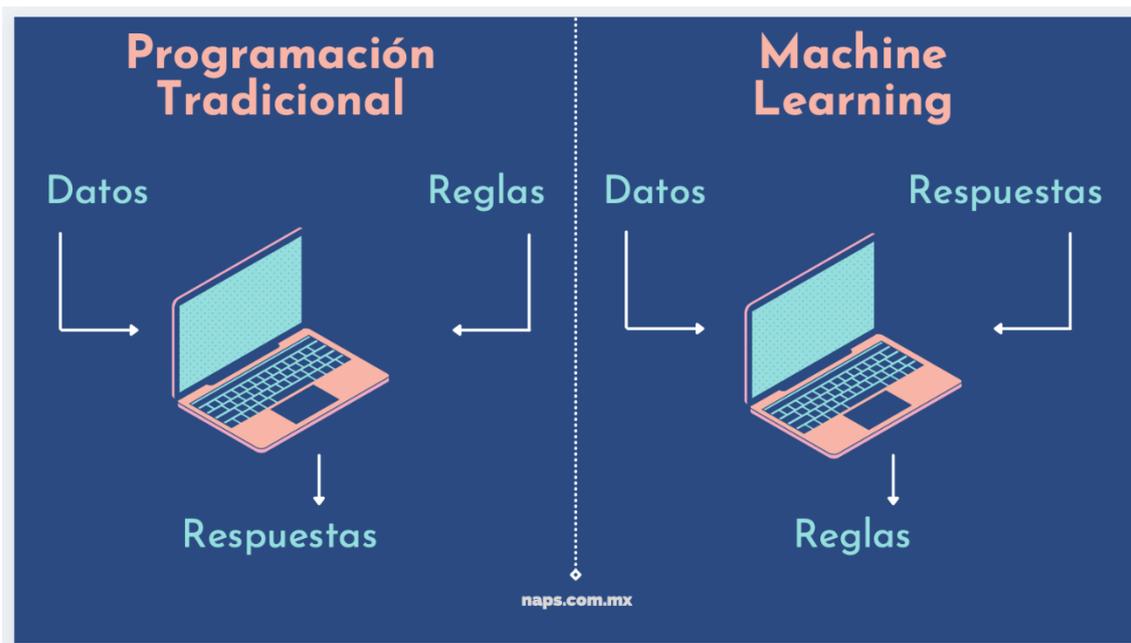


Figura 10. Programación tradicional vs Machine Learning

Existen varios tipos de modelos respecto al tipo de entrenamiento que reciben:

Los modelos con aprendizaje supervisado son aquellos donde las variables de entrada son etiquetadas con un valor de salida. El objetivo del modelo será encontrar una función la cual sea capaz de predecir el valor objetivo usando las variables de entrada [21]. Dependiendo del tipo de variable objetivo se pueden diferenciar dos tipos de modelo:

- Modelos de clasificación: La predicción en este tipo de modelos será de carácter cualitativo, es decir, se le otorgará una etiqueta que indicará la categoría a la que pertenece. Un ejemplo claro de este tipo de modelo sería un clasificador que diferencia entre limones y naranjas, donde las dos etiquetas posibles serían “Limón” y “Naranja”.
- Modelos de regresión: La predicción en un modelo de regresión es numérica y, generalmente, continúa. En este caso los datos de salida formarán una función de salida que puede ser lineal o polinomial.
 

En los modelos lineales existe una relación lineal entre la variable independiente (entrada) y la dependiente (salida). En estos casos el modelo trata de encontrar una ecuación lineal o línea recta que se ajuste lo máximo posible a los datos.

En el caso de que existan más variables independientes se trata entonces de regresión lineal múltiple, donde la ecuación lineal pasa a ser una combinación de los diferentes coeficientes atribuidos a cada variable.

La regresión polinomial se da cuando la relación entre las variables no es lineal y se debe utilizar polinomios para ajustar la curva a los datos. Esto permite adaptarse a datos con patrones complejos.

Los modelos con aprendizaje no supervisado son aquellos donde los datos de entrada no están asociados con una variable objetivo. En este caso, el objetivo de los modelos es describir la estructura de los datos para poder encontrar algún tipo de relación que simplifique el análisis. Un

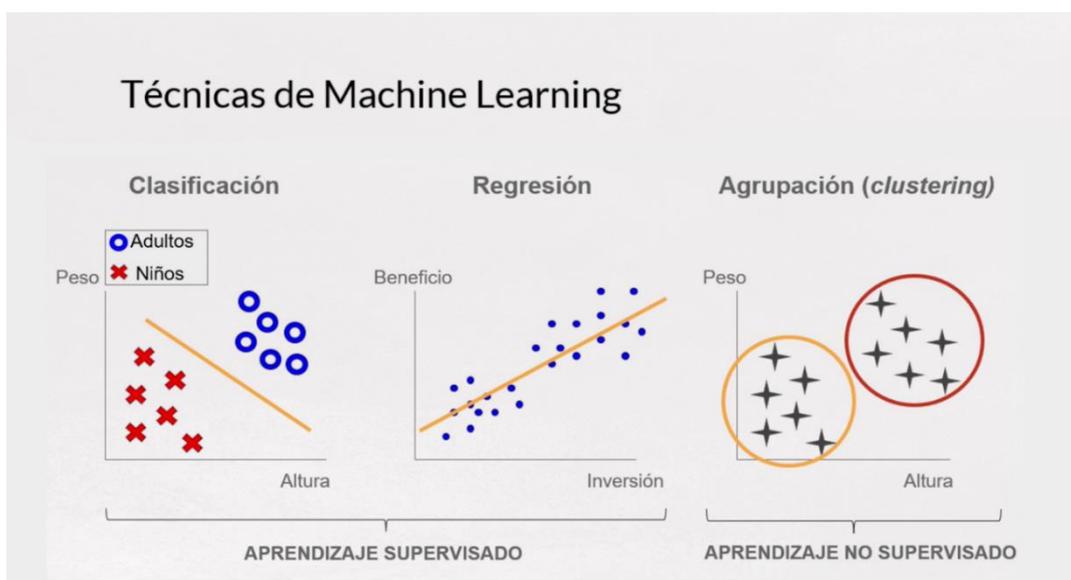


Figura 11. Técnicas de Machine Learning

ejemplo son los algoritmos de *clustering*, que buscan agrupaciones entre los datos basándose en sus similitudes.

Existen más tipos de aprendizaje, como el aprendizaje semisupervisado, por refuerzo o transducción, pero son menos utilizados.

Los modelos de Machine Learning pueden ser implementados utilizando diferentes algoritmos. En este caso se hace hincapié en las Redes Neuronales, ya que son las utilizadas en este trabajo.

Las Redes Neuronales son un algoritmo que intenta imitar el funcionamiento del cerebro humano conectando diferentes capas de “neuronas”. Está formada por la capa de entrada, las capas ocultas y la capa de salida. La capa de entrada recibe los datos y su número de neuronas es igual al de variables de entrada, las capas ocultas son las encargadas de los cálculos intermedios y el número de capas como de neuronas puede variar con el fin de obtener mejores resultados, y la capa de salida devuelve el resultado obtenido.

