



industriales
etsii

Escuela Técnica
Superior
de Ingeniería
Industrial

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería Industrial

Problema “Motor imagery” con dispositivo ActiChamp

TRABAJO FIN DE GRADO

GRADO EN INGENIENIERIA ELECTRONICA INDUSTRIAL Y
AUTOMATICA

Autor: Francisco José Martínez Moreno
Director: Julio José Ibarrola Lacalle
Codirector: José Manuel Cano Izquierdo

Cartagena, 27 de Septiembre de 2017



Universidad
Politécnica
de Cartagena



Índice

1. Introducción.....	2
1.1. Motivación.....	2
1.2. Objetivos.....	3
2. Cascos de señales	5
2.1. Estado del arte del hardware	6
2.2. Estado del arte de las aplicaciones.....	10
2.3. Dispositivo ActiCHamp.....	14
2.4. Pruebas realizadas.....	21
3. Base de datos.....	24
3.1. Procedimiento.....	28
3.2. Motor Imagery.....	30
4. Modelo.....	33
4.1. Adecuación de datos.....	34
4.2. Preprocesamiento.....	36
4.3. Procesamiento de datos.....	38
4.4. Algoritmos de clasificación.....	41
4.5. Validación.....	42
5. Conclusiones y análisis de resultados.....	44
5.1. Análisis de resultados.....	44
5.2. Conclusión.....	46
6. Bibliografía.....	49
7. Anexos.....	52
7.1. Anexo 1. Programa para generación de estados.....	52
7.2. Anexo 2. Código para generación de vectores de etiquetas.....	57
7.3. Anexo 3. Guía rápida para adquisición de señales.....	59
7.4. Anexo 4. Guía rápida para procesamiento de datos.....	61
7.5. Anexo 5. Guía rápida para clasificación.....	62
8. Agradecimientos.....	63





Índice de figuras

Figura 1: Emotiv EPOC.....	6
Figura 2: Emotiv Insight.....	7
Figura 3: Casco OpenBCI.....	7
Figura 4: NeuroSky Mindwave.....	8
Figura 5: BrainLink Pro.....	8
Figura 6: Dispositivo ActiCap.....	9
Figura 7: Fases de un BCI.....	11
Figura 8: Silla de ruedas controlada con la mente.....	12
Figura 9: Neuroprótesis de mano.....	13
Figura 10: MindBall.....	14
Figura 11: Dispositivo ActiCHamp.....	14
Figura 12: Esquema del electrodo utilizado.....	15
Figura 13: Imagen real del electrodo.....	16
Figura 14: Disposición de los electrodos utilizada.....	17
Figura 15: Dispositivo ActiCHamp(arriba) y ActiPower (abajo).....	18
Figura 16: Montaje completo.....	19
Figura 17: Interfaz del PyCorder.....	20
Figura 18: Visualización de señales en PyCorder.....	21
Figura 19: Potencial evocado P300.....	24
Figura 20: Ventana de configuración de PyCorder.....	26
Figura 21: Aplicación de consola de generación de estados.....	27
Figura 22: Formulario del experimento.....	29
Figura 23: Actividad cerebral.....	30
Figura 24: Interferencias en la señal EEG.....	33
Figura 25: Gráfica del canal 1.....	35
Figura 26: Gráfica de los canales 3 y 4.....	36
Figura 27: Fichero con los nombres de todos los sensores.....	37
Figura 28: Tipos de filtrado para una EEG.....	38
Figura 29: Vector de etiquetas.....	40



Figura 30: Densidad Espectral de Potencia (PSD)	40
Figura 31: Aciertos medios de los experimentos	43
Figura 32: Porcentajes de acierto en tanto por uno del sujeto 1	44
Figura 33: Matiz de confusión	45



CAPÍTULO 1.

INTRODUCCIÓN



1. Introducción.

Brain Products es una empresa que fue fundada en 1997, con el objetivo de dar soporte a los problemas neurofisiológicos. Desde entonces, dicha empresa se dedica a la venta de dispositivos relacionados con la adquisición de señales cerebrales que se utilizan en el campo médico.

Todos los dispositivos de Brain Products cuentan con unas características excepcionales, ya que se tratan de aparatos que cumplen con la normativa pertinente que habilitan su uso para medicina e investigación (93/42/EEC).

Es por esto por lo que la Universidad Politécnica de Cartagena adquirió este casco, aunque debido al paso del tiempo, esta unidad ha quedado en desuso.

Debido a esta obsolescencia, se ha considerado que era necesario realizar una puesta a punto del ActiCHamp, pudiendo así, dar pie a futuras investigaciones que necesiten de su uso.

1.1. Motivación.

El departamento de Ingeniería de Sistemas y Automática disponía del casco de señales ActiCap con el amplificador ActiCHamp, el cual se encontraba en desuso, ya que algunos de los componentes no se encontraban dentro del maletín de Brain Products. Otro motivo por el cual no se había realizado antes una puesta a punto del dispositivo, es que no se había encontrado la manera de establecer una conexión entre el casco y Matlab.

Dicho casco tiene un gran potencial y está dotado de unas características dignas de reconocimiento. Sin embargo, era necesario explotar todas sus funcionalidades y hacer que estuviera completamente operativo para su uso.

Otro impulsor para la realización de este proyecto ha sido ver la gran cantidad de aplicaciones prácticas en el mundo del BCI, que día a día se están convirtiendo en técnicas más accesibles para todo el mundo.

Además, otra gran motivación ha sido el interés mostrado por el mundo del BCI tanto por parte de mis directores de proyecto como por la mía.



1.2. Objetivos.

El principal objetivo que se pretende alcanzar en este Trabajo de Fin de Grado es la utilización del dispositivo ActiCHamp para la investigación en el campo del BCI. Para esto, se realizará una puesta en marcha del dispositivo ActiCHamp y de todos sus elementos. Además, se realizará una comprobación del funcionamiento de nuestro dispositivo mediante la aplicación de algoritmos neurodifusos de clasificación para el estudio del *Motor Imagery*. Estos estudios constan de tres pruebas distintas:

- Pensar en mover la mano derecha.
- Pensar en mover la mano izquierda.
- Pensar en una palabra que empiece por una letra determinada.

Para conseguir mantener una homogeneidad a lo largo de las distintas pruebas, dichos experimentos se realizarán utilizando unas condiciones comunes a todos los pacientes.

Las fases a cumplimentar para satisfacer este proyecto son las siguientes:

- Estudio del hardware y puesta en marcha.
- Construir una base de datos con las señales obtenidas.
- Aplicación de algoritmos neurodifusos de clasificación.
- Análisis y discusión de los resultados obtenidos.

Proponer futuras mejoras y posibles aplicaciones.



CAPÍTULO 2.

CASCOS DE SEÑALES



2. Cascos de señales

Cada vez están apareciendo nuevas tecnologías y métodos para la adquisición y evaluación de señales procedentes de los impulsos cerebrales, estas tecnologías dan pie a lo que se conoce como el *Brain computer interface* (BCI).

Hoy en día, los sistemas BCI se consideran una herramienta con un gran potencial para establecer alternativas de comunicación, reestablecer funciones y ofrecer procesos de rehabilitación a pacientes con discapacidad neuromotora (Fgcsic.es, 2012).

Las interfaces cerebro-ordenador miden la actividad de las neuronas para obtener la señal que luego se procesará. Se diferencian dos tipos de dispositivos según el método de obtención de la señal:

- Dispositivos invasivos: La medición se realiza directamente desde el cerebro del usuario, con lo que es necesario realizar una intervención quirúrgica. El sensor puede penetrar la corteza cerebral de forma que mide la actividad eléctrica de neuronas individuales, o bien, puede colocarse en la superficie del córtex para medir la actividad eléctrica de grupos de neuronas. La señal obtenida es muy nítida como consecuencia de tomar las medidas directamente desde el cerebro, aunque debido a los riesgos que supone la intervención y también a consideraciones éticas, la mayor parte de las investigaciones con aplicaciones en rehabilitación se realizan con animales.
- Dispositivos no invasivos: El sensor se coloca de manera que no se realiza una intrusión sobre el cuerpo humano. Son los dispositivos más utilizados en la tecnología BCI y claramente con menos riesgos para el usuario que en los métodos invasivos debido a que no es necesario realizar ninguna intervención quirúrgica. La actividad eléctrica se mide en la superficie del cuero cabelludo (electroencefalografía EEG). La señal obtenida es la superposición de todas las neuronas del cerebro (no de neuronas individuales o grupos localizados) y tiene una resolución más pobre debido a que el cráneo del usuario debilita y distorsiona las señales generadas por las neuronas.

El EEG no es el único método utilizado para la adquisición de datos, también se utilizan los siguientes procedimientos:

- Electroencefalografía (EEG)
- Magnetoencefalografía (MEG)
- Tomografía por emisión de positrones (PET)



- Resonancia magnética funcional (fMRI)
- Imagen óptica

Tanto PET como MEG, fMRI e imagen óptica necesitan de una tecnología cara y sofisticada. Además, las tres últimas mencionadas necesitan de monitorizar el flujo sanguíneo, por lo que resultaría imposible la comunicación en tiempo real.

Sin embargo, la EEG es la más utilizada, ya que es tiene una buena respuesta temporal, es barata y es relativamente portátil. También cabe destacar que su uso es más sencillo que las técnicas que se basan en resonancia magnética, es por tanto que la accesibilidad a estas tecnologías EEG es cada vez mayor, teniendo cada vez un mercado más amplio y al alcance de todos.

2.1. Estado del arte del hardware.

Como ya se ha comentado, el mercado del BCI cada vez es más extenso y accesible a todo el mundo, por lo que cada vez están apareciendo más dispositivos capaces de captar las señales procedentes del cerebro.

Debido a que cada vez surgen más aplicaciones disponibles para estos sistemas, ya no tienen un objeto de estudio únicamente médico, por lo que es más fácil encontrar dispositivos de bajo coste y completamente configurables.

El Departamento de Ingeniería de Sistemas y Automática cuenta con algunos dispositivos de bajo coste, así como dispositivos médicos para la adquisición de señales.

Uno de estos cascos es el Emotiv EPOC, el cual consta de 14 canales y de conexión inalámbrica. Este casco está diseñado para la investigación y el avance en las aplicaciones BCI. En la **Figura 1** podemos ver una ilustración del dispositivo.



Figura 1: Emotiv EPOC.



Por otra parte, el departamento cuenta con el Emotiv Insight, un casco de bajo coste, con 5 electrodos, el cual está destinado para un uso diario. Este dispositivo cuenta también con conexión inalámbrica y está optimizado para producir señales limpias y robustas (Emotiv, 2017).

En la **Figura 2** podemos ver el Emotiv Insight.



Figura 2: Emotiv Insight

También debemos mencionar en este apartado a los cascos de OpenBCI, empresa que está introduciendo el concepto de un casco de señales, que sea programable y configurable. Este dispositivo tiene la carcasa impresa en 3D, 8 canales de entrada, y frecuencia de muestreo de hasta 16Kz. Además, cuenta con conexión inalámbrica y slot para tarjeta SD para guardar las sesiones realizadas. Pero la ventaja fundamental de este dispositivo es el precio, el cual es mucho más bajo que el de los cascos anteriormente mencionados. En la **Figura 3** podemos ver este dispositivo.

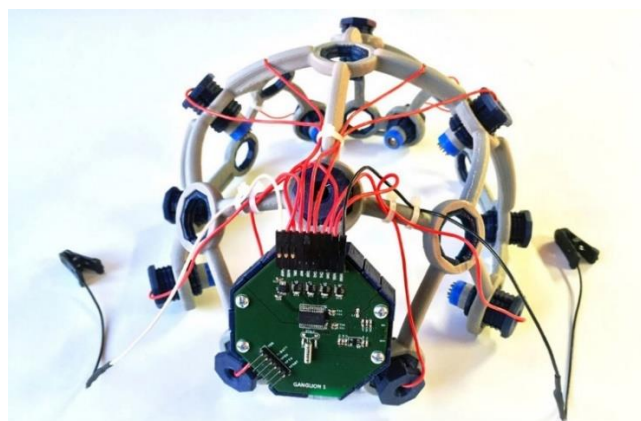


Figura 3: Casco OpenBCI



Por otra parte, tenemos el casco NeuroSky Mindwave, el cual se caracteriza por ser muy ligero y portátil. Este casco está intentando abrirse paso en el mundo del BCI, además, está muy enfocado hacia el mundo de la educación y los videojuegos (además de la investigación). También cuenta con una batería de 8 horas de duración. La gran ventaja de este dispositivo es su precio (el cual ronda los 80€). En la **Figura 4** se muestra una imagen de este dispositivo.



Figura 4: NeuroSky Mindwave

Otro dispositivo de la marca NeuroSky es el BrainLink Pro, el cual está orientado a mejorar el bienestar mental mediante aplicaciones para Smartphone y juegos, tiene unas características mejores que el casco anterior, por consiguiente, su precio es 100€ mayor al anterior. En la **Figura 5** se puede ver el BrainLink Pro.



Figura 5: BrainLink Pro



Por ultimo debemos mencionar a los dispositivos de la marca Brain Products, los cuales están dedicados a la investigación de problemas neurofisiológicos. Uno de sus cascos es el ActiCap. Su principal campo de aplicación es el médico, por lo que su precio es el más elevado de los cascos mencionados, sin embargo, sus características son las mejores, cuenta con frecuencias de muestreo de hasta 100kHz, con un máximo de 160 canales. Su carcasa no es rígida, es una malla donde se pueden colocar los electrodos libremente (pudiendo cambiar su configuración).

En la **Figura 6** podemos ver una ilustración de este dispositivo.



