

UNIVERSIDAD DEL VALLE DE GUATEMALA  
Facultad de Ingeniería



**Análisis y Reconocimiento de Patrones de Señales Biomédicas  
Relacionadas a las Etapas de Sueño**

Trabajo de graduación presentado por Oscar Rodrigo López Portillo  
para optar al grado académico de Licenciado en Ingeniería Mecatrónica

Guatemala

2021







UNIVERSIDAD DEL VALLE DE GUATEMALA  
Facultad de Ingeniería



**Análisis y Reconocimiento de Patrones de Señales Biomédicas  
Relacionadas a las Etapas de Sueño**

Trabajo de graduación presentado por Oscar Rodrigo López Portillo  
para optar al grado académico de Licenciado en Ingeniería Mecatrónica

Guatemala,

2021



Vo.Bo.:



(f) \_\_\_\_\_  
Dr. Luis Alberto Rivera Estrada

Tribunal Examinador:



(f) \_\_\_\_\_  
Dr. Luis Alberto Rivera Estrada



(f) \_\_\_\_\_  
Ing. Andrés Monterroso



(f) \_\_\_\_\_  
MSc. Renato Conedera

Fecha de aprobación: Guatemala, 07 de Enero de 2021.



<b>Lista de figuras</b>	VIII
<b>Lista de cuadros</b>	IX
<b>Resumen</b>	XI
<b>Abstract</b>	XIII
<b>1. Introducción</b>	<b>1</b>
<b>2. Justificación</b>	<b>3</b>
<b>3. Objetivos</b>	<b>5</b>
3.1. Objetivo general . . . . .	5
3.2. Objetivos específicos . . . . .	5
<b>4. Marco teórico</b>	<b>7</b>
4.1. Ondas cerebrales . . . . .	7
4.1.1. Ondas Delta 0.5 Hz a 4 Hz . . . . .	7
4.1.2. Ondas Theta 4 Hz a 7 Hz . . . . .	8
4.1.3. Ondas Alfa 7 Hz a 12 Hz . . . . .	8
4.1.4. Ondas Beta 13 Hz a 39 Hz . . . . .	8
4.1.5. Ondas Gamma 39Hz en adelante . . . . .	8
4.2. Señales binaurales . . . . .	9
4.3. El ciclo del sueño . . . . .	11
4.4. Electro - Cap . . . . .	12
4.5. Sistema Internacional 10-20 . . . . .	14
4.6. OpenBCI . . . . .	15
4.7. Cyton Board y Dongle . . . . .	16
4.8. Machine Learning . . . . .	18
4.8.1. Aprendizaje supervisado . . . . .	18
4.8.2. Aprendizaje no supervisado . . . . .	20
4.8.3. Red Neuronal Artificial . . . . .	20

4.8.4. Máquinas de Vector Soporte (Support Vector Machines, SVMs)	22
4.8.5. K Vecinos más cercanos	23
4.8.6. Características para análisis de señales	24
<b>5. Obtención, análisis y clasificación de señales de Physionet</b>	<b>27</b>
5.1. Obtención de los datos	27
5.2. Análisis de las señales	28
5.3. Clasificación de las señales	29
<b>6. Obtención de señales con el Electro-Cap</b>	<b>35</b>
6.1. Familiarización con la Cyton Board y el Electro - cap	35
6.2. Conexión Matlab – Cyton Board	36
6.3. Validación del algoritmo de obtención de las señales	38
<b>7. Análisis y clasificación de las señales del Electro-Cap</b>	<b>41</b>
7.1. Validación de clasificación y extracción de características	41
7.2. Obtención de las señales de entrenamiento	44
7.3. Análisis y clasificación de señales	47
<b>8. Almacenamiento de las señales</b>	<b>53</b>
8.1. Opciones de bases de datos	53
8.1.1. FireBase	53
8.1.2. MySQL	54
8.2. Desarrollo de la base de datos.	55
<b>9. Conclusiones</b>	<b>57</b>
<b>10.Recomendaciones</b>	<b>59</b>
<b>11.Bibliografía</b>	<b>61</b>

1. Ondas cerebrales [2]. . . . .	9
2. Categorías principales ondas cerebrales utilizadas en la música binaural beats y sus beneficios [2]. . . . .	10
3. Hipnograma estándar del sueño [6]. . . . .	11
4. Electro Cap y sus componentes [9]. . . . .	13
5. Ubicación de electrodos según el Sistema Internacional 10-20 [10]. . . . .	15
6. Interfaz gráfica de OpenBCI [11]. . . . .	16
7. Cyton Board de OpenBCI [12]. . . . .	17
8. Dongle de OpenBCI [12]. . . . .	18
9. Principales modelos de RNAs con aprendizaje supervisado [14]. . . . .	19
10. Modelo de la neurona artificial [15]. . . . .	20
11. Modelo de la neurona artificial [14]. . . . .	21
12. Comportamiento de la curtosis [21]. . . . .	25
13. Vista de la página de donde se obtuvieron los datos. . . . .	27
14. Archivos de datos disponibles. . . . .	28
15. Matrices de confusión para la red neuronal con un canal de datos. . . . .	30
16. Matrices de confusión para la máquina de vectores de soporte con kernel Lineal con un canal de datos. . . . .	31
17. Matrices de confusión para k- nearest neighbor con un canal de datos. . . . .	32
18. Matrices de confusión para la red neuronal. . . . .	33
19. Matrices de confusión para la máquina de vectores de soporte con kernel Lineal. . . . .	33
20. Matrices de confusión para k- nearest neighbor. . . . .	34
21. Captura 5 repeticiones cerrando el ojo izquierdo. . . . .	35
22. Captura 4 repeticiones remachando los dientes. . . . .	36
23. Captura de gestos utilizando el algoritmo desarrollado. . . . .	37
24. Captura de gestos de los 8 canales utilizando OpenBCI GUI. . . . .	37
25. Ondas Gamma obtenidas con el programa en Matlab. . . . .	38
26. Ondas Beta obtenidas con el programa en Matlab. . . . .	38
27. Ondas Delta obtenidas con el programa en Matlab. . . . .	39
28. Captura de secuencia de gestos utilizando el algoritmo desarrollado. . . . .	42

29. Matriz de confusión para la secuencia de gestos utilizando la máquina de vectores de soporte. . . . .	42
30. Matriz de confusión para la secuencia de gestos utilizando k- nearest neighbor.	43
31. Matriz de confusión para la secuencia de gestos utilizando la red neuronal. . .	44
32. Estado despierto. . . . .	45
33. Etapa 1 del sueño. . . . .	45
34. Etapa 2 del sueño. . . . .	46
35. Etapa 3 del sueño. . . . .	46
36. Etapa de sueño REM. . . . .	46
37. Matriz de confusión para SVM con duración de 2 horas. . . . .	47
38. Matriz de confusión para k- nearest neighbor con duración de 2 horas. . . . .	48
39. Matriz de confusión para Red neuronal con duración de 2 horas. . . . .	49
40. Matriz de confusión para SVM con duración de 10 horas. . . . .	50
41. Matriz de confusión para k- nearest neighbor con duración de 10 horas. . . . .	50
42. Matriz de confusión para Red neuronal con duración de 10 horas. . . . .	51
43. Esquema del proceso para la clasificación de las señales. . . . .	52
44. Página de inicio de FireBase. . . . .	54
45. Vista del almacenamiento de gestos pt1. . . . .	55
46. Vista del almacenamiento de gestos pt2. . . . .	55

---

Lista de cuadros

---

1. Pruebas preliminares . . . . .	14
2. Etiquetas respectivas para cada gesto analizado. . . . .	41



Este proyecto consistió en la realización de la segunda fase de un sistema de clasificación automática de las etapas del sueño, capaz de generar pulsos binaurales para combatir trastornos del sueño en los atletas. En esta segunda fase se realizó un algoritmo capaz de obtener los datos provenientes de la conexión entre el electro - cap y la Cyton Board, para luego procesar los datos y poderlos utilizar en distintos clasificadores, los cuales están basados en los principios de *Machine Learning*. Para poder aplicar esto primero se analizaron diferentes características que sirvieran para poder extraer la mayor información posible a las señales electroencefalográficas (EEG) y así los clasificadores pudieran identificar los distintos patrones correspondientes a cada una de las etapas del sueño. Para el entrenamiento de los clasificadores se utilizaron datos de distintas polisomnografías utilizando las características previamente analizadas las cuales fueron: Zero Crossing (ZC), Mean Absolute Value (MAV), Max Min Distance (MMD), Slow wave activity (SWA) y Kurtosis. Todos los resultados obtenidos y sus señales finalmente fueron almacenados en una base de datos que se realizó en conjunto con los clasificadores y así poder tener un conjunto de datos biomédicos según estándares internacionales.



This project consisted in the realization of the second phase of an automatic classification system of sleep stages capable of generating binaural pulses to combat sleep disorders in athletes. In this second phase, an algorithm capable of obtaining the data from athletes was carried out of the connection between the electro-cap and the Cyton Board to later process the data and be able to use it in different classifiers which are based on the principles of *Machine Learning*. In order to apply this, different characteristics were first analyzed and were used to extract as much information as possible from the electroencephalographic signals (EEG) and thus the classifiers could identify the different patterns corresponding to each of the sleep stages. For the training of the classifiers, data from different polysomnographies were used using the previously analyzed characteristics which were: Zero Crossing (ZC), Mean Absolute Value (MAV), Max Min Distance (MMD), Slow wave activity (SWA) and Kurtosis. All the results obtained and their signals were finally stored in a database that was carried out in conjunction with the classifiers and thus be able to have a set of biomedical data according to international standards.



# CAPÍTULO 1

---

## Introducción

---

En los últimos años el desarrollo en el campo científico y tecnológico ha sido bastante grande y se han desarrollado distintas tecnologías que mejoran la calidad de vida de las personas. A pesar de todos estos avances, aún hay varios campos que no se han explorado con más profundidad. Este es el caso del análisis del sueño en diferentes aplicaciones, tanto para la vida cotidiana como para la profesional y la deportiva. Este es un campo que tiene un gran peso debido a que dormir es una de las actividades esenciales para el desarrollo de cualquier ser humano. Al dormir, una persona se recupera tanto física como psicológicamente.

Teniendo noción de lo anterior se busca enfocar esta área al ámbito deportivo en donde el proceso de recuperación y descanso del atleta es un factor importante que afecta en su rendimiento. Debido a que el agotamiento provocado por la exigencia y presión del día a día en estos deportistas ocasiona una mayor susceptibilidad a padecer trastornos del sueño con el paso de los años.

Dado que un atleta profesional debe de optimizar al máximo sus horas de sueño para tener el mejor rendimiento y recuperación posible, es importante que este explore distintas alternativas para poder tener un descanso adecuado. Con este trabajo se busca crear el diseño de un sistema capaz de monitorear la actividad cerebral de un atleta mientras duerme. Además de poder identificar en qué etapas de sueño se encuentra y así poder tener un control del proceso de recuperación y descanso. Con el fin de poder aplicar diferentes medidas para mejorar la recuperación del atleta.

Este trabajo tiene como objetivo mostrar resultados que sirvan como base para futuras investigaciones dentro del área biomédica e ingeniería aplicada al deporte. También se espera que sirva para el desarrollo de distintas fases del proyecto que ayuden a mejorar este proceso de identificación y aplicación de las etapas del sueño. Además, se espera que este trabajo ayude a contribuir a la comunidad científica en temas como el comportamiento de las ondas cerebrales y sus diferentes técnicas de análisis, el uso de *machine learning* para el análisis de estas señales y poder reconocer las diferentes etapas del sueño en tiempo real. Cabe mencionar que este proyecto se realizó como continuación de una fase anterior.

Con este trabajo también se busca poder implementar diferentes técnicas de reconocimiento de patrones para poder analizar señales biomédicas y así poder monitorear las distintas etapas de sueño en tiempo real. Para esto se plantearon distintas metas para poder llevar a cabo el proyecto. Dichas metas consistieron en obtener una colección de señales biomédicas que cumplan con los distintos estándares internacionales y poder utilizar dichas señales en futuros análisis, ya sea para alguna continuación de este proyecto o de algún otro que surja en el futuro. También se busca aplicar distintas técnicas de selección de características para el reconocimientos de patrones y poder analizar cuáles tienen un mejor resultado en la clasificación de las señales biomédicas. Por último se busca poder realizar e implementar un herramienta de hardware y software que permita analizar y reconocer las distintas etapas del sueño. Teniendo en consideración lo anterior, se logro organizar el documento de tal forma que fuera entendible el proceso que se realizo desde la de obtención de las señales, su clasificación y almacenamiento.

Se espera que con esta investigación, toda persona interesada en el tema pueda comprender los diferentes procesos que se realizaron para que puedan replicarlos, también se espera que con el tiempo mas estudiantes de la Universidad del Valle de Guatemala se interesen en el tema y decidan continuar con este proyecto y así lograr mejores resultados y poder expandir el alcance de este estudio.

A lo largo de su vida una persona duerme aproximadamente una tercera parte de ella. Esto provoca que el estudio del sueño tenga una gran importancia, pero lamentablemente no se han realizado tantas investigaciones en esta área. Dormir es un proceso de gran importancia debido a que es la manera en la que el cuerpo humano se desarrolla y restaura. Esto tiene un mayor impacto en los atletas ya que ellos requieren de un buen descanso para poder tener un buen rendimiento en sus respectivas disciplinas. Según diferentes estudios se ha demostrado que gran parte de los atletas poseen desórdenes de conciliación y mantenimiento del sueño. En otras palabras, estos atletas poseen un sueño menos eficiente al del resto de las personas lo cual puede tener repercusiones en su salud.

Los estudios del sueño pueden ayudar al médico a diagnosticar trastornos respiratorios relacionados con el sueño, como la apnea del sueño, trastornos convulsivos relacionados con el sueño, trastornos del movimiento relacionados con el sueño y trastornos del sueño que causan cansancio extremo durante el día como la narcolepsia. Los médicos, también, pueden usar los estudios del sueño para ayudar a diagnosticar o descartar el síndrome de las piernas inquietas.

Según un estudio realizado por la Universidad de Finlandia Oriental y la Clínica de Sueño Oivauni, un 22 % de los atletas estudiados reportaron tener problemas para dormir mientras se encontraban fuera de temporada. Además, estando fuera de temporada un 4 % de ellos también comentaron tomar algún medicamento para poder dormir mejor. Estos números aumentan significativamente a un 46 % y 17 % respectivamente cuando los atletas están en plena temporada de competición [1].

Adicionalmente, durante los últimos años se ha ido desarrollando el área de inteligencia artificial lo cual ha permitido el desarrollo de algoritmos clasificadores. En base a una colección de datos, los clasificadores pueden entrenarse para poder reconocer patrones y con esto poder tomar decisiones o realizar acciones automáticamente. Estos clasificadores se han utilizado para la clasificación de señales bioeléctricas. Esto es de gran importancia ya que utilizando estas herramientas se pueden obtener y clasificar las señales EEG producidas al dormir y con esto detectar en qué etapa de sueño se encuentra la persona.