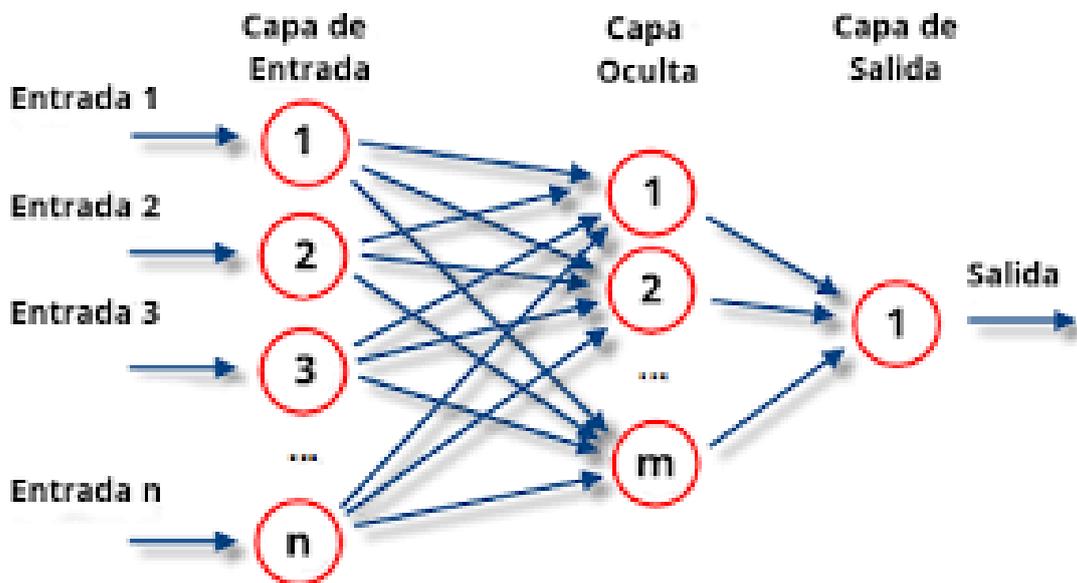


Figura 12. Estructura de una red neuronal

En las redes neuronales, una neurona recibe como entrada los valores de todas las neuronas de la capa anterior. Estos valores son ajustados con un peso o coeficiente por cada neurona de entrada. A estos valores se les aplica una función y se propaga el resultado a todas las neuronas de la capa posterior. Entrenar una red neuronal consiste en ajustar los pesos y las conexiones entre

las neuronas de forma que se adecue lo máximo posible al patrón. Para ello se tiene en cuenta el error obtenido y cuanto ha aportado cada neurona al resultado final. A este algoritmo se le conoce como *Backpropagation* o propagación hacia atrás.

Las redes neuronales pueden ser configuradas haciendo uso de diferentes parámetros. Algunos de ellos son:

- Capas ocultas: es posible configurar el número de capas ocultas así como el número de neuronas por capa, lo que puede aumentar la capacidad de la red para adaptarse al patrón.
- Función de activación: es la función matemática que se le aplica a los valores dentro de una neurona.
- *Solver*: se trata del algoritmo de optimización para los pesos. El algoritmo “adam” funciona mejor para muestras de datos grandes, mientras que “lbfgs” converge más rápido y ofrece mejores resultados con muestras de datos pequeñas
- Iteraciones máximas: el número máximo de veces que se pueden iterar los datos.
- Ratio de aprendizaje: es posible configurar la magnitud de los cambios a los pesos en cada iteración. Un ratio elevado supondrá que las correcciones afectarán en gran medida a los pesos, mientras que con un ratio menor las modificaciones de cada iteración no tendrán un gran impacto. Es posible ajustar este ratio para que sea constante o dinámico.

Una práctica común en el entrenamiento de modelos es dividir los datos en datos de entrenamiento y datos de validación. Los datos de entrenamiento son los datos con los que se entrena el modelo, mientras que los datos de validación se utilizan durante el entrenamiento para comprobar la efectividad

del modelo y llevar a cabo modificaciones en los parámetros en consecuencia.

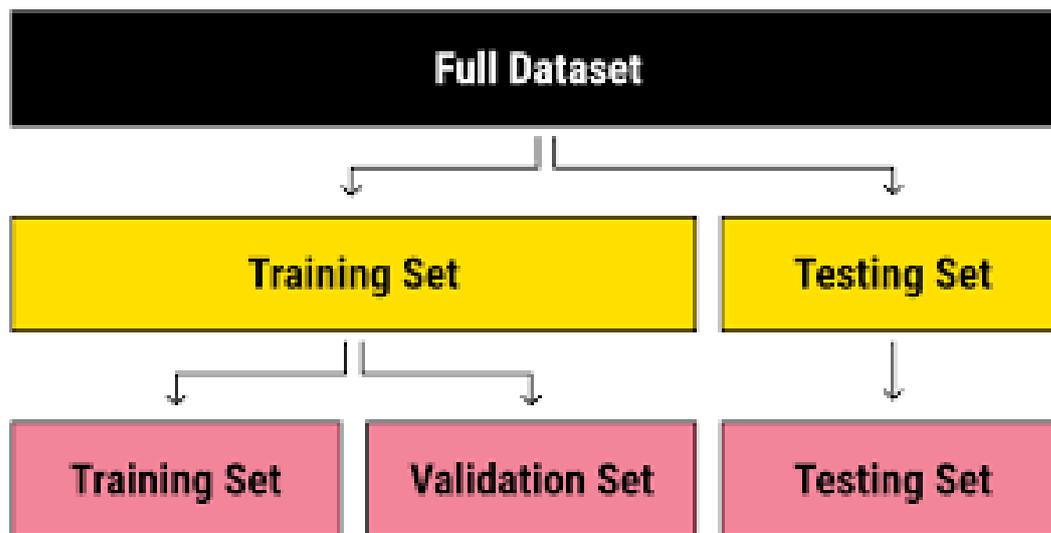


Figura 13. Partición de los datos

Normalmente se lleva a cabo una división en torno al 70%-80% de datos de entrenamiento y 30%-20% de datos de validación. Además, hay que tener en cuenta los datos de test, que son aquellos que sirven para medir la precisión del modelo después de la fase de entrenamiento de forma definitiva. Se distinguen de los de validación en que estos se usan de forma iterativa en cada entrenamiento para evaluar los resultados, por lo que se conocen como “de desarrollo”.

### 3. Diseño e implementación de la herramienta

La herramienta ha sido desarrollada usando Python [22]. Python es un lenguaje de alto nivel de programación. Soporta parcialmente la orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, dinámico y multiplataforma. Python es el lenguaje predilecto para el procesamiento de grandes cantidades de datos. Proporciona una gran variedad de librerías para la gestión de datos como NumPy o Pandas, y además permite incorporar librerías para el procesamiento de señales EEG como MNE. Además, librerías como sci-kit learn lo han convertido en uno de los principales lenguajes para el Machine Learning. Aunque no sea tan rápido como lenguajes compilados (C++), proporciona la velocidad necesaria para el tratamiento masivo de datos.