Figura 6: Dispositivo ActiCap.

En este apartado también tenemos que hablar de los tipos de electrodos que existen:

- Electrodos sobre el cuero cabelludo.
- En la base del cráneo.
- Sobre el cerebro expuesto o en las profundidades del mismo.

Con tal distribución podemos clasificar los electrodos en tres tipos:

- Superficiales: Aplicables sobre el cuero cabelludo. La señal resultante es el electroencefalograma.
- Especiales o basales: Aplicables en la base del cráneo sin necesidad de procedimientos quirúrgicos.



- Neuroquirúrgicos: Porque necesitan de este medio para su implantación. Pueden ser:
 - Corticales: Electroencefalograma.
 - Intracerebrales: Estereoecefalograma.

En la realización de las pruebas se utilizarán electrodos superficiales. Éstos deben de ser de fácil y rápida colocación y remoción, indoloro y permanecer colocado sin moverse durante el tiempo del experimento.

Existen tres tipos de electrodos superficiales:

- Electrodo adhesivos.
- Electrodo de contacto.
- Electrodo de aguja.

En nuestras pruebas, se utilizarán electrodos de contacto, los cuales son sujetos con una banda elástica y constan de un tubo de plata enroscado a través de un soporte de plástico. Éstos son más incómodos que los adheridos y no están recomendados para estudios de sueño o para niños.

Por último, podemos afirmar que hay una gran variedad de dispositivos con características muy dispares y de precios completamente diferentes. Este mercado se encuentra en auge por lo que cada vez aparecen mayor número de dispositivos con buenas características y con precios cada vez más bajos.

Al igual que con los cascos, veremos en el siguiente apartado como las aplicaciones que encontramos para esos dispositivos también se encuentran en constante evolución.

2.2. Estado del arte de las aplicaciones.

Para resumir a groso modo todos los procesos a realizar en un BCI tenemos que acudir a la **Figura 7**, donde se muestran todas las fases que se han de seguir para pasar de una señal cerebral a una aplicación práctica.

En el siguiente apartado podemos ver algunas de las posibles aplicaciones que tienen los BCI, desde aplicaciones médicas hasta videojuego. Pero para llegar a ese punto se necesita una señal limpia y bien procesada, de ahí que se tengan que aplicar todos los filtros y algoritmos neurodifusos para clasificar los distintos pensamientos, asignándole así una tarea a cada uno de ellos.

Como podemos ver en la **Figura 7** también es necesario realizar una fase de entrenamiento, donde se obtienen los patrones personalizados para cada



usuario. Estos patrones serán detectados en el experimento final y serán traducidos a comandos.



Figura 7: Fases de un BCI.

Los sistemas BCI pueden resultar de gran utilidad, ya que facilitan una nueva forma de interactuar con los diferentes dispositivos presentes en su entorno habitual. Así, es posible satisfacer algunas necesidades fundamentales de comunicación, confort, ocio y desplazamiento. Por tanto, estos sistemas contribuyen a un incremento de la autonomía de las personas en situación de dependencia, mejorando tanto su calidad de vida como su integración en la sociedad.

Las aplicaciones más habituales están dirigidas a facilitar la comunicación, el control de sillas de ruedas y prótesis o el control del entorno. También se han desarrollado aplicaciones orientadas al control del ordenador y la navegación a través de Internet.

Las aplicaciones orientadas a facilitar la comunicación se han desarrollado a partir de sistemas BCI, que presentan al usuario las letras del abecedario en forma de matriz o de teclado de ordenador. El usuario forma palabras y frases mediante la selección de las letras que las conforman. La técnica de selección de letras y caracteres se ha implementado con sistemas BCI de varios tipos: basados en P300, en potenciales corticales lentos o en imágenes motoras. Actualmente, se está dedicando un gran interés y esfuerzo en el desarrollo de sistemas que combinen la selección de letras con la navegación por Internet, de forma que



existen aplicaciones BCI para la publicación de mensajes en Twitter y navegadores BCI para la exploración de páginas web.

También se han empleado diferentes tipos de sistemas BCI, basados en P300 o en imágenes motoras, para el control del movimiento de sillas de ruedas, como vemos en la **Figura 8**. Dichos sistemas permiten activar o parar el movimiento de la silla, girarla a la derecha o a la izquierda e incluso desplazarse hasta un punto cercano que el usuario ha seleccionado.



Figura 8: Silla de ruedas controlada con la mente.

Además, las sillas están dotadas de sensores y otros dispositivos que permiten detectar posibles obstáculos y ayudan a esquivarlos para garantizar siempre la seguridad del usuario. Una aplicación similar, de control de trayectorias, movimientos y giros, que ha sido implementada con sistemas BCI, es la de teleoperación de robots, es decir, control de robots que pueden estar ubicados a miles de kilómetros de distancia.

También se está investigando en la aplicación de sistemas BCI orientados a la realización de tareas mentales que mejoren las capacidades cognitivas de los usuarios. Las primeras aplicaciones desarrolladas resultan prometedoras, aunque desde el punto de vista del entrenamiento cognitivo y del retraso de los efectos del envejecimiento, queda aún un largo camino por recorrer.

El BCI puede ayudar también a las personas a comunicarse mediante un modelo computacional cognitivo capaz de realizar abstracciones lingüísticas



continuamente, y de cotejarlas con las preexistentes, para confirmar o corregir, y aprender continuamente, para posibilitar la comprensión y adecuación a las nuevas situaciones comunicativas.

La Interacción Multimodal o Multimodalidad consiste en un proceso en el cual diversos dispositivos y personas son capaces de llevar a cabo una interacción (auditiva, visual, táctil y gestual) conjunta desde cualquier sitio. Mediante estas tecnologías se podría mejorar dicha comunicación, haciendo posible la comunicación con aquellas personas que problemas para interactuar con los demás (Modelo Cognitivo Computacional, 2010).

Las neuroprótesis también están a la orden del día, facilitando el movimiento a aquellas personas que por diferentes motivos ya no pueden mover una parte de su cuerpo. En la **figura 9** podemos ver una prótesis de mano controlada con la mente.

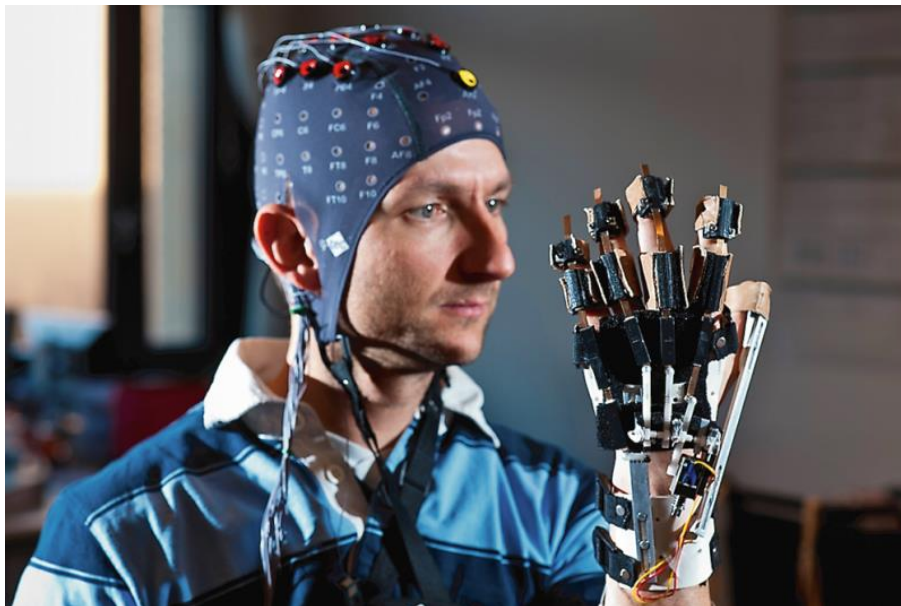


Figura 9: Neuroprótesis de mano.

La técnica del BCI se aplica a la industria del ocio. Un ejemplo sencillo de un dispositivo BCI en este ámbito es MindBall (mostrado en la **Figura 10**). Este dispositivo permite el movimiento de una pelota por un tablero y lo hace mediante la detección de relajación del usuario. El MindBall es un juego donde dos personas compiten por el control del movimiento de una pelota a través de sus ondas cerebrales, es decir, gana aquel en el que sus ondas sean percibidas como que está relajado y concentrado. Además, como ya se sabe la industria del videojuego se encuentra siempre en alza, por lo que es quizá el área en la que puede ser más explotada (W3c.es, 2017).



Figura 10: MindBall.

2.3. Dispositivo ActiChamp.

En este Trabajo de Fin de Grado utilizaremos el dispositivo ActiChamp el cual se muestra en la **Figura 11**, que fue desarrollado con fines electrofisiológicos (EEG/ERP, BCI, etc.), el cual, dispone de un amplificador de 24 bits que soporta hasta 160 canales, aunque en este proyecto se usarán 32 canales.

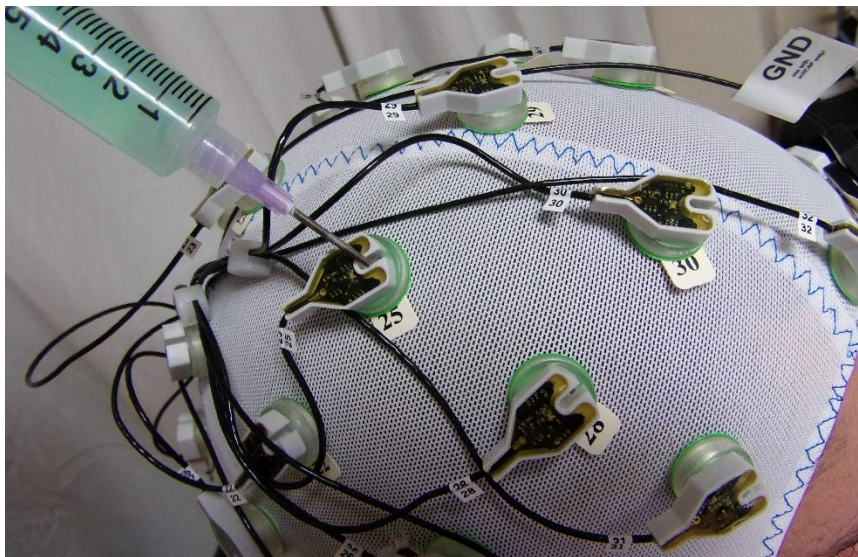


Figura 11: Dispositivo ActiChamp.

El ActiChamp es capaz de realizar grabaciones, donde la frecuencia de muestreo llega hasta los 100KHz.



En lo referente a los electrodos, cabe destacar que en este trabajo se utilizaran 32 electrodos + 1 electrodo (GND) tal y como se muestra en la **Figura 12**. (Brainproducts.com, 2017).

Este casco utiliza electrodos de tubo de plata, los cuales necesitan de un gel electrolítico de alta conducción, los cuales son sujetos mediante una malla, la cual ejerce la presión suficiente para que los electrodos se mantengan constantemente en contacto con el cuero cabelludo.

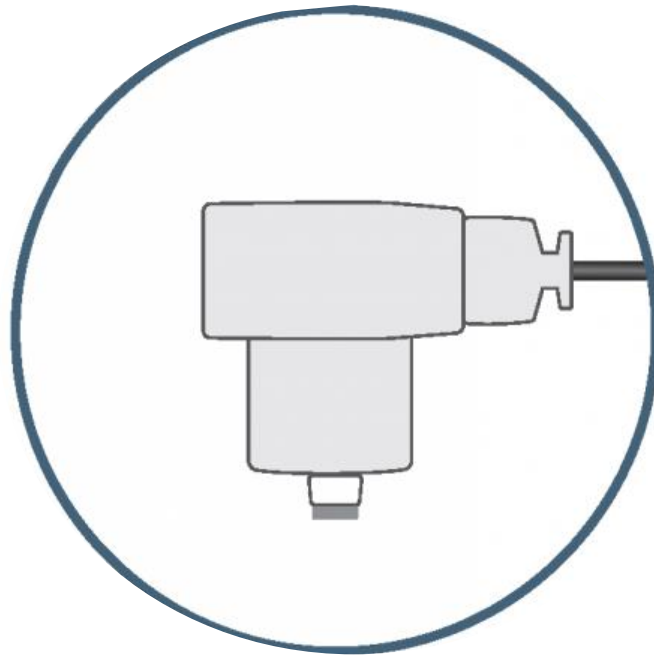


Figura 12: Esquema del electrodo utilizado.

En la **Figura 13** podemos ver una imagen real de uno de los 32 electrodos de nuestro dispositivo, una vez que se encuentra colocado en su sitio se le introduce el gel por el agujero de la parte superior.



Figura 13: Imagen real del electrodo.

La disposición de los electrodos es variable, dependiendo de que queramos señales procedentes de una parte u otra del cerebro. En nuestro caso, usaremos la configuración que aparece en la **Figura 14**.

Ésta configuración no es la única, existen numerosas configuraciones, las cuales son utilizadas dependiendo del objeto de estudio y de las señales de salida que queramos obtener.

Como ya se ha comentado, también se pueden colocar electrodos bajo el ojo por ejemplo con el fin de detectar expresiones faciales, pestañeos, sonrisa, ceño fruncido, etc.

En la **Figura 14** se muestran también el número de electrodo utilizado, así como el nombre de dichos sensores, también se puede observar el recorrido y la distancia que hay entre los electrodos, pudiendo observarse en esta configuración recorridos más cortos que en otras configuraciones donde los electrodos se encuentran situados en las zonas periféricas del cráneo. Dichas configuraciones son comúnmente utilizadas para el estudio de los gestos y de las expresiones faciales.