Según lo mencionado anteriormente lo que se busca con este proyecto es aprovechar las diferentes herramientas que se han desarrollado para poder generar una nueva que sea capaz de obtener y procesar las señales producidas al dormir. Todo esto con el fin de poder identificar las diferentes etapas del sueño a medida que estas se vayan dando.

Para ello se busca realizar pruebas con distintos tipos de clasificadores y características en diferentes dominios (tiempo, frecuencia, estadístico) para poder observar los diferentes resultados que estos presenten y poder realizar la comparación entre ellos para ver cuales presentan mejores resultados.

### 3.1. Objetivo general

Implementar técnicas y algoritmos de reconocimiento de patrones a señales biomédicas para su aplicación en un sistema inteligente de monitoreo de etapas de sueño.

### 3.2. Objetivos específicos

- Obtener una colección de señales biomédicas relacionadas a etapas de sueño, según estándares internacionales.
- Analizar técnicas de selección de características y reconocimiento de patrones de señales biomédicas, y seleccionar aquellas que permitan la clasificación de las etapas de sueño.
- Aplicar las técnicas seleccionadas a la colección de señales obtenidas, y validar resultados mediante asesoría médica y comparación con estudios similares.
- Diseñar e implementar una herramienta de hardware y software que permita monitorear las etapas de sueño en tiempo real.



## 4.1. Ondas cerebrales

La raíz de todos nuestros pensamientos, emociones y comportamientos es la comunicación entre las neuronas dentro de nuestros cerebros. Las ondas cerebrales son producidas por pulsos eléctricos sincronizados de masas de neuronas que se comunican entre sí. Las ondas cerebrales se detectan mediante sensores colocados en el cuero cabelludo. Se dividen en anchos de banda para describir sus funciones, pero se considera mejor como un espectro continuo de conciencia; desde lento, ruidoso y funcional, hasta rápido, sutil y complejo. Nuestras ondas cerebrales cambian de acuerdo con lo que estamos haciendo y sintiendo. Cuando las ondas cerebrales más lentas son dominantes, podemos sentirnos cansados, lentos, lentos o soñadores. Las frecuencias más altas son dominantes cuando nos sentimos conectados, o hiper-alerta [2].

La velocidad de transmisión de las ondas cerebrales se mide en hercios (ciclos por segundo) y se dividen en bandas que delimitan las ondas lentas, moderadas y rápidas. Esta subdivisión se puede apreciar de mejor manera en la Figura 1.

### 4.1.1. Ondas Delta 0.5 Hz a 4 Hz

Están asociadas a un estado de sueño profundo. En estas ondas la mente opera de forma totalmente integrada. Sus estados psíquicos correspondientes son el dormir sin sueños, el trance y la hipnosis profunda. Si se consigue entrar y permanecer consciente en el nivel delta, se alcanza el estado de superconsciencia que es el último peldaño de la meditación, donde desaparecen las barreras entre consciente, subconsciente e inconsciente [3].

#### **4.1.2. Ondas Theta 4 Hz a 7 Hz**

Estas ondas son producidas durante estados de meditación profunda y durante periodos de concentración intelectual. Las ondas Theta se asocian a un estado emocional cuyas características principales son la memoria plástica, la estimulación de la imaginación y creatividad y un incremento en la capacidad de aprendizaje [2].

#### **4.1.3. Ondas Alfa 7 Hz a 12 Hz**

Se las registran en los momentos antes de dormirse, se producen en un estado de no actividad y relajación mental. Son más lentas y de mayor amplitud [3].

#### **4.1.4. Ondas Beta 13 Hz a 39 Hz**

Este tipo de ondas están asociados a estados de actividad mental intensa, resolviendo problemas, estudiando, concentración o ansiedad. Es decir, se producen cuando el cerebro está despierto, cuando hay mayor extroversión, cuando hay actividades mentales y hay percepción a través de los cinco sentidos, así como también cuando se realiza trabajos del tipo físico o corporal. Es la más rápida de las cuatro [3].

#### **4.1.5. Ondas Gamma 39Hz en adelante**

Son las ondas más rápidas y de menor amplitud. Pueden aparecer en cualquier parte del cerebro y se asocian a estados donde una persona esta resolviendo problemas lógicos o matemáticos, favoreciendo el procesamiento de la información [2].

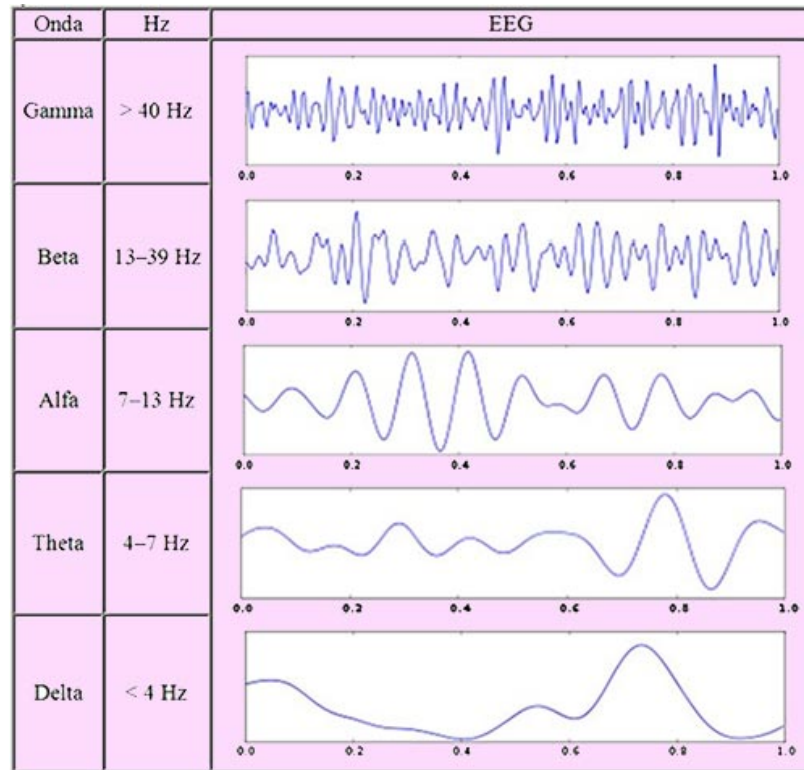


Figura 1: Ondas cerebrales [2].

## 4.2. Señales binaurales

La palabra binaural significa "tener o relacionarse con dos oídos". Esto representa el proceso, que funciona enviando simultáneamente una frecuencia de sonido diferente a cada oído a través de los auriculares. El arrastre ocurre dentro del cerebro y es causado por una respuesta fisiológica. Al escuchar dos tonos de diferentes frecuencias, enviados simultáneamente a los oídos izquierdo y derecho, el cerebro percibe un nuevo tono, o tercer tono como algunos lo llaman, basado en la diferencia matemática entre las dos frecuencias. Luego, el cerebro sigue a la nueva frecuencia y produce ondas cerebrales a la misma velocidad de Hertz (Hz), quedando así arrastrado a esa frecuencia [4]. En la Audición Binaural los oídos forman canales receptores que son independientes entre sí, es decir, no hay interferencias entre ellos, ni combinaciones de las frecuencias recibidas por cada uno; los sonidos se reciben independientemente por cada oído y crean efectos diferentes en distintas partes del cerebro. Dentro de la audición binaural existen dos factores que determinan la dirección de llegada de un sonido: su intensidad relativa en los dos oídos y el intervalo de tiempo de llegada a los oídos. Para localizar una fuente de sonido, el sistema auditivo utiliza la diferencia de tiempo existente entre el sonido que llega al oído derecho y al izquierdo, conocido como ITD (Interaural Time Difference) [3].

Estos tonos son sonidos con un comportamiento sinusoidal, escuchados con un casco, tienden a sincronizar los dos hemisferios del cerebro. La diferencia de frecuencias induce dos efectos:

- Se comienza por decir que esta diferencia de frecuencias esta tratada en los núcleos olivarios superiores de cada hemisferio ellos producen asi ondas cerebrales de la misma frecuencia que la diferencia; estas nuevas ondas se propagan progresivamente a partir de estos núcleos en toda la corteza. Así pues, por ejemplo para hacer producir ondas alfa se hará escuchar sonidos que difieren de 8 a 13 Hz, lo que corresponde a la frecuencia de las ondas alfa [5].
- Estos sonidos permiten también sincronizar los dos hemisferios cerebrales , sincronización que induciría un estado de bienestar, una mejora del humor, mejores resultados atencionales y de memoria, un tratamiento de las emociones y una disminución de la ansiedad [5].

Los latidos binaurales pueden llevar al cerebro a diferentes estados del ser, de los cuales hay cinco categorías clave. El efecto que los ritmos binaurales tienen en el oyente depende del estado de la onda cerebral de la música [4].

Por ejemplo: para inducir una relajación profunda, podemos escuchar música de estado Theta, Para que el cerebro duerma más profundamente, podemos escuchar la música del estado de Delta [4]. En la Figura 2 se puede apreciar de manera más clara las diferentes ondas cerebrales y a que actividades están relacionadas para poder estimularlas.

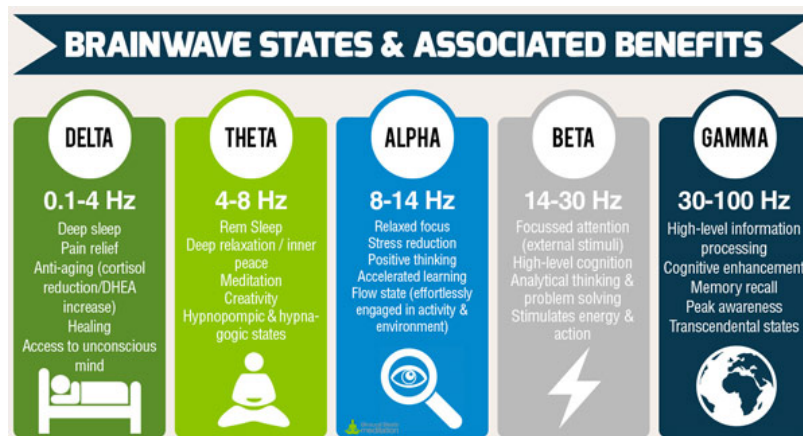


Figura 2: Categorías principales ondas cerebrales utilizadas en la música binaural beats y sus beneficios [2].

La audición binaural permite al sistema auditivo realizar un correcto análisis de la información que portan las ondas sonoras al posibilitar la comparación de la señal que llega a cada oído. La relación de ambos lados de la vía auditiva sucede casi desde el inicio de la misma, ya en el tronco cerebral, mediante decusaciones de fibras al lado contralateral de la vía auditiva. De esta manera, la estimulación sonora influye en la actividad del sistema nervioso y, posteriormente, en el comportamiento del individuo. De hecho, pequeñas diferencias interaurales de frecuencia en un sonido que estimule bilateralmente al individuo inducen una sensación auditiva rítmica con una frecuencia de batido igual a la diferencia de frecuencias interaurales de los sonidos estimulantes [5].

### 4.3. El ciclo del sueño

Cuando una persona duerme su inconsciente va progresado a través de distintas etapas del sueño que conforman lo que se conoce como el ciclo del sueño. A las primeras cuatro etapas se les denomina como etapa 1, etapa 2, etapa 3 y etapa 4 y se clasifican como sueño NREM por el acrónimo que se forma con sus siglas en inglés (Non-Rapid Eye Movement), ya que durante estas etapas apenas existe movimiento ocular. Por otro lado, la última etapa es conocida y clasificada como la etapa REM también llamada así por sus siglas en inglés (Rapid Eye Movement). En esta etapa, al contrario de lo que ocurre en las etapas NREM, los ojos se mueven constante y rápidamente. A lo largo de toda la noche una persona atraviesa todas las etapas de manera cíclica, empezando por la etapa 1 y terminando en la etapa REM. Generalmente completar el ciclo del sueño dura entre 90-110 minutos, por lo que se recorre entre 4 y 6 veces por noche dependiendo de cuanto tiempo se duerma. Durante el transcurso de la noche cada etapa tiene una duración aproximada entre 5-15 minutos dependiendo de cuanto tiempo lleve la persona dormida. Comúnmente, durante los primeros ciclos las etapas de sueño REM son cortas con largos periodos de la etapa 4. Conforme avanza la noche la situación se revierte y el sueño REM aumenta su duración mientras que la etapa 4 aparece por menos tiempo e incluso llega a desaparecer en los últimos tramos de la noche [6].

Estas etapas pueden representarse con lo que se conoce como hipnograma el cual es un registro gráfico de las distintas etapas del sueño que se genera en base a las mediciones de señales bioeléctrica en el cuerpo mientras una persona duerme. Este gráfico es el que les permite a los expertos diagnosticar entre los distintos trastornos de sueño que hay. Un ejemplo de este puede apreciarse en la Figura 3 en la cual se muestra un diagrama del ciclo de una persona normal, en el puede apreciarse la distribución de tiempo según cada etapa y más adelante se muestra más a detalle cada una de las diferentes etapas mencionadas anteriormente [6].

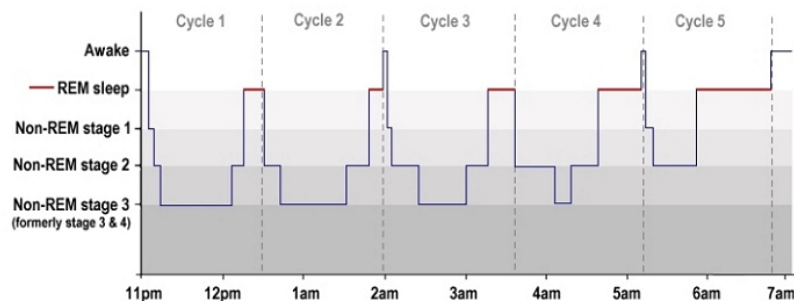


Figura 3: Hipnograma estándar del sueño [6].

- Etapa I o de adormecimiento: Inicia al acostarnos, la somnolencia o sueño ligero, los músculos se relajan y las ondas cerebrales son irregulares y rápidas. Al producirse por primera vez durante la noche, suele pasar un lapso comprendido de 30 segundos a 7 min, en ella disminuye la frecuencia cardiaca y respiratoria, Se produce una onda theta y en esta etapa es fácil despertar al sujeto [7].
- Etapa II o de sueño ligero: ocupa aproximadamente el 50% de los ciclos de sueño. Generalmente ocupa el 20% de la noche y las ondas cerebrales son de mayor amplitud

y menor frecuencia. Es posible que pasen por la mente ideas fragmentadas o imágenes, pero no tenemos presencia de nuestro entorno y aunque nuestros ojos puedan estar abiertos, no podemos ver. Dura de 10 a 15 minutos, se produce una disminución de la temperatura y la tasa metabólica [7].