Los datos usados para el entrenamiento son los obtenidos en experimentos previos del grupo de investigación. En concreto, los obtenidos en el estudio sobre las interfaces adaptativas [23]. Se tratan de sesiones de 8 minutos, por lo que la cantidad de muestras excede los 120.000 datos.

La herramienta desarrollada en este trabajo de fin de grado está compuesta por tres partes: la limpieza de los datos, la disposición de los datos en el formato adecuado y finalmente el entrenamiento del modelo de Machine Learning.

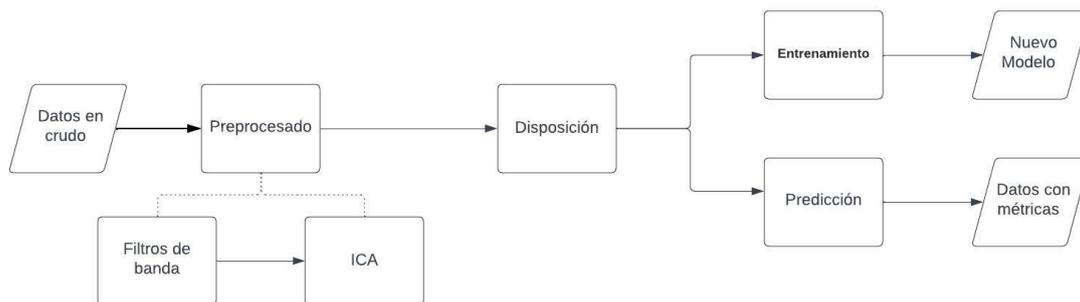


Figura 14. Flujo de los datos respecto a los diferentes procesos

### 3.1 Limpieza de datos

La librería MNE cumple un rol clave en el preprocesado de los datos. MNE es una librería especializada en la exploración, visualización y análisis de datos neurofisiológicos (EEG, MEG, NIRS, etc) [11]. Es un proyecto *open-source* en el que han colaborado instituciones como el NIH (National Institute of Health), el NSF (National Science Foundation), Google, Amazon y prestigiosas universidades de todo el mundo.

El objetivo de la limpieza de datos es aplicar los conceptos expuestos en el apartado 2.3 a través del uso de librerías especializadas para eliminar los principales tipos ruido.

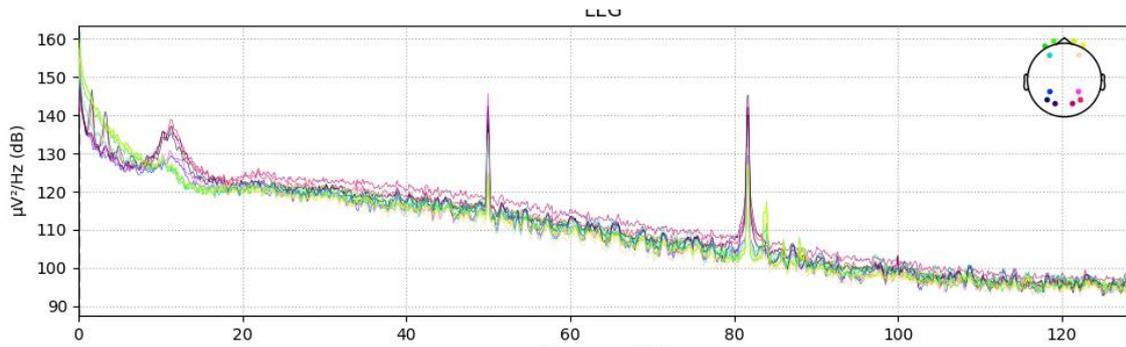


Figura 15. Transformada de Fourier antes del filtro de banda

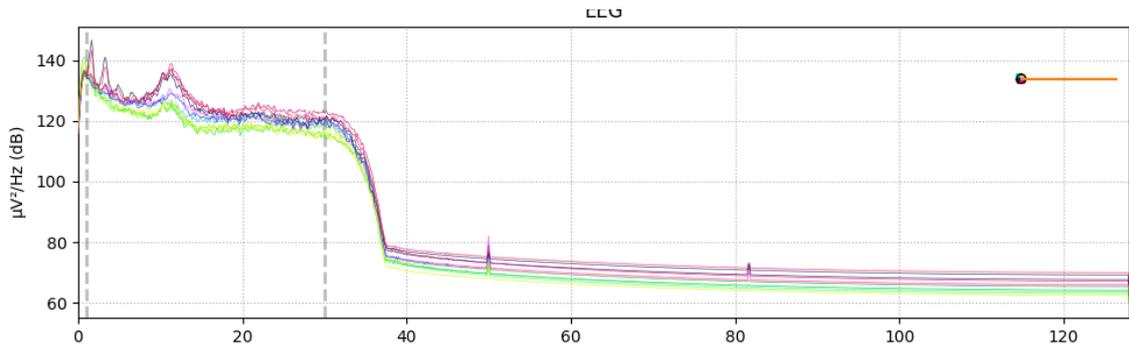


Figura 16. Transformada de Fourier después del filtro de banda

En primer lugar, se importan los datos en crudo del archivo CSV generado a través de SennsLab. Estos datos se guardan en un *array* de la librería NumPy.

A continuación, los datos se cargan en un *RawArray* de la librería MNE. Esta estructura de datos nos permite disponer los datos de forma continua. Para generar el *RawArray*, es necesario introducir la configuración que contiene información como la frecuencia de muestreo o el nombre de los canales.

El primer método de limpieza de datos que se aplica es un filtro de banda. La librería MNE nos facilita una función que permite filtrar los datos del *RawArray* y obtener solo las frecuencias que se encuentren entre los dos parámetros proporcionados, que en este caso son 1Hz y 30Hz.

Como se ve en la comparación entre la Figura 9 y la Figura 10, la transformada de Fourier refleja la distribución del potencial eléctrico a lo largo de las frecuencias. Se aprecia cómo, después del filtro, las frecuencias mayores a 30Hz decrecen considerablemente, al igual que el pico al inicio que llegaba a alcanzar un valor de 160 ha decrecido.

El segundo método de limpieza que se aplica es el algoritmo ICA. Como se menciona en el apartado 2.3.2, el objetivo de aplicar este algoritmo es eliminar el ruido generado principalmente por los movimientos oculares así como por el parpadeo. La librería MNE también proporciona una función para ejecutar este algoritmo sobre unos datos continuos. Usando esta función, es posible identificar los diferentes componentes existentes en las ondas.