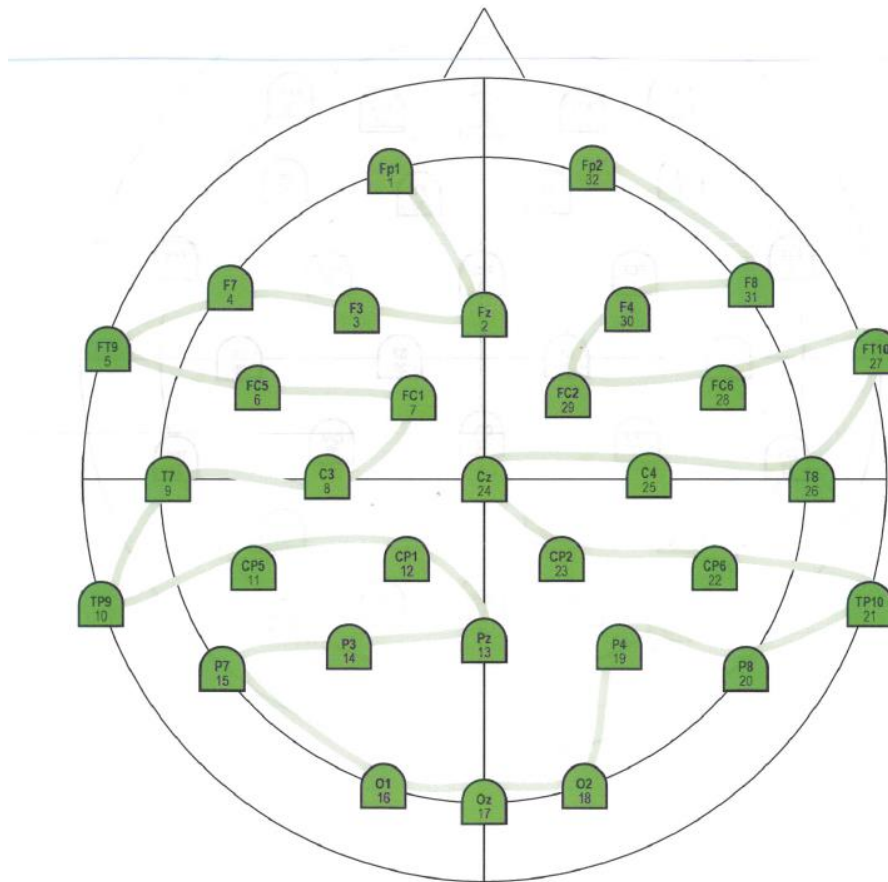


Figura 14: Disposición de los electrodos utilizada.

Por otra parte, el ActiPower es el encargado de suministrar la energía necesaria al amplificador y a todos los electrodos.

Su modo de funcionamiento es idéntico al de una PowerBank, el ActiPower debe de ser cargado con el Charger V9, y posteriormente estará disponible para hacer funcionar todos los elementos durante unas 10 horas.

El conjunto de todos estos elementos nos permite llevar a cabo la detección y adquisición de señales.

Una de las fases de este proyecto es la puesta en marcha del conjunto de dispositivos que nos permiten realizar la grabación de los datos, desde el amplificador, hasta los electrodos. Ésta es una de las fases más importantes, donde debemos ser muy meticulosos para que las condiciones de grabación sean las óptimas.

Para la puesta en marcha es necesario hacer un recuento y montaje de todos los elementos, para ello seguiremos en orden los siguientes puntos.



- En primer lugar, debemos asegurarnos de que el ActiPower se encuentra cargado.
- El amplificador se debe colocar en la parte superior del ActiPower, tal y como se muestra en la **Figura 15**.
- Se deben conectar el ActiPower y el amplificador.
- Los electrodos van conectados al módulo del amplificador y el electrodo GND a la parte frontal del dispositivo ActiCHamp, en caso de que se añadan más canales, se deben añadir más módulos.
- Se deben de colocar los electrodos en la malla que irá colocada en la cabeza del usuario. Se ha de tener en cuenta la importancia de la limpieza de los electrodos, ya que la suciedad puede generar ruido en la señal obtenida.
- En la **Figura 16** podemos observar su montaje final, donde el equipo estaría preparado para comenzar la grabación.
- Ha de conectarse el amplificador al ordenador, mediante conexión USB.
- Se ha de añadir gel conductor entre el cuero cabelludo y el electrodo, con el fin de que exista una conductividad que permita obtener una señal limpia.
- Por último, debemos evitar movimientos bruscos, evitando así posibles tirones de los cables de los electrodos, también debemos de tener cuidado de que todos nuestros dispositivos se encuentren en un lugar fresco y seco, evitando así que el equipo se estropee o no funcione de forma adecuada.
- Cabe destacar que en nuestros experimentos el electrodo GND se colocará en la parte frontal de la cabeza del sujeto, en otros experimentos distintos puede ser colocado en la parte lateral del cráneo, en la parte inferior del ojo, o incluso, en la mano.



Figura 15: Dispositivo ActiCHamp (arriba) y ActiPower (abajo).



En la **Figura 16** podemos ver como es el montaje completo del caso.

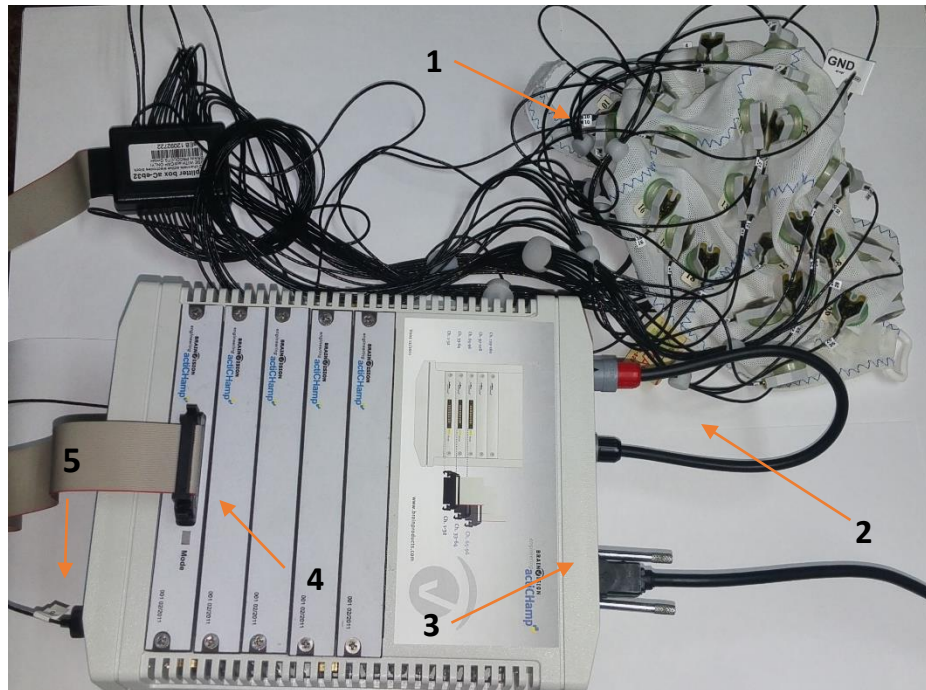


Figura 16: Montaje completo; 1) ActiCap 2) Conexión entre el amplificador y el ActiPower 3) Conexión USB 4) Conexión entre el amplificador y el ActiCap 5) Electrodo GND.

Del mismo modo que se realiza una puesta en marcha “física” de nuestro sistema, también es necesario hacer lo correspondiente con el software.

Con el fin de poder grabar las señales cerebrales, debemos de utilizar el programa PyCorder.

PyCorder está basado en el lenguaje de programación Python, además todo el software es open source, por lo que se puede configurar libremente, pudiendo añadir o quitar módulos que nos permiten utilizar más electrodos.

PyCorder es utilizado en conjunto con el amplificador ActiCHamp, con los electrodos activos y con el PC.

Junto con PyCorder se han de instalar las siguientes librerías de Python:

- NumPy 1.4.1 (Numerical Library)
- SciPy 0.8.0 (Scientific Library)
- PyQt 4.5.4 (GUI)
- PyQwt 5.2.0 (Plot Library)
- lxml 2.2.4 (XML Library)



Además de PyCorder y las librerías de Python es necesario instalar los drivers del dispositivo ActiCHamp.

Una vez que se hayan instalado todos los archivos, podremos acceder al programa donde se realizan las grabaciones, dentro de este programa se deben configurar los numerosos parámetros, tales como la frecuencia de muestreo, la posibilidad de añadir filtros de corte, elegir el electrodo de tierra, seleccionar el directorio para guardar las grabaciones, seleccionar opciones para plotear las señales, aplicar corrección de vaselina, etc.

También tenemos diferentes modos para utilizar el ActiCHamp como podemos observar en la **Figura 17**.

- Default Mode
- Shielding Mode
- Impedance Mode
- Test Mode
- Start/Stop

También tenemos la opción de cargar o guardar las configuraciones realizadas, para ello tenemos que pinchar en el menú desplegable de la parte superior, donde nos aparecen las opciones de cargar o guardar una configuración determinada.

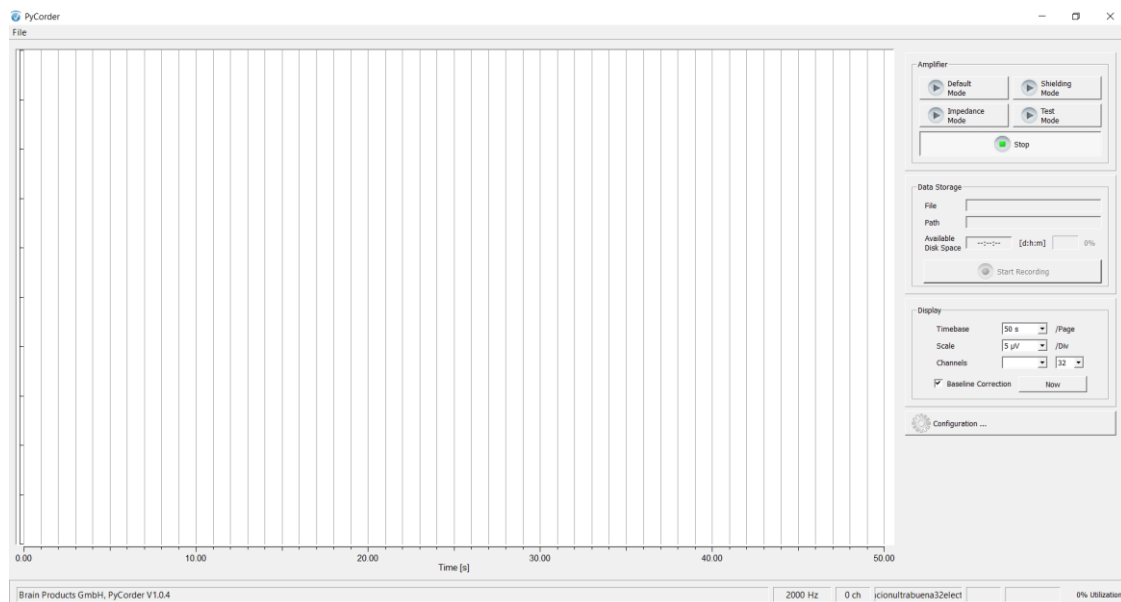


Figura 17: Interfaz del PyCorder.



En la parte inferior de la **Figura 17** nos aparece la información básica, la frecuencia utilizada para la grabación, el tanto por ciento de uso del amplificador, la tensión suministrada por la batería y el número de canales. Además, en el caso de que se produjese un error, se notificaría en rojo en la parte izquierda.

Una vez que se ha realizado una puesta en marcha tanto del software como del hardware podremos realizar nuestras grabaciones y visualizaciones, tal y como se muestra en la **Figura 18**.

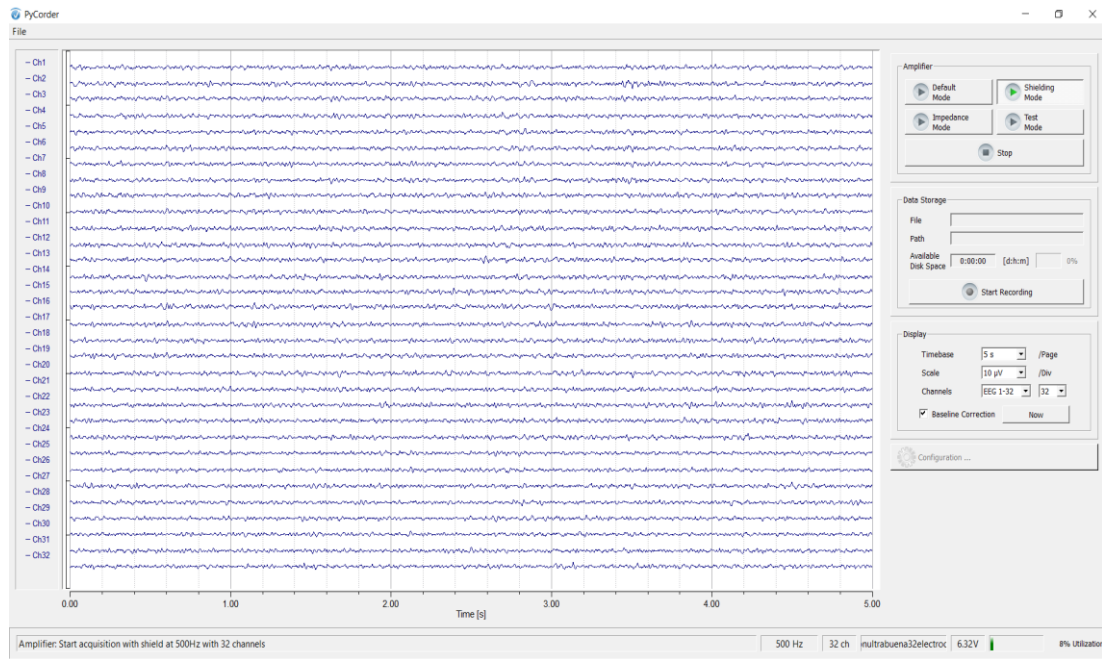


Figura 18: Visualización de señales en PyCorder.