- Etapa III o de transición: es una etapa corta, de dos o tres minutos, que acerca al sueño profundo. Si la persona se despierta durante esta fase se puede sentir confusa. Es en ella en la que se dan también los terrores nocturnos, el sonambulismo y la micción. Las ondas predominantes son las delta. Durante las fases III y IV es cuando el cuerpo se encuentra en estado de profunda relajación y se dan los picos de segregación de la hormona del crecimiento [8].
- Etapa IV o sueño delta: suele ocupar un 20% aproximadamente del total del ciclo de sueño. Es la etapa en la que se está profundamente dormido y se descansa física y psíquicamente. Por tanto, es la que determina la calidad del sueño. Durante esta fase también es difícil despertarse, y el ritmo respiratorio y la presión arterial son muy bajos [8].
- Etapa REM (Rapid Eye Movement, Movimiento Rápido del Ojo) o etapa del sueño paradójico: ocupa un 25% del ciclo total de sueño, entre 15 y 30 minutos. Se denomina así por el movimiento constante de los globos oculares bajo los párpados. Esta fase se caracteriza por tener una alta actividad cerebral, similar a la que se tiene cuando la persona está despierta. El ritmo cardíaco y la presión arterial suben pero los músculos se encuentran bloqueados. En ella se tienen los sueños en forma de historia y, si nos despertamos, somos capaces de recordarlos. Las ondas predominantes son las theta [8].

#### 4.4. Electro - Cap

Electro-Caps son una técnica de aplicación de electrodos EEG. Están hechas de una tela elástica tipo spandex con electrodos empotrados de estaño puro unidos a la tela. Los electrodos en las tapas estándar se colocan según el método internacional 10-20 de colocación de electrodos [9]. En la Figura 4 se muestra el Electro - Cap y sus accesorios correspondientes para llevar a cabo su funcionamiento.



Figura 4: Electro Cap y sus componentes [9].

El Electro-Cap posee una terminal DB-25 donde se puede acceder a los distintos electrodos que se encuentran incorporados en el gorro, los cuales en algunos modelos es necesario conseguirlos por aparte. En el Cuadro 1 se presenta de manera detallada la distribución e interconexión de los pines con los electrodos [9].

Pin	Ubicación	Color
1	FP1	Café
2	F3	Rojo
3	C3	Naranja
4	P3	Amarillo
5	O1	Verde
6	F7	Azul
7	T3	Morado
8	T5	Gris
9	GND	Blanco
10	FZ	Negro
11	A1	-
12	OZ	-
13	-	-
14	FP2	Café
15	F4	Rojo
16	C4	Naranja
17	P4	Amarillo
18	O2	Verde
19	F8	Azul
20	T4	Morado
21	T6	Gris
22	CZ	Blanco
23	PZ	Negro
24	A2	-

Cuadro 1: Pines de salida del Electro-Cap [9].

## 4.5. Sistema Internacional 10-20

Este método determina la distancia entre los electrodos en forma porcentual, de ahí el nombre de 10-20 ya que los electrodos están separados entre sí por 10 o 20% de una distancia determinada. La razón de utilizar porcentajes en vez de longitud es el hecho de que el tamaño del cráneo varía de una persona a otra [10].

Para colocar los electrodos de acuerdo con el sistema 10-20 se requiere realizar las siguientes mediciones que sirven de referencia para obtener los porcentajes 10 y 20 . Se mide la distancia que hay entre el inión y el nasión, que representa 100 % en el plano sagital, y la distancia entre el trago de un lado y el trago del lado opuesto, que es 100% en sentido coronal. El electrodo central Cz se ubica a la mitad de la distancia entre el inión y el nasión (50%), y a la mitad de la distancia trago-trago (50%) [10].

Un 20% de la distancia inión-nasión, hacia adelante de Cz y en la línea media, corresponde a la ubicación del electrodo Fz, y 20% hacia atrás de Cz se ubica Pz. En dirección lateral a Cz se encuentran del lado derecho C4 y T4, al 20 y 40% de la distancia trago-trago, respectivamente, y del lado izquierdo están C3 y T3, también al 20 y 40% de la distancia

trago-trago. Como se puede ver, la ubicación de los electodos se determina con una letra y un número; las letras hacen referencia al lóbulo cerebral sobre el que se coloca el electrodo: F, P, T y O para frontal, parietal, temporal y occipital. La letra C se refiere al plano coronal y no corresponde a lóbulo alguno. La Figura 5 muestra una representación gráfica de como se colocan los electodos con este sistema [10].

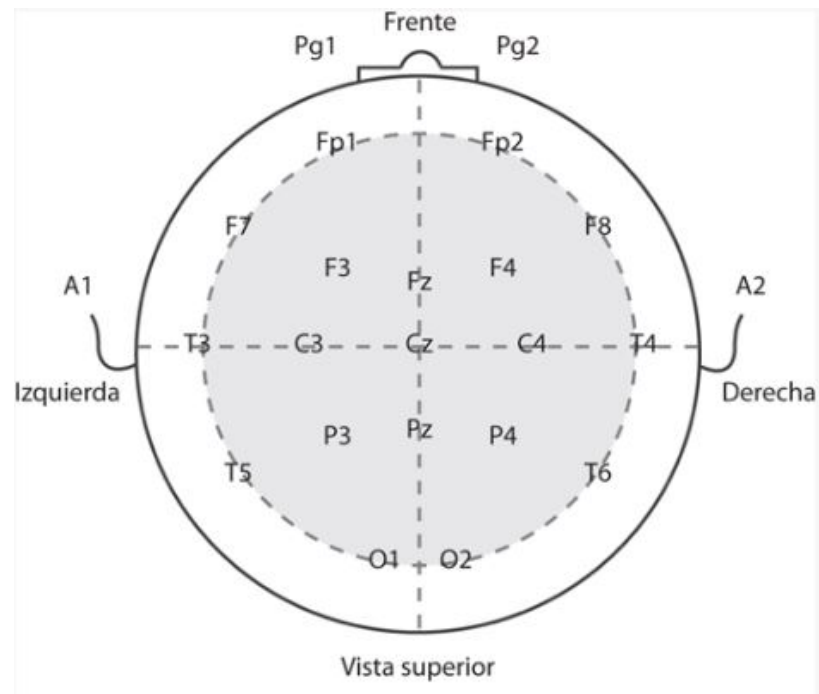


Figura 5: Ubicación de electodos según el Sistema Internacional 10-20 [10].

## 4.6. OpenBCI

OpenBCI es una plataforma/comunidad de interfaz cerebro-computadora de código abierto o mas conocida como Open Source ,de investigadores, ingenieros, científicos, etc. que tienen como objetivo que todos aquellos que tengan pasión e interés en las señales eléctricas del cuerpo y cerebro humano tengan a su alcance las herramientas necesarias para descubrir nuevas cosas acerca de su comportamiento. Esta comunidad provee de sistemas que pueden ser usados para monitorear la actividad cerebral (EEG), la actividad muscular (EMG) y la actividad cardiaca (ECG). Además de estos sistemas, esta comunidad también proveen de tutoriales y software para poder utilizar sus equipos correctamente y tener una interfaz amigable donde se puedan interactuar con las mediciones [11].

Específicamente para la parte de la EEG tienen a disposición cascos y gorros con electodos para monitorear las ondas cerebrales. Incluso tienen disponibles los diseños de sus cascos y placas para que una persona los pueda replicar por su cuenta como una opción más económica. Por otro lado, también cuentan con una interfaz de usuario a cual entre una de sus opciones permite observar las mediciones en tiempo real la cual se llama OpenBCI\_GUI, esta se puede observar en la siguiente figura [11].

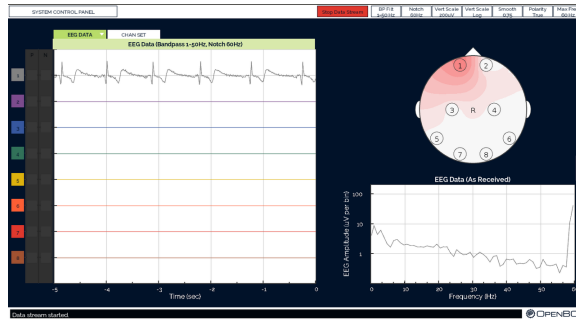


Figura 6: Interfaz gráfica de OpenBCI [11].

## 4.7. Cyton Board y Dongle

OpenBCI Cyton Board es una interfaz neuronal de 8 canales compatible con Arduino con un procesador de 32 bits. En esencia, la placa OpenBCI Cyton implementa el microcontrolador PIC32MX250F128B, lo que le brinda mucha memoria local y velocidades de procesamiento rápidas. La placa viene preinstalada con el gestor de arranque chipKIT y el último firmware OpenBCI [12].

Los datos se muestrean a 250Hz en cada uno de los ocho canales. OpenBCI Cyton Board se puede utilizar para tomar muestras de actividad cerebral (EEG), actividad muscular (EMG) y actividad cardíaca (ECG) [12]. La Cyton Board cuenta con los siguientes aspectos técnicos:

- 8 canales de entrada diferenciales, de alta ganancia y bajo ruido
- Compatible con electrodos activos y pasivos
- ADC ADS1299 de Texas Instruments
- Microcontrolador PIC32MX250F128B
- Módulo RFduino BLE para comunicación Bluetooth
- 24 bits de resolución en el canal de datos
- Ganancia programable: 1, 2, 4, 6, 8, 12, 24
- Voltaje de operación digital de 3.3V
- Voltaje de operación analógico de  $\pm 2.5V$
- Voltaje de entrada de 3.3 a 12V
- Acelerómetro LIS3DH
- Ranura para tarjeta micro SD
- 5 pines GPIO, 3 de los cuales pueden ser analógicos

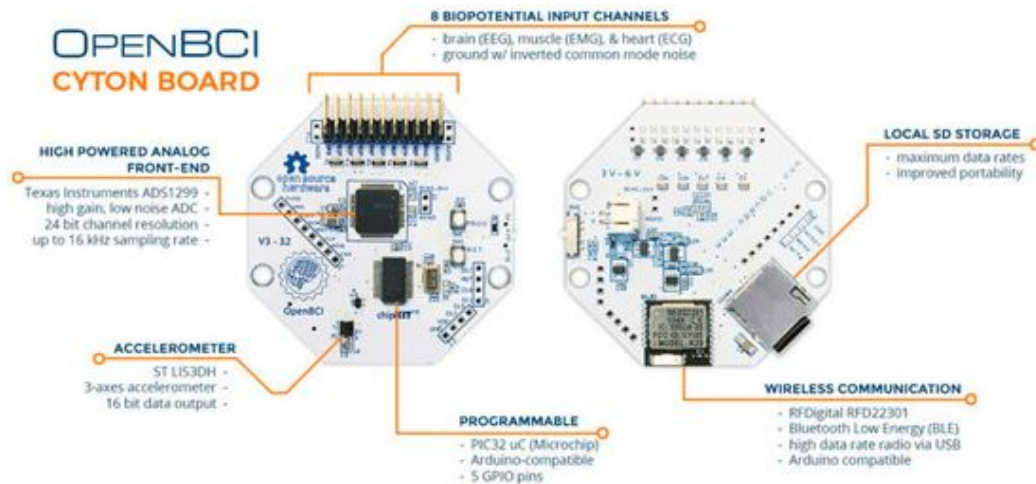


Figura 7: Cyton Board de OpenBCI [12].

En la figura anterior se puede apreciar ambos lados de la Cyton board y como es que se encuentran distribuidos sus componentes. Para poder realizar la comunicación entre la placa y la computadora se utiliza un dispositivo intermedio conocido como Dongle, dicho dispositivo se puede observar a detalle más adelante. Este módulo encarga de recibir los datos de la Cyton Board por medio de una conexión inalámbrica con Bluetooth, por lo cual ambos dispositivos cuentan con un módulo de Arduino llamado RFDuino BLE que permite establecer este tipo de comunicación. Con el RFDuino BLE se pueden conseguir mayores tasas de transmisión que con los módulos convencionales. Posteriormente el Dongle de OpenBCI transmite los datos recibidos por el puerto serial de la computadora o procesador con un baudaje de 115200 [12]. El Dongle cuenta con los siguientes aspectos técnicos:

- Módulo de radio RFD22301 de RFDigital
- Convertidor de USB a serie FT231X de FTDI
- Módulo RFDuino BLE para comunicación Bluetooth
- Alimentación vía USB (Únicamente)

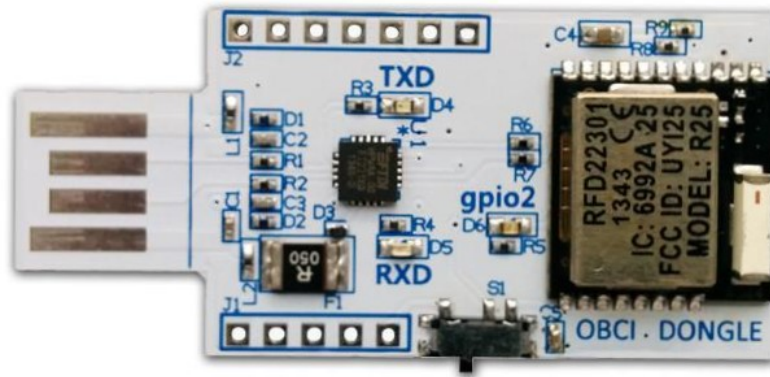


Figura 8: Dongle de OpenBCI [12]

## 4.8. Machine Learning

Un subconjunto de la inteligencia artificial (IA), el aprendizaje automático (*Machine Learning* - ML) es el área de la ciencia computacional que se centra en el análisis y la interpretación de patrones y estructuras de datos que hacen posible el aprendizaje, el razonamiento y la toma de decisiones sin interacción humana. Dicho de otro modo, el aprendizaje automático permite que el usuario alimente un algoritmo informático con una cantidad ingente de datos, a partir de los cuales el ordenador analiza toda la información y es capaz de tomar decisiones y hacer recomendaciones basándose únicamente en los datos introducidos. En el caso de identificar correcciones, el algoritmo puede incorporar esa información para mejorar la toma de decisiones futura [13]. El aprendizaje automático consta de tres partes:

- El algoritmo computacional, situado en el núcleo de la toma de determinaciones.
- Las variables y las funciones que conforman la decisión.
- El conocimiento base según el cual se sabe la respuesta que permite aprender al sistema.

Inicialmente, el modelo se alimenta de datos de parámetros para los que se conoce la respuesta. Se ejecuta entonces el algoritmo y se realizan ajustes hasta que el resultado del algoritmo coincide con la respuesta conocida. En este momento, se va aumentando la cantidad de datos que se introducen para ayudar al sistema a aprender y procesar un número mayor de decisiones computacionales [13].

Las redes neuronales pueden ser sometidas a distintos tipos de entrenamientos ya sea de aprendizaje supervisado o no supervisado en donde:

### 4.8.1. Aprendizaje supervisado

El aprendizaje supervisado se caracteriza por la presencia de un agente externo que controla el proceso de entrenamiento, estableciendo la respuesta que debería generar la red

a partir de una entrada determinada. De esta forma el supervisor comprueba la salida de la red, la compara con la salida deseada y, si existen diferencias, ajusta iterativamente los pesos hasta que su salida tiende a ser la deseada [14].