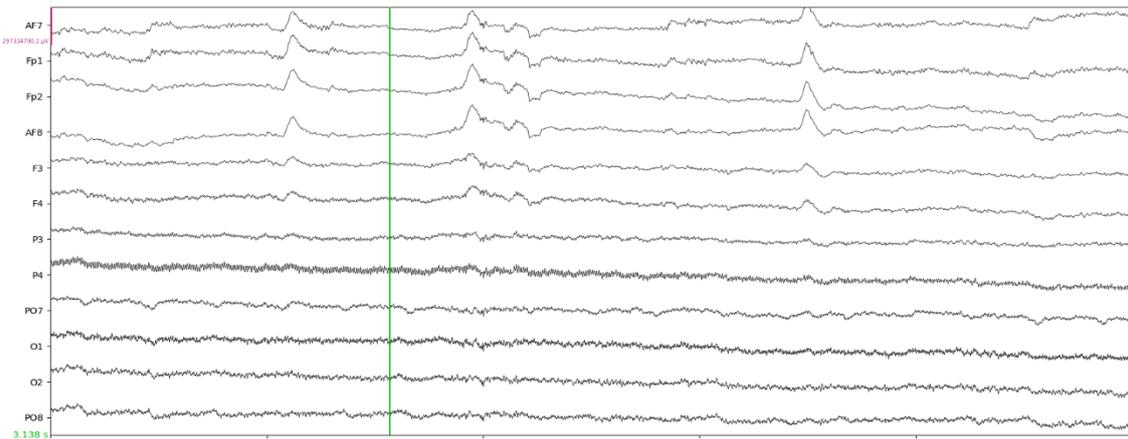


Figura 17. Datos antes del algoritmo ICA

Una vez se han extraído los componentes, se pueden encontrar los que son fuentes de ruido. Para ello, se seleccionan dos canales próximos a la fuente de ruido (en este caso los sensores frontales Fp1 y AF8), y un límite o *threshold*, que en este caso es el recomendado por defecto.

Si se comparan las figuras 11 y 12, se aprecia como después del algoritmo ICA los picos en los primeros sensores (los sensores frontales) desaparecen.

Al finalizar estos procesos, se ha conseguido eliminar el ruido proveniente del offset generado por los componentes electrónicos y se han eliminado los picos causados por el parpadeo.

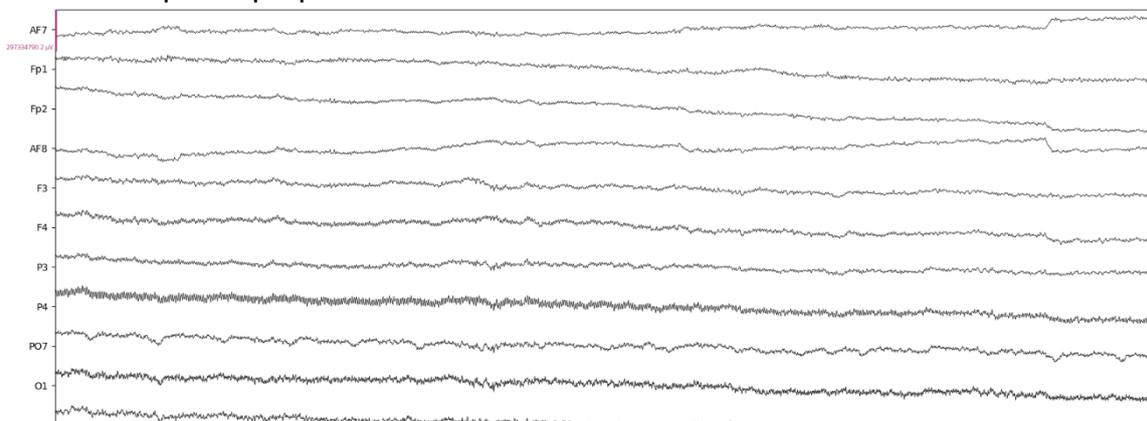


Figura 18. Datos después del algoritmo ICA

## 3.2 Disposición de los datos

La fase de disposición de los datos pretende organizar los datos en un formato adecuado para ser procesados por el modelo.

time	AF7	Fp1	Fp2	AF8	F3	F4	P3	P4	P07	O1	O2	P08	
0	0.088800	-1.795043e+08	-2.239372e+08	-2.359885e+08	-2.108387e+08	-9.124046e+03	-9.258404e+03	-2.854801e+03	-2.663801e+03	-1.399713e+03	-1.177221e+03	-1.204208e+03	-1.484835e+03
1	0.083900	7.693366e+06	6.539337e+06	4.984654e+06	2.538483e+06	5.897552e+06	1.170437e+07	6.598888e+06	1.884473e+07	4.118948e+06	7.032719e+06	4.968756e+06	9.775847e+06
2	0.007812	1.264614e+07	1.085832e+07	8.017979e+06	4.092569e+06	9.911866e+06	1.935895e+07	1.135488e+07	3.247775e+07	7.948397e+06	1.295354e+07	8.524651e+06	1.642955e+07
3	0.011719	1.341979e+07	1.175378e+07	8.218147e+06	3.868467e+06	1.093832e+07	2.044283e+07	1.389996e+07	3.781388e+07	1.117891e+07	1.768225e+07	9.821041e+06	1.824883e+07
4	0.015625	1.632468e+07	9.697891e+06	5.548512e+06	2.039112e+06	9.073521e+06	1.538182e+07	1.177838e+07	3.588525e+07	1.351158e+07	1.983231e+07	8.920487e+06	1.564476e+07

Figura 19. Disposición de los datos después del filtrado

Durante esta fase cabe diferenciar dos procesos. El primero es el que ha sido llevado a cabo para el entrenamiento de los modelos, y el segundo es el que se ejecuta a la hora de obtener nuevas métricas. La principal diferencia entre estos dos procesos es que para el entrenamiento de los modelos ha sido necesario importar el archivo con los valores de las métricas.

Durante el entrenamiento, las métricas obtenidas en una hoja de cálculo son convertidas a un archivo CSV y cargadas en un DataFrame de Pandas (librería *open-source* especializada en el procesamiento y la manipulación de datos.).

Dependiendo de la métrica con la que se quiera trabajar, se elige una configuración específica que contiene los sensores y las frecuencias necesarias. Las configuraciones elegidas para los modelos han sido:

	Sensores	Frecuencias
<b>Valencia</b>	AF7, AF8, Fp1, Fp2, F3, F4	Alfa
<b>Memorización</b>	P3, P4, F3, F4	Alfa, Beta, Theta, Gamma
<b>Carga cognitiva</b>	P3, P4	Alfa, Theta
<b>Implicación</b>	P3, P4	Alfa, Beta, Theta

Tabla 1 Configuración de los datos para cada métrica

Con estos datos, las estructuras de datos que contienen a las métricas y a los datos en crudo se juntan con el fin de crear una asociación entre los valores de los sensores en cada instante y el valor de la métrica correspondiente.

Posteriormente, se eliminan los datos que están marcados como “No válidos” en las métricas de forma que solo se procesarán las muestras que no tengan ningún valor faltante.

Como se explica en el apartado 2.1.2, las métricas obtenidas de SennsMetrics estás dispuestas en grupos de 8. Esto significa que son necesarias las ocho mediciones para obtener el valor de la métrica. Por ello, se han unido las muestras pertenecientes del mismo grupo con el fin de crear una única muestra de cara al modelo. Esto quiere decir que cada muestra tendrá 8 grupos de valores denominados como T1, T2... T8 donde cada grupo T estará formado por los datos en crudo de las frecuencias y sensores necesarios.