2.4. Pruebas realizadas.

Una vez que se ha realizado una puesta en marcha completa del dispositivo, se han de hacer las pruebas pertinentes para comprobar su correcto funcionamiento. La finalidad de este trabajo de fin de grado, como ya se ha comentado antes, es la adecuación del casco, así como su comprobación mediante su investigación del problema *Motor Imagery*.

Por tanto, se han realizado pruebas antes de realizar los experimentos definitivos, donde principalmente se ha realizado una comprobación de la conductividad de los electrodos.

También se han realizado pruebas para comprobar que todos los canales funcionan correctamente, comprobando uno por uno como se comportaba la



señal ante un estímulo muscular. Se ha utilizado este estímulo ya que es el que más influencia tiene en la señal.

Una vez que se han realizado unas cuentas pruebas y se ha validado el buen funcionamiento aparente del dispositivo se ha de proceder a la comprobación del funcionamiento más exhaustivo. Dichos experimentos quedarán totalmente definidos en el siguiente apartado.



CAPÍTULO 3.

BASE DE DATOS



3. Base de datos.

La realización o imaginación del movimiento de determinadas partes del cuerpo tiene efectos detectables en la señal cerebral y se producen los llamados ritmos sensorimotors. Por otra parte, cuando se produce un estímulo, se produce una respuesta detectable P300.

La mayoría de gente presenta un ritmo cerebral dominante cuando está en reposo entre las frecuencias 8-12Hz (sin la realización de movimiento). A esta oscilación se le denomina ritmo mu. Dicha oscilación viene asociada con ritmos beta entre las frecuencias de 18-25Hz.

El objetivo final de nuestro experimento, es realizar un análisis en frecuencia entre 8 y 30 Hz pudiendo ver a si tanto ritmos mu como ritmos beta, ya que los cambios en estos ritmos se producen cuando se llevan a cabo funciones sensoriales o motoras.

Pero no solo quedarán recogidas en la grabación los ritmos mu y beta, sino que también nos pueden aparecer potenciales evocados P300 (provocados por estímulos externos) dando lugar a cambios de voltaje a lo largo del tiempo como se puede ver en la **Figura 19**.

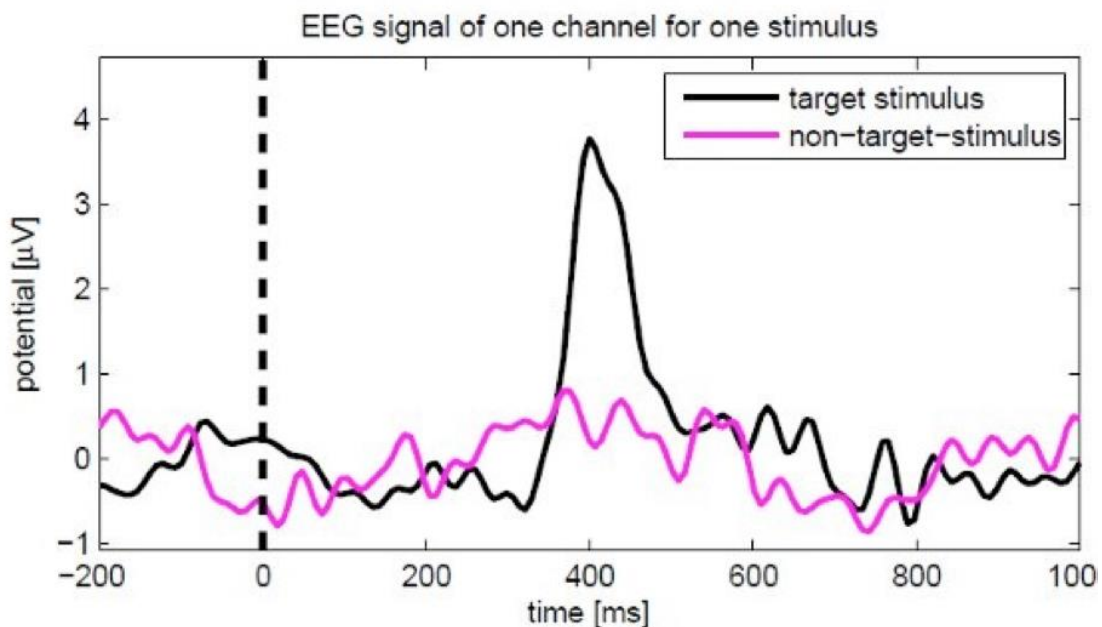


Figura 19: Potencial evocado P300.



El experimento se llevará a cabo con usuarios sanos, los cuales se someterán a 4 sesiones de 4 minutos de duración cada una, realizando un descanso de unos 5-10 minutos entre sesión y sesión.

El usuario se deberá sentar en una silla normal, reposando los brazos sobre las piernas y con el cuerpo recto.

Se realizarán 3 tareas distintas dentro de cada sesión, las cuales irán cambiando cada 15 segundos, dando lugar a 16 cambios. Estas tareas son:

- Imaginar un movimiento con la mano derecha (de forma repetitiva).
- Imaginar un movimiento con la mano izquierda (reiterativamente).
- Pensar en numerosas palabras que empiecen por una letra determinada, la cual cambia el programa de forma aleatoria.

Una vez terminado el experimento completo (las cuatro sesiones), el sujeto deberá rellenar un formulario, con el fin de que, en futuros experimentos, se realicen mejoras de cara al usuario.

Como ya hemos comentado, todos los experimentos se realizarán en condiciones similares, manteniendo la homogeneidad a lo largo de todas las pruebas (lugar, hora del experimento, etc).

Dentro del programa PyCorder, tenemos la pestaña de configuración donde deberemos ajustar los parámetros pertinentes.

En primer lugar, habilitaremos los 32 electrodos que tiene nuestro casco, posteriormente, estableceremos la frecuencia de muestreo en 500Hz, tal y como debe ser en las pruebas de *Motor Imagery*.

En la **Figura 20**, se puede observar la ventana de configuración.

Además, dentro de esta ventana, tenemos que seleccionar el directorio donde se van a guardar las sesiones realizadas, en nuestro caso, guardadas en un dispositivo USB.

También podremos establecer un nombre predeterminado para las sesiones, haciendo así que se enumeren automáticamente.

En nuestro caso se nombrarán como *exp(número de experimento)_(nombre asignado al sujeto)*.

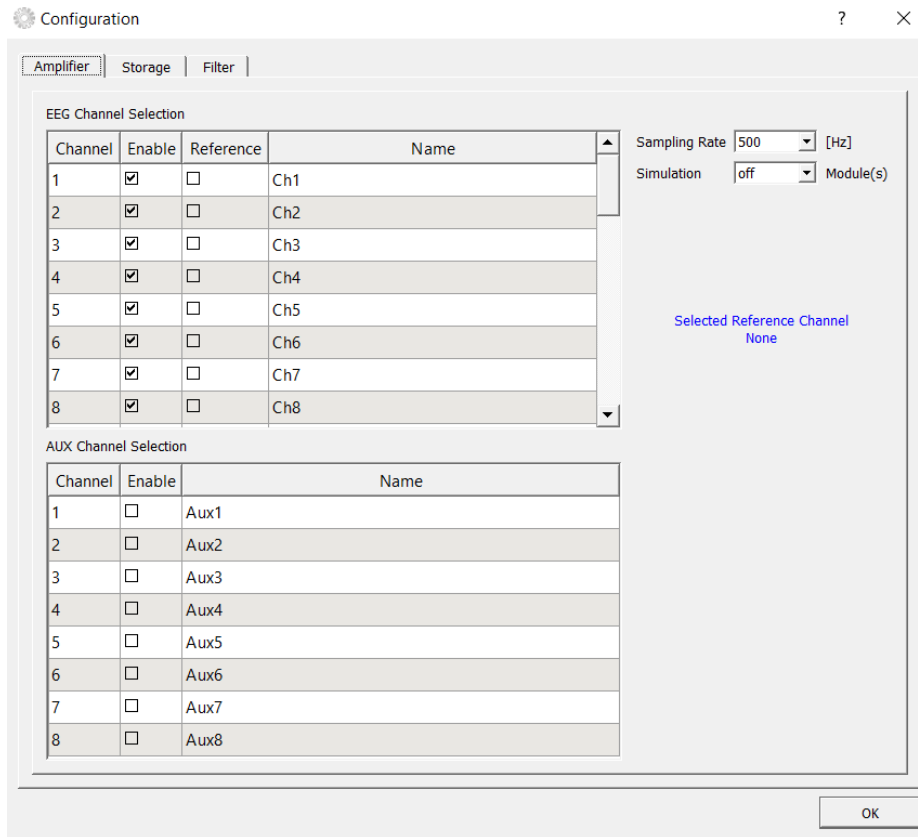


Figura 20: Ventana de configuración de PyCorder.

Como ya se dijo, en cada experimento del problema *Motor Imagery* se iban a realizar tres pruebas distintas (mano derecha, mano izquierda y palabra). Con el fin de generar dichos estados, se corre un programa realizado en C++, donde los estados cambian secuencialmente dependiendo del contenido de un vector (LWP Comunidad de programadores, 2017).

En el **Anexo 1** se puede observar cómo está construido el programa.

Este programa contiene un pequeño menú donde se seleccionará el experimento a realizar.

En la **Figura 21**, se puede observar el menú anteriormente mencionado, además, vemos por pantalla que tenemos que pensar en mover la mano izquierda.



```
"C:\Users\Francisco\Desktop\programa c++\tfg\bin\Debug\tfg.exe"
1) Experimento 1
2) Experimento 2
3) Experimento 3
4) Experimento 4
2
Usted ha seleccionado el experimento 2
<--
```

Figura 21: Aplicación de consola de generación de estados.

Como podemos ver en la **Figura 21** es una aplicación de consola, con el fin de que se pueda ejecutar rápidamente y sin retrasos.

Para la realización definitiva de los experimentos deberemos utilizar de manera simultánea el programa de generación de estados (en C++) y el programa de grabación (PyCorder), uno se encargará de mostrarnos las pruebas a realizar y el otro se ocupará de grabar toda la información.

Pasados los cuatro minutos de la sesión, se debe de parar la grabación de forma manual en el PyCorder.

Todos los archivos deben de ser guardados y enumerados de forma muy cuidadosa, ya que cada usuario generará cuatro archivos (uno por cada experimento).

La importancia a la hora de nombrar los archivos se debe a que en los programas que se utilizan para la realización del modelo, cargan los archivos directamente mediante la función `load()`, por tanto, para agilizar los cálculos y mantener el orden en el procesamiento de datos utilizaremos siempre la misma nomenclatura.

Estos archivos pasarán a formar parte de una base de datos, donde los programas usados para el procesamiento de datos y para la creación del modelo entrarán para coger los ficheros.



3.1. Procedimiento.

La creación de una base de datos es una de las fases más importante de este Trabajo de Fin de Estudios. Se ha de tener en cuenta que es de gran importancia ser cuidadoso con la construcción de la base de datos.

Dicha base de datos será creada con las señales provenientes de distintos usuarios, los cuales serán sometidos a cuatro experimentos, tres para la generación de un modelo, y un cuarto experimento para la comprobación del mismo.

Estas señales serán grabadas con el dispositivo ActiCHamp, intentando hacer los experimentos sobre el mayor número de individuos posibles. Además de estas señales, se importarán y comprobarán señales procedentes de otros dispositivos, como es el Emotiv EPOC.

Todos estos datos serán almacenados según el tipo de dato y según el usuario del que proceda.

En nuestro caso iremos almacenando los datos obtenidos a la salida de todos los subprocesos, por tanto, quedarán almacenados los datos en raw (procedentes de la adquisición de datos mediante Pycorder), los datos.mat (que provienen de la conversión de los datos raw a datos en Matlab), los archivos después de aplicar el Filtro Laplaciano, el vector de etiquetas asociado a cada experimento, y, por último, los datos que se obtienen cuando se hace la PSD (junto con su nuevo vector de etiquetas).

Cabe destacar la importancia de tener el máximo número de usuarios, a ser posible, de diferente edad, sexo y mano hábil.

Una vez que ha finalizado la grabación de un usuario, se pasará un formulario como el de la **figura 22**. La finalidad de esta encuesta es obtener información acerca del usuario y de su opinión acerca del experimento.

Es importante conocer la experiencia de cada uno de los usuarios, y no solo con la finalidad de que el experimento se realice en mejores condiciones, sino con el fin de recopilar datos que nos sirvan posteriormente a sacar conclusiones acerca del problema *Motor Imagery*.

Es importante ver cómo afecta el porcentaje de acierto del modelo (mano derecha, mano izquierda y palabra) con la mano hábil del sujeto, su edad o su sexo.

Además, es interesante ver la capacidad de adaptación de los sujetos al casco, ya que los sistemas BCI no funcionan correctamente con todos los usuarios.



Por último, cabe destacar la importancia de saber si el sujeto está cómodo o no, ya que un sujeto incómodo, se puede traducir en un sujeto inquieto y distraído, lo que empeoraría los resultados.