Este tipo de aprendizaje se puede subdividir en tres tipos los cuales son:

- **Aprendizaje por corrección de error**  
 Este tipo constituye el tipo de aprendizaje supervisado mas utilizado en la práctica. Su funcionamiento se basa en el ajuste de pesos de las conexiones de la red a partir de la diferencia entre los valores deseados y los obtenidos por el sistema, esto es, en función del error cometido en la salida [14].
- **Aprendizaje por refuerzo**  
 En este tipo de aprendizaje la tarea del supervisor se limita a indicar, mediante una señal de refuerzo (Éxito = +1 o fracaso = -1), si la salida obtenida por la red se ajusta o no a la deseada y, en función de ello, se procede al ajuste de los pesos utilizando un mecanismo basado en probabilidades, por lo que este tipo de aprendizaje también se denomina de “premio – castigo”.  
 Este aprendizaje resulta mas lento que el aprendizaje por corrección de error, al no indicar de forma expresa al sistema la salida que se desea ante una determinada entrada, por lo que el supervisor actúa mas como un critico que como un maestro [14].
- **Aprendizaje estocástico**  
 Este tipo de aprendizaje se basa en la introducción de cambios aleatorios en los valores de los pesos de la red, evaluando su efecto a partir de la salida deseada y de una distribución de probabilidad. En el aprendizaje estocástico suele establecerse una analogía en términos termodinámicos, asociando la red con un sólido físico que presenta cierto estado energético, por lo que el aprendizaje consiste en minimizar la energía del sistema a través del ajuste de los pesos. Para ellos, se realizan los cambios estocásticos de los valores de los pesos, determinándose la energía de la red tras estas modificaciones; si la energía es menor después del cambio, se acepta la modificación; si la energía no es menor, la aceptación del cambio depende de una distribución de probabilidad determinada y preestablecida [14].

Tipo de aprendizaje supervisado		Modelo de red
Aprendizaje por corrección de error	Off-line	Perceptrón
		Adaline/Madaline
		Backpropagation
		Brain-State-in-a-Box
Aprendizaje por refuerzo	On-line	Linear Reward Penalty
		Adaptive Reward Penalty
		Adaptive Heuristic Critic
Aprendizaje estocástico	Off-line	Boltzmann Machine
		Cauchy Machine

Figura 9: Principales modelos de RNAs con aprendizaje supervisado [14].

### 4.8.2. Aprendizaje no supervisado

Aprendizaje no supervisado Las redes con aprendizaje o supervisado o requieren de información externa para ajustar los pesos de las conexiones neuronales. En el aprendizaje no supervisado se presenta a la red un conjunto de patrones sin adjuntar la respuesta deseada por lo que la red, por medio del algoritmo de aprendizaje, estima la función de densidad probabilística que describe la distribución de patrones. Para ello, el sistema reconoce las peculiaridades, correlaciones o categorías presentes en el conjunto de entradas, extrayendo los rasgos o agrupando patrones según su similitud. Para que la red obtenga resultados de calidad, es necesario un cierto nivel de redundancia en las entradas presentadas al sistema, esto es, debe disponerse de un número suficiente de patrones de aprendizaje [14].

### 4.8.3. Red Neuronal Artificial

El primer modelo matemático de una neurona artificial, creado con el fin de llevar a cabo tareas simples, fue presentado en el año 1943 en un trabajo conjunto entre el psiquiatra y neuroanatomista Warren McCulloch y el matemático Walter Pitts [15].

La siguiente figura muestra un ejemplo de modelo neuronal con  $n$  entradas, que consta de:

- Un conjunto de entradas  $x_1, \dots, x_n$ .
- Los pesos sinápticos  $w_1, \dots, w_n$ , correspondientes a cada entrada.
- Una función de agregación.
- Una función de activación,  $f$ .
- Una salida,  $Y$ .

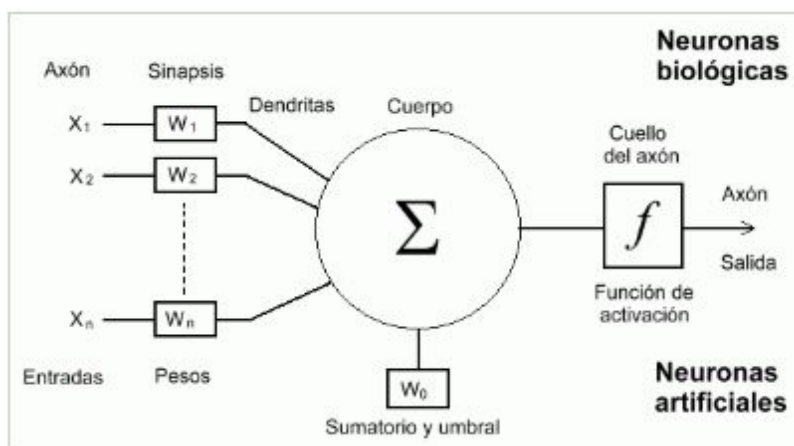


Figura 10: Modelo de la neurona artificial [15].

Las entradas son el estímulo que la neurona artificial recibe del entorno que la rodea, y la salida es la respuesta a tal estímulo. La neurona puede adaptarse al medio circundante y

aprender de él modificando el valor de sus pesos sinápticos, y por ello son conocidos como los parámetros libres del modelo, ya que pueden ser modificados y adaptados para realizar una tarea determinada [15].

En este modelo, la salida neuronal  $Y$  está dada por:

$$Y = f\left(\sum_i^N w_i x_i\right)$$

Ya habiendo definido y mostrado la estructura de una neurona artificial, ya se puede definir una Red Neuronal Artificial (RNA), la cual es un modelo matemático inspirado en el comportamiento biológico de las neuronas y en cómo se organizan formando la estructura del cerebro. El cerebro puede considerarse un sistema altamente complejo, donde se calcula que hay aproximadamente 100 mil millones (1011) neuronas en la corteza cerebral (humana) y que forman un entramado de más de 500 billones de conexiones neuronales [15].

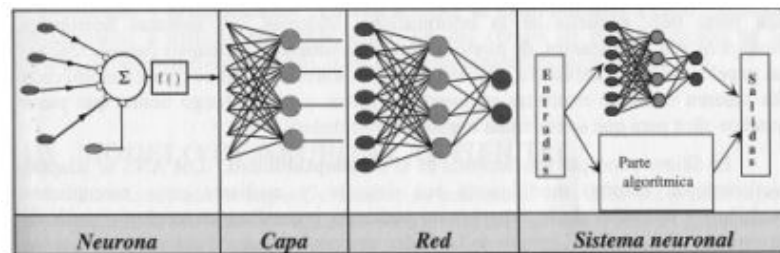


Figura 11: Modelo de la neurona artificial [14].

El objetivo de las redes neuronales es emular las siguientes funcionalidades de una red biológica:

- Procesamiento en paralelo: Las redes de neuronales son capaces de operar en paralelo un proceso producto de la gran cantidad de neuronas que intervienen en dicho proceso.
- Memoria distribuida: En estas redes no hay posiciones de memoria definidas si no que la información se distribuye a lo largo de la red, existiendo redundancia para evitar la pérdida de información.
- Adaptabilidad: Las redes de neuronales son capaces de aprender de la experiencia, siendo capaces de modificar su comportamiento en respuesta a su entorno.

Por estas funcionalidades y muchas otras más las redes neuronales son común en el proceso de reconocimiento de patrones. En dicho proceso se entrena una red neuronal para asignar clases determinadas a un conjunto de entradas. Dadas las características de la red, una vez entrenada es capaz de clasificar y aprender de patrones que nunca antes había visto.

#### 4.8.4. Máquinas de Vector Soporte (Support Vector Machines, SVMs)

El método de clasificación-regresión Máquinas de Vector Soporte (Vector Support Machines, SVMs) fue desarrollado en la década de los 90, dentro de campo de la ciencia computacional. Si bien originariamente se desarrolló como un método de clasificación binaria, su aplicación se ha extendido a problemas de clasificación múltiple y regresión. SVMs ha resultado ser uno de los mejores clasificadores para un amplio abanico de situaciones, por lo que se considera uno de los referentes dentro del ámbito de aprendizaje estadístico y machine learning. Las Máquinas de Vector Soporte se fundamentan en el Maximal Margin Classifier, que a su vez, se basa en el concepto de hiperplano [16].

##### Hiperplano y Maximal Margin Classifier

En un espacio  $p$ -dimensional, un hiperplano se define como un subespacio plano y afín de dimensiones  $P - 1$ . El término afín significa que el subespacio no tiene por qué pasar por el origen. En un espacio de dos dimensiones, el hiperplano es un subespacio de 1 dimensión, es decir, una recta. En un espacio tridimensional, un hiperplano es un subespacio de dos dimensiones, un plano convencional. Para dimensiones  $P > 3$  no es intuitivo visualizar un hiperplano, pero el concepto de subespacio con  $P - 1$  dimensiones se mantiene [16].

##### Clasificación binaria empleando un hiperplano

Cuando se dispone de  $n$  observaciones, cada una con  $p$  predictores y cuya variable respuesta tiene dos niveles, se pueden emplear hiperplanos para construir un clasificador que permita predecir a que grupo pertenece una observación en función de sus predictores. Este mismo problema puede abordarse también con otros métodos (regresión logística, LDA, árboles de clasificación...) cada uno con ventajas y desventajas [16].

Para facilitar la comprensión, las siguientes explicaciones se basan en un espacio de dos dimensiones, donde un hiperplano es una recta. Sin embargo, los mismos conceptos son aplicables a dimensiones superiores.

##### Support Vector Machines para más de dos clases

El concepto de hiperplano de separación en el que se basan los SVMs no se generaliza de forma natural para más de dos clases. Se han desarrollado numerosas estrategias con el fin de aplicar este método de clasificación a situaciones con  $k > 2$ -clases, de entre ellos, los más empleados son: one-versus-one, one-versus-all y DAGSVM [16].

##### One-versus-one

Supóngase un escenario en el que hay  $K > 2$  clases y que se quiere aplicar el método de clasificación basado en SVMs. La estrategia de one-versus-one consiste en generar un total de  $K(K - 1)/2$  SVMs, comparando todos los posibles pares de clases. Para generar una

predicción se emplean cada uno de los  $K(K - 1)/2$  clasificadores, registrando el número de veces que la observación es asignada a cada una de las clases. Finalmente, se considera que la observación pertenece a la clase a la que ha sido asignada con más frecuencia. La principal desventaja de esta estrategia es que el número de modelos necesarios se dispara a medida que aumenta el número de clases, por lo que no es aplicable en todos los escenarios [16].

### One-versus-all

En la clasificación de uno contra todos, para el conjunto de datos de instancias de clase  $N$ , tenemos que generar los modelos de clasificador  $N$ -binario. El número de etiquetas de clase presentes en el conjunto de datos y el número de clasificadores binarios generados deben ser iguales [17].

### DAGSVM

DAGSVM (Directed Acyclic Graph SVM) es una mejora del método one-versus-one. La estrategia seguida es la misma, pero consiguen reducir su tiempo de ejecución eliminando comparaciones innecesarias gracias al empleo de una directed acyclic graph (DAG). Supóngase un set de datos con cuatro clases (A, B, C, D) y 6 clasificadores entrenados con cada posible par de clases (A-B, A-C, A-D, B-C, B-D, C-D). Se inician las comparaciones con el clasificador (A-D) y se obtiene como resultado que la observación pertenece a la clase A, o lo que es equivalente, que no pertenece a la clase D. Con esta información se pueden excluir todas las comparaciones que contengan la clase D, puesto que se sabe que no pertenece a este grupo.

En la siguiente comparación se emplea el clasificador (A-C) y se predice que es A. Con esta nueva información se excluyen todas las comparaciones que contengan C. Finalmente solo queda emplear el clasificador (A-B) y asignar la observación al resultado devuelto. Siguiendo esta estrategia, en lugar de emplear los 6 clasificadores, solo ha sido necesario emplear 3. DAGSVM tiene las mismas ventajas que el método one-versus-one pero mejorando mucho el rendimiento [16].

#### 4.8.5. K Vecinos más cercanos

Los Vecinos Más Cercanos o K-NN (K Nearest Neighbors) es un método de clasificación supervisada (Aprendizaje, estimación basada en un conjunto de entrenamiento y prototipos) que sirve para estimar la función de densidad  $F(X/C_j)$  de las predictoras  $x$  por cada clase  $C_j$  [18].

Este es un método de clasificación no paramétrico. Que estima el valor de la función de densidad de probabilidad o directamente la probabilidad a posteriori de que un elemento  $x$  pertenezca a la clase  $C_j$  a partir de la información proporcionada por el conjunto de prototipos. En el proceso de aprendizaje no se hace ninguna suposición acerca de la distribución de las variables predictoras [18].

En el reconocimiento de patrones, el algoritmo K-NN es usado como método de clasificación de objetos basados en un entrenamiento mediante ejemplos cercanos en el espacio de los elementos K-NN es un tipo de “Lazy Learning”, donde la función se aproxima solo localmente y todo el computo es diferido a la clasificación [18].

#### 4.8.6. Características para análisis de señales

##### Slow Wave Activity

Se define como la potencia en la banda delta (0.5-4 Hz) de la señal EEG y se considera como el mejor indicador de la intensidad de sueño profundo. Para calcular esta característica, se ha obtenido primero el espectro de la señal mediante el periodograma modificado de Welch [19].

##### Zero Crossing

La característica Zero Crossing (ZC) informa del número de veces que la señal cambia de valor positivo a negativo en cada época tomando como referencia un umbral para evitar el efecto del ruido en las señales [20].

##### Sample Entropy

Sample Entropy (SE) es una modificación de Approximate entropy (ApEn), usada para evaluar la complejidad de una serie temporal fisiológica. Se define de la siguiente manera:

$$SampEn(m, r) = \lim_{n \rightarrow \infty} [\ln(\Phi^m(r) - \Phi^{m+1}(r))]$$

donde r es la tolerancia y, m, la dimensión (embedded dimension). [19].

##### Curtosis

La curtosis (o apuntamiento) es una medida de forma que mide cuán escarpada o achata está una curva o distribución. Este coeficiente indica la cantidad de datos que hay cercanos a la media, de manera que a mayor grado de curtosis, más escarpada (o apuntada) será la forma de la curva [21].