	T1-Alpha-P3	T1-Alpha-P4	T1-Theta-P3	T1-Theta-P4	T2-Alpha-P3	T2-Alpha-P4	T2-Theta-P3	...	T7-Theta-P3	T7-Theta-P4	T8-Alpha-P3	T8-Alpha-P4	T8-Theta-P3	T8-Theta-P4	Valor
0	4.912998	11.698234	2.738544	11.927930	4.578848	11.623741	3.624489	...	3.326860	14.779850	-1.626165	-2.628637	3.153265	14.188342	61.701626
1	-2.526837	-4.659310	2.906528	13.336901	-3.188398	-6.144022	2.592183	...	0.271711	4.701675	-1.938324	-3.933871	-0.280650	3.159930	61.879887
2	-1.205945	-2.575278	-0.840488	1.585290	-0.472406	-1.293145	-1.399010	...	-3.906976	-3.566857	2.083695	1.254428	-4.308940	-6.272592	62.398320
3	2.097184	0.945477	-0.666842	-6.843806	2.068755	0.611810	-4.976941	...	-5.729658	-0.125847	-0.065717	0.980030	-5.708059	-8.114380	62.754787
4	-0.544977	1.346461	-6.625470	-0.069500	-1.012534	1.784433	-5.480732	...	-3.778866	-7.197354	-1.783878	1.674110	-3.237166	-6.905772	60.836948
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
15164	7.409690	2.591133	-1.599812	-1.254458	9.624470	5.561592	-1.105940	...	0.864535	-0.177829	5.512144	9.369639	1.163936	-0.830883	6.209429
15165	2.787088	7.474805	1.010668	0.055396	0.018763	5.184604	1.680121	...	1.973015	-0.364707	-7.115100	-6.095333	1.095364	-0.533932	5.692502
15166	-6.075182	-6.4099787	2.028686	-0.698163	-4.588499	-5.222701	2.084047	...	2.939945	-0.864814	3.614088	-0.356068	3.236257	-0.677834	5.410284
15167	3.605534	0.411715	3.561890	-0.426543	3.122258	0.948841	3.902960	...	5.301366	1.866399	-2.896558	0.925792	5.392556	2.232519	4.973958
15168	-3.142030	0.071239	5.382228	2.547023	-2.956610	0.983109	5.257181	...	2.734397	2.608828	3.366412	1.825120	1.076616	2.251396	4.416993

Figura 20. Disposición de los datos previa al entrenamiento

Una vez transformados los datos, están listos para ser procesados por el modelo.

Para realizar predicciones utilizando los modelos no será necesario añadir el archivo de métricas a los datos obtenidos del filtrado, bastará con reorganizar los datos y aplicarles el modelo.

### 3.3 Modelo de Machine Learning

El objetivo de esta parte es entrenar cuatro modelos de Machine Learning, uno por cada métrica, con el fin de realizar las predicciones sobre los datos. Para implementar estos modelos se hace uso de la librería *scikit-learn*. Esta es una librería especializada en Machine Learning, y proporciona todas las funciones necesarias tanto para el entrenamiento como para la predicción y evaluación de los modelos.

Para el entrenamiento de los modelos se ha hecho uso de la plataforma Google Colab, una plataforma de Google especializada en el procesamiento de datos y modelos de Machine Learning, que pone a disposición de los usuarios computadores con potentes GPUs y memorias.

En primer lugar, se importan los datos en formato de DataFrame desde un archivo csv. Después, tiene lugar la partición en datos de entrenamiento y de validación, siguiendo una proporción de 80% de datos de entrenamiento y 20% de datos de validación.

Como se ha comentado en el apartado 2.5, se ha usado una red neuronal de regresión como algoritmo. La librería *scikit-learn* tiene una clase *MLPRegressor* que implementa un modelo con el algoritmo *Multi-layer Perceptron regressor* (MLP). Este modelo se puede personalizar con algunos atributos como los números de capas y neuronas, el algoritmo de resolución o el número máximo de iteraciones.

Posteriormente, se procede a su entrenamiento, proceso que puede tener una duración considerable dependiendo de la cantidad de datos.

Finalmente, se utilizan los datos de validación para llevar a cabo el análisis del resultado. Esto se consigue separando las variables de entrada y el valor objetivo para posteriormente realizar predicciones para los valores de entrada. Una vez obtenidas las predicciones de los valores, se comparan con los valores reales y se obtiene el error.

El modelo es guardado como archivo usando la librería *joblib*, que permite guardar y posteriormente cargar los modelos.

A través de este proceso se ha conseguido guardar los modelos para la valencia, memorización, carga cognitiva e implicación en archivos que permitirán hacer predicciones para los datos obtenidos en futuros experimentos. Para ello, a los datos se les deberá aplicar los procedimientos de limpieza y disposición de datos, a excepción de la importación de las métricas como valores objetivo.

## 4. Trabajos relacionados

Aún que el uso de señales EEG para la obtención de métricas es un campo relativamente nuevo y experimental, ha habido otros proyectos aparte de los propios de BitBrain que, siguiendo diferentes enfoques, han conseguido identificar algunos estados cognitivo – emocionales en las personas.

Uno de estos estudios utilizó los modelos de Machine Learning para asociar unos eventos categorizados como positivos, neutros y negativos para asociarlos a determinados fenómenos cerebrales [24]. En este estudio, se propone un enfoque completamente diferente al propuesto por BitBrain. En lugar de calcular las métricas usando una serie de reglas obtenidas a través del estudio del comportamiento del cerebro, directamente se presentan al usuario una serie de estímulos clasificados con una etiqueta de positivo, neutro o negativo.

En este caso estaríamos ante un modelo de clasificación, en lugar de regresión, ya que no se cuantifica la intensidad de las emociones si no que se asigna una categoría. Estos estímulos son clasificados según el impacto esperable que causarían en la mayoría de la población. Por ejemplo, algunos de los estímulos presentados son: la escena de la muerte de un perro en una película, videos divertidos de perros o el número musical inicial de la película “La La Land”.

Este enfoque presenta un problema. Si bien estos estímulos están fuertemente asociados a una emoción, hay personas que pueden no sentir la emoción esperada al verlos, por lo que no se establecería la clasificación correcta. Por ejemplo, una persona que detesta los musicales presentaría una reacción negativa al ver el musical la cual sería etiquetada como positiva.

Aun así, este enfoque resulta interesante debido a que se proporciona una manera más directa de establecer la relación entre los fenómenos cerebrales y el estado del usuario. Podría ser interesante si se hace un estudio previo adecuado sobre cada usuario en concreto y se encuentran estímulos para los que haya cierto grado de confianza.

## 5. Validación.

La validación consiste en comparar las predicciones hechas por el modelo con los valores reales de un conjunto de datos. Esta validación se puede realizar a través de diferentes indicadores:

- Error medio absoluto: El error medio absoluto es la media aritmética de los errores de cada pareja de valores. Se define como:

$$\text{MAE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} |y_i - \hat{y}_i|.$$

En esta fórmula  $y$  es el valor real e  $\hat{y}$  es el valor predicho por el modelo.

- Error medio cuadrado: El error medio cuadrado es similar al error medio absoluto pero los errores son elevados al cuadrado con el fin de castigar más los errores grandes:

$$\text{MSE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} (y_i - \hat{y}_i)^2.$$

- Error medio absoluto porcentual: Se calcula de forma parecida al error medio absoluto, pero en este caso se divide el error por el valor real con el fin de relativizar el error a la magnitud de los valores que lo generan:

$$\text{MAPE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} \frac{|y_i - \hat{y}_i|}{\max(\epsilon, |y_i|)}$$

El símbolo  $\epsilon$  representa un número entero pequeño y positivo en caso de que el valor real sea 0.