Bloque 1. Información Personal

Nombre y Apellidos: _____ Sexo: _____

Fecha de nacimiento: _____ Zurdo o diestro: _____

¿Tiene algún conocimiento previo del mundo de las EEG?: Sí No

Bloque 2. Comodidad Emoc

¿Le ha parecido cómodo el procedimiento de colocación del dispositivo BCI?

1. Muy incómodo 2. Incómodo 3. Normal 4. Cómodo 5. Muy cómodo

¿Le ha resultado molesto realizar los experimentos con el casco puesto?

1. Muy molesto 2. Molesto 3. Normal 4. Cómodo 5. Muy cómodo

¿Se ha sentido a gusto y relajado en la posición en la que ha realizado el experimento?

1. Muy incómodo 2. Incómodo 3. Normal 4. Relajado 5. Muy Relajado

Bloque 3. Dificultad de los experimentos

¿Le ha parecido difícil realizar los experimentos?

1. Muy fácil 2. Fácil 3. Normal 4. Difícil 5. Muy difícil

¿Le ha generado cansancio la realización del experimento?

1. Ninguno 2. Muy poco 3. Un poco 4. Bastante 5. Excesivo

¿El tiempo de descanso entre cada sesión ha sido el correcto?

1. Demasiado corto 2. Corto 3. Adecuado 4. Largo 5. Demasiado largo

¿Considera que la realización del experimento ha sido satisfactoria en líneas generales?

1. Para nada 2. Un poco 3. Normal 4. Bastante 5. Completamente

¿En cuál de los tres experimentos le ha resultado más difícil concentrarse?

Mano derecha Mano izquierda Palabra

Para finalizar nos gustaría saber globalmente que le ha parecido la realización del experimento y qué aspectos podríamos mejorar de cara al futuro:

.....

.....

Figura 22: Formulario del experimento.

En este formulario se pueden encontrar los siguientes apartados:

- Información personal.
- Comodidad del casco y del experimento.
- Dificultad del experimento.
- Sugerencias para posibles mejoras.

Una vez que se han realizado todos estos pasos, podemos concluir en que se han llevado a cabo los experimentos pertinentes para decir que ha quedado construida una base de datos con distintos usuarios, realizando la prueba del *Motor imagery*.



3.2. Motor Imagery.

El *Motor Imagery* es definido como aquella tarea cognitiva que se refiere a la reproducción interna, en este caso mental, de un movimiento específico sin ningún otro tipo de estímulo motor, esto es posible gracias al estímulo que se produce en el cerebro a pesar de no realizar ningún movimiento.

En la **Figura 23** se puede observar la actividad cerebral producida con un movimiento y con un pensamiento. Además, se puede ver como la actividad se produce en la misma zona, pero con distinta intensidad.

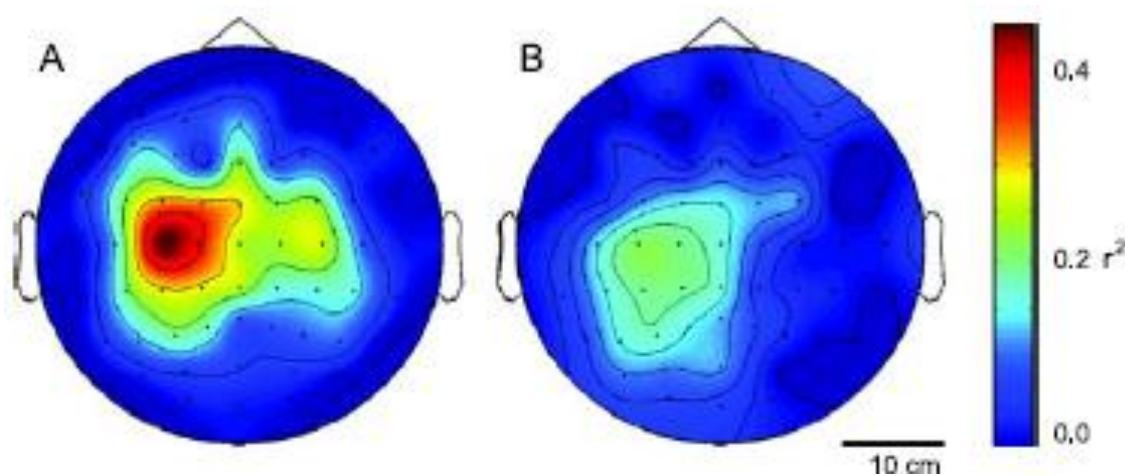


Figura 23: Actividad cerebral.

Los diferentes estudios desarrollados sobre el cerebro, han demostrado que existe una fuerte relación entre la excitabilidad del mismo y la fuerza/debilidad muscular. Entender los procesos excitatorios e inhibitorios que se dan en el cerebro pueden ayudar a mejorar el rendimiento muscular, e incluso, superar los efectos adversos ante una lesión cerebral. (Morales-Osorio and Mejía-Mejía, 2012).

Motor Imagery es un proceso activo mediante el cual se imagina la realización de una actividad física específica, sin la ejecución de la misma. Gracias a estas técnicas se pueden conseguir las mejoras mencionadas anteriormente. Ante la inmovilización de una extremidad causada por un accidente, se ha demostrado que es posible reducir el tiempo de recuperación, así como disminuir también las pérdidas de fuerza y control tras dicha inmovilización.

Existen evidencias también de que el *Motor Imagery* sirve de ayuda para mejorar la calidad de los pacientes que sufren dolores crónicos, aunque se necesita una



investigación de mayor alcance para establecer que existe una correlación evidente entre el *Motor Imagery* y los síntomas de dolores crónicos.

En definitiva, mediante la imaginería motora, se pueden tratar y mejorar algunos problemas que van desde la parálisis física causada por lesiones cerebrales, hasta dolores crónicos. Pero no solo esa es su aplicación, sino que su campo de aplicación está en constante crecimiento gracias a los sistemas que han sido denominados como interfaces de cerebro-computadora o BCI (Brain-Computer Interface) (Gutiérrez-Martínez et al., 2013).

Estos sistemas descifran la intención del usuario de moverse o comunicarse a través del análisis de señales eléctricas cerebrales que incluyen potenciales corticales lentos, potenciales evocados visuales, potenciales P300 y ritmos beta o mu (Técnicasfisioterapia.webnode.es, 2012).



CAPÍTULO 4.

MODELO



4. Modelo.

Una vez que se hayan realizado todos los experimentos, se deben de tratar los datos con el fin de poder trabajar con ellos. Como ya sabemos, el objetivo de este Trabajo de Fin de Grado, es realizar estudios sobre las bases del *Motor Imagery*. El programa que utilizaremos para tratar los datos, será Matlab, ya que nos proporciona gran versatilidad y potencia de cómputo para trabajar con este tipo de datos.

La realización de un modelo es uno de los apartados más importantes dentro de este Trabajo de Fin de Grado, para llevarlo a cabo partimos de que se realizarán tres experimentos con cada usuario, y una cuarta sesión de comprobación.

A todos y cada uno de estos experimentos se le han de aplicar los algoritmos neurodifusos necesarios para obtener una señal limpia y útil para nuestro objeto de estudio.

Con algoritmos neurodifusos nos referimos a un Filtro Laplaciano y a la aplicación de la Power Spectral Density (PSD) o densidad del espectro de potencia, así como algoritmos de clasificación que nos permiten establecer el porcentaje de acierto de nuestro modelo.

Es necesario aplicar el filtrado mencionado anteriormente ya que tenemos numerosas interferencias y muchas fuentes de ruido que se han de eliminar, tales como ruido aportado por la red eléctrica, pestañeos, movimientos oculares o movimientos musculares.

En la **figura 24** se puede observar cómo afectan estas interferencias a nuestra señal.

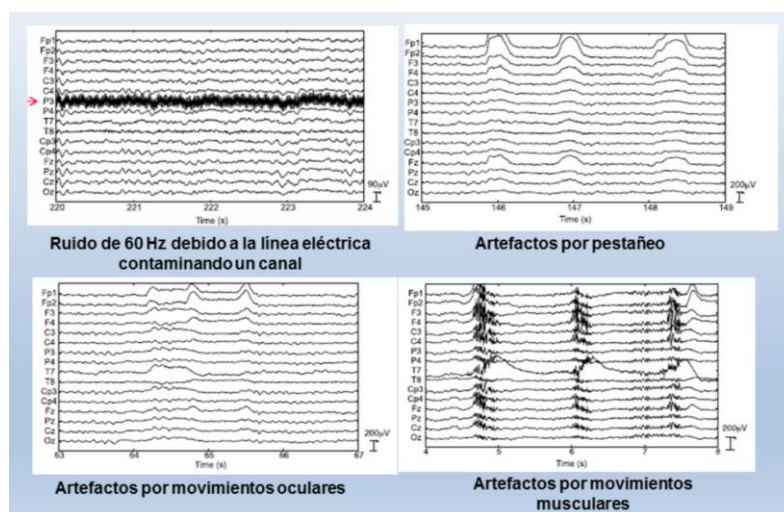


Figura 24: Interferencias en la señal EEG.



Cabe destacar el gran número de interferencias generadas por todos los aparatos electrónicos de nuestro alrededor.

A efectos prácticos el tipo de interferencia que más nos ha influido ha sido la de movimientos musculares, ya que pestañeos y movimientos involuntarios, hacían que la señal sufriese perturbaciones

4.1. Adecuación de datos.

Cada vez que realizamos una grabación con el programa PyCorder, se generan tres archivos distintos, con el mismo nombre y con diferente extensión (.eeg, .vhdr y .vmk). El principal problema que nos aparece es que no tenemos forma alguna de trabajar con estos formatos, por ello, el principal objetivo que tenemos es obtener un formato del tipo “.mat”, para poder abrirlo con Matlab.

Para poder hacer esto último, debo utilizar el programa de Matlab llamado `bva_loadeeg.m`, el cual me permite abrir un fichero .vhdr (proporcionado por PyCorder) y posteriormente, guardarlo como un “.mat”.

Por tanto, si se quisiera guardar el archivo “exp1.vhdr” como “exp1.mat” deberíamos hacer lo siguiente:

- `eeg = bva_loadeeg('exp1.vhdr');`
- `save exp1.mat eeg;`

Una vez que tenemos el fichero .mat, podemos observar que este es una matriz de 32 filas (una fila por cada uno de los canales) por un número de columnas, el cual depende de las muestras que hayamos tomado.

También podremos hacer un plot de cualquiera de los canales. En la **figura 25**, podemos ver la gráfica del canal 1 en una de las pruebas para la puesta en marcha de nuestro dispositivo. En esta gráfica podemos ver la variación de potencial eléctrico a lo largo del tiempo.

Para hacer este plot utilizaremos el siguiente comando

- `Plot(eeg(1,:));`

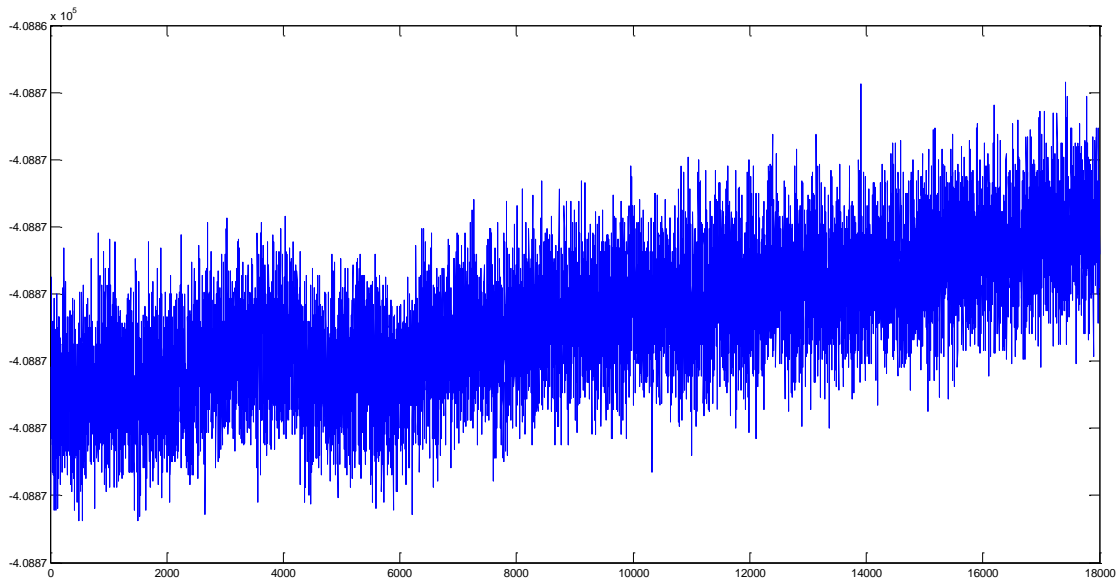


Figura 25: Gráfica del canal 1.

Una de las ventajas que tenemos al trabajar con Matlab y PyCorder es que podemos ver y grabar las señales a nuestro antojo. En otros softwares específicos como es el caso de Emotiv, tendremos que pagar una suscripción para poder hacerlo.

Por la comodidad para futuros experimentos, se han de guardar los datos haciendo la traspuesta, ya que, en los programas, las filas corresponden con el número de muestras tomadas y las columnas con el número de canales de nuestro casco.

Cabe destacar que con el comando `subplot()` de Matlab podemos realizar la visualización de datos de los distintos canales de forma simultánea. Pudiendo ver así como son las formas de las gráficas en un determinado instante de tiempo, tal y como se aprecia en la **Figura 26**.