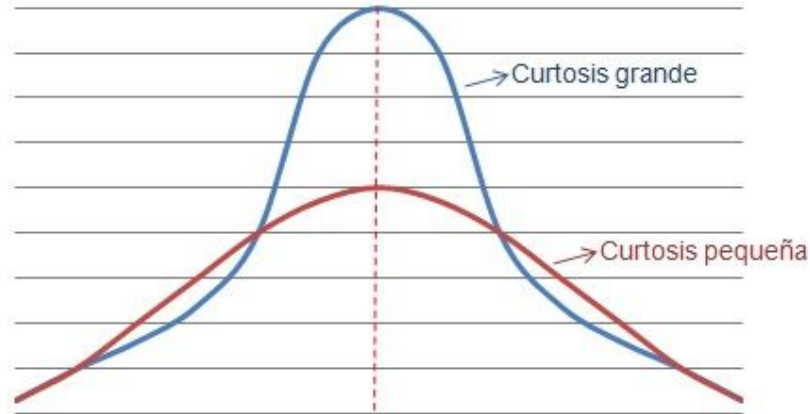


Figura 12: Comportamiento de la curtosis [21].

La curtosis se mide promediando la cuarta potencia de la diferencia entre cada elemento del conjunto y la media, dividido entre la desviación típica elevado también a la cuarta potencia [21]. La curtosis se puede obtener mediante la siguiente expresión:

$$k = \frac{E(X - \mu)^4}{\sigma^4}$$

### Sesgo

El sesgo es una medida de asimetría que indica el nivel en que los datos se reparten por encima y por debajo de la tendencia central. Si el sesgo es negativo, los datos se encuentran a la izquierda de la media. Si es positivo, la forma de la gráfica se reparte hacia la parte derecha de la media [22]. El sesgo se puede calcular utilizando la siguiente expresión:

$$s = \frac{E(X - \mu)^3}{\sigma^3}$$

### Valor eficaz

Debido a que la señal tiene valores positivos y negativos, la media aritmética puede tener valores muy próximos al cero. Por lo tanto, se añade como característica el valor eficaz, denominado en inglés RMS (root mean square). Se define como el valor cuadrático medio de una magnitud eléctrica. En el caso actual, se ha calculado el valor eficaz de manera discreta, para cada época de la señal [23].

$$\text{ValorEficaz} = \sqrt{\text{media}(\text{datos})^2}$$



---

## Obtención, análisis y clasificación de señales de Physionet

---

### 5.1. Obtención de los datos

Una vez realizada la fase de investigación y tener claros los objetivos que se deseaban lograr, era importante tener datos de prueba para poder realizar distintas pruebas y mediciones para poder poner en practica todo lo que se aprendió durante la fase de investigación. Una de las opciones que se encontró fue la base de datos de Physionet, la cual posee una amplia colección de datos y señales biomédicas. Una de las principales características por la que se escogió esta base de datos es porque posee una aplica colección de señales relacionadas a estudios del sueño, las cuales se encuentran clasificadas por expertos. Esta además cuenta con un explicación detalla del ambiente en el que se realizó la prueba para que sea posible poder replicar algunos estudios o mediciones.

The screenshot shows the PhysioNet website interface. At the top, there is a navigation bar with 'PhysioNet', 'Find', 'Share', 'About', 'News', 'Account', and a search box. Below the navigation bar, the page title is 'Sleep-EDF Database' with a 'Database' and 'Open Access' badge. The page content includes a 'When using this resource, please cite the original publication:' section with a citation for Kemp et al. (2000). Below that is a 'Please include the standard citation for PhysioNet:' section with a citation for Goldberger et al. (2000). The 'Expanded Database' section states that a greatly expanded version is now available. The 'Data Description' section explains that files have .rec and .hyp extensions and are in European Data Format (EDF). It also mentions the 'edfplus' format and the 'WFDB Software Package'. The 'Access' section includes an 'Access Policy' and a 'License (for files): Open Data Commons Attribution License v1.0'. The 'Discovery' section shows the DOI: https://doi.org/10.13026/C2C30 and a list of topics: sleep, multiparameter, polysomnogram.

Figura 13: Vista de la página de donde se obtuvieron los datos.

## Files

Total uncompressed size: 314.6 MB.

### Access the files

- [Download the ZIP file](#) (314.6 MB)
- Access the files using the Google Cloud Storage Browser [here](#). Login with a Google account is required.
- Access the data using Google Cloud "gsutil": `gsutil -m cp -r gs://sleep-edf-1.0.0.physionet.org DESTINATION`
- Download the files using your terminal: `wget -r -N -c -np https://physionet.org/files/sleep-edf/1.0.0/`

[Visualize waveforms](#)

Folder Navigation: <base>			
Name		Size	Modified
RECORDS		208 B	2012-02-25
SHA256SUMS.txt		1.3 KB	2019-02-20
sc4002e0.hyp		6.0 KB	2002-12-09
sc4002e0.rec		49.2 MB	2007-09-11
sc4012e0.hyp		6.1 KB	2002-12-09
sc4012e0.rec		49.6 MB	2007-09-11
sc4102e0.hyp		6.1 KB	2002-12-09
sc4102e0.rec		49.7 MB	2007-09-11
sc4112e0.hyp		5.9 KB	2002-12-09
sc4112e0.rec		48.4 MB	2007-09-11
st702j0.hyp		3.1 KB	2002-12-09
st702j0.rec		31.1 MB	2002-12-09
st705j0.hyp		3.0 KB	2002-12-09
st705j0.rec		30.3 MB	2002-12-09
st712j0.hyp		2.9 KB	2002-12-09
st712j0.rec		29.2 MB	2002-12-09
st713j0.hyp		2.8 KB	2002-12-09
st713j0.rec		27.2 MB	2002-12-09

Figura 14: Archivos de datos disponibles.

Ya teniendo acceso a las colecciones de datos se paso a descargar distintos sets de datos para realizar las diferentes pruebas pero se encontró con un problema el cual es que estas señales se trabajan con un formato .EDF el cual no es un formato que se pueda trabajar en Matlab. Para poder solucionar esto se exploraron diferentes alternativas hasta encontrar la herramienta llamada **Sleep EDFx Toolbox** la cual puede ser encontrada en Github y sirve para no solo convertir el formato de las señales EEG a un formato que matlab pudiera interpretar sino que también extrae las diferentes anotaciones de las señales. Dichas anotaciones son las que contienen la información de a que etapa del sueño pertenece cada época de información. Dicha información servirá para poder generar las etiquetas que se utilizaron para entrenar y validar los resultados de los clasificadores que se desarrollaron.

## 5.2. Análisis de las señales

Ya habiendo logrado obtener las señales de la base de datos se paso a buscar diferentes alternativas para realizar un análisis de las señales y de esta manera poder obtener la mayor información posible. Como se menciona en [19] la mejor manera de analizar las señales biomédicas es mediante la extracción de características en distintos dominios, no solo en el dominio del tiempo y frecuencia sino que también en dominios como el estadístico y el de potencia. Considerando esto se exploraron diferentes características con las cuales

poder trabajar y poder obtener una buena cantidad de información para brindarle a los clasificadores.

Para poder llevar a cabo esto se desarrolló un algoritmo en Matlab que tomara las señales que se obtuvieron anteriormente y extrajera las diferentes características de diferentes dominios para lograr formar los diferentes vectores que se implementaron en la clasificación. Para la primera versión del programa se trabajaron únicamente con tres características y uno de los canales disponibles para verificar el funcionamiento del programa. Estas características fueron:

- Mean Absolute value.
- Zero Crossing.
- Max Min Distance

Para poder realizar eso era importante conocer la frecuencia de muestreo con la que se tomaron los datos debido a que estas mediciones con señales EEG se realizan en intervalos de tiempo a los cuales se le denominan épocas o segundos por página. Conociendo esto se revisó la configuración con la que se realizaron las mediciones de Physionet y se encontró que se utilizó una frecuencia de muestreo de 100Hz y épocas de 30 segundos por lo que en el algoritmo se debían armar sets de 3000 datos y con estos obtener cada una de las características. Una vez verificada la funcionalidad de este algoritmo se agregaron tres canales más, junto con dos características más las cuales son :

- Slow wave activity
- Kurtosis

Ya habiendo incorporado lo anterior y verificado su funcionamiento generando los diferentes vectores para el entrenamiento de los diferentes clasificadores se pasó a realizar cada uno de los distintos clasificadores, los cuales para este proyecto fueron maquinas de vectores de soporte (SVM), Redes neuronales y K - Vecinos Más Cercanos(k - Nearest neighbors, KNN).

### 5.3. Clasificación de las señales

Ya habiendo logrado generar los vectores de características y las etiquetas se pasó a desarrollar y evaluar los distintos clasificadores mencionados anteriormente, los cuales fueron creados utilizando las herramientas de *deep learning* y *machine learning* de Matlab. En las primeras etapas de clasificación se trabajo con la primera versión del algoritmo de extracción de características para observar el comportamiento y verificar los resultados que se obtenían. Estos resultados se presentan en forma de matrices de confusión las cuales son herramientas que permiten la visualización de algoritmos de aprendizaje supervisado. También se obtuvieron los porcentajes de error de las distintas pruebas utilizando validación cruzada y con esto tener una mejor visualización del rendimiento de los clasificadores. Para las primeras pruebas se obtuvieron los siguientes resultados:



Figura 15: Matrices de confusión para la red neuronal con un canal de datos.

En la figura anterior se presentan las matrices de confusión para la red neuronal, en donde el eje Y representa las etiquetas obtenidas de la calcificación y el eje X son las etiquetas reales y en la columna de la derecha se presentan los porcentajes de acierto (Verde) y de error (Rojo) de cada etapa al igual que en la última fila. La intersección de ambas representa el acierto total de la clasificación al igual que su error total. Para esta prueba con este clasificador se obtuvo un acierto de 78% de error lo cual no es un mal resultado pero se buscaba obtener una mejor calificación.

Para el clasificador de vectores de soporte se trabajó con un kernel lineal, con lo que se obtuvieron los siguientes resultados:

**Confusion Matrix**

Output Class	1	78 10.3%	22 2.9%	6 0.8%	4 0.5%	4 0.5%	68.4% 31.6%
	2	5 0.7%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0.0% 100%
	3	7 0.9%	7 0.9%	217 28.7%	9 1.2%	13 1.7%	85.8% 14.2%
	4	4 0.5%	0 0.0%	10 1.3%	207 27.3%	0 0.0%	93.7% 6.3%
	5	10 1.3%	29 3.8%	17 2.2%	0 0.0%	107 14.1%	65.6% 34.4%
			75.0% 25.0%	0.0% 100%	86.8% 13.2%	94.1% 5.9%	85.6% 14.4%
		∧	∨	⊖	⊗	⊘	
		<b>Target Class</b>					

Figura 16: Matrices de confusión para la máquina de vectores de soporte con kernel Lineal con un canal de datos.

y al aplicar la validación cruzada se obtuvo un porcentaje de error de 19.6%.

Para el clasificador de k- nearest neighbor se trabajó con un valor de 10 para el número de vecinos, el cual fue el valor que mejores resultados presentó, los cuales fueron:

**Confusion Matrix**

1	67 8.9%	21 2.8%	5 0.7%	3 0.4%	5 0.7%	66.3% 33.7%
2	4 0.5%	7 0.9%	0 0.0%	0 0.0%	2 0.3%	53.8% 46.2%
3	13 1.7%	8 1.1%	219 28.9%	9 1.2%	22 2.9%	80.8% 19.2%
4	10 1.3%	0 0.0%	11 1.5%	208 27.5%	0 0.0%	90.8% 9.2%
5	10 1.3%	22 2.9%	15 2.0%	0 0.0%	96 12.7%	67.1% 32.9%
	64.4% 35.6%	12.1% 87.9%	87.6% 12.4%	94.5% 5.5%	76.8% 23.2%	78.9% 21.1%
	1	2	3	4	5	
	<b>Target Class</b>					

Figura 17: Matrices de confusión para k- nearest neighbor con un canal de datos.

y al aplicar la validación cruzada se obtuvo un porcentaje de error de 21.1 %.

Al igual que con las redes neuronales en ambos clasificadores se obtuvieron porcentajes de acierto mayor a 75 %, lo que dio la motivación a probar con más características y más canales de información por lo que se utilizó el algoritmo que extraía la información de cuatro canales y el resto de las características. Al utilizar la información generada por el segundo algoritmo se obtuvieron mejores resultados que los obtenidos al utilizar un solo canal, estos quedaron de la siguiente manera:



Figura 18: Matrices de confusión para la red neuronal.

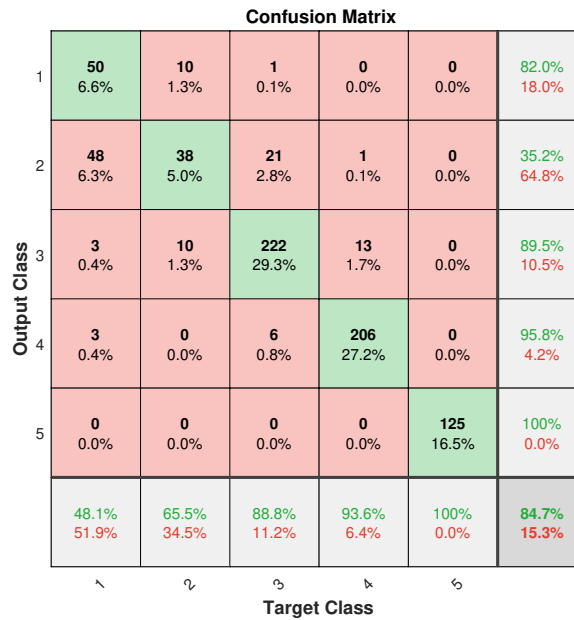


Figura 19: Matrices de confusión para la máquina de vectores de soporte con kernel Lineal.

**Confusion Matrix**

Output Class	1	96 12.7%	23 3.0%	2 0.3%	0 0.0%	1 0.1%	78.7% 21.3%
	2	3 0.4%	13 1.7%	1 0.1%	0 0.0%	0 0.0%	76.5% 23.5%
	3	2 0.3%	13 1.7%	227 30.0%	17 2.2%	5 0.7%	86.0% 14.0%
	4	3 0.4%	1 0.1%	9 1.2%	202 26.7%	0 0.0%	94.0% 6.0%
	5	0 0.0%	8 1.1%	11 1.5%	1 0.1%	119 15.7%	85.6% 14.4%
			92.3% 7.7%	22.4% 77.6%	90.8% 9.2%	91.8% 8.2%	95.2% 4.8%
		1	2	3	4	5	
		<b>Target Class</b>					

Figura 20: Matrices de confusión para k- nearest neighbor.

Como se puede apreciar en las figuras anteriores al utilizar más canales y extraer más características la clasificación mejora significativamente ya que en la red neuronal se obtuvo un 87% de efectividad, en la máquina de vectores de soporte al hacer la validación cruzada se obtuvo un porcentaje de error de 15.8520% y en k- nearest neighbor se obtuvo un porcentaje de error de 12.2853%. Ya viendo que se puede lograr una buena clasificación con los diferentes clasificadores utilizando cuatro canales y cinco características en diferentes dominios se decidió pasar a la clasificación en tiempo real de las señales.