- Error máximo: Se trata del peor caso donde la diferencia entre el valor predicho y el real es la mayor.

$$\text{Max Error}(y, \hat{y}) = \max(|y_i - \hat{y}_i|)$$

Para evaluar los modelos se van a utilizar el indicador del error medio absoluto, ya que refleja la desviación promedia de los valores, y el error mediano absoluto, que ofrece una mayor resistencia frente a los errores de los valores atípicos y va a permitir observar el impacto de estos en el error medio absoluto.

Con el fin de obtener la mejor precisión, se han probado diferentes configuraciones para las capas, y se ha descubierto que, debido a las diferentes características y complejidad en el cálculo de cada métrica, cada modelo funciona mejor con una configuración particular.

A continuación, se exponen los resultados obtenidos para cada métrica con las diferentes configuraciones. En este caso, se han configurado los modelos con una sola capa oculta.

- **Valencia**

Como se observa en la Tabla 2, para la valencia se ha probado a configurar las neuronas de la capa oculta con 40, 20 y 10 neuronas. Se puede observar que el error disminuye ligeramente cuando hay menos neuronas, pero al ser un cambio tan sutil no es representativo.

<b>Neuronas</b>	<b>Error medio absoluto</b>	<b>Error mediano absoluto</b>
<b>40</b>	43.74	33.75
<b>20</b>	43.70	33.79
<b>10</b>	43.40	33.58

Tabla 2. Precisión en los modelos de valencia

- **Carga cognitiva**

En la carga cognitiva (Tabla 3), las variaciones del error son mínimas por lo que tampoco se toman como una diferencia significativa.

<b>Neuronas</b>	<b>Error medio absoluto</b>	<b>Error mediano absoluto</b>
<b>40</b>	22.57	15.24
<b>20</b>	22.14	14.79
<b>10</b>	22.34	15.11

Tabla 3. Precisión en los modelos de carga cognitiva

- **Memorización**

En este caso, en la Tabla 4 sí que se observa una ligera mejoría de la precisión con un menor número de capas.

<b>Neuronas</b>	<b>Error medio absoluto</b>	<b>Error mediano absoluto</b>
<b>40</b>	20.84	15.26
<b>20</b>	20.46	15.15
<b>10</b>	20.06	15.04

Tabla 4. Precisión en los modelos de memorización

- **Implicación**

Finalmente, podemos observar en la Tabla 5 como en la implicación la mejoría de los resultados cuando se aumenta el número de capas no es significativa.

<b>Neuronas</b>	<b>Error medio absoluto</b>	<b>Error mediano absoluto</b>
<b>40</b>	24.66	19.40
<b>20</b>	24.70	19.30
<b>10</b>	24.73	19.44

Tabla 5. Precisión en los modelos de implicación

Como se puede observar, la configuración de las capas ocultas no tiene impacto en la precisión del modelo, ya que las variaciones de un modelo a otro resultan insignificantes. Esto puede deberse a que el patrón es suficientemente sencillo para ser capturado con las configuraciones más simples, por lo que aumentar el número de neuronas puede ser hasta contraproducente.

Se han probado los algoritmos sgd y lbfgs para todos los modelos, pero los resultados no han sido satisfactorios. Para el algoritmo lbfgs, no se ha conseguido que el modelo convergiera en ninguno de los casos, incluso aumentando el número de iteraciones máximas. Para el algoritmo sgd, los

resultados han sido considerablemente peores que en los modelos entrenados con adam, Por ello, se ha decidido utilizar exclusivamente el algoritmo adam.

Se puede observar que los errores oscilan entorno a los 20 puntos, a excepción de la valencia. Esta variación en la valencia se debe a que, como se explicó en el punto 2.4.1, los valores de esta métrica están comprendidos entre -100 y 100 en lugar de 0 y 100. Al tener un rango con el doble de valores, el error medio se duplica. En el anexo se puede ver un ejemplo de como son los datos de salida generados por los modelos.

Se puede observar que los valores del error mediano absoluto son considerablemente menores que los del error medio absoluto. Esto indica que existen una serie de valores anómalos en las predicciones que tienen un gran impacto en el cálculo del error, y que, a través de algún método de detección, pueden ser eliminados y con ello ambos errores disminuirían. Una posible mejora sería establecer umbrales que sirvan para detectar valores anómalos, y que marque como valores inválidos aquellos que se encuentren por encima o por debajo.

Un error medio de 20 puntos sobre una escala de 0 a 100 se considera aceptable, ya que el objetivo de este trabajo no es elaborar una herramienta con una gran precisión numérica, si no poder conocer dentro de unos rangos de confianza el estado del usuario.

## **6. Conclusión**

En este trabajo se ha realizado una profunda exploración teórica sobre el funcionamiento de las ondas EEG, su procesado, la limpieza de los datos así como de las métricas y su relación con la actividad cerebral. Se espera que toda esta información detallada en este documento pueda servir de base para futuros desarrollos.

Además, se han desarrollado *scripts* tanto para la limpieza de datos como para su disposición en el formato usado por los modelos.

Como resultado final se han obtenido los modelos así como las *scripts* para su entrenamiento y validación.

El trabajo ha tenido una complejidad mayor a la planeada. En un primer momento se pensaba que la bibliografía existente entraría en mayor profundidad en lo relativo al cálculo de las métricas. Sin embargo, se ha visto que los estudios realizados en este ámbito son en gran parte experimental. En ellos se proponen hipótesis sobre la relación entre la actividad cerebral y las métricas, las cuales son comprobadas mediante experimentos. Aún que se han encontrado relaciones suficientes para llevar a cabo el trabajo, en muchos casos no había unanimidad sobre cuál es la mejor forma de calcularlas.

Aun así se ha conseguido cumplir la mayoría de los objetivos propuestos. El único objetivo que no se ha podido completar debido a la inesperada carga de trabajo ha sido la aplicación para la visualización de datos, la cual se espera que sea desarrollada en trabajos futuros.

Por lo tanto, la valoración final del trabajo es satisfactoria. Se espera que el procesamiento y los modelos obtenidos en este trabajo sean la base de futuros proyectos.

## **7. Trabajo futuro**

Se espera que la herramienta creada en este trabajo sea el punto de partida en un proyecto mayor con el objetivo de crear una aplicación de laboratorio que facilite las tareas al equipo.

El objetivo es integrar estos modelos en una aplicación que satisfaga algunas de las necesidades del equipo de investigación. Esta aplicación no pretende ser un sustituto del software de BitBrain, si no ser una herramienta complementaria que permita usar funcionalidades no implementadas en él.

Una de estas funciones es la obtención de las métricas en tiempo real. Como se ha explicado en el apartado 2.1, el flujo de datos de BitBrain obliga a subir los datos a SennsCloud para su procesado. En el equipo se desearía poder obtener las métricas durante el propio experimento. Esto es posible gracias al SDK del Diadem, que a través de Bluetooth se permite mandar los datos en bruto en el

momento que se capturan. Usando este *Software Development Kit* de Python, es posible leer, procesar y obtener valores para los datos en tiempo real.