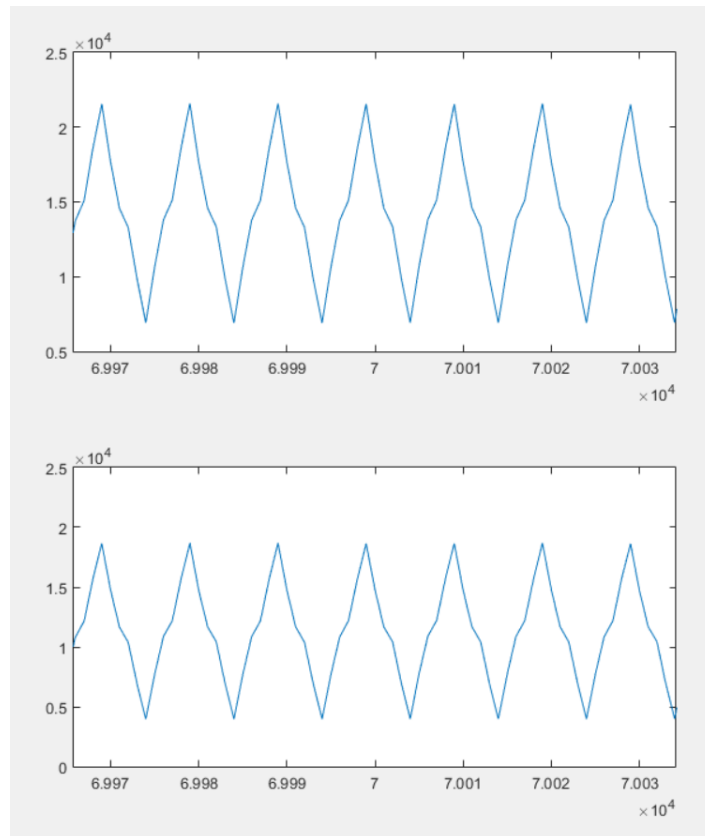


Figura 26: Gráficas de los canales 3 y 4.

4.2. Preprocesamiento.

El filtro Laplaciano que se aplica filtra la señal para minimizar el impacto del ruido que afecta a todos los sensores en diferentes áreas del cerebro (Indexing, 2017).

El código del fichero LaplacianZoneSetV.m se basa en una aproximación al algoritmo original donde se resta a la señal en un sensor una media ponderada de la señal medida en el resto de los sensores, teniendo una menor contribución en esa resta los sensores más lejanos. Por eso se han de definir cuáles son los sensores del casco, para que el sistema sea capaz de extraer las posiciones y desde ahí hacer el cálculo. Dichos sensores se introducen manualmente en un fichero de Matlab, llamado electrodes.mat.

Además de ejecutar estos dos códigos, se debe de cargar al *Workspace* un fichero que contenga el nombre y las posiciones X e Y de todos nuestros sensores.

En la **Figura 27**, podemos ver dicho fichero.



	1	2	3	4	5	6	7	8	9	10	11	12	
1	'FP1'	'FPZ'	'FP2'	'AF3'	'AF4'	'F7'	'F5'	'F3'	'F1'	'FZ'	'F2'	'F4'	'F6'
2													
3													
4													
5													
6													
7													
8													
9													
10													
11													

Figura 27: Fichero con los nombres de todos los sensores.

Para aplicar el Filtro Laplaciano se realizarán los siguientes pasos:

- En primer lugar, debemos comprobar que todos los sensores que utilizemos se encuentran dentro de del fichero de “electrodes.mat”, si no es así, se deben introducir manualmente (introduciendo nombre del sensor y posición X e Y).
- Después, debemos cargar el archivo de “electrodes.mat” en el *workspace*.
- También debemos de cargar el archivo (.mat) del experimento grabado.
- Posteriormente, se debe de aplicar la función del Filtro Laplaciano en todos los sensores para cada instante de tiempo. Para ello escribiremos las siguientes líneas de código:

```
[a,b]=size(eeg)
For i=1:b
X1(:,i)=LaplacianFilterZoneSetV(eeg(:,i),x,y);
End
```

Una vez que se haya realizado todo el cómputo de datos, guardaremos la traspuesta del resultado obtenido, con el fin de tener en las columnas el número de sensores y en las filas el número de muestras (esto se realiza por la comodidad para realizar la PSD).

El Filtro Laplaciano no es el único tipo de filtrado que podemos hacer, en la **Figura 28**.

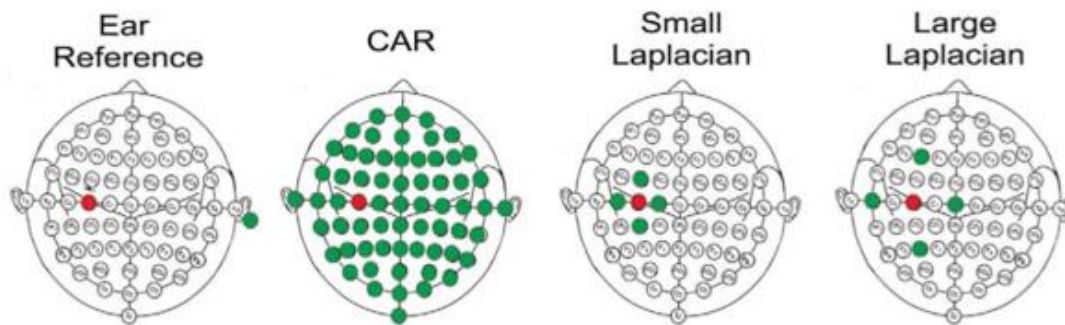


Figura 28: Tipos de filtrado para una EEG.

4.3. Procesamiento de datos.

La Densidad Espectral de una señal es una función matemática que nos informa de cómo está distribuida la potencia de dicha señal sobre las distintas frecuencias de las que está formada.

Con respecto a la PSD, lo que se hace es cambiar la señal del dominio del tiempo al de la frecuencia y ver qué potencia de señal hay en cada frecuencia. Para esto normalmente se define un rango de frecuencias que se le pasa a la función. Por ejemplo, entre 8 y 30 Hz a saltos de 2 Hz (8, 10, 12, 14, 16... 28, 30), lo que devuelve 12 datos por cada canal. Es decir, que se pasa de tener una matriz con 32 canales (32 sensores) a $32 \times 12 = 384$ datos (Es.mathworks.com, 2017).

En este punto se explicará cómo se ha realizado el modelo paso a paso.

Por otra parte, se debe hacer un estudio de la Densidad Espectral de Potencia. Para ello debemos construir un vector de etiquetas, donde se indique en cada instante temporal, en qué se estaba pensando. Para ello, cargaremos el archivo donde se ha aplicado el Filtro Laplaciano. Posteriormente, utilizaremos el script que aparece en el **Anexo 2**. Como ya se ha comentado, este programa nos genera un vector con 3 posibles valores (3, 5 ó 7) dependiendo de si se piensa en mover la mano derecha, la izquierda o si se piensa en la palabra.

Para la realización del siguiente paso debemos tener claros unos conceptos fundamentales. Debemos tener en cuenta que para la realización de la PSD tenemos que crear una ventana de datos. Dentro de esta ventana, se comparan



la matriz de datos y el vector de etiquetas, asignando así una etiqueta a un conjunto de datos determinados. Esto se realiza viendo cual es el dato predominante dentro de la ventana de datos.

Para realizar este proceso debemos tener claro los datos de entrada que tenemos que introducir en la función que realiza la PSD. Estos parámetros son los siguientes:

- Matriz de datos (después del Filtro Laplaciano).
- Frecuencia de muestreo (en nuestro caso 500Hz).
- Vector de frecuencias (de 8 a 30Hz a saltos de 2Hz).
- Tamaño de ventana en número de muestras por ventana (500).
- Salto de ventana, el cual, está determinado en el experimento (500/16).
- Vector de etiquetas.

Si por ejemplo quisiéramos hacer la PSD para para una matriz de datos llamada *fichero1* y un vector de etiquetas llamado *vector1* deberemos introducir en Matlab el siguiente comando:

```
[PSDDATA,PSDWINDOW]=windowsprocessingMatrix(fichero1,500,(8:2:30),500,31,vector1);
```

Una vez introducido este comando, el programa nos devuelve un vector de etiquetas nuevo como el de la **figura 29**, y los datos una vez aplicada la densidad espectral (los cuales se muestran en la **figura 30**).

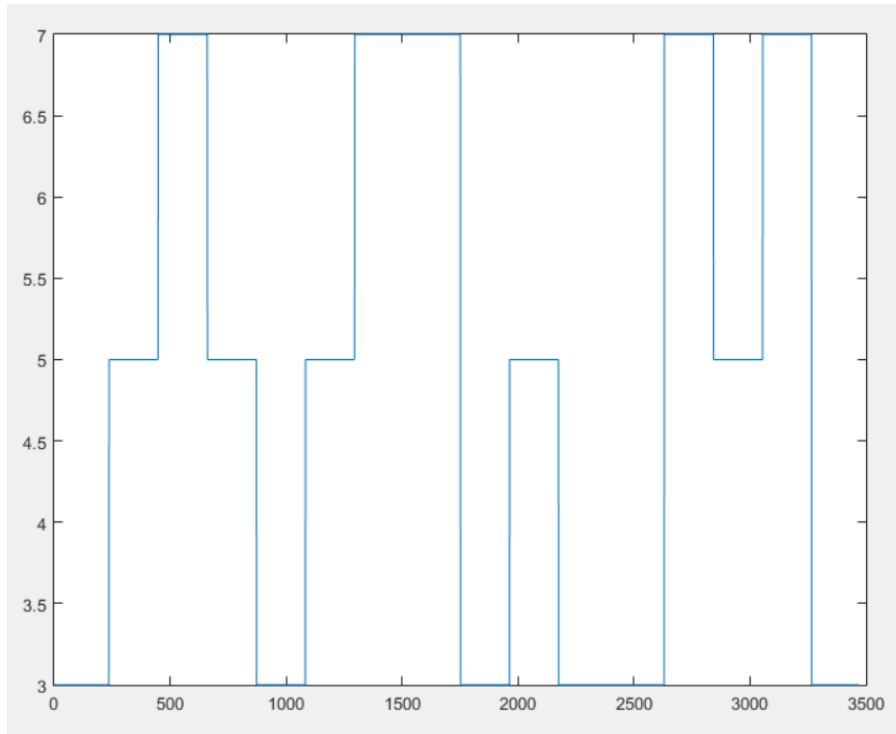


Figura 29: Vector de etiquetas, en el eje X los posibles valores y en el eje Y el tiempo.

Como aquí se puede comprobar, los valores generados, oscilan entre 3, 5 y 7 correspondiendo cada uno de estos valores con los tres posibles estados del *Motor Imagery*.

En la **figura 30**, sin embargo, podemos observar los puntos en los que los que la potencia de la señal en determinadas frecuencias es más elevada.

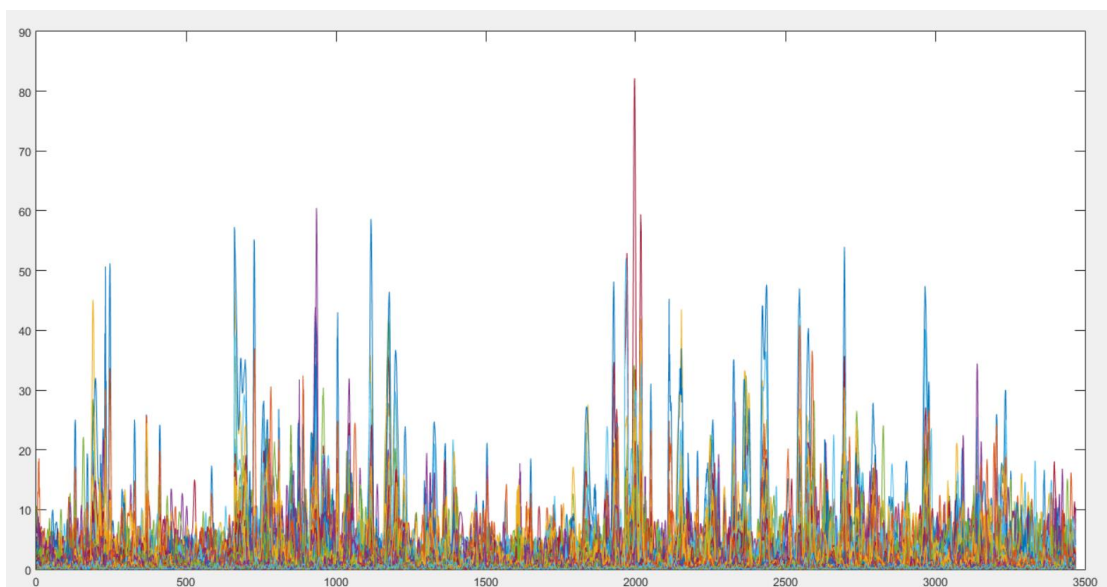


Figura 30: Densidad espectral de potencia (PSD).



4.4. Algoritmos de clasificación.

Una vez que tenemos tanto la ventana de datos como los resultados de la PSD, se puede llevar a cabo la realización del modelo. Para ello debemos realizar una sesión de aprendizaje con cada una de las sesiones utilizadas para la realización del modelo. Por último, se realizará una cuarta sesión para la comprobación de dicho modelo.