---

## Obtención de señales con el Electro-Cap

---

### 6.1. Familiarización con la Cyton Board y el Electro - cap

Una vez obtenidas las distintas mediciones de la base de datos y haberlas utilizado en los distintos clasificadores y ver sus resultados fue necesario empezar a conocer y explorar el funcionamiento y comunicación entre la Cyton Board por lo que se utilizó la herramienta de OpenBCI\_GUI. La cual es propia de la empresa que fabrica la Cyton Board y ha sido utilizada en los distintos proyectos anteriores a este para verificar el funcionamiento del equipo y así saber qué tipos de datos se trabajan y la forma en la que este los obtiene.

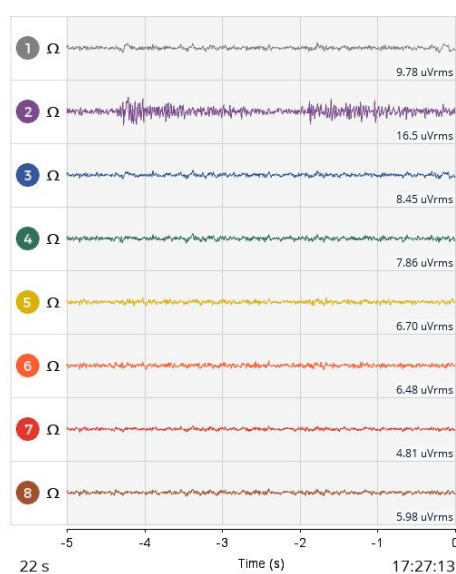


Figura 21: Captura 5 repeticiones cerrando el ojo izquierdo.

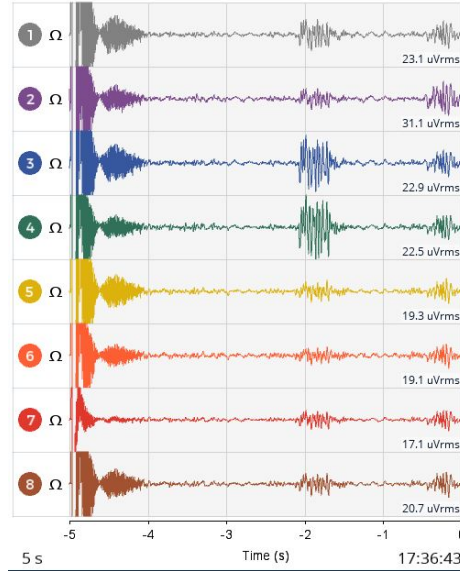


Figura 22: Captura 4 repeticiones remachando los dientes.

Como se puede apreciar en la figuras [21](#) y [22](#) se realizaron diferentes secuencias de gestos para ver la actividad en las distintas áreas de la cabeza. En cada uno de estos canales se puede apreciar la diferente actividad que se presentan junto con los diferentes niveles de voltaje que estos generan.

## 6.2. Conexión Matlab – Cyton Board

Ya con el conocimiento de cómo se debían de ver las señales de la Cyton se pasó a investigar la manera de obtener los datos mediante Matlab para poder procesarlos y así poderlos utilizar más adelante y al revisar la documentación dentro del sitio web de OpenBCI se encontró una herramienta de Python que permite la comunicación con la Cyton Board y así poder transmitirla a Matlab de manera inmediata utilizando *lab streaming layer*.

Ya teniendo la comunicación con Matlab se debía de procesar los datos, ya que estos se envían de manera “cruda” y no pueden ser utilizados de manera inmediata por lo que se le aplico dos tipos de filtros, los cuales fueron un pasa banda que permitiera señales entre 0.5 y 50 Hz y un filtro rechaza banda de 60 Hz para eliminar el desfase DC.

Ya con el filtrado hecho se realizaron pruebas de la misma manera que se realizaron con el programa de OpenBCI pero esta vez usando dicho programa para verificar el funcionamiento adecuado de los distintos canales. Uno de los ejercicios que se realizó para verificar el funcionamiento fue el de remachar los dientes en distintas ocasiones y así comparar los resultados, los cuales se pueden observar en las Figuras [23](#) y [24](#).

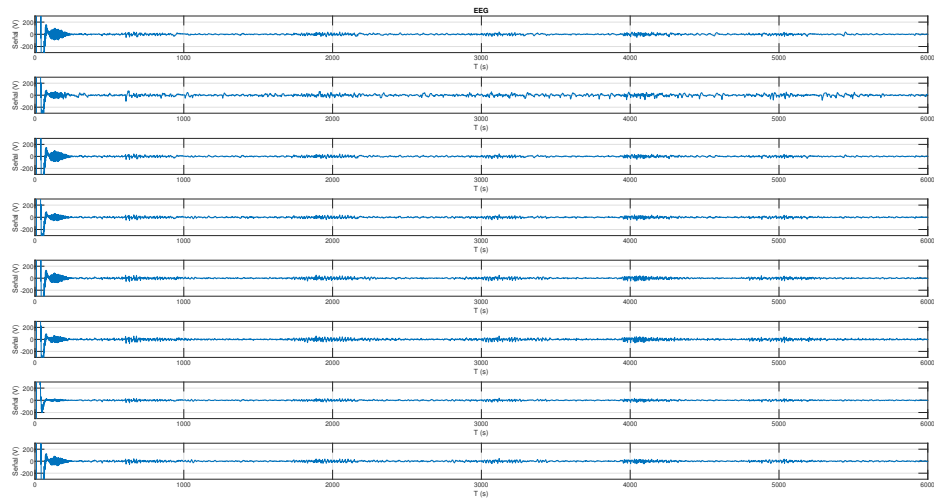


Figura 23: Captura de gestos utilizando el algoritmo desarrollado.

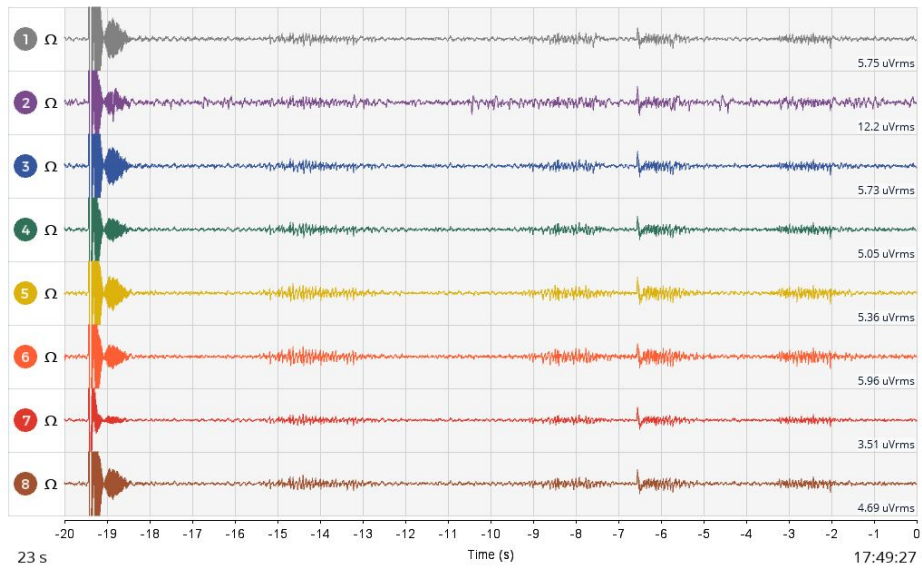


Figura 24: Captura de gestos de los 8 canales utilizando OpenBCI\_GUI.

Como se puede observar los resultados obtenidos por Matlab son bastante similares con los obtenidos en la interfaz de OpenBCI por lo que esto ayudo a corroborar el funcionamiento adecuado del sistema y así poder pasar a realizar las distintas pruebas.

### 6.3. Validación del algoritmo de obtención de las señales

Ya teniendo una respuesta adecuada a distintos gestos realizados se hizo un análisis más extenso para verificar que la obtención fuera lo más adecuada posible. Esto porque el objetivo del algoritmo es lograr captar las diferentes frecuencias que se experimentan en cada etapa del sueño. Para poder observar esto se realizó una grabación de aproximadamente 2 horas en donde se durmió con el casco para verificar que en efecto se obtuvieran dichas señales. Ya habiendo terminado la grabación se pasó a analizar las grabaciones y con ayuda del Dr. Jorge Donis se reconocieron los distintos cambios de frecuencias y en las siguientes Figuras se muestran algunas de estas frecuencias.

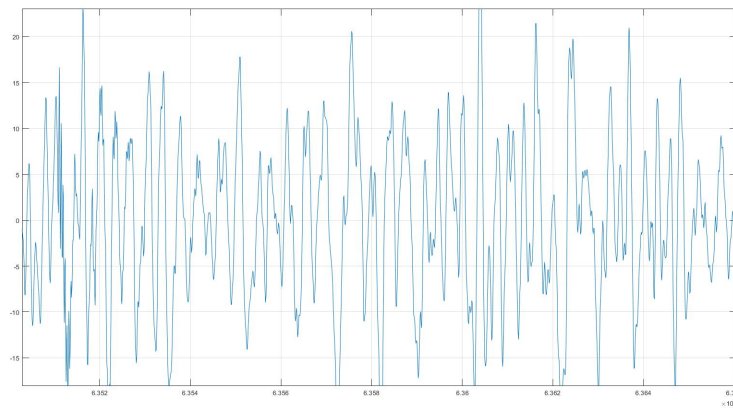


Figura 25: Ondas Gamma obtenidas con el programa en Matlab.

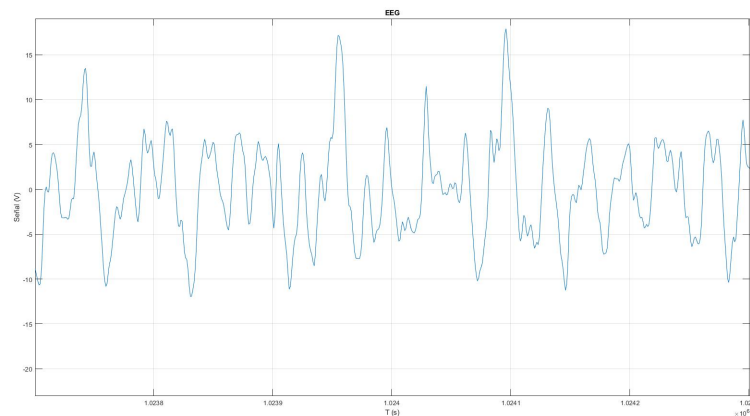


Figura 26: Ondas Beta obtenidas con el programa en Matlab.

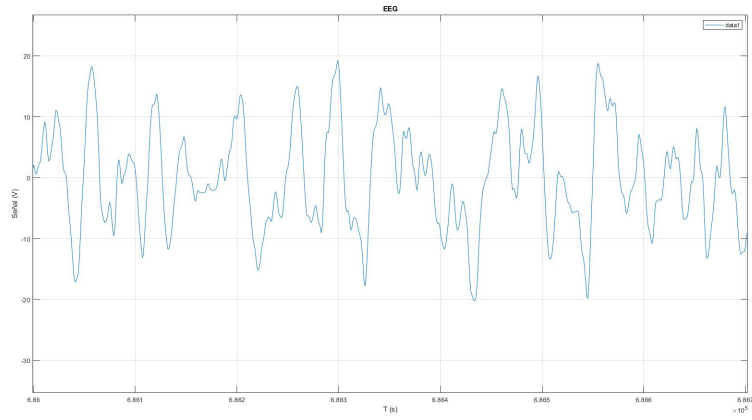


Figura 27: Ondas Delta obtenidas con el programa en Matlab.



---

## Análisis y clasificación de las señales del Electro-Cap

---

### 7.1. Validación de clasificación y extracción de características

Ya habiendo logrado capturar la información necesaria en las señales, se realizaron pruebas para validar que el algoritmo fuera capaz de extraer las diferentes características y poderlas aplicar en los diferentes clasificadores para verificar su funcionamiento adecuado. Estas pruebas consistieron en la realización de una secuencia de diferentes gestos similares a los que se aprecian en la Figura 23. Dicha secuencia constó de 4 gestos diferentes repetidos 9 veces cada uno en cierto orden a los cuales se les asignó una etiqueta diferente y así poder realizar la clasificación, dichas designaciones se pueden apreciar en la Tabla 2 y en la Figura 28 se pueden apreciar las diferentes señales obtenidas al realizar la secuencia de movimientos.

Gesto	Etiqueta
Sin actividad	1
Remachando dientes	2
Levantamiento de cejas	3
Cerrando ojo izquierdo	4
Girando la cabeza a la derecha	5

Cuadro 2: Etiquetas respectivas para cada gesto analizado.

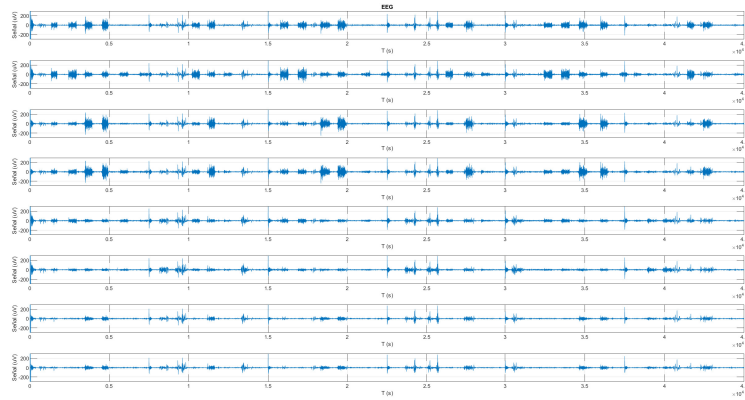


Figura 28: Captura de secuencia de gestos utilizando el algoritmo desarrollado.

Ya habiendo realizado las mediciones y obtenido las características, se pasó a realizar las pruebas correspondientes en la clasificación, Utilizando los tres clasificadores. Luego de aplicar la clasificación se obtuvo un porcentaje de error del 50 % para la maquina de vectores de soporte, 6.3 % para la red neuronal y un 56.3 % para el k- nearest neighbor. La clasificación puede apreciarse de mejor manera en las Figuras [29](#) [30](#) [31](#), en donde se muestran las matrices de confusión para cada uno de los distintos clasificadores.

**Confusion Matrix**

Output Class	1	7 14.6%	2 4.2%	0 0.0%	0 0.0%	0 0.0%	77.8% 22.2%
	2	3 6.3%	1 2.1%	0 0.0%	2 4.2%	0 0.0%	16.7% 83.3%
	3	8 16.7%	1 2.1%	4 8.3%	2 4.2%	0 0.0%	26.7% 73.3%
	4	2 4.2%	2 4.2%	2 4.2%	0 0.0%	0 0.0%	0.0% 100%
	5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	12 25.0%	100% 0.0%
			35.0% 65.0%	16.7% 83.3%	66.7% 33.3%	0.0% 100%	100% 0.0%
		Target Class					

Figura 29: Matriz de confusión para la secuencia de gestos utilizando la máquina de vectores de soporte.