Esto abriría un gran abanico de posibilidades, ya que se podría recibir *feedback* instantáneo del estado del usuario mientras se desarrolla el experimento, permitiendo así modificar los estímulos presentados o variar las condiciones del experimento durante el mismo.

Otra función que puede resultar de interés es la visualización de las métricas. A pesar de que la visualización de las métricas en la plataforma de BitBrain es muy potente, sería posible crear una aplicación donde se pueda personalizar y adaptar esta visualización a las necesidades del equipo de investigación.

Más allá de las oportunidades en cuanto a una aplicación de laboratorio, encontramos la posibilidad de integrar los modelos en aplicaciones que lleven el procesamiento de las métricas a un nuevo nivel. Por ejemplo, aplicaciones que reciban en tiempo real la información de las métricas y realicen una serie de procesos automatizados en consecuencia. Esto abre la puerta a un sinfín de posibles desarrollos, como los sistemas adaptativos centrados en el usuario, que pueden adaptarse a sus necesidades y modificar su comportamiento atendiendo al estado mental del usuario. Por ejemplo, simplificando una tarea en caso de que la carga mental sea excesiva o modificando una interfaz en caso de que la valencia sea negativa.

Gracias a este trabajo se han conseguido una serie de modelos preliminares que, aunque no cuentan con la máxima precisión, dan indicios de la viabilidad del proyecto y sirven de base para los trabajos futuros. Por ello, para el trabajo futuro queda la posibilidad de mejorar la precisión de los modelos. Ya sea adoptando nuevas configuraciones de los sensores y bandas, añadiendo nuevos pasos en la limpieza de datos o modificando los parámetros de los modelos es posible mejorar los resultados obtenidos.

## 8. Referencias

- [1] Bitbrain. Disponible en: <https://www.bitbrain.com/>. [Último acceso: septiembre 2023]
- [2] Nuwer MR. (2018). 10 – 10 electrode system for EEG recording, *Clinical Neurophysiology*, Volume 129, Issue 5, , Page 1103, ISSN 1388-2457, <https://doi.org/10.1016/j.clinph.2018.01.065>.
- [3] Nayak CS, Anilkumar AC. (2023). EEG Normal Waveforms. Jul 24. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; Jan–. PMID: 30969627.
- [4] Gevins, A., Smith, M. E., Leong, H., McEvoy, L., Whitfield, S., Du, R., & Rush, G. (1998). Monitoring working memory load during computer-based tasks with EEG pattern recognition methods. *Human Factors*, 40(1), 79–91.
- [5] Klimesch W. (1999) EEG alpha and theta oscillations reflect cognitive and memory performance: a review and analysis. *Brain Res Brain Res Rev.* 29(2-3):169-95. doi: 10.1016/s0165-0173(98)00056-3. PMID: 10209231.
- [6] Stikic M, Berka C, Levendowski DJ, Rubio RF, Tan V, Korszen S, Barba D, Wurzer D. (2014). Modeling temporal sequences of cognitive state changes based on a combination of EEG-engagement, EEG-workload, and heart rate metrics. *Frontiers in Neuroscience*, 8(342), 1–14.
- [7] Brouwer, A.-M., Hogervorst, M. A., van Erp, J. B. F., Heffelaar, T., Zimmerman, P. H., & Oostenveld, R. (2012). Estimating workload using EEG spectral power and ERPs in the n-back task. *Journal of Neural Engineering*, 1–14
- [8] Klimesch, W., Doppelmayr, M., Schimke, H., & Ripper, B. (1997). Theta synchronization and alpha desynchronization in a memory task. *Psychophysiology*. Retrieved from FC
- [9] Sederberg PB, Kahana MJ, Howard MW, Donner EJ, Madsen JR. (2003) Theta and Gamma Oscillations during Encoding Predict Subsequent Recall
- [10] Klimesch, W., Doppelmayr, M., Russegger, H., & Pachinger, T. (1996). Theta Band Power in the Human Scalp EEG and the Encoding of New Information. *NeuroReport*, 7, 1235–1240.
- [11] Long, N. M., Burke, J. F., & Kahana, M. J. (2014). Subsequent memory effect in intracranial and scalp EEG. *NeuroImage*, 84, 488–494.
- [12] Klimesch W, Schack B, Sauseng P. (2005). The functional significance of theta and upper alpha oscillations. *Experimental psychology* 52, 2, 99–108

- [13] Pope AT, Bogart EH, Bartolome DS.. (1995). Biocybernetic system evaluates indices of operator engagement in automated task. *Biol Psychol*.
- [14] Mikulka, P. J., Scerbo, M. W., & Freeman, F. G. (2002). Effects of a biocybernetic system on vigilance performance. *Human Factors*, 44(4), 654–664.
- [15] Freeman, F. G., Mikulka, P. J., Prinzel, L. J., & Scerbo, M. W. (1999). Evaluation of an adaptive automation system using three EEG indices with a visual tracking task. *Biological Psychology*, 50(1),61–76.
- [16] Stikic, M., Berka, C., Levendowski, D. J., Rubio, R. F., Tan, V., Korszen, S., ... Wurzel, D. (2014). Modeling temporal sequences of cognitive state changes based on a combination of EEG-engagement, EEG-workload, and heart rate metrics. *Frontiers in Neuroscience*, 8(342), 1–14.
- [17] Harmon-Jones, E., Gable, P. A., & Peterson, C. K. (2010). The role of asymmetric frontal cortical activity in emotion-related phenomena: A review and update. *Biological Psychology*, 84(3), 451–462.
- [18] Allen, J. J. B., Coan, J. A., & Nazarian, M. (2004). Issues and assumptions on the road from raw signals to metrics of frontal EEG asymmetry in emotion. *Biological Psychology*, 67, 183–218.
- [19] Davidson, R. J. (1992). Anterior Cerebral Asymmetry and the Nature of Emotion. *Brain and Cognition*, 20, 125–151.
- [20] Newman A J. Artifacts in EEG data. Disponible en :[https://neuralsciencetech.io/7-ee/erp\\_artifacts.html](https://neuralsciencetech.io/7-ee/erp_artifacts.html) [Último acceso: septiembre 2023]
- [21] Colaboradores de Wikipedia. Python. Wikipedia, la enciclopedia libre. Disponible en: <https://es.wikipedia.org/wiki/Python> [Último acceso: septiembre 2023]
- [22] Recuerdo de los Santos P. Tipos de aprendizaje en Machine Learning: supervisado y no supervisado. Telefonía tech. Disponible en: <https://telefoniatech.com/blog/que-algoritmo-elegir-en-ml-aprendizaje> [Último acceso: septiembre 2023]
- [23] MNE-python: Disponible en: <https://github.com/mne-tools/mne-python> [Último acceso: septiembre 2023]
- [24] Ayisha D. Analyzing EEG brainwave data to detect emotions – on cAlInvas. Disponible en: <https://medium.com/ai-techsystems/analyzing-eeeg-brainwave-data-to-detect-emotions-on-cainvas-48833f3f0811> [Último acceso: septiembre 2023]
- [25] Gaspar-Figueiredo D., Abrahao S, Insfran E., and Vanderdonckt J. (2023). Measuring User Experience of Adaptive User Interfaces using EEG: A Replication Study. In *Proceedings of the 27th International Conference on*