Para la realización del modelo se deben seguir los siguientes pasos:

- Debemos modificar el fichero Aprende.m. En primer lugar, debemos cargar en la variable X los datos de la PSD donde las filas son el número de ventanas y las columnas son el número de sensores por el número de frecuencias (12). También debemos copiar en la variable Y el vector de etiquetas. Por último, debemos modificar el parámetro Ar hasta que las unidades utilizadas sean del orden de 300 (a mayor valor de Ar, mayor serán las unidades utilizadas).
- Después se debe de correr el programa Buscar_parámetros.m, este programa se encarga de encontrar los parámetros para la comprobación del modelo. Este programa, se debe ejecutar junto con el anterior, combinándolo de forma que se puedan extraer todos los parámetros. Dicho orden será:
 - Se ejecuta Aprende.m con la sesión 1 y Buscar_parámetros.m con las sesiones 2 y 3 para obtener los parámetros $At(1,2)$, $\delta(1,2)$, $At(1,3)$ y $\delta(1,3)$.
 - Se ejecuta Aprende.m con la sesión 2 y Buscar_parámetros.m con las sesiones 1 y 3 para obtener los parámetros $At(2,1)$, $\delta(2,1)$, $At(2,3)$ y $\delta(2,3)$.
 - Se ejecuta Aprende.m con la sesión 3 y Buscar_parámetros.m con las sesiones 1 y 2 para obtener los parámetros $At(3,1)$, $\delta(3,1)$, $At(3,2)$ y $\delta(3,2)$.
- Una vez que se han obtenido todos parámetros hemos de utilizar el programa SdFasArt_coordenadas_proceso_total.m cambiando los parámetros delta y At por los que hemos obtenido. Quedaría entonces:

```
delta_bak=[delta(1,2) delta(1,3) delta(2,1) delta(2,3) delta(3,1) delta(3,2)];
```



$At_bak = [At(1,2) \ At(1,3) \ At(2,1) \ At(2,3) \ At(3,1) \ At(3,2)];$

- Cuando este programa finaliza su ejecución se muestran por pantalla los porcentajes de acierto para cada una de las pruebas a realizar en el experimento. Los valores de acierto serán discutidos y analizados en el siguiente apartado.

4.5. Validación.

La finalidad de la realización de los estudios de investigación sobre el *Motor Imagery* es la comprobación del buen funcionamiento del casco.

Estadísticamente consideraremos que el casco tiene un mal funcionamiento cuando el índice de acierto sea inferior al 33%, ya que es el porcentaje mínimo de acierto debido al azar.

En nuestro caso podemos considerar que la adecuación del casco ha sido un éxito, ya que el índice de acierto para todos los usuarios ronda entre el 52-56%.

De acuerdo con el índice de acierto, los resultados obtenidos pueden ser considerados como válidos y se discutirán en el apartado siguiente se ha realizado un análisis exhaustivo de los resultados.

En la **Figura 31** podemos ver el índice de acierto medio para todos los sujetos.

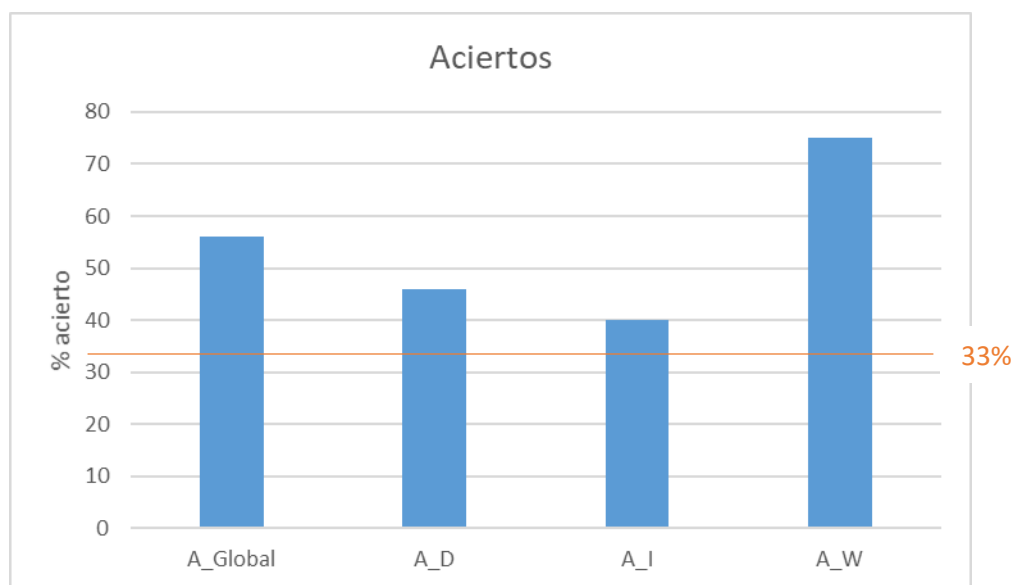


Figura 31: Aciertos medios de los experimentos, aciertos mínimos en naranja.



CAPÍTULO 5.

CONCLUSIONES Y ANÁLISIS DE RESULTADOS



5. Conclusiones y análisis de resultados.

Resulta esencial realizar un análisis de los resultados obtenidos, hacer un recuento de los objetivos cumplidos a lo largo de este proyecto y obtener conclusiones que nos permitan llevar a cabo una valoración global del trabajo.

5.1. Análisis de resultados

En este apartado se comentarán los datos obtenidos en los diferentes experimentos realizados y, a partir de ellos, sacaremos nuestras propias conclusiones.

En la **Figura 32** se muestran los porcentajes en tanto por uno de aciertos para cada una de las pruebas realizadas.

Workspace	
Name ▲	Value
acierto_D	0.4811
acierto_I	0.3892
acierto_W	0.8258

Figura 32: Porcentajes de acierto en tanto por uno del sujeto 1.

Por tanto, para un sujeto (sujeto 1) sano, diestro y de 22 años, se han dado los siguientes resultados:

- Acierto en mano derecha 48.11%
- Acierto en mano izquierda 38.92%
- Acierto en palabra 82.58%
- Acierto global 56.53%

Para el sujeto 2 de 58 años, también sano y diestro se han obtenido los siguientes porcentajes:

- Acierto en mano derecha 44.56%
- Acierto en mano izquierda 41.44%
- Acierto en palabra 71.98%
- Acierto global 52.56%



Además de los porcentajes se muestra también lo que se denomina matriz de confusión. En el campo de la inteligencia artificial una matriz de confusión es una herramienta que permite la visualización del desempeño de un algoritmo que se emplea en aprendizaje supervisado. Cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fila representa a las instancias en la clase real. Uno de los beneficios de las matrices de confusión es que facilitan ver si el sistema está confundiendo dos clases.

En la **Figura 33** podemos ver la matriz de confusión obtenida para el sujeto 1.

```
matriz_confusion =
    0    0    0    0    0    0    0
    0    0    0    0    0    0    0
    0    0   52    0    9    0   76
    0    0    0    0    0    0    0
    0    0   49    0   18    0   97
    0    0    0    0    0    0    0
    0    0   11    0   12    0  109
```

Figura 33: Matriz de confusión.

Por otra parte, los resultados de las encuestas realizadas son los siguientes:

- Todos los usuarios opinan que resulta incómodo la colocación del casco, sin embargo, es durante la realización de los experimentos donde los sujetos se sienten más cómodos.
- La mayoría de usuarios no tiene conocimientos previos del mundo de las EEG's.
- A todos los sujetos les parece más complicada la prueba de pensar en el movimiento de mano derecha o izquierda, sintiendo más confianza en el pensamiento de palabras.
- Los sujetos acaban muy fatigados, y generalmente con dolor de cabeza o pesadez.
- Los tiempos de descanso parecen adecuados, aunque en la última sesión debería de ser mayor para evitar dicha fatiga.
- La conductividad se ve muy reducida con gente con abundante pelo
- Se considera que la realización del experimento ha sido satisfactoria en líneas generales.



Después de obtener los datos y analizando todos los resultados, podemos sacar las siguientes conclusiones:

- El modelo se encuentra dentro de los porcentajes globales de acierto esperados, acertando como mínimo la mitad de las veces.
- Resulta más sencillo identificar el pensamiento de una palabra que empiece por una letra determinada que pensar en el movimiento de cualquiera de las manos. Aquí es donde se ha conseguido el mejor resultado, obteniendo una tasa de acierto del 82.58%.
- Al tratarse de un sujeto diestro, se puede observar como la tasa de acierto de su mano hábil es casi un 10% mayor que la otra mano.
- Se ha supuesto que el índice de acierto, se ha visto reducido debido al gran número de interferencias, sobre todo musculares.

Como podemos ver en este apartado, todos los datos cumplen con lo previsto, por lo que se puede afirmar sin lugar a dudas, que el casco de señales ActiCap y el amplificador ActiCHamp se encuentran completamente operativos para su uso, habiendo comprobado su funcionamiento mediante investigaciones del problema *Motor Imagery*.

Para concluir el apartado de conclusiones, se propondrán nuevas líneas por las que se puede seguir esta investigación. En primer lugar, se deberían de realizar los experimentos a un número mayor de sujetos. También se podrían probar diferentes filtros para la señal, por ejemplo, utilizar el filtro CAR (Common Average Reference), el ICA (Análisis de componentes independientes) o el CSP (Patrones espaciales comunes). También se puede realizar un estudio de otras fuentes de interferencias y comprobar en qué medida afectan al índice de error.

5.2. Conclusión.

Cuando se ha llegado a este punto del proyecto es momento de realizar un balance de los objetivos marcados al inicio de este proyecto y, por supuesto, un análisis de los resultados obtenidos.

- Se ha realizado un estudio exhaustivo del hardware y de los sistemas basados en BCI, comprendiendo así el funcionamiento de nuestro dispositivo.



- También se ha realizado una puesta en marcha ActiCHamp y de todos sus componentes, recurriendo a documentación proporcionado por Brain Products. Además, se han instalado todos los programas, librerías y drivers necesarios para el correcto funcionamiento del sistema.
- Se ha creado una base de datos en la cual se encuentran todos los archivos utilizados en el procesamiento de datos, gracias a esta clasificación de los ficheros, nos resulta más fácil realizar el procesamiento de datos. Además, una gran ventaja de ir salvando nuestros ficheros al final de cada subproceso, es que tenemos más facilidades para continuar con el computo de datos en caso de error. También podemos ir supervisando los archivos en busca de incoherencias.
- Se han aplicado correctamente los algoritmos neurodifusos de clasificación. Mediante los programas de Matlab pertinentes, se ha conseguido filtrar una señal y clasificarla con el fin de crear un modelo de predicción. Dicho modelo de predicción es capaz de determinar la intención motora.
- Se ha realizado un análisis y discusión de los resultados obtenidos, donde se han tenido en cuenta todos los datos recopilados, tanto por el modelo como por el informe realizado por los sujetos. Se ha contrastado la información y se ha comprobado que el modelo se encuentra entre los valores de acierto estimados. Dicha estimación sirve para afirmar que nuestro dispositivo está operativo y funciona de forma correcta.
- Se han propuesto nuevas líneas para el desarrollo de este proyecto, además se han mostrado posibles aplicaciones para los sistemas BCI. También se han propuesto futuras mejoras para este Trabajo de Fin de Grado.

Por todo esto se puede decir que se han logrado satisfacer todos los objetivos propuestos al inicio de este Trabajo de Fin de Estudios, por contrapartida, se deberían haber tomado más muestras para terminar de comprobar la robustez de los algoritmos neurodifusos aplicados.



CAPÍTULO 6.

BIBLIOGRAFÍA



6. Bibliografía.

Morales-Osorio, M. and Mejía-Mejía, J. (2012). Tratamiento con imaginería motora graduada en el síndrome de miembro fantasma con dolor: una revisión sistemática. *Rehabilitación*, [online] 46(4), pp.310-316. Available at: <http://www.elsevier.es/es-revista-rehabilitacion-120-articulo-tratamiento-con-imaginaria-motora-graduada-S0048712012000679>

Gutiérrez-Martínez, J., Cantillo-Negrete, J., Cariño-Escobar, R. and Elías-Viñas, D. (2013). *Los sistemas de interfaz cerebro-computadora: una herramienta para apoyar la rehabilitación de pacientes con discapacidad motora*. [ebook] pp.pp 62-69 Vol. 2, Núm. 2. Available at: <http://www.medigraphic.com/pdfs/invdiss/ir-2013/ir132c.pdf>

Brainproducts.com. (2017). *Brain Products GmbH - Solutions for neurophysiological research*. [online] Available at: <http://www.brainproducts.com/> [Accessed 27 Sep. 2017].

Modelo Cognitivo Computacional. (2010). [Blog] *Linguistiqua Blog de linguistique Linguistic blog art, psychology, literature, and linguistics*. Available at: <http://osqvar.over-blog.com/article-7049945.html> [Accessed 27 Sep. 2017].

W3c.es. (2017). *Guía Breve de Interacción Multimodal*. [online] Available at: <http://www.w3c.es/Divulgacion/GuiasBreves/Multimodalidad> [Accessed 27 Sep. 2017].

Tecnicasfisioterapia.webnode.es. (2012). *Aplicaciones del BCI en videojuegos y ocio: Tecnicasfisioterapia*. [online] Available at: <http://tecnicasfisioterapia.webnode.es/news/aplicaciones-del-bci-en-videojuegos-y-ocio/> [Accessed 27 Sep. 2017].

Fgcsic.es. (2012). *Brain-Computer Interface (BCI) aplicado al entrenamiento cognitivo y control domótico para prevenir los efectos del envejecimiento*. [online] Available at: <http://www.fgcsic.es/lychnos/es/es/articulos/Brain-Computer-Interface-aplicado-al-entrenamiento-cognitivo> [Accessed 27 Sep. 2017].

Indexing, A. (2017). *Matrices and Arrays - MATLAB & Simulink - MathWorks United Kingdom*. [online] Es.mathworks.com. Available at:



https://es.mathworks.com/help/matlab/learn_matlab/matrices-and-arrays.html [Accessed 27 Sep. 2017].