Figura 31: Matriz de confusión para la secuencia de gestos utilizando la red neuronal.

Como se puede apreciar no se obtuvieron resultados tan elevados exceptuando la red neuronal que logro un buen nivel de clasificación con un 93.8% de acierto, debido a que el reconocimiento de este tipo de movimientos requiere de distintas consideraciones y procesamiento a las que se realizan para el análisis de las etapas del sueño. Por lo que el fin de esta prueba era poder verificar que las características que se seleccionaron y trabajaron anteriormente permitieran reconocer distintos patrones utilizando el algoritmo de obtención de señales desde el Electro Cap al igual que probar la efectividad en la clasificación.

## 7.2. Obtención de las señales de entrenamiento

Ya teniendo el algoritmo que obtenía las señales del Electro Cap y también extraía las características, se pasó a realizar las mediciones del sueño. Para poder realizar esto era

necesario obtener señales clasificadas por un experto, en donde se conocieran las condiciones en las que se realizaron para poderlas replicar con el casco. Para obtener estas señales se acudió a la Clínica de Asistencia Respiratoria S.A, la cual brindó el apoyo necesario para obtener las señales y clasificarlas, además de brindar consejos de como reconocer cada etapa del sueño para futuras mediciones y también poder proporcionar más señales de otros estudios para la realización de más pruebas con los clasificadores. Para la obtención de estas señales se trabajó con un estudio que constaba de cuatro canales EEG además de lectores de actividad ocular y lector de EMG para detectar la actividad de la barbilla. En la sección de anexos se podrán encontrar imágenes de cómo es que se colocaron los diferentes electrodos junto al equipo que se utilizó. En las siguientes figuras se podrá apreciar el comportamiento de cada una de las etapas del sueño durante la polisomnografía.

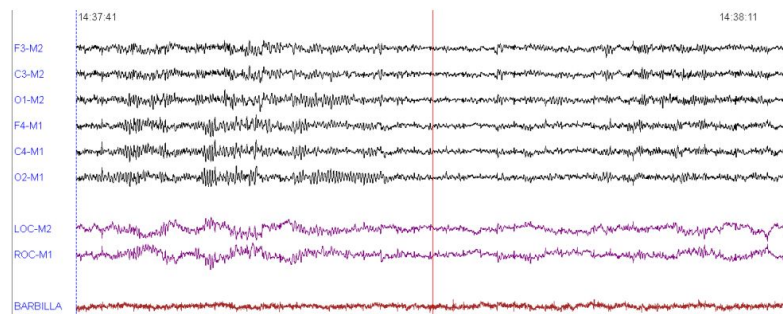


Figura 32: Estado despierto.

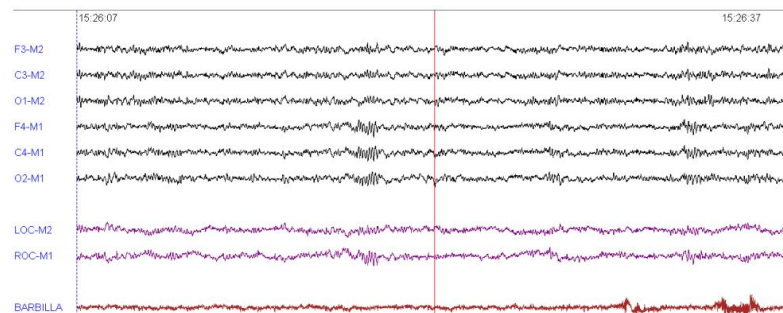


Figura 33: Etapa 1 del sueño.

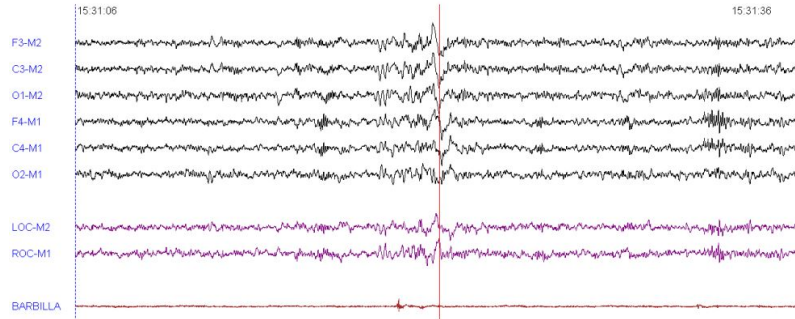


Figura 34: Etapa 2 del sueño.

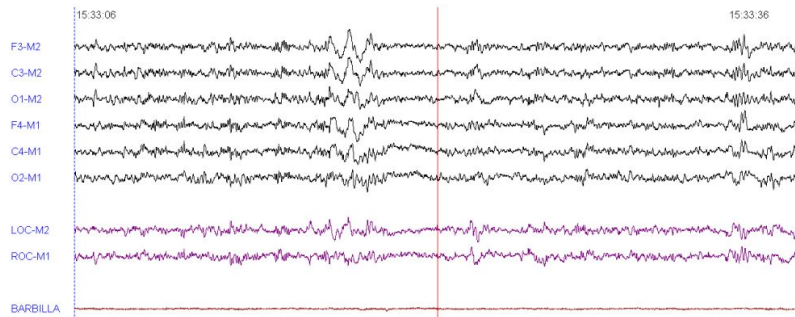


Figura 35: Etapa 3 del sueño.

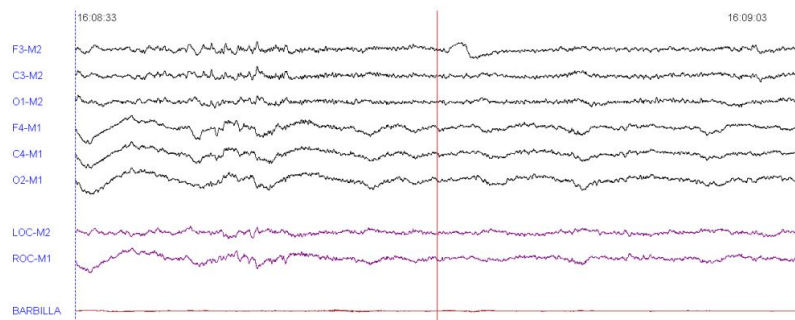


Figura 36: Etapa de sueño REM.

Según los comportamientos observados en las figuras anteriores es que los expertos logran reconocer cada una de las etapas ya que se analiza no solo los diferentes canales de EEG que se tienen, sino que también los dos canales de señales EOG y EMG los cuales ayudan a

reconocer las diferentes etapas del sueño en según la actividad que se tenga en cada uno de los canales.

### 7.3. Analisis y clasificación de señales

Ya habiendo conseguido las diferentes señales ya procesadas con las anotaciones indicadas por parte de la Clínica, se pasó a realizar la clasificación. Para poder realizar esto se le extrajeron las mismas características que a las señales obtenidas de la base de datos de Physionet, las cuales fueron:

- Mean Absolute value.
- Zero Crossing.
- Max Min Distance.
- Slow wave activity.
- Curtosis.

Ya habiendo extraído las diferentes características y armando los diferentes vectores necesarios, se pasó a realizar las pruebas con los tres clasificadores con los que se estuvo trabajando anteriormente y así poder seguir comparando los diferentes resultados obtenidos, para la primera prueba se utilizó las señales del estudio que se realizó, el cual tuvo una duración de dos horas y con esto se obtuvieron los siguientes resultados:

**Confusion Matrix**

Output Class	1	19 8.8%	10 4.6%	9 4.2%	13 6.0%	1 0.5%	36.5% 63.5%
	2	11 5.1%	15 6.9%	10 4.6%	1 0.5%	0 0.0%	40.5% 59.5%
	3	12 5.6%	11 5.1%	22 10.2%	11 5.1%	0 0.0%	39.3% 60.7%
	4	12 5.6%	3 1.4%	11 5.1%	31 14.4%	0 0.0%	54.4% 45.6%
	5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	14 6.5%	100% 0.0%
			35.2% 64.8%	38.5% 61.5%	42.3% 57.7%	55.4% 44.6%	93.3% 6.7%
		Target Class					

Figura 37: Matriz de confusión para SVM con duración de 2 horas.

**Confusion Matrix**

Output Class	1	13 6.0%	4 1.9%	9 4.2%	7 3.2%	2 0.9%	37.1% 62.9%
	2	8 3.7%	17 7.9%	5 2.3%	1 0.5%	0 0.0%	54.8% 45.2%
	3	13 6.0%	12 5.6%	30 13.9%	14 6.5%	0 0.0%	43.5% 56.5%
	4	20 9.3%	6 2.8%	8 3.7%	34 15.7%	11 5.1%	43.0% 57.0%
	5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 0.9%	100% 0.0%
			24.1% 75.9%	43.6% 56.4%	57.7% 42.3%	60.7% 39.3%	13.3% 86.7%
		Target Class					

Figura 38: Matriz de confusión para k- nearest neighbor con duración de 2 horas.



Figura 39: Matriz de confusión para Red neuronal con duración de 2 horas.

Como se puede apreciar en las figuras 37 y 38 la clasificación utilizando máquina de vectores de soporte y k- nearest neighbor no fue tan buena ya que se obtuvieron porcentajes de error de 53.2% y 55.6% debido a que no a que por la duración del estudio no se obtuvieron los suficientes datos para realizar una clasificación adecuada. En cambio la red neuronal presento resultados más favorables, esto se puede apreciar en la Figura 39 en donde se obtuvo un porcentaje de acierto del 89.8%, el cual es un valor aceptable para la clasificación considerando la poca duración del estudio.

Luego de observar los resultados obtenidos con el estudio anterior se pasó a realizar la clasificación utilizando un estudio que tuvo una duración aproximada de 10 horas para observar y comparar los resultados con los obtenidos con el estudio anterior. Estos resultados quedaron de la siguiente manera:

**Confusion Matrix**

Output Class	1	49 4.8%	7 0.7%	47 4.6%	9 0.9%	0 0.0%	43.8% 56.3%
	2	18 1.8%	11 1.1%	3 0.3%	0 0.0%	0 0.0%	34.4% 65.6%
	3	134 13.1%	27 2.6%	404 39.6%	20 2.0%	1 0.1%	68.9% 31.1%
	4	14 1.4%	0 0.0%	18 1.8%	115 11.3%	0 0.0%	78.2% 21.8%
	5	2 0.2%	0 0.0%	0 0.0%	0 0.0%	142 13.9%	98.6% 1.4%
			22.6% 77.4%	24.4% 75.6%	85.6% 14.4%	79.9% 20.1%	99.3% 0.7%
		1	2	3	4	5	
		Target Class					

Figura 40: Matriz de confusión para SVM con duración de 10 horas.

**Confusion Matrix**

Output Class	1	121 11.9%	22 2.2%	39 3.8%	0 0.0%	2 0.2%	65.8% 34.2%
	2	0 0.0%	3 0.3%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	3	77 7.5%	20 2.0%	417 40.8%	25 2.4%	11 1.1%	75.8% 24.2%
	4	17 1.7%	0 0.0%	16 1.6%	119 11.7%	3 0.3%	76.8% 23.2%
	5	2 0.2%	0 0.0%	0 0.0%	0 0.0%	127 12.4%	98.4% 1.6%
			55.8% 44.2%	6.7% 93.3%	88.3% 11.7%	82.6% 17.4%	88.8% 11.2%
		1	2	3	4	5	
		Target Class					

Figura 41: Matriz de confusión para k- nearest neighbor con duración de 10 horas.



Figura 42: Matriz de confusión para Red neuronal con duración de 10 horas.

Para estas pruebas la clasificación mejoró en los tres clasificadores ya que como se puede apreciar en las figuras 40, 41 y 42 en donde se obtuvieron porcentajes de error de 29.4%, 22.9% y 8.9% respectivamente donde claramente la red neuronal logro una mejor clasificación en comparación a los otros dos clasificadores. Ya que a pesar que se redujeron los porcentajes de error, no lograron obtener un mejor resultado. Entonces ya habiendo realizado dos pruebas de diferentes se pudo apreciar que para poder lograr una mejor clasificación en las etapas era necesario tener mediciones de idealmente 10 horas para poder garantizar que al momento de utilizar el Electro Cap y el usuario duerma, el algoritmo sea capaz de realizar la clasificación en tiempo real de manera adecuada.

Considerando lo anterior para poder realizar los experimentos en tiempo real se tener un clasificador previamente entrenado y de esta manera el algoritmo irá procesando las señales provenientes del Electro Cap, extrayendo las diferentes características mencionadas anteriormente para luego ser enviadas al clasificador y que este regrese a que etapa pertenece dicho intervalo de tiempo. Este proceso se muestra de mejor manera en el siguiente esquema:

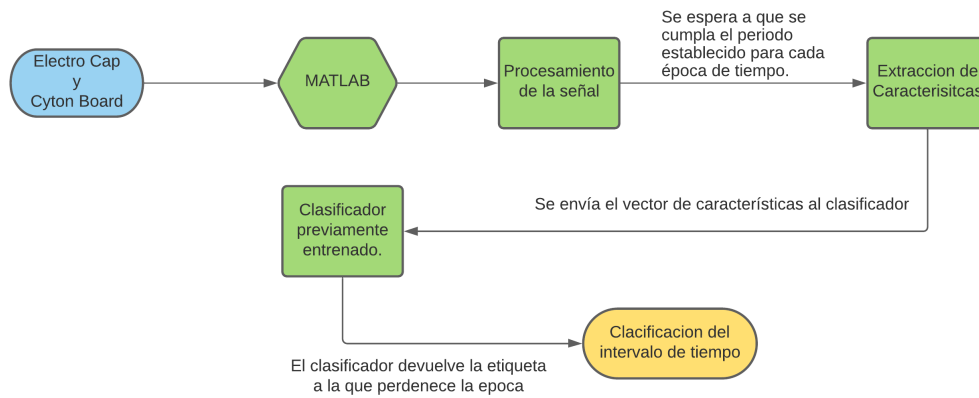


Figura 43: Esquema del proceso para la clasificación de las señales.

En el diagrama anterior se muestra el proceso en que se sigue para realizar la clasificación en tiempo real en donde el Electro cap se encuentra enviando datos constantemente a Matlab y este procesa las señales y separando en intervalos de tiempo en donde usualmente se trabajan de 30 segundos por época. Luego utilizando dichas épocas se extraen las características y se forma el vector para poderlo enviar al clasificador y así este devuelva a que etapa pertenece dicha época.

---

## Almacenamiento de las señales

---

Ya que parte de este proyecto requería de una manera de poder almacenar todas las mediciones que se fueran realizando durante el desarrollo del proyecto al igual que las señales que obtuvieron en la ida a la clínica y los resultados de la clasificación de las señales obtenidas en Physionet. Debido a todo esto se vio en la necesidad de desarrollar una base de datos en donde no solo se almacenaran los valores crudos o procesados de las mediciones, sino que también las diferentes características obtenidas para así poder ser utilizadas en futuros análisis.