Evaluation and Assessment in Software Engineering (EASE '23). Association for Computing Machinery, New York, NY, USA, 52–61.  
<https://doi.org/10.1145/3593434.3593452>

# Anexo 1. Imágenes

```
*****
Predicción: -15.223210598200312
Etiqueta verdadera: -20.16476631
*****
Predicción: -78.48387768570467
Etiqueta verdadera: -67.50634766
*****
Predicción: 129.8760492020124
Etiqueta verdadera: 148.3344879
*****
Predicción: -29.27809611011071
Etiqueta verdadera: -10.63788605
*****
Predicción: 80.33254245114821
Etiqueta verdadera: 84.81185913
*****
Predicción: 1.1052199596979664
Etiqueta verdadera: 10.00813484
*****
Predicción: -13.51801065540356
Etiqueta verdadera: -0.829784751
*****
Predicción: 46.16337665460303
Etiqueta verdadera: 58.1657753
*****
Predicción: -50.71373276622808
Etiqueta verdadera: -89.16159821
*****
Predicción: 54.21956353077066
Etiqueta verdadera: 78.54371643
*****
```

Figura 21. Salida de datos de Valencia

```
*****
Predicción: 15.21108999783237
Etiqueta verdadera: 25.38847923
*****
Predicción: 52.11124594739241
Etiqueta verdadera: 31.18228149
*****
Predicción: 41.097455356512995
Etiqueta verdadera: 29.05646706
*****
Predicción: 42.22555115780508
Etiqueta verdadera: 51.99866486
*****
Predicción: 38.73646794587842
Etiqueta verdadera: 46.09077072
*****
Predicción: 45.24131766973617
Etiqueta verdadera: 25.51364708
*****
Predicción: 58.42024877382372
Etiqueta verdadera: 32.67840195
*****
Predicción: 30.477322932232013
Etiqueta verdadera: 17.21225548
*****
Predicción: 32.80422662878435
Etiqueta verdadera: 3.673604488
*****
Predicción: 38.11373193741397
Etiqueta verdadera: 36.41207504
*****
```

Figura 22. Salida de datos de carga cognitiva

```

*****
Predicción: 22.51879971161653
Etiqueta verdadera: 1.277436972
*****
Predicción: 29.504792981240456
Etiqueta verdadera: 12.96903515
*****
Predicción: 16.69007542275419
Etiqueta verdadera: 22.67229843
*****
Predicción: 22.675388909730714
Etiqueta verdadera: 1.979209423
*****
Predicción: 33.169167751781664
Etiqueta verdadera: 9.248719215
*****
Predicción: 37.547507916697086
Etiqueta verdadera: 34.98892593
*****
Predicción: 29.290325910856204
Etiqueta verdadera: 11.03373337
*****
Predicción: 21.37680783063534
Etiqueta verdadera: 28.15288162
*****
Predicción: 13.73558591676715
Etiqueta verdadera: -3.145241022
*****
Predicción: 22.07369589837863
Etiqueta verdadera: 20.06635475
*****

```

Figura 23. Salida de datos de memorización

```

*****
Predicción: 12.814310670946567
Etiqueta verdadera: 4.394617081
*****
Predicción: 1.5810988136864417
Etiqueta verdadera: 10.45493221
*****
Predicción: 16.691890443724716
Etiqueta verdadera: 4.705832481
*****
Predicción: 28.608682535468446
Etiqueta verdadera: 3.955220222
*****
Predicción: 12.777324322187464
Etiqueta verdadera: 31.56133461
*****
Predicción: 21.032829779202824
Etiqueta verdadera: 10.43664551
*****
Predicción: 22.56277180063882
Etiqueta verdadera: 22.15491867
*****
Predicción: 20.358287766227303
Etiqueta verdadera: 18.798172
*****
Predicción: 18.1923639573691
Etiqueta verdadera: 26.25152969
*****
Predicción: 16.18264988816209
Etiqueta verdadera: -1.031606197
*****

```

Figura 24. Salida de datos de implicación

## ANEXO 2. OSD

### OBJETIVOS DE DESARROLLO SOSTENIBLE

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

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

Las aplicaciones de este trabajo en la percepción del estado de las personas, y más concretamente y en la línea del grupo de investigación, en la evaluación de interfaces, resaltan su compromiso con varios de los Objetivos De Desarrollo Sostenible.

En primer lugar, este trabajo tiene una fuerte relación con el ODS 4 (Educación de calidad). Este objetivo pretende aumentar la tasa de escolarización así como el éxito de los niños y adolescentes en la escuela. Ese trabajo puede ser clave para ello, ya que podría jugar un rol clave evaluando como las diferentes tareas educativas afectan a los alumnos, permitiendo mejorar los recursos educativos y el proceso de aprendizaje.

Respecto al ODS 5 (Igualdad de género) y al OSD 10 (Reducción de las desigualdades), este proyecto puede tener aplicaciones muy interesantes. Al ser capaces de evaluar el estado del usuario a lo largo de una tarea, es posible estudiar el impacto que diversas situaciones tienen sobre personas atendiendo a diversos factores, como el sexo, edad, raza o nivel académico, y a través de ello sacar conclusiones sobre posibles sesgos o factores que dificulten la completa integración de todos los usuarios

A su vez, el ODS 8 (Trabajo decente y crecimiento económico) es otro objetivo que se alinea con este proyecto. El uso de esta herramienta en la evaluación de interfaces mejora directamente la experiencia de los usuarios. Debido a la alta integración de las herramientas digitales en el ámbito laboral, la experiencia de usuario es un factor clave que influye directamente en la productividad y satisfacción del empleado. . Más allá, el uso de estas herramientas en aplicaciones comerciales también puede mejorar su popularidad de cara al público, aumentando los beneficios y el crecimiento de la empresa.

El OSD 9 (Industria, innovación e infraestructuras) es quizá el que mayor relación tiene con este proyecto. A parte de las aplicaciones en evaluación de interfaces, este proyecto abre la puerta a un amplio abanico de posibilidades gracias a la integración de esta herramienta en diferentes entornos. Esto comprende desde aplicaciones que detecten el estado del usuario y modifiquen su comportamiento en tiempo real para adaptarse a sus necesidades, hasta su integración en proyectos de IoT que puedan controlar componentes de hardware y hacer que el entorno físico del usuario actúe de acuerdo a su estado mental.

Como conclusión, este proyecto pretende en primer lugar mejorar la experiencia del usuario respecto al uso de interfaces, lo que en una sociedad altamente digitalizada puede tener un gran impacto en cuanto a la calidad de vida y el desarrollo de las personas; y en segundo lugar permite el desarrollo de proyectos que se adapten al usuario según su estado cognitivo – emocional, lo que permite cumplir necesidades de las personas de forma más directa. Estas aplicaciones reflejan el compromiso del proyecto con el desarrollo sostenible, la igualdad y la mejora de la calidad de vida de las personas.