Es.mathworks.com. (2017). *Periodogram power spectral density estimate - MATLAB periodogram - MathWorks United Kingdom*. [online] Available at: https://es.mathworks.com/help/signal/ref/periodogram.html?s_tid=doc_t [Accessed 27 Sep. 2017].

Es.mathworks.com. (2017). *Spectral Analysis - MATLAB & Simulink - MathWorks United Kingdom*. [online] Available at: <https://es.mathworks.com/help/signal/spectral-analysis.html> [Accessed 27 Sep. 2017].

Es.mathworks.com. (2017). *Welch Spectrum Estimates - MATLAB & Simulink - MathWorks United Kingdom*. [online] Available at: <https://es.mathworks.com/help/signal/ug/welch-spectrum-estimates.html> [Accessed 27 Sep. 2017].

LWP Comunidad de programadores. (2017). *Dev - C++ - Funciones sleep() y delay()*. [online] Available at: <http://www.lawebdelprogramador.com/foros/Dev-C/125179-Funciones-sleep-y-delay.html> [Accessed 27 Sep. 2017].

Emotiv. (2017). *User | Developer | Researcher Support - EMOTIV*. [online] Available at: <https://www.emotiv.com/support/> [Accessed 27 Sep. 2017].

Es.mathworks.com. (2017). *Multi-Threading - MATLAB Answers - MATLAB Central*. [online] Available at: <https://es.mathworks.com/matlabcentral/answers/38117-multi-threading> [Accessed 27 Sep. 2017].

YouTube. (2017). *Brain Products' actiCAP Xpress - Recording EEG the fast & easy way*. [VÍdeo online] Available at: <https://www.youtube.com/watch?v=jivuR0yxavs> [Accessed 27 Sep. 2017].

YouTube. (2017). *Demo Dry Electrode at "Lange Nacht der Wissenschaft"*. [VÍdeo online] Available at: https://www.youtube.com/watch?v=y_uRYJzDv_E [Accessed 27 Sep. 2017].



CAPÍTULO 7.

ANEXOS



7. Anexos.

7.1. Anexo 1. Programa para generación de estados.

```
#include <iostream>
#include <string>
#include <string>
#include <vector>
#include <sstream>
#include <dos.h>
#include <array>
#include <chrono>
#include <dos.h>
#include <windows.h>
#include <random>

using namespace std;

void experimento1 (vector<int> exp1){
    for (int i=0;i<exp1.size();i++){
        if (exp1[i]==3){
            cout<<"<!--"<<endl;
            Sleep(15000);
        }
        else if (exp1[i]==5){
            cout<<"-->"<<endl;
            Sleep(15000);
        }
        else {
            char strrnd;
```



```
        srand(time(NULL));
    strrnd=97+ rand() % (122-97);
    cout<<strrnd<<endl;
        Sleep(15000);
    }
}

}

void experimento2 (vector<int> exp2){
    for (int j=0;j<exp2.size();j++){
        if (exp2[j]==3){
            cout<<"<!--"<<endl;
            Sleep(15000);

        }
        else if (exp2[j]==5){
            cout<<"-->"<<endl;
            Sleep(15000);
        }
        else {
            char strrnd;
            srand(time(NULL));
            strrnd=97+ rand() % (122-97);
            cout<<strrnd<<endl;
            Sleep(15000);
        }
    }
}

}

void experimento3 (vector<int> exp3){
```



```
for (int k=0;k<exp3.size();k++){
    if (exp3[k]==3){
        cout<<"<!--"<<endl;
        Sleep(15000);
    }
    else if (exp3[k]==5){
        cout<<"-->"<<endl;
        Sleep(15000);
    }
    else {
        char strrnd;
        srand(time(NULL));
        strrnd=97+ rand() % (122-97);
        cout<<strrnd<<endl;
        Sleep(15000);
    }
}
```

```
void experimento4 (vector<int> exp4){
    for (int n=0;n<exp4.size();n++){
        if (exp4[n]==3){
            cout<<"<!--"<<endl;
            Sleep(15000);
        }
        else if (exp4[n]==5){
            cout<<"-->"<<endl;
            Sleep(15000);
        }
    }
}
```



```
        }
        else {
            char strrnd;
            srand(time(NULL));
            strrnd=97+ rand() % (122-97);
            cout<<strrnd<<endl;
            Sleep(15000);
        }
    }

}

int main() {
    vector<int> exp1;
    exp1={3,5,7,5,3,5,7,7,3,5,3,3,7,5,7,3};
    vector<int> exp2;
    exp2={3,7,3,5,7,5,5,3,7,5,5,7,3,7,3,5};
    vector<int> exp3;
    exp3={7,3,5,3,7,3,7,7,5,7,3,5,5,3,7,5};
    vector<int> exp4;
    exp4={5,3,3,7,7,5,3,7,5,7,5,7,3,5,3,3};

    cout<<"1) Experimento 1"<<endl;
    cout<<"2) Experimento 2"<<endl;
    cout<<"3) Experimento 3"<<endl;
    cout<<"4) Experimento 4"<<endl;

    int aux;
    cin>> aux;

    switch(aux)
    {
```



```
    case 1: cout << "Usted ha seleccionado el experimento 1"<<endl;
    experimento1(exp1);
    break;
    case 2: cout << "Usted ha seleccionado el experimento 2"<<endl;
    experimento2(exp2);
    break;
    case 3: cout << "Usted ha seleccionado el experimento 3"<<endl;
    experimento3(exp3);
    break;
    case 4: cout << "Usted ha seleccionado el experimento 4"<<endl;
    experimento4(exp4);
    break;
    default: cout << "Usted ha ingresado una opción incorrecta";
}

return 0;
}
```



7.2. Anexo 2. Código para generación de vectores de etiquetas.

```
[a,b]=size(fichero1);  
num=a/16;  
for i=1:a  
    if (i<num)  
        matrix(i)=3;  
  
    elseif (i<2*num && i>num)  
        matrix(i)=5;  
  
    elseif (i<3*num && i>2*num)  
        matrix(i)=7;  
  
    elseif (i<4*num && i>3*num)  
        matrix(i)=5;  
  
    elseif (i<5*num && i>4*num)  
        matrix(i)=3;  
  
    elseif (i<6*num && i>5*num)  
        matrix(i)=5;  
  
    elseif (i<7*num && i>6*num)  
        matrix(i)=7;  
  
    elseif (i<8*num && i>7*num)  
        matrix(i)=7;  
  
    elseif (i<9*num && i>8*num)
```



```
matrix(i)=3;

elseif (i<10*num && i>9*num)
matrix(i)=5;

elseif (i<11*num && i>10*num)
matrix(i)=3;

elseif (i<12*num && i>11*num)
matrix(i)=3;

elseif (i<13*num && i>12*num)
matrix(i)=7;

elseif (i<14*num && i>13*num)
matrix(i)=5;

elseif (i<15*num && i>14*num)
matrix(i)=7;

else matrix(i)=3;
end
end
matrix=matrix';
```




7.3. Anexo 3. Guía rápida para adquisición de señales.

En esta guía se explica paso a paso lo que se ha de hacer para utilizar el casco de la formas más sencilla y rápida posible para realizar una grabación de una sesión.

Los pasos a seguir son los siguientes:

- En primer lugar, debemos cargar el dispositivo ActiPower, para realizar una carga completa se debe dejar conectado unas ocho horas.
- Una vez que el ActiPower este cargado debemos de realizar el montaje de todos los elementos:
 - Conectar cable de alimentación (desde el ActiCHamp hasta el ActiPower).
 - Conectar el cable USB (Desde el ActiCHamp hasta el PC).
 - Conectar el casco de señales al ActiCHamp.
 - Conectar el electrodo de tierra a la parte frontal del ActiCHamp (donde se encuentra escrito GND).
 - Poner el cabezal del electrodo de tierra en la parte delantera del casco.
- Una vez realizado el montaje se debe comprobar que los electrodos se encuentran bien situados (con la disposición deseada).
- Si los electrodos se encuentran bien situados, debemos asegurarnos de que se encuentran limpios.
- Después debemos de iniciar el programa PyCorder.
- Una vez iniciado el programa se le debe colocar el casco al usuario.
- Con el casco bien puesto debemos seleccionar en la ventana principal del programa la opción “Impedance Mode” donde veremos que todos los electrodos se encuentran en color rojo.
- Una vez que estemos en esa ventana es el momento de poner el gel electrolítico mediante la aguja de aplicación. A medida que se vaya echando el gel, se podrá ver como el color del electrodo pasa a color amarillo o verde.
- En el caso de que no haya buena conexión entre el electrodo y el cuero cabelludo, se moverá dicho electrodo hasta que se ponga de color verde.
- Se repetirá este paso con todos los electrodos hasta que la impedancia sea aceptable (color amarillo o verde).
- Una vez que todos los electrodos estén bien colocados debemos habilitarlos en la pestaña de configuración del programa.



- Después se debe elegir la frecuencia de muestreo en una ventana desplegable (dentro de configuración).
- Se debe elegir también el directorio donde se guardarán las grabaciones.
- Con todos los ajustes realizados debemos pulsar el botón de “Shielding Mode” y veremos nuestras señales en la pantalla.
- A continuación, podremos cambiar la escala de visualización del programa, tanto el potencial eléctrico (eje y) como el tiempo (eje x).
- Una vez que queramos hacer la grabación se seleccionara la opción “Start recording”.
- Para pausar o detener la grabación, se pulsarán los botones de “Pause” o “Stop”.
- Cuando se pulsa el botón “Stop” se generan los archivos de las grabaciones.



7.4. Anexo 4. Guía rápida para procesamiento de datos.

Una vez que se han realizado las grabaciones pertinentes se han de seguir los siguientes pasos para realizar una investigación sobre el problema *Motor Imagery*.

- Abrir el script `bva_loadeeg.m`
- Introducir el comando: `eeg=bva_loadeeg('Nombre del experimento');`
- Guardar ese archivo usando: `save nombre del fichero eeg;`
- Repetir el proceso con las cuatro sesiones grabadas
- A continuación debemos de abrir el fichero `LaplacianFilterZoneSetV.m` y una vez cargados los ficheros `.mat` de las sesiones introducir los siguientes comandos:
 - `[a,b]=size(eeg)`
 - `For i=1:b`
 - `X1'(:,i)=LaplacianFilterZoneSetV(eeg(:,i),x,y);`
 - `End`
- Una vez que tenemos todos los archivos en Matlab del filtro Laplaciano creamos los vectores de etiquetas, corriendo el script `vectordeetiquetas1.m`
- Se repite este paso para las otras tres sesiones, utilizando los scripts (`vectordeetiquetas2.m`, `vectordeetiquetas3.m` y `vectordeetiquetas4.m`).
- Cuando tengamos los ficheros del filtro y el vector de etiquetas (4 archivos y 4 vectores) debemos de abrir el script `windowsprocessingMatrix` y poner el comando:
 - `[PSDDATA,PSDWINDOW]=windowsprocessingMatrix(fichero1,500,(8:2:30),500,31,vector1);`
- Repetimos este proceso para las otras tres sesiones (solo tenemos que cambiar `fichero1` y `vector1`).



7.5. Anexo 5. Guía rápida para clasificación.

- Se ejecuta Aprende.m con los resultados obtenidos de la sesión 1 y Buscar_parámetros.m con las sesiones 2 y 3 para obtener los parámetros $At(1,2)$, $\delta(1,2)$, $At(1,3)$ y $\delta(1,3)$.
- Se ejecuta Aprende.m con los resultados obtenidos de la sesión 2 y Buscar_parámetros.m con las sesiones 1 y 3 para obtener los parámetros $At(2,1)$, $\delta(2,1)$, $At(2,3)$ y $\delta(2,3)$.
- Se ejecuta Aprende.m con los resultados obtenidos de la sesión 3 y Buscar_parámetros.m con las sesiones 1 y 2 para obtener los parámetros $At(3,1)$, $\delta(3,1)$, $At(3,1)$ y $\delta(3,2)$.
- Una vez que se han obtenido todos parámetros hemos de utilizar el programa SdFasArt_coordenadas_proceso_total.m cambiando los parámetros delta y At por los que hemos obtenido.
- Cuando se termina de correr este último programa se muestra por pantalla los porcentajes de acierto para todas las pruebas realizadas.



8. Agradecimientos.

Me gustaría mostrar mi agradecimiento a todas aquellas personas que de una forma u otra han hecho posible la realización de este trabajo gracias a su ayuda y cariño.

Quiero agradecer a mis padres su preocupación, comprensión y cariño que han tenido hacia mi persona. Ellos me han enseñado que tengo que luchar por mi futuro, que tengo que ser perseverante y constante, y sobre todo, me han enseñado a no rendirme.

También quiero agradecer a mi hermana su atención y su disposición en todo momento.

Agradecer también a mi sobrina María, que, con tan solo 4 añitos, has sido capaz de inundarme de alegría en todo momento y de desearme suerte “para ir al cole” todos los días.

Mil gracias también a Marta y Manolo, por toda la paciencia que han tenido conmigo, por enseñarme a mantener la calma y por demostrarme que los amigos de la universidad de verdad pueden ser para siempre.

Gracias también a Virginia, por ser un gran apoyo durante estos casi cuatro años, ayudándome a levantarme tras cada caída y recordándome que soy capaz de hacer todo lo que me proponga.

Gracias a mi tutor Julio, por darme la oportunidad de inmiscuirme en este proyecto.

Y, por último, a mi codirector José Manuel, por toda la paciencia que ha tenido conmigo, por todos los conocimientos que me ha aportado y por saber guiarme a lo largo de este proyecto.