### 8.1. Opciones de bases de datos

Se evaluaron distintas bases de datos, en busca de una opción que permitiera realizar el almacenamiento de manera local o en algún servidor en línea.

#### 8.1.1. FireBase

La primera opción que se encontró fue el servicio de almacenamiento FireBase de Google, el cual permite diferentes acciones de almacenamiento en tiempo real en servidores online, con la facilidad de poder acceder a ella a partir de diferentes herramientas tales como la programación en Python, Java y Node además de poder enlazar una página web a ella para hacerla más interactiva y útil.

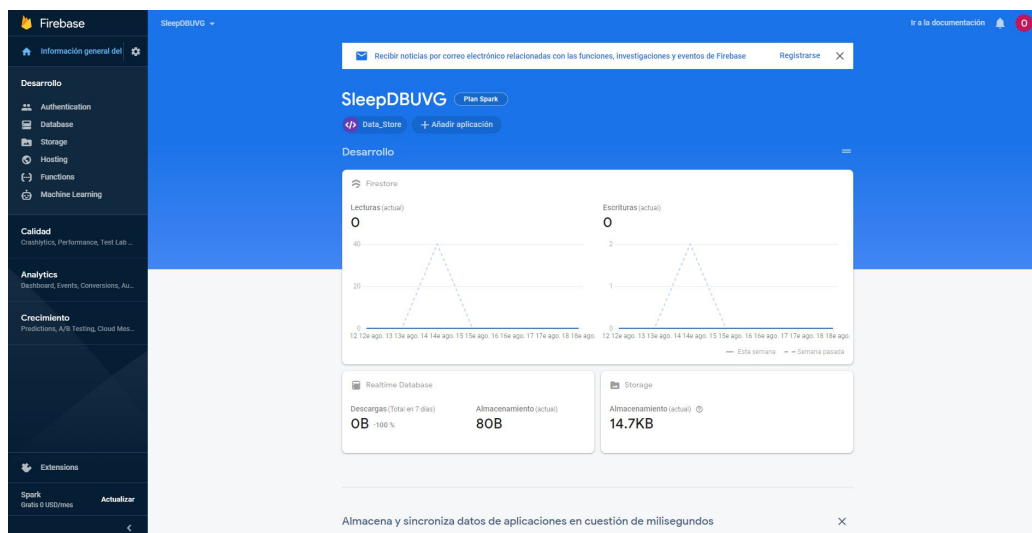


Figura 44: Página de inicio de FireBase.

En la Figura 44 se puede apreciar la pagina principal de la base de datos en donde se muestra toda la información de consultas y la cantidad de data que se encuentra almacenada junto con la cantidad de consultas realizadas.

Uno de los problemas que se encontró al realizar distintas pruebas con esta base de datos fue que al cargar los diferentes datos de las mediciones. Al ser vectores demasiado grandes no se almacenaban todos los datos de manera correcta ya que por el tamaño la página los acortaba y solo representaba el tamaño con el uso de tres puntos y esto afectaba la descarga de estos datos ya que estos se descargaban de manera resumida y con los tres puntos. Entonces se perdía casi toda la información, por esta razón fue que no se decidió utilizar esta plataforma a pesar de las diferentes ventajas que esta ofrece.

### 8.1.2. MySQL

La segunda opción que se exploró fue MySQL, el cual es la herramienta más utilizada para la realización de base de datos y ofrece una amplia variedad de beneficios con respecto a su construcción y uso. Uno de los beneficios es que se puede realizar la estructuración de la base de datos de manera rápida y fácil. Además de que ofrece la capacidad de ingresar los datos en distintos formatos tales como Varchar, JSON, etc. . .

Para esta herramienta se tomó la decisión de realizar el almacenamiento de manera local, con la posibilidad de poder adecuarla para poder ser almacenada online en el futuro. Lo cual en comparación con la herramienta anterior resultó más sencilla la comunicación y la manera en la que se almacenan los datos. Al tener resultados satisfactorios respecto al rendimiento y posibilidades se decidió utilizar esta herramienta y así poder continuar a almacenar las distintas pruebas.

## 8.2. Desarrollo de la base de datos.

Ya habiendo decidido la herramienta que se utilizaría se comenzó a analizar la manera de almacenar las amplias cadenas de datos de manera eficiente y funcional, ya que no todas las alternativas son funcionales para este tipo de datos. Ya que previamente se había decidido que el almacenamiento sería local. Se optó en almacenar la dirección de cada archivo con los datos para evitar realizar operaciones computacionalmente pesadas debido a las largas cadenas de datos que se trabajan y así a la hora de realizar consultas solo es necesario obtener la dirección para poder llegar a los datos y así poder leer los datos de manera mas rápida y eficiente.

Ya con todo lo anterior definido se pasó a realizar la base de datos la cual se dividió en dos tablas. Una tabla dedicada a almacenar los datos de las mediciones del sueño y otra tabla para poder almacenar los datos de mediciones adicionales. Como distintos gestos que se utilizaron para validar el funcionamiento de la conexión entre la cyton board y Matlab, además de que estos pueden ser utilizados para distintas aplicaciones y proyectos.

Mov_ID	Mov_F1	Mov_F2	Mov_F3	Mov_F4
1				
Remachando los dientes	C:/Users/Public/SleepDatabase/Remachando los dient...	C:/Users/Public/SleepDatabase/Remachando los dient...	C:/Users/Public/SleepDatabase/Remachando los dient...	C:/Users/Public/SleepDatabase/Remachando los dient...
Levantamiento de cejas	C:/Users/Public/SleepDatabase/Levantamiento de cej...	C:/Users/Public/SleepDatabase/Levantamiento de cej...	C:/Users/Public/SleepDatabase/Levantamiento de cej...	C:/Users/Public/SleepDatabase/Levantamiento de cej...
Girando la cabeza a la derecha	C:/Users/Public/SleepDatabase/Girando la cabeza a ...	C:/Users/Public/SleepDatabase/Girando la cabeza a ...	C:/Users/Public/SleepDatabase/Girando la cabeza a ...	C:/Users/Public/SleepDatabase/Girando la cabeza a ...
Cerrando ojo izquierdo	C:/Users/Public/SleepDatabase/Cerrando ojo izquier...	C:/Users/Public/SleepDatabase/Cerrando ojo izquier...	C:/Users/Public/SleepDatabase/Cerrando ojo izquier...	C:/Users/Public/SleepDatabase/Cerrando ojo izquier...

Figura 45: Vista del almacenamiento de gestos pt1.

Mov_F5	Mov_Filt	Mov_Descrip	Mov_Raw
C:/Users/Public/SleepDatabase/Remachando los dient...	C:/Users/Public/SleepDatabase/Remachando los dient...	4 repeticiones del movimiento con duracion de 3 se...	C:/Users/Public/SleepDatabase/Remachando los dient...
C:/Users/Public/SleepDatabase/Levantamiento de cej...	C:/Users/Public/SleepDatabase/Levantamiento de cej...	4 repeticiones del movimiento con duracion de 2 se...	C:/Users/Public/SleepDatabase/Levantamiento de cej...
C:/Users/Public/SleepDatabase/Girando la cabeza a ...	C:/Users/Public/SleepDatabase/Girando la cabeza a ...	4 repeticiones del movimiento con duracion de 3 se...	C:/Users/Public/SleepDatabase/Girando la cabeza a ...
C:/Users/Public/SleepDatabase/Cerrando ojo izquier...	C:/Users/Public/SleepDatabase/Cerrando ojo izquier...	4 repeticiones del movimiento con duracion de 3 se...	C:/Users/Public/SleepDatabase/Cerrando ojo izquier...

Figura 46: Vista del almacenamiento de gestos pt2.

En las figuras 45 y 46 se muestran algunos gestos que se almacenaron y de cómo es que se distribuyeron cada uno de los vectores de datos. Donde las columnas F1 - F5 representan cada una de las características o features que se le extrajeron a las señales, las columnas de Raw y Filt almacenan los datos en bruto y los datos filtrados y por ultimo las columnas de ID y Descrip son las columnas que almacenan la información de qué gesto se realizó y una descripción mas detallada de las pruebas.



- El algoritmo que se desarrolló para la obtención de datos mostró ser capaz de obtener las diferentes ondas que se experimentan a lo largo del ciclo del sueño.
- Las redes neuronales demostraron ser un clasificador bastante efectivo ya que mostró los resultados más elevados en la clasificación utilizando solamente señales EEG con diferentes rangos de duración.
- Para la clasificación en tiempo real es necesario poder tener señales de entrenamiento con una amplia duración para poder tener una mejor clasificación.
- Las características utilizadas junto con la utilización de 4 canales de información demostraron un buen rendimiento en la clasificación de las etapas del sueño.
- MySQL demostró ser una muy buena herramienta para la implementación y desarrollo de bases de datos ya que permite la interacción con diferentes lenguajes de programación al momento de cargar y descargar los datos.
- El algoritmo de almacenamiento de señales demostró ser eficiente al momento de almacenar las amplias cadenas de datos además de permitir una posible transición al almacenamiento en línea de manera sencilla.
- Para verificar la efectividad de la clasificación en tiempo real es necesario realizar las pruebas en compañía de un especialista en el área para poder ir realizando las anotaciones correspondientes y así poder obtener las matrices de confusión o porcentajes de acierto de los clasificadores para validar su funcionamiento.



- Considerando que la conexión entre el Electro Cap y Matlab funcionó de manera adecuada, cumpliendo con los requerimientos necesarios, es necesario darle un seguimiento a las nuevas herramientas que vayan surgiendo que permitan mejorar la obtención y procesamiento de los datos o señales
- Para futuras fases de este proyecto es importante considerar la implementación de lectores de señales EOG y EMG debido a que estas señales no solo brindarían más información útil para realizar la clasificación de las etapas, sino que también son parte de los estándares utilizados en los estudios del sueño junto a las señales EEG. por lo que sería importante incluirlas para tener una buena referencia según los estándares establecidos.
- Para futuras fases es importante lograr validar de mejor manera la clasificación en tiempo real del clasificador por lo que sería útil implementar una herramienta de visualización de las señales en tiempo real y que esta permita realizar anotaciones, de esta manera ir generando las etiquetas para verificar que se hayan clasificado bien las etapas.
- Sería importante poder unificar la clasificación de las etapas del sueño con la generación de los pulsos biaurales en futuras fases para poder realizar pruebas más detalladas y así poder ir explorando los diferentes resultados.



- 
- [1] A. informationDisclosure statementNo potential conflict of interest was reported by the authors., *Sleep of professional athletes: Underexploited potential to improve health and performance*. dirección: <https://www.tandfonline.com/doi/full/10.1080/02640414.2016.1184300>.
- [2] *What are Brainwaves?* Dirección: <https://brainworksneurotherapy.com/what-are-brainwaves>.
- [3] C. A. Vinueza Bonilla y col., “Sistema para monitoreo de ondas cerebrales en estudios de pulsos binaurales con ritmo theta sobre los procesos cognitivos y emocionales”, B.S. thesis, Quito: USFQ, 2016, 2016.
- [4] *How Do Binaural Beats Work?* Dirección: [https://www.binauralbeatsmeditation.com/the-science/#The\\_Discovery\\_of\\_Binaural\\_Beats](https://www.binauralbeatsmeditation.com/the-science/#The_Discovery_of_Binaural_Beats).
- [5] R. Villar-Documet, *Accueil*. dirección: <https://www.rvd-psychologue.com/es/neuroestimulacion-sonidos-binaurales.html>.
- [6] *Stages of Sleep - Non-REM and REM Sleep Cycles*. dirección: <https://www.tuck.com/stages/>.
- [7] N. M. Molina Culqui, “Sistema de monitoreo de las ondas cerebrales (electroencefalograma o EEG) presentes en el sueño: Análisis de frecuencia y coherencia del EEG en ambos hemisferios”, B.S. thesis, Quito, 2016., 2016.
- [8] E. Quirantes, *Medicina General*. dirección: <https://salud.ideal.es/general-61/2857-los-ciclos-del-sue%C3%B1o.html?ref=https://www.google.com/>.
- [9] *Home*. dirección: <https://electro-cap.com/index.cfm/caps/>.
- [10] F. G. N. E., *Manual de laboratorio de fisiología*. McGraw-Hill Interamericana, 2015.
- [11] *Third-Party Hardware · OpenBCI Documentation*. dirección: <https://docs.openbci.com/docs/05ThirdParty/ThirdPartyLanding>.
- [12] *Cyton Biosensing Board (8-channels)*. dirección: <https://shop.openbci.com/collections/frontpage/products/cyton-biosensing-board-8-channel?variant=38958638542>.

- [13] NetApp, *¿Qué es el aprendizaje automático (ML) y por qué es importante?*, ene. de 1970. dirección: <https://www.netapp.com/es/info/what-is-machine-learning-ml.aspx>.
- [14] R. F. López y J. M. F. Fernández, *Las redes neuronales artificiales*. Netbiblo, 2008.
- [15] F. S. Caparrini, *Redes Neuronales: una visión superficial*. dirección: <http://www.cs.us.es/~fsancho/?e=72>.
- [16] J. A. R. j.amatrodrigo@gmail.com, *Máquinas de Vector Soporte (Support Vector Machines, SVMs)*, abr. de 2017. dirección: [https://www.cienciadedatos.net/documentos/34\\_maquinas\\_de\\_vector\\_soporte\\_support\\_vector\\_machines](https://www.cienciadedatos.net/documentos/34_maquinas_de_vector_soporte_support_vector_machines).
- [17] A. Band, *Multi-class Classification-One-vs-All amp; One-vs-One*, mayo de 2020. dirección: <https://towardsdatascience.com/multi-class-classification-one-vs-all-one-vs-one-94daed32a87b>.
- [18] M. Rodriguez e I. Proyecto, “Comparación de métricas de distancia en el algoritmo K-Vecinos Más Cercanos para el problema de Reconocimiento Automático de Dígitos Manuscritos”, *Pontificia Universidad Católica de Valparaíso, Facultad de Ingeniería, Escuela de Ingeniería Informática*, 2015.
- [19] C. Saló Casajuana, *Clasificación de las fases del sueño utilizando señales EEG*, ene. de 2019.
- [20] S. Mallat, “Zero-crossings of a wavelet transform”, *IEEE Transactions on Information theory*, vol. 37, n.º 4, págs. 1019-1033, 1991.
- [21] B. R. Serra, E. dice: R. d. 2. septiembre, R. dice: F. dice: R. d. 4. julio, J. dice: M. dice: R. d. 9. marzo, A. dice: y et al., *Curtosis*, mar. de 2020. dirección: <https://www.universoformulas.com/estadistica/descriptiva/curtosis/>.
- [22] *Sesgo-Y- Curtosis - Estadística - IPN*. dirección: <https://www.studocu.com/es-mx/document/instituto-politecnico-nacional/estadistica/apuntes/sesgo-y-curtosis/2757897/view>.
- [23] K. Nakashima, “Valor médio e eficaz”, *Setembro, 2013a. Universidade Federal de Itajubá*, 2007